УДК 681.3.06

А. А. Сущеня, Е. А. Блинова
Белорусский государственный технологический университет

МАТЕМАТИЧЕСКАЯ МОДЕЛЬ СТЕГАНОГРАФИЧЕСКОЙ СИСТЕМЫ С ИСПОЛЬЗОВАНИЕМ СТЕГАНОГРАФИЧЕСКОГО КОНТЕЙНЕРА В ВИДЕ ЭЛЕКТРОННОЙ КНИГИ ФОРМАТА EPUB

Описывается модель стеганографической системы, основанная на использовании предлагаемого метода, в котором в качестве контейнера применяется файл-архив электронного формата EPUB. При помощи указанного метода предлагается скрывать тайное сообщение в файлах различного формата, что обеспечит повышение стеганографической стойкости системы. Новизна рассматриваемого метода заключается в размещении тайного сообщения, а также его контрольной суммы в трех видах контейнеров: XHTML-файлах, в которых находится текстовое содержание электронной книги, JPG-изображении, которое представляет собой обложку электронного издания, и CSS-файлах, в которых описаны правила визуального отображения книги. Излагаются возможности использования стеганографического метода в электронных изданиях для осаждения цифровых водяных знаков с целью защиты документов-контейнеров от несанкционированного копирования и распространения. Представлено разработанное программное средство «EPUB Modifier», демонстрирующее работу описанной стеганографической модели. Показан пользовательский интерфейс приложения, технология разработки, а также основные структурные элементы его архитектуры. Для грамотного построения архитектуры был использован паттерн проектирования «Chain of responsibility». Программное средство работает с электронным документом формата EPUB, изменяя его основные семантические части для создания стеганографического контейнера.

Ключевые слова: стеганография, формат EPUB, XHTML, CSS, LSB.

A. A. Sushchenia, E. A. Blinova
Belarusian State Technological University

MATHEMATICAL DESCRIPTION OF A STEGANOGRAPHIC SYSTEM FOR EMBEDDING INFORMATION IN THE EPUB FORMAT CONTAINER

The article describes a model of the steganographic system based on the proposed method, in which a file-archive of an EPUB electronic format is used as a container. Using this method, it is proposed to hide the secret message in files of different formats, which increases the steganographic stability of the system. The novelty of the proposed method is to place the secret message, as well as its checksum in three types of containers: XHTML-files, which contain the text content of the e-book, JPG-image, which is the cover of the electronic edition and CSS-files, which describe the rules of visual display of the book. The possibilities of using the steganographic method in electronic publications for the deposition of digital watermarks in order to protect container documents from unauthorized copying and distribution are explored. The developed software “EPUB Modifier” demonstrating the operation of the described steganographic model is presented. The user interface of the application, the development technology, as well as the main structural elements of its architecture are described. The “Chain of responsibility” design pattern was used for competent architecture construction. The software works with an electronic document of the EPUB format, changing its basic semantic parts to create a steganographic container.

Key words: steganography, format EPUB, XHTML, CSS, LSB.

Введение. Цифровой контент приобретает все большую популярность. Электронные книги не являются исключением. Сегодня намного быстрее и удобнее можно приобрести электронный вариант издания через Интернет, при этом не выходя из дома. Однако наряду с удобством также возрастает риск несанкционированного распространения и копирования цифровых экземпляров книг. Проблема защиты информации в процессе передачи от одного абонента к другому является актуальной для человечества на протяжении длительного периода времени. На сегодняшний день найдено большое количество способов, которые позволяют тайно осуществить процесс обмена информацией для обеих сторон. Способы скрытной передачи информации изучает стеганография [1].

Общей чертой таких способов является то, что скрываемое сообщение встраивается в не привлекающий внимание объект, который открыто отправляется адресату.

Секретность системы защиты передаваемых сообщений должна содержаться в ключе – фрагменте информации, предварительно разделенном между адресатами. Ключом выступает алгоритм внедрения информации в объект. Объект, в который внедрена информация, называется стеганографическим контейнером [2].

В качестве контейнера может выступать электронная книга одного из самых популярных форматов на сегодняшний день – EPUB, который по своей сути представляет собой набор XHTML-, JPG- и CSS-файлов [3].

