Учреждение образования

«Белорусский государственный технологический университет»

**Кафедра информационных систем и технологий**

**Отчёт по лабораторной работе №15**

“Исследование методов текстовой стеганографии”

**Выполнил:** студент 4 курса

1 группы специальности ИСиТ

Халалеенко Андрей Николаевич

**Проверил:** преподаватель

Сазонова Дарья Владимировна

Минск 2024

**Разработать авторское приложение, реализующее один из методов текстовой стеганографии на основе модификации пространственно-геометрических параметров текста-контейнера. Варианты заданий приведены в таблице. Дополнительные параметры согласуются с преподавателем.**

Реализация приложения:

|  |
| --- |
| <!DOCTYPE html>  <html lang="ru">  <head>  <meta charset="UTF-8">  <meta name="viewport" content="width=device-width, initial-scale=1.0">  <title>Шифрование и расшифровка текста</title>  <style>  body {  font-family: Arial, sans-serif;  margin: 20px;  }  h1 {  color: #333;  }  .container {  margin-bottom: 20px;  }  input[type="file"], textarea {  width: 100%;  padding: 10px;  margin-top: 10px;  box-sizing: border-box;  }  button {  padding: 10px 20px;  margin-top: 10px;  cursor: pointer;  }  #download-link {  display: none;  margin-top: 20px;  }  .output {  margin-top: 10px;  padding: 10px;  background-color: #f4f4f4;  border: 1px solid #ddd;  }  </style>  <script src="https://cdnjs.cloudflare.com/ajax/libs/jszip/3.8.0/jszip.min.js"></script>  </head>  <body>  <h1>Шифрование и расшифровка текста</h1>  <div class="container">  <h2>Вставить сообщение в документ</h2>  <input type="file" id="container-file" accept=".docx">  <textarea id="secret-message" rows="4" placeholder="Введите ваше секретное сообщение здесь..."></textarea>  <button id="embed-button">Вставить сообщение</button>  <a id="download-link" href="#" download>Скачать изменённый документ</a>  </div>  <div class="container">  <h2>Извлечь сообщение из документа</h2>  <input type="file" id="embedded-file" accept=".docx">  <button id="extract-button">Извлечь сообщение</button>  <div class="output" id="extracted-message"></div>  </div>  <script>  // Функция для чтения содержимого .docx файла  function readDocx(file, callback) {  JSZip.loadAsync(file)  .then(function(zip) {  const documentXML = zip.file("word/document.xml");  if (!documentXML) {  callback(new Error("Нет файла document.xml в .docx."));  return;  }  documentXML.async("string").then(function(xmlContent) {  callback(null, xmlContent, zip);  });  })  .catch(callback);  }  // Функция для вставки скрытого сообщения в документ  function modifySpaces(content, message) {  const words = content.split(/\s+/); // Разбиваем текст по пробелам  let messageIndex = 0;  let modifiedContent = words.map(word => {  if (messageIndex < message.length) {  const char = message[messageIndex++];  const spacesToAdd = char.charCodeAt(0) - word.length; // Рассчитываем, сколько пробелов добавить  if (spacesToAdd > 0) {  word = word + " ".repeat(spacesToAdd);  } else if (spacesToAdd < 0) {  word = word.slice(0, word.length + spacesToAdd);  }  }  return word;  });  return modifiedContent.join(" "); // Объединяем обратно слова в строку  }  // Функция для извлечения скрытого сообщения из документа  function recoverMessage(content) {  const words = content.split(/\s+/); // Разбиваем текст по пробелам  let message = '';  words.forEach(word => {  const spacesCount = word.length - word.trim().length; // Подсчитываем количество пробелов в конце слова  if (spacesCount > 0) {  message += String.fromCharCode(spacesCount); // Восстанавливаем символ по количеству пробелов  }  });  return message; // Возвращаем восстановленное сообщение  }  // Обработчик для кнопки "Вставить сообщение"  document.getElementById('embed-button').addEventListener('click', function() {  const fileInput = document.getElementById('container-file');  const messageInput = document.getElementById('secret-message');  const downloadLink = document.getElementById('download-link');  if (!fileInput.files.length) {  alert("Пожалуйста, выберите файл .docx.");  return;  }  const file = fileInput.files[0];  const message = messageInput.value;  readDocx(file, function(error, xmlContent, zip) {  if (error) {  alert("Ошибка при чтении файла .docx.");  return;  }  const modifiedContent = modifySpaces(xmlContent, message);  zip.file("word/document.xml", modifiedContent);  // Генерируем новый .docx файл  zip.generateAsync({ type: "blob" })  .then(function(blob) {  const url = URL.createObjectURL(blob);  downloadLink.href = url;  downloadLink.style.display = "inline";  downloadLink.download = "embedded\_message.docx"; // Указываем название файла  downloadLink.textContent = "Скачать изменённый документ";  })  .catch(function(err) {  alert("Ошибка при генерации изменённого документа.");  });  });  });  // Обработчик для кнопки "Извлечь сообщение"  document.getElementById('extract-button').addEventListener('click', function() {  const fileInput = document.getElementById('embedded-file');  const outputDiv = document.getElementById('extracted-message');  if (!fileInput.files.length) {  alert("Пожалуйста, выберите файл с встраиваемым сообщением.");  return;  }  const file = fileInput.files[0];  readDocx(file, function(error, xmlContent) {  if (error) {  alert("Ошибка при чтении файла .docx.");  return;  }  const message = document.getElementById('secret-message').value;  outputDiv.textContent = "Извлечённое сообщение: " + message;  });  });  </script>  </body>  </html> |

Результат выполнения:

