Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
АМУРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
(ФГБОУ ВО «АмГУ»)

Факультет математики и информатики
Кафедра информационных и управляющих систем
Направление подготовки 09.03.02 – Информационные системы и технологии
Направленность (профиль) образовательной программы Безопасность информационных систем

ДОПУСТИТЬ К ЗАЩИТЕ
Зав. кафедрой
_____А.В. Бушманов
«_____» _____2022 г.

БАКАЛАВРСКАЯ РАБОТА

на тему: Разработка программной модели скрытой передачи данных для защиты информации средствами C#

Исполнитель
студент группы 855-об _____ Д.В. Климентьев
(подпись, дата)

Руководитель
доцент _____ И.М. Акилова
(подпись, дата)

Консультант по безопасности
и экологичности _____ А.Б. Булгаков
доцент, канд. техн. наук (подпись, дата)

Нормоконтроль _____ В. Н. Адаменко
инженер (подпись, дата)

Благовещенск 2022

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
АМУРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
(ФГБОУ ВО «АмГУ»)

Факультет Математики и информатики

Кафедра Информационных и управляющих систем

УТВЕРЖДАЮ
Зав. кафедрой
_____А.В. Бушманов
«_____» _____2022 г.

ЗАДАНИЕ

К выпускной квалификационной работе студента Климентьев Д.В.

1. Тема выпускной квалификационной работы: Разработка программной модели скрытой передачи данных для защиты информации средствами C#

(утверждена приказом 679-уч от 05.04.2022)

2. Срок сдачи студентом законченной работы 17.06.2022

3. Содержание выпускной квалификационной работы: анализ предметной области; проектирование программного обеспечения; разработка программного продукта; описание программного продукта; руководство пользования; безопасность и экологичность.

4. Консультанты по выпускной квалификационной работе (с указанием относящихся к ним разделов): А.Б. Булгаков, доцент, канд. техн. наук, раздел 5 «Безопасность и экологичность»

5. Дата выдачи задания: 07.02.2022

Руководитель выпускной квалификационной работы: Акилова И.М., доцент

Задание принял к исполнению (07.02.2022) _____

(подпись студента)

РЕФЕРАТ

Дипломная (бакалаварская) работа содержит 108 с., 52 рисунка, 7 таблиц, 37 источников и 4 приложения.

КРИПТОГРАФИЯ, СТЕГАНОГРАФИЯ, LSB-СТЕГАНОГРАФИЯ, ДЕСКТОПНОЕ ПРИЛОЖЕНИЕ, АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ, ПРОЕКТИРОВАНИЕ, РАЗРАБОТКА, БЕЗОПАСНОСТЬ И ЭКОЛОГИЧНОСТЬ.

В работе исследуется возможности совмещения методов криптографии и стеганографии для создания более стойкой защиты информации от угрозы, связанной с получением той или иной информации, даже если злоумышленник смог перехватить файл, содержащий информацию.

Целью исследования: изучение алгоритмов LSB-стеганографии и криптографии и их объединение.

Выполнение проекта включает ряд этапов:

- Анализ предметной области;
- Проектирование ПО. Анализ требований к продукту, выделение логической части ПО, описание взаимодействия функциональных подсистем;
- Разработка ПО.

Результатом является программная модель для скрытой передачи дынных.

СОКРАЩЕНИЯ, ОБОЗНАЧЕНИЯ И ОПРЕДЕЛЕНИЯ

ПК – персональный компьютер;

РФ – Российская Федерация;

ОС – операционная система;

ПО – программное обеспечение;

ТЗ – техническое задание;

ЭЦП – электронной цифровой подписи

ЭВМ – электронно-вычислительная машина

ПЭВМ – персональная электронно-вычислительная машина

НОРМАТИВНЫЕ ССЫЛКИ

В настоящей бакалаврской работе использованы ссылки на следующие стандарты и нормативные документы:

ГОСТ 2.104-68 ЕСКД Основные надписи

ГОСТ 2.105-95 ЕСКД Общие требования к текстовым документам

ГОСТ 2.106-96 ЕСКД Текстовые документы

ГОСТ 2.111-68 ЕСКД Нормоконтроль

ГОСТ 2.113-75 ЕСКД Групповые конструкторские документы

ГОСТ 3.1103-83 ЕСКД Основные надписи

ГОСТ 3.1130-93 ЕСКД Основные требования к формам и бланкам документов

ГОСТ 3.1105-84 ЕСКД Правила оформления документов общего назначения

ГОСТ 7.80-2000 Библиографическая запись. Заголовок.

ГОСТ 7.82–2001 Межгосударственный стандарт. Система стандартов по информации, библиотечному и издательскому делу. Библиографическая запись. Библиографическое описание электронных ресурсов. Общие требования и правила составления

ГОСТ Р 1.5-2002 Государственная система стандартизации Российской Федерации. Стандарты. Общие требования к построению, изложению, оформлению, содержанию и обозначению

ГОСТ 7.1-2003 Система стандартов по информации, библиотечному и издательскому делу. Библиографическая запись. Библиографическое описание. Общие требования и правила составления

ГОСТ 7.11–2004 Межгосударственный стандарт. Система стандартов по информации, библиотечному и издательскому делу. Библиографическая запись. Сокращение слов и словосочетаний на иностранных европейских языках.

ГОСТ Р 1.5-2004 Стандартизация в РФ. Стандарты национальные РФ. Правила построения, изложения, оформления и обозначения

ГОСТ 7.05-2008 Система стандартов по библиотечному и издательскому делу. Библиографическая ссылка. Общие требования и правила составления

ГОСТ Р 7.0.12–2011 Система стандартов по информации, библиотечному и издательскому делу. Библиографическая запись. Сокращение слов на русском языке. Общие требования и правила

СТО СМК 4.2.3.15-2016 Стандарт организации. Требования к структуре и оформлению локальных нормативных документов университета.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ

Задача защиты информации от несанкционированного доступа решалась во все времена на протяжении истории человечества. Уже в древнем мире выделилось два основных направления решения этой задачи, существующие и по сегодняшний день: криптография и стеганография. Целью криптографии является скрытие содержимого сообщений за счет их шифрования. Стеганографии – скрывается сам факт существования тайного сообщения.

В работе исследуется возможности совмещения методов криптографии и стеганографии для создания более стойкой защиты информации от угрозы, связанной с получением той или иной информации, даже если злоумышленник смог перехватить файл, содержащий информацию. Разработана программная модель шифрования и дешифрования информации. Для устойчивости информация шифруется криптографически, а после вноситься в исходное изображение. Обратная дешифровка происходит в обратном порядке. Рассмотрены характеры угроз и возможные ограничения модели.

1 АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ

1.1 Шифрование

Шифрование – это преобразование данных (открытого текста) в иную форму (как правило, бессмысленный набор символов) с целью сокрытия их от третьих лиц, не имеющих ключа для расшифровывания. [1]

Шифрование в нашей жизни применяется повсеместно. Рассмотрим самые яркие примеры использования технологии шифрования:

– все сайты, в адресной строке которых стоит https. Именно символ «s» и свидетельствует, что данные зашифрованы. Символ «s» расшифровывается как «Secure Sockets Layer» или «уровень защищенных сокетов», цель которого – добавить в протокол стандарт шифрования. Ключ, при этом, от зашифрованных данных имеется на веб-сервере сайта;
– все системы сдачи отчетности. Данные шифруются, чтобы конфиденциальность данных не была нарушена;
– все чаты, такие как WhatsApp, Viber, Telegram шифруют передаваемые сообщения для идентификации личности, сохранения целостности информации и т. д.;
– письма, отправляемые злоумышленниками для шифрования Ваших данных с целью получения денег за расшифровку.

Все вышеперечисленное наглядно демонстрирует, что ежедневно каждый из нас сталкивается с шифрованием данных, и при этом не имеет значения сколько вам лет или где вы работаете.

1.1.1 Криптографическое шифрование

На данный момент, при употреблении термина «шифрования», у людей могут складываться разное о представление. Некоторые могут вспомнить различные шифры, которые придумывали в школьные годы, другие представляют себе разные фильмы, в которых использовались различные системы шифрования. Настоящая криптография ушла далеко вперед и в

информационную эпоху уже представляет из себя систему секретности, которая в повседневное время окружает нас и должна обеспечивать принципы конфиденциальности, целостности и доступности информации, чтобы можно было надежно защитить от дешифровки важной информации, как крупных организаций, так и персональных данных обычного пользователя. Настоящая криптография ранее использовалась только в военных целях. Однако, так же, как и многие другие вещи, с её развитием она стала повседневной частью быта любого пользователя и основополагающей для защиты важной информации организаций.

Широкое распространение криптографии является одним из немногих способов защитить человека от ситуации, когда он вдруг обнаруживает, что кто-то может контролировать каждый его шаг.

Криптография – это наука о методах обеспечения конфиденциальности, целостности и доступности информации.

Изначально криптография занималась только методами шифрования информации, то есть обратимого преобразования открытого (исходного) текста, с помощью секретного ключа или алгоритма, который называется шифртектом. В настоящее время, традиционная криптография представляет из себя системы симметричных и асимметричных криптосистем, с использованием одного и того же секретного ключа или разных, соответственно. Так же криптография включает в себя различные алгоритмы электронной цифровой подписи (ЭЦП), хеш-функций, управление ключами, получение скрытой информации, квантовую криптографию.

1.1.1.1 История криптографии

Первый период криптографии характеризуется созданием моноалфавитных шифров. В основу вошла замена алфавита шифртекста на другие буквы алфавита или различные символы. Пример такого шифрования придумал известный древнеримский государственный и политический деятель Гай Юлий Цезарь, и его честь данный шифр был назван, шифр Цезаря, который

предполагает замену каждой буквы исходного текста на букву стоящей на расстоянии равной трем буквам в алфавите. Шаг в 3 буквы называется «Ход конем», но в дальнейшем данный способ, для придания большей конфиденциальности, видоизменили и, теперь, можно выбирать любой шаг для смещения букв.

В начало второго периода вошел полиалфавитный шифр. Самый распространенный метод является шифр Виженера. Данный шифр использует дополнительное (ключевое) слово для перестановки букв согласно таблицы. Для начала расписывается исходный текст, а после, под каждой буквой, расписывается буквы ключевого слова так, чтобы под каждой буквой исходного слова была буква ключевого. После каждая буква исходного текста заменяется на букву в соответствующей строке таблицы и заменяется на букву соответствующей столбцу буквы ключевого слова.

Третий период привело в начало появление электронных и механический устройств. В этого время начали создавать различные механизмы шифровальщиков, с помощью которых шифровалась информация. Это позволило ускорить время шифрования и уменьшить риск ошибок за счет человеческого фактора. Но в данный период еще продолжались использоваться полиалфавитные шифры.

Четвертый период ознаменовался с переходом к математической криптографии. С начала середины XX века люди научились ставить строгие определения информации и количества информации. И, вплоть до конца данного периода, криптография еще считалась классической и использовалась только в военных целях и для шифрования особо секретной информации. В эти периоды криптография представляла из себя шифртекст с секретным ключом.

В современный период появилось новое направление – криптография с отрытым ключом. Основополагающей для развития данного появление стало развитие информационных систем, а также распространением криптографии частными лицами. В настоящее время существует множество криптосистем в

разных странах, из-за их обилия появились различные запрети и стандарты на использования таких систем.

1.1.1.2 Запреты

В РФ коммерческая деятельность, связанная с использованием криптографических средств, подлежит обязательному лицензированию. Постановление Правительства РФ от 16.04.2012 N 313 (ред. от 28.12.2021) "Об утверждении Положения о лицензировании деятельности по разработке, производству, распространению шифровальных (криптографических) средств, информационных систем и телекоммуникационных систем, защищенных с использованием шифровальных (криптографических) средств, выполнению работ, оказанию услуг в области шифрования информации, техническому обслуживанию шифровальных (криптографических) средств, информационных систем и телекоммуникационных систем, защищенных с использованием шифровальных (криптографических) средств (за исключением случая, если техническое обслуживание шифровальных (криптографических) средств, информационных систем и телекоммуникационных систем, защищенных с использованием шифровальных (криптографических) средств, осуществляется для обеспечения собственных нужд юридического лица или индивидуального предпринимателя)" [2], из которого можно выделить следующие положения о лицензировании деятельности:

- Распространению шифровальных (криптографических) средств;
- Техническому обслуживанию шифровальных (криптографических) средств;
- Предоставлению услуг в области шифрования информации;
- Разработке, производству шифровальных (криптографических) средств, защищенных с использованием шифровальных (криптографических) средств информационных и телекоммуникационных систем.

Так же данное постановление содержит очень жесткие требования к лицу-соискателю лицензии. Она может включать, как личную информацию, так и требования к безопасности при работе с криптосистемами.

В настоящее время действует также Приказ ФСБ России от 9 февраля 2005 г. N 66 «Об утверждении положения о разработке, производстве, реализации и эксплуатации шифровальных (криптографических) средств защиты информации (положение пкз-2005)» [3], в котором в полном объеме расписаны все этапы разработки и эксплуатации криптосистем.

Так же был издан Указ Президента РФ от 3 апреля 1995 N 334 «О мерах по соблюдению законности в области разработки, производства, реализации и эксплуатации шифровальных средств, а также предоставления услуг в области шифрования информации»[4], постановивший «Запретить использование государственными организациями и предприятиями в информационно-телекоммуникационных системах шифровальных средств, включая криптографические средства обеспечения подлинности информации (электронная подпись), и защищенных технических средств хранения, обработки и передачи информации, не имеющих сертификата Федерального агентства правительственной связи и информации при Президенте Российской Федерации, а также размещение государственных заказов на предприятиях, в организациях, использующих указанные технические и шифровальные средства, не имеющие сертификата Федерального агентства правительственной связи и информации при Президенте Российской Федерации».

1.1.1.3 Криптографический протокол

Криптографический протокол - это абстрактный или конкретный протокол, включающий набор криптографических алгоритмов. В основе протокола лежит набор правил, регламентирующих использование криптографических преобразований и алгоритмов в информационных процессах. [5]

К функциям криптографического протокола можно отнести:

- Формирование ключей;
- Обмен ключами;
- Аутентификация сторон;
- Доказательство целостности и происхождения данных (ЭЦП);
- Разделение ключей;
- Безопасные распределённые вычисления;
- Обеспечение конфиденциальности данных;
- Обеспечение невозможности отказа;
- Обеспечение целостности данных;
- Обеспечение целостности соединения;
- Разграничение доступа.

Классификация

Протоколы шифрования и расшифрования. В основе протокола содержится различные алгоритмы симметричного или асимметричного шифрования и расшифрования. Сами алгоритмы основываются на передачи сообщения, по открытому каналу связи, отправителю некоторого шифртекста, который был преобразован согласно стандарту шифрования и использованием открытого ключа для симметричного шифрования, а также открытого и закрытого ключа для ассиметричного шифрования. Алгоритм расшифрования преобразует, на основе ключа шифрования, шифртекст обратно в открытую форму.

Протоколы ЭЦП. Данный протокол основывается на вычислении электронно-цифровой подписи в информации и его проверки согласно стандартам алгоритмов Таким образом данный протокол проверяет целостность и доступность данных, но не шифрует сам открытый текст. Если для принятого сообщения необходимо обеспечить безотказность системы, то дополнительно, поле обнародования ЭЦП, она может быть уничтожена сразу или через определенный срок.

Протоколы идентификации и аутентификации. Главной функцией данного протокола, является точная идентификация объекта и проверка на действительность, то есть на то, что данный объект является тем, за кого себя выдает. Обычно такая информация сохраняется в защищенной базе, в которой не допускается сохранения данных о аутентификации, а хранится, как минимум, в формате хеш-функций, для дополнительной защиты.

При необходимости, если при использовании протокола присутствует ЭЦП, то именно она будет является секретным ключом и при проверке протокола проверяется с помощью ключа ЭЦП.

1.1.1.4 Требования к безопасности

а) Аутентификация (нешироковещательная):
   1) аутентификация субъекта;
   2) аутентификация сообщения;
   3) защита от повтора.

б) Аутентификация при рассылке по многим адресам или при подключении к службе подписки/уведомления:
   1) неявная (скрытая) аутентификация получателя;
   2) аутентификация источника.

в) Авторизация (доверенной третьей стороной);

г) Свойства совместной генерации ключа:
   1) аутентификация ключа;
   2) подтверждение правильности ключа;
   3) защищенность от чтения назад;
   4) формирование новых ключей;
   5) защищенная возможность договориться о параметрах безопасности.

д) Конфиденциальность;

е) Анонимность:
   1) защита идентификаторов от прослушивания (несвязываемость);

2) защита идентификаторов от других участников.

ж) Ограниченная защищенность от атак типа «отказ в обслуживании»;

и) Инвариантность отправителя;

к) Невозможность отказа от ранее совершенных действий:

1) подотчетность;

2) доказательство источника;

3) доказательство получателя.

л) Безопасное временное свойство

1.1.1.5 Современная криптография

С развитием криптографии появилось множество алгоритмов криптографических преобразований. В современной криптографии принято использовать криптографические протоколы с использованием вычислительных систем. Из-за стремительного развития этого направления, различные алгоритмы заменяются другими, более надежными и проверенными. Это связано с высокой активностью злоумышленников, которые различными способами пробуют обнародовать информации в личных целясь. Именно из-за этой угрозы существуют различные правила и политики для установки различных криптографических систем, начиная от цикла смены пароля в определенные сроки, закачивая полной замены системы, если это необходимо.

На данный момент самыми распространенными алгоритмами криптографических преобразований является:

– симметричные;

– асимметричные;

– хэш-функций.

1.1.1.6 Криптоанализ

Криптоанализ - наука о методах получения исходного значения зашифрованной информации, не имея доступа к секретной информации (ключу), необходимой для этого. [6] Термин был введён американским

криптографом Уильямом Ф. Фридманом в 1920 году, более подробно описал Девид Кан в 2000 году.

На данный момент существуют различные профессии, связанные с криптоанализом, но персонала, которые направлены на эту деятельность крайне мало, связи с её узко направленностью и сложностью.

Классический криптоанализ. Понятие криптоанализ относительно ново, но способы взлома существовали уже давно, еще до первых ЭВМ. Начало криптоанализ получил в писмености арабского ученого Ал-Кинди ещё в 9 веке. В нем рассказывалось об частотном анализе, который в нынешнее время имеет большую популярность. В этом научном труде содержится описание метода частотного анализа.

Частотный криптоанализ - основной инструмент для взлома большинства классических шифров перестановки или замены. В этом методе главным фактором является предположение о том, что символы, повторяющиеся по ходу шифртекста, имеет некое начало и конец блока шифрования данных Самым простым примером такого криптоанализа может послужить банальный подсчёт каждого повторяющегося символа и деления числа символов на количество всех символов в шифртексте. Для процентного соотношения данное число необходимо умножить на сто. Далее, используя ранее существующую таблицу вероятностей распределения букв, текст переводиться в отрытый.

1.1.2 Стеганографическое шифрование

Стеганография – это, в общем случае, искусство передачи скрытого сообщения. [7] Принято считать, что впервые этот термин использовал Иоганн Тритемий (1462–1516) в своей работе под тем же названием: «Стеганография».

1.1.2.1 История возникновения

Слово «стеганография» в переводе с греческого буквально означает «тайнопись» (от греч. στεγανός (steganos) – скрытый + γράφω (grafo) – пишу). Местом зарождения стеганографии многие называют Египет, хотя первыми

«стеганографическими сообщениями» можно назвать и наскальные рисунки древних людей. Первое упоминание о стеганографических методах в литературе приписывается Геродоту, который описал случай передачи сообщения Демартом. Он соскабливал воск с дощечек, писал письмо прямо на дереве, а потом заново покрывал дощечки воском.

Другой эпизод, который относят к тем же временам, – передача послания на голове раба. Для передачи тайного сообщения голову раба обривали, наносили на кожу татуировку и, когда волосы отрастали, отправляли с осланием.

В Китае письма писали на полосках щелка. Для сокрытия сообщений полоски с текстом письма сворачивались в шарики, покрывались воском и затем глотались посыльными.

Темное средневековье породило не только инквизицию: усиление слежки привело к развитию как криптографии, так и стеганографии. Именно в средние века впервые было применено совместное использование шифров и стеганографических методов. [8]

1.1.2.2 Классификация и принципы стеганографии

В стеганографии, в отличие от криптографии, скрывается сам факт передачи сообщения. Здесь принципиальным является помещение информации в какой-либо нейтральный, не вызывающий подозрений объект, называемый контейнером (чаще всего в компьютерной "тайнописи" им является текстовый, графический, аудио- или видеофайл) и незаметное распределение в нем. Своеобразным шифром автора такого сообщения выступает определение "гнезд", в которые вносится информация, порядок ее внесения, внешняя незаметность изменений контейнера, сохранение различных статистических характеристик контейнера и сам факт, что в этом безобидном файле может быть что-то скрыто. Использование тайнописи, не подкрепленное средствами криптографической защиты, вскоре сочли ненадежным, и с появлением все новых методов шифрования стеганография начала оставаться "в тени"

криптографии. До сих пор книг и публикаций, посвященных стеганографии, гораздо меньше, чем различных материалов по криптографии. [8]

Однако в современном мире, где огромную роль играет цифровое представление информации у стеганографии появилось много новых областей применения. Развитие вычислительной техники создало новые исследования и научных предложений в области компьютерной стеганографии. Одной из причин работы в направлении этих исследований заключается в том, что во многих странах существуют ограничения на использование различных средств криптографии. Другая причина - необходимость защиты права собственности на цифровую информацию. На данный момент компьютерная стеганография является полноценным направлением в области защиты информации.

Стеганографию можно разделить на 3 раздела:

- классическая стеганография – включает в себя все “некомпьютерные методы”;
- компьютерная стеганография – направление классической стеганографии, основанное на особенностях компьютерной платформы;
- цифровая стеганография – направление классической стеганографии, основанное на сокрытии или внедрении дополнительной информации в цифровые объекты, вызывая при этом некоторые искажения этих объектов. Используется избыточность аудио- и визуальной информации.[9]

Основные положения стеганографии:

- Методы скрытия должны обеспечивать аутентичность и целостность файла;
- Предполагается, что криптографу полностью известны возможные стеганографические методы;
- Безопасность методов основывается на сохранении стеганографическим преобразованием основных свойств открыто передаваемого файла при

внесении в него секретного сообщения и некоторой неизвестной противнику информации – ключа;

– Даже если факт скрытия сообщения стал известен противнику через сообщника, извлечение самого секретного сообщения представляет сложную вычислительную задачу. [10]

Компьютерная стеганография базируется на двух принципах. Первый заключается в том, что файлы, содержащие оцифрованное изображение или звук, могут быть до некоторой степени видоизменены без потери функциональности, в отличие от других типов данных, требующих абсолютной точности.

Второй принцип состоит в неспособности органов чувств человека различить незначительные изменения в цвете изображения или качестве звука, что особенно легко использовать применительно к объекту, несущему избыточную информацию, будь то 16-битный звук, 8-битное или, еще лучше, 24-битное изображение. Если речь идет об изображении, то изменение значений наименее важных битов, отвечающих за цвет пикселя, не приводит к сколь-нибудь заметному для человека изменению цвета.

Особенностью стеганографического подхода является то, что он не предусматривает прямого оглашения факта существования защищаемой информации. Это обстоятельство позволяет в рамках традиционно существующих информационных потоков или информационной среды решать некоторые важные задачи защиты информации ряда прикладных областей.

