Ю. Н. Мельников, Ю. А. Колошеин

ВОЗМОЖНОСТИ СТЕГАНОГРАФИЧЕСКОГО СОКРЫТИЯ ИНФОРМАЦИИ В ТЕКСТОВЫХ ФАЙЛАХ-КОНТЕЙНЕРАХ

Описаны угрозы, которые может нести в себе текстовый файл, отформатированный специальным образом. Представлены результаты исследований предельных возможностей сокрытия информации в текстовых файлах-контейнерах.

В настоящее время практически на любом предприятии имеет место передача текстовых документов как по внутренним каналам предприятия, так и в рамках информационного обмена с контрагентами. Способы сокрытия информации в текстовых документах известны человечеству еще с древних времен [1, 10, 11]. Таким образом, текстовые файлы несут в себе потенциальную угрозу утечки конфиденциальной информации. Особое место занимают тексты, отформатированные определенным образом — выровненные по обоим краям при помощи добавления пробелов между словами (в настоящее время подобным образом можно отформатировать текст в текстовых редакторах *vim* и *emac*, разработанных под *Unix*-подобные операционные системы (*Linux*, *Solaris* и т. п.); также в сети *Интернет* довольно много электронных книг, отформатированных подобным образом). До сих пор считается, что если информация встроена за счет незначительного изменения расстановки пробелов внутри строк текста, то практически невозможно обнаружить факт наличия встроенной в текст информации.

Цель настоящих исследований — подтвердить либо опровергнуть это мнение, дав соответствующие количественные характеристики (главный вопрос настоящей работы).

Столь значительное внимание именно текстовым файлам было уделено потому, что обмен текстовыми файлами носит наиболее массовый характер при обмене данными практически на любом предприятии.

Основные определения

Слово «*стеганография*» имеет греческие корни и буквально означает «тайнопись», которая осуществляется самыми различными способами. Общей чертой этих способов является то, что скрываемое сообщение встраивается в некоторый безобидный, не привлекающий внимание объект. Затем этот объект открыто транспортируется адресату [10].

Контейнер — любая информация, предназначенная для сокрытия тайных сообщений. Пустой контейне р — контейнер без встроенного сообщения, заполненный контейнер, или *стего* — контейнер, содержащий встроенную информацию [6, 15].

Встроенное (скрытое) сообщение — сообщение, встраиваемое в контейнер [6, 15].

Стеганографический канал (открытый канал) — канал передачи стего.

Стегоанализ — научное направление, в рамках которого изучаются вопросы обнаружения и извлечения скрытых сообщений.

Стегоаналитик (наблюдатель, атакующий) — объект (устройство, программное обеспечение или человек), осуществляющий стегоанализ.

Обобщенная модель системы передачи скрытых сообщений приведена на рис. 1.

Отметим, что некоторые алгоритмы встраивания скрываемой информации, например, [1—5], помимо контейнера и самой скрываемой информации, используют дополнительные данные, которые должны быть известны как отправителю, так и получателю, но ни в коем случае не должны стать известны стегоаналитику. На рис. 1 эти дополнительные данные названы *секретной кодовой комбинацией*.

Моноширинный текст — текст, выровненный по левому и правому краям при помощи пробелов так, что длина строки остается постоянной, за исключением, быть может, первой и последней строк абзаца.

Стойкость к обнаружению (вычислительная) — признак, позволяющий судить, насколько сложно обнаружить факт наличия скрытого сообщения в сообщении, передаваемом по стеганографическому каналу.

Рис. 1. Обобщенная модель системы передачи скрытых сообщений

Несанкционированное извлечение информации — извлечение стегоаналитиком скрытой информации внутри контейнера, передаваемого по стеганографическому каналу.

Исторический обзор

Если обратиться к истории, то первым «научным трудом» по стеганографии стала книга монаха Тритемиуса (1462—1516), получившая название «Steganographia» [13], в ней было собрано большое количество методов сокрытия тайных сообщений, в том числе, в тексте. Фактически это была первая попытка систематизировать в одном труде знания, накопленные в данной области.

В 1996 году состоялась первая международная конференция [14], на которой была предпринята первая попытка, в частности, выработать единую терминологию. В том же году в IBM Systems Journal вышла статья W. Bender «Techniques for Data Hiding» [15], в которой авторы указали области применения стеганографии, качественную зависимость стойкости к обнаружению от количества встроенной информации, а также описали несколько способов встраивания информации в текст; дальнейшие исследования W. Bender и его соавторов в этом направлении приведены в [16], вышедшей в свет в 2000 году.

Среди русскоязычных работ по стеганографии приведем [6], [10], [17].

Изучая литературу, авторам настоящей статьи не удалось обнаружить описаний стеганографически стойких алгоритмов сокрытия информации в текстовый файл в открытой литературе, а также результатов исследований стойкости текстовых файлов к стегоанализу. Авторы надеются, что работы [1—5], а также настоящая статья восполнят этот пробел.

Математическая модель

Кратко представим математическую формулировку задачи вычисления максимальной емкости текста (файла-контейнера), выровненного по левому и правому краям при помощи пробелов [3], т. е. определения верхней границы $\overline{C}$.

Пусть текст T моноширинный, содержит N строк.

Пусть появление 0 и 1 в кодовой последовательности, которую мы рассматриваем в качестве скрываемой внутри контейнера, равновероятно (мера Хардли [7]).

Пусть манипуляции с расстановкой пробелов в строке ограничены только этой строкой.

Пусть строка текста i содержит n_i интервалов между словами, а также m_i избыточных пробелов. Задача определения количества информации K_i, которая может содержаться в строке i, сводится к определению количества способов S_i, которыми можно разместить m_i избыточных пробелов в n_i интервалах.

$$S_i = C^{m_i}_{n_i-1+m_i} = \frac{(n_i-1+m_i)!}{m_i!(n_i-1)!} \qquad (1)$$

$$K_i = [\log_2 S_i] \qquad (2) \qquad [7]$$

$$K_T = \sum_{i=1}^{N} K_i \qquad (3)$$

В (3) K_T — это количество встраиваемой в файл-контейнер информации, которую может содержать в себе текстовый файл T (на основе допущения).

Необходимо отметить, что практические методы, известные к настоящему моменту, используют далеко не все возможные комбинации расстановки пробелов в интервалах строки. Это обусловлено стремлением как можно меньше исказить исходное форматирование текста [8].

$\overline{C} = K_T$ — это верхняя оценка количества информации, которая может быть скрыта в тексте T.

$\underline{C} = 0$ — это нижняя оценка количества информации, которая может быть скрыта в тексте T. Это обусловлено тем, что текст должен иметь избыточные пробелы, с помощью которых текст выравнивается по ширине [1—4], иными словами, не каждый текст удовлетворяет этим требованиям.

Пусть C — количество информации, которое можно скрыть в тексте любым методом из класса, тогда

$$\underline{C} < C < \overline{C}. \qquad (4)$$

Алгоритмы встраивания информации

Далее приведены 2 алгоритма заполнения контейнера информацией, которые примечательны тем, что знания только самих алгоритмов для достоверного извлечения информации не достаточно, необходимо знать дополнительные данные, которые известны отправителю и получателю. Алгоритм, подробно описанный в [1] (далее — А1), интересен тем, что *не вносит* значительных изменений в исходное форматирование строки, однако C_T для данного алгоритма значительно меньше $\overline{C}$. Другой алгоритм (далее —А2, описан ниже) знаменателен тем, что $C_T \approx \overline{C}$, однако исходное форматирование строки может быть сильно искажено.

Блок-схема алгоритма А1 приведена на рис. 2.

Для встраивания информации с помощью А1 подходит любой моноширинный текст, содержащий хотя бы одну строку, такую, что в каждом интервале содержится не менее 2 пробелов.

Отметим, что некоторые текстовые редакторы оставляют последнюю строку абзаца моноширинного текста выровненной по левому краю, т. е. между словами ровно один пробел.

Первая строка абзаца может иметь отступ слева, состоящий из нескольких пробелов либо знака табуляции.

Правило встраивания информации: ноль кодируется нечетным числом пробелов, а единица — четным.

