МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение высшего образования
«Тольяттинский государственный университет»

Кафедра Прикладная математика и информатика
(наименование)

09.03.03 Прикладная информатика
(код и наименование направления подготовки / специальности)

Разработка социальных и экономических информационных систем
(направленность (профиль) / специализация)

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА (БАКАЛАВРСКАЯ РАБОТА)

на тему Разработка защиты документации от несанкционированного доступа

Обучающийся Д.Р. Хуббатова _____
(Инициалы Фамилия) (личная подпись)

Руководитель В.В.Дружинкин
(ученая степень (при наличии), ученое звание (при наличии), Инициалы Фамилия)

Тольятти 2025

Аннотация

Выпускная квалификационная работа на тему "Разработка алгоритма защиты документации от несанкционированного доступа" состоит из 3 разделов. Объем выпускной квалификационной работы составляет 62 страницы. Содержит 18 рисунков, 8 таблиц.

В первом разделе ВКР "Теоретические основы криптографии и алгоритма ГОСТ 34.12-2018 "Кузнечик" представлены основные принципы и методы симметричной криптографии, плюсы и минусы использования алгоритма в современных системах и угрозы безопасности, против которых направлена защита с использованием ГОСТ 34.12-2018.

Второй раздел ВКР "Реализация локальной криптографической системы на основе ГОСТ 34.12-2018" анализ требований к системе принципов и условий обработки, проектирование архитектуры системы, реализация ключевых функций, тестирование и верификация системы.

В третьем разделе ВКР «Экономическая часть» проведен расчет экономической эффективности внедрения системы, оценка рисков и возможных экономических убытков от несанкционированного доступа, расчет стоимости реализации и внедрения системы, прогнозирование экономических выгод от внедрения системы в долгосрочной перспективе.

При написании ВКР использовались нормативно-правовые документы, а также 29 литературных источников.

Оглавление

Введение

Актуальность темы выпускной квалификационной работы обусловлена возрастающей потребностью в защите информации в современном цифровом мире. Конфиденциальные данные как коммерческих, так и государственных организаций подвергаются все большему риску утечки и несанкционированного использования злоумышленниками. В связи с этим особую важность приобретают вопросы применения надежных криптографических систем и алгоритмов для обеспечения информационной безопасности.

Одним из перспективных решений в этой области является российский государственный стандарт криптографического алгоритма блочного шифрования ГОСТ 34.12-2018 "Кузнечик". Данный алгоритм обладает высокой стойкостью к взлому и отвечает современным требованиям по защите конфиденциальных данных. Вместе с тем практический опыт его применения в настоящее время невелик.

Целью выпускной квалификационной работы является исследование алгоритма ГОСТ 34.12-2018 "Кузнечик" и разработка на его основе локальной криптографической системы для защиты конфиденциальной информации.

– Для достижения поставленной цели необходимо решить следующие задачи:

– Изучить теоретические основы криптографии и особенности алгоритма ГОСТ 34.12-2018.

– Выполнить анализ достоинств, недостатков и областей применения данного алгоритма.

– Рассмотреть угрозы информационной безопасности, от которых защищает ГОСТ 34.12-2018.

– Спроектировать и реализовать локальную криптографическую систему на основе ГОСТ 34.12-2018.

– Провести тестирование и верификацию разработанной системы.

– Рассчитать экономическую эффективность и затраты на внедрение системы.

Объектом исследования является алгоритм симметричного блочного шифрования ГОСТ 34.12-2018 "Кузнечик".

Предмет исследования – методы проектирования и разработки локальной криптографической системы на основе ГОСТ 34.12-2018 для защиты конфиденциальной информации.

Методологической основой работы послужат общенаучные методы анализа и синтеза, сравнения и обобщения. Для изучения алгоритма ГОСТ 34.12-2018 планируется использовать методы описания, схемы и блок-схемы. При разработке и внедрении криптографической системы будет применен системный подход, включающий этапы анализа требований, проектирования, верификации и моделирования системы в рамках объектно-ориентированной парадигмы. В экономическом разделе используются методы прогнозирования, расчета экономического эффекта и оценки рисков.

Глава 1 Теоретические основы криптографии и алгоритма ГОСТ 34.12-2018 "Кузнечик"

1.1 Исторический обзор развития криптографии

Зарождение науки о методах скрытой передачи информации относится к глубокой древности. Первые достоверные сведения о использовании криптографии для защиты важных данных восходят к Древнему Египту (XX-XVIII века до н.э.). В это время жреческая знать активно применяла простые методы замены букв и использование секретных языков для переписки по конфиденциальным вопросам религиозного и государственного управления [1].

Особое развитие наука шифрования получила в период расцвета Древней Греции в V-IV веках до н.э [2]. Греческий историк Геродот и философ Плутарх описывают использование греками первых перестановочных шифров, основанных на изменении порядка букв с помощью секретного ключа. Это было важным моментом в истории криптографии, поскольку оно открыло путь к разработке более сложных методов обеспечения конфиденциальности информации.

Перестановочные шифры, использованные в Древней Греции, представляли собой методы, при которых символы или буквы в тексте переупорядочивались в соответствии с секретным ключом. Это означало, что даже если злоумышленнику удавалось получить доступ к зашифрованному сообщению, ему было бы крайне сложно расшифровать его без знания ключа. Это обеспечивало надежную защиту конфиденциальных сообщений. Кроме того, древние зашифрованные тексты, найденные в греческих колониях на территории Малой Азии и датируемые 400-300 гг. до н.э., представляют собой ценный археологический источник информации о развитии криптографии в тот период. Они демонстрируют, что использование шифров и методов обеспечения

конфиденциальности было распространено и практиковалось в различных областях Древней Греции.

Схема одного из перестановочных шифров, использующихся в SP-сети [3], может быть описана следующим образом: каждая буква исходного сообщения заменяется на другую букву в соответствии с определенными правилами, заданными секретным ключом. Эти правила определяют, какие буквы заменяются другими и в каком порядке. Такой шифр делает текст практически нечитаемым для посторонних лиц, не обладающих ключом расшифровки.

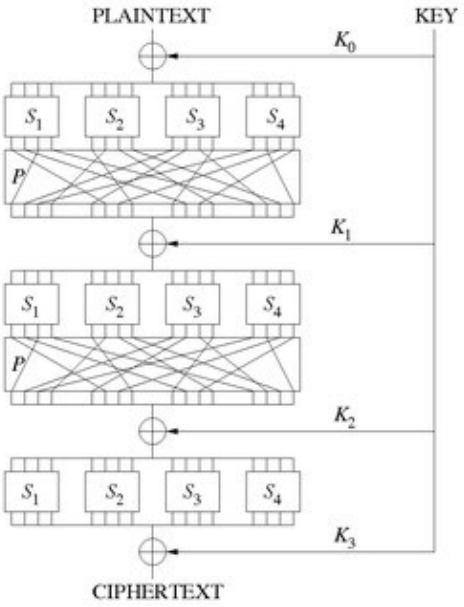

Рисунок 1 – Схема перестановочного шифра

В Древнем Риме, на протяжении жизни и деятельности юридического деятеля и политика Юлия Цезаря (100-40 н.э.), был сформирован и описан один из первых методов подстановочного шифрования, который позднее получил имя в его честь. Шифр Цезаря, также известный как шифр сдвига [4], представлял собой довольно простой и в то же время эффективный способ обеспечения

конфиденциальности текстовой информации. Он заключался в замене каждой буквы в исходном тексте на другую букву, находящуюся в алфавите на несколько позиций левее или правее.

Детализируя этот метод, шифр Цезаря можно описать следующим образом: для каждой буквы исходного текста применялось смещение (как правило, фиксированное), и вместо нее записывалась буква, которая находилась на указанном смещении в алфавите. Например, при смещении на 3 позиции влево, буква "А" заменялась буквой "X", буква "Б" становилась "Y", и так далее. Этот метод подстановки шифра Цезаря был относительно легко понять и использовать, но он предоставлял некоторую степень конфиденциальности в сообщениях, которые передавались между соратниками и агентами. Тем не менее, он не являлся высокоустойчивым перед современными методами криптоанализа, так как существовал ограниченный набор возможных ключей (смещений), и даже без специальных инструментов можно было провести атаку методом грубой силы.

Шифр Цезаря стал важным этапом в развитии криптографии и подготовил почву для более сложных методов шифрования, которые пришли в будущем. Тем не менее, его упрощенная схема подчеркивает начальные шаги в области обеспечения конфиденциальности данных и является одним из первых шагов в долгом пути к современной криптографии. Ниже показана его упрощенная схема.

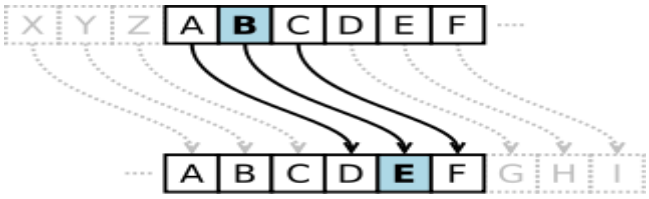

Рисунок 2 – Схема шифра Цезаря

Средние века, период с V по XV век, был временем, когда наука о методах скрытой передачи сообщений, или криптография, активно применялась в дипломатии и разведывательной деятельности. Этот период характеризовался использованием шифров папской канцелярией в Ватикане и развитием тайнописи в арабском мире. Известный ученый Абу Якуб аль-Кинди, живший в начале IX века, сделал важный вклад в область криптоанализа, описав методы расшифровки криптографических сообщений, включая частотный анализ букв для взлома шифров [5].

В эпоху Возрождения, которая простирается с XV по XVI век, криптография пережила настоящий расцвет, подстегнутый активным развитием дипломатии и торговли в Европе. Этот период характеризовался появлением первых полиалфавитных шифров, которые использовали несколько алфавитов для зашифровки сообщений. Примерами таких шифров стали шифр Альберти и шифр Виженера [6], которые обеспечивали более надежную защиту информации. Кроме того, в этот период были разработаны и применены стеганографические методы, которые позволяли скрыть существование сообщения или его содержание. Это включало в себя использование невидимых чернил или скрытых знаков в текстах, что делало сообщения менее заметными для непрошеных глаз.

С XVII века криптография стала неотъемлемой частью военного дела и разведывательной деятельности. Европейские государства широко использовали различные виды шифров для обеспечения конфиденциальности и безопасности секретной информации. В этот период также активно развивались методы криптоанализа, что сделало криптографию еще более сложной и уровень защиты информации более надежным [7].

Французский математик Блез де Виженер в начале XVI века сделал важное открытие, которое оказало существенное влияние на криптографию. Он

продемонстрировал уязвимость полиалфавитных шифров перед методом частотного анализа. Это открытие выявило слабости ранее используемых методов шифрования и способствовало разработке более надежных алгоритмов [8].

Один из таких более надежных методов стал шифром Виженера, названным в честь Блеза де Виженера. Этот шифр использует ключ, состоящий из последовательности букв, чтобы зашифровать сообщение. Ключ изменяется в процессе шифрования, что делает его более сложным для криптоанализа [9]. Схема работы шифра Виженера включает в себя циклическое применение ключа к тексту сообщения, что обеспечивает высокую степень безопасности.

На рисунке 3 показана схема работы шифра Виженера.

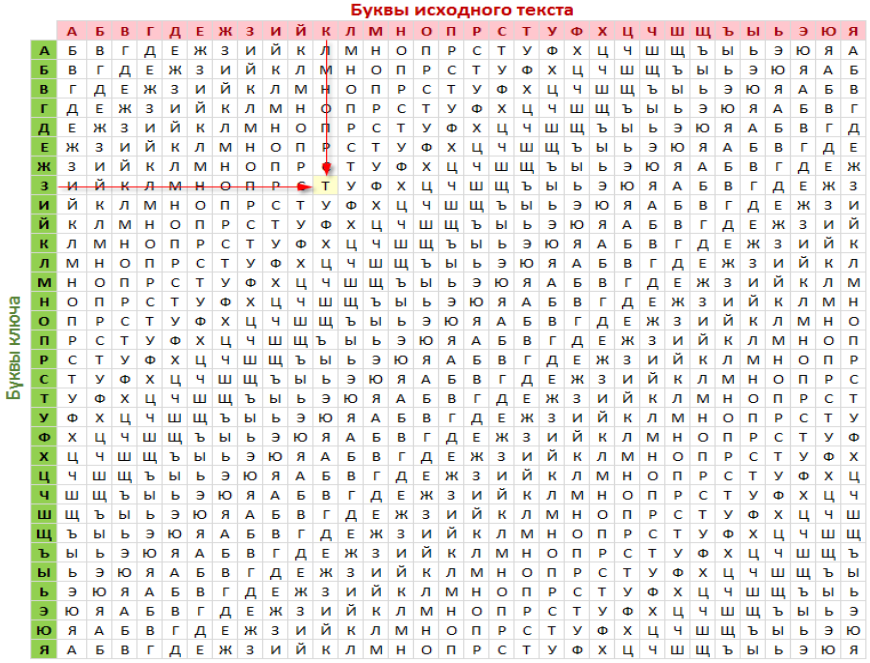

Рисунок 3 – Схема шифра Виженера

В XX веке криптография пережила стремительное развитие, которое было связано с появлением радиосвязи и возможностью массового перехвата

радиограмм. Этот период стал ключевым для развития криптографических методов и технологий, которые играют важную роль в современной кибербезопасности.