Основным определяющим моментом в стеганографии является *стеганографическое преобразование*. До недавнего времени стеганография, как наука, в основном изучала отдельные методы сокрытия информации и способы их технической реализации. Разнообразие принципов, заложенных в стеганографических методах, по существу, тормозило развитие стеганографии как отдельной научной дисциплины и не позволило ей сформироваться в виде некоторой науки со своими теоретическими положениями и единой

концептуальной системой, которая обеспечила бы формальное получение качественных и количественных оценок стеганометодов. В этом история развития стеганографии резко отличается от развития криптографии. [8]

Использование стеганографических систем является наиболее эффективным при решении проблемы защиты информации с ограниченным доступом. Это означает возможность скрытой передачи информации. На рисунке 1 приведена наиболее общая классификация методов.

Рисунок 1 – Общая классификация методов компьютерной стеганографии

1.1.2.3 Методы стеганографии

Под методами стеганографии подразумеваются различные способы достижения принципов стеганографии с помощью различных способов, таких как замена семантике информации, различные методы, связанные с лингвистическим подходом, использования не содержащих текстовые информацию файлов, например, изображение и звук, а также основывающиеся

на модификации текстового содержания, но которые не меняют смысловой подтекст информации.

Различают различные методы стеганографии: [8]

Line-shift coding (изменение расстояния между строками электронного текста). Также называемым методом изменения межстрочных интервалов. Его суть заключается в том, что используется текст с различными межстрочными расстояниями (Рисунок 2).

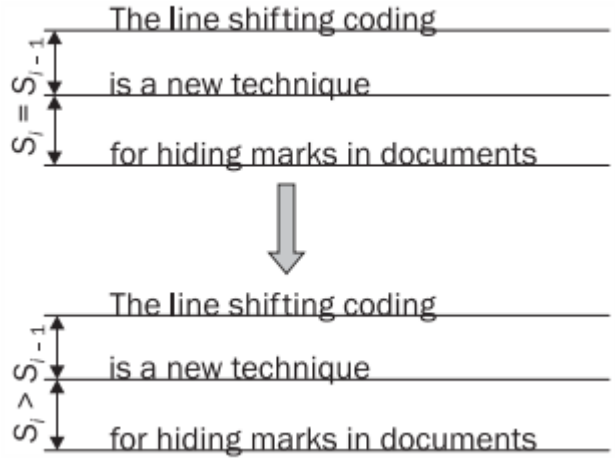

Рисунок 2 – Иллюстрация метода на основе изменения межстрочных интервалов

Разница в межстрочных расстояниях авторами изменялась на 1/300 дюйма (это расстояние было привязано к существовавшей в то время разрешающей способности монитора). Очевидным недостатком метода является его низкая эффективность: размер в битах осаждаемой информации не может превысить количество строк в контейнере.

Word-shift coding (изменение расстояния между словами в одной строке электронного текста). Сущность способа складывается в том, что оседание информации организовано для изменения расстояния промеж словами текста-контейнера. Подобно прошлому методу, отделяется наибольшее и наименьшее расстояния меж словами, значащие соответственно символ «1» и «0», и остальные расстояния, или некоторые из них, увеличивают или уменьшают до

размеров уже выделенных. Частный случай этого метода – метод изменения количества пробелов (Рисунок 3).

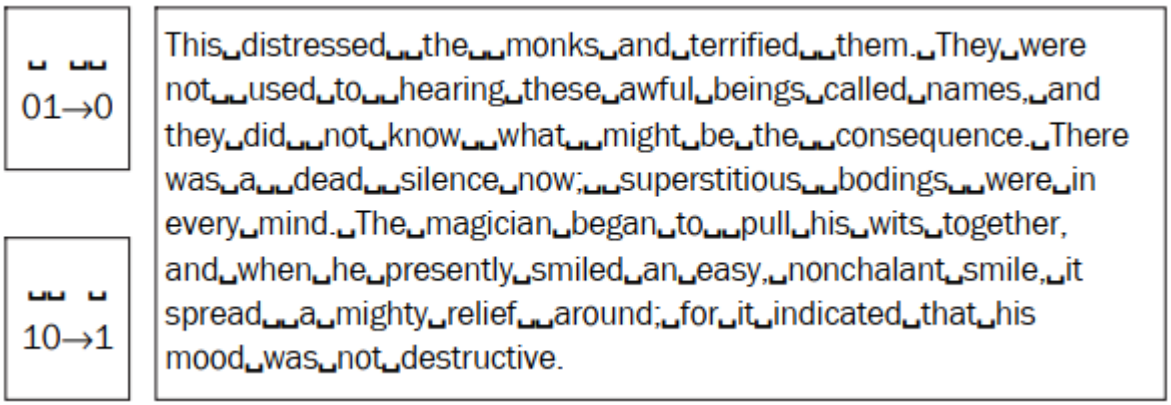

Рисунок 3 – Пример использования метода на основе длин пробелов

На данном рисунке видно, как в текст-контейнер внедряется бинарная последовательность. Как видно, переход с одинарного пробела на двойной кодирует «1», переход же с двойного пробела на одинарный кодирует «0».

Метод изменения интервала табуляции. Аналогичен вышеописанному методу изменения количества пробелов, только в этом случае меняется не количество пробелов, а соответственно расстояние между строками и интервал табуляции.

Null chipper (дословно – несуществующий, нулевой лепет). Данный метод предполагает размещение информации в необходимых местах или позициях, из-за которого текст теряет логический смысл.

Метод увеличения длины строки. Предусматривает искусственное увеличение длины каждой строки за счет пробелов: например, одному пробелу соответствует логический 0, двум – 1 (Рисунок 4).

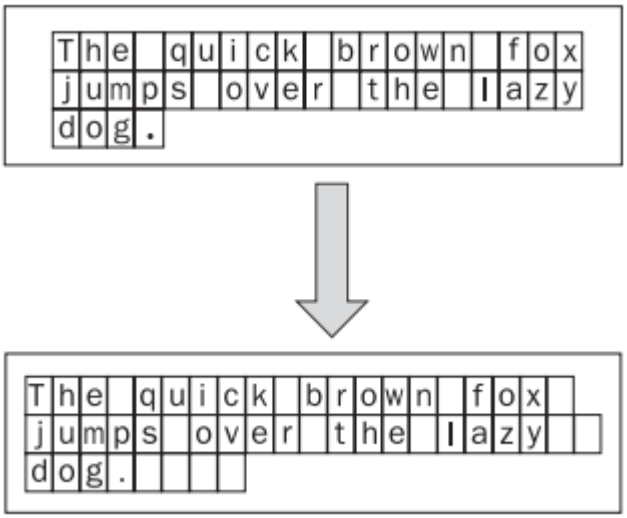

Рисунок 4 – Иллюстрация метода увеличения длины строки

Преимущество такого метода кодирования состоит в том, что оно может быть выполнено с любым текстом; изменения в формате резко не бросаются в глаза читателю, обеспечивается передача большего числа скрытых данных по сравнению с предыдущим методом (примерно 1 бит на 80 байт содержимого контейнера). Недостаток метода состоит в том, что некоторые компьютерные программы (например, Sendmail) могут неосторожно удалять дополнительные пробелы. Помимо этого, скрытые таким образом данные не всегда могут быть восстановлены с печатной копии документа.

Использование регистра букв. Для того, чтобы указать логическую единицу скрываемого сообщения, данным метод использует символ нижнего регистра, и верхнего, если нужно указать логический ноль скрываемого сообщения. Например, для использования двоичного представления кода символа «А» – «01000001» необходимо записать слово «кВартирА».

LSB-метод. Метод LSB же основывается на ограниченных способностях зрения или слуха человека, вследствие чего людям тяжело различать незначительные вариации цвета или звука. Рассмотрим это на примере 24-битного растрового RGB-изображения. Каждая точка кодируется 3 байтами, каждый байт определяет интенсивность красного (Red), зеленого (Green) и синего (Blue) цветов. Совокупность интенсивностей цвета в каждом из 3

каналов определяет оттенок пикселя. Представим пиксель тремя байтами в битовом виде, как это показано на рисунке 5.

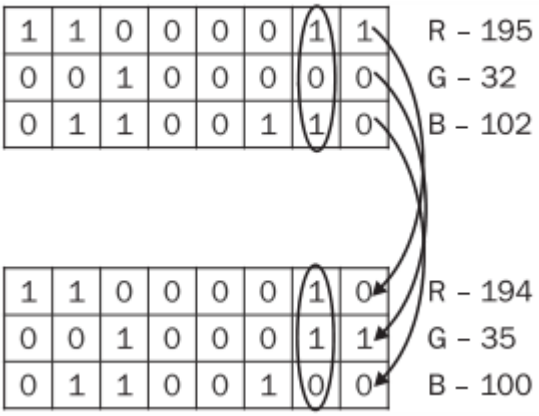

Рисунок 5 – изменения наименьшего бита в цвете

Младшие биты (справа) дают незначительный вклад в изображение по сравнению со старшими. Замена одного или двух младших бит для человеческого глаза будет почти незаметна.

Для извлечения этих данных прочитаем наименьший значащий бит как битовый поток и приведем к нужному виду. После этого такой поток переводится в необходимую кодировку и таким образом получается открытый текст.

1.1.2.4 Методы стенографического анализа и защита от них

Терминология стеганоанализа очень похожа на терминологию криптоанализа, но не полностью совпадает с ней, а имеет расхождения, важные именно для стеганографии. Если криптоанализ используется, в первую очередь, для дешифровки шифртекста, то стегоанализ для установления самого факта наличия текста и передаваемой информации. Основной целью стеганоанализа является создание и проектирования различных стеганосистем и, в дальнейшем, их исследование для выявления качественных и количественных оценок нахождения текста внутри системы или анализе устойчивости таких систем.

Обычно выделяют несколько этапов пассивных атак на стеганографической системы:

– нахождения факта самого присутствия информации;
– обнародование открытого текста из скрытой информации.

Возможны и активные атаки на стеганографической системы:

– нарушение целостности открытого текста;
– блокировка дальнейшие использования скрытой информация даже на принципах санкционированного доступа.

По аналогии с криптоанализом выделяют следующие общие виды атак на стеганосистемы: [7]

Атака на основании известного заполненного контейнера. Во время данной атаки используется уже заготовленные заполненные контейнеры. Задача нарушителя может заключаться в выявлении факта осаждения информации (наличия стеганоканала), а также в извлечении данных или определении ключа.

Атака на основании известного встроенного сообщения. Этот тип атаки характерен для систем защиты права интеллектуальной собственности, например, на текстовые документы или коды программ. Задачей анализа является получение ключа. Если подходящий тайному сообщению заполненный контейнер неизвестен, то такую задачу решить практически невозможно.

Адаптивная атака на основании выбранного сообщения. Эта атака является частным случаем предыдущей. При этом нарушитель имеет возможность выбирать сообщения для навязывания их передачи адаптивно, в зависимости от результатов анализа предшествующих контейнеров-результатов.

Атака на основании выбранного заполненного контейнера. Этот тип атаки более характерен для систем с использованием ЦВЗ. У аналитика есть некий анализатор заполненных контейнеров в виде «черного ящика» и несколько таких контейнеров. Анализируя выявленные скрытые сообщения, нарушитель пытается раскрыть ключ.

Атака на основании известного пустого контейнера. Если у нарушителя есть эталонный файл контейнера, то он может сравнить его с уже заполненным и тем самым сделать выводы если ли в данном контейнере информации или нет. Этот тип атаки является уникальным только для стеганографии, т.к. он ссылается на визуальное представление контейнера.

Атака на основании выбранного пустого контейнера. В этом случае злоумышленник способен принудить воспользоваться предложенным им контейнером. В этом случае мы также не найдем сходства с криптографией. Как и в последнем случае.

Визуальная атака. Суть его заключается в формировании новых изображений на основе исходного, состоящих из наименее значащих бит различных цветовых плоскостей. И если создать такое изображения можно увидеть артефакты свидетельствующие, что в изображении присутствует текст. Пример такой визуальной атаки показан на рисунке 6.

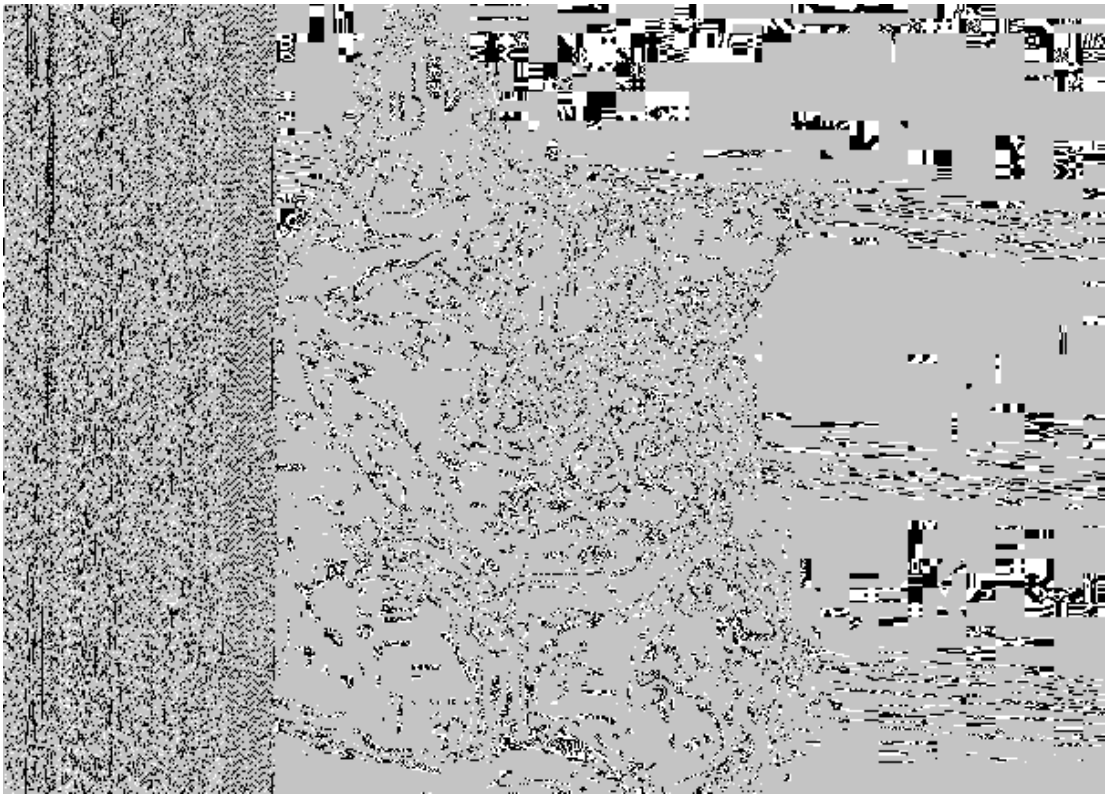

Рисунок 6 – Изображение состоящие из наименее значащих бит различных цветовых плоскостей

Для сопротивления визуальной атаки можно генерировать случайные числа и заполнять значение битов в координаты данный значений. Таким образом изображение все равно поменяется, но без эталона будет не сразу понятно, что в данном изображении может присутствовать текст. Пример показан на рисунке 7.

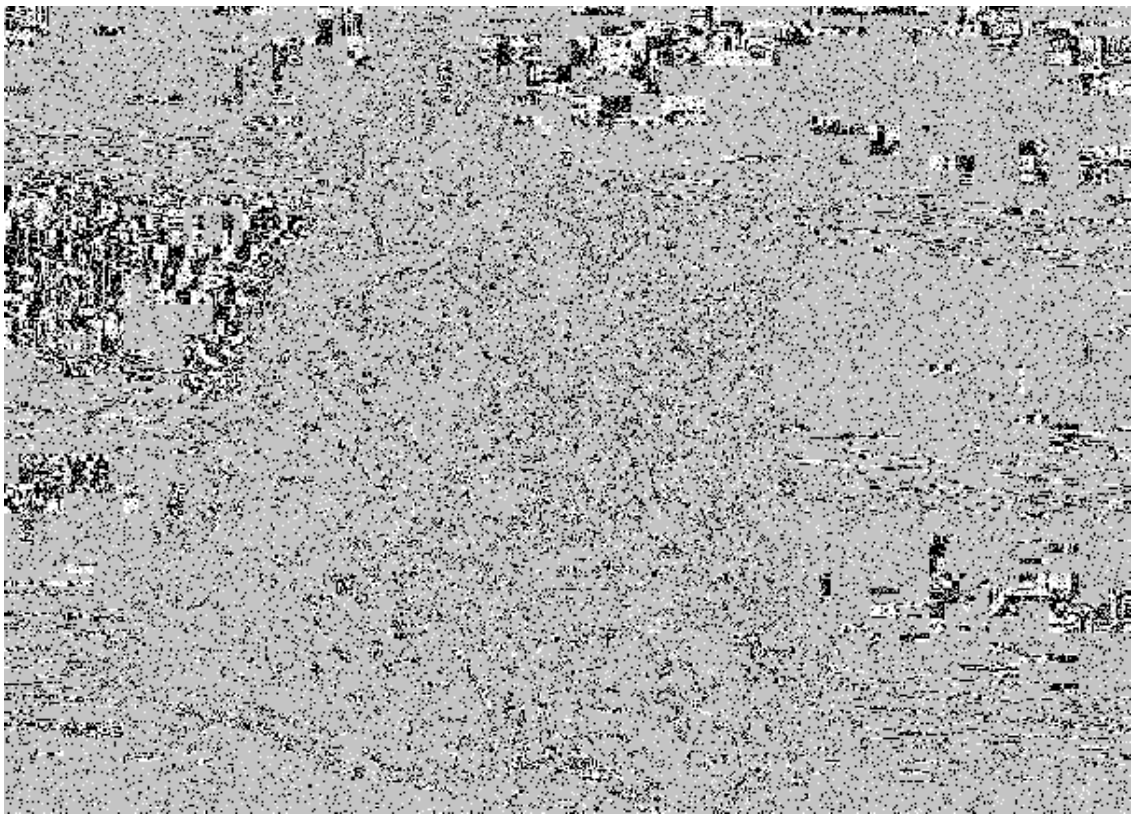

Рисунок 7 - Изображение состоящие из наименее значащих бит различных цветовых плоскостей с случайной генерацией вставляемых бит.

RS-метод. Метод статистического стеганоанализа под названием RS-метод [11-13], впервые опубликованный в 2001 г. коллективом ученых под руководством Дж. Фридрих. Сокращение в названии расшифровывается как Regular-Singular, то есть «регулярно-сингулярный».

Метод содержит несколько подготовительных этапов.

– Изображение разделяется на группы из n количества пикселей, данное число может быть произвольным и не имеет особой важности при анализе, за исключением слишком маленького числа до 5 значений;

– Затем необходимо определить функцию гладкости. Это необходимо, чтобы зафиксировать «регулярность» группы пикселей G. Чем шумнее группа пикселей G=(x_1, ... , x_n), тем большее значение функции гладкости;
– К каждой группе пикселей применяется одна из функций флиппинга и на основании значения функции-дескриминанта до и после флиппинга, определяется тип группы пикселей: обычный (Regular), единичный/необычный (Singular), и непригодный (unusable).

Таким образом если применять RS-метод для LSB-метода, то можно получить коэффициент оценки содержания текста.

Данный метод имеет большую погрешность, поэтому его использует только для определения текста, но не его точное количество.

1.2 Анализ языка программирования C#

C# – это объектно-ориентированный язык программирования. Об был создан инженерами Microsoft в 1999 году под руководством Андерса Хейлсберга и Скотта Вильтаумота. [14]

Язык входит в семью C-подобных языков. Синтаксис приближен к Java и C++. Его особенности:

– статистическая типизация;
– поддерживаются основные принципы объектно-ориентированного программирования (инкапсуляция, наследование, полиморфизм, абстракция);
– поддерживается перегрузка операторов;
– Поддержка библиотек .Net.

Основные преимущества языка. C# популярен за счет своей простоты. Данное преимущество достигается за счет простого, интуитивного, и четкого синтаксиса, удобной конструкции языка и поддержкой официальной документации с постоянным обновлением базы. Доступный и понятный синтаксис позволяет начинающему разработчику быстрее освоить материал,

приблизительно за полгода можно освоить базовые элементы работы с языком и уже начинать создавать свои программы, что нельзя сказать о C++, а продвинутому запомнить намного больше элементов построения кода. Из-за этого ускоряется скорость обучения и разработки, упрощается процесс рефакторинга и исправления ошибок.

Так же стоит отменить о популярности языка. Связано это с его относительной молодостью и распространением во многих направлениях разработки, начиная от прикладных приложений, заканчивая web-сервисами. Из-за такой универсальности, повышается спрос и количество вакансий. Но все равно спрос сильно превышает количество вакансий из-за этого данный язык является высокооплачиваемым.

Основные направления разработки. Язык C# практически универсален. Можно выделить следующие направления разработки на языке C#:

- Видеоигры;
- ПО для защиты систем;
- Приложения для Windows;
- Мобильные приложения;
- Web-разработка;
- Кроссплатформенные приложения.

Поддержка платформы .NET. Когда говорят C#, нередко имеют в виду технологии платформы .NET (Windows Forms, WPF, ASP.NET, Xamarin). И, наоборот, когда говорят .NET, нередко имеют в виду C#. Язык C# был создан специально для работы с фреймворком .NET, однако само понятие .NET несколько шире. Можно выделить следующие ее основные черты: [15]

- Поддержка нескольких языков. .NET поддерживает несколько языков: наряду с C# это также VB.NET, C++, F#, а также различные диалекты других языков, привязанные к .NET, например, Delphi.NET. При компиляции код пересобирается в общий машинный язык называемым Common Intermediate Language. Именно поэтому на одном проекте

возможно добавление различных языков, ведь при сборке весь код автоматически приводится к одному машинному коду и тем самым облегчает разработку;

- Кроссплатформенность. .NET является переносимой платформой (с некоторыми ограничениями). Используя различные технологии на платформе .NET, можно разрабатывать приложения на языке C# для самых разных платформ - Windows, MacOS, Linux, Android, iOS, Tizen;
- Сильная библиотека классов. .NET дает возможность использовать одну библиотеку классов для разных языков. И какое бы приложение мы не собирались писать на C# - текстовый редактор, чат или сложный веб-сайт - так или иначе мы задействуем библиотеку классов .NET;
- Разнообразие технологий. Так как язык C# является общеязыковой, то он позволяет использовать множества технологий для разработки различных систем в различных направлениях и на различных платформах. Например, для работы с базами данных в этом стеке технологий предназначена технология ADO.NET и Entity Framework Core. Для создания качественных графических интерфейсов широко используются WPF и WinUI, если необходимо создать простой интерфейс без трудности в разметке можно использовать Windows Forms. Для разработки кроссплатформенных мобильных и десктопных приложений используется Xamarin/MAUI. Для создания веб-сайтов и веб-приложений - ASP.NET и т.д.;
- Производительность. Согласно ряду тестов веб-приложения на .NET 6 в ряде категорий сильно опережают веб-приложения, построенные с помощью других технологий. Приложения на .NET 6 в принципе отличаются высокой производительностью.

1.2.1 Обзор библиотеки iTextSharp

iText — это библиотека PDF, которая позволяет создавать, адаптировать, проверять и обслуживать документы в формате Portable Document Format

(PDF), что позволяет с легкостью добавлять функции PDF в программные проекты.[16] Данная библиотека применяется не только на языке C#, но и для Java и C++.

iText является мировым лидером в области инновационного программного обеспечения PDF, отмеченного наградами. Он используется миллионами пользователей - как с открытым исходным кодом, так и коммерческих - по всему миру для создания цифровых документов для различных целей: счета-фактуры, выписки по кредитным картам, мобильные посадочные талоны, юридическое архивирование и многое другое.

iTextSharp может выполнять следующие действия:

- Создание и генерация PDF;
- Преобразование HTML в PDF;
- PDF-редактирование;
- Множественная языковая поддержка;
- Извлечение данных;
- PDF в изображение;
- Шифрование;
- Цифровые подписи;
- Поддержка SVG;
- Распознавание текста (OCR);
- Оптимизация PDF;
- MS Office в PDF.