Особенности: в каждой строке присутствует *служебный интервал*, позволяющий выровнять строку после встраивания информации с точностью до одного знака; местоположение служебного интервала определяется при помощи ГСПЧ (криптографически стойкого [11]); параметры ГСПЧ — та самая дополнительная информация, которая известна отправителю и получателю, но не должна быть известна стегоаналитику.

Стойкость А1. Пусть текст T содержит n строк, причем в строке i m_i интервалов. Тогда вероятность обнаружения местоположения служебного интервала строки i $P_{O_i} = \frac{1}{m_i}$. Заметим, что вероятности обнаружения местоположения служебного интервала в строках i и j (для $i \neq j$) независимые. Следовательно, вероятность обнаружения местоположения служебного интервала в строке n при условии, что во всех $1 \ldots (n\text{-}1)$ строках служебный интервал обнаружен, т. е. вероятность несанкционированного извлечения информации $P_{НИ} = \prod_{i=1}^{n} P_{O_i} = \prod_{i=1}^{n} \frac{1}{m_i}$.

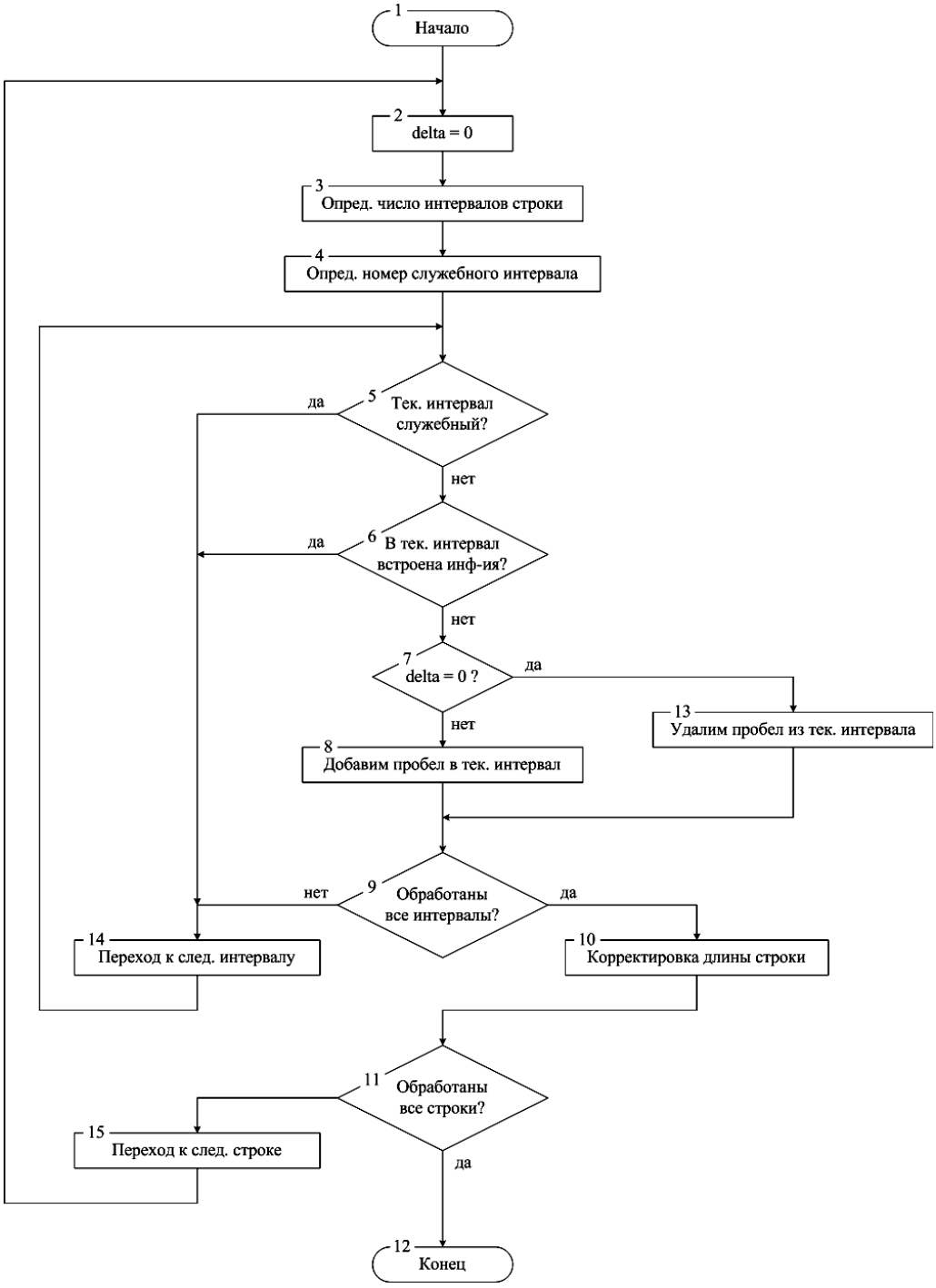

Рис. 2. Блок-схема алгоритма A1

Пример. Пусть текст T следующий:

```
Проблема защиты информации становится острее с каждым годом. Если на
заре Интернета и глобальных сетей вопросам безопасности уделяли внимание
в основном военные организации и крупные корпорации, то сейчас хакеры
представляют серьёзную угрозу для интернет-банков.
```

