МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное
учреждение высшего образования
«Воронежский государственный технический университет»

Кафедра систем автоматизированного проектирования
и информационных систем

МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ПО ВЫПОЛНЕНИЮ ЛАБОРАТОРНЫХ РАБОТ ПО ДИСЦИПЛИНЕ «ИНФОРМАЦИОННАЯ БЕЗОПАСНОСТЬ И ЗАЩИТА ИНФОРМАЦИИ»

для обучающихся по направлению 09.03.02 «Информационные системы и технологии», профиль «Технологии искусственного интеллекта» очной формы обучения

Воронеж 2024

УДК 681.3

Составители: А.В. Питолин, Б.Н. Тишуков

Методические рекомендации по выполнению лабораторных работ по дисциплине «Информационная безопасность и защита информации» для обучающихся по направлению "Информационные системы и технологии", профиль "Технологии искусственного интеллекта"
очной формы обучения / ФГБОУ ВО «Воронежский государственный технический университет»; сост.: А.В. Питолин, Б.Н. Тишуков. – Воронеж: Изд-во ВГТУ, 2024. – 24 с.

Приводится описание выполнения лабораторных работ по курсу «Информационная безопасность и защита информации» для обучающихся по направлению 09.03.02 «Информационные системы и технологии», профиль «Технологии искусственного интеллекта» очной формы обучения

УДК 681.3

Рецензент - к.т.н., доцент Королев Е.Н.

Рекомендовано методическим семинаром кафедры САПРИС и методической комиссией ФИТКБ Воронежского государственного технического университета в качестве методических материалов

Лабораторная работа № 1

Поточные шифры. Моделирование работы 8-ми (16-ти) разрядного скремблера

Цель работы: Исследовать побитное непрерывное шифрование данных. Ознакомиться с кодированием информации при помощи скремблера.

Краткие теоретические сведения

Шифр Вернама можно считать исторически первым поточным шифром. Так как поточные шифры осуществляют поэлементное шифрование потока данных без задержки в криптосистеме, их важнейшим достоинством является высокая скорость преобразования, соизмеримая со скоростью поступления входной информации. Таким образом, обеспечивается шифрование практически в реальном масштабе времени вне зависимости от объема и разрядности потока преобразуемых данных.

Поскольку каждый знак открытого текста в архитектуре поточного шифра не рассматривается как независимая единица, то возникает вопрос о том, что будет происходить, если при передаче сообщения в него вкрадется ошибка. Будет ли она распространяться дальше, и если да, то насколько далеко? Больше всего нас, как программистов, может заинтересовать вопрос, будут ли все последующие символы расшифрованы правильно, если один из них был изменен в цепочке шифротекста.

Именно по этим признакам и можно создать вполне естественную классификацию поточных шифров: на синхронные и самосинхронизирующиеся.

В синхронных поточных шифрах (рис. 1) гамма формируется независимо от входной последовательности, каждый элемент (бит, символ, байт) которой таким образом шифруется независимо от других элементов.

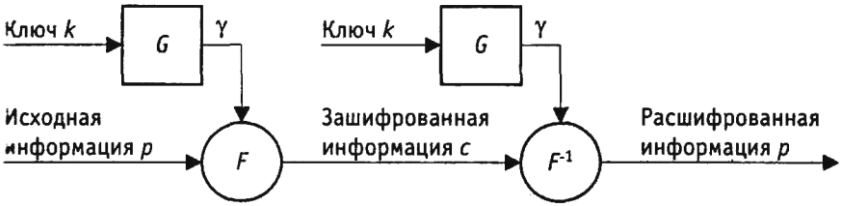

Рис.1 - Шифрование информации методом гаммирования (схема синхронного поточного шифра): G – генератор псевдослучайных кодов, F – линейная или нелинейная функция гаммирования, F^{-1} – функция, обратная F, γ - гамма шифра

В синхронных поточных шифрах отсутствует эффект размножения ошибок, т.е. число искаженных элементов в расшифрованной последовательности равно числу искаженных элементов зашифрованной последовательности, пришедшей из канала связи. Вставка или выпадение элемента зашифрованной последовательности недопустимы, т.к. из-за нарушения синхронизации это приведет к неправильному расшифрованию всех последующих элементов.

В самосинхронизирующихся поточных шифрах элементы входной последовательности зашифровываются с учетом N предшествующих элементов (рис. 2), которые принимают участие в формировании ключевой последовательности.

В самосинхронизирующихся шифрах имеет место эффект размножения ошибок, в то же время в отличие от синхронных восстановление синхронизации происходит автоматически через N элементов зашифрованной последовательности.