1.2.2 Обзор библиотеки OpenXML

Open XML SDK предоставляет инструменты для работы с документами Office Word, Excel и PowerPoint. Он поддерживает такие сценарии, как [17]:

- Высокопроизводительное создание текстовых документов, электронных таблиц и презентаций;
- Заполнение содержимого файлов Word из источника данных XML;

- Разделение (уничтожение) файла Word или PowerPoint на несколько файлов и объединение нескольких файлов Word/PowerPoint в один файл;
- Извлечение данных из документов Excel;
- Поиск и замена содержимого в Word/PowerPoint с использованием регулярных выражений;
- Обновление кэшированных данных и встроенных электронных таблиц для диаграмм в Word/PowerPoint;
- Модификация документа, например удаление отслеживаемых версий или удаление неприемлемого содержимого из документов.

1.3 Анализ среды разработки

Интегрированная среда разработки (IDE) — это многофункциональная программа, которая поддерживает многие аспекты разработки программного обеспечения. [18] Существует разные среды разработки поддерживающие программирование на C#. Рассмотрим следующие:

Visual Studio. Считается самой «правильной» средой разработки. Многие начинают писать программы именно на данной среде разработки. [18] Можно выделить следующие плюсы Visual Studio:

- Официальная. Из-за того, что язык и среда разработки была создана одной компанией Microsift, то данная среда будет являться более функциональной и грамотно построенной для разработки на языке C#. Во многих ситуациях для разработки необходимо использовать Visual Studio, так как только в нем могут присутствовать необходимы технологии разработки или данные технологии развиты на высоком уровне;
- Бесплатная. Visual Studio не является полностью бесплатной, в ней существуют различные лицензии для лучшей работы в среде, но так же есть бесплатная «Community edition» версия, которой будет достаточно для многих задач, как начинающего разработчика, так и для профессионала;

– Функциональная. С распространением различных технологий даже большие компании могут не успевать за разработкой новых систем для более удобной работы со средой. Именно поэтому разработчики Visual Studio внедрили возможность создания различных плагинов, чтобы сами пользователи могли расширить функционал работы со средой;
– Поддерживает платформы .NET. Visual Studio с использованием C# языка очень хорошо развита за счет .NET платформы, который на данный момент занимает одну из лидирующих мест в его сегменте;
– Облачные хранилища. В эпоху сетевой паутины развитие облачного хранилища является важной частью любой системы, связанной с сохранением большого объема информации. Данные технологии присутствуют и в Visual Studio;
– Корпоративность. Технология бэклога позволяет членам команды взаимодействовать при гибкой методологии разработки.

Из минусов можно отметить:

– Баги при переходах с триал-версии. У многих пользователей возникали проблемы с открытием проектов и настройками среды после перехода на платную версию и с этом пока что ничего не сделали;
– Сложность. Из-за того, что Visual Studio предлагает очень большой функционал в начале работы очень сложно ориентироваться в его элементах, поэтому порог вхождения в данную среду достаточно высокий.

Project Rider. Rider — это кросс-платформенная IDE для .NET-разработчиков, основанная на платформе IntelliJ и ReSharper. Rider поддерживает .NET Framework, новую платформу .NET Core и проекты на основе Mono. IDE позволяет разрабатывать десктопные приложения, .NET-сервисы и библиотеки, игры на движке Unity, мобильные приложения Xamarin, веб-приложения ASP.NET и ASP.NET Core. [19] Rider, по мимо главной своей

задачи, имеет ряд функций, которыми обладает Visual Studio, так и новыми. К плюсам можно отнести:

- ReSharper. Данный плагин был создан для упрощения работы с Visual Studio, но с появлением собственной среды разработки он дополнил функционал среды;
- Поддержка полного цикла. Данная черта является уникальной для продуктов данной компании, она позволяет пройти все этапы жизненного цикла и хорошо координирует разработчиков по этим же этапам;
- Функциональность. Project Rider позволяет подключить MSBuild и XBuild, работать с CLI-проектами и организовать отладку приложений .NET and Mono. Множество опций для быстрого создания кода улучшает производительность;
- Multiple runtime. Поддержка нескольких запущенных программ;
- Кроссплатформенность. Project Rider работает с Windows, Linux и MacOS;
- Контроль версий. Инструмены Rider позволяют без лишних загрузок контролировать работы в различных сервисах, таких как Git, Mercurial и TFS.

К минусам можно отнести:

- Молодость. Часть функциональности еще в разработке, не все стартовые баги исправлены;
- Стоимость. Самая дешевая версия Project Rider обойдется в 139 долларов за первый год использования. Но есть триал-версия и специальные предложения для студентов и непрофильных организаций.

Среды разработки, которые были показаны, на данным момент, являются самыми популярными и востребованы, но для создания полноценного ПО так же необходимы выбрать правильный фрейворк для создания интерфейса ПО. В ряд фреймворков для C#, поддерживаемых вышесказанным средам разработки можно отнести:

Windows Forms – это платформа пользовательского интерфейса для создания классических приложений Windows. Она обеспечивает один из самых эффективных способов создания классических приложений с помощью визуального конструктора; [20]

Windows Presentation Foundation (WPF) – фреймворк для быстрого создания интерфейсов. Данная технология позволяет создавать графические интерфейсы с использованием, так называемого, языка разметки XAML. XAML представляет собой расширяемый язык разметки приложений. Данный способ, при необходимых навыках очень удобен и полезен. С его помощью можно создавать множество элементов современных графических интерфейсов. А с использованием дополнительных программных средств, таких как Blend, можно изменить структуру каждого стандартного элемента до неузнаваемости. Но стоит отметить, что для работы с WPF необходимо иметь на машине клиента платформы .NET Framework 3.0, 3.5, 4 или выше;

Xamarin – фреймворк, добавляющий C# функции компилирования кода с адаптацией под различные платформы, включая Windows, Android и iOS. Проще говоря, он делает ПО универсальным;

ASP.NET – технология для создания «мостов» между серверным кодом и клиентской частью программ. Технология позволяет соединять прикладный приложения с средствами сетевой паутины, качественно и просто построить страницы и создавать полноценные веб-сервисы для работы пользователя.

1.4 Обзор и анализ существующих проектных решений, выявление их достоинств и недостатков

Хоть и соединение способа стеганографии и криптографии относительно новое, но на данный момент уже существуют различные аналоги ПО. Рассмотрим такие средства скрытой передачи данных, как QuickStego, Xiao Steganography, OpenStego, Camouflage, SilentEye и OpenPuff. Приведено лишь небольшое количество существующих решений, однако, можно выделить в них

общие достоинства или недостатки, которые также будут встречаться и в других решениях.

Многие из приложений имеют один большой недостаток, это использование только стеганографических методов защиты информации, который дает куда меньше времени для крипто анализа, тем самым данные могут быть быстро обнародованы. Другие программы, такие как QuickStego, имеют в своем арсенале около 30 криптографических способов и 25 хеш-функций, но при этом время шифрования таких данный будет увеличено в разы, что не допустимо в передаче больших объемов данных.

Так же хочется отметить финансирования таких ПО. Очень много ПО являются бесплатными, что дает право любому злоумышленнику «вскрыть» ПО и посмотреть способ реализации шифрования, ведь почти все они основаны одном алгоритме и из-за этого, если не применяется криптографические методы защиты, можно очень просто раскрыть. Другие же ПО, которые имеют в своем арсенале, по мимо стеганографиских методов защиты, криптографические методы защиты, генерацию случайных пикселей и др. являются платными и порой могут брать за месячную работу до 500 $, что может быть не целесообразно если ПО пользоваться не на постоянной основе.

Таким образом, после изучения различных аналогов ПО можно выделить их достоинства и недостатки, показанные на таблице 1.

Таблица 1 – Достоинства и недостатки аналогов.

Достоинства	Недостатки
Не требуется финансирование для разработки, если ПО бесплатное;	Использование только стеганографических методов защиты информации;
Если существует криптографические алгоритмы, они могут достигать 30 методов и 25 хеш-функций.	Финансирования таких ПО может достигать, за месячную работу, до 500$
	Бесплатные ПО дают право любому злоумышленнику «вскрыть» ПО и посмотреть способ реализации шифрования

Продолжение таблицы 1

	В случае сложных алгоритмов время шифрования таких данный будет увеличено в разы

1.5 Обоснование необходимости разработки программы

Обоснованиями создания ПО для скрытой передачи данный можно назвать стоимость готовых решений, их уязвимость, а также сложность использования.

В связи с большой активностью различных хакерских атак для получения данный и перехвата информации, одним из важных требований при разработке какой-либо информационной системы будет требование безопасности.

Данное ПО вмещает в себя способы криптографии и стеганографии, а также использование алгоритмов рандомизации обработанных пикселей. Для устойчивости информация шифруется криптографически, а после вноситься в исходное изображение. Обратная дешифровка происходит в обратном порядке. Рассмотрены характеры угроз и возможные ограничения модели.

Модель скрытой передачи данных необходимо не столько для защиты самой передачи информации, сколько для получения времени на использование информации, который заполучил злоумышленник. Если злоумышленник потратит больше времени на то, чтобы вскрыть информацию для отдела или команды информационной безопасности для закрытия уязвимостей. А времени на вскрытие может уйти достаточно, ведь злоумышленнику необходимо не просто вскрыть информацию, но и, в первую очередь, понять, что у него находится важная информация, а только после этого пытаться изъять эту информацию, что будет является уже проблемой.

2 ПРОЕКТИРОВАНИЕ

2.1 Разработка концепции, архитектуры построения и платформы реализации программы

В настоящее время наиболее распространенными архитектурами являются:

файл-сервер;

клиент-сервер;

многоуровневая архитектура.

Файл-серверная архитектура подразумевает под собой то, что сервер возлагает на себя лишь функцию хранения данных, а обработка производится на клиентских машинах. Это означает, что данные необходимо передавать по сети, что приведет к сильной загрузке сетевого трафика. А это в свою очередь приведет к снижению производительности при увеличении числа пользователей. Также при реализации архитектуры файл-сервер, проблема целостности, согласованности и одновременного доступа к данным решается децентрализовано: данные хранятся на сервере, а обрабатываются на клиенте. Вследствие этого снижается надежность приложения. Еще одним недостатком являются высокие затраты на модернизацию и сопровождение сервисов бизнес - логики на каждой клиентской рабочей станции. Однако данная архитектура обладает и рядом преимуществ, таких как низкая стоимость разработки, высокая скорость разработки и невысокая стоимость обновления и изменения программного обеспечения.

Архитектура клиент-сервер лишена недостатков вышеописанной архитектуры, т.к. сервер баз данных не только обеспечивает доступ к общим данным, но и выполняет их обработку. Клиент посылает на сервер запросы, на языке «понятном» серверу, а он в свою очередь обрабатывает запрос, контролируя при этом целостность и согласованность данных, и возвращает на клиента результат отработанного запроса. В результате нагрузка на сеть

снижается: клиенту больше не нужно обрабатывать промежуточные данные. Хранение и обработка производится централизовано, поэтому данная архитектура надежнее архитектуры файл-сервер. К недостаткам клиент-серверной архитектуры относятся, во-первых, достаточная сложность разработки системы из-за необходимости исполнять бизнес-логику и обеспечивать интерфейс с пользователем в одной программе и высокие требования к рабочим станциям по той же причине.

Следующей ступенью развития архитектур ИС стала многоуровневая архитектура, в которой бизнес-логика выполняется на сервере приложений. Многоуровневая архитектура обладает следующими достоинствами:

- масштабируемость;
- конфигурируемость - изолированность уровней друг от друга позволяет быстро и простыми средствами переконфигурировать систему при возникновении сбоев или при плановом обслуживании на одном из уровней;
- высокая безопасность;
- высокая надёжность;
- низкие требования к скорости канала (сети) между терминалами и сервером приложений;
- низкие требования к производительности и техническим характеристикам терминалов, как следствие снижение их стоимости.
- Однако, несмотря на неоспоримые достоинства, данная система не получила распространения, по следующим причинам:
- сложность разработки систем на основе многоуровневой архитектуры, т.к., очень сложно «состыковать» различные модули, особенно если они написаны разными группами. А изменение в одном модуле, как правило, вызывает лавинообразные изменения в остальных, и с этой точки зрения даже простую систему, основанную на многоуровневой архитектуре, будет сложнее выполнить в 2 раза;

– высокие требования к производительности серверов приложений и сервера базы данных, а, значит, и высокая стоимость серверного оборудования;
– высокие требования к скорости канала (сети) между сервером базы данных и серверами приложений;
– высокая сложность администрирования.

Рассмотрев все достоинства и недостатки каждой из архитектур, в виду отсутствия за ненадобностью сервера и сетевого взаимодействия, мы можем выделить многоуровневый шаблон.

В качестве платформ выберем Windows виде исполняемого PE файла для использования в Windows-системах.

2.2 Структура программы

Состав программы – искусственно выделенные программистом взаимодействующие части программы. В рассматриваемой программе можно выделить несколько основных частей, без которых функционирование будет невозможно или слишком осложнено:

– Считыватель изображений и текстовых документов различных форматов;
– Генератор случайных последовательностей символов, который будет привязан только к индивидуальному ключу потребителя;
– Преобразователь изображения в матрицу бинарных значений для внедрения информации;
– Алгоритм криптографического шифрования исходного текста, который не будет привязан к определенному ключу, для невозможности подбора.

2.3 Техническое обеспечение

– операционная система: Windows;
– оперативная память не менее 2 Гб;
– встроенная память – не менее 50 Мб для хранения приложения.

2.4 Описание логической структуры ПО

Структура ПО - это логическая взаимосвязь отдельных элементов, представленная в иерархическом порядке. Определение структуры программного обеспечения - это абсолютно необходимый этап в работе над проектом, поэтому ему всегда уделяется особое внимание. При разработке структуры необходимо учесть тему будущего ресурса, доступность информации для пользователя, тщательно продумать возможный путь посетителей по окнам, разработать навигацию, которая будет понятна для пользователя с первого взгляда. Структура программного обеспечения включает в себя логическую и физическую структуры.

Структура программного обеспечения для скрытой передачи данных вмещает в себя:

а) Окно «Зашифровать»:
   1) Загрузка изображения;
   2) Загрузка текста;
   3) Окно ввода пароля и его хеширование;
   4) Модуль шифрования:
      – Добавление метки шифрования;
      – Добавление пароля;
      – Добавление текста;
   5) Сохранение изображения;

б) Окно «Расшифровать»:
   1) Загрузка изображения;
   2) Ввод пароля;
   3) Хеширование пароля;
   4) Модуль расшифрования:
      – Изъятие и проверка метки шифрования;
      – Изъятие и проверка пароля;
      – Разоблачение текста;

5) Сохранение текста.

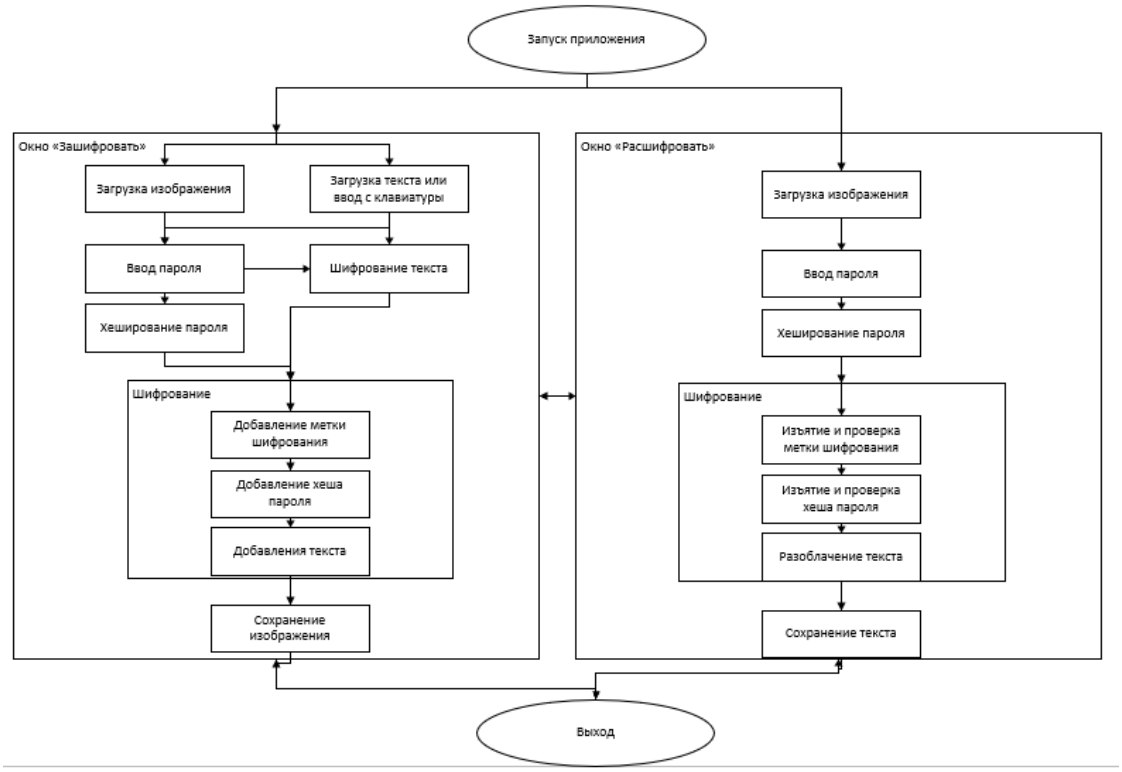

Рисунок 8 - Логическая структура ПО

2.5 Создание классов и их использование

Для того, чтобы созданное программное обеспечение было удобным в использовании были созданы несколько классов и модулей.

Сначала был создан абстрактный класс CryptoPhotoController.cs. Он вмещает в себе все связанные действия над шифрование и расшифрованием изображения. Такие как:

- Константные переменные для точного размера метки шифрования, количества символов в изображении, максимального количества символов, количество символом для ключа шифрования;
- Метод для преобразования количества символом до количества равному константе;
- Метод сохранения изображения;
- Метод загрузки изображения;
- Методы загрузки текста;
- Методы сохранения текста;

– Иные методы для конвертации различных типов данных.

От этого класса были созданы и наследованы еще 2 класса StartEncrypClass.cs и DeStartEncrypClass.cs, которые отвечают за главную логику шифрования и расшифрования изображения. Они работают по принципы LSB-метода.

Класс StartEncrypClass.cs содержит в себе следующую логику:

– Добавление метки шифрования в изображение;
– Добавление количества символов в изображение;
– Добавление ключа в изображение;
– Добавление текста в изображение.

При каждом добавлении класс проверяет значение с константой и, если оно больше данного значения работа класса немедленно заканчивается.

Класс DeStartEncrypClass.cs содержит в себе следующую логику:

– Изъятие метки шифрования;
– Изъятие количества символов в изображение;
– Изъятие ключа в изображение;
– Изъятие текста в изображение согласно количеству символов в изображении.

Для криптографических операций создан класс ShifrController.cs. Он обрабатывает текст и шифрует его в выбранной криптографической схеме AES.

2.6 Описание физической структуры программного обеспечения

После создания логической структуры ПО и объявлении основных классов и модулей необходимо перейти к физической структуре.

Физическая структура программного обеспечения состоит из форм, модулей с их заголовочными файлами и файлами реализации, установок параметров проекта, ресурсов и т.д. Вся эта информация размещается в файлах. На рисунках 9-10 показана конечная физическая структура программного обеспечения

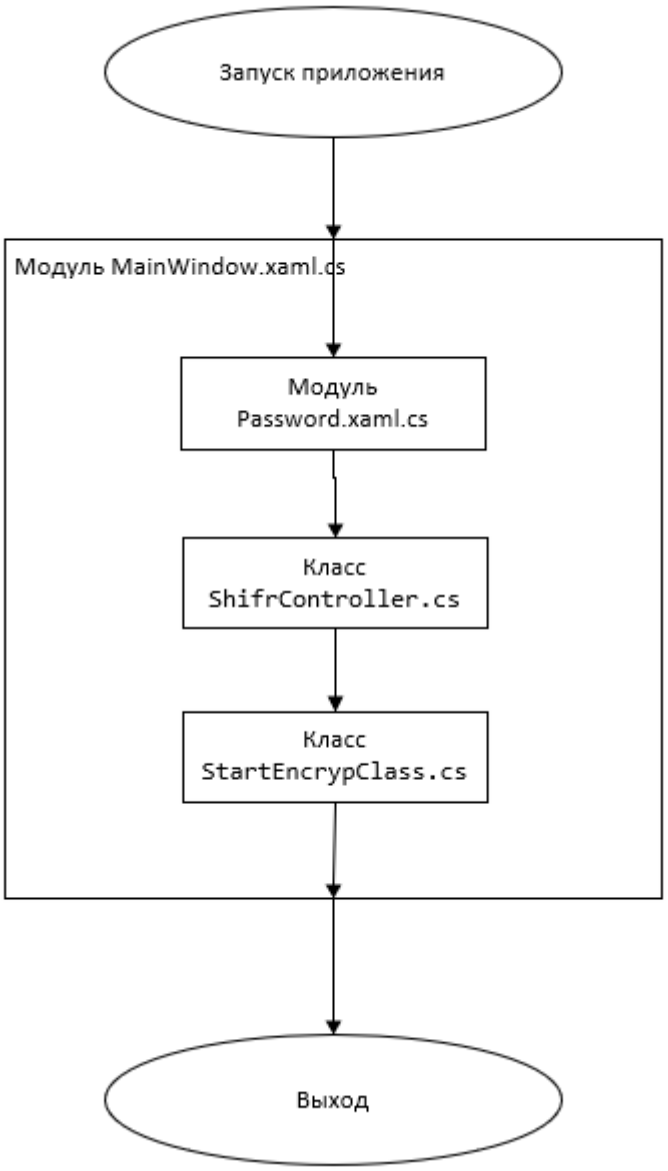

Рисунок 9 – Физическая структура ПО окна «Зашифровать»

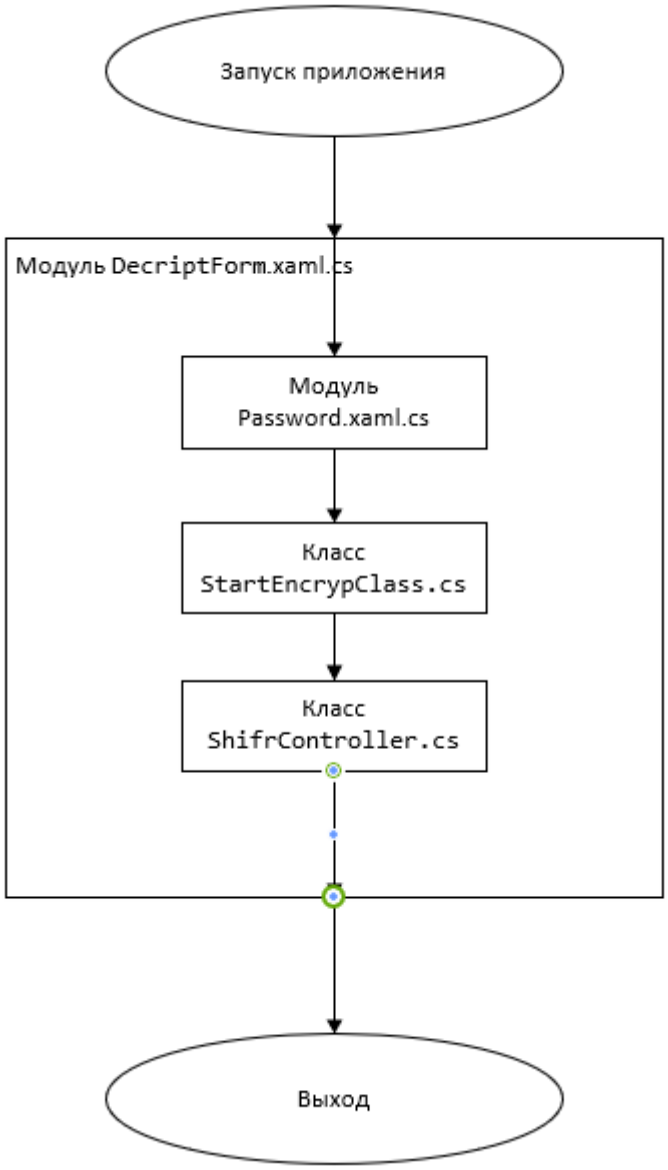

Рисунок 10 – Физическая структура ПО окна «Расшифровать»

3 РАЗРАБОТКА ПРОГРАММНОГО ПРОДУКТА

3.1 Обзор программах средств

Данной программное обеспечение может применяться различных видах деятельности где необходима скрытая передача данных по открытому каналу.