В него можно встроить 23 бита информации (по числу пробелов, за исключением служебных; последняя строка не участвует). Пусть это будет следующая информация: 0100 0000 0100 0011 1010 000. Пусть служебным интервалом для всех строк будет последний интервал.

После встраивания информации текст T будет выглядеть так:

```
Проблема   защиты   информации   становится   острее  с
каждым  годом. Если  на
заре  Интернета   и   глобальных  сетей   вопросам
безопасности  уделяли  внимание
в   основном   военные   организации   и  крупные
корпорации, то  сейчас хакеры
представляют серьёзную угрозу для интернет-банков.
```

Для того чтобы извлечь информацию, встроенную в текст T, достаточно проанализировать длину интервалов строк за исключением служебных. Наиболее наглядно этот процесс поясняет рис. 3.

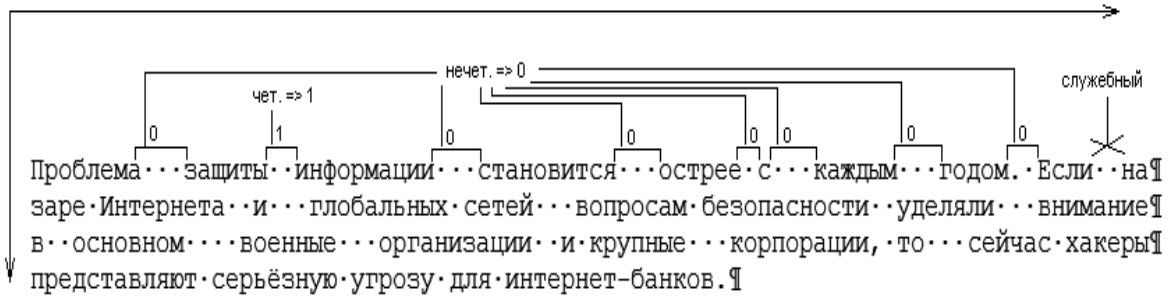

Рис. 3. Извлечение информации, встроенной согласно А1

В результате обработки первой строки получили: 0100 0000

Упрощенная блок-схема алгоритма А2 приведена на рис. 4.

Правило встраивания информации. Генерируется таблица вариантов размещений избыточных пробелов в интервалах строки, после чего каждому варианту ставится в соответствие кодовая комбинация. Отметим, что вариантов размещения пробелов в интервалах будет больше, чем кодовых комбинаций — это обусловлено формулами (1) и (2), следовательно, часть вариантов размещений останется неиспользованной.

Пример такой таблицы (для строки, содержащей 3 интервала и 2 избыточных пробела) — табл. 1.

Таблица 1.

