УДК 003.26

Е. А. Блинова, П. П. Урбанович
Белорусский государственный технологический университет

СТЕГАНОГРАФИЧЕСКИЙ МЕТОД НА ОСНОВЕ ВСТРАИВАНИЯ ДОПОЛНИТЕЛЬНЫХ ЗНАЧЕНИЙ КООРДИНАТ В ИЗОБРАЖЕНИЯ ФОРМАТА SVG

Приведено формальное описание метода и алгоритма встраивания цифрового водяного знака в файлы векторных изображений формата SVG на основе использования дополнительных параметров в описании путей элементов файла. Рассмотрены стеганографические методы, которые могут быть адаптированы для векторных изображений формата SVG. При этом учитываются форматы файлов (как подмножество формата XML), особенности формирования векторного изображения на основе описания путей элементов, описания текста и цветовых параметров элементов. Рассмотрены возможные ключевые последовательности для описания цифрового водяного знака, предназначенного для защиты права интеллектуальной собственности от незаконного копирования и распространения изображения. Осаждение скрытой информации предусматривает добавление точек, распределенных по элементам файла векторного изображения в соответствии со значениями цифровой ключевой последовательности с учетом минимального увеличения исходного файла векторного изображения. Рассмотрен алгоритм обратного стеганографического преобразования для доказательства подлинности и целостности цифрового векторного изображения. Проанализирована возможность совместного применения различных стеганографических методов с целью формирования многоключевой стеганографической системы, предназначенной для идентификации копии цифрового векторного изображения. Произведена оценка возможности встраивания ключевой последовательности в файл цифрового векторного изображения.

Ключевые слова: стеганография, векторное изображение, SVG, авторское право, описание координат.

E. A. Blinova, P. P. Urbanovich
Belarusian State Technological University

A STEGANOGRAPHIC METHOD BASED ON THE EMBEDDING OF ADDITIONAL COORDINATES INTO IMAGES OF SVG FORMAT

A formal description of the algorithm for embedding the digital watermark in vector SVG image files is presented, based on the addition of additional elements to the description of the paths of the file elements. Steganographic methods are considered that are applicable for vector images of SVG format, due to the capabilities of the file format, both subsets of XML, and features of vector image format, such as description of element paths, text description and color parameters of elements. Possible key sequences for describing the digital watermark are considered. The embedding of hidden information involves adding additional points distributed over the elements of the vector image file in accordance with the values of the digital key sequence, taking into account the minimum increase in the original vector image file. An algorithm of reverse steganographic transformation for proving the authenticity and integrity of a digital vector image is considered. The possibility of combined application of various steganographic methods for the purpose of forming a multi-key steganographic system designed to identify a copy of a digital vector image is considered. An estimation of the possibility of embedding this key sequence in a file of a digital vector image was made. The digital watermark is designed to protect the right of intellectual property from illegal copying and distribution.

Key words: steganography, vector graphics, SVG, copyright, path description.

Введение. Цифровые технологии не только дают возможность хранить и передавать различные типы данных (изображения, тексты, звук и др.), но и являются способом их создания. Но преимущества, которые дают цифровые технологии, легко переносятся на реализацию различных деструктивных действий: незаконное копирование, распространение, использование или даже уничтожение информации. В связи с этим все более острой является проблема разработки и использования методов и инструментальных средств защиты информации, в том числе защиты прав интеллектуальной собственности [1–3].

Одним из направлений решения проблемы являются технологии цифрового водяного знака.

Особый интерес представляет защита графической информации (файлов графических

форматов), как одного из самых востребованных в настоящее время видов продукции. Особенностями защиты является сравнительно большой объем, необходимый для хранения такого рода информации, и как следствие – широкие возможности встраивания (осаждения) различных невидимых меток.

В основном исследования сосредотачиваются на растровых форматах изображений, таких как bmp, jpg и др., для которых разработано большое число методов защиты, в том числе и стеганографических. Однако в связи с набирающим популярность использованием векторных изображений интерес представляет исследование нового класса стеганографических методов для обеспечения целостности таких изображений и защиты прав интеллектуальной собственности.

В данной работе предлагается новый стеганографический метод для файлов векторных изображений в формате SVG и описывается алгоритм его реализации.

Основная часть. Файлы SVG (Scalable Vector Graphics) – векторные графические файлы, предназначенные для описания двумерной векторной и смешанной векторной и растровой графики в формате XML. Они часто используются для создания анимированных изображений торговых марок, инфографики, экспорта различных чертежей, выполненных в САПР для отображения в сторонних приложениях, деталей сайтов – иконок, фонов, кнопок, иногда довольно сложных и детализированных. Особенностями данного формата являются небольшой размер файлов, масштабируемость, интеграция с HTML документами, возможность встраивания растровой графики, возможность редактирования в текстовых редакторах и поддержка в большинстве современных браузеров, таких как Google Chrome, Internet Explorer, Mozilla Firefox и Safari. Редактировать SVG файлы можно в большинстве графических редакторов (в данной статье использовались редакторы CorelDraw 10 и Inkscape). В пакет Microsoft Office 2016 была добавлена поддержка прямого импорта таких файлов.

Файл SVG включает в себя три типа объектов: фигуры, изображения и текст, причем для них всех может быть задана анимация. При описании геометрических фигур используется плоская координатная модель, на которой можно задать координатами вершин точки, линии, многоугольники, основные геометрические фигуры и кривые Безье.

Поскольку SVG файлы являются подмножеством файлов формата XML, то к ним могут быть применены классические методы текстовой стеганографии, такие как метод конечных пробелов и табуляций, а также методы, характерные для файлов разметки, такие как метод замены регистра тегов, методы подмены и перестановки атрибутов. Кроме того, при описании геометрических фигур используется цветовая модель RGB, что позволяет внедрять скрытую информацию в незначительном изменении параметров цвета, используя, например, стеганографический метод LSB.

Отметим также, что особенности формата позволяют использовать и другие методы осаждения скрытой информации. Формат тегов описания геометрических фигур, таких как точки, линии, полиномы и др., позволяет размещать скрытую информацию в добавлении дополнительных элементов в геометрических фигурах.

Рассмотрим текст, продемонстрированный на рис. 1. Он формирует исходные полигоны в файле векторного изображения. На рис. 2 – отображение этого текста в SVG для браузера Google Chrome.

```
viewBox="0 0 21000 29700" xmlns:xlink="http://www.w3.org/1999/xlink">
<g id="Слой_x0020_1">
  <metadata id="CorelCorpID_0Corel-Layer"/>
  <polygon id="Poly-1" stroke="black" stroke-width="19" fill="white"
           points="100 100 1500 100 1500 1500 100 1500">
   </polygon>
  <polygon id="Poly-2" stroke="blue" stroke-width="30" fill="white"
           points="1700 100 3500 100 3500 2500 1700 2500">
   </polygon>
 </g>
</svg>
```

Рис. 1. Часть текста файла SVG
с исходными полигонами

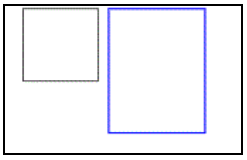

Рис. 2. Файл SVG с исходными полигонами

Рассмотрим также на рис. 3 и 4 соответственно – текст, формирующий полигоны с установленными дополнительными точками, и отображение этого текста в браузере Google Chrome.

```
<g id="Слой_x0020_1">
  <metadata id="0-Layer"/>
  <polygon id="Poly-1" stroke="black" stroke-width="19" fill="white"
           points="100 100  1000 100  1500 100 1500 1500 100 1500">
   </polygon>
  <polygon id="Poly-2" stroke="blue" stroke-width="30" fill="white"
           points="1700 100  2500 100  3500 100 3500 2500 1700 2500">
   </polygon>
 </g>
</svg>
```

Рис. 3. Часть текста файла SVG с полигонами
с установленными дополнительными точками

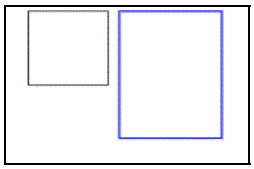

Рис. 4. Файл SVG с полигонами
с установленными дополнительными точками

Дополнительные точки, установленные на отрезке, не отображаются. Поэтому можно установить любое количество дополнительных точек на любом отрезке геометрической фигуры и разместить в особенности их расположения авторскую информацию.