Скремблером называется программная или аппаратная реализация алгоритма, позволяющего шифровать побитно непрерывные потоки информации.

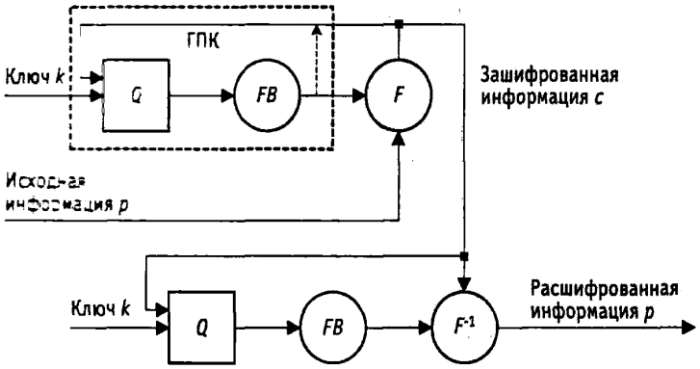

Рис. 2 - Шифрование информации методом гаммирования с обратной связью (схема самосинхронизирующегося поточного шифра): G – генератор псевдослучайных кодов, Q – элементы памяти ГПК, FB – функция обратной связи ГПК, F – линейная или нелинейная функция, F^{-1} – функция, обратная F

Шифрование 8-и и 16-ти - разрядными скремблерами реализуется следующим образом: из начального объема данных, заданных ключом, выбираются определенным образом биты и складываются по модулю 2 (XOR, $\oplus$) между собой. Какие именно биты - определяется позициями тех битов скремблера, где находятся единицы. Все биты очередного ключа сдвигаются влево на 1 позицию, а результат сложения помещают в освободивший младший разряд. Бит кодирующей последовательности получается так: самый старший бит ключа, находившийся там до сдвига, складывается по модулю 2 с очередным битом исходного текста.

Таким образом, с помощью скремблера и начального значения ключа мы получаем псевдослучайную последовательность, которая должна обладать тремя свойствами:

1. Сбалансированность: для каждого интервала последовательности количество двоичных единиц должно отличаться от числа двоичных нулей не больше чем на один элемент.

Рис. 3 - Шифрование текста с помощью полученного ключа

2. Цикличность: циклом называют непрерывную последовательность одинаковых двоичных чисел. Появление иной двоичной цифры автоматически начинает новый цикл. Длина цикла равна количеству в нем. Необходимо, чтобы половина всех “полосок» (подряд идущих идентичных компонентов последовательности) имела длину 1, одна четвертая – длину 2, одна восьмая – длину 3 и т.д.

3. Корреляция: если часть последовательности и её циклично сдвинутая копия поэлементно сравниваются, желательно, чтобы число совпадений отличалось от числа несовпадений не более чем на 1.

Благодаря применению только операции XOR, декодирование заскремблированных последовательностей происходит по той же самой схеме, что и кодирование.

При достаточно долгой работе скремблера неизбежно возникает его зацикливание. По выполнении определенного числа тактов в ячейках скремблера создастся комбинация бит, которая в нем уже однажды оказывалась, и с этого момента кодирующая последовательность начнет циклически повторяться с фиксированным периодом. Данная проблема неустранима по своей природе, так как в N разрядах скремблера не может пребывать более 2^N комбинаций бит, и, следовательно, максимум, через, 2^N-1 циклов повтор комбинации обязательно произойдет. Последовательность бит, генерируемая таким скремблером, называется последовательностью наибольшей длины (ПНД).

Чтобы построить N-разрядный скремблер, создающий ПНД, пользуются неприводимыми многочленами (такими,

которых нельзя представить в виде произведения никаких других многочленов). Найденный неприводимый многочлен степени N записывается в двоичном виде, затем отбрасывается единица, соответствующая самому старшему разряду.

Для 8-ми разрядного скремблера можно привести такие примеры схем, которые позволяют сгенерировать ПНД: 10110001 и 10111000, которым соответствуют простые неприводимые полиномы $x^8+x^6+x^5+x^1$ и $x^8+x^6+x^5+x^4$, соответственно. Период последовательности кодирующих бит в случае использования этих схем скремблирования равен максимуму - 255.

Работа скремблера осуществляется через операции сдвига (>> или <<) и сложения по модулю 2 (^).

Алгоритм генерации последовательности шифротекста при использовании 8-битового и 16-битовых ключей одинаков.

Приведем пример работы 8-ми разрядного скремблера, генерирующего последовательность с периодом равным T=9. Этот скремблер имеет вид "11111111". Начальный ключ имеет вид: "10000000". Последовательность генерируемых ключей представлена на рисунке 4.