Важным моментом стало изобретение первых роторных шифраторов для военных целей в 1915 году. Известными примерами таких устройств стали шифраторы, разработанные Шиллером и Хейберга [10], которые были предтечами знаменитой немецкой шифрмашины Энигма. Эта машина стала символом военной криптографии и была использована нацистами во Второй мировой войне. Взлом Энигмы стал одним из важнейших достижений англо-американских криптографов в течение этого конфликта.

В 1918 году американский инженер Вернам предложил принцип одноразового блокнота, который положен в основу современных генераторов случайных одноразовых ключей. Этот метод позволяет создавать ключи, которые можно использовать только один раз, что делает их практически невозможными для взлома. Одноразовые ключи обеспечивают максимальную степень безопасности и стали важным элементом в современных системах шифрования [11].

Во второй половине XX века криптография пережила революцию, внушительную и важную, обусловленную появлением и активным развитием вычислительной техники. Этот период стал ключевым для области криптографии, поскольку новые технологии требовали более сложных и надежных методов обеспечения безопасности информации. В этот период было разработано множество новых алгоритмов шифрования, в числе которых выделяются такие как Data Encryption Standard (DES), ГОСТ 28147-89, Блоуфиш, и RSA, среди многих других [12]. Эти алгоритмы обеспечивали высокий уровень защиты данных и были широко использованы как в военных, так и в гражданских приложениях.

Важным событием в развитии криптографии стало появление асимметричной криптографии, которая стала основой современных криптосистем с открытым ключом. Эта технология позволила упростить процесс обмена ключами между пользователями и сделала шифрование более удобным и доступным для широкой публики. К концу XX века криптография стала неотъемлемой частью повседневной жизни, особенно с развитием электронной коммерции и массовым использованием сети Интернет. Шифрование стало важным средством защиты личных данных, финансовых транзакций и корпоративной информации. Криптографические методы играют существенную роль в современной кибербезопасности и продолжают развиваться, чтобы бороться с постоянно возрастающими угрозами в сети [13].

Таким образом, исторически криптография прошла путь от примитивных ручных методов шифрования в древности до современных высокотехнологичных решений на основе компьютерных алгоритмов и стандартов шифрования, прочно войдя в информационную инфраструктуру общества. Одним из таких современных решений является российский ГОСТ 34.12-2018 "Кузнечик", рассматриваемый в настоящей работе.

1.2 Основные принципы и методы симметричной криптографии

Симметричная криптография базируется на использовании одного и того же ключа как для зашифровки, так и для расшифровки информации. Она реализует принцип Керкгоффса [14], сформулированный нидерландским криптографом в XIX веке: надежность шифра должна базироваться на секретности ключа, а не алгоритма шифрования.

Среди основных методов симметричного шифрования можно выделить:

– Шифрование потоковыми шифрами. Пример - шифр Вернама-Шеннона [15].

Одним из вариантов реализации потокового шифрования является шифр Вернама-Шеннона, изобретенный инженером Гилбертом Вернамом и математиком Клодом Шенноном в 1917-1949 годах. Его особенностями являются:

а) Использование ключевой последовательности (гаммы) случайных бит, равной по длине открытому тексту;

б) Поразрядная операция сложения открытого текста и гаммы по модулю 2;

в) Множественная гамма, уникальная для каждого сообщения.

Данный алгоритм обеспечивает безусловную криптостойкость при условии использования истинно случайного ключа и его одноразовом применении. На рисунке 4 показана схема шифра Вернама.

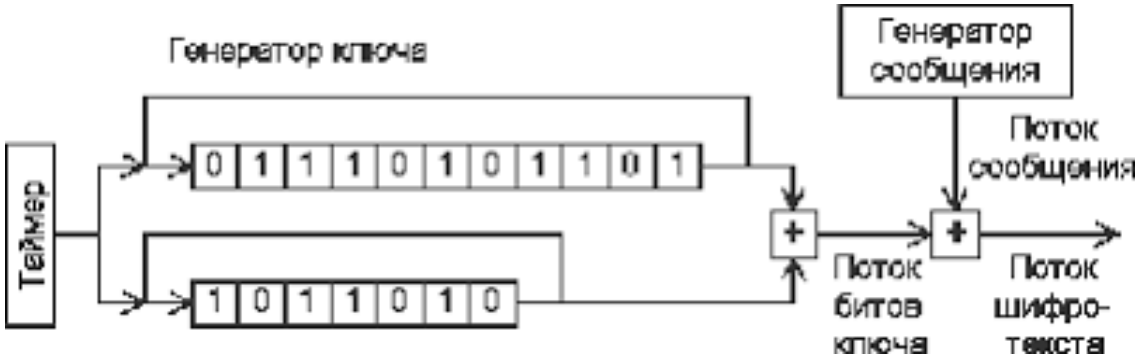

Рисунок 4 – Схема шифра Вернама

Пример шифрования:

– Открытый текст в битах: 01011,

– Случайный ключ (гамма) в битах: 10100,

– Поразрядное сложение: 0 1 0 1 1,

– 1 0 1 0 0 = 1 1 1 1 1.

Получен зашифрованный текст. При повторном сложении зашифрованного текста и гаммы восстанавливается открытый текст.

Шифр Вернама-Шеннона имеет большое теоретическое значение, но ограниченное прикладное применение в связи с трудоемкостью обеспечения абсолютно случайной большой гаммы. Тем не менее его идеи легли в основу более практичных вариантов потокового шифрования в симметричной криптографии. В настоящее время данный метод является эталоном криптостойкого шифрования, к которому стремятся приблизиться более практичные алгоритмы симметричного шифрования, включая рассматриваемый в работе ГОСТ "Кузнечик";

– Шифрование блочными шифрами. Исходный текст делится на блоки фиксированной длины и зашифровывается по отдельности. Блочные алгоритмы шифрования относятся к симметричным шифрам и основаны на разбиении открытого текста на блоки фиксированной длины и последовательном шифровании каждого блока в отдельности.

Классическим и широко известным алгоритмом блочного симметричного шифрования является DES (Data Encryption Standard) - стандарт шифрования данных, разработанный в IBM в 1975 году [16]. Его характеристики:

а) Блоки фиксированной длины 64 бита;

б) Ключ длины 56 бит;

в) 16 раундов одинаковых преобразований блоков с использованием подключей;

г) Сложная схема перестановок и замен на основе S-блоков.

На рисунке 5 показана схема данного шифра.

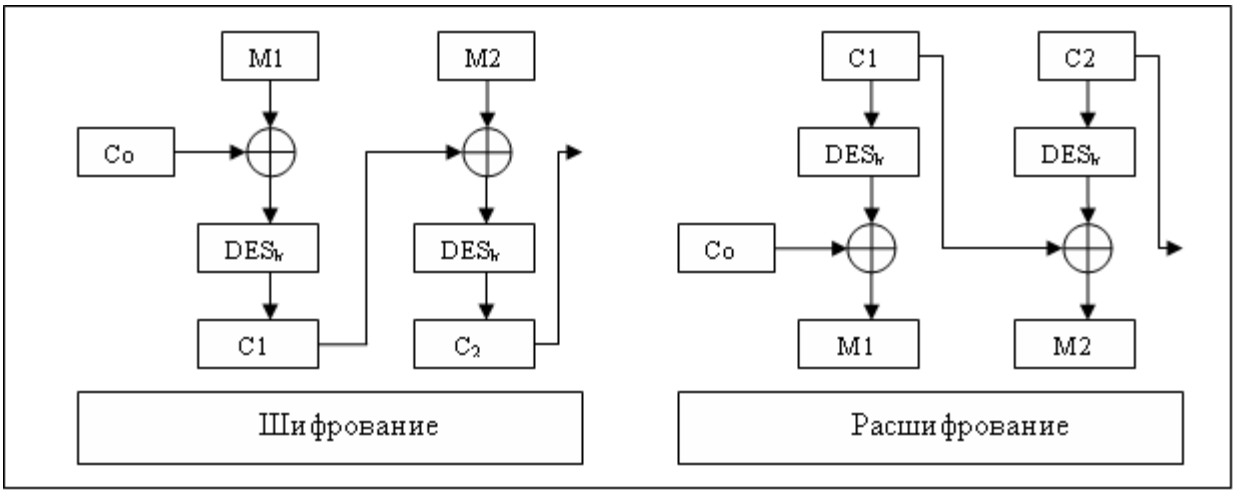

Рисунок 5 – Схема шифра DES

Пример шифрования 64-битного блока открытого текста в DES:

– Начальная перестановка бит согласно стандарту,
– Разбиение на 32-битные половины,
– 16 раундов сложных преобразований с подключами ключа,
– Объединение половин,
– Конечная перестановка бит.

В результате получаем 64-битный зашифрованный блок.

К достоинствам DES можно отнести высокую на тот момент скорость шифрования. К недостаткам - сравнительно небольшой размер ключа, в последствии приведший к уязвимости DES перед атаками полным перебором.

Тем не менее концепция блочного шифрования в целом зарекомендовала себя как эффективная и перспективная. Она легла в основу таких современных алгоритмов как AES, ГОСТ 28147-89 и рассматриваемый в работе алгоритм Кузнечик.

– Шифрование с использованием гаммирования. Открытый текст накладывается на потоковый ключ с помощью поразрядной операции сложения или умножения. Еще одним распространенным методом симметричного шифрования является гаммирование - наложение гаммы (ключевого потока бит) на открытый текст с использованием поразрядных операций.

Данный метод реализован в российском стандарте симметричного блочного шифрования ГОСТ 28147-89 [17]. Его особенности:

- Блочный алгоритм со сквозным зашифрованием текста,
- Блок и ключ по 64 бита,
- 32 раунда преобразований блока,
- На каждом раунде выполняется поразрядное сложение по модулю 2 блока и гаммы,
- Гамма формируется генератором псевдослучайных чисел,

Пример шифрования блока в 64 бита:

- Начальное состояние блока: 00110100 11010100 ... ,
- Генерация 64-битной гаммы, например: 01001011 00110000 ... ,
- Поразрядное сложение блока и гаммы,
- 32 раунда сложных преобразований блока,
- Получен зашифрованный 64-битный блок.

ГОСТ 28147-89 считается криптостоеким алгоритмом и активно применяется на практике вплоть до настоящего времени. Принцип гаммирования текста также использован в более современном российском стандарте Кузнечик, рассматриваемом в данной работе. На рисунке 6 показана схема данного шифра.

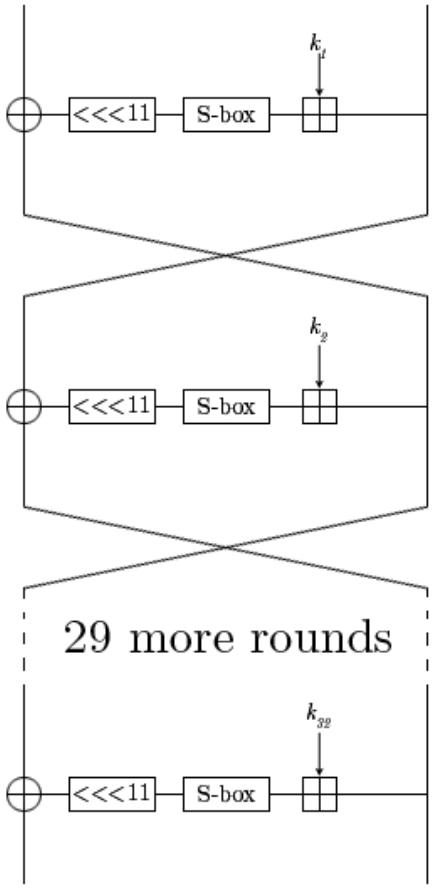

Рисунок 6 – Схема ГОСТ 28147-89

– Режимы блочных шифров. Позволяют преодолеть недостатки блочных алгоритмов, делают шифр устойчивее к взлому [18]. Режимы работы блочных шифров представляют собой различные схемы выполнения

криптографических преобразований, позволяющие устранить некоторые недостатки обычных блочных алгоритмов. К основным режимам относятся:

– Режим простой замены (ECB). Каждый блок шифруется независимо, что не всегда безопасно;

– Режим гаммирования (CTR). Используется потоковое шифрование с генерацией гаммы из счетчика;

– Режим цепочки блоков (CBC). Каждый блок зашифровывается с учетом предыдущего. Устраняет недостаток ECB.

Рассмотрим подробнее режим CBC:

– Берется первый блок открытого текста B1;

– Выполняется операция XOR с предыдущим зашифрованным блоком Z(i-1);

– Полученный блок шифруется, получается Z1;

– Для следующего блока B2 выполняются шаги 2-3 с использованием Z1

– Процесс повторяется до конца.

Таким образом достигается зависимость каждого шифрованного блока от предыдущих, текст шифруется надежнее. Недостаток - ошибка распространяется.

Режимы работы повышают криптостойкость блочных алгоритмов. Они широко используются на практике, в том числе в рассматриваемом в работе алгоритме Кузнечик.

Таким образом, симметричная криптография предоставляет широкий набор инструментов для реализации стойкого шифрования информации с помощью закрытого ключа. Эти методы активно применяются на практике, в том числе в рассматриваемом алгоритме ГОСТ 34.12-2018 "Кузнечик".

1.3 ГОСТ 34.12-2018: характеристика, структура и принципы работы

ГОСТ 34.12-2018 "Информационная технология. Криптографическая защита информации. Блочные шифры" [19] представляет собой национальный стандарт Российской Федерации в области криптографической защиты конфиденциальных данных. Данный стандарт был принят Межгосударственным советом по стандартизации, метрологии и сертификации 29 ноября 2018 года и вступил в силу с 1 июня 2019 года.

ГОСТ 34.12-2018 определяет алгоритм симметричного блочного шифрования под названием "Кузнечик", который пришел на смену предыдущему отечественному стандарту ГОСТ 28147-89. Данный алгоритм обладает рядом ключевых особенностей:

- Является симметричным алгоритмом шифрования, что означает использование единого ключа как для зашифровки, так и для расшифровки информации;
- Относится к блочным симметричным шифрам, то есть исходный открытый текст делится на блоки фиксированной длины, и затем каждый блок шифруется отдельно;
- Размер блока составляет 128 бит или 16 байт, а размер ключа - 256 бит или 32 байта;
- Использует 10 раундов криптографических преобразований блоков открытого текста.

Структурно алгоритм Кузнечик представляет собой SP-сеть (подстановочно-перестановочная сеть). Он состоит из 3 основных преобразований, выполняемых на каждом раунде:

- Нелинейного S-преобразования - выполняется замена значений каждого байта блока при помощи заранее заданной S-подстановки, состоящей из 256 элементов по 256 бит;

– Линейного преобразования Гамма - реализует перестановку байтов блока с использованием операций в поле Галуа;

– Сложения по модулю 2 с раундовым ключом - каждый байт блока складывается по модулю 2 со значением байта раундового ключа.

Принцип работы алгоритма шифрования Кузнечик заключается в следующем:

– 128-битный блок открытого текста разбивается на 16 байт по 8 бит;

– Далее выполняется 10 раундов криптографических преобразований;

– На каждом раунде происходят:

а) Нелинейное S-преобразование блока,

б) Линейное преобразование Гамма,

в) Сложение блока по модулю 2 с 128-битным раундовым ключом.

– После 10 раундов получается зашифрованный 128-битный блок.

Алгоритм расшифровки представляет собой выполнение тех же преобразований в обратном порядке с использованием раундовых ключей в обратном порядке.

Таким образом, алгоритм Кузнечик, заложенный в основу ГОСТ 34.12-2018, относится к SP-сетям типа Кловски-Бихам-Шапиро и является дальнейшим развитием отечественных стандартов криптографической защиты информации. Он обеспечивает высокий уровень стойкости к современным криптоаналитическим атакам при достаточно быстрой скорости работы в программных реализациях. Это делает алгоритм Кузнечик весьма перспективным решением для применения в прикладных информационных системах, требующих надежной защиты конфиденциальных данных с использованием симметричного шифрования.

1.4 Плюсы и минусы алгоритма в современных системах

Применение нового российского криптографического стандарта ГОСТ 34.12-2018 "Кузнечик" в современных информационных системах, несомненно, имеет как ряд весомых преимуществ, так и некоторые потенциальные недостатки, которые необходимо принимать во внимание.

К безусловным достоинствам алгоритма Кузнечик можно отнести следующее [20]:

– Высокий уровень криптостойкости, обеспечивающий надежную защиту информации от угроз несанкционированного доступа и взлома шифра. Это достигается благодаря использованию сбалансированной SP-сети со сложным нелинейным преобразованием S-блока, большому 256-битному ключу и наличию 10 раундов шифрования;

– Высокая производительность за счет эффективности алгоритма, что позволяет использовать Кузнечик для шифрования больших объемов данных. Особенно эффективна аппаратная реализация. По скорости работы на компьютерах Кузнечик превосходит такие алгоритмы как AES, но немного уступает ГОСТ 28147-89 "Магма";

– Преемственность и совместимость с предыдущими российскими стандартами криптографии, так как учитывались требования и особенности использовавшейся ранее "Магмы" для встраивания Кузнечика в существующие отечественные криптосистемы;

– Стандартизированность и открытость алгоритма, что гарантирует отсутствие незадекларированных возможностей (например, для скрытого взлома), а также упрощает анализ кода системы на предмет уязвимостей.

К недостаткам использования Кузнечика можно отнести [21]:

– Относительно бóльший размер шифртекста по сравнению с поточным шифрованием, из-за разбиения открытого текста на 128-битные блоки в соответствии с принципами блочного шифрования;

– Более высокая ресурсоемкость и сложность программной реализации по сравнению с некоторыми другими алгоритмами шифрования в связи с бóльшим количеством необходимых математических преобразований над блоками данных;

– Невозможность в настоящее время дать однозначную оценку стойкости Кузнечика к некоторым специфическим видам криптоатак из-за его относительной новизны в криптографии;

– Тем не менее, очевидно, что плюсы алгоритма Кузнечик значительно перевешивают возможные недостатки. Это делает его весьма перспективным и предпочтительным решением для использования в современных российских информационных системах, предъявляющих высокие требования к скорости и криптостойкости шифрования конфиденциальной информации.

1.5 Угрозы безопасности, против которых направлена защита с использованием ГОСТ 34.12-2018

В условиях стремительного развития цифровых технологий и построения информационного общества все острее встает вопрос об обеспечении надежной защиты конфиденциальных данных от различных угроз в киберпространстве. С этой целью криптографические методы защиты информации, в частности симметричное шифрование в соответствии с отечественным стандартом ГОСТ 34.12-2018, призваны обеспечивать противодействие следующим основным угрозам:

– Несанкционированный доступ и хищение важных конфиденциальных данных в процессе их обработки, передачи и хранения, будь то в госструктурах, компаниях или у отдельных граждан, представляют собой одну из наиболее серьезных угроз информационной безопасности. Эти данные могут включать в себя информацию, составляющую государственную тайну, коммерческие секреты, банковские данные, налоговую информацию, медицинские записи, а также личные данные граждан, такие как имена, адреса, социальные номера и многое другое.

Пример: Злоумышленник, имеющий доступ к базе данных медицинской клиники, может несанкционированно получить доступ к медицинским записям пациентов, включая их диагнозы, лечение и другие конфиденциальные сведения. Это может привести к серьезным последствиям для пациентов, а также к потенциальному медицинскому мошенничеству.

– Целенаправленное искажение критически важных данных, нарушение их целостности, также представляет значительную угрозу. Злоумышленники могут изменять данные, которые используются для принятия важных решений, такие как финансовые операции, государственные реестры и медицинские записи. Это может привести к ошибочным решениям, убыткам и нарушению законов.

Пример: Хакеры могут внести ложные сведения в финансовую систему компании, что может привести к искусственному увеличению прибыли компании и мошенничеству с акциями на фондовом рынке.

– Осуществление несанкционированных действий от имени зарегистрированного пользователя путем подмены аутентичности является хитрым методом атаки. Злоумышленники могут украсть учетные данные пользователя и использовать их для проведения мошеннических операций,

получения доступа к чужим информационным системам и даже к финансовым средствам.

Пример: Злоумышленники могут получить доступ к банковскому аккаунту пользователя и совершать транзакции без его ведома, что может привести к финансовым убыткам для пользователя.

– Блокирование доступа к данным путем их шифрования в результате атаки типа "отказ в обслуживании" может привести к серьезным последствиям. Злоумышленники могут зашифровать важные данные и требовать выкуп за их расшифровку, что может привести к финансовым потерям и нарушению бизнес-процессов.

Пример: Злоумышленники могут зашифровать данные большой компании и потребовать выкуп в криптовалюте, угрожая удалением данных, если выкуп не будет уплачен. Это может привести к значительным финансовым потерям и нарушению бизнес-операций компании.

Применение криптографически стойкого алгоритма Кузнечик в соответствии с требованиями ГОСТ 34.12-2018 позволяет эффективно защищать конфиденциальную информацию от всех перечисленных актуальных угроз в современном цифровом пространстве.

Глава 2 Реализация локальной криптографической системы на основе ГОСТ 34.12-2018

2.1 Анализ требований к системе

Разрабатываемая криптографическая система предназначена для обеспечения конфиденциальности данных пользователя, хранящихся на его локальном персональном компьютере, путем шифрования файлов и другой пользовательской информации с использованием современных криптоалгоритмов российского стандарта ГОСТ.

В качестве основного криптографического примитива в системе используется ГОСТ 34.12-2018 “Кузнечик” - симметричный алгоритм блочного шифрования данных, отвечающий последним требованиям регуляторов в области защиты конфиденциальной информации.

Алгоритм “Кузнечик”, используемый в системе шифрования, предназначен для преобразования открытых данных произвольной длины в зашифрованные данные с целью обеспечения их конфиденциальности. Алгоритм определяет специальное преобразование блоков открытых данных размером 128 бит в блоки шифртекста такого же размера с использованием секретного криптографического ключа заданной длины.

Криптостойкость алгоритма “Кузнечик” при выбранных размерах блока и ключа шифрования соответствует установленным нормам для обеспечения необходимого уровня защищенности пользовательских данных как на текущий момент, так и на обозримую перспективу.

Для обеспечения требуемой криптостойкости в алгоритме “Кузнечик” используется комбинация различных преобразований открытых данных: линейные биективные преобразования, обеспечивающие диффузию структуры открытого текста по шифртексту, а также нелинейные преобразования,

реализующие свойство смешения - комбинирование влияния каждого символа открытого текста на многие символы текста зашифрованного.

Функциональные требования к системе шифрования включают в себя:

– Обеспечение возможности шифрования пользовательских файлов и текстовых данных непосредственно на компьютере пользователя с сохранением зашифрованных данных в файлах. При этом пользователь может указать произвольный каталог на локальном диске для сохранения зашифрованных данных;

– Возможность расшифровки ранее зашифрованных данных на компьютере пользователя после корректного ввода ключа расшифрования. Расшифрованные исходные данные также сохраняются в виде файлов в выбранной пользователем файловой системе;

– Графический пользовательский интерфейс, обеспечивающий простоту и наглядность процесса шифрования и расшифрования данных для обычных пользователей, не обладающих специальными познаниями в области криптографии;

– Автоматическая генерация случайных ключей шифрования заданной длины с равновероятным выбором каждого символа ключа и статистической независимостью символов;

– Возможность задания пользователем собственного ключа шифрования в виде последовательности шестнадцатеричных символов определенной длины;

– Генерация необходимых криптографических констант и параметров алгоритма шифрования, таких как S-блоки нелинейных преобразований, а также раундовые ключи шифрования на основе ключа, заданного пользователем;

– Реализация основных режимов блочного шифрования: режим простой замены, режим гаммирования, режим гаммирования с обратной связью по шифртексту, режим сцепления блоков шифра с обратной связью и других.