3.1.3 Системные требования

На таблице 2 изображены минимальные системные требования к ПО для нормальной работы.

Таблица 2 - Минимальные системные требования

Компонент	**Минимальные требования**
Операционная система	Windows 7, 8, 8.1, 10, 11
Процессор	Intel Core i3
Оперативная память	2 Гб ОЗУ
Место на жестком диске	50 Мб
Дополнительное оборудование	Клавиатура, мышь, монитор
Дополнительно	.Net Framework 4.7.2

3.1.4 Общие сведения

Программа разработана в среде Microsoft Visual Studio C# 2019 Community, ПО разработана с использованием OpenXml и iTextSharp.

3.1.5 Входные и выходные дынные

Входными и выходными данными являются:

а) Текст:

1) Записанные с клавиатуры;
2) Загруженные из файлов формата TXT;
3) Загруженные из файлов формата PDF;
4) Загруженные из файлов формата DOCX.

б) Изображение:

1) Загруженные из файлов формата PNG;
2) Загруженные из файлов формата JPEG;
3) Загруженные из файлов формата JPG;
4) Загруженные из файлов формата BMP.

3.1.6 Рекомендации по входным данным

На таблице 3 изображены рекомендации по входным данный, для того, чтобы пользователь понимал, что именно нужно вводит в ПО.

Таблица 3 - Рекомендации по входным данным

Тип	Рекомендация
Минимальный размер текста	1 символ
Максимальный размер текста	$9 * 10^6$ символов
Минимальная высота изображения	16 пикселей
Минимальная длина изображения	2 пикселя
Максимальная высота изображения	6000 пикселей
Максимальная длина изображения	6000 пикселей
Формат текста	TXT, DOCX, PDF
Формат изображения	PNG, BMP, JPEG, JPG
Содержание изображения	В основном не однотонное
Содержание текста	Все символы Windows-1251

3.2 Создание главного меню

Для разработки главного меню необходимо учитывать несколько параметров:

- Главное меню должно быть интуитивно понятным;
- Главное меню должно содержать клавиши перехода между формами;

– Из главного меню можно будет выходить из приложения;
– Интерфейс главное меню не должно вызывать отторжения у пользователя.

Создание главного меню проводилось в среде разработки WPF с использованием приложения Blend для улучшения качества визуальной части. В его функционал входит:

– Кнопка перехода на окно шифрования. Для ее создания использовалась стандартный элемент Button, которое было модифицировано. Данная кнопка отрывает окно для шифрования информации в изображении;
– Кнопка перехода на окно шифрования. Для ее создания использовалась стандартный элемент Button, которое было модифицировано. Данная кнопка отрывает окно для расшифрования информации из изображения;
– Кнопка выхода из приложения. Для ее создания использовалась стандартный элемент Button, которое было модифицировано. Данная кнопка закрывает ПО и дополнительно отчищает потоки и процессы;
– Кнопка перехода в руководство пользователя. Для ее создания использовалась стандартный элемент Button, которое было модифицировано вставкой изображение вопросительного знака для перехода в pdf файл, который появляется при установке ПО;
– Кнопка сворачивания ПО. Для ее создания использовалась стандартный элемент Button, которое было модифицировано. Она находится в верхнем правом углу, где пользователь интуитивно понимает предназначение данной кнопки;
– Изображение с название ПО. Для ее создания использовалась стандартный элемент Image.

3.3 Создание окна шифрования

Для разработки основного окна шифрования использовался так же WPF и приложение Blend. В ее функционал входит:

– Кнопка загрузки текста. При нажатии на данную кнопку открывается диалоговое окно для загрузки текста в форматах pdf, docx, doc, txt. Для загрузки текста использовались команды из библиотек ITextSharp, OpenXML и стандартные библиотеки C#. При загрузке файлов отличных от txt текст файл сначала переходит в поток, затем разделяется на абзацы и сохраняются в отдельной переменной, текст которой переходит в просмотр и редактирования текста. После этого поток закрывается и не потребляет лишнюю память. Так же текст сразу же сохранятся в шифрованном виде согласно алгоритму, выбранному пользователем;
– Кнопка загрузки изображения. При нажатии на данную кнопку открывается диалоговое окно для загрузки изображения в форматах PNG, BMP, JPEG, JPG. Каждый из форматов загружается с помощью встроенной библиотеки C#. Каждое изображение переводится сначала в формат bmp, а затем сохраняется в переменной с типом BitmaImage;
– Компонент просмотра и редактирования текста. В данном компоненте можно загрузить, изменить или добавить текст для шифрования;
– Компонент просмотра изображения. В данном компоненте можно посмотреть загруженное изображение;
– Ползунок выбора битности. Данный ползунок используется для ручного задания количества битов для заполнения в цветах изображения. По умолчанию стоит значение 2;
– Кнопка выбора метода шифрования. Данная кнопка открывает окно выбора метода шифрования;
– Кнопка возврата в главное меню. Данная кнопка открывает окно главного меню;
– Кнопка шифрования. При нажатии данной кнопки вначале открывается окно ввода пароля, а после его ввода начинается алгоритм шифрования. Вначале к матрице изображения добавляется плюс-минус один алгоритм для дополнительной защиты. Далее производится проверка на

нахождения в тексте знака шифрования, если такая метка присутствует, то алгоритм заканчивается и появляется диалоговое окно ошибки. Если такого знака нет, то вначале она наносится в изображение, после этого наносится хеш пароля, длина текста, алгоритм шифрования и битность. После этого шифрованный текст в формате битов вносится в изображения с использованием LSB-метода (Рисунок 11) в случайные места изображения.

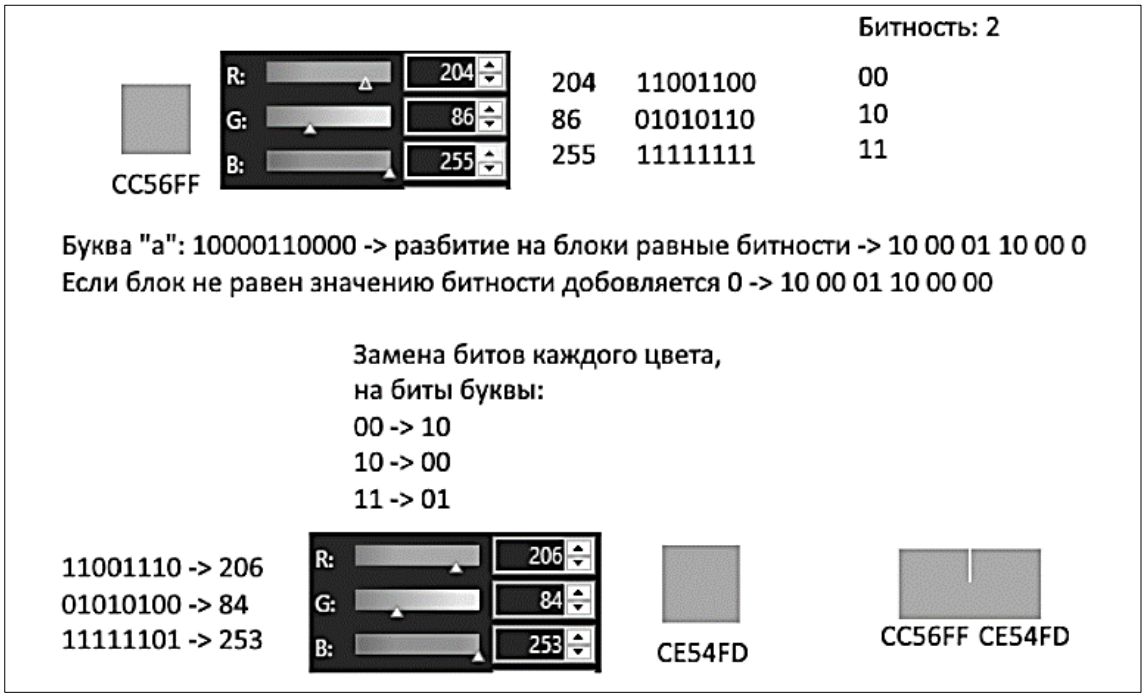

Рисунок 11 – Алгоритм LSB с выбором битности замены

– Кнопка сохранения изображения. После шифрования текста появляется данная кнопка, при нажатии на которую, отрывается диалоговое окно сохранения изображения в форматах PNG, BMP, JPEG, JPG;
– Текст показа количества битов текста и изображения. Данный текст показывается после загрузки текста или изображения, с помощью которого можно понять вместится ли текст в изображения или нужно что-то изменить.

Так же при запуске алгоритма если в программе были введены некорректные данные или текст не вмешается в изображение, то появляется

диалоговое окно ошибки. В случае сохранения появляется диалоговое окно, с помощью которого можно открыть файл.

3.3.1 Создание окна выбора метода криптографической защиты

Для достижения более устойчивой защиты было создано окно выбора алгоритма криптографических преобразований. Данное окно представляет из себя список различных алгоритмов доступных для шифрования. Если данное оно не было открыто, то информация будет шифроваться алгоритмов AES. Так же возможно использование кнопки сброса, в таком случае криптографические преобразования применятся к тексту не будут. После выбора алгоритма программа получает данные в виде номера, с помощью которого находится выбранных алгоритм и, согласно документации, об используемом алгоритме, шифруется.

Из-за различной специфики алгоритмов в некоторых из их по мимо ключа шифрования (Key) используется еще и вектора инициализации (IV), оба эти значения берутся из пароля введенным пользователем, но для сложности разоблачения и универсальности вначале пароль кэшируется и после, в качестве семени случайной генерации, создается новое число, как для ключа шифрования, так и для вектора инициализации. Для таких действий был создан отдельный метод для создания случайного числа. Сделан он был для того, чтобы сократить код, а также из-за того, что не во всех алгоритмах нужен вектор инициализации, таким образом создавать отдельное число для каждого значения будет эффективнее.

На данный момент были организованны следующие криптографические алгоритмы: AES, XOR, Rijndael, RC2, RC4, DES, Атбаш, Виженер и Base64. В дальнейшем этот список будет пополняться новыми алгоритмами, включая ассиметричные алгоритмы, а также добавиться выбор алгоритма хеш-функций.

3.4 Создание окна расшифрование

Окно расшифрования необходим для изъятия информации из изображения. Перед тем, чтобы в нем работать необходимо создать уже

зашифрованное изображения с помощью окна шифрования. Дальше разберем основной функционал данного окна:

– Кнопка загрузки изображения. При нажатии на данную кнопку открывается диалоговое окно для загрузки изображения в форматах PNG, BMP, JPEG, JPG. Каждый из форматов загружается с помощью встроенной библиотеки C#. Каждое изображение переводится сначала в формат bmp, а затем сохраняется в переменной с типом BitmaImage;
– Кнопка расшифрования. При нажатии данной кнопки вначале открывается окно проверки пароля, а после его ввода начинается алгоритм расшифрования. Вначале производится проверка на нахождения в тексте знака шифрования, если такая метка не присутствует, то алгоритм заканчивается и появляется диалоговое окно ошибки. Если такой знак присутствует, то из изображения достается хеш пароля, длина текста, алгоритм шифрования и битность. После этого текст, согласно битности, изымается из изображения в формате битов с использованием LSB-метода (Рисунок 12).

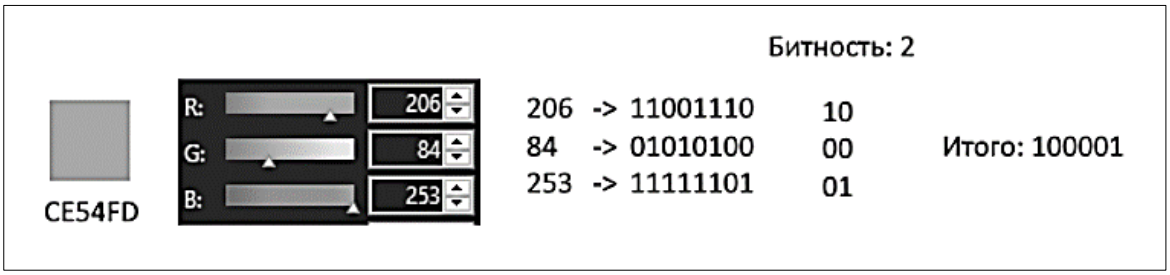

Рисунок 12 – Алгоритм расшифровки LSB с выбором битности замены

Так как ранее для семени случайной генерации использовался хеш пароля, то с тем же семенем биты изымаются. Далее, согласно номеру шифрования, текст передается методу для криптографического расшифрования. После этого, на место пред. просмотра изображения, появляется окно просмотра и изменения текста, и появляется кнопка сохранения;

- Кнопка сохранения текста. После расшифрования текста появляется данная кнопка, при нажатии на которую, отрывается диалоговое окно сохранения текста в форматах TXT, DOCX, PDF;
- Кнопка возврата в главное меню. Данная кнопка открывает окно главного меню;
- Компонент просмотра изображения. В данном компоненте можно посмотреть загруженное изображение;
- Компонент просмотра и редактирования текста. В данном компоненте можно просмотреть и изменить текст после расшифрования.

3.5 Создания дополнительной защиты данных с помощью парольной защиты

Бывают случаи, когда ПО для защиты данных передается мошенникам и таким образом, если не организовать дополнительную защиту, любую информацию можно будет изъять. Именно поэтому дополнительная защита должна быть организованна и в данном ПО. Для защиты данных была выбрана парольная защита любой длины. Выполнена она в формате нового окна с полем ввода с возможностью просмотреть пароль, если есть сомнения в его правильности с помощью специальной кнопки, находящейся с правой стороны от поля ввода. После ввода пароля необходимо нажать кнопка принятия и после этого пароль будет сохранен и отправлен на дальнейшую обработку. Так как создание нового пароля и его проверка его на правильность имеет одинаковое, для обоих случаев, поле ввода, то данное окно используется как для создания пароля, так и для его проверки.

4 ОПИСАНИЕ ПОЛЬЗОВАТЕЛЬСКОГО ИНТЕРФЕЙСА

4.1. Знакомство с программным обеспечением

Перед тем как приступить к работе с ПО, необходимо ознакомиться со следующей информацией:

– Главное окно ПО;
– Окно зашифрования;
– Окно расшифрования;
– Окно настроек;
– Окно ввода пароля.

4.2. Главное окно ПО

После запуска ПО "CriptoSteg" откроется главное окно (Рисунок 13)

Рисунок 13 - Главное окно ПО

В данном окне присутствуют все необходимы кнопки для работы с ПО:

– Кнопка Зашифровать (Рисунок 14);

– Кнопка Расшифровать (Рисунок 15);

– Кнопка Выход (Рисунок 16);

– Кнопка справки (Рисунок 17);

Рисунок 14 - Кнопка "#ЗАШИФРОВАТЬ"

Рисунок 15 - Кнопка "#РАСШИФРОВАТЬ"

Рисунок 16 - Кнопка "#ВЫХОД"

Рисунок 17 - Кнопка справки

При нажатии на кнопку справки откроется данное руководство.

При необходимости можно опустить окно используя (Рисунок 18), а также двигать окно задержав левую кнопку мыши на верхней части окна.

Рисунок 18 – Кнопка просмотра пароля

4.3. Окно зашифрования

При нажатии на клавишу Зашифровать в главном окне откроется новое окно (Рисунок 19)

В данном окне присутствуют компоненты:

– Кнопка загрузки текста (Рисунок 20);

– Кнопка загрузки изображения (Рисунок 21);

– Область для добавления и редактирования текста (Рисунок 22);

– Область миниатюры изображения (Рисунок 23);

– Ползунок для выбора количества перезаписываемых битов в изображении (Рисунок 24);

– Кнопка настроек (Рисунок 25);

– Кнопка возврата на главное окно (Рисунок 26);

– Кнопки зашифровать (Рисунок 27);

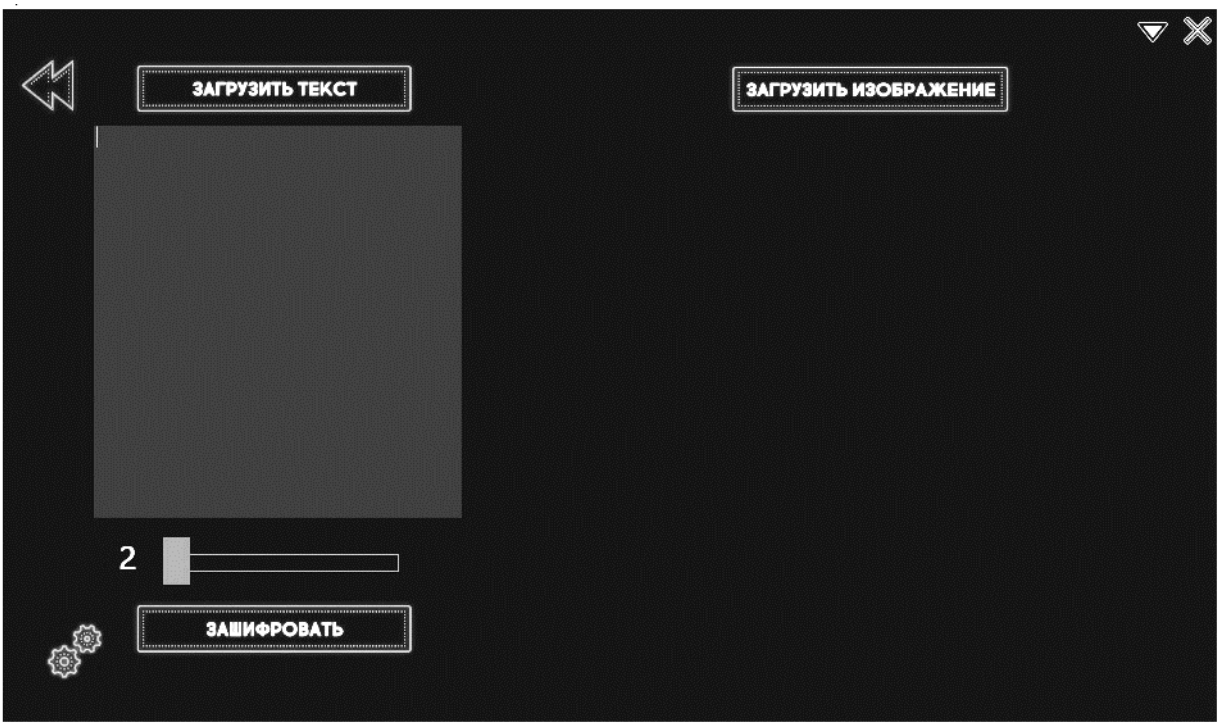

Рисунок 19 - Окно зашифрования

Для начала работы необходимо загрузить текст и изображение с помощью соответствующий кнопок. После нажатия на любую из кнопок откроется диалоговое окно где необходимо выбрать файл для загрузки. В случае добавление текста его можно добавить с клавиатуры с помощью области для добавления обработки текста (Рисунок 22).

Рисунок 20 - Кнопка загрузки текста

Рисунок 21 - Кнопка загрузки изображения

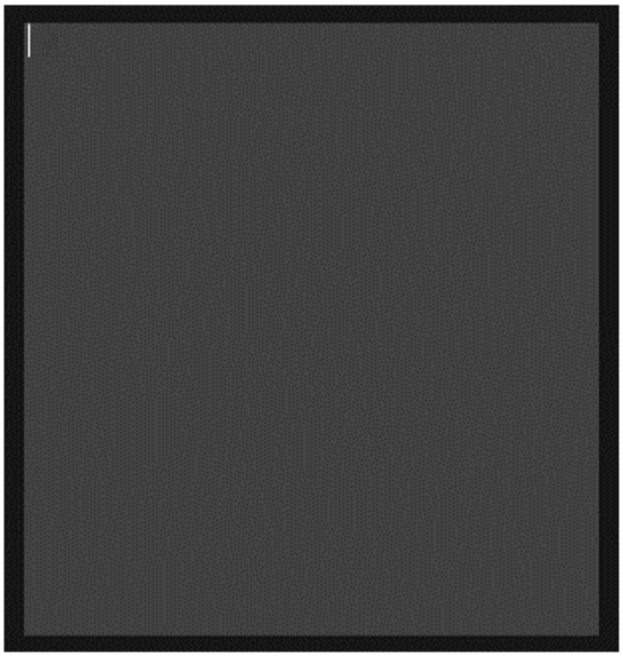

Рисунок 22 - Область для добавления и редактирования текста

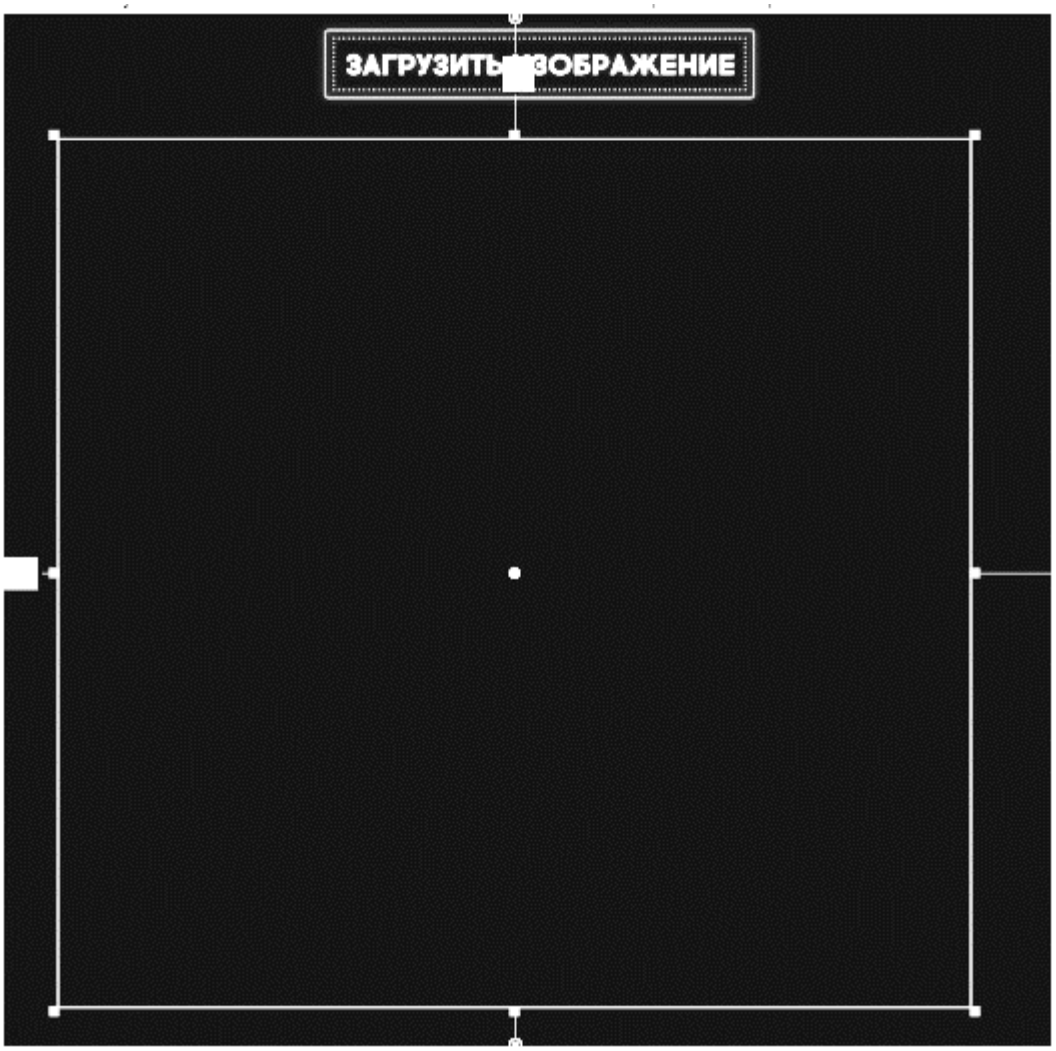

Рисунок 23 - Область миниатюры изображения

Если текст и изображение добавлены правильно над ними появится их длина и размер, соответственно. После данных манипуляций окно зашифрования примет рабочий вид (Рисунок 28).

