Министерство образования Республики Беларусь

Учреждение образования «Белорусский государственный технологический университет»

Факультет информационных технологий

Кафедра информационных систем и технологий

**Отчет к лабораторной работе №15**:

Исследование методов текстовой стеганографии

Выполнила:

студентка 3 курса 6 группы

Слемнёв В.Н.

Преподаватель:

Сазонова Д.В.

**2023 г.**

1. **Теоретические сведения**

Было отмечено, что к текстовой стеганографии относятся методы, предусматривающие использование в качестве контейнера файла-документа текстового типа.

Многообразие методов текстовой стеганографии подразделяется на синтаксические методы, которые не затрагивают семантику текстового сообщения, и лингвистические, которые основаны на эквивалентной трансформации текстовых файлов-контейнеров, сохраняющей смысловое содержание текста, его семантику.

Для понимания сущности некоторых из методов полезно познакомиться с важнейшими особенностями и параметрами использования стилей (в т. ч. – пространственно-геометрическими параметрами шрифтов), на основе которых строится текстовый файл-контейнер. На рис. 11.1 показаны основные из параметров шрифта.



Рисунок 1.1 – Параметры шрифта

К синтаксическим методам компьютерной стеганографии, которые характеризуются сравнительно невысокой эффективностью (с точки зрения объема осаждаемой информации) относятся следующие (такие методы мы отнесем к числу базовых синтаксических методов):

* изменение расстояния между строками электронного текста (Line-Shift Coding); называется методом изменения межстрочных интервалов; сущность заключается в том, что используется текст с различными межстрочными расстояниями: выделяется максимальное и минимальное расстояния между строками, позволяющее кодировать соответственно символы «1» и «0» осаждаемого сообщения;
* изменение расстояния между словами в одной строке электронного текста (Word-Shift Coding); суть метода состоит в том, что осаждение информации основано на модификации расстояния между словами текста-контейнера;
* изменение количества пробелов между словами (частный случай метода Word-Shift Coding); Основан та том, что, например, чередование одинарного пробела и двойного (хх­\_хх\_\_хх) кодирует «1», переход же с двойного пробела на одинарный кодирует «0» (хх\_\_хх\_хх);
* на основе внесения специфических изменений в шрифты, т. е. начертания отдельных букв (Feature Coding); заключается в изменении написания отдельных букв используемого стандартного шрифта: визуально заметны различные образы, соответствующие буквам с верхними (например, l, t, d) или нижними (например, a, g) выносными элементами (см. рис. 13.1); например, букву «А» можно модифицировать, незначительно укорачивая длинную нижнюю часть буквы (см. рис. 1.2);

а) пустой контейнер; б) заполненный контейнер

Рисунок 1.2 - Пример применения метода Feature Coding

* изменение 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); предполагает размещение тайной информации на установленных позициях слов или в определенных словах текста-контейнера, который, как правило, лишен логического смысла (как видно, действительно лепет);
* увеличение длины строки; предусматривает искусственное увеличение длины каждой строки за счет пробелов: например, нет пробела (определяется положением знака перехода на новую строку) – «0», один пробел – «1» (рис. 13.3);
* использование регистра букв; для обозначения бита секретного сообщения, представленного единицей, используется символ нижнего регистра, а нулем — верхнего (или наоборот);



Рисунок 1.3 - Пример реализации метода увеличения длины строки

* использование невидимых символов; знак пробел кодируется символом с кодом 32, но в тексте его можно заменить также символом, имеющим код 255 (или 0), который является «невидимым» и отображается как пробел.

Рассмотренные базовые методы могут применяться независимо и совместно, сохраняют исходный смысл текста, а обеспечиваемые ими показатели плотности кодирования при совмещении складываются.

Еще одна важная особенность. Перечисленные методы работают успешно до тех пор, пока тексты представлены в коде ASCII.

Методы также легко применяются к любому тексту, независимо от его содержания, назначения и языка. Синтаксические системы стеганографии легко реализуются в программном коде, так как они полностью автоматические и не требуют вмешательства оператора. Однако синтаксические методы неустойчивы к форматированию текста (вспомним робастность систем на основе ЦВЗ), и поэтому информация может быть потеряна при простом применении иного стиля форматирования текста-контейнера, скрывающего в себе стегосообщение. К тому же с помощью синтаксических методов можно передать незначительное количество информации.

Существуют также стеганографические методы, которые интерпретируют текст как двоичное изображение. Необходимо отметить, что данные методы нечувствительны к изменению масштаба документа, что обеспечивает им хорошую устойчивость к большинству искажений, которые могут иметь место при активных атаках.