Предлагаемая модель стеганографической системы основана на использовании метода, комбинирующего три разных типа контейнеров для улучшения стеганографической стойкости.

Основная часть. Объектом исследования в данной работе являются стеганографические методы защиты прав интеллектуальной собственности на электронные книги. Предметом – модели стеганографических процессов.

Предлагаемая модель строится на основе следующих обозначений и положений:

пусть **M** – это конечное множество сообщений, которые могут быть тайно размещены в контейнере: $\mathbf{M} = \{M_1, M_2, \ldots, M_n\}$. В предлагаемом методе **M** подразделяется на M_O – само внедряемое сообщение и M_H – контрольная сумма M_O, вычисленная на основе алгоритма MD5;

C – это конечное множество всех допустимых контейнеров (файлов-контейнеров или документов-контейнеров): $\mathbf{C} = \{C_1, C_2, \ldots, C_p\}$, причем $p > n$;

K – множество всех ключей, под которыми в общем случае понимаются методы или алгоритмы осаждения сообщения в контейнер или иные операции по предварительному преобразованию осаждаемого сообщения либо выбору элементов контейнера для такого осаждения: $\mathbf{K} = \{K_1, K_2, \ldots, K_z\}$ [4].

Произвольное тайное сообщение M_i можно скрыть в контейнере C_j при использовании ключа K_m: $M_i \in \mathbf{M}$, $i = 1, 2, \ldots, n$; $C_j \in \mathbf{C}$, $j = 1, 2, \ldots, p$; $K_m \in \mathbf{K}$, $m = 1, 2, \ldots, z$. Результатом такого типа преобразований будет заполненный контейнер (или стеганосообщение) S_q, относящийся к множеству заполненных контейнеров или стеганосообщений **S**: $\mathbf{S} = \{S_1, S_2, \ldots, S_r\}$, $q = 1, 2, \ldots, r$.

Функцию **F**, определенную на $\mathbf{M} \times \mathbf{C} \times \mathbf{K}$ со значениями в **S**, будем отождествлять с осаждением или встраиванием сообщения M_i из множества **M** в контейнер C_j из множества **C** на основе ключа из множества **K**, предусматривающего использование соответствующего алгоритма осаждения:

$$\mathbf{F}: \mathbf{M} \times \mathbf{C} \times \mathbf{K} \longrightarrow \mathbf{S}. \qquad (1)$$

Соотношение (1) формально описывает процедуру осаждения тайного сообщения M_i в контейнер C_j на основе выбранного ключа K_m.

Функцию $\mathbf{F}^{-1}$, определенную на $\mathbf{S} \times \mathbf{K}$ со значениями в **M**, будем отождествлять с извлечением тайного сообщения $M_i \in \mathbf{M}$ из стеганосообщения $S_q \in \mathbf{S}$:

$$\mathbf{F}^{-1}: \mathbf{S} \times \mathbf{K} \longrightarrow \mathbf{M}, \mathbf{C}. \qquad (2)$$

Таким образом, выражение (2) определяет обратное по отношению к (1) отображение, которое каждому элементу S_q множества **S** и фиксированному элементу множества **K** ставит в соответствие элемент M_i множества **M** и элемент C_j множества **C**.

Соотношение (2) формально описывает процедуру извлечения сообщения из контейнера на основе того же выбранного метода [1]. При извлечении сообщения для подтверждения того, что оно не было модифицировано, вычисляется его контрольная сумма и сравнивается с M_H.

С учетом всех вышеуказанных обозначений опишем стеганографическую систему, в которой в качестве контейнера используется электронный документ формата EPUB.

В составе электронной книги выделим три контейнера, в которые производится внедрение информации:

$$\mathbf{C} = \{C_{JPG}, C_{CSS}, C_{XHTML}\}, \qquad (3)$$

где C_{JPG} – обложка книги, изображение, представленное в формате JPG; C_{CSS} – файл каскадных таблиц стиля, хранящий конфигурацию отображения книги; C_{XHTML} – набор XHTML-файлов, количество которых зависит от глав в книге, а также несущий основную текстовую информацию.

Для выполнения процедуры внедрения информации в соответствии с контейнерами (3) используются следующие стеганографические ключи:

$$\mathbf{K} = \{K_{LSB}, K_{CSS}, K_Q\}, \qquad (4)$$

где K_{LSB} – ключ, представляющий собой осаждение информации стеганографическим методом LSB (Least Significant Bits) [1]. Суть метода замены наименее значащего бита заключается в сокрытии информации путем изменения последних битов изображения, кодирующих цвет, на биты скрываемого сообщения. Разница между пустым и заполненным контейнерами должна быть не ощутима для органов восприятия человека. Рассмотрим этот метод на примере 24-битного растрового RGB-изображения.

Приведенный на рис. 1 пример показывает, как сообщение «100011» может быть скрыто в двух пикселях 24-битного изображения.

Пустой контейнер:	11111010	10101111
	10000101	10110110
	11000101	10010101
Осаждаемое сообщение:	100 011	
Заполненный контейнер:	1111101**1**	1010111**0**
	1000010**0**	1011011**1**
	1100011**0**	1001010**1**

Рис. 1. Пример осаждения двоичного сообщения с использованием метода LSB

Каждый пиксель кодируется тремя байтами, каждый байт определяет интенсивность красного (Red), зеленого (Green) и синего (Blue) цвета. Совокупность интенсивностей цвета в каждом из трех каналов определяет оттенок пикселя. Изменяя наименее значащий бит, меняется значение байта на единицу.

Таким образом, контейнер C_{JPG} является основным из используемых и содержит непосредственно само осаждаемое сообщение M_O.

K_{CSS} – ключ, реализующий осаждение сообщения в файл каскадных таблиц стилей. Для визуального оформления XHTML-разметки предназначена технология CSS. CSS (англ. Cascading Style Sheets – каскадные таблицы стилей) – технология описания внешнего вида документа, оформленного языком разметки. Если XHTML предоставляет информацию о структуре электронной книги, то таблицы стилей сообщают, как она должна выглядеть. Стиль – это совокупность правил, применяемых к элементу гипертекста и определяющих способ его отображения. Стиль включает все типы элементов дизайна: шрифт, фон, цвета ссылок, поля и расположение объектов. Таблица стилей – это совокупность стилей, применимых к гипертекстовому документу. Каскадирование – это порядок присвоения различных стилей.

При создании CSS-файлов существует практика хранения изображений в формате base64.

Для реализации данного подхода необходимо закодировать изображение в формате base64. Далее положить получившуюся строку в CSS-файл, заменяя «ТИП» на MIME-тип изображения – JPEG/PNG/GIF или BMP и «КОД» на нужную строку в base64 (листинг 1).

```
.some_class {
    background-image: url("КОД");
}
```

Листинг 1. Пример использования кодировки base64 для задания фона в стиле

При помощи ключа K_{CSS} происходит внедрение обложки книги в файл со стилями [5]. Предварительно в обложку внедряется сообщение с использованием стеганографического алгоритма LSB. Контейнер C_{CSS} предназначен для дублирования основного сообщения, что в свою очередь повышает стеганографическую стойкость.

K_Q – ключ, реализующий осаждение сообщения в главы книги с использованием метода замены кавычек в файлах XHTML. XHTML – это основанный на XML язык разметки гипертекста, максимально приближенный к стандартам HTML. XHTML отличается от HTML строгостью написания кода. Если HTML позволяет писать практически любые конструкции и браузер их корректно распознает, то с появлением XHTML это стало невозможным. XHTML требует строгого соблюдения всех правил, предъявляемых W3C.

Известно, что интерпретатор XHTML-документа не придает значения, какой тип кавычек используется при его создании. Следовательно, если заменить какую-нибудь пару кавычек в валидном XHTML-документе, например с двойной на одинарную, то семантический смысл документа не изменится. Используя эту технику, в XHTML можно осадить бинарную последовательность.