Для того чтобы задать количество перезаписываемых бит можно воспользоваться двумя способами (Рисунок 24)

– С помощью удержания левой клавиши мыши на ползунке и перетягиванием в нужную сторону;

– Нажав левую клавишу мыши на значения битов и ввести значение с клавиатуры.

При изменении значения ползунка или ввода с клавиатуры нужного значения длина текста изменится, тем самым можно подобрать оптимальную длину, если длина текста больше размера изображения.

Предупреждение. Если задать количество бит слишком большое, то изображение может сильно исказиться, таким образом потеряется сам факт скрытой передачи.

Рисунок 24 - Ползунок для выбора количества перезаписываемых битов в изображении

Если необходимо сначала криптографически зашифровать текст, то можно воспользоваться кнопкой настроек (Рисунок 25). Подробнее об окне настроек.

Рисунок 25 - Кнопка настроек

По окончанию работы в окне зашифрования можно просто закрыть, свернуть или перейти на главное окно с помощью кнопки возврата (Рисунок 26).

Рисунок 26 - Кнопка возврата на главное окно

После всех пред. настроек для того чтобы скрыть текст в изображении необходимо нажать на кнопку "Зашифровать" (Рисунок 27). После этого необходимо придумать пароль в окне ввода пароля. Если операция прошла успешно появиться новая кнопка сохранения изображения (Рисунок 29). При нажатии на кнопку появиться диалоговое окно, в котором необходимо выбрать путь, имя файла и формат.

Если в файле уже присутствует текст, то появится диалоговое окно и шифрование прекратиться.

Рисунок 27 - Кнопка зашифровать

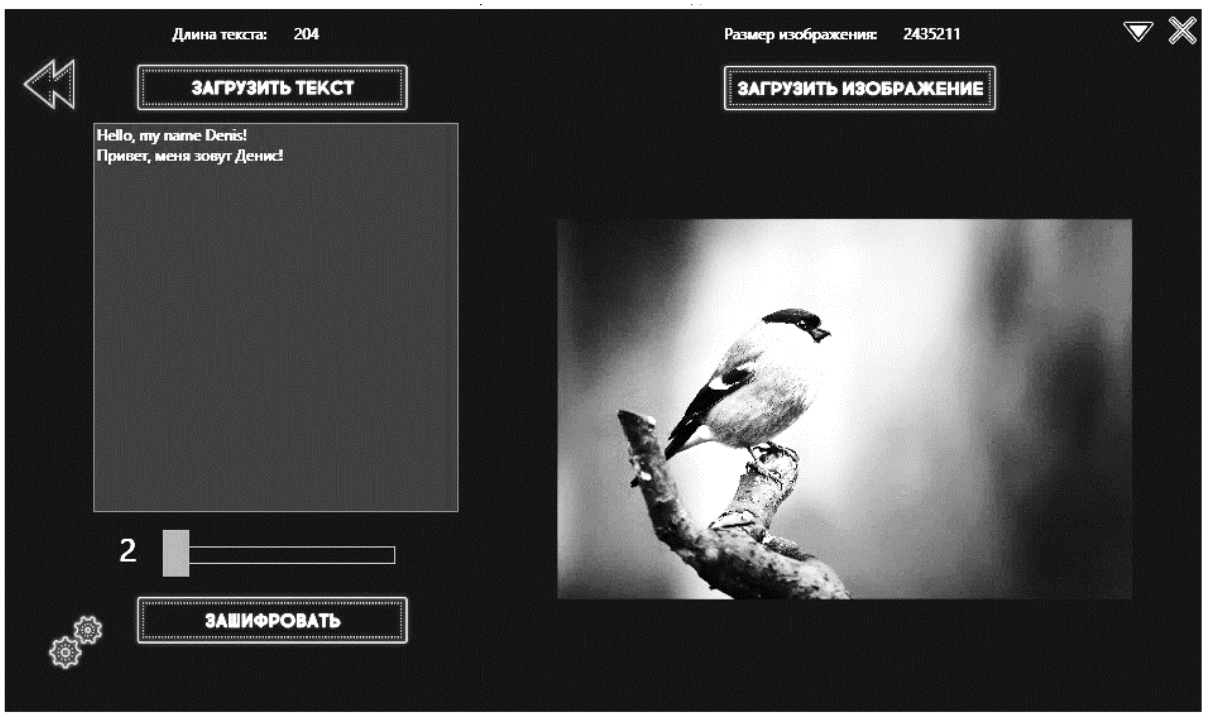

Рисунок 28 - Рабочий вид окна

Рисунок 29 - Кнопка сохранения

При загрузки текста и изображения, а так же при шифровании может появиться окно загрузки (Рисунок 30) которое автоматически отключится после окончании процедуры.

Рисунок 30 - Окно загрузки

4.4. Окно расшифрования

При нажатии на клавишу Расшифровать в главном окне откроется новое окно (Рисунок 31)

В данном окне присутствуют компоненты:

– Кнопка загрузки изображения (Рисунок 32);

– Область миниатюры изображения (Рисунок 33);

– Кнопка возврата на главное окно (Рисунок 34);

– Кнопка расшифровать (Рисунок 35);

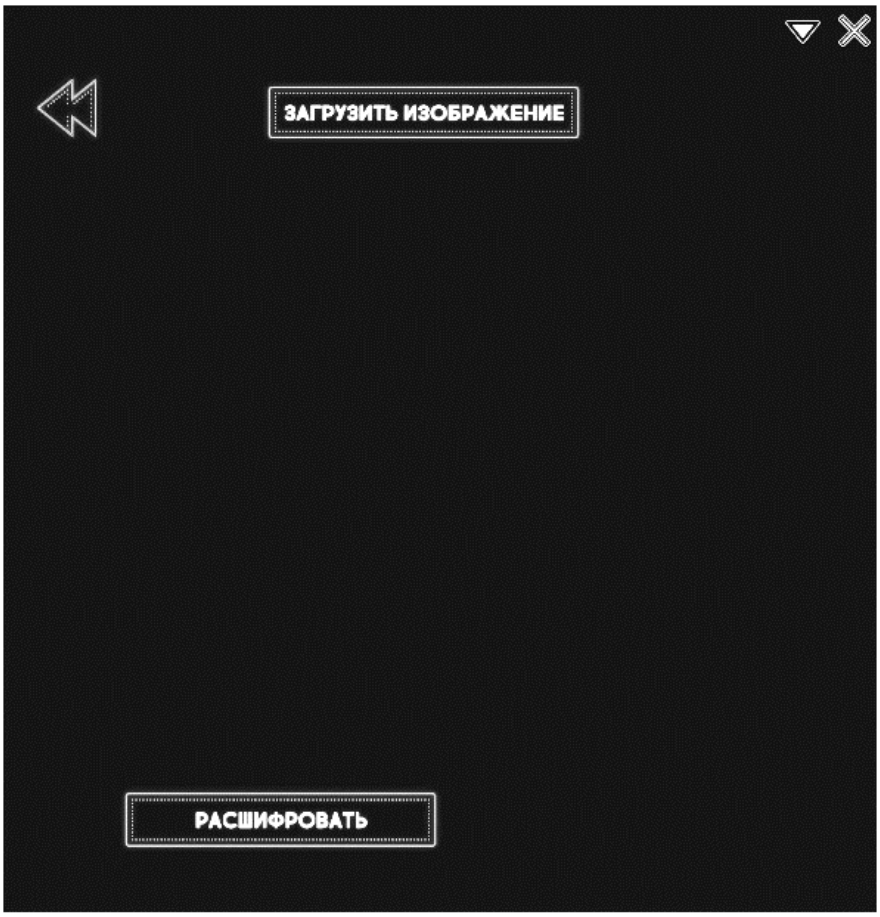

Рисунок 31 - Окно расшифрования

Для начала работы необходимо загрузить изображение с помощью соответствующей кнопки (Рисунок 32). После нажатия на кнопку откроется диалоговое окно где необходимо выбрать файл для загрузки.

Рисунок 32 - Кнопка загрузки изображения

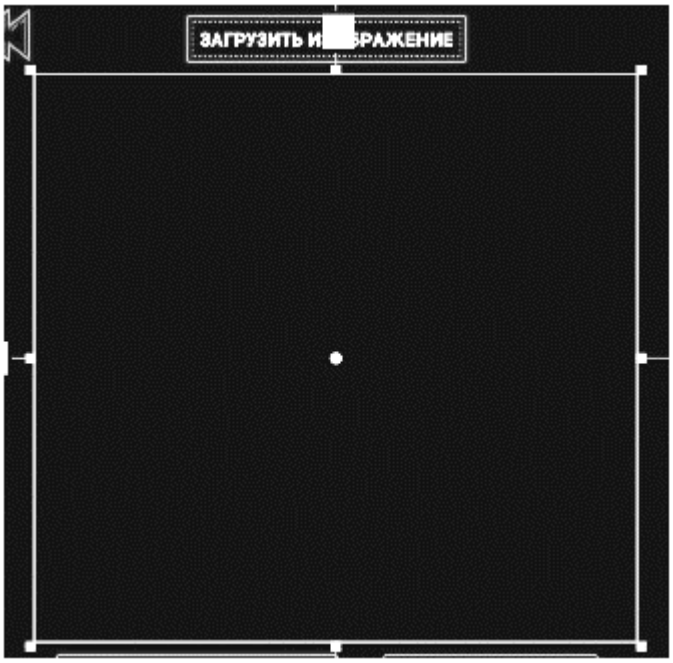

Рисунок 33 - Область миниатюры изображения

После загрузки файла в область миниатюры загрузиться выбранное изображение и окно примет рабочий вид (Рисунок 36).

По окончанию работы в окне расшифрования можно просто закрыть, свернуть или перейти на главное окно с помощью кнопки возврата (Рисунок 34).

Рисунок 34 - Кнопка возврата на главное окно

После всех пред. настроек для того чтобы разоблачить текст в изображении необходимо нажать на кнопку "Расшифровать" (Рисунок 35). После этого необходимо ввести пароль в окне ввода пароля. Если операция прошла успешно исчезнет область миниатюры и появиться область текста (Рисунок 37), а также новая кнопка сохранения изображения (Рисунок 38). При

нажатии на кнопку появиться диалоговое окно, в котором необходимо выбрать путь, имя файла и формат. В ином случае программа выдаст ошибку.

Рисунок 35 - Кнопка "Расшифровать"

Рисунок 36 - Рабочий вид окна

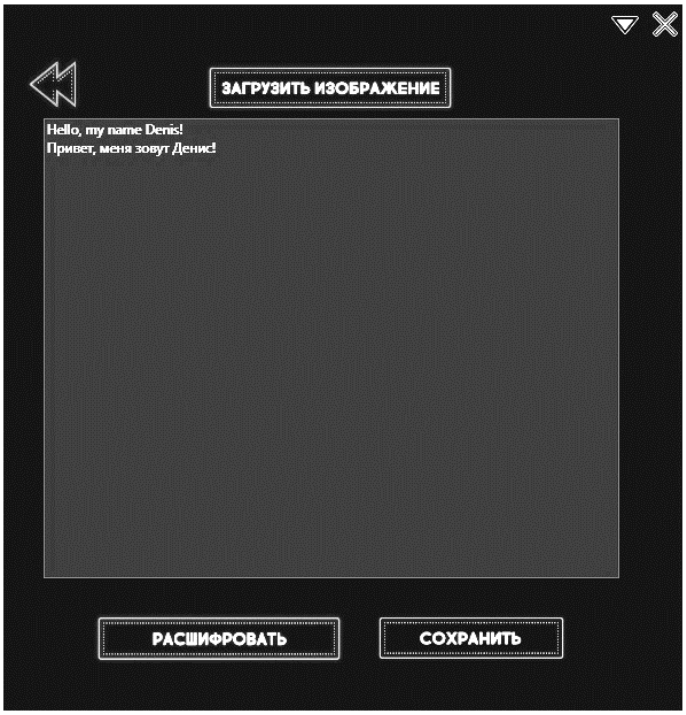

Рисунок 37 - Область просмотра и изменения текста

Рисунок 38 - Кнопка сохранения

При изображения, а так же при разоблачении текста может появиться окно загрузки (Рисунок 39) которое автоматически отключится после окончания процедуры.

Рисунок 39 - Окно загрузки

4.5. Окно настроек

При нажатии на кнопку настроек в окне зашифрования откроется новое диалоговое окно настроек (Рисунок 40).

В данном окне присутствуют компоненты:

– Список доступных алгоритмов шифрования (Рисунок 41);

– Кнопка сохранения (Рисунок 42);

– Кнопка сброса (Рисунок 43).

Рисунок 40 - Окно настроек

Для того, чтобы задать нужный алгоритм шифрования текста необходимо выбрать нужный алгоритм шифрования (Рисунок 41) и нажать дважды левой клавишей мыши на него или одним нажатием левой клавишей мыши и нажатием на кнопку сохранить (Рисунок 42).

Рисунок 41 - Список доступных алгоритмов шифрования

Рисунок 42 - Кнопка сохранения

Если вы уже выбрали алгоритм шифрования, но вам не нужно шифровать данные, то можно воспользоваться клавишей сброса (Рисунок 43).

Рисунок 43 - Кнопка сброса

4.6. Окно ввода пароля

Перед началом шифрования в окне зашифрования или расшифрования в окне расшифрования откроется новое диалоговое окно ввода пароля (Рисунок 44, Рисунок 45).

В данном окне присутствуют компоненты:

– Область ввода пароля (Рисунок 46);

– Кнопка подтверждения пароля (Рисунок 47);

– Кнопка показа пароля (Рисунок 48).

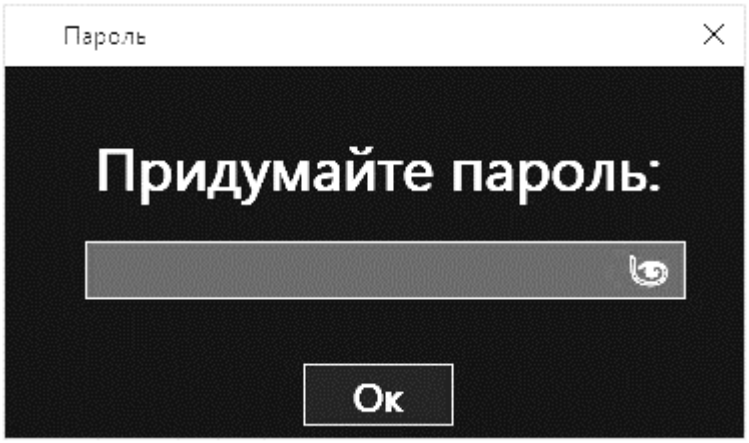

Рисунок 44 - Окно ввода пароля в окне зашифрования

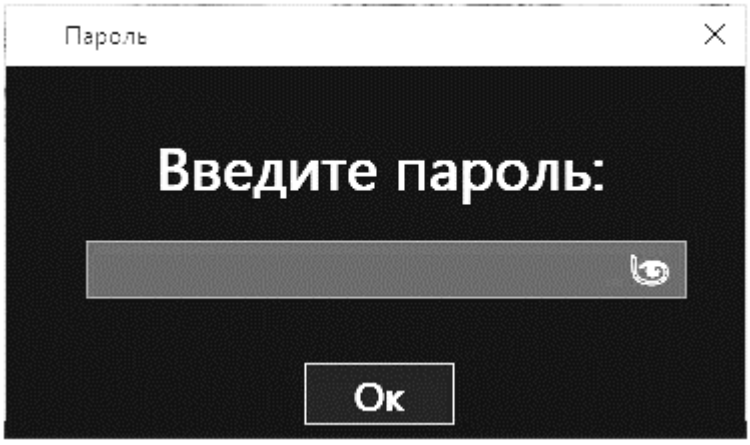

Рисунок 45 - Окно ввода пароля в окне расшифрования

Для того, чтобы ввести пароль необходимо в область ввода пароля ввести с клавиатуры необходимый пароль (Рисунок 46). После этого необходимо

нажать на кнопку подтверждения пароля (Рисунок 47). После этого диалоговое окно закроется.

Рисунок 46 - Область ввода пароля

Рисунок 47 - Кнопка подтверждения пароля

Для того, чтобы посмотреть пароль который вы ввели необходимо нажать на кнопку показа пароля (Рисунок 48), кнопка измениться (Рисунок 49). После этого пароль будет виден, и вы сможете его проверить (Рисунок 50).

Рисунок 48 - Неактивная кнопка показа пароля

Рисунок 49 - Активная кнопка показа пароля

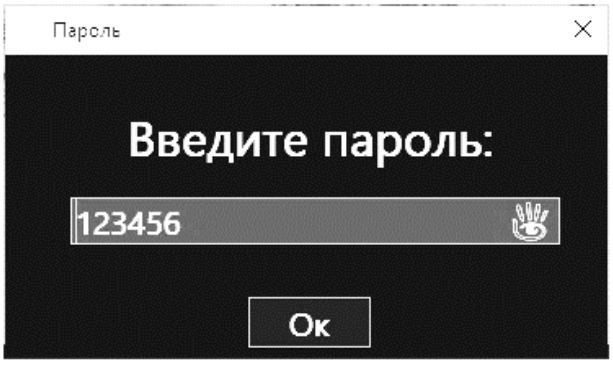

Рисунок 50 - Окно ввода пароля и показом введенного пароля

5 БЕЗОПАСНОСТЬ И ЭКОЛОГИЧНОСТЬ

При работе с программным обеспечение и его сопровождением неотъемлемую часть несет сохранность здоровья пользователя, которая выражается как при работе в помещении, так и при работе с ПЭВМ. Именно поэтому просто необходимо разобрать мероприятия для уменьшения риска трав для пользователей согласно нормативным документам и стандартам (СанПин), а именно позаботиться о освещенности, уровне шума и звука и микроклимате. Так же нужно изучить опасные и вредоносные факторы на рабочем месте, проанализировать помещения для правильной работы с ПЭВМ на рабочем месте.

Безопасность жизнедеятельности (БЖД) – совокупность мероприятий, знаний и навыков человека, обеспечивающих определенный уровень защиты человека, его родных и близких, а также имущества от действий других людей и неблагоприятного воздействия окружающей среды. [21]

Помимо анализа мероприятий, связанных с изучением и безопасности при работе за ПЭВМ, необходимо заранее изучить различные несчастные случаи, аварии, взрывы, пожары и т.п. и разработать мероприятия и требования для уменьшения риска их возникновения. Это позвонит создать более безопасные условия для устойчивой работы с ПЭВМ в работе пользователя.

Изучая различные мероприятия, так же необходимо разобрать различные компоненты содержащие токсичные вещества, для того чтобы обезопасить пользователей и окружающую среду. Для этого нужно разработать комплекс мероприятий для правильной утилизации компонентов.

Абсолютно любой труд связанный с работой за ПЭВМ подразумевает собой статическое положение для пользователей, именно поэтому для поддержания здоровья пользователя необходимо разработать ряд физических упражнений, направленных для поддержания тонуса организма.

5.1 Безопасность

5.1.1 Опасные и вредные факторы на рабочем месте пользователя ПЭВМ

При работе с ПЭВМ необходимо соблюдать требования норм и правил.

Если ссылаться на ГОСТ 12.0.003-2015, то опасными и наносящими вред здоровью факторами при работе с ПЭВМ являются:[22]

- повышенным уровнем общей и локальной вибрации;
- повышенным уровнем и другими неблагоприятными характеристиками шума;
- повышенным уровнем инфразвуковых колебаний (инфразвука);
- повышенным уровнем ультразвуковых колебаний (воздушного и контактного ультразвука);
- повышенным образованием электростатических зарядов;
- наличием электростатического поля, чрезмерно отличающегося от поля Земли;
- наличием постоянного магнитного поля, чрезмерно отличающегося от геомагнитного поля Земли;
- наличием электромагнитных полей промышленных частот;
- наличием электромагнитных полей радиочастотного диапазона
- отсутствие или недостаток необходимого естественного освещения;
- отсутствие или недостатки необходимого искусственного освещения;
- повышенная или пониженная температура воздуха рабочей зоны;
- выделение в воздух рабочей зоны ряда химических веществ;
- повышенная или пониженная влажность воздуха;
- утомляемость глаз;
- монотонность трудового процесса;
- нервно-эмоциональные перегрузки;
- радиоактивное загрязнение поверхностей и материалов производственной среды.

Для предотвращения или снижения действий данных факторов необходимо сформировать ряд мероприятий и требований для пользователя ПЭВМ, направленные на помещение, организации рабочего места, освещение, уровень шума и вибрации, а также разработать ряд рекомендаций для пользователей ПЭВМ.

5.1.2 Организация рабочего места

Рабочее место пользователя – это зона где находится работник и средства его труда, которые определяются на основе технических и эргономических нормативов. Рабочее место представляет собой совокупность факторов окружающей среды, в том числе вредных. Вредный производственный фактор — фактор среды и трудового процесса, воздействие которого на работающего при определенных условиях может вызвать профессиональное заболевание, другое нарушение состояния здоровья, временное или стойкое снижение работоспособности, привести к повреждению здоровья потомства.[23]

В соответствии с требованиями к рабочему месту, оборудованному ПЭВМ, предъявляются следующие требования:[24]

- площадь на одно рабочее место сотрудника, проводящего за компьютером более четырех часов в день, должна составлять не менее 6 кв. м (если у компьютера монитор на базе электронно-лучевой трубки) или 4,5 кв. м (если монитор жидкокристаллический или плазменный);
- высота рабочей поверхности стола для взрослых пользователей должна регулироваться в пределах 680 – 800 мм; при отсутствии такой возможности высота рабочей поверхности должна составлять 725 мм;
- рабочий стол должен иметь пространство для ног высотой не менее 600 мм, шириной – не менее 500 мм, глубиной на уровне колен – не менее 450 мм и на уровне вытянутых ног – не менее 650 мм;
- поверхность сиденья должна иметь ширину и глубину не менее 400 мм, иметь с закруглённый передний край, регулироваться в пределах 400

– 550 мм и углами наклона вперед до 15 градусов и назад до 5 град. угол наклона спинки в вертикальной плоскости должен обеспечивать ±30 градусов;
– стационарные или съемные подлокотники сиденья должны иметь длину не менее 250 мм и ширину 50 – 70 мм, регулироваться над сиденьем в пределах 230±30 мм и внутреннего расстояния между подлокотниками в пределах 350 – 500 мм;
– рабочее место пользователя ПЭВМ должно быть оборудовано подставкой для ног, имеющей ширину не менее 300 мм, глубину не менее 400 мм, регулировку по высоте в пределах 150 мм и по углу наклона опорной поверхности подставки до 20 градусов;
– клавиатура должна располагаться на поверхности стола на расстоянии 100 – 300 мм от края, обращенного к пользователю или на специальной, регулируемой по высоте рабочей поверхности, отделенной от основной столешницы.