К числу основных лингвистических методов относятся:

* метод синонимов;
* метод переменной длины слова; основан на том, что длина слов в сообщении зависит от содержания секретного сообщения и способа кодирования слов: обычно одно слово текста-контейнера определенной длины кодирует два бита информации из стеганосообщения; например, слова текста длиной в 4 и 8 символов могут означать комбинацию бит «00», длиной в 5 и 9 — «01», 6 и 10 — «10», 7 и 11 букв — «11»; слова короче 4 и длиннее 11 букв можно вставлять где угодно для лексической и грамматической связки слов в предложении – программное приложение, которое декодирует принятое сообщение (извлекает сообщение из стеганоконтейнера), будет просто игнорировать их;
* метод первой буквы – программа-помощник в этом методе накладывает ограничение уже не на длину слова, а на первую (можно на вторую) букву; обычно одну и ту же комбинацию могут кодировать несколько букв, например, комбинацию «101» означают слова, начинающиеся с «А», «Г» или «Т;
* мимикрия; мимикрия генерирует осмысленный текст, используя синтаксис, описанный в Context Free Grammar (CFG), и встраивает информацию, выбирая из CFG определенные фразы и слова; грамматика CFG – это один из способов описания языка, который состоит из статических слов и фраз языка, а также узлов.

1. **Практическая часть**

В данной лабораторной работе было необходимо разработать авторское приложение, реализующее один из методов текстовой стеганографии на основе модификации пространственно-геометрических параметров текста-контейнера. Были реализованы методы текстовой стенографии путем модификации числа пробелов и модификации цвета текста.

Для работы с текстовыми файлами использовалась библиотека Aspose.Words, позволяющее обрабатывать и форматировать документы Microsoft Word при помощи языка C#.

Сперва необходимо реализовать текстовую стенографию путем модификации числа пробелов в тексте. Цикл, реализующий это, представлен на рисунке 2.1

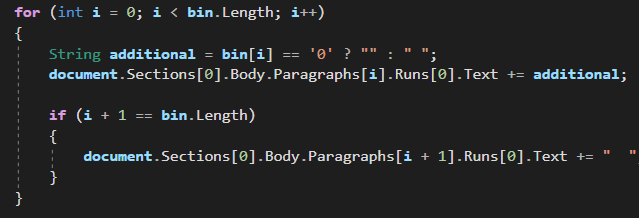
****

Рисунок 2.1 – Модификация пробелов

В цикле бит сообщения представляется следующим образом: 0 – нет пробела, 1 – есть пробел. Пробелы расставляются в конец абзаца. Таким образом, максимальное количество символов, которые возможно зашифровать равно количеству абзацев в исходном документе Word. Работа приложения представлена на рисунке 2.2.

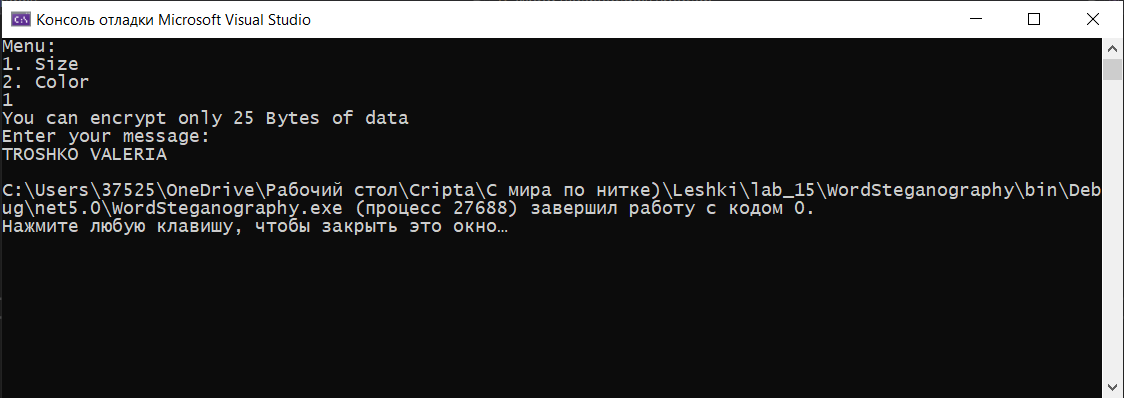
****

Рисунок 2.2 – Запись секретного сообщения в документ

Для расшифрования сообщения был написан цикл, представленный на рисунке 2.3.