При встраивании последовательности бит условимся, что единице будет соответствовать двойная кавычка, а нулю – одинарная. Начиная с первой пары кавычек в документе, будем ставить ей в соответствие бит встраиваемого сообщения и, при необходимости, изменять тип кавычки на противоположный. (Например, первая пара кавычек в документе двойная, а первый бит осаждаемой последовательности нулевой, следовательно, необходимо тип кавычек заменить на одинарный.) После того, как место в одном XHTML-файле закончилось, следует перейти к следующей главе. Данную процедуру необходимо выполнять до тех пор, пока сообщение не закончится, либо же когда глав в книге больше не останется [5, 6]. Таким образом, контейнер C_{XHTML} используется в качестве дополнительного и хранит хэш основного сообщения M_H.

С учетом всех описанных элементов стеганографической системы функция встраивания сообщения **F** будет выглядеть следующим образом:

$$\mathbf{F}\colon \mathbf{M}\ \{M_O, M_H\} \times \mathbf{C}\ \{C_{JPG}, C_{CSS}, C_{XHTML}\} \times \\ \times\ \mathbf{K}\ \{K_{LSB}, K_{CSS}, K_Q\} \longrightarrow \mathbf{S}.$$

Функция извлечения $\mathbf{F}^{-1}$ будет следующей:

$$\mathbf{F}^{-1}\colon \mathbf{S} \times \mathbf{K}\ \{K_{LSB}, K_{CSS}, K_Q\} \longrightarrow \mathbf{M}\ \{M_O, M_H\}, \\ \mathbf{C}\ \{C_{JPG}, C_{CSS}, C_{XHTML}\}.$$

Таким образом, стеганографическая модель при помощи нового метода была адаптирована для осуществления процедуры внедрения информации в цифровой файл формата EPUB.

Для демонстрации описанной стеганографической системы осаждения метки в электронный документ формата EPUB создано программное средство «EPUB Modifier». В качестве технологии для создания приложения была выбрана Windows Forms, которая позволяет разработать приложение с полнофункциональным графическим интерфейсом, простое в развертывании и обновлении, способное работать при наличии или отсутствии подключения к Интернету и использующее более безопасный доступ к ресурсам на локальном компьютере по сравнению с традиционными приложениями Windows. В Windows Forms форма – это визуальная поверхность, на которой выводится информация для пользователя [7]. Приложение Windows Forms строится путем помещения элементов управления на форму и написания кода для реагирования на действия пользователя, такие как щелчки мыши или нажатия клавиш. Элемент управления – это отдельный элемент пользовательского интерфейса, предназначенный для отображения или ввода данных.

Выбранная технология обладает достаточным набором инструментов для осуществления процедуры осаждения/извлечения информации в стеганографический контейнер формата EPUB [7].

В процессе внедрения сообщения M_i контейнер **C** проходит через несколько этапов обработки, число которых совпадает с количеством используемых в модели ключей **K**. Для грамотного построения процедуры осаждения был применен шаблон проектирования «Chain of responsibility». Цепочка обязанностей («Chain of responsibility») – поведенческий шаблон проектирования, благодаря которому удается избежать «жесткой» привязки отправителя запроса к получателю, позволяя тем самым нескольким объектам обработать запрос. Все возможные обработчики запроса образуют цепочку, а сам запрос перемещается по этой цепочке, пока один из ее объектов не обработает запрос. Каждый объект при получении запроса выбирает, либо обработать запрос, либо передать выполнение запроса следующему по цепочке. Данный шаблон применяется в следующих случаях: наличие более одного объекта, который может обработать определенный запрос; необходимость передачи запроса на выполнение одному из нескольких объектов, точно не определяя, какому именно; необходимость задания объектов динамически. Исходя из описания шаблона «Chain of responsibility», можно сказать, что его использование в разработке приложения является целесообразным. UML-диаграмма паттерна «Chain of responsibility» представлена на рис. 2.