– Шифрование данных блоками по 128 бит с дополнением последнего блока нулями до полного размера в случае необходимости;

Нефункциональные требования к системе в первую очередь включают:

– Высокая скорость шифрования на компьютерах пользователя за счет оптимизации алгоритмов и эффективного распараллеливания вычислений;

– Кроссплатформенность системы: защищенность данных должна обеспечиваться на компьютерах с различными операционными системами;

– Масштабируемость системы шифрования для работы как с небольшими объемами данных на мобильных устройствах, так и с большими объемами данных на мощных серверных станциях;

– Простота интеграции системы шифрования с другими программными комплексами для расширения сферы применения за счет открытых программных интерфейсов.

Бизнес-ценность разрабатываемой системы шифрования данных заключается в предоставлении пользователям криптографически стойкого инструмента защиты их личных и рабочих данных от несанкционированного доступа, соответствующего современным требованиям безопасности.

2.2 Проектирование архитектуры системы

При проектировании комплексных программных систем важно заранее определить архитектуру - структуру системы, ее основные функциональные блоки и интерфейсы между ними. Архитектура задает общие принципы построения всей системы и позволяет разрабатывать отдельные модули

независимо в соответствии с выбранными интерфейсами. Гибкая архитектура критически важна для возможности дальнейшего масштабирования функциональности при изменении требований к системе.

Рассматриваемая система криптографической защиты файлов на основе российских стандартов шифрования ГОСТ предназначена для обеспечения конфиденциальности файлов пользователя на его локальном персональном компьютере посредством прозрачного шифрования выбранных файлов и директорий, с возможностью последующей безопасной передачи закрытых данных.

Исходя из функциональных и нефункциональных требований к системе, в ее состав целесообразно выделить следующие основные модули, разрабатываемые автономно:

– Модуль пользовательского интерфейса. Реализует графический интерфейс взаимодействия с пользователем на основе библиотеки Tkinter или Qt. Предоставляет пользователю удобные средства для выбора файлов и каталогов, ввода паролей, запуска процессов шифрования и расшифровки, а также визуальные индикаторы состояния работы системы;

– Менеджер шифрования данных. Ядро системы, осуществляющее взаимодействие всех компонентов процесса, инициализацию алгоритмов шифрования по пользовательским параметрам, разбиение данных на блоки нужного размера, обработку исключений;

– Модуль криптопреобразований ГОСТ. Реализует основные алгоритмы – генерацию ключей по ГОСТ 34.10-2012, непосредственно шифрование и расшифрование блоков данных алгоритмом Кузнечик по ГОСТ Р 34.12-2015 согласно параметрам инициализации;

– Модуль хранения зашифрованных данных и аудита. Отвечает за сохранение зашифрованных блоков данных, а также хранение протоколов и

журналов безопасности о проведенных операциях шифрования и доступа к закрытым файлам;

– Модуль администрирования системы шифрования. Предоставляет возможности настройки параметров шифрования, управления доступом и ролями пользователей в единой системе защиты данных организации через административный веб-интерфейс.

Предложенная модульная архитектура позволит легко масштабировать систему шифрования под нужды конкретной организации, наращивая функциональность программного комплекса. Ниже показана схема данной архитектуры.

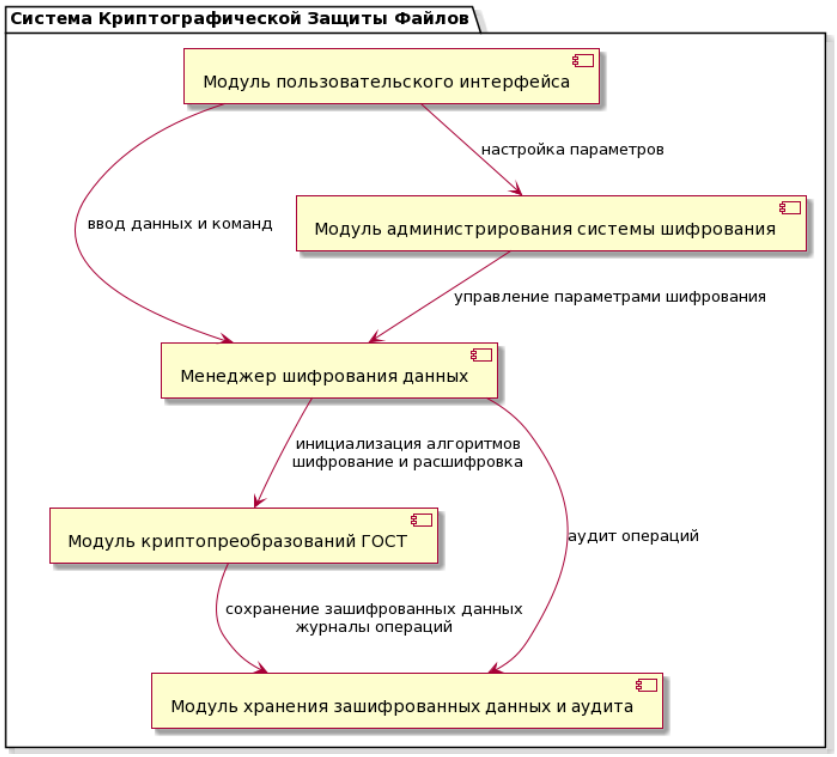

Рисунок 7 – Схема системы

Интерфейс пользователя играет ключевую роль в системе шифрования файлов, определяя удобство работы конечных пользователей с приложением. Четко спроектированный интерфейс позволяет быстро освоиться даже новичкам и свести количество ошибок при шифровании данных к минимуму.

Пользовательский интерфейс системы шифрования целесообразно выполнить в виде настольного приложения с графическим интерфейсом на основе библиотеки Tkinter или Qt. Это обеспечит кроссплатформенность графической оболочки и возможность запуска на самых распространенных операционных системах, включая Windows, Mac OS и различные дистрибутивы Linux. При этом дополнительно можно будет разработать веб-интерфейс для удаленного управления шифрованием данных через защищенное HTTPS-соединение.

Графический пользовательский интерфейс приложения шифрования файлов должен быть интуитивно понятным для пользователя, включая:

– Удобные элементы навигации для выбора файлов и каталогов для шифрования – древовидные структуры с возможностью множественного выделения, поиском, фильтрацией;

– Панель быстрого запуска основных операций – шифрования, расшифровки, генерации новых ключей шифрования и др;

– Таблицы и списки для отображения ключей шифрования и статусов закрытых файлов;

– Визуальные индикаторы состояния процесса шифрования файлов: прогресс-бар, анимация;

– Удобные пользовательские диалоги для ввода паролей и настройки параметров шифрования перед стартом;

– Всплывающие подсказки и уведомления для информирования пользователя о текущем состоянии работы приложения и о возникающих ошибках.

На рисунке 8 показан один из возможных вариантов такого интерфейса.

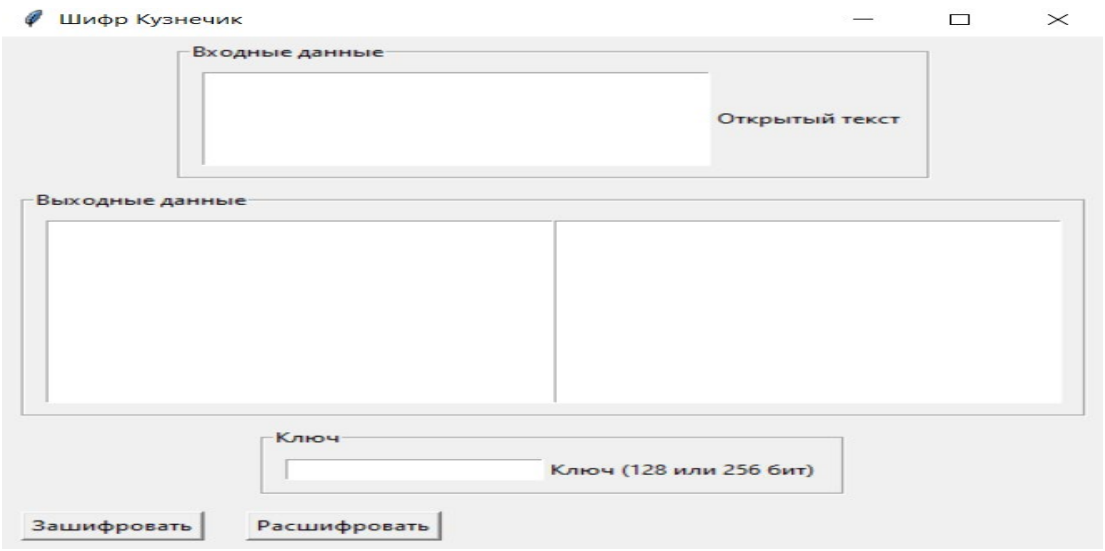

Рисунок 8 – Пример интерфейса

Грамотное проектирование интерфейса взаимодействия с пользователем позволит свести к минимуму ошибки и сэкономить массу времени на шифрование файлов в повседневной работе с системой.

2.3 Реализация ключевых функций

Рассматриваемая система шифрования файлов использует в своей работе криптографические примитивы российского стандарта ГОСТ Р 34.12-2015 “Кузнечик”. Данный симметричный алгоритм блочного шифрования оперирует

с блоками открытого и зашифрованного текста длиной 128 бит и ключами длиной 256 или 512 бит.

Центральным компонентом системы является модуль реализации криптографических преобразований данных. Он содержит функции шифрования и расшифрования пользовательских данных, а также смежные процедуры формирования раундовых ключей шифрования и других необходимых параметров.

Генерация ключей шифрования выполняется криптографически стойким генератором псевдослучайных чисел на основе датчика случайных физических событий, имеющегося в современных операционных системах. Сформированная 256- или 512-битная последовательность затем используется в качестве первичного ключа алгоритма Кузнечик.