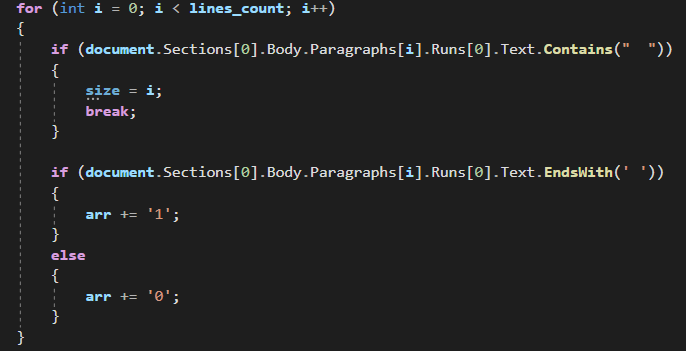
****

Рисунок 2.3 – Расшифрование секретного сообщения

Цикл перебирает все абзацы в исходном документе Word, проверяя окончания каждого абзаца. Если абзац оканчивается пробелом – в расшифрованное сообщение в бинарном виде записывается 1, если нет, то 0. Концом цикла является либо окончание всех абзацев в документе, либо окончание одного из абзацев двойным пробелом, что говорит о конце тайного сообщения. Расшифрование сообщения представлено на рисунке 2.4.

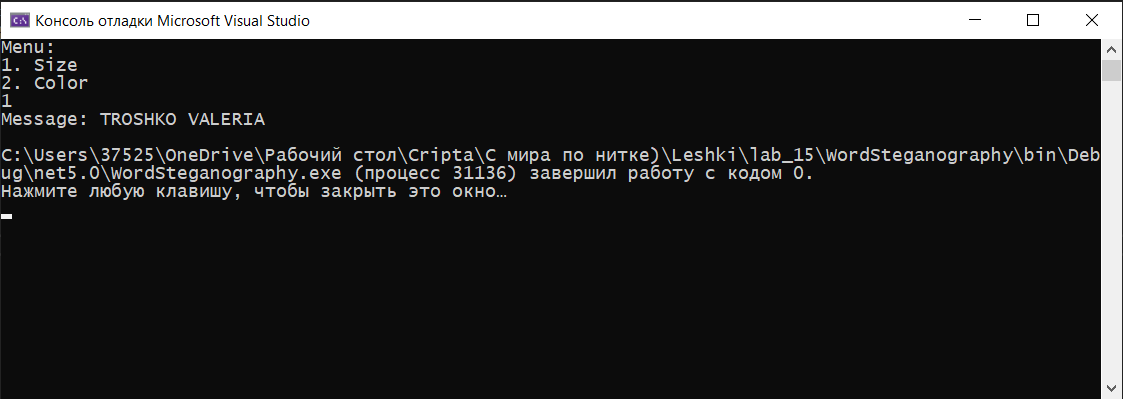
****

Рисунок 2.4 – Расшифрование секретного сообщения в приложении

Затем было необходимо реализовать текстовую стенографию путем модификации цвета символов. Цикл, реализующий это, представлен на рисунке 2.5.

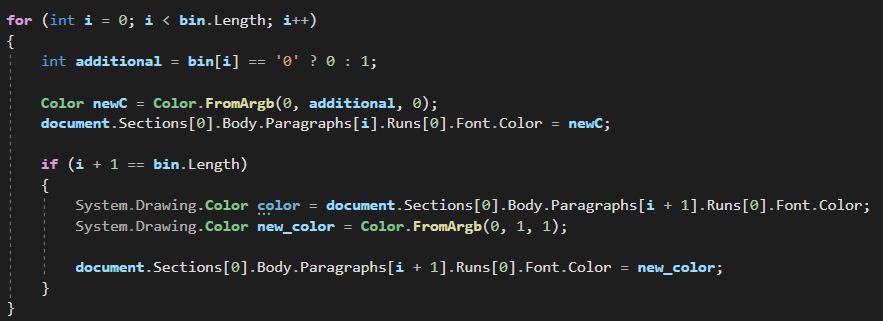
****

Рисунок 2.5 – Модификация пробелов

Засекречивание сообщения происходит при помощи канала зеленого цвета в системе RGB. Если бит исходного сообщения 1 – значит цвет одной строки текста будет равен (0, 1, 0). В случае нулевого бита в сообщении цвет текста будет черным (0, 0, 0). Окончание сообщения осуществляется при помощи синего канала, в этом случае цвет строки будет (0, 1, 1). Реализация в приложении представлена на рисунке 2.6.