На рисунке 51 представлено рекомендуемое размещение пользователя ПЭВМ.

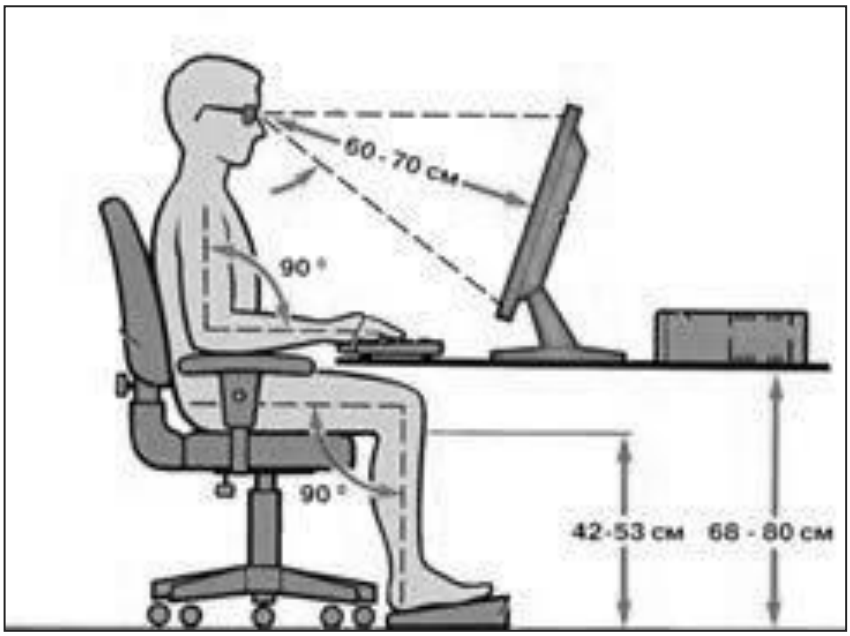

Рисунок 51 – Рекомендуемое размещение пользователя ПЭВМ

5.1.3 Освещение

Освещение является важным требованием, предъявляемым к помещениям с ПЭВМ. Для более производительного труда необходимо организовать правильное освещение по всей области помещения. Это необходимо для того, чтобы при работе с ПЭВМ у пользователя не повышалась нагрузка на глаза. Так же плохое зрение способствует утомляемости и недостаточной концентрации при работе, в худшем случае ослепление.

Разбираясь в освещении можно выделить следующие виды:

- естественное;
- искусственное;
- совмещенное;
- аварийное.

Естественным освещением является освещение от самой земной поверхности за счет прямых лучей солнца, и такое освещение должно присутствовать в любом помещении, где организуется работа пользователя или персонала. Такое освещения может быть либо верхним, либо боковым. Так же может присутствовать комбинированный тип, для лучшего освещения внутри помещения. Если же искусственного освещения недостаточно, то к нему добавляется искусственный.

Системы искусственного освещения обуславливаются способами размещения светильников. По способам размещения светильников в помещениях различают системы общего и комбинированного освещения. [25] Если расстояние между источниками света приблизительно одинаковые, то освещение считают равномерным, если источники света располагаются ближе к производственному оборудованию, то освещение называют локализованным. Местное и общее освещения, применяемые совместно, образуют систему комбинированного освещения. Применяется она в помещениях с точными зрительными работами, требующими высокой освещенности.

Так же при работе с ПЭВМ могут возникнуть различные аварийные ситуации, например, пожар, взрыв, отравление персонала, в таких случаях необходимо организовать, так называемое, аварийное освещение или освещение безопасности.

Освещение безопасности должно создавать на рабочих поверхностях в производственных помещениях и на территориях предприятий, требующих обслуживания при отключении рабочего освещения, наименьшую освещенность величиной 5 % освещенности, нормируемой для рабочего освещения от общего освещения, но не менее 2 лк внутри зданий и не менее 1 лк – для территорий предприятий. [25]

5.1.4 Шум и вибрации

На рабочем месте оператора источниками шума являются технические средства, а также внешний шум. Уровни акустических шумов согласно санитарным нормам СП 51.13330.2011 на рабочих местах при работе аппаратуры должны удовлетворять требованиям закона. Допустимые значения уровней звукового давления в октавных полосах представлены в таблице 4.

Таблица 4 – Допустимые значения уровней звукового давления

Уровни звукового давления в октавных полосах со среднегеометрическими частотами									Уровн извука, дБ
31,5 Гц	63 Гц	125 Гц	250 Гц	500 Гц	1000 Гц	2000 Гц	4000 Гц	8000 Гц	
86 дБ	71 дБ	61 дБ	54 дБ	49 дБ	45 дБ	42 дБ	40 дБ	38 дБ	50 дБ

Уровни виброскорости, виброускорения и вибросмещения в производственных помещениях при работе на ПК не должны превышать следующих значений на частотах 2, 4, 8, 16, 31,5, 63 Гц соответственно 79, 73, 67, 67, 67, 67 дБ для виброскорости, 25, 25, 25, 31, 37, 43 дБ для виброускорения и 133, 121, 109, 103, 97, 91 дБ для вибросмещения. Корректированные значения и их уровни в дБА - 72 дБ.

5.1.5 Микроклимат

Микроклимат производственных помещений – это комплекс физических

факторов, оказывающих влияние на теплообмен человека и определяющих самочувствие, работоспособность, здоровье и производительность труда. Поддержание микроклимата рабочего места в пределах гигиенических норм – важнейшая задача охраны труда. Показатели микроклимата должны обеспечивать сохранение теплового баланса человека с окружающей средой и поддержание оптимального или допустимого теплового состояния организма. [26]

Показатели микроклимата:

- Температура воздуха;
- Относительная влажность воздуха;
- Скорость движения воздуха;
- Мощность теплового излучения.

Помимо внешних факторов, таких как температура отгружающего воздуха, источником повышенной температуры является ПЭВМ. Если температура превышает допустимых показателей, то падает работоспособность пользователей, появляется сонливость и утомляемость. В крайних случаях потеря сознания или более тяжелые случая. Оптимальные микроклиматические условия установлены по критериям оптимального теплового и функционального состояния человека. Они обеспечивают общее и локальное ощущение теплового комфорта в течение 8-часовой рабочей смены при минимальном напряжении механизмов терморегуляции, не вызывают отклонений в состоянии здоровья, создают предпосылки для высокого уровня работоспособности и являются предпочтительными на рабочих местах.[23]

На таблице 5 показаны оптимальные величины показателя микроклимата на рабочих местах производственного помещения.

Таблица 5 – Оптимальные величины показателей микроклимата на рабочих местах производственных помещений

Период года	Категория работ по уровню энергозатрат, Вт	Температура воздуха, °С	Температура поверхностей, °С	Относительная влажность воздуха, %	Скорость движения воздуха, м/с
Холодный	Iа (до 139)	22-24	21-25	60-40	0,1
	Iб (140-174)	21-23	20-24	60-40	0,1
	IIа (175-232)	19-21	18-22	60-40	0,2
	IIб (233-290)	17-19	16-20	60-40	0,2
	III (более 290)	16-18	15-19	60-40	0,3
Теплый	Iа (до 139)	23-25	22-26	60-40	0,1
	Iб (140-174)	22-24	21-25	60-40	0,1
	IIа (175-232)	20-22	19-23	60-40	0,2
	IIб (233-290)	19-21	18-22	60-40	0,2
	III (более 290)	18-20	17-21	60-40	0,3

В помещении должны соблюдаться оптимальные величины температуры воздуха 22-24 градуса, при относительной влажности воздуха 60-40%.

Для поддержания микроклимата в помещении используются системы вентиляции. Сама система вентиляции представляет собой каналы для воздуха, которые постоянно выгоняют воздух и насыщают помещение чистым и свежим воздухом. Если системы вентиляции недостаточно для требуемого уровня микроклимата, то применяется система кондиционирования или отопления.

Для поддержания постоянной температуры, влажности и очистки от вредных веществ используются системы кондиционирования. Данные системы позволяют решить проблему с задержанием углекислого газа в помещении.

5.1.6 Анализ помещения с ПЭВМ

В помещения с ПЭВМ площадь на одно рабочее место пользователей ПЭВМ с ВДТ на базе электроннолучевой трубки (ЭЛТ) должна составлять не

менее 6 м, с ВДТ на базе плоских дискретных экранов (жидкокристаллические, плазменные) - 4,5 м.

При использовании ПВЭМ с ВДТ на базе ЭЛТ (без вспомогательных устройств - принтер, сканер и др.), отвечающих требованиям международных стандартов безопасности компьютеров, с продолжительностью работы менее 4 часов в день допускается минимальная площадь 4,5 м на одно рабочее место пользователя.

Для внутренней отделки интерьера помещений, где расположены ПЭВМ, должны использоваться диффузно-отражающие материалы с коэффициентом отражения для потолка - 0,7-0,8; для стен - 0,5-0,6; для пола - 0,3-0,5.

Помещения, где размещаются рабочие места с ПЭВМ, должны быть оборудованы защитным заземлением (занулением) в соответствии с техническими требованиями по эксплуатации.

Не следует размещать рабочие места с ПЭВМ вблизи силовых кабелей и вводов, высоковольтных трансформаторов, технологического оборудования, создающего помехи в работе ПЭВМ.

Основные характеристики рабочего помещения приведены в таблице 6 [24]

Таблица 6 – Исходные характеристики помещения с ПЭВМ

№ пп	Характеристика помещения	Показатель характеристики
1.	Этаж, на котором расположено помещение	2
2.	Размеры помещения, А*В*Н, м3	6х7х3
3.	Количество работающих в помещении (max), чел.	6
4.	Количество ПЭВМ и другой оргтехники, шт.	7
5.	Площадь на одно рабочее место, м2	7
6.	Объем на одно рабочее место, м3	21
7.	Ориентация оконных проемов (часть света)	север

Продолжение таблицы 6

8.	Окраска помещения: стены, потолок, пол, оборудование	светло-оранжевый белый красно-коричневый белый, светло-серый
9.	Системы обеспечения параметров микроклимата: отопление, вентиляция, кондиционирование	центральное водяное естественная есть
10.	Освещение: - естественное (наличие) - рабочее (тип и количество): - светильники, - лампы; - аварийное (наличие)	есть люминесцентные лампы на потолке (6 светильников по 4 лампы); нет
11.	Наличие штор, занавесей, жалюзи и т.д.	4 жалюзи
12.	Конструктивные параметры рабочего места с ПЭВМ: стула; стола	0,5 140x110x75;

План размещения оборудования в помещении с указанием оконных и дверных проемов представлен рисунке 52.

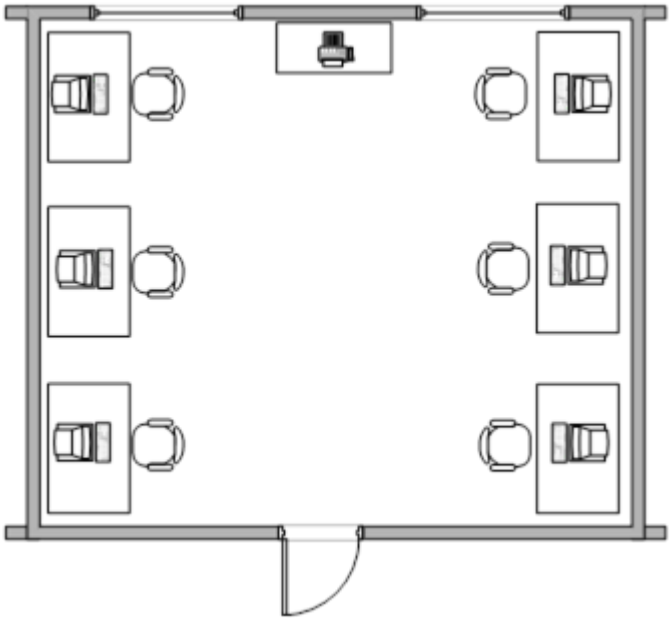

Рисунок 52 – План размещения оборудования

5.2 Экологичность

При работе с ПЭВМ не происходит химических реакций, а значит нет образования газов, но многие компоненты могут содержать ртуть, щелочи, никель и цинк и другие токсичные вещества опасные как для человека, так и для окружающей среды. Именно поэтому необходимо организовать правильную утилизацию данных компонентов, а также провести правильный уход и регулярный надзор за компонентами.

В ряд таких мероприятий входит:

- сортировка металлических и неметаллических частей;
- переплавка отсортированных металлических компонентов, согласно их составу;
- правильную утилизацию неметаллических или не перерабатываемых повторно компонентов.

На сегодняшний день малоотходные технологии в ряде отраслей промышленности находится только на начальном этапе, однако она имеет достаточно большие перспективы развития.

5.3 Чрезвычайные ситуации

5.3.1 Аварийные ситуации

При работе с техникой, такой как ПЭВМ, могут возникнуть различные аварийные ситуации, такие как обрыв проводов питания, повреждение оборудование, возгорание различных компонентов при неправильном подключении и т.п.. Именно, поэтому, необходимо составить рад рекомендаций для пользователей в случае аварийных ситуаций.

При возникновении аварийной ситуации работник обязан:

- в случае обнаружения загорания (пожара) необходимо немедленно сообщить об этом в пожарную охрану по тел. «01», указав адрес, место возникновения пожара, а также фамилию и номер телефона, с которого производится вызов;

- при электротравме немедленно освободить пострадавшего от действия тока и до прибытия врача оказать потерпевшему первую доврачебную помощь;
- при любых случаях сбоя в работе технического оборудования или программного обеспечения немедленно вызвать системного администратора или иного специалиста, отвечающего за этот участок работы;
- при возгорании оборудования отключить питание, вызвать пожарную команду, принять меры к тушению очага пожара при помощи первичных средств пожаротушения (углекислотного или порошкового огнетушителя) и сообщить о происшествии непосредственному руководителю;
- при иных аварийных ситуациях действовать по обстановке, имея в виду, что основная ценность - это люди, принимать все меры для их спасения.

На таблице 7 показаны возможные чрезвычайные ситуации и мероприятия по защите от них. [27]

Таблица 7 – Чрезвычайные ситуации и мероприятия по защите от них

Наименование ЧС	Причины возникновения ЧС	Последствия ЧС	Мероприятия и технические средства по защите от ЧС
Пожар	- неосторожное и халатное обращение с огнем (бросание горящих окурков либо спичек, оставление без присмотра электронагревательных устройств и т.П.); - неверное устройство либо неисправность отопления; - неисправность оборудования и	Травмы, ожоги различной степени тяжести, интоксикация вредными веществами.	обучение работающих противопожарным правилам; проведение бесед, инструктажей и т.п.; правильная эксплуатация техники и оборудования,

Продолжение таблицы 7

	нарушение режима производственного процесса; - неверное устройство и неисправность систем вентиляции; - самовоспламенение и самовозгорание отдельных веществ; - взрывы пыли, газов, паров. КЗ, перегрузки, огромные переходные сопротивления, искрение и электрические дуги, статическое электричество		правильное содержание зданий и территорий; соблюдение противопожарных правил при устройстве отопления, вентиляции; запрещение курения в не установленных местах.
Отключение электроэнергии	аварии на электростанциях, обрывы проводов, аварии на оборудовании поддержки поставки электричества в помещения с ПЭВМ	Потеря не сохраненных данных, остановка работы	установка резервного оборудования для поддержания питания помещений электроэнергией, инспекция состояния оборудования, создание резервных копий данных
Удар молнии	Удар молнии является природным явлением, и предсказать его появление невозможно.	Удар молнии может привести к серьезным травмам, к гибели.	установка на здании с ПЭВМ средств молниезащиты

5.3.2 Меры пожарной безопасности на рабочих местах

При расстановке технологического и другого оборудования должно быть обеспечено наличие проходов к путям эвакуации и эвакуационным выходам.

В Федеральном законе № 69 «О пожарной безопасности» ПЭВМ должен быть установлен на надежную опору (тумбочку, подставку, кронштейн и т. п.), не допускающую его падения. Запрещается устанавливать ПЭВМ:

- в нишах мебельных «стенок», в тумбочках и т.п.;
- ближе 1 метра от электронагревательных приборов и от горючих предметов (тюлей, занавесок, гардин, штор; декоративных украшений, новогодних ёлок и т. п.);
- ближе 0.7 метров от проходов, путей передвижения и эвакуации людей.

Перед началом работы с ПЭВМ необходимо провести ряд действий для уменьшения риска возгорания техники:

- провести внешний осмотр места установки ПЭВМ и убедиться в выполнении требований безопасности, предъявляемых выше;
- провести внешний осмотр ПЭВМ, электрошнура, электровилки и убедиться в их исправности, если корпус, электрошнур, электровилка, задняя крышка повреждены, то ПЭВМ эксплуатировать запрещается;
- при наличии на, над и около ПЭВМ и монитора горючих предметов и ёмкостей с жидкостью – убрать их;
- убедиться, что вентиляционные отверстия на крышке ПЭВМ не закрыты различными твердыми или не пропускающими воздух предметами;
- убедится, что в непосредственной близости с ПЭВМ присутствуют огнетушители или огнеупорные ткани.

5.4 Комплексы физических упражнений для сохранения и укрепления индивидуального здоровья и обеспечения полноценной профессиональной деятельности

Основные нагрузки при работе с ПЭВМ падают на зрение и опорно-двигательный аппарат, что может послужить ухудшением здоровья работника

или пользователя ПЭВМ. Для поддержания здоровья организма необходимо проводить комплексы физических упражнений, направленны непосредственно сказанные выше проблемные точки, а также, для поддержания сохранности и укрепления здоровья, выполнять профилактические упражнения. Для этого необходимо составить ряд рекомендаций для обеспечения полноценной профессиональной деятельности при работе за ПЭВМ. Например, необходимо делать небольшие перерывы после одного или двух часов работы. В это время необходимо вставать рабочего места и провести комплекс физических упражнений. Не допускается отдых, связанный с продолжением нахождением на рабочем месте и просмотре на экране монитора различных, по мнение пользователя, расслабляющих и дающих отдых организму.

В начале рекомендуется выполнить ряд самостоятельных заданий, таких как:

- утренняя гигиеническая гимнастика;
- лечебная гимнастика (гимнастика для глаз);
- занятия физкультурой по избранной программе;
- физкультурная пауза во время работы;
- элементы самомассажа;
- закаливание организма.

Данный список может быть расширен или видоизменен в соответствии с индивидуальными качествами работника или пользователя, например, страдающих близорукостью или имеющие травмы позвоночника.

В ряд физических упражнений может входить:

- **Повороты головы вперед-назад влево-вправо**. Необходимо сесть ровно и выполнить движение головы сначала вперед-назад затем влево и вправо. Выполнять это упражнение 4-5 раз.
- **Вращение плечевого сустава.** Упражнение может быть выполнено стоя или сидя на кресле. Необходимо начать движение плечевого сустава обоих рук сначала по часовой стрелке 10 раз, затем против часовой стрелки 10 раз. Выполнять это упражнение 4 раза.

- **Наклоны корпуса.** Необходимо, стоя, вытянуть ладони вверх и начать наклона корпуса на 30-45 градусов в левую строну, задержаться на 2-3 секунды и вернуться в исходное положение. Повторить упражнение в правую строку. Выполнять упражнение 5-7 раз.
- **Повороты поясничного отдела.** Исходное положение стоя и руку на груди. Совершить поворот поясничного отдела в левую сторону так, чтобы выпадающее плече было почти параллельно стопам. Задержаться в этом положении на 1-2 секунды и вернуться в исходное положение. Повторить упражнение в правую строку. Выполнять упражнение 5-7 раз.
- **Разминка для глаз.** Каждое упражнение необходимо выполнить 6 раз через каждые 1,5-2 часа работы. Сначала идет движение глаз влево-вправо, затем вверх-вниз. После этого движение глаз по диагонали. Затем необходимо глазами нарисовать горизонтальные и вертикальные спирали. После необходимо крепко зажмурится на 2-3 секунды и открыть глаза. Далее нужно нарисовать глазами вертикальные и горизонтальные восьмерки. После необходимо выполнить сведения глаз к носу (можно использовать палец для упрощения процесса). В конце нужно переключить взять на самый дальний объект, которые только можно увидеть.

Так же стоит выделить ряд профилактических рекомендаций, направленных на личную проверку состояния здоровья:

- **Стараться двигаться как можно больше**. Если нет возможности провести полноценную разминку, можно просто пройтись по коридору, дать отдохнуть своим глазам и дать небольшое напряжения на органы, которые во время работы за ПЭВМ получают недостаточную нагрузку;
- **Постоянно следить за осанкой**. Работая за ПЭВМ сгорбившись можно только больше устать, а в худшем случая получить травму позвоночника или просто искривить его;
- **Проверять освещенность рабочего места**. Правильная освещённость

уменьшит напряжение на глаза, а небольшая разминка для глаз уменьшит усталость.

- **Стараться контролировать вес.** Постоянная работа за ПЭВМ не позволяет сжигать необходимое количество калорий из-за этого могут возникнуть различные опасные, для состояния здоровья, проблемы. Именно поэтому необходимо следить за весом и индексом массы тела.

ЗАКЛЮЧЕНИЕ

Целью выпускной квалификационной работы являлась разработка программной модели скрытой передачи данный для защиты информации средствами C#.

При выполнении выпускной квалификационной работы был проведен анализ предметной области, в котором были разобраны вопросы шифрования, криптографии стеганографии, проанализирован язык программирования C#, дополнительные библиотеки для работы, среды разработки, а также изучение существующие ПО и выявлены их достоинства и недостатки.

Было проведено проектирование ПО, в котором созданы физическая и логическая структуры, в Visio 2019, описаны основные алгоритмы работы, выполнена структура технического обеспечения.

Таким образом, было разработано ПО для скрытой передачи данных с использование LSB-метода с учетом его возможных уязвимостей на языке программирования C#. Так же, для удобства пользователей, было создано руководство пользователя, в котором описаны все манипуляции над продуктами, его требования и структура. Был создан файл установки, который автоматически устанавливает необходимые файлы для работы ПО.

Разработанный продукт позволяет пользователям скрыть важную информацию и передать ее адресату без сомнения в том, что данные будут разоблачены злоумышленником.

При выполнении выпускной квалификационной работы были разобраны и выписаны рекомендации по безопасности и экологичности, а, так же, составлен комплекс физических упражнений.

В итоге, можно сделать вывод, что для защиты информации по открытому каналу связи соединения криптографических и стеганографических методов не просто возможно, но и позволяет качественней сохранить важную информацию.