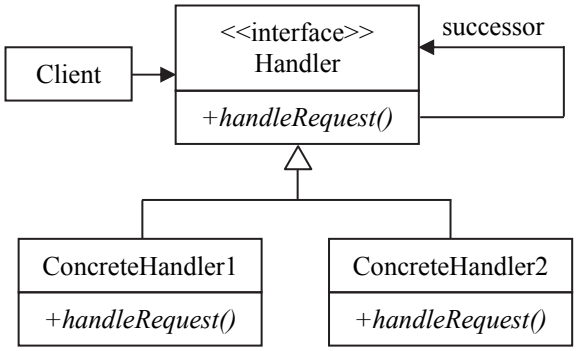

Рис. 2. UML-диаграмма паттерна «Chain of responsibility»

Handler определяет интерфейс для обработки запроса, а также может определять ссылку на следующий обработчик запроса. *ConcreteHandler1* и *ConcreteHandler2* – конкретные обработчики, которые реализуют функционал для обработки запроса. *Client* отправляет запрос объекту *Handler*.

В приложении «EPUB Modifier» описанный паттерн реализован на основе двух классов *Embedder* и *Extracter*. В классе *Embedder* содержатся свойства для хранения следующего звена цепочки, обрабатываемой книги, внедряемого сообщения и для директории, в которой находится файл-контейнер. Основным методом является *EmbedMessage*, который выполняет переопределенный метод *Embed* класса потомка, а также запускает выполнение следующего звена цепочки, если оно есть. Для добавления класса в цепочку осаждения сообщения необходимо реализовать класс *Embedder*, а также переопределить абстрактный метод *Embed*. Далее следует разместить объект класса в свойстве *Next* предыдущего звена цепочки.

Реализация родительского класса *Embedder* представлена в листинге 2.

```
abstract class Embedder{
    public Embedder Next;
    public EpubBook Book;
    public Dictionary<string, string> files
        = new Dictionary<string, string>();
    public string ContentDirectoryPath;
    public string embedMessage;
    3 references
    public Embedder(EpubBook book, string message){
        Book = book;
        ContentDirectoryPath = book.Schema.ContentDirectoryPath;
        embedMessage = message;}
    2 references
    public void EmbedMessage() {
        Embed();
        if (Next != null) Next.EmbedMessage();}
    4 references
    public abstract void Embed();
}
```

Листинг 2. Программный код класса *Embedder*

Использование цепочки обязанностей дает следующие преимущества: ослабление связан-

ности между объектами (отправителю и получателю запроса ничего неизвестно друг о друге; клиенту неизвестна цепочка объектов, какие именно объекты составляют ее, как запрос в ней передается); в цепочку можно добавлять новые типы объектов, которые реализуют общий интерфейс; расположение последовательности объектов-обработчиков в цепочке в зависимости от их приоритета.

Заключение. Рассмотренная в данной статье модель основана на использовании комбинации стеганографических методов. В список методов входят LSB, метод закодированного изображения в каскадных таблицах стилей, а также метод замены кавычек в файлах разметки. Применение перечисленного сочетания методов позволяет увеличить стеганографическую стойкость системы и добавить возможность проверки осажденного сообщения в цифровом файле формата EPUB.

Представленная модель стеганографической системы позволяет проводить процедуру внедрения сообщения в электронные книги формата EPUB с учетом особенностей содержимого файла-архива. Система на основе рассмотренной модели может быть применена для внесения цифрового водяного знака в электронные книги с целью защиты авторского права на интеллектуальную собственность и подтверждения целостности документа, а также для размещения различных скрытых стеганографических меток в каждую копию электронной книги, для выявления канала несанкционированного копирования и распространения.

В сравнении с системой, в которой в качестве контейнера используется только метод замены кавычек и закодированного изображения в каскадных таблицах стилей, описанная система обладает более надежной стеганографической стойкостью за счет применения дополнительного контейнера для дублирования сообщения, а также отдельного контейнера для хранения контрольной суммы исходного сообщения.

Представлено программное средство, позволяющее производить процедуру осаждения/извлечения сообщения в стеганографический контейнер формата EPUB.

Литература

1. Урбанович П. П. Защита информации методами криптографии, стеганографии и обфускации. Минск: БГТУ, 2016. 220 с.

2. Конахович Г. Ф., Пузыренко А. Ю. Компьютерная стеганография. Теория и практика. Киев: МК-Пресс, 2006. 288 с.