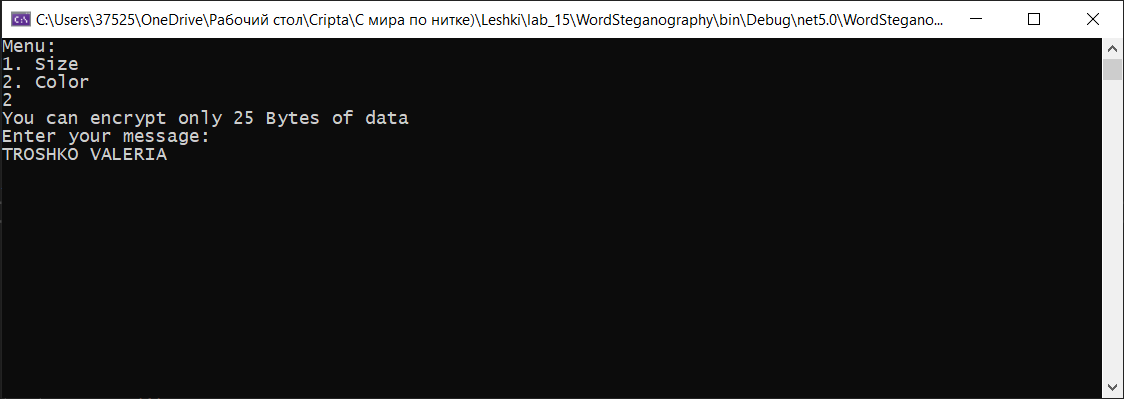
****

Рисунок 2.6 – Запись секретного сообщения в документ

Для расшифрования сообщения был написан цикл, представленный на рисунке 2.7.

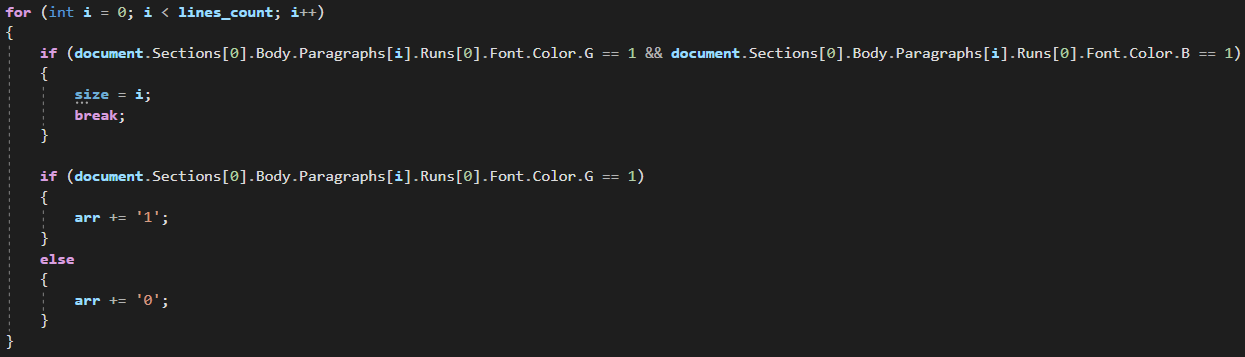
****

Рисунок 2.7 – Расшифрование секретного сообщения

Цикл перебирает все строки текстового документа и смотрит их цвета. Если цвет (0, 0, 0) – значит в битовую последовательность искомого сообщения записывается 0, если цвет (0, 1, 0) – записывается 1, если же цвет (0, 1, 1), то это говорит о конце сообщения.

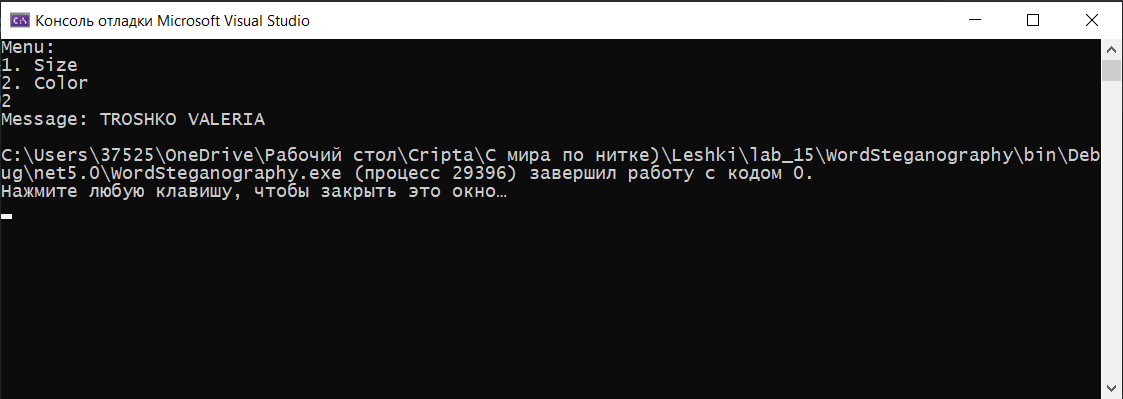
****

Рисунок 2.8 – Расшифрование секретного сообщения в приложении

**Вывод:** в данной лабораторной работе я закрепила теоретические знания из области текстовой стеганографии, классификации, моделирования стеганосистем подобного вида и сущности основных методов, изучила основные алгоритмы осаждения/извлечения тайной информации на основе методов текстовой стеганографии, а также познакомилась с методиками оценки стеганографической стойкости методов.