Текст в SVG-файлах также может быть описан как текст или конвертирован в кривые. При конвертировании в кривые скрытые данные могут быть осаждены в дополнительные точки на кривых, что отображено на рис. 5.

```
  <path class="fil0" d="M2115 3600150 -5c3,21 8,37 17,50 9,13 22
11,-23 11,-35 0,-13 -4,-24 -11,-33 -7,-9 -19,-17 -36,-24 -11,-4
-15,-32 -15,-51 0,-20 6,-39 18,-57 11,-18 28,-31 51,-41 22,-9 46
3,-25 -12,-44 -28,-57 -16,-13 -39,-20 -70,-20 -33,0 -56,6 -71,18
23,11 40,25 52,41 11,17 16,36 16,58 0,21 -6,41 -18,60 -12,19 -30
58,-48 -14,-21 -21,-45 -22,-72z"/>
  <path id="1" class="fil0" d="M2624 37301-156 -405 58 0 105 294
  <path id="2" class="fil0" d="M3075 357110 -47 171 0 0 149c-26,
57,-42 -73,-74 -17,-32 -25,-68 -25,-108 0,-39 8,-76 25,-110 16,-
14,17 24,39 31,671-49 13c-6,-21 -13,-37 -22,-49 -9,-12 -22,-21 -
11,12 -19,25 -25,40 -10,25 -16,52 -16,81 0,37 7,67 19,91 12,24 3
-118 0z"/>
```

Рис. 5. Часть текста файла SVG
с преобразованным в кривые текстом

Возможность осаждения скрытой информации в SVG-изображениях исследуется в многочисленных работах.

В частности, в работах [4–5] рассматривается механизм внедрения скрытой информации в SVG-изображения, который основан на модификации дробных частей координат вершин геометрических фигур, что аналогично методу LSB для растровых изображений. Недостатком таких методов является хоть и незначительное, но изменение форм элементов изображения, что может сказаться на его точности, особенно это касается импортированных из САПР чертежей. В источнике [6] предлагается устанавливать дополнительные точки в ребрах геометрических фигур таким образом, чтобы для сообщения, состоящего из нулей и единиц, для элемента {0} устанавливать дополнительную точку на согласованном расстоянии, а для элемента {1} – на удвоенном.

Недостатком данного метода является внедрение большого числа точек даже для сравнительно небольшого скрытого сообщения, что резко увеличивает размер файла изображения, повышая, таким образом, вероятность обнаружения наличия скрытой информации.