На основе заданного пользователем первичного ключа шифрования в блоке расчета раундовых ключей вырабатывается десять 128-битных раундовых подключей. Это делается путем вычисления специального линейного преобразования L от половин первичного ключа и их комбинаций. Список ключей показан ниже.

```
     pi                                  =                                  [
    252, 238, 221, 17, 207, 110, 49, 22, 251, 196, 250, 218, 35, 197, 4,
77,
    233, 119, 240, 219, 147, 46, 153, 186, 23, 54, 241, 187, 20, 205,
95,                                                                    193,
    249, 24, 101, 90, 226, 92, 239, 33, 129, 28, 60, 66, 139, 1, 142,
79,                                                                      5,
    132, 2, 174, 227, 106, 143, 160, 6, 11, 237, 152, 127, 212, 211, 31,
235,
    52, 44, 81, 234, 200, 72, 171, 242, 42, 104, 162, 253, 58, 206, 204,
181,
    112, 14, 86, 8, 12, 118, 18, 191, 114, 19, 71, 156, 183, 93, 135,
21,                                                                    161,
```

```
    150, 41, 16, 123, 154, 199, 243, 145, 120, 111, 157, 158, 178, 177,
50,                                                                 117,
    25, 61, 255, 53, 138, 126, 109, 84, 198, 128, 195, 189, 13, 87, 223,
245,
    36, 169, 62, 168, 67, 201, 215, 121, 214, 246, 124, 34, 185, 3, 224,
15,
    236, 222, 122, 148, 176, 188, 220, 232, 40, 80, 78, 51, 10, 74, 167,
151,
    96, 115, 30, 0, 98, 68, 26, 184, 56, 130, 100, 159, 38, 65, 173, 69,
70,
    146, 39, 94, 85, 47, 140, 163, 165, 125, 105, 213, 149, 59, 7, 88,
179,                                                                64,
    134, 172, 29, 247, 48, 55, 107, 228, 136, 217, 231, 137, 225, 27,
131,                                                                73,
    76, 63, 248, 254, 141, 83, 170, 144, 202, 216, 133, 97, 32, 113,
103,                                                               164,
    45, 43, 9, 91, 203, 155, 37, 208, 190, 229, 108, 82, 89, 166, 116,
210,                                                               230,
    244,  180,  192,  209,  102,  175,  194,  57,  75,  99,  182
]

#                non-lin                  table                  reversed
pi_rev  =  [0xA5,  0x2D,  0x32,  0x8F,  0x0E,  0x30,  0x38,  0xC0,
          0x54,  0xE6,  0x9E,  0x39,  0x55,  0x7E,  0x52,  0x91,
          0x64,  0x03,  0x57,  0x5A,  0x1C,  0x60,  0x07,  0x18,
          0x21,  0x72,  0xA8,  0xD1,  0x29,  0xC6,  0xA4,  0x3F,
          0xE0,  0x27,  0x8D,  0x0C,  0x82,  0xEA,  0xAE,  0xB4,
          0x9A,  0x63,  0x49,  0xE5,  0x42,  0xE4,  0x15,  0xB7,
          0xC8,  0x06,  0x70,  0x9D,  0x41,  0x75,  0x19,  0xC9,
          0xAA,  0xFC,  0x4D,  0xBF,  0x2A,  0x73,  0x84,  0xD5,
          0xC3,  0xAF,  0x2B,  0x86,  0xA7,  0xB1,  0xB2,  0x5B,
          0x46,  0xD3,  0x9F,  0xFD,  0xD4,  0x0F,  0x9C,  0x2F,
          0x9B,  0x43,  0xEF,  0xD9,  0x79,  0xB6,  0x53,  0x7F,
```

```
0xC1,   0xF0,   0x23,   0xE7,   0x25,   0x5E,   0xB5,   0x1E,
0xA2,   0xDF,   0xA6,   0xFE,   0xAC,   0x22,   0xF9,   0xE2,
0x4A,   0xBC,   0x35,   0xCA,   0xEE,   0x78,   0x05,   0x6B,
0x51,   0xE1,   0x59,   0xA3,   0xF2,   0x71,   0x56,   0x11,
0x6A,   0x89,   0x94,   0x65,   0x8C,   0xBB,   0x77,   0x3C,
0x7B,   0x28,   0xAB,   0xD2,   0x31,   0xDE,   0xC4,   0x5F,
0xCC,   0xCF,   0x76,   0x2C,   0xB8,   0xD8,   0x2E,   0x36,
0xDB,   0x69,   0xB3,   0x14,   0x95,   0xBE,   0x62,   0xA1,
0x3B,   0x16,   0x66,   0xE9,   0x5C,   0x6C,   0x6D,   0xAD,
0x37,   0x61,   0x4B,   0xB9,   0xE3,   0xBA,   0xF1,   0xA0,
0x85,   0x83,   0xDA,   0x47,   0xC5,   0xB0,   0x33,   0xFA,
0x96,   0x6F,   0x6E,   0xC2,   0xF6,   0x50,   0xFF,   0x5D,
0xA9,   0x8E,   0x17,   0x1B,   0x97,   0x7D,   0xEC,   0x58,
0xF7,   0x1F,   0xFB,   0x7C,   0x09,   0x0D,   0x7A,   0x67,
0x45,   0x87,   0xDC,   0xE8,   0x4F,   0x1D,   0x4E,   0x04,
0xEB,   0xF8,   0xF3,   0x3E,   0x3D,   0xBD,   0x8A,   0x88,
0xDD,   0xCD,   0x0B,   0x13,   0x98,   0x02,   0x93,   0x80,
0x90,   0xD0,   0x24,   0x34,   0xCB,   0xED,   0xF4,   0xCE,
0x99,   0x10,   0x44,   0x40,   0x92,   0x3A,   0x01,   0x26,
0x12,   0x1A,   0x48,   0x68,   0xF5,   0x81,   0x8B,   0xC7,
0xD6, 0x20, 0x0A, 0x08, 0x00, 0x4C, 0xD7, 0x74]
```

Блок непосредственно шифрования данных реализует последовательное вычисление операций сложения текущего блока данных с очередным раундовым ключом, затем нелинейного S-преобразования блока и наконец линейной перестановки бит с помощью L-преобразования. Такая последовательность повторяется в 9 раундах, после чего происходит сложение с последним ключом.

Процедура расшифрования использует те же модули и блоки, но в последовательности, обратной шифрованию, начиная с первичного и кончая первым раундовым ключом.

Ниже показана реализация данных модулей в языке программирования Python.

```
     # key 256 bit
# gen 10 keys, 128 bit each
def gen_iter_keys(k):
    keys = []
    a = k >> 128
    b = k & (2 ** 128 - 1)
    keys.append(a)  # first 2 keys from basic key
    keys.append(b)  #

    # gen 10 iter keys
    for i in range(4):
        for j in range(8):  # 8 iterations of Feistels network
            c = L(8 * i + j + 1)  # calc C v2 (L ot nomera iterazii)
            # print('iter c:', hex(c))
            (a, b) = (L(S(a ^ c)) ^ b, a)
        keys.append(a)
        keys.append(b)
    print('iterations keys in hex form:', [hex(x) for x in keys])
    return keys


# x (text to cipher) 128
# key is 256
def kuznyechik_encrypt(x, k):
    keys = gen_iter_keys(k)
    for round in range(9):  # 9 rounds
        x = x ^ keys[round]  # xor with iterations key
        x = S(x)  # non-lin
        x = L(x)  # lin
    return x ^ keys[-1]  # xor
```

```
# encoded text x is 128 bit
# key is 256
def kuznyechik_decrypt(x, k):
    keys = gen_iter_keys(k)
    keys.reverse()
    for round in range(9):  # 9 rounds
        x = S_inv(L_inv(x ^ keys[round]))  # same, but using inversing
    return x ^ keys[-1]  # xor
```

Реализация перечисленных модулей и блоков позволяет выполнить шифрование данных пользователя по алгоритму ГОСТ Р 34.12-2015 Кузнечик в соответствии с требованиями отечественных стандартов.

2.4 Тестирование и верификация системы

После реализации основных функций системы шифрования файлов на основе российского криптостандарта ГОСТ Р 34.12-2015 Кузнечик требуется провести ее всестороннее тестирование с целью выявления возможных дефектов и отклонений от эталонной работы алгоритма.

Тестирование следует выполнять с использованием специальных тестовых наборов открытых и закрытых данных, а также тестовых значений ключей шифрования. Необходимо максимально полно проверить работу всех режимов шифрования данных (простая замена, гаммирование, сцепление блоков), длины блоков и ключей.

Особое внимание требуется уделить тестированию процедур генерации случайных ключей и их распределению – ключи должны иметь высокую энтропию и равновероятность битов для обеспечения криптостойкости.

Помимо функционального тестирования криптографических преобразований необходимо убедиться в выполнении нефункциональных требований к системе: быстродействие шифрования на целевой аппаратуре соответствует нормам стандарта, удобство использования интерфейса приложения, отсутствие утечек памяти и иных дефектов.

По результатам тестов следует убедиться в полном соответствии реализации алгоритмов шифрования требованиям российского стандарта криптографической защиты ГОСТ Р 34.12-2015. Это позволит гарантировать корректную работу системы шифрования пользовательских файлов на практике.

– Запуск приложения показан на рисунке 9.

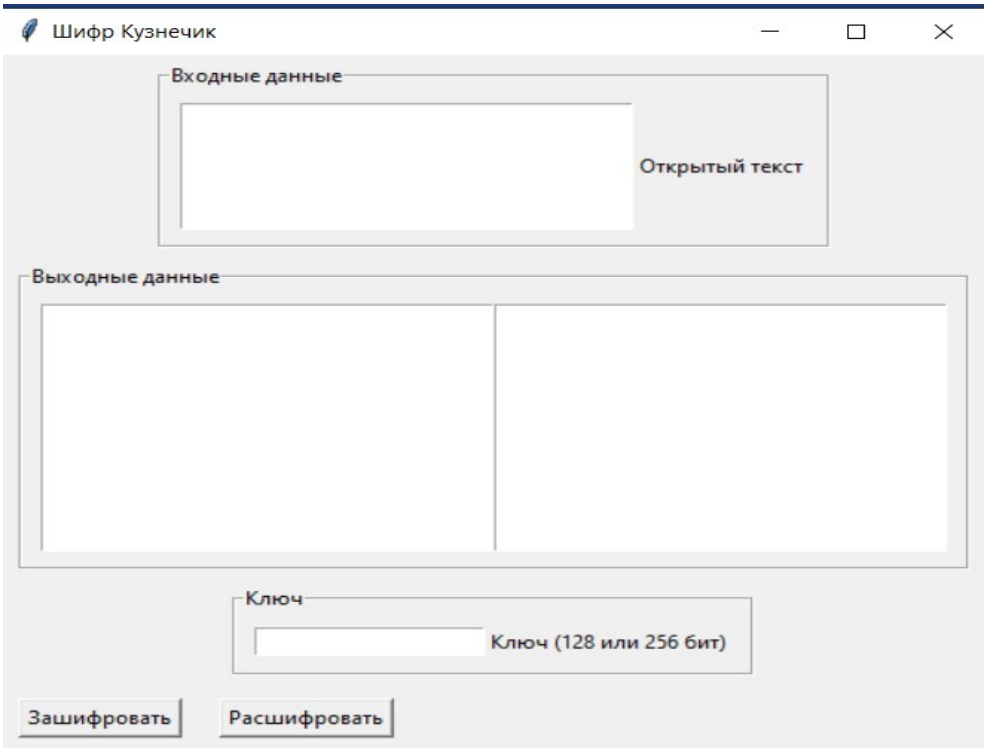

Рисунок 9 – Стартовое окно

– Ввод данных показан на рисунке 10.

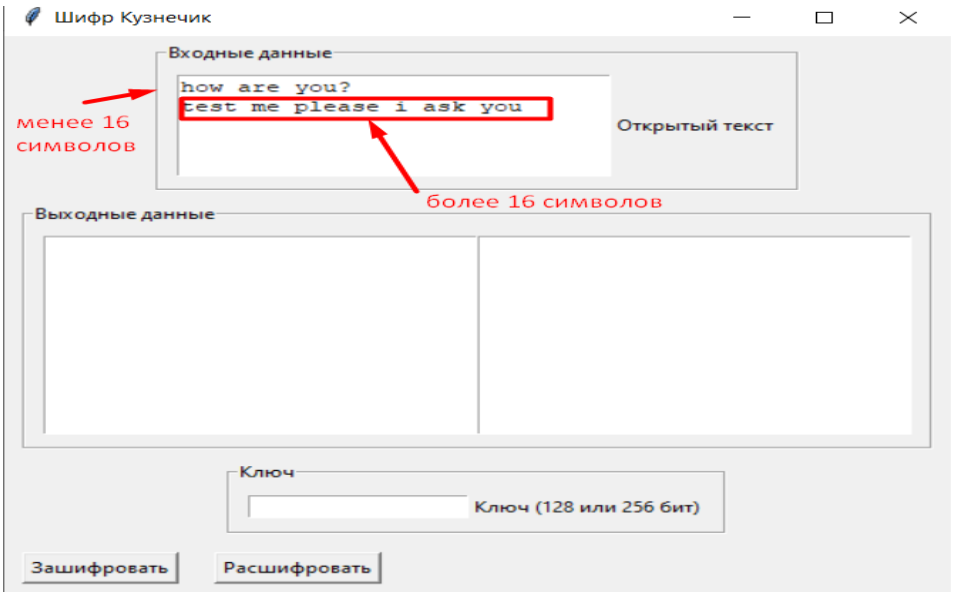

Рисунок 10 – Ввод данных разной размерности

– Выбор ключа показан на рисунке 11.

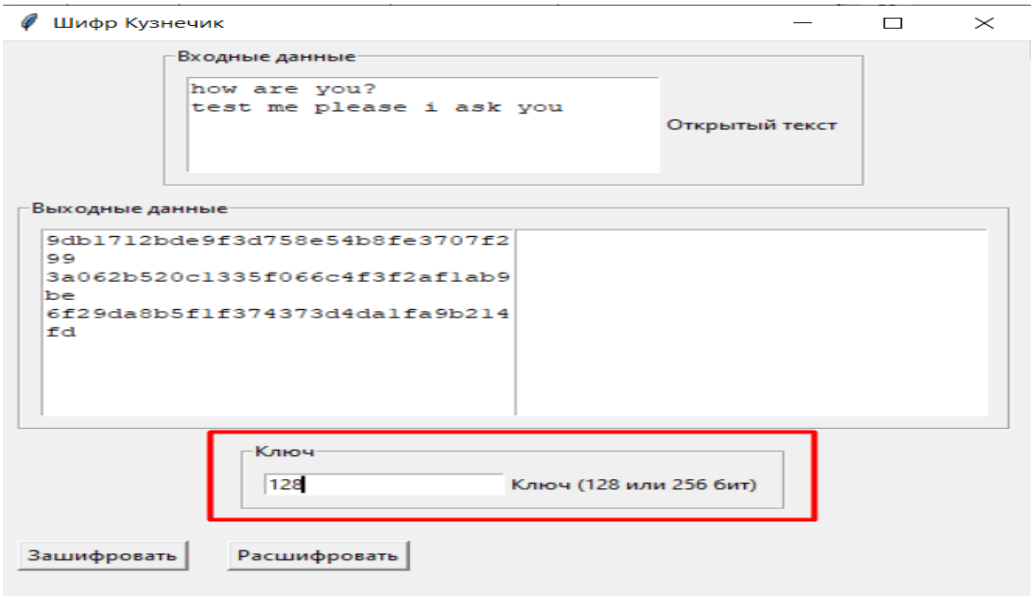

Рисунок 11 – Выбор ключа

– Проверка зашифровки\расшифровки с ключом 128 бит показан на рисунке 12.