Период повторений последовательности кодирующих бит в случае использования этого скремблера равен минимуму - 9-ти.

Задание на лабораторную работу

На одном из языков программирования реализовать приложение, позволяющее смоделировать работу скремблера. Его задачи состоят в следующем:

Исходные данные:

- Открытый текст
- Скремблер
- Начальная комбинация бит для скремблера (ключ).

1. Получить зашифрованный текст при известных начальном ключе и скремблере.

2. По зашифрованному тексту получить его открытый вариант при известных начальном ключе и скремблере.

3. По начальному ключу и скремблеру получить период скремблера.

4. Показать, что последовательность ключей, генерируемая скремблером является/не является псевдослучайной.

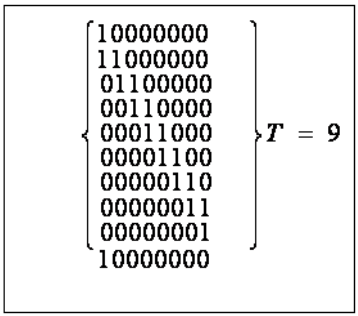

Рис. 4 - Последовательность генерируемых ключей для 8-ми разрядного скремблера

Содержание отчета

Отчет должен содержать:

1. Название и цель лабораторной работы;
2. Краткий теоретический материал по теме работы;
3. Блок-схема алгоритма разрабатываемой программы;
4. Отчет о разработке приложения с приведенным кодом программы;
5. Выводы.

Контрольные вопросы

1. На какие два класса можно разделить поточные шифры?
2. Дайте определение скремблера, принцип его работы.
3. Как определить максимальную длину скремблера?
4. Какая последовательность нулей и единиц называется псевдослучайной, назовите ее свойства?
5. В чем заключаются преимущества и недостатки использования скремблера?

Лабораторная работа № 2

Комбинированные криптографические алгоритмы

Цель работы: научиться применять теоретические знания в области криптографии для составления алгоритмов и программ, реализующих функции шифрования и дешифрования исходного текста с помощью комбинированных криптографических алгоритмов.

Краткие теоретические сведения

При выполнении данной лабораторной работы рекомендуется задействовать алгоритмы замены или перестановок.

Например, возможна реализация следующего комбинированного криптографического алгоритма шифрования и дешифрования данных при использовании шифров замены и перестановок:

Шифрование:

Замена символов исходного текста на двузначные цифры (например, А→01; Б→02 и т.д.); перестановка трех соседних цифр, например, в таком порядке: 1→3, 2→1, 3→2 (циклический сдвиг влево).

Пример:

Исходный текст: КОД

После замены: 121605(К→12; О→16; Д→05)

После перестановки: 211056 (121→211, 605→056)

Дешифрование:

Обратная перестановка трех соседних цифр исходного текста в порядке: 3→1, 1→2, 2→3 (циклический сдвиг вправо); замена пары цифр на символ, согласно алфавиту замены (01→А; 02→Б и т.д.).

Пример:

Исходный текст: КОД

После замены: 121605(К→12; О→16; Д→05)

После перестановки: 211056 (121→211, 605→056)

Дробные шифры. В этих шифрах каждая буква сначала зашифровывается в две (или более) буквы или в два (или более) числа, затем полученные символы каким-либо способом перемешиваются (например, с помощью транспозиции), после чего их можно снова перевести в первоначальный алфавит. Таким образом, используя в качестве ключа перемешанный 25-буквенный алфавит, можно перевести буквы в двухзначные пятеричные числа с помощью таблицы 1.

Таблица 1 – Таблица соответствия букв латинского алфавита двузначному пятиричному коду

	0	1	2	3	4
0	L	Z	Q	C	P
1	A	G	N	O	U
2	R	D	M	I	F
3	K	Y	H	V	S
4	X	B	T	E	W

Например, букве В соответствует число 41. После того как полученный ряд чисел подвергнут некоторой перестановке, его можно снова разбить на пары чисел и перейти к буквам.

Кадры. В кодах слова (или иногда слоги) заменяются группами букв. Иногда затем применяется шифр того или иного вида.

Оценка секретных систем

Имеется несколько различных критериев, которые можно было бы использовать для оценки качества предлагаемой секретной системы. Наиболее важные из них приведены ниже.

Количество секретности. Некоторые секретные системы являются совершенными в том смысле, что положение

противника не облегчается в результате перехвата любого количества сообщений.

Другие системы, хотя и дают противнику некоторую информацию при перехвате очередной криптограммы, но не допускают единственного «решения». Системы, допускающие единственное решение, очень разнообразны, как по затрате времени и сил, необходимых для получения этого решения, так и по количеству материала, который необходимо перехватить для получения единственного решения.