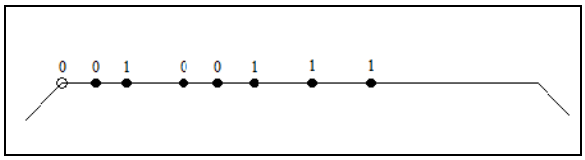

Рис. 6. Алгоритм внедрения
дополнительных точек в SVG файл

В статьях [7–8] обосновывается возможность использования внедрения дополнительных вершин в описание геометрических фигур в файле SVG для осаждения уникального цифрового водяного знака (Digital Fingerprint) для подтверждения авторства изображения. В качестве такого знака должна быть выбрана числовая последовательность, которая, не раскрывая каких-либо личных сведений об авторе, позволит, тем не менее, однозначно его идентифицировать. Для генерации такой числовой последовательности предлагается использовать цепочку 16-значных числовых последовательностей номеров кредитных/дебетовых карт, выданных автору, плюс 4 знака даты срока действия карты. Такая последовательность позволяет автору точно идентифицировать себя предъявлением своих договоров на банковские карты, однако не раскрывает его персональных данных. При наличии у автора документа нескольких карт предлагается включать их в цепочку цифрового водяного знака и исключать их из цепочки, как только срок действия карты закончился. Таким образом, при извлечении водяного знака можно сделать вывод о периоде, в который данный цифровой водяной знак был внесен. Для предлагаемого стеганографического метода цифровая последовательность может выбираться и по другому принципу. Предъявляемое к ней требование – она должна состоять из конечного числа цифр и служить доказательством авторства.

Алгоритм внедрения дополнительных вершин может быть представлен в виде следующей последовательности шагов.

Шаг 1. Подсчет количества геометрических объектов N в файле изображения.

Шаг 2. Подсчет количества вершин геометрических объектов и помещение его в массив:

$$\left(n_i\right)_{i=1}^{N}. \qquad (1)$$

Шаг 3. Подсчет общего количества вершин M в файле:

$$M = \sum_{i=1}^{N} n_i . \qquad (2)$$

Шаг 4. Помещение координат вершин в набор массивов C:

$$((x_{11}, y_{11}), (x_{12}, y_{12}), \dots (x_{1n_1}, y_{1n_1})),$$
$$((x_{21}, y_{21}), (x_{22}, y_{22}), \dots (x_{2n_2}, y_{2n_2})), \dots$$
$$((x_{N1}, y_{N1}), (x_{N2}, y_{N2}), \dots (x_{Nn_N}, y_{Nn_N})). \quad (3)$$

Шаг 5. Подсчет количества знаков L в цепочке ключевой цифровой последовательности водяного знака, где l_j – j-й элемент ключевой последовательности:

$$\left(l_j\right)_{j=1}^{L}. \quad (4)$$

Шаг 6. Подсчет отношения P, где

$$P = \frac{M}{L}. \quad (5)$$

Из данного соотношения можно сделать вывод о возможности наложения цифрового знака на изображение. Хотя из реализации метода следует, что такое встраивание возможно при $L \le M$, но при реальном использовании метода желательно не превышать отношения как минимум вдвое.

Шаг 7. Количество дополнительных вершин в полигоне P_i, $i \in [1; N]$ определяется следующим соотношением:

$$P_i = \frac{n_i L}{M}. \quad (6)$$

При этом P_i округляются вниз до ближайшего целого, за исключением максимального значения количества дополнительных вершин на полигон $P_{\max}$, которое вычисляется следующим образом:

$$P_{\max} = L - \sum_i P_i, \quad (7)$$

где $i \in [1; N]$, $i \ne i_{\max}$.

Шаг 8. Отношение Q_i исходного количества вершин i-го полигона n_i к количеству встраиваемых вершин P_i определяется следующим соотношением:

$$Q_i = \frac{n_i}{P_i}. \quad (8)$$

При этом Q_i округляются вниз до ближайшего целого.

Шаг 9. Разбиение цепочки ключевой последовательности на отрезки по два цифровых символа:

$$\left([a_k,\ b_k]\right)_{k=1}^{\frac{L}{2}}. \quad (9)$$

Причем, если значение очередного цифрового символа равно нулю, то оно заменятся значением 10.

Шаг 10. Начиная с первой вершины первого полинома, отступить $(Q_1 - 1)$ вершин и поставить дополнительную вершину в отношении λ:

$$\lambda = \frac{a_1}{b_1} \quad (10)$$

в точке со следующими координатами:

$$X = \frac{x_t + \lambda x_{t+1}}{1+\lambda}, \quad (11)$$

$$Y = \frac{y_t + \lambda y_{t+1}}{1+\lambda}. \quad (12)$$

Шаг 11. Повторение шага 10, с использованием в качестве стартовой вершины (x_{t+1}, y_{t+1}), $t \in [1; P_i]$, и переходя к следующему полиному, как только количество дополнительных вершин достигнет P_i, $i \in [1; N]$, до завершения цепочки символов.