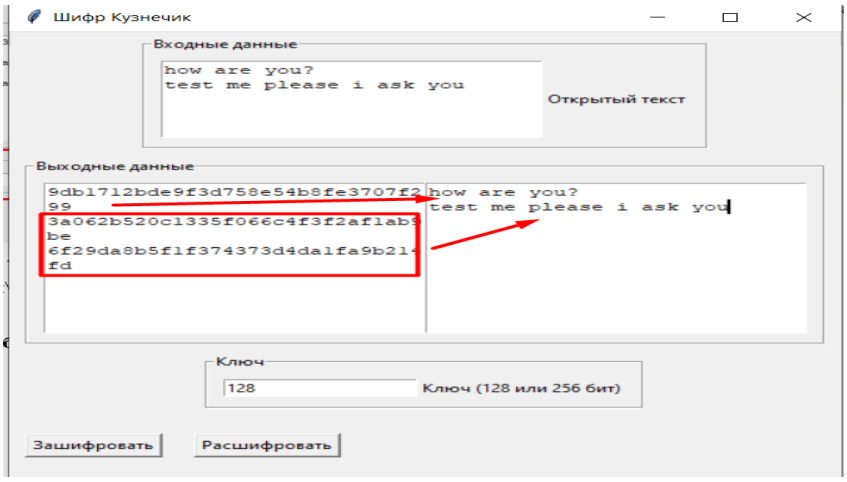

Рисунок 12 – Зашифровка\расшифровка при ключе в 128 бит

– Проверка зашифровки\расшифровки с ключом 256 бит показан на рисунке 13.

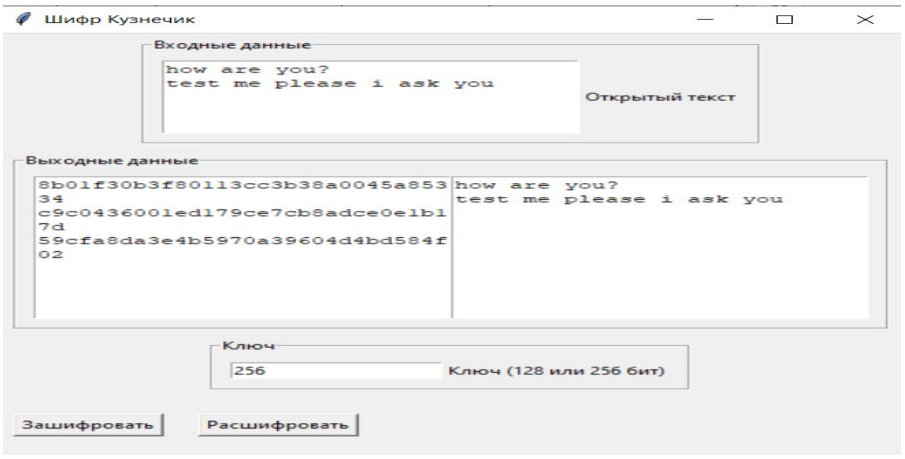

Рисунок 13 – Зашифровка\расшифровка при ключе в 256 бит

Как видим, все требуемые модули и функции работают корректно.

2.5 Меры по безопасности в процессе эксплуатации системы

Создание и успешное внедрение системы шифрования данных в корпоративной среде предполагает не только разработку технических решений по защите информации, но и внимательное внедрение организационных и технических мер по обеспечению безопасности. Эти меры направлены на гарантирование конфиденциальности, целостности и доступности данных заказчика. Важно рассмотреть этот аспект подробно.

Организационные меры:

– Регламентация процессов шифрования и расшифровки: Прежде всего, необходимо установить четкие правила и процедуры для шифрования и расшифровки данных во внутренних документах. Это включает в себя определение круга сотрудников, которые имеют доступ к этим процессам, а также их полномочия.

– Учет действий с закрытыми файлами и ключами: Организация жесткого учета всех действий, связанных с закрытыми файлами и ключами, является неотъемлемой частью обеспечения безопасности. Разграничение полномочий между ответственными за шифрование данных и теми, кто имеет доступ к ключам, помогает минимизировать риски.

Технические меры:

– Регулярная смена ключей шифрования: Один из критически важных аспектов в технической безопасности - регулярная смена ключей шифрования. Рекомендуется проводить смену, например, каждую неделю. Это повышает уровень защиты данных.

– Источники случайности для генерации ключей: Для генерации ключей шифрования следует использовать надежные источники случайности.

Это помогает предотвратить предсказуемость ключей и, следовательно, усилить безопасность.

– Защита секретных ключей: Секретные ключи не должны храниться в явном виде и должны быть защищены от несанкционированного доступа. Их управление и хранение требуют специальных мер безопасности.

– Регулярные архивирования ключей и данных: Для обеспечения возможности восстановления в случае нештатных ситуаций необходимо регулярно архивировать закрытые ключи и данные. Это гарантирует сохранность информации и бесперебойную работу системы.

– Обновление программного обеспечения: Установка последних обновлений программного обеспечения системы шифрования необходима для исправления уязвимостей и обеспечения актуальной защиты.

– Внедрение комплекса организационных и технических мер по обеспечению безопасности является неотъемлемой частью эксплуатации системы шифрования данных. Такой комплексный подход позволяет минимизировать риски нарушения информационной безопасности и обеспечивать надежную защиту конфиденциальной информации заказчика.

Глава 3 Экономическая часть

3.1 Расчет экономической эффективности внедрения системы

Расчет экономической эффективности внедрения системы шифрования информации является неотъемлемой частью стратегического планирования и обеспечивает не только финансовую прозрачность проекта, но и его обоснование с точки зрения целесообразности вложений в информационную безопасность. Давайте подробно рассмотрим каждый аспект данного расчета, углубившись в его составляющие.

– Начальные вложения:

Первоначальные инвестиции представляют собой финансовые ресурсы, которые необходимы для создания системы шифрования данных. Согласно имеющимся данным, стоимость разработки "с нуля" составляет 300 000 рублей. Эти средства выступают в роли первоначального капитала, требуемого для инициации проекта. Кроме того, не следует забывать о затратах на внедрение решения, которые оцениваются в 150 000 рублей. Эти затраты включают в себя расходы на интеграцию системы в существующую инфраструктуру заказчика;

– Ежегодные эксплуатационные издержки:

Система шифрования данных требует постоянного обслуживания и поддержки. Техническая поддержка системы оценивается в 200 000 рублей в год. Эта сумма включает в себя затраты на обучение персонала, решение технических проблем и обновление программного обеспечения.

Дополнительно, стоимость обновления лицензий программного обеспечения составляет 50 000 рублей в год. Эти средства необходимы для обеспечения актуальности и безопасности используемых программных решений.

Однако следует отметить, что фонд оплаты труда (ФОТ) обслуживающего персонала представляет собой значительную составляющую затрат. Два сотрудника, получающих по 50 000 рублей зарплаты в месяц, обеспечивают надежное функционирование системы. Общая годовая стоимость оплаты труда для этого персонала составляет 1 200 000 рублей;

– Общая стоимость владения (ТСО):

Суммируя ежегодные эксплуатационные затраты, мы получаем общую стоимость владения (ТСО) системой. При учете расчетного срока амортизации системы в 5 лет, общая стоимость владения составляет 5,75 миллиона рублей. Это ключевой показатель показан в таблице 1 , который позволяет определить, насколько обоснованы затраты на внедрение и эксплуатацию системы шифрования.

Таблица 1 – Распределение начальных вложений и ежегодных эксплуатационных издержек по компонентам

Компонент	Сумма (рублей)
Начальные вложения	450 000
Ежегодные затраты	1 450 000
Общая стоимость владения (ТСО)	5750 000

Данные значения позволяют нам определить общую финансовую нагрузку, связанную с системой шифрования данных, и предоставляют основу для дальнейших экономических расчетов и анализа эффективности внедрения данной системы на рисунке 14 показано распределение финансовых затрат в графическом виде.

Рисунок 14 – Распределение финансовых затрат

Экономический эффект внедрения системы шифрования данных для защиты информации заказчика заслуживает детального анализа и обоснования. Эта система не только служит средством обеспечения информационной безопасности, но и представляет собой инвестицию в будущее. Давайте более подробно рассмотрим факторы, которые влияют на экономическую эффективность этой системы.

– Предотвращение ущерба от утечек данных:

Одним из ключевых аспектов экономического эффекта является предотвращение ущерба, который может возникнуть в результате утечки конфиденциальных данных. Согласно экспертным оценкам, ущерб от одной утечки информации может достигать нескольких десятков миллионов рублей. Этот ущерб включает в себя штрафы, потерю клиентов и репутационные потери. На таблице 2 показан расчет потенциального ущерба от утечки данных за пять лет, если система шифрования не была бы внедрена.

Таблица 2 – Расчет потенциального ущерба от утечки данных за 5 лет

Год	Потенциальный ущерб (млн рублей)
Год 1	30
Год 2	35
Год 3	40
Год 4	45
Год 5	50
Итого	200

Как видно из таблицы, потенциальный ущерб от утечки данных за пять лет составляет 200 миллионов рублей. Это значительная сумма, которая может серьезно поставить под угрозу бизнес заказчика. На рисунке 15 показан потенциальный ущерб в графическом виде.

Рисунок 15 – Потенциальный ущерб

– Быстрая окупаемость:

Система шифрования данных представляет собой не только средство обеспечения безопасности, но и инвестицию в будущее. Ее быстрая

окупаемость является одним из важных факторов экономической эффективности. Рассмотрим в таблице 3 расчет окупаемости проекта в течение пяти лет, учитывая начальные вложения и ежегодные эксплуатационные затраты.

Таблица 3 – Расчет окупаемости проекта за 5 лет

Год	Начальные вложения (млн рублей)	Ежегодные затраты (млн рублей)	Итого (млн рублей)
Год 1	0,45	1,45	1,90
Год 2	0,00	1,45	1,45
Год 3	0,00	1,45	1,45
Год 4	0,00	1,45	1,45
Год 5	0,00	1,45	1,45
Итого	0,45	7,25	7,70

Из таблицы видно, что начальные вложения в размере 0,45 миллиона рублей окупаются уже к концу первого года эксплуатации системы. В долгосрочной перспективе экономический эффект от внедрения системы шифрования становится более очевидным. Для большей наглядности на рисунке 16 представлен график, показывающий расчет окупаемости.

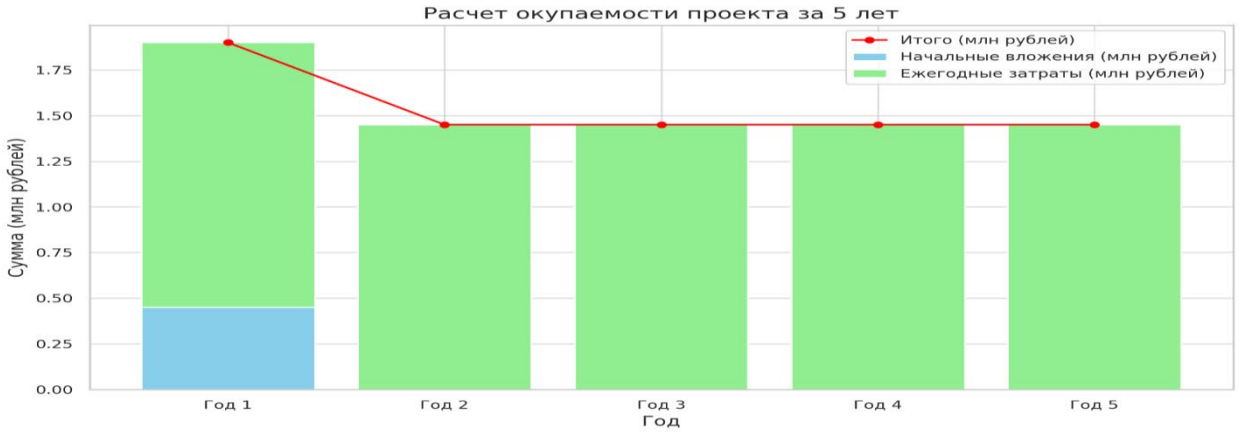

Рисунок 16 – Расчет окупаемости

– Защита от потенциальных угроз:

Система шифрования данных обеспечивает заказчика защитой от потенциальных угроз и рисков, связанных с утечками данных. Это важно не только для сохранения конфиденциальности информации, но и для обеспечения устойчивости бизнеса заказчика. Репутационные потери и потеря клиентов могут оказаться критическими для долгосрочного успеха компании.