Объем ключа. Ключ должен быть передан из передающего пункта в приемный пункт таким способом, чтобы его нельзя было перехватить. Иногда его нужно запомнить. Поэтому желательно иметь ключ настолько малый, насколько это возможно.

Сложность операций шифрования и дешифрования. Операции шифрования и дешифрования должны быть по возможности простыми. Если эти операции производятся вручную, то их сложность приводит к потере времени, появлению ошибок и т.д. Если они производятся механически, то сложность вызывает необходимость использования больших и дорогих устройств.

Разрастание числа ошибок. В некоторых типах шифров ошибка в одной букве, допущенная при шифровании или передаче, приводит к большому числу ошибок в расшифрованном тексте. Такие ошибки разрастаются в результате операции дешифрирования, вызывая значительную потерю информации и часто требуя повторной передачи криптограммы. Желательно минимизировать возрастание числа ошибок.

Увеличение объема сообщения. В некоторых типах секретных систем объем сообщения увеличивается в результате операции шифрования. Этот нежелательный эффект можно наблюдать в системах, в которых делается попытка потопить статистику сообщения в массе добавляемых нулевых символов, или там, где используются многократные замены. Он имеет место также во многих системах типа

«маскировки» (которые не являются обычными секретными системами в смысле данного выше определения).

Стандарты шифрования

В 1992 г. Гостехкомиссия (ГТК) при Президенте РФ разработала и опубликовала 4 руководящих документов, посвященных вопросам защиты информации в АС и ее обработки.

ГОСТ 28147-89 «Системы обработки информации. Защита криптографическая. Алгоритм криптографического преобразования».

Выдержка: «Настоящий стандарт устанавливает единый алгоритм криптографического преобразования для систем обработки информации в сетях электронных вычислительных машин (ЭВМ), отдельных вычислительных комплексах и ЭВМ, который определяет правила шифрования данных и выработки эмитовставки.

Алгоритм криптографического преобразования предназначен для аппаратной или программной реализации, удовлетворяет криптографическим требованиям и по своим возможностям не накладывает ограничений на степень секретности защищаемой информации.

Стандарт обязателен для организаций, предприятий и учреждений, применяющих криптографическую защиту данных, хранимых и передаваемых в сетях ЭВМ, в отдельных вычислительных комплексах или в ЭВМ».

ГОСТ Р 34.10-94 «Информационная технология. Криптографическая защита информации. Процедуры выработки и проверки электронной цифровой подписи на базе асимметричного криптографического алгоритма».

Настоящий стандарт определяет процедуру выработки и проверки электронной цифровой подписи на базе асимметричного криптографического алгоритма с применением функции хэширования. Электронная цифровая подпись обеспечивает целостность сообщений (документов), передаваемых по незащищенным телекоммуникационным каналам общего пользования в системах обработки

информации различного назначения, с гарантированной идентификацией ее автора (лица, подписавшего документ).

Выдержка: «Настоящий стандарт устанавливает процедуры выработки и проверки электронной цифровой подписи (ЭЦП) сообщений (документов), передаваемых по незащищенным телекоммуникационным каналам общего пользования в системах обработки информации различного назначения на базе асимметричного криптографического алгоритма с применением функции хэширования.

Внедрение системы ЭЦП на базе настоящего стандарта обеспечивает защиту передаваемых сообщений от подделки, искажения и однозначно позволяет доказательно подтвердить подпись лица, подписавшего сообщение».

ГОСТ Р 34.11-94 «Информационная технология. Криптографическая защита информации. Функция хэширования».

Криптографические методы защиты информации являются объектом серьезным научных исследований и стандартизации на национальных, региональных и международных уровнях. Данный стандарт определяет процедуру вычисления хэш-функции для любой последовательности двоичных символов. Функция хэширования заключается в сопоставлении произвольного набора данных в виде последовательности двоичных символов и его образа фиксированной небольшой длины, что позволяет использовать эту функцию в процедурах электронной цифровой подписи для сокращения времени подписывания и проверки подписи. Эффект сокращения времени достигается за счет вычисления подписи только под образом подписываемого набора данных.

Выдержка: «Настоящий стандарт определяет алгоритм и процедуру вычисления хэш-функции для любой последовательности двоичных символов, которые применяются в криптографических методах обработки и защиты информации, в том числе для реализации процедур электронной цифровой подписи (ЭЦП) при передаче,

обработке и хранении информации в автоматизированных системах.

Определенная в настоящем стандарте функция хэширования используется при реализации систем электронной цифровой подписи на базе асимметричного криптографического алгоритма по ГОСТ Р 34.10-94».