Алгоритм доказательства авторства состоит в следующем.

Шаг 1. Подсчет количества геометрических объектов N в файле изображения.

Шаг 2. Подсчет количества вершин R геометрических объектов r_i, $i \in [1; N]$, в файле изображения:

$$R = \sum_{i=1}^{N} r_i. \quad (13)$$

Шаг 3. Подсчет количества знаков L в цепочке водяного знака.

Шаг 4. Подсчет знаков в исходном файле:

$$M = R - L. \quad (14)$$

Шаг 5. Помещение координат вершин в набор массивов C':

$$((x_{11}, y_{11}), (x_{12}, y_{12}), \dots (x_{1r_1}, y_{1r_1})),$$
$$((x_{21}, y_{21}), (x_{22}, y_{22}), \dots (x_{2r_2}, y_{2r_2})), \dots$$
$$((x_{N1}, y_{N1}), (x_{N2}, y_{N2}), \dots (x_{Nr_N}, y_{Nr_N})). \quad (15)$$

Шаг 6. Начиная с первой вершины первого полинома выбрать три последовательных вершины с координатами (x_t, y_t), (x_{t+1}, y_{t+1}) и (x_{t+2}, y_{t+2}). Если для этих трех вершин выполняется соотношение

$$(x_t - x_{t+1})^2 + (y_t - y_{t+1})^2 + (x_{t+1} - x_{t+2})^2 + (y_{t+1} - y_{t+2})^2 = (x_t - x_{t+2})^2 + (y_t - y_{t+2})^2, \quad (16)$$

то удалить вершину с координатами (x_{t+1}, y_{t+1}) и сформировать массив Z, состоящий из следующих элементов:

$$Z = \{n_k, t_k, (x_k, y_k)\}, \quad (17)$$

где n_k – номер текущего полинома, t_k – номер текущей вершины в данном полиноме, (x_k, y_k) – координаты удаляемой вершины.

Шаг 7. Подсчет количества элементов массива Z. Если $Z = L$, произвести внедрение элементов числовой последовательности в модифицированный файл по описанному ранее алгоритму, используя элементы массива Z для определения позиции внедрения дополнительных элементов.

Шаг 8. При совпадении исходного и полученного файла можно сделать вывод о соответствии цифровой метки исходной цепочке, и, следовательно, доказательстве авторства.

Заключение. Описанный метод и алгоритм его реализации позволяют минимизировать увеличение размера исходного файла, что дает возможность использования метода и для передачи скрытых сообщений. Кроме того, наше предложение позволяет распределить внедренную информацию по всем фигурам, что при некотором искажении исходного изображения (контейнера) не приводит к потере осажденной информации.

Наиболее подходящими, с точки зрения встраивания скрытой информации, являются изображения, содержащие большое число различных геометрических элементов, либо конвертированные из растровых изображений.

Одновременное применение нескольких стеганографических методов позволяет решить две задачи. С одной стороны, различные методы могут быть использованы для передачи конфиденциальной информации нескольким абонентам. С другой стороны, комбинирование методов может помочь контролировать целостность осаждаемой информации, что может быть использовано, например, при решении задачи защиты права интеллектуальной собственности на изображения.

Литература

1. Text steganography application for protection and transfer of the information / P. P. Urbanovich [et al.] // New Electrical and Electronic Technologies and their Industrial Implementation: proc. of the 6-th Intern. Conf., Zakopane, Poland, 23–26.06.2009 / Lublin University of Technology; Media Patronage “Przeglad Elektrotechniczny”. Lublin. 2009. P. 60–61.

2. Shutko N., Blinova E. The use of aprosh and kerning in text steganography // New Electrical and Electronic Technologies and their Industrial Implementation; proc. of the 9-th Intern. Conf., Zakopane, Poland, 23–26.06.2015 / Lublin University of Technology; Media Patronage “Przeglad Elektrotechniczny”. Lublin. 2015. P. 77.