Варианты размещения избыточных пробелов в интервалах				Кодовая комбинация
№ п/п	Интер-вал 1	Интер-вал 2	Интер-вал 3	
1	OO			00
2	O	O		01
3	O		O	10
4		OO		11
5		O	O	-
6			OO	-

Особенности. Для строки i часть вариантов размещений k_i останется неиспользованной. Для того чтобы определить, какие именно (для данной строки!), используем ГСПЧ (криптографически

стойкий [11]); параметры ГСПЧ — та самая дополнительная информация, которая известна отправителю и получателю, но не должна быть известна стегоаналитику.

$$k_i = S_i - 2^{K_i} \qquad (7)$$

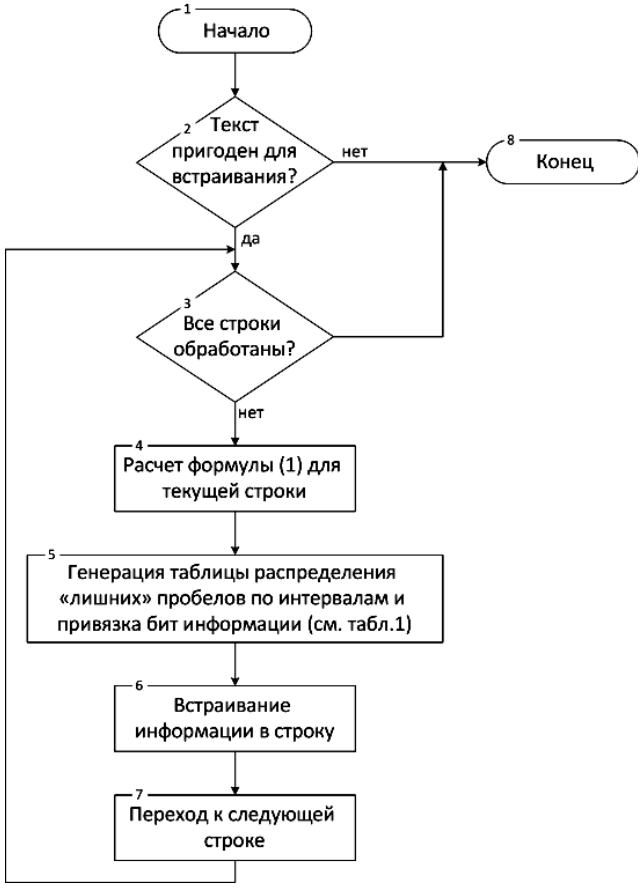

Рис. 4. Блок-схема алгоритма A2

Стойкость. Вероятность несанкционированного извлечения информации для всего текста $P_{НИ} = \prod_{i=1}^{n} P_{O_i}$, где P_{O_i} — вероятность обнаружения информации в строке i (аналогично A1). Однако расчет P_{O_i} существенно усложняется для случая, когда $k_i \geq 2$.

Пример. Пусть текст T следующий:

```
EfiColor is a trademark of Electronics for
Imaging
Corporation. Hewlett-Packard is a registered
trademark and
LaserJet trademark and DisplayWrite are
trademarks of
International Business Machines Corporation.
```

Рассмотрим вторую строку. В ней 6 интервалов и 5 избыточных пробелов. Запишем варианты размещения избыточных пробелов, поставив им в соответствие кодовые комбинации. Отметим, что из (1) и (2)

— вариантов размещения избыточных пробелов — 210,

— бит информации, которые можно встроить — 7.

Таким образом, из 210 вариантов размещений задействованными окажутся 128. Соответствие вариантов размещения кодовым комбинациям приведено в табл. 2.

Таблица 2.

Варианты размещения «лишних» пробелов в интервалах							Биты информации
№ п\п	1	2	3	4	5	6	Кодовая комбинация
1	ooooo	–	–	–	–	–	0000000
2	oooo	o	–	–	–	–	0000001
…							
64	ooo	o	–	–	o	–	0111111
…							
128	oo	–	oo	o	–	–	1111111
129	0	0	00	0	–	–	–
…							
210	–	–	–	–	–	ooooo	–

Встроим в строку кодовую комбинацию 0111111. После встраивания информации строка 2 будет выглядеть следующим образом:

Corporation. ·¸¸¸Htwlett - Packard ·¸is · a · registered ·¸trademark · and¶

Извлечение информации сводится к сопоставлению варианта размещения кодовой комбинации на основании таблицы. В нашем случае это табл. 2.

