УДК 003.26:004.056.5

МЕТОД ШАБЛОННОГО ПРИХОВУВАННЯ ДАНИХ У ВЕКТОРНІ ЗОБРАЖЕННЯ

Владислав Ковтун, Олексій Кінзерявий, Олександр Стокіпний

В роботі авторами запропоновано новий стеганографічний метод шаблонного приховування інформації в структуру векторного зображення. Приховування даних за запропонованим методом здійснюється шляхом поступового розбиття кривих Без'є на візуально однакові сукупності сегментів при використанні наперед визначеної таблиці співвідношень різних значень елементів шаблону із різними кроками побудови кривих Без'є. Проведено експериментальне дослідження програмної реалізації запропонованого методу з приховування інформації у векторні зображення формату SVG. Отримані результати експерименту були порівняні з результатами існуючого побітового методу приховування інформації у криві Без'є. Запропонований метод показав виграш (більш ніж в 2-а рази) в зменшенні розмірів одержуваного стеганоконтейнера та часу необхідного для приховування даних в структуру SVG зображення.

Ключові слова: *захист інформації, стеганографія, векторні зображення, метод шаблонного приховування даних, метод побітового приховування даних, SVG зображення, криві Без'є, алгоритм де Кастельжо.*

Вступ. З входженням людства в еру інформаційного суспільства, інформація відіграє все більшу роль у житті людини, що надало поштовх до активного розвитку засобів захисту інформації. Серед відомих і добре зарекомендованих методів захисту інформації, що досить активно розвивається, є стеганографія. Основна відміна стеганографії від інших методів захисту інформації, полягає саме у приховуванні факту існування секретного повідомлення в іншому не примітному об'єкті – контейнері. Одним з найбільш використовуваних контейнерів є зображення, які широко використовуються в мультимедійних технологіях, поліграфічній діяльності, веб-просторі, електронних документах, тощо. Зображення бувають статичного та динамічного типів, та поділяються за видом подання на растрові, фрактальні і векторні ($2D$ та $3D$) [1-3]. В роботі [4] запропоновано метод побітового приховування даних в структуру векторного зображення шляхом подання векторних фігур у вигляді послідовності кривих Без'є, з подальшим їх поділом на візуально однакові сукупності сегментів з секретною інформацією. Даний метод завдяки властивостям векторної графіки та кривих Без'є забезпечує захист приховуваних даних від активних атак стеганоаналізу на основі афінних перетворень. Однак, недоліком методу [4] є значне збільшення розмірів одержуваного стеганоконтейнера (контейнера-результата) внаслідок збільшення кількості опорних точок необхідних для задання послідовностей кривих Без'є з секретною інформацією. Дослідження спрямовані на підвищення вмістимості приховуваної інформації та зменшення розмірів одержуваного стеганоконтейнера за методом [4] дозволить забезпечити подальший розвиток стеганографічних методів, що використовують векторні зображення у якості контейнера. Актуальність цієї науково-технічної задачі не викликає сумнівів.

Мета роботи полягає у підвищенні ефективності приховування інформації в структуру векторного зображення шляхом розробки методу шаблонного приховування даних з визначеною таблицею співвідношень різних значень елементів шаблону із різними кроками побудови кривих Без'є.

Основна частина. *Метод побітового приховування даних у векторні зображення.* Коротко опишемо метод побітового приховування даних у векторні зображення [4]. Приховування здійснюється в частині векторного зображення, що містить криві Без'є. Для цього обирається крива або група кривих Без'є, які поступово діляться на сегменти, таким чином, щоб візуально не спотворити зображення. Біти секретного повідомлення використовуються для отримання проміжних сегментів, які утворюються у результаті поділу основної кривої або її частин.

Нагадаємо, що крива Без'є задається наступною формулою [5, 6]:

$$B(t)=\sum_{i=0}^{n} b_{i,n}(t)P_i,\ t=t+\Delta t,\ t\in[0,1],$$

де P_i – опорні точки, $i\in\overline{0,n}$, i – індекс опорних точок, n – порядок визначальної функції базису Бернштейна і сегментів поліноміальної кривої, t – параметр побудови кривої, Δt – крок зміни параметра t, $b_{i,