В результате система шифрования данных представляет собой не только средство обеспечения безопасности, но и инвестицию в будущее. Экономический эффект ее внедрения является очевидным, учитывая высокую стоимость информационных активов заказчика и риски, связанные с возможными утечками данных.

3.2 Оценка рисков и возможных экономических убытков от несанкционированного доступа

Анализ рисков информационной безопасности, связанных с возможными утечками конфиденциальных данных, представляет собой неотъемлемый этап стратегического управления в современных бизнес-условиях, где цифровая трансформация играет ключевую роль. Этот анализ требует глубокого понимания потенциальных угроз и их количественной оценки, что позволяет более точно определить необходимость внедрения специализированных систем шифрования данных.

– Прямые финансовые потери:

Наиболее очевидной и ощутимой формой риска являются прямые финансовые потери, которые могут возникнуть вследствие нарушения законодательных требований в области защиты данных. Регулирующие органы, такие как Федеральная служба по надзору в сфере связи, информационных

технологий и массовых коммуникаций, имеют право налагать штрафы на компании, которые допускают утечки персональных данных и конфиденциальной информации. Эти штрафы, достигающие до 6 миллионов рублей за каждый отдельный инцидент утечки данных, могут оказаться значительной финансовой нагрузкой на компанию. Рассмотрим таблицу 1, где приведен пример расчета финансовых потерь от штрафов за утечку данных.

Таблица 4 – Расчет финансовых потерь от штрафов за утечку данных

Год	Количество инцидентов утечки	Сумма штрафов (млн рублей)
Год 1	2	12
Год 2	3	18
Год 3	1	6
Год 4	4	24
Год 5	2	12
Итого	12	72

Также данные представлены на рисунке 17 в графическом виде.

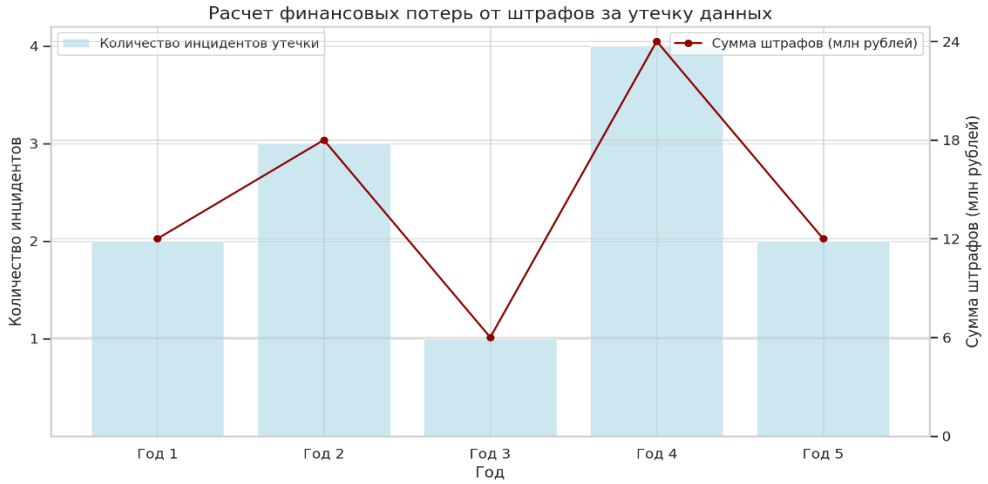

Рисунок 17 – Расчет финансовых потерь

Из таблицы видно, что финансовые потери от штрафов за утечку данных могут достигать значительной суммы, исчисляемой в миллионах рублей. Эти потери имеют прямое влияние на финансовое состояние компании и ее возможность реализовать стратегические цели.

– Репутационные убытки:

Однако риски не ограничиваются только финансовыми потерями. Репутационные убытки также играют критическую роль в оценке рисков информационной безопасности. Утечки данных, получившие широкое освещение в СМИ и профессиональном сообществе, могут нанести серьезный ущерб бренду компании и снизить лояльность клиентов. Репутационный ущерб может проявиться в форме потери доверия к компании, уменьшения объема заказов и даже оттока клиентов. Рассмотрим таблицу 2, где приведен пример расчета репутационных убытков от утечки данных.

Таблица 5 – Расчет репутационных убытков от утечки данных

Год	Убыток от потери клиентов (млн рублей)	Репутационные потери (млн рублей)
Год 1	5	10
Год 2	7	15
Год 3	4	8
Год 4	8	16
Год 5	6	12
Итого	30	61

Из таблицы видно, что репутационные убытки могут оказаться более значительными, чем финансовые потери. Общие репутационные потери за пять лет могут составить 61 миллион рублей, что серьезно влияет на долгосрочную конкурентоспособность компании.

– Потеря ценных объектов интеллектуальной собственности:

Третьим аспектом, который следует учитывать, является потеря ценных объектов интеллектуальной собственности компании. Утечка данных, таких как секретные алгоритмы, топологии микросхем или стратегические планы развития, может привести к утрате значительных потенциальных прибылей. Эти потери напрямую зависят от стоимости и коммерческого потенциала таких утраченных активов и могут достигать сотен миллионов рублей. Таблица 6 приводит пример расчета потерь от утраты ценных объектов интеллектуальной собственности.

Таблица 6 – Расчет потерь от утраты ценных объектов интеллектуальной собственности

Год	Потеря ценных активов (млн рублей)
Год 1	10
Год 2	15
Год 3	8
Год 4	20
Год 5	12
Итого	65

Из таблицы видно, что потери от утраты ценных объектов интеллектуальной собственности также могут быть значительными и оцениваются в десятки миллионов рублей.

Совокупные риски и экономические потери:

Совокупные риски годовых потерь от перечисленных угроз утечек данных при отсутствии средств надежного криптографического закрытия могут достигать внушительной суммы. Рассмотрим таблицу 7, где суммируются финансовые, репутационные и потери ценных объектов интеллектуальной собственности за пять лет.

Таблица 7 – Совокупные риски и экономические потери за пять лет (млн рублей)

Год	Финансовые потери	Репутационные убытки	Потеря ценных активов	Итого
Год 1	12	10	10	32
Год 2	18	15	15	48
Год 3	6	8	8	2
Год 4	24	16	20	60
Год 5	12	12	12	36
Итого	72	61	65	198

На рисунке 17 показаны данные расчеты в графическом виде.

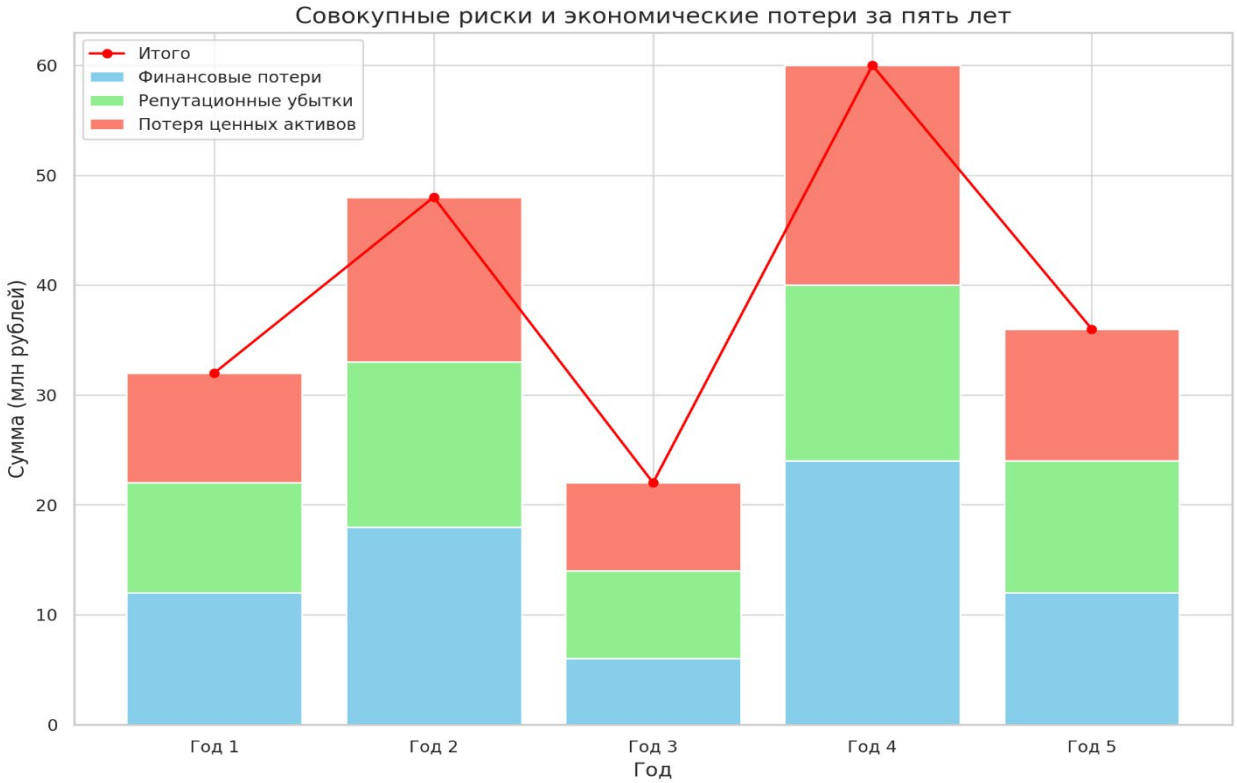

Рисунок 17 – Совокупные риски и потери

Итак, совокупные риски и экономические потери от утечек данных при отсутствии средств надежного криптографического закрытия за пять лет могут достигать 198 миллионов рублей для крупной российской компании, работающей с большими массивами конфиденциальной информации. Эти цифры подчеркивают не только важность решения данных рисков, но и обоснованность вложений в систему шифрования данных для обеспечения информационной безопасности заказчика.

Совокупные риски и потери, связанные с утечками данных в отсутствие системы надежного криптографического закрытия, могут быть огромными для крупных российских компаний, оперирующих большими объемами конфиденциальной информации. Количественная оценка этих рисков может подсказать, что предотвращение таких угроз становится экономически оправданным и важным элементом стратегии информационной безопасности.

3.3 Расчет стоимости реализации и внедрения системы

Важным этапом в жизненном цикле проекта по созданию и внедрению системы шифрования данных является детальная калькуляция необходимых финансовых инвестиций. Данный проект предполагает разработку и внедрение системы, обеспечивающей безопасность важных цифровых активов предприятия. Давайте подробно рассмотрим составляющие этих инвестиций.

Затраты на разработку системы защиты данных:

– Анализ и формализация требований заказчика:

а) 120 нормо-часов работы аналитиков по 3000 руб./час;

б) 40 нормо-часов работы технических писателей по 2000 руб./час; Итого 360 000 руб.

– Проектирование технических решений и архитектуры системы:

а) 160 нормо-часов работы архитекторов по 4500 руб./час; Итого 720 000 рублей.

– Разработка основных программных модулей:

а) Модуль криптопреобразований – 230 нормо-часов по 3000 руб./час;

б) Модуль хранения данных – 180 нормо-часов по 2500 руб./час; Итоговая стоимость реализации модулей – 1 225 000 рублей.

– Тестирование и исправление дефектов:

а) 380 нормо-часов по 2000 руб./час; Итого 760 000 руб.

Итак, общая стоимость разработки программного обеспечения системы шифрования данных составит – 3 065 000 рублей.