Алгоритмы анализа параметров текста сводятся к вычислению (1)—(7), поэтому их блок-схемы не приводятся.

Результаты исследований

Были описаны угрозы, которые может нести в себе текстовый файл, отформатированный специальным образом.

Описана математическая модель текста, позволяющая вычислить максимальное количество информации, которое теоретически может быть встроено в контейнер (верхняя оценка).

Описаны также два алгоритма заполнения контейнера информацией (A1 и A2), которые примечательны тем, что знания только самих алгоритмов для достоверного извлечения информации недостаточно, необходимо знать дополнительные данные, которые известны отправителю и получателю. A1 интересен тем, что *не вносит* значительных изменений в исходное форматирование строки, однако C_T для данного алгоритма значительно меньше $\overline{C}$. A2 интересен тем, что $C_T \approx \overline{C}$, однако исходное форматирование строки может быть сильно искажено.

СПИСОК ЛИТЕРАТУРЫ

1. *Колошеин Ю. А.* Разработка алгоритма стеганографического сокрытия защищаемой информации в текстовом файле // Доклады IX Международной научно-практической конференции «Стратегия развития пищевой промышленности». Вып. 8. М.: МГТА, 2003. С. 88–91.
2. *Колошеин Ю. А.* Стеганографическое сокрытие информации в текстовом файле при работе в вычислительных сетях // Доклады международной конференции «Информационные средства и технологии». Т. 3. М.: МЭИ (ТУ), 2003. С. 170–173.
3. *Колошеин Ю. А.* Модификация стеганографического алгоритма, основанного на замене символов контейнера // Тезисы докладов 12-й международной научно-технической конференции студентов и аспирантов. М.: МЭИ (ТУ), 2006. С. 320.
4. *Колошеин Ю. А., Мельников Ю. Н.* Возможности сокрытия банковской информации в текстовых файлах // Банковские технологии. 2003. № 11 (95). С. 32–34.
5. *Колошеин Ю. А., Мельников Ю. Н.* Возможности сокрытия банковской информации в текстовых файлах // Банковские технологии. 2003. № 12 (96). С. 32–34.
6. *Генне О. В.* Основные положения стеганографии // Защита информации. Конфидент. 2000. № 3. С. 20–24.
7. *Дмитриев В. И.* Прикладная теория информации. Учебник для студентов вузов по специальности «Автоматизированные системы обработки информации и управления». М.: Высшая школа, 1989.
8. *Колошеин Ю. А., Мельников Ю. Н.* Возможности обнаружения банковской информации в текстовых файлах // Банковские технологии. 2006. № 5. С. 65–68.
9. *Гмурман В. Е.* Теория вероятностей и математическая статистика: Учеб. пособие. 13-е изд., перераб. М.: Высшее образование, 2006. – (Основы наук).
10. *Грибунин В. Г., Оков И. Н., Туринцев И. В.* Цифровая стеганография. М.: СОЛОН-Пресс, 2002.
11. *Шнайер Б.* Прикладная криптография. Протоколы, алгоритмы, исходные тексты на языке Си. М.: Триумф, 2002.
12. *Карташов Д. В., Чижухин Г. Н.* Текстовая стеганография // Труды научно-технической конференции «Безопасность информационных технологий». Т. 4. Пенза, 2003. С. 19–21.
13. *Trithemius Johannes*. Steganographia. Frankfurt, 1606. http://www.esotericarchives.com/tritheim/stegano.htm.
14. *Anderson R., ed.* // Proc. Int. Workshop on Information Hiding: Lecture Notes in Computer Science. Springer-Verlag, Cambridge, 1996.
15. *Bender W., Gruhl D., Morimoto N., Lu A.* Techniques for Data Hiding // IBM Systems Journal 35, nos 3&4, 1996.
16. *Bender et all.* Applications for data hiding // IBM Systems Journal. Vol. 39, nos 3&4, 2000.
17. *Мельников Ю. Н., Баршак А. Д., Егоров П. Е.* Сокрытие банковской информации нестандартными способами // Банковские технологии. 2001. № 9. С. 25–27.