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

4. Zhou X., Pan X. Watermark-Based Scheme to Protect Copyright of SVG Data // ICCIAS. 2006. Vol. 2. P. 1199–1202.

5. Topology-Preserving Watermarking of Vector Graphics / S. Huber [et al.] // International Journal of Computational Geometry & Applications. 2014. Vol. 1. P. 61–86.

6. Steganographic Algorithm For Information Hiding Using Scalable Vector Graphics Images / B. Madoš [et al.] // Acta Electrotechnica et Informatica. 2014. Vol. 14, no. 4. P. 42–45.

7. Blinova E., Shutko N. The use of steganographic methods in SVG format graphic files // New Electrical and Electronic Technologies and their Industrial Implementation; proc. of the 10-th Intern. Conf., Zakopane, Poland, 23–26.06.2017 / Lublin University of Technology; Media Patronage “Przeglad Elektrotechniczny”. Lublin. 2017. P. 45.

8. Блинова Е. А. Применение стеганографических методов при хранении картографической информации в экспертной системе прогнозирования последствий пролива нефтепродуктов // Сахаровские чтения 2017 года: Экологические проблемы XXI века: материалы 17-й МНК, 18–19 мая 2017 / Международный государственный экологический институт им. Д. А. Сахарова Белорусского государственного университета. Минск. 2017. С. 223–224.

References

1. Urbanovich P. P., Chourikov K. V., Rimarev A. V., Urbanovich N. P. Text steganography application for protection and transfer of the information. Proc. of the 6th Intern. Conf. (New Electrical and Electronic Technologies and their Industrial Implementation). Lublin, 2009, pp. 60–61.

2. Shutko N., Blinova E. The use of aprosh and kerning in text steganography. Proc. of the 9th Intern. Conf. (New Electrical and Electronic Technologies and their Industrial Implementation). Lublin, 2015, p. 77.

3. 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., 2017. 220 p.

4. Zhou X., Pan X. Watermark-Based Scheme to Protect Copyright of SVG Data. *ICCIAS*, 2006, vol. 2, pp. 1199–1202.

5. Huber S., Held M., Kwitt R., Meerwald P. Topology-Preserving Watermarking of Vector Graphics. *International Journal of Computational Geometry & Applications*, 2014, vol. 1, pp. 61–86.

6. Madoš B., Hurtuk J., Čopjak M., Hamaš P., Ennert M. Steganographic Algorithm For Information Hiding Using Scalable Vector Graphics Images. *Acta Electrotechnica et Informatica*, 2014, vol. 14, no. 4, pp. 42–45.

7. Blinova E., Shutko N. The use of steganographic methods in SVG format graphic files. Proc. of the 10th Intern. Conf. (New Electrical and Electronic Technologies and their Industrial Implementation). Lublin, 2015, p. 45.

8. Blinova E. A., Smelov V. V. [An application of the steganographic methods in the storage of cartographic information in the expert forecast system of the consequences of petroleum products spillage]. *Materialy 17 Mezhdunarodnoy nauchnoy konferentsii (Sakharovskiye chteniya 2017 "Ekologicheskiye problemy XXI veka")* [Materials of the International Scientific Conference (Sakharov Readings 2017: "Enviromental Problems of the XXI century")]. Minsk, 2017, pp. 223–224 (In Russian).

Информация об авторах

Блинова Евгения Александровна – аспирант, 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

Урбанович Павел Павлович – доктор технических наук, профессор кафедры информационных систем и технологий. Белорусский государственный технологический университет (220006, г. Минск, ул. Свердлова, 13а, Республика Беларусь). E-mail: pav.urb@yandex.by

Information about the authors

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

Urbanovich Pavel Pavlovich – DSc (Engineering), Professor, Professor, the Department of Information Systems and Technologies. Belarusian State Technological University (13a, Sverdlova str., 220006, Minsk, Republic of Belarus). E-mail: pav.urb@yandex.by

Поступила 27.11.2017