БИБЛИОГРАФИЧЕСКИЕ ССЫЛКИ

1 Что такое шифрование данных? [Электронный ресурс]. URL: https://lan-star.ru/poleznye-stati/32-chto-takoe-shifrovanie-dannyh.html

2 Постановление Правительства РФ от 16.04.2012 N 313 (ред. от 28.12.2021) "Об утверждении Положения о лицензировании деятельности по разработке, производству, распространению шифровальных (средств, информационных систем и телекоммуникационных систем, защищенных с использованием шифровальных средств, выполнению работ, оказанию услуг в области шифрования информации, техническому обслуживанию шифровальных средств, информационных систем и телекоммуникационных систем, защищенных с использованием шифровальных (криптографических) средств" : [Электронный ресурс]. Доступ из справ.-правовой системы «КонсультантПлюс»

3 Приказ ФСБ России от 9 февраля 2005 г. N 66 «Об утверждении положения о разработке, производстве, реализации и эксплуатации шифровальных (криптографических) средств защиты информации (положение пкз-2005)» : [Электронный ресурс]. Доступ из справ.-правовой системы «КонсультантПлюс»

4 Указ Президента РФ от 3 апреля 1995 N 334 «О мерах по соблюдению законности в области разработки, производства, реализации и эксплуатации шифровальных средств, а также предоставления услуг в области шифрования информации» : [Электронный ресурс]. Доступ из справ.-правовой системы «КонсультантПлюс»

5 Черёмушкин А. В. Криптографические протоколы: основные свойства и уязвимости : научная статья / А. В. Черёмушкин. – Москва : Институт криптографии, связи и информатики, 2009. – 36 с.

6 Дэвид Кан Взломщики кодов. — М.: Центрполиграф, 2000. — 473 с

7 Стеганография, цифровые водяные знаки и стеганоанализ / А. В. Аграновский [и др.]. — М.: Вузовская книга, 2009. — 220 с.

8 Урбанович, П. П. Защита информации методами криптографии, стеганографии и обфускации : учеб.-метод. пособие / П. П. Урбанович. – Минск : БГТУ, 2016. – 220 с.

9 В.Г. Грибунин, И.Н. Оков, И.В. Туринцев. Цифровая стеганография / В.Г. Грибунин, И.Н. Оков, И.В. Туринцев - М.: СОЛОН-ПРЕСС, 2009 - 272 с.

10 Конахович Г. Ф., Пузыренко А. Ю. Компьютерная стеганография. Теория и практика. - К.: "МК-Пресс", 2006.

11 T. Filler, J. Fridrich, Gibbs Construction in Steganography, IEEE Transactions on Information Forensics and Security, December 2010

12 V. Holub, J. Fridrich, Designing Steganographic Distortion Using Directional Filters, IEEE Workshop on Information Forensic and Security, Tenerife, Canary Islands, December 2–5, 2012

13 V. Holub, J. Fridrich, T. Denemark, Universal Distortion Function for Steganography in an Arbitrary Domain, EURASIP Journal on Information Security, (Section:SI: Revised Selected Papers of ACM IH and MMS 2013), 2014

14 Язык C# и платформа .NET [Электронный ресурс]. URL: https://metanit.com/sharp/tutorial/1.1.php

15 Шарп, Д. Microsoft Visual C#. Подробное руководство / Д. Шарп. – СПб.: «Питер», 2017. – 848 с.

16 iText [Электронный ресурс]. URL: https://itextpdf.com/ru

17 OpenXML [Электронный ресурс]. URL: https://www.nuget.org/packages/DocumentFormat.OpenXml/

18 Введение в интегрированную среду разработки Visual Studio | C# [Электронный ресурс]. URL: https://docs.microsoft.com/ru-ru/visualstudio/get-started/csharp/visual-studio-ide?view=vs-2022

19 Rider [Электронный ресурс]. URL: https://www.jetbrains.com/ru-ru/rider/

20 Руководство по классическим приложениям [Электронный ресурс]. URL:https://docs.microsoft.com/ru-ru/dotnet/desktop/winforms/overview/?view=netdesktop-6.0

21 Основы безопасности жизнедеятельности [Электронный ресурс]. URL: https://resh.edu.ru/subject/lesson/5829/train/104190/

22 ГОСТ 12.0.003-2015 Издания. Система стандартов безопасности труда. Опасные и вредные производственные факторы ; введ. 2017-03-01. – МКС : Межгосударственный совет по стандартизации, метрологии и сертификации ; М. : Изд-во стандартов, 2015.

23 Тарасенко Н.Ю., Волкова З.А. Профессиональные вредности // Большая медицинская энциклопедия : в 30 т. / гл. ред. Б.В. Петровский. — 3 изд. — Москва : Советская энциклопедия, 1983. — Т. 21. Преднизон - Растворимость. — 560 с.

24 Куренкова Г. В. Гигиенические особенности условий труда и здоровье профессиональных пользователей персональных компьютеров и видеодисплейных терминалов / Г. В. Куренкова // Сиб. мед. журн. - 2004. - N 6. - С. 14-17

25 Виды и системы освещения [Электронный ресурс]. URL: http://electricalschool.info/main/lighting/1939-vidy-i-sistemy-osveshhenija.html

26 Федеральный закон от 30.03.1999 N 52-ФЗ "О санитарно-эпидемиологическом благополучии населения"

27 СанПиН 51.13330.2011 Защита от шума.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1 В.Г. Грибунин, И.Н. Оков, И.В. Туринцев. Цифровая стеганография / В.Г. Грибунин, И.Н. Оков, И.В. Туринцев - М.: СОЛОН-ПРЕСС, 2009 - 272 с.

2 Введение в интегрированную среду разработки Visual Studio | C# [Электронный ресурс]. URL: https://docs.microsoft.com/ru-ru/visualstudio/get-started/csharp/visual-studio-ide?view=vs-2022

3 Виды и системы освещения [Электронный ресурс]. URL: http://electricalschool.info/main/lighting/1939-vidy-i-sistemy-osveshhenija.html

4 ГОСТ 12.0.003-2015 Издания. Система стандартов безопасности труда. Опасные и вредные производственные факторы ; введ. 2017-03-01. – МКС : Межгосударственный совет по стандартизации, метрологии и сертификации ; М. : Изд-во стандартов, 2015.

5 Дэвид Кан Взломщики кодов. — М.: Центрполиграф, 2000. — 473 с

6 Конахович Г. Ф., Пузыренко А. Ю. Компьютерная стеганография. Теория и практика. - К.: "МК-Пресс", 2006.

7 Криптографический протокол [Электронный ресурс]. URL: http://ru.wikipedia.org/wiki/Криптографический_протокол

8 Куренкова Г. В. Гигиенические особенности условий труда и здоровье профессиональных пользователей персональных компьютеров и видеодисплейных терминалов / Г. В. Куренкова // Сиб. мед. журн. - 2004. - N 6. - С. 14-17

9 Основы безопасности жизнедеятельности [Электронный ресурс]. URL: https://resh.edu.ru/subject/lesson/5829/train/104190/

10 Постановление Правительства РФ от 16.04.2012 N 313 (ред. от 28.12.2021) "Об утверждении Положения о лицензировании деятельности по разработке, производству, распространению шифровальных (средств, информационных систем и телекоммуникационных систем, защищенных с использованием шифровальных средств, выполнению работ, оказанию услуг в области шифрования информации, техническому обслуживанию

шифровальных средств, информационных систем и телекоммуникационных систем, защищенных с использованием шифровальных (криптографических) средств" : [Электронный ресурс]. Доступ из справ.-правовой системы «КонсультантПлюс»

11 Приказ ФСБ России от 9 февраля 2005 г. N 66 «Об утверждении положения о разработке, производстве, реализации и эксплуатации шифровальных (криптографических) средств защиты информации (положение пкз-2005)» : [Электронный ресурс]. Доступ из справ.-правовой системы «КонсультантПлюс»

12 Руководство по классическим приложениям [Электронный ресурс]. URL:https://docs.microsoft.com/ru-ru/dotnet/desktop/winforms/overview/?view=netdesktop-6.0

13 СанПиН 51.13330.2011 Защита от шума.

14 Стеганография, цифровые водяные знаки и стеганоанализ / А. В. Аграновский [и др.]. — М.: Вузовская книга, 2009. — 220 с.

15 Тарасенко Н.Ю., Волкова З.А. Профессиональные вредности // Большая медицинская энциклопедия : в 30 т. / гл. ред. Б.В. Петровский. — 3 изд. — Москва : Советская энциклопедия, 1983. — Т. 21. Преднизон - Растворимость. — 560 с.

16 Тидвелл, Д. Разработка пользовательских интерфейсов/ Д. Тидвелл. – СПб.: «Питер», 2011. – 480 с.

17 Указ Президента РФ от 3 апреля 1995 N 334 «О мерах по соблюдению законности в области разработки, производства, реализации и эксплуатации шифровальных средств, а также предоставления услуг в области шифрования информации» : [Электронный ресурс]. Доступ из справ.-правовой системы «КонсультантПлюс»

18 Урбанович, П. П. Защита информации методами криптографии, стеганографии и обфускации : учеб.-метод. пособие / П. П. Урбанович. – Минск : БГТУ, 2016. – 220 с.

19 Федеральный закон от 30.03.1999 N 52-ФЗ "О санитарно-эпидемиологическом благополучии населения"

20 Черёмушкин А. В. Криптографические протоколы: основные свойства и уязвимости : научная статья / А. В. Черёмушкин. – Москва : Институт криптографии, связи и информатики, 2009. – 36 с.

21 Что такое шифрование данных? [Электронный ресурс]. URL: https://lan-star.ru/poleznye-stati/32-chto-takoe-shifrovanie-dannyh.html

22 Шарп, Д. Microsoft Visual C#. Подробное руководство / Д. Шарп. – СПб.: «Питер», 2017. – 848 с.

23 Язык C# и платформа .NET [Электронный ресурс]. URL: https://metanit.com/sharp/tutorial/1.1.php

24 Язык программирования C#: краткая история, возможности и перспективы [Электронный ресурс]. URL: https://timeweb.com/ru/community/articles/chto-takoe-csharp

25 Cancelli G., Cox I.J., Doerr G. Improved LSB matching steganalysis based on the amplitude of local extrema // IEEE International conference on Image Processing, October 2008

26 Fridrich J., Du R., Meng L. Steganalysis of LSB Encoding in Color Images, ICME 2000, New York City, July 31—August 2, New York.

27 habr.com [Электронный ресурс]. URL: https://habr.com/ru/post/422593/

28 iText [Электронный ресурс]. URL: https://itextpdf.com/ru

29 J. Friedrich, G. Miroslav, R. Du. Reliable Detection of LSB Steganography in Color and Grayscale Images. Binghampton, New York: SUNY, 2001.

30 OpenXML [Электронный ресурс]. URL: https://www.nuget.org/packages/DocumentFormat.OpenXml/

31 Rider [Электронный ресурс]. URL: https://www.jetbrains.com/ru-ru/rider/

32 sautinsoft.com [Электронный ресурс]. URL: https://sautinsoft.com/products/document/help/net/

33 securelist.ru [Электронный ресурс]. URL: https://securelist.ru/steganography-in-contemporary-cyberattacks/79090/

34 spy-soft.net [Электронный ресурс]. URL: https://spy-soft.net/cifrovaya-steganografiya-sposoby-realizacii/

35 T. Filler, J. Fridrich, Gibbs Construction in Steganography, IEEE Transactions on Information Forensics and Security, December 2010

36 V. Holub, J. Fridrich, Designing Steganographic Distortion Using Directional Filters, IEEE Workshop on Information Forensic and Security, Tenerife, Canary Islands, December 2–5, 2012

37 V. Holub, J. Fridrich, T. Denemark, Universal Distortion Function for Steganography in an Arbitrary Domain, EURASIP Journal on Information Security, (Section:SI: Revised Selected Papers of ACM IH and MMS 2013), 2014

ПРИЛОЖЕНИЕ А

1 Введение

1.1 Наименование программы

Наименование программы – «CriptoSteg».

1.2 Краткая характеристика области применения

Скрытая, аннотация документов Медицинские снимки, картография;

Скрытая связь Военные и разведывательные приложения, а также применение в случаях, когда криптографию использовать нельзя или её недостаточно;

Защита от копирования Электронная коммерция, контроль за копированием, распространение мультимедийной информации (видео по запросу).

2 Основания для разработки

Основанием для разработки является Договор. Договор утвержден Заказчиком, и Климентьевым Денисом Викторовичем, именуемым в дальнейшем исполнителем.

Согласно Договору, Исполнитель обязан разработать и установить «CriptoSteg» на оборудовании Заказчика не позднее 12.07.2022, предоставить исходные коды и документацию к разработанной системе не позднее 09.07.2022.

Условное обозначение темы разработки (шифр темы) – «Criptosteg».

3 Назначение разработки

Программа будет использоваться в офисе всеми членами кадров.

3.1 Функциональное назначение

Данной программное обеспечение может применяться в различных видах деятельности где необходима скрытая передача данных по открытому каналу

3.2 Эксплуатационное назначение

Программа должна эксплуатироваться на персональных компьютерах организации.

4 Требования к программе или программному изделию

4.1 Требования к функциональным характеристикам

4.1.1 Требования к составу выполняемых функций

После запуска программы пользователю отображается форма шифрования данных.

Для пользователя программа предоставляет следующие возможности:

1) Стеганографически скрывать данные в изображении с возможностью выбирать количество перезаписывающихся бит;

2) Криптографически шифровать данные;

3) Парольная защита изображения;

4) Изъятие данных из изображения;

Примерный вид окна должен выглядеть. В левой стороне должен быть организован ввод данных и кнопка загрузки текста в форматах TXT, DOC, DOCX, PDF. В правой стороне должен быть организована кнопка загрузки изображения в форматах BMP, PNG, JEPG, JPG и окно просмотра изображения.

Так же должна быть парольная защита, организованная в другом окне с использованием скрытого ввода данных, кнопка сохранения и кнопка переключения формы. Изображение должно быть сохранено в форматах BMP, PNG, JEPG, JPG.

При переключении формы на форму расшифрования должна быть кнопка загрузки изображения в форматах BMP, PNG, JEPG, JPG и окно просмотра изображения.

В нижней части должны присутствовать кнопки расшифровки и сохранения текста. Сохранение текста должно быть в форматах TXT, DOC, DOCX, PDF.

При расшифровании должно появляться окно ввода пароля с использованием скрытого ввода данных. После правильного ввода пароля на место окна просмотра должно появляться окно просмотра и редактирования

текста.

Шифрование данных должно быть заранее зашифровано выбранным алгоритмом.

4.1.2 Требования к организации входных и выходных данных

Входными и выходными данными являются:

1) Текст

– Записанные с клавиатуры;

– Загруженные из файлов формата TXT;

– Загруженные из файлов формата PDF;

– Загруженные из файлов формата DOCX.

2) Изображение

– Загруженные из файлов формата PNG;

– Загруженные из файлов формата JPEG;

– Загруженные из файлов формата JPG;

– Загруженные из файлов формата BMP.

4.1.3 Требования к временным характеристикам

После загрузки текста, он должен быть загружен не позднее чем за $n*10^{-5}$ секунд, где n – количество символов в файле.

После загрузки изображения, он должен быть загружен не позднее чем за 5 секунд.

После нажатии на клавишу зашифровать или расшифровать данные должны быть скрыты не позднее чем за 7 секунд.

4.2 Требования к надежности

Вероятность безотказной работы системы должна составлять не менее 99.99% при условии исправности техники.

4.2.1 **Требования к обеспечению надежного (устойчивого) функционирования программы**

В связи с тем, что в файлах может храниться важная информации

необходимо выполнение программы без использования интернет соединения.

Надежное (устойчивое) функционирование программы должно быть обеспечено выполнением заказчиком совокупности организационно-технических мероприятий, перечень которых приведен ниже:

организацией бесперебойного питания технических средств;

использованием лицензионного программного обеспечения;

регулярным выполнением рекомендаций Министерства труда и социального развития РФ, изложенных в Постановлении от 23 июля 1998 г. «Об утверждении межотраслевых типовых норм времени на работы по сервисному обслуживанию ПЭВМ и оргтехники и сопровождению программных средств»;

регулярным выполнением требований ГОСТ 51188-98. Защита информации. Испытания программных средств на наличие компьютерных вирусов.

4.2.2 Время восстановления после отказа

Время восстановления после отказа, вызванного сбоем электропитания технических средств (иными внешними факторами), не фатальным сбоем (не крахом) операционной системы, не должно превышать 10 минут при условии соблюдения условий эксплуатации технических и программных средств.

Время восстановления после отказа, вызванного неисправностью технических средств, фатальным сбоем (крахом) операционной системы, не должно превышать времени, требуемого на устранение неисправностей технических средств и переустановки программных средств.

4.2.3 Отказы из-за некорректных действий оператора

Отказы программы возможны вследствие некорректных действий оператора (пользователя) при взаимодействии с операционной системой.

4.3 Условия эксплуатации

Программа (клиент) запускается на компьютере работника офиса.

Окно программы должно быть открыто не на весь экран, должна быть возможность закрыть, свернуть приложение или запустить любое стороннее программное обеспечение.

4.3.1 Климатические условия эксплуатации

Специальные условия не требуются.

4.3.2 Требования к видам обслуживания

Программа не требует проведения каких-либо видов обслуживания.

4.3.3 Требования к численности и квалификации персонала

В процессе эксплуатации с программой работают сотрудники офиса. Сотрудник должен изучить руководство пользования.

4.4 Требования к составу и параметрам технических средств

Состав технических средств:

Компьютер сотрудника, включающий в себя:

процессор x86 с тактовой частотой, не менее 1 ГГц;

оперативную память объемом, не менее 1 Гб;

видеокарту, монитор, мышь, клавиатура.

4.5 Требования к информационной и программной совместимости

Должно быть исключено появление посторонних устройств в сети.

4.6 Требование к маркировке и упаковке

Программное изделие передается в руки директора в виде диска. Специальных требований к маркировке не предъявляется. Для проверки подлинности программного обеспечения рекомендуется проверять контрольные суммы загруженных файлов со значениями, указанными на диске.

4.7 Требования к транспортированию и хранению

Специальных требований не предъявляется.

4.8 Специальные требования

Специальных требований не предъявляется.

5 Требования к программной документации

Предварительный состав программной документации:

техническое задание (включает описание применения);

программа и методика испытаний;

руководство пользователя;

ведомость эксплуатационных документов.

6 Технико-экономические показатели

«CriptoSteg» пригодна для офиса, не рассматривающих возможность распространения информации в сети Internet.

Функциональность программы совпадает с аналогами .

7 Стадии и этапы разработки

Разработка должна быть проведена в три стадии:

техническое задание;

технический (и рабочий) проекты;

внедрение.

На стадии «Техническое задание» должен быть выполнен этап разработки, согласования и утверждения настоящего технического задания.

На стадии «Технический (и рабочий) проект» должны быть выполнены перечисленные ниже этапы работ:

разработка программы;

разработка программной документации;

испытания программы.

На стадии «Внедрение» должен быть выполнен этап разработки «Подготовка и передача программы».

Содержание работ по этапам:

На этапе разработки технического задания должны быть выполнены перечисленные ниже работы:

постановка задачи;

определение и уточнение требований к техническим средствам;

определение требований к программе;

определение стадий, этапов и сроков разработки программы и документации на нее;

согласование и утверждение технического задания.

На этапе разработки программы должна быть выполнена работа по программированию (кодированию) и отладке программы.

На этапе разработки программной документации должна быть выполнена разработка программных документов в соответствии с требованиями ГОСТ 19.101-77.

На этапе испытаний программы должны быть выполнены перечисленные ниже виды работ:

разработка, согласование и утверждение порядка и методики испытаний;

проведение приемо-сдаточных испытаний;

корректировка программы и программной документации по результатам испытаний.

На этапе подготовки и передачи программы должна быть выполнена работа по подготовке и передаче программы и программной документации в эксплуатацию на объектах заказчика.

8 Порядок контроля и приемки

Приемосдаточные испытания программы должны проводиться согласно разработанной исполнителем и согласованной заказчиком «Программы и методики испытаний».

Ход проведения приемо-сдаточных испытаний заказчик и исполнитель документируют в протоколе испытаний.