ГОСТ Р 50739-95 «Средства вычислительной техники. Защита от несанкционированного доступа к информации. Общие технические требования».

Выдержка: «Настоящий стандарт устанавливает единые функциональные требования к защите средств вычислительной техники (СВТ) от несанкционированного доступа (НСД) к информации; к составу документации на эти средства, а также номенклатуру показателей защищенности СВТ, описываемых совокупностью требований к защите и определяющих классификацию СВТ по уровню защищенности от НСД к информации.

Под СВТ в данном стандарте понимается совокупность программных и технических элементов систем обработки данных, способных функционировать самостоятельно или в составе других систем.

Применение в комплекте СВТ средств криптографической защиты информации может быть использовано для повышения гарантий качества защиты.

Требования настоящего стандарта являются обязательными.

В данном стандарте различают дискретизационный (дискреционный) и мандатный принципы контроля доступа.

Для реализации дискретизационного контроля доступа комплекс средств защиты (КСЗ) должен контролировать доступ именованных субъектов (пользователей) к именованным объектам (файлам, программам, томам). Для каждой пары (субъект-объект) в средствах вычислительной техники (СВТ) должно быть задано явное и недвусмысленное перечисление допустимых типов доступа (читать, писать), то есть тех типов доступа, которые являются

санкционированными для данного субъекта (индивида или группы индивидов) к данному ресурсу СВТ (объекту). Контроль доступа должен быть применим к каждому объекту и к каждому субъекту. Механизм, реализующий дискретизационный принцип контроля доступа, должен предусматривать санкционированное изменение правил, разграничения доступа (ПРД), в том числе санкционированное изменение списка пользователей СВТ и списка защищаемых объектов. Право изменять ПРД должно быть предоставлено выделенным субъектам (администрация, служба безопасности). Должны быть предусмотрены средства управления, ограничивающие распространение прав доступа.

Для реализации мандатного принципа контроля доступа каждому субъекту и каждому объекту присваивают классификационные метки, отражающие их места в соответствующей иерархии. С помощью этих меток субъектам и объектам должны быть назначены классификационные уровни, являющиеся комбинациями уровня иерархической классификации и иерархических категорий. Данные метки должны служить основой мандатного принципа разграничения доступа».

Лабораторное задание

Используя один из описанных криптографических алгоритмов (замены или перестановок), на одном из языков программирования разработать приложение для шифрации и дешифрации текста.

Содержание отчета

Отчет должен содержать:

1. Название и цель лабораторной работы;
2. Краткие теоретические сведения;
3. Блок-схема алгоритма разрабатываемой программы;
4. Отчет о разработке приложения с приведенным кодом программы;
5. Выводы.

Контрольные вопросы

1. Что называют комбинированными криптографическими алгоритмами?
2. Поясните, как происходит построение комбинированного алгоритма.
3. Какие шифры называются дробными?
4. Перечислить основные критерии оценки секретности систем.
5. Как определяется количество секретности системы?
6. В каких известных вам шифрах в результате операции шифрования происходит увеличение объема сообщения?
7. К каким последствиям приводит разрастание числа ошибок?

Лабораторная работа № 3

Арифметические алгоритмы шифрования

Цель работы: исследование и разработка основных методов симметричных криптосистем.

Краткие теоретические сведения

По мере развития и усложнения средств, методов и форм автоматизации процессов обработки информации повышается зависимость общества от степени безопасности используемых им информационных технологий, которая определяется степенью защищенности и устойчивости как компьютерных систем в целом, так и отдельных программ.

Криптография – обеспечивает сокрытие смысла сообщения с помощью шифрования и открытия его расшифрованием, которые выполняются по специальным алгоритмам с помощью ключей.

Ключ – конкретное секретное состояние некоторых параметров алгоритма криптографического преобразования данных, обеспечивающее выбор только одного варианта из всех возможных для данного алгоритма.

Криптоанализ – занимается вскрытием шифра без знания ключа (проверка устойчивости шифра).

Кодирование – (не относится к криптографии) – система условных обозначений, применяемых при передаче информации. Применяется для увеличения качества передачи информации, сжатия информации и для уменьшения стоимости хранения и передачи.

Криптосистемы разделяются на симметричные и с открытым ключом.

В симметричных криптосистемах и для шифрования, и для дешифрования используется один и тот же ключ.

В системах с открытым ключом используются два ключа - открытый и закрытый, которые математически связаны друг с другом. Информация шифруется с помощью открытого ключа, который доступен всем желающим, а расшифровывается с помощью закрытого ключа, известного только получателю сообщения.