3. Сущеня А. А. Программное средство стеганографического преобразования текстов-контейнеров на основе языка разметки XML // 69-я науч.-техн. конф. учащихся, студентов и магистрантов. Минск, 2–13 апр. 2018 г. / Белорус. гос. технол. ун-т. Минск, 2018. С. 81–84.

4. Шутько Н. П., Романенко Д. М., Урбанович П. П. Математическая модель системы текстовой стеганографии на основе модификации пространственных и цветовых параметров символов текста // Труды БГТУ. 2015. № 6: Физ.-мат. науки и информатика. С. 152–156.

5. Сущеня А. А. Стеганографический метод внедрения текстовой информации в контейнер формата EPUB // 70-я науч.-техн. конф. учащихся, студентов и магистрантов. Минск, 15–20 апр. 2019 г. / Белорус. гос. технол. ун-т. Минск, 2019. С. 247–251.

6. Шутько Н. П. Особенности и формальное описание процесса осаждения секретной информации в текстовые документы на основе стеганографии // Труды БГТУ. 2014. № 6: Физ.-мат. науки и информатика. С. 121–124.

7. Docs.microsoft.com: сайт. URL: https://docs.microsoft.com/ru-ru/dotnet/framework/winforms/windows-forms-overview (дата обращения: 02.11.2019).

References

1. Urbanovich P. P. *Zashchita informatsii metodami kriptografii, steganografii i obfuskatsii* [The protection of information based on the methods by cryptography, steganography and obfuscation]. Minsk, BGTU Publ., 2016. 220 p.

2. Konakhovich G. F., Puzyrenko A. Yu. *Komp'yuternaya steganografiya. Teoriya i praktika* [Computer steganography. Theory and practice]. Kyiv, MK-Press Publ., 2006. 288 p.

3. Sushchenia A. A. Software tool for steganographic transformation of container texts based on XML markup language. *69-ya nauchno-tekhnicheskaya konferentsiya uchashchikhsya, studentov i magistrantov* [69th Scientific and technical conference of students, students and undergraduates]. Minsk, 2018, pp. 81–84 (In Russian).

4. Shutko N. P., Romanenko D. M., Urbanovich P. P. Mathematical model of textual shorthand system based on modification of spatial and color parameters of text symbols. *Trudy BGTU* [Proceedings of BSTU], 2015, no. 6: Physics and Mathematics. Informatics, pp. 152–156 (In Russian).

5. Sushchenia A. A. Steganographic method of introduction of textual information into the container of EPUB format. *70-ya nauchno-tekhnicheskaya konferentsiya uchashchikhsya, studentov i magistrantov* [70th Scientific and technical conference of students, students and undergraduates]. Minsk, 2019, pp. 247–251 (In Russian).

6. Shutko N. P. Features and formal description of the process of deposition of secret information in text documents based on shorthand. *Trudy BGTU* [Proceedings of BSTU], 2014, no. 6: Physics and Mathematics. Informatics, pp. 121–124 (In Russian).

7. Docs.microsoft.com. Available at: https://docs.microsoft.com/ru-ru/dotnet/framework/winforms/windows-forms-overview (accessed 02.11.2019).

Информация об авторах

Сущеня Артем Александрович – магистрант. Белорусский государственный технологический университет (220006, г. Минск, ул. Свердлова, 13а, Республика Беларусь). E-mail: asuschenya@gmail.com

Блинова Евгения Александровна – старший преподаватель кафедры информационных систем и технологий. Белорусский государственный технологический университет (220006, г. Минск, ул. Свердлова, 13а, Республика Беларусь). E-mail: evgenia.blinova@belstu.by

Information about the authors

Sushchenia Artsiom Aleksandrovich – Master's degree student. Belarusian State Technological University (13a, Sverdlova str., 220006, Minsk, Republic of Belarus). E-mail: asuschenya@gmail.com

Blinova Evgeniya Aleksandrovna – Senior Lecturer, the Department of Information Systems and Technology. Belarusian State Technological University (13a, Sverdlova str., 220006, Minsk, Republic of Belarus). E-mail: evgenia.blinova@belstu.by

Поступила после доработки 13.11.2019