ПРИЛОЖЕНИЕ Б

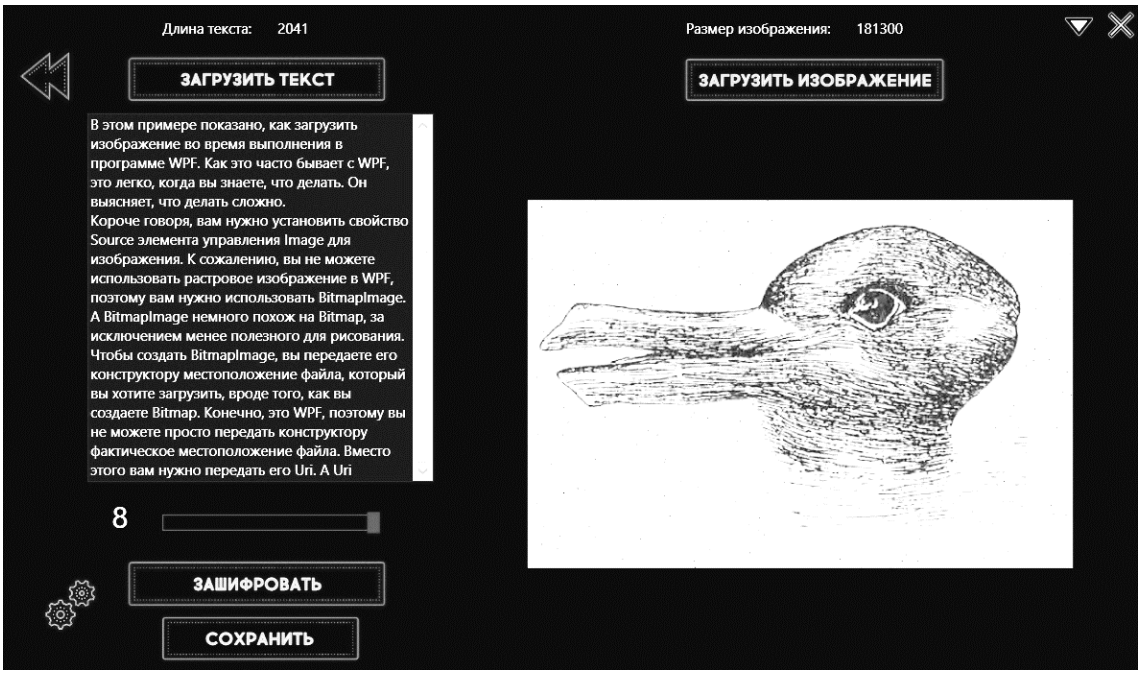

Рисунок Б.1 – Шифрование текста в изображении используя 8 бит каждого цвета

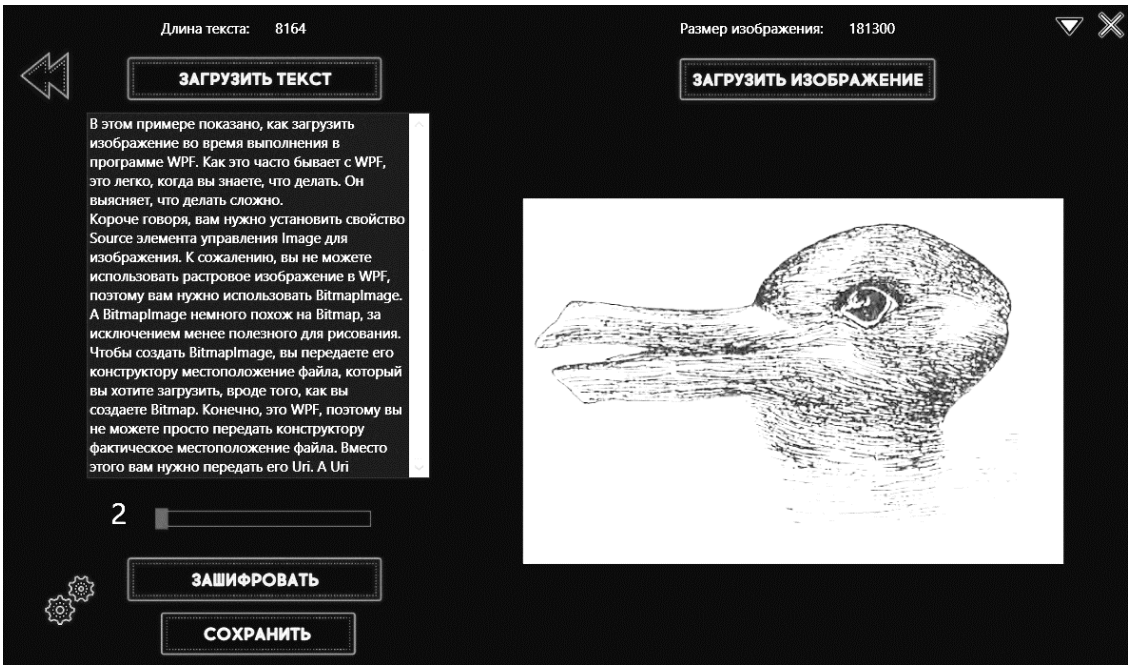

Рисунок Б.2 – Шифрование текста в изображении используя 2 бита каждого цвета

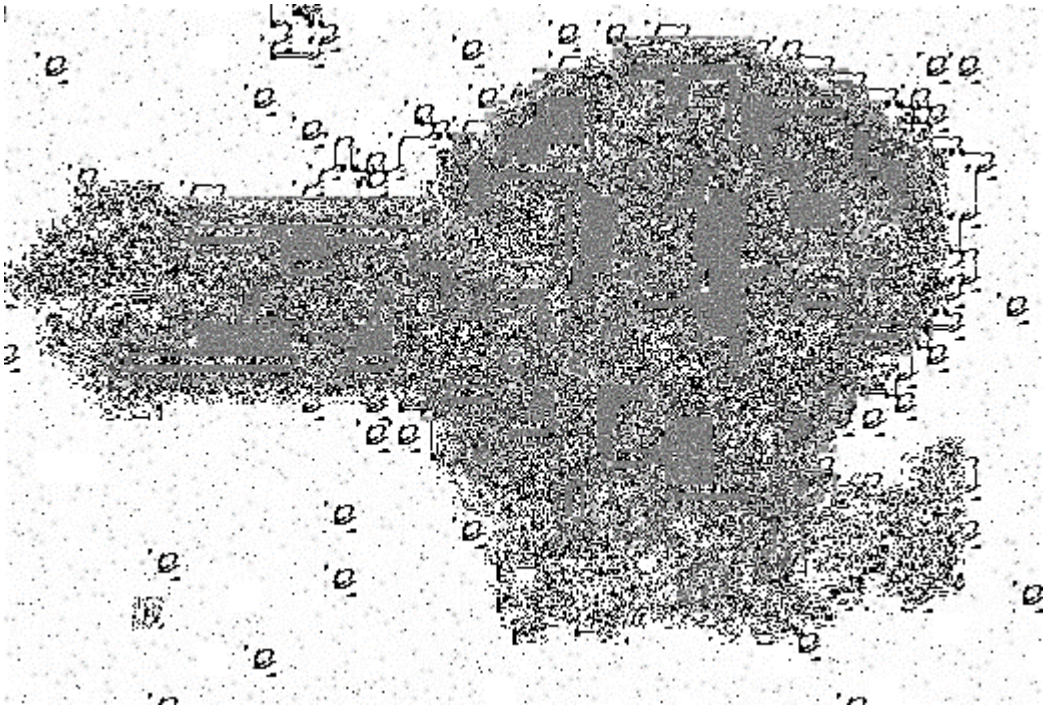

Рисунок Б.3 – Пример спектра НЗБ для зашифрованного контейнера в 8 бит

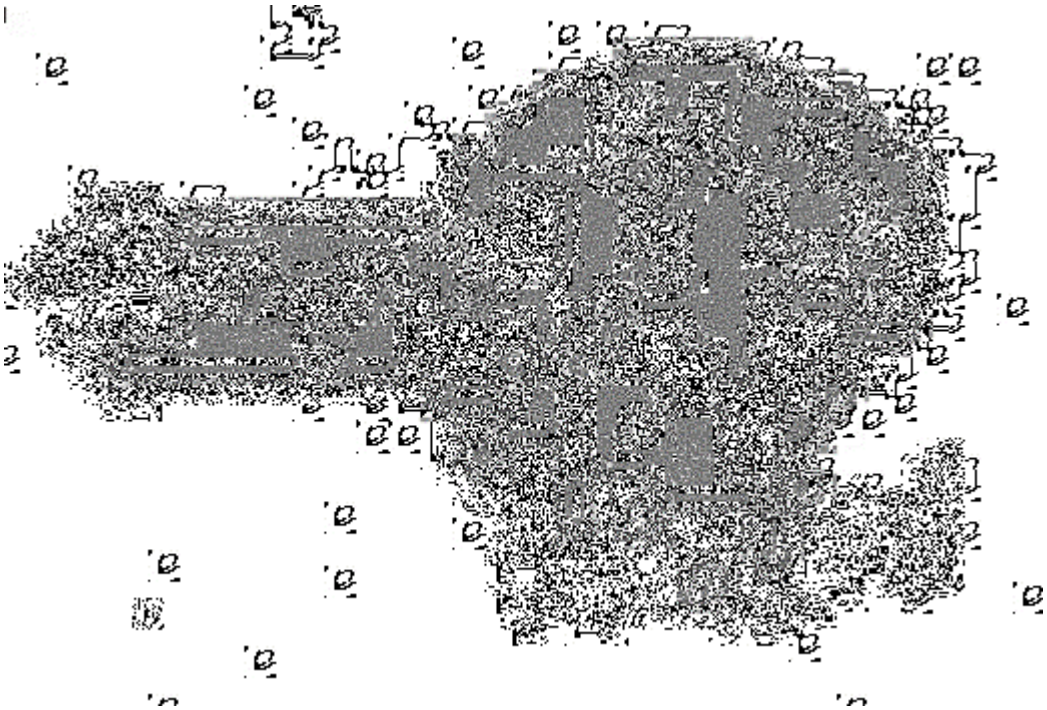

Рисунок Б.3 – Пример спектра НЗБ для зашифрованного контейнера в 2 бита

Продолжение Приложения Б

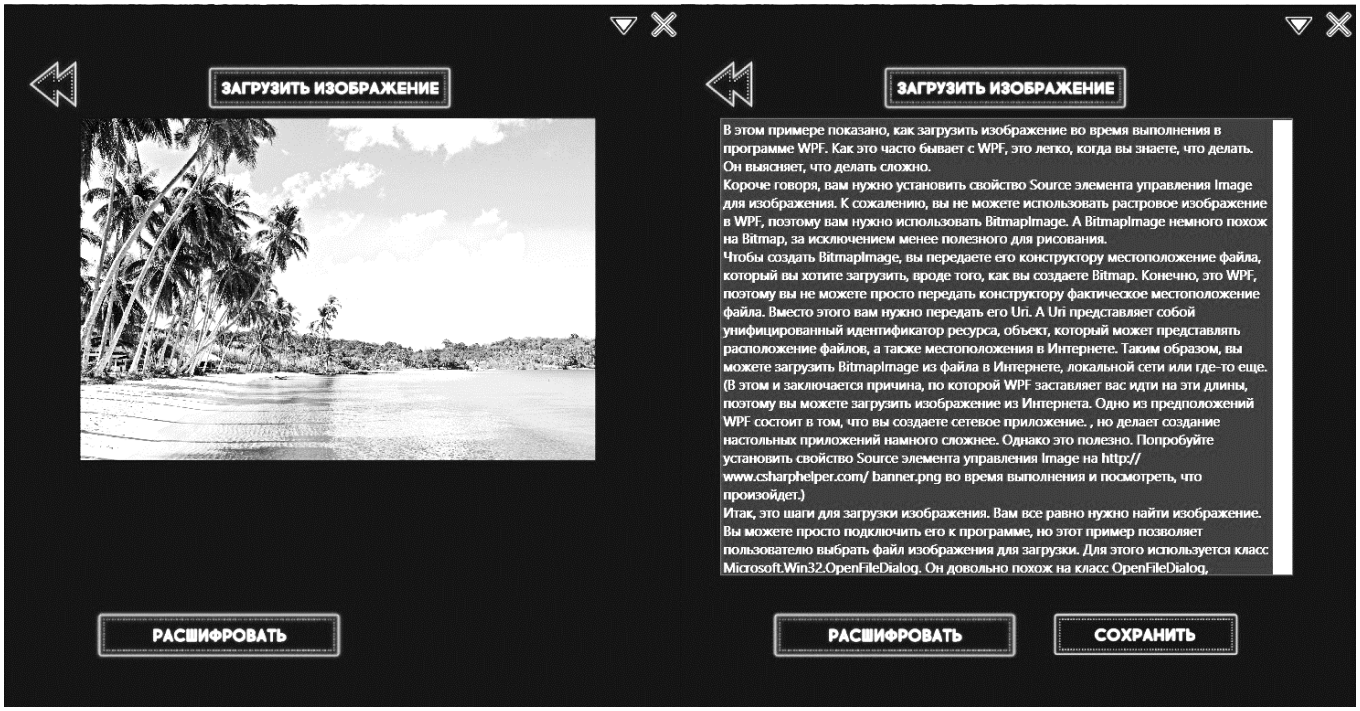

Рисунок Б.4 – Пример окна расшифрования

ПРИЛОЖЕНИЕ В

Рисунок В.1 – Пример шифрования текста с использования алгоритма AES

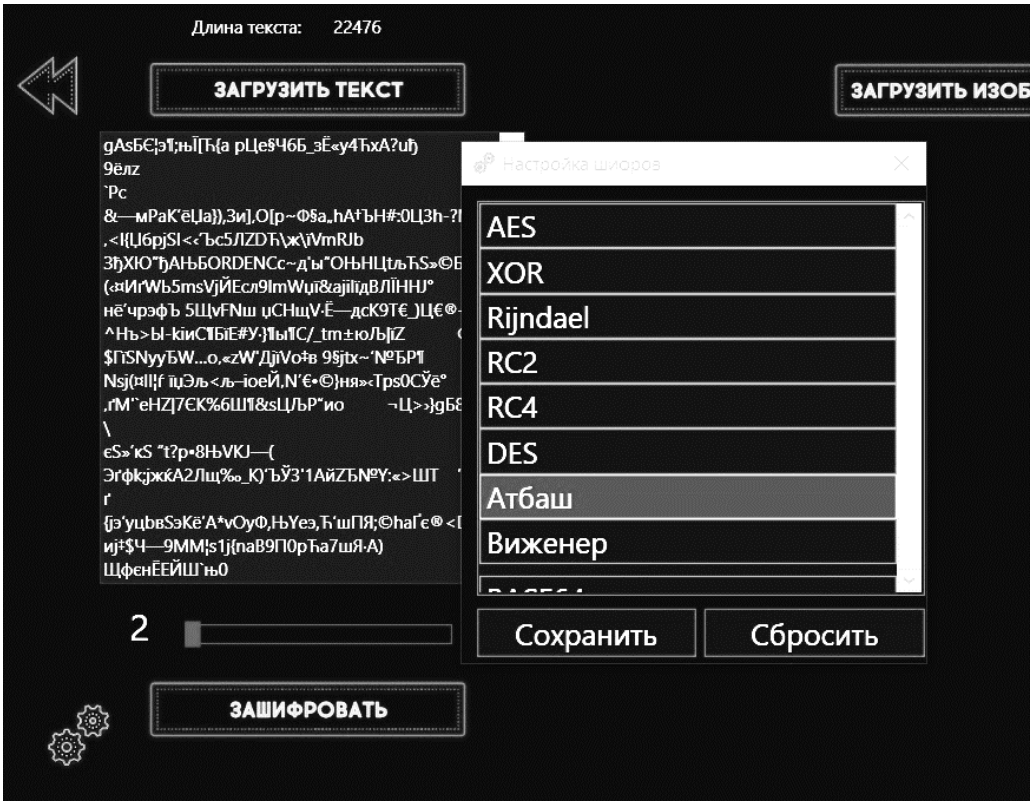

Рисунок В.1 – Пример шифрования текста с использования алгоритма Атбаш

ПРИЛОЖЕНИЕ Г

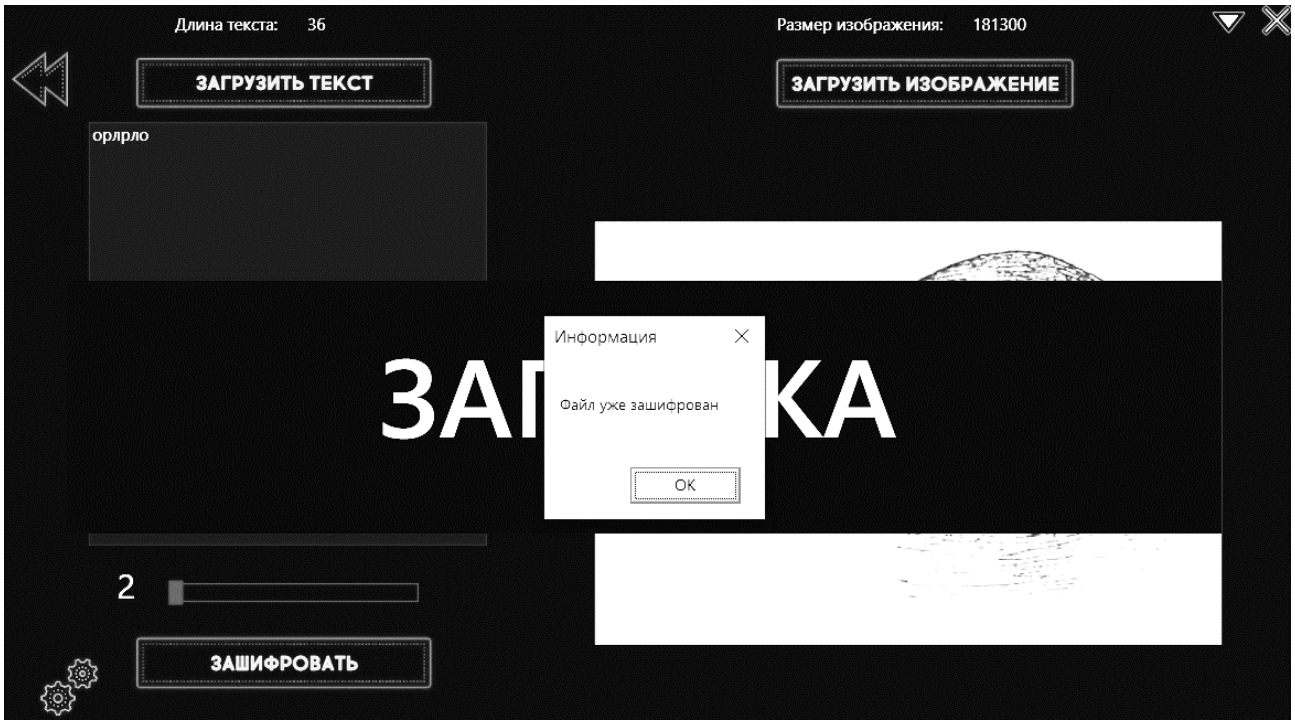

Рисунок Г.1 – Ошибка проверки знака шифрования

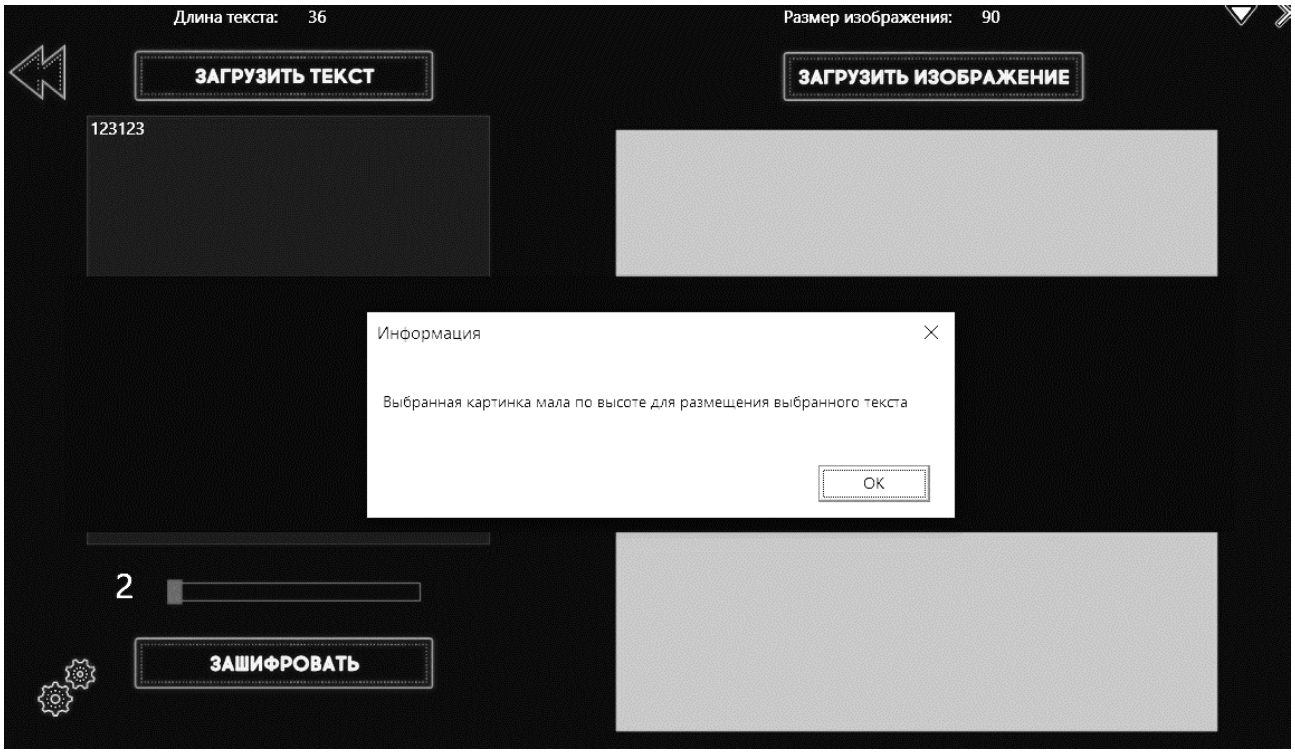

Рисунок Г.2 – Ошибка высоты изображения

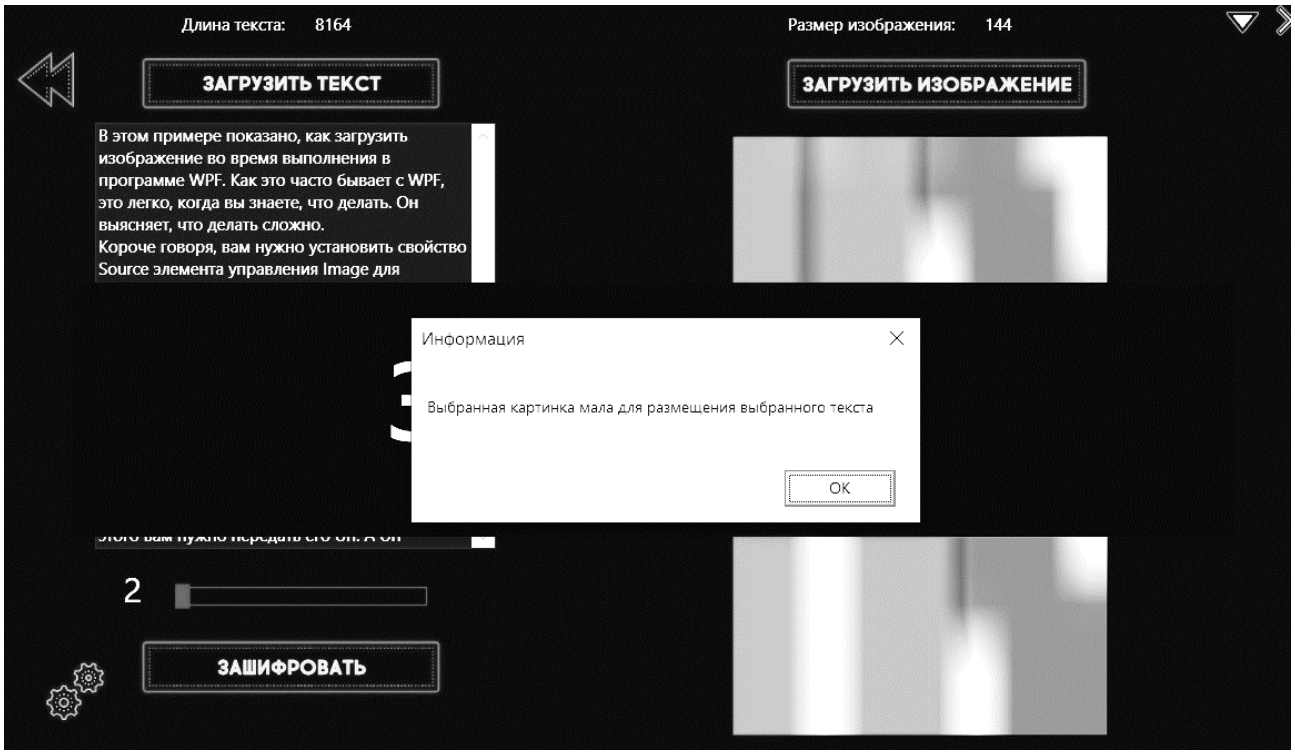

Рисунок Г.3 – Ошибка размера изображения или текста

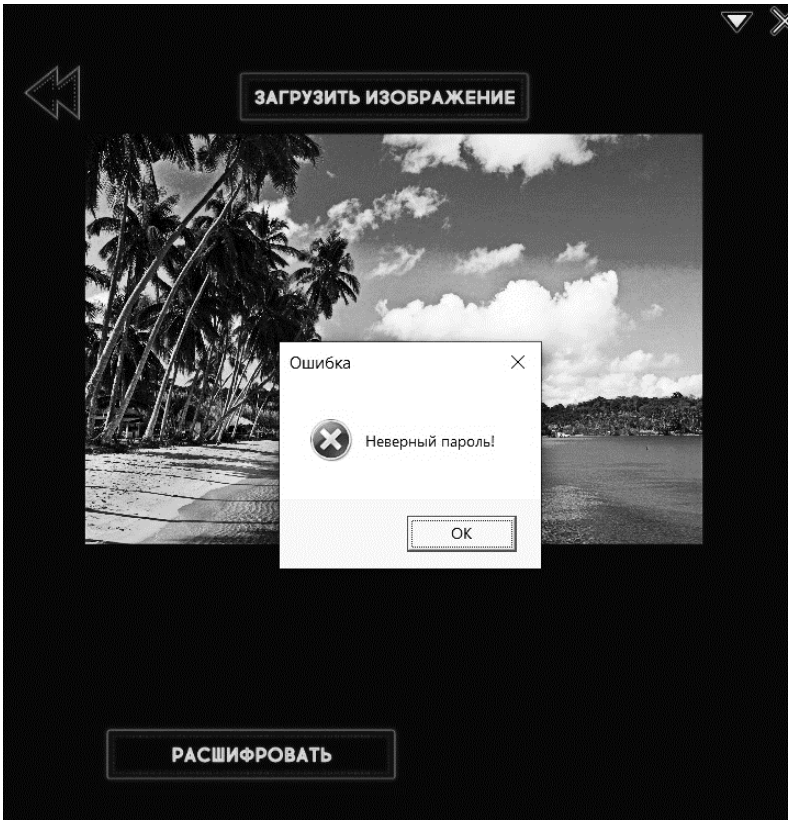

Рисунок Г.4 – Ошибка неверного ввода пароля