Криптографические преобразования имеют цель обеспечить недоступность информации для лиц, не имеющих ключа, и поддержание с требуемой надежностью обнаружения несанкционированных искажений. Большинство средств защиты информации базируется на использовании криптографических шифров и процедур шифрования - расшифрования. В соответствии со стандартом ГОСТ 28147-89 под шифром понимают совокупность обратимых преобразований множества открытых данных на множество зашифрованных данных, задаваемых ключом и алгоритмом преобразования.

В криптографии используются следующие основные алгоритмы шифрования:

- алгоритм замены (подстановки) – символы шифруемого текста заменяются символами того же или другого алфавита в соответствии с заранее обусловленной схемой замены;

- алгоритм перестановки – символы шифруемого текста переставляются по определенному правилу в пределах некоторого блока этого текста;

- гаммирование – символы шифруемого текста складываются с символами некоторой случайной последовательности;

- аналитическое преобразование – преобразование шифруемого текста по некоторому аналитическому правилу (формуле).

С алгоритмами замены и перестановки вы познакомились при выполнении лабораторной работы № 3.

Процессы шифрования и расшифрования осуществляются в рамках некоторой криптосистемы. Для симметричной криптосистемы характерно применение одного и того же ключа, как при шифровании, так и при расшифровании сообщений. В асимметричных криптосистемах для зашифрования данных используется один (общедоступный) ключ, а для расшифрования – другой (секретный) ключ.

Симметричные криптосистемы.

Шифры перестановки. В шифрах средних веков часто использовались таблицы, с помощью которых выполнялись простые процедуры шифрования, основанные на перестановке букв в сообщении. Ключом в данном случае является размеры таблицы. Например, сообщение “Сегодня новый день” записывается в таблицу из 4 строк и 4 столбцов по столбцам.

Таблица 2 – Пример представления сообщения

С	Д	О	Д
Е	Н	В	Е
Г	Я	Ы	Н
О	Н	Й	Ь

Для получения шифрованного сообщения текст считывается по строкам и группируется по 4 букв: СДОД_ЕНВЕ _ГЯЫН_ОНЙЬ

Несколько большей стойкостью к раскрытию обладает метод одиночной перестановки по ключу. Он отличается от предыдущего тем, что столбцы таблицы переставляются по ключевому слову, фразе или набору чисел длиной в строку таблицы. Используя в качестве ключа слово Ваза, получим таблицу 3

В верхней строке левой таблицы записан ключ, а номера под буквами ключа определены в соответствии с естественным порядком соответствующих букв ключа в алфавите. Если в ключе встретились бы одинаковые буквы, они бы нумеровались слева направо. Получается шифровка: ДДСО_НЕЕВ_ЯНГЫ_НЬОЙ.

Таблица 3 – Пример использования алгоритма

До перестановки					После перестановки			
В	А	З	А		А	А	В	З
3	1	4	2		1	2	3	4
С	Д	О	Д		Д	Д	С	О
Е	Н	В	Е		Н	Е	Е	В
Г	Я	Ы	Н		Я	Н	Г	Ы
О	Н	Й	Ь		Н	Ь	О	Й

Для обеспечения дополнительной скрытности можно повторно шифровать сообщение, которое уже было зашифровано. Для этого размер второй таблицы подбирают так, чтобы длины ее строк и столбцов отличались от длин строк и столбцов первой таблицы. Лучше всего, если они будут взаимно простыми.

Кроме алгоритмов одиночных перестановок применяются алгоритмы двойных перестановок. Сначала в таблицу записывается текст сообщения, а потом поочередно переставляются столбцы, а затем строки. При расшифровке порядок перестановок будет обратный. Число вариантов двойной перестановки достаточно быстро возрастает с увеличением размера таблицы: для таблицы 3 х 3 их 36, для 4 х 4 их 576, а для 5*5 их 14400.

Пример данного метода шифрования показан в следующих таблицах. Ключом к шифру служат номера столбцов 2413 и номера строк 4123 исходной таблицы :

Таблица 4 – Пример шифрования методом двойных перестановок

	2	4	1	3			1	2	3	4			1	2	3	4
4	С	Е	Г	О		4	Г	С	О	Е		1	Я	Д	Н	Н
1	Д	Н	Я	Н		1	Я	Д	Н	Н		2	Ы	О	Й	В
2	О	В	Ы	Й		2	Ы	О	Й	В		3	Н	Д	Ь	Е
3	Д	Е	Н	Ь		3	Н	Д	Ь	Е		4	Г	С	О	Е

Двойная перестановка столбцов и строк

В результате перестановки получена шифровка: ЯДННЫОЙВНДЬЕГСОЕ. В средние века для шифрования применялись и магические квадраты. Магическими

квадратами называются квадратные таблицы с вписанными в их клетки последовательными натуральными числами, начиная с единицы, которые дают в сумме по каждому столбцу, каждой строке и каждой диагонали одно и то же число. Для шифрования необходимо вписать исходный текст по приведенной в квадрате нумерации и затем переписать содержимое таблицы по строкам. В результате получается шифротекст, сформированный благодаря перестановке букв исходного сообщения.