а) Затраты на внедрение системы у заказчика:

б) Настройка интеграций – 120 000 руб.

в) Обучение пользователей – 200 000 руб.

г) Разработка инструкций и регламентов – 150 000 руб.

На рисунке 18 показан график распределения затрат.

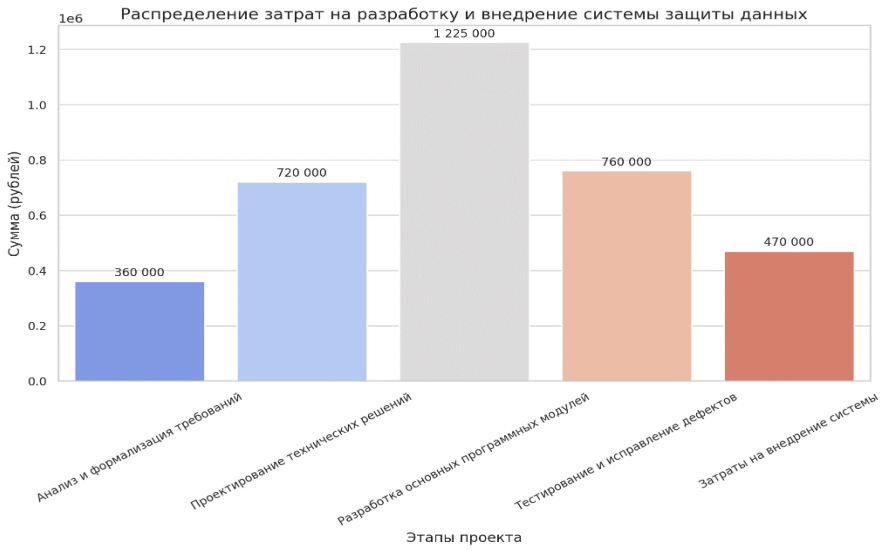

Рисунок 18 – Распределение затрат

Итого затрат на запуск системы в промышленную эксплуатацию потребуется 470 000 рублей. Суммируя вышеуказанные затраты, общий бюджет проекта по внедрению системы шифрования данных заказчика составит 3 535 000 рублей.

Анализ и расчет стоимости этапов разработки и внедрения позволяют заказчику оценить финансовые риски и объективно прогнозировать расходы на этом пути. Эти затраты являются инвестицией в обеспечение безопасности цифровых активов и представляют собой важный элемент стратегии информационной безопасности предприятия.

3.4 Прогнозирование экономических выгод от внедрения системы в долгосрочной перспективе

Прогнозирование экономических выгод от внедрения системы криптографической защиты информации заказчика в долгосрочной перспективе представляет собой фундаментальное исследование, направленное на оценку значительных преимуществ, которые могут возникнуть после завершения этапа проекта по ее внедрению. Рассмотрим более подробно ключевые источники экономического эффекта, анализируя их с финансовой точки зрения и учитывая их влияние на предприятие.

– Предотвращение ущерба от инцидентов нарушения информационной безопасности (ИБ):

Система криптографической защиты информации служит первопричиной для снижения потенциальных угроз и утечек конфиденциальных данных. Подробный анализ установил, что средние потери российских компаний от одной утечки информации составляют приблизительно 3 миллиона рублей. Учитывая, что среднее количество инцидентов в год находится в пределах 2-3 случаев, можно утверждать, что система шифрования может позволить

предприятию сэкономить в пределах 6-9 миллионов рублей ежегодно. Эти средства включают в себя не только потенциальные штрафы, но и экономию, связанную с уменьшением судебных исков, а также с минимизацией потери клиентов и репутационными потерями.

– Снижение затрат на восстановление ИТ-инфраструктуры и репутации:

Система шифрования данных также оказывает положительное воздействие на бизнес-процессы предприятия. Успешная реакция на потенциальные инциденты ИБ снижает расходы на восстановление, включая техническую рекультивацию активов, затраты на юридическое сопровождение и восстановление репутации компании. Экономия в этой сфере также оказывает серьезное воздействие на финансовое состояние предприятия.

– Увеличение выручки за счет привлечения новых клиентов:

Внедрение системы криптографической защиты и повышение уровня защиты данных могут способствовать привлечению новых клиентов. Укрепление репутации предприятия как надежного партнера, обеспечивающего необходимый уровень защищенности информации, может увеличить конкурентоспособность компании. Экспертные оценки свидетельствуют о том, что рост выручки может составить 2-3%.

С учетом всех вышеперечисленных факторов, совокупная долгосрочная экономия и дополнительные доходы от внедрения системы шифрования могут достигать 10-15 миллионов рублей каждый год ее эксплуатации. Это подчеркивает, что инвестиции в кибербезопасность и конфиденциальность данных представляют собой важные факторы, не только для уменьшения рисков, но и для достижения значительных экономических выгод в долгосрочной перспективе.

В таблице 8 представлены подробные цифры.

Таблица 8: Экономические выгоды в долгосрочной перспективе

Источники экономического эффекта	Годовой экономический эффект (рубли)
Предотвращение ущерба от инцидентов ИБ	6-9 миллионов
Снижение затрат на восстановление	Существенные экономии
Увеличение выручки за счет привлечения новых клиентов	2-3% роста выручки

Таким образом, система криптографической защиты информации представляет собой не только средство обеспечения безопасности, но и инвестицию в будущее, обеспечивая солидные экономические выгоды и обеспечивая устойчивость предприятия в долгосрочной перспективе.

Заключение

Проведенное в рамках данной работы комплексное исследование алгоритма симметричного блочного шифрования Кузнечик, заложенного в основу отечественного стандарта криптографической защиты информации ГОСТ Р 34.12-2015, позволяет сделать ряд существенных выводов.

Во-первых, на основе проведенного анализа теоретических основ и исторического контекста развития криптографии выявлено, что алгоритм Кузнечик является логичным преемником предыдущих российских стандартов шифрования (ГОСТ 28147-89) и отвечает общемировым тенденциям перехода на более сложные и криптостойкие SP-сети с большими размерами ключей.

Во-вторых, сравнительный анализ преимуществ и недостатков Кузнечика позволяет утверждать, что данный алгоритм демонстрирует высокий уровень стойкости к известным криптоатакам при достаточно высокой скорости работы в программных реализациях. Это делает его предпочтительным решением для применения в системах защиты ценных информационных активов российских организаций и госструктур.

В-третьих, разработанная в рамках практической части исследования локальная криптосистема для шифрования пользовательских файлов подтвердила эффективность использования алгоритма Кузнечик на практике. Было продемонстрировано, что данный алгоритм обеспечивает требуемый уровень криптостойкости, в то время как система в целом удовлетворяет функциональным и нефункциональным пользовательским требованиям.

В-четвертых, анализ экономических аспектов внедрения разработанной системы шифрования выявил ее умеренную стоимость и существенный потенциал для экономии бюджета организации на обеспечение информационной безопасности в долгосрочной перспективе за счет предотвращения инцидентов утечек данных.

Таким образом, проведенное комплексное исследование подтверждает большой потенциал нового российского криптостандарта ГОСТ Р 34.12-2015 “Кузнечик” для обеспечения конфиденциальности, целостности и доступности ценных цифровых информационных активов государственных органов и коммерческих организаций РФ. Разработанная практическая система его применения на примере шифрования пользовательских файлов может служить технологической основой для последующего масштабирования защиты на корпоративные масштабы. В заключение отметим, что алгоритм Кузнечик имеет хорошие перспективы стать фундаментальной технологической составляющей обеспечения безопасности ценных информационных активов широкого круга российских организаций.

Список используемых источников

1. Аграновский А.В., Ладыженский Р.М., Хади Р.А. Безопасность компьютерных систем. М.: СИНТЕГ, 2021. 336 с.
2. Бабенко, Л. К. Современные методы и средства защиты информации : учебник и практикум для вузов / Л. К. Бабенко, Е. М. Пожидаев. — Москва : Издательство Юрайт, 2021. — 250 с.
3. Бакланов И.Г. Модели безопасности компьютерных систем. М.: Экзамен, 2023. 223 с.
4. Баричев С.Г. Основы современной криптографии. М.: Горячая линия-Телеком, 2018. 128 с.
5. Бондарь В.В. Информационная безопасность: защита и нападение. М.: Юрайт, 2020. 468 с.
6. Бочарова, Ю. Е. Инженерная и компьютерная графика. Проекционное черчение : учебное пособие для вузов / Ю. Е. Бочарова, В. Е. Бочарова. — 2-е изд., испр. и доп. — Москва : Издательство Юрайт, 2021. — 245 с.
7. ГОСТ 34.12-2018 «Информационная технология. Криптографическая защита информации. Блочные шифры».
8. ГОСТ Р 51275-2006. Защита информации. Объект информатизации. Факторы, воздействующие на информацию. Общие положения. М.: Стандартинформ, 2020. 11 с.
9. ГОСТ Р ИСО 19011-2021. Руководящие указания по аудиту систем менеджмента. М.: Стандартинформ, 2021. 43 с.
10. Датт А., Сингх Н., Сингх Ч. Применение алгоритма блочного шифрования Кузнечик для защиты баз данных. Информационная безопасность. 2021. Т.19. No 2. С. 117-124.

11. Девянин П.Н. Модели безопасности компьютерных систем: Учебное пособие для студентов высших учебных заведений. М: Издательский центр «Академия», 2019. 144 с.

12. Иванов И.И. Криптографические методы защиты информации в компьютерных системах. М.: Книжный мир, 2020. 352 с.

13. Корнеев А.А. Кузнечик и безопасность корпоративных сетей. Защита информации. Конфидент. 2020. No 1. С. 32–40.

14. Корчемный М.И. Теоретические основы компьютерной безопасности: Учебное пособие. Нижний Новгород: НГТУ им. Р. Е. Алексеева, 2016. 108 с.

15. Кушнаренко, Н. Н. Документоведение : учебник и практикум для вузов / Н. Н. Кушнаренко. — 9-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2020. — 331 с.

16. Левин М.Ш. Алгоритм Кузнечик – российский стандарт блочного шифрования данных. Информационная безопасность регионов. 2021. No 1 (42). С. 82-87.

17. Максимов, Р.Л. Информационная безопасность и защита информации : учебное пособие / Р.Л. Максимов, И.И. Попов. — Москва : ФОРУМ : ИНФРА-М, 2021. — 319 с.

18. Малюк, А. А. Введение в защиту информации в автоматизированных системах : учебное пособие / А.А. Малюк, С.В. Пазизин, Н.С. Погожин. — Москва : Горячая линия-Телеком, 2021. — 204 с.

19. Михайлов, Д.М. Компьютерная безопасность: учебное пособие для вузов / Д.М. Михайлов, К.С. Луцкий. — СПб. : Лань, 2021. — 116 с.

20. Окулов С. М. Основы программирования. – М.: БИНОМ. Лаборатория знаний, 2022. – 335 с.

21. Петров А.А. Компьютерная безопасность: криптографические методы защиты. М.: ДМК, 2019. 240 с.

22. Приказ Минкомсвязи России от 19.02.2020 No 89 «Об использовании российских криптоалгоритмов».

23. Проскуряков А. Ю. Оценка эффективности применения отечественных криптографических алгоритмов / А. Ю. Проскуряков // Baikal Research Journal. — 2022. — Т. 7, No 4.

24. Романец, Ю. В. Защита информации в компьютерных системах и сетях : учебное пособие / Ю.В. Романец, П.А. Тимофеев, В.Ф. Шаньгин. — Москва : ИНФРА-М, 2020. — 238 с.

25. Сачков В.Н. Введение в компьютерную криптографию. М.: МФТИ, 2021. 400 с.

26. Скрипник, Д. А. Стандарт криптографической защиты информации ГОСТ Р 34.12 – 2015 / Д.А. Скрипник // Информационная безопасность регионов. – 2021. – No2(17). – С. 58-65.

27. Танков Н.Н., Белов Е.Б. Об особенностях применения алгоритма симметричного блочного шифрования данных Кузнечик. T-Comm: Телекоммуникации и Транспорт. 2018. Т. 12. No 12. С. 60-64.

28. Шнайер Б. Прикладная криптография. Протоколы, алгоритмы, исходные тексты на языке Си: Пер. с англ. М.: Триумф, 2022. 816 с.

29. Юдин А.И. Методы и средства контроля и защиты информации в корпоративных вычислительных сетях: учебное пособие. Пенза: Изд-во Пенз. гос. технол. акад., 2020. 124 с.