Лабораторное задание

На одном из языков программирования разработать приложение для шифрования и дешифрования текстового файла одним из указанных методов (гаммирование или аналитических преобразований).

Содержание отчета:

Отчет должен содержать:

1. Название и цель лабораторной работы;
2. Краткие теоретические сведения;
3. Блок-схема алгоритма разрабатываемой программы;
4. Отчет о разработке приложения с приведенным кодом программы;
5. Выводы.

Контрольные вопросы

1. Цель и задачи криптографии.
2. Шифры одиночной перестановки и перестановки по ключевому слову.
3. Шифры двойной перестановки. Шифрование с помощью магического квадрата.
4. Метод гаммирования.
5. Метод аналитических преобразований.

Лабораторные работы № 4 – 7

Алгоритмы криптосистем с открытом ключом

Цель работы: исследование и анализ основных методов асимметричных криптосистем и криптосистем с открытым ключом.

Краткие теоретические сведения

Как бы ни были сложны и надежны криптографические системы - их слабое мест при практической реализации - проблема распределения ключей. Для того, чтобы был возможен обмен конфиденциальной информацией между двумя субъектами ИС, ключ должен быть сгенерирован одним из них, а затем каким-то образом опять же в конфиденциальном порядке передан другому. Т.е. в общем случае для передачи ключа опять же требуется использование какой-то криптосистемы.

Для решения этой проблемы на основе результатов, полученных классической и современной алгеброй, были предложены системы с открытым ключом.

Суть их состоит в том, что каждым адресатом ИС генерируются два ключа, связанные между собой по определенному правилу. Один ключ объявляется открытым, а другой закрытым. Открытый ключ публикуется и доступен любому, кто желает послать сообщение адресату. Секретный ключ сохраняется в тайне.

Исходный текст шифруется открытым ключом адресата и передается ему. Зашифрованный текст в принципе не может быть расшифрован тем же открытым ключом. Дешифрование сообщение возможно только с использованием закрытого ключа, который известен только самому адресату.

В самом определении необратимости присутствует неопределенность. Под необратимостью понимается не теоретическая необратимость, а практическая невозможность

вычислить обратное значение используя современные вычислительные средства за обозримый интервал времени.

Поэтому чтобы гарантировать надежную защиту информации, к системам с открытым ключом (СОК) предъявляются два важных и очевидных требования:

6. Преобразование исходного текста должно быть необратимым и исключать его восстановление на основе открытого ключа.

7. Определение закрытого ключа на основе открытого также должно быть невозможным на современном технологическом уровне. При этом желательна точная нижняя оценка сложности (количества операций) раскрытия шифра.

Схема шифрования Эль Гамаля. Алгоритм шифрования Эль Гамаля основан на применении больших чисел для генерации открытого и закрытого ключа, криптостойкость же обусловлена сложностью вычисления дискретных логарифмов.

Последовательность действий пользователя:

1. Получатель сообщения выбирает два больших числа P и G, причем P > G.
2. Получатель выбирает секретный ключ - случайное целое число X < P.
3. Вычисляется открытый ключ $Y= G^{x} \bmod P$.
4. Получатель выбирает целое число K, 1< K< P-1.
5. Шифрование сообщения (M): $a= G^{K} \bmod P$, $b=Y^{K} M \bmod P$, где пара чисел (a,b) является шифротекстом.

Криптосистема шифрования данных RSA. Предложена в 1978 году авторами Rivest, Shamir и Aldeman и основана на трудности разложения больших целых чисел на простые сомножители.

Они воспользовались тем фактом, что нахождение больших простых чисел в вычислительном отношении осуществляется легко, но разложение на множители произведения двух таких чисел практически невыполнимо. Доказано, что раскрытие шифра RSA эквивалентно такому разложению. Поэтому для любой длины ключа можно дать нижнюю оценку числа операций для раскрытия шифра, а с

учетом производительности современных компьютеров оценить и необходимое на это время.

Возможность гарантированно оценить защищенность алгоритма RSA стала одной из причин популярности этой СОК на фоне десятков других схем. Поэтому алгоритм RSA используется в банковских компьютерных сетях, особенно для работы с удаленными клиентами (обслуживание кредитных карточек).

В настоящее время алгоритм RSA используется во многих стандартах, среди которых SSL, S-HHTP, S-MIME, S/WAN, STT и PCT.

Последовательность действий пользователя:

1. Получатель выбирает 2 больших простых целых числа p и q, на основе которых вычисляет N=pq; M=(p-1)(q-1).
2. Получатель выбирает целое случайное число d, которое является взаимопростым со значением M, и вычисляет значение e из условия ed=1(mod M).
3. d и N публикуются как открытый ключ, e и M являются закрытым ключом.
4. Если S –сообщение и его длина: 1<Len(S)<N, то зашифровать этот текст можно как $S'=S^d \pmod N$, то есть шифруется открытым ключом.
5. Получатель расшифровывает с помощью закрытого ключа: $S=S'^e \pmod N$.

Пример: Зашифруем сообщение "CAB". Для простоты будем использовать маленькие числа (на практике применяются гораздо большие).

1. Выберем p=3 и q=11.
2. Определим n=3*11=33.
3. Найдем (p-1)(q-1)=20. Следовательно, в качестве d, взаимно простое с 20, например, d=3.
4. Выберем число e. В качестве такого числа может быть взято любое число, для которого удовлетворяется соотношение (e*3) (mod 20) = 1, например 7.
5. Представим шифруемое сообщение как последовательность целых чисел с помощью отображения: A1,

В2, С3. Тогда сообщение принимает вид (3,1,2). Зашифруем сообщение с помощью ключа {7,33}.

ШТ1 = (3^7) (mod 33) = 2187 (mod 33) = 9,

ШТ2 = (1^7) (mod 33) = 1 (mod 33) = 1,

ШТ3 = (2^7) (mod 33) = 128 (mod 33) = 29.

6. Расшифруем полученное зашифрованное сообщение (9,1,29) на основе закрытого ключа {3,33}:

ИТ1 = (9^3) (mod 33) = 729 (mod 33) = 3,

ИТ2= (1^3) (mod 33) = 1 (mod 33) = 1,

ИТ3 = (29^3) (mod 33) = 24389 (mod 33) = 2.

Итак, в реальных системах алгоритм RSA реализуется следующим образом: каждый пользователь выбирает два больших простых числа, и в соответствии с описанным выше алгоритмом выбирает два простых числа e и d. Как результат умножения первых двух чисел (p и q) устанавливается n.

{e,n} образует открытый ключ, а {d,n} - закрытый (хотя можно взять и наоборот).

Открытый ключ публикуется и доступен каждому, кто желает послать владельцу ключа сообщение, которое зашифровывается указанным алгоритмом. После шифрования, сообщение невозможно раскрыть с помощью открытого ключа. Владелец же закрытого ключа без труда может расшифровать принятое сообщение.

Лабораторное задание

На одном из языков программирования разработать приложение для шифрования и дешифрования текстового файла одним из указанных методов.

№ л.р.	Задание на лабораторную работу (алгоритмы для реализации)
4	Алгоритм шифрации двойным квадратом. Шифр Enigma.
5	Алгоритм шифрования DES. Алгоритм шифрования ГОСТ 28147-89.
6	Алгоритм шифрования RSA.
7	Алгоритм шифрования Эль Гамаля.

Отчет должен содержать:

1. Название и цель лабораторной работы;
2. Краткие теоретические сведения;
3. Блок-схема алгоритма разрабатываемой программы;
4. Отчет о разработке приложения с приведенным кодом программы;
5. Выводы.

Контрольные вопросы

1. Криптосистема с открытым ключом.
2. Идея криптосистемы с открытым ключом
3. Схема шифрования с открытым ключом
4. Основные принципы построения криптосистем с открытым ключом
5. Криптография с несколькими открытыми ключами
6. Криптоанализ алгоритмов с открытым ключом
7. Асимметричные алгоритмы шифрования и их отличительные особенности.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Анин Б.Ю. Защита компьютерной информации / Б.Ю. Анин. – СПб. : БХВ-Петербург, 2002. – 384 с.

2. Баричев С.Г. Основы современной криптографии / С.Г. Баричев, В.В. Гончаров, Р.Е. Серов. – М.: Горячая линия – Телеком, 2001. – 120 с.

3. Барсуков В.С. Современные технологии безопасности /В.С. Барсуков, В.В. Водолазкий. – М.: Нолидж, 2000. – 496 с.

4. Грушо А.А. Теоретические основы защиты информации / А.А. Грушо, Е.Е. Тимонина. – М.: Изд-во Агентства «Яхтсмен», 1996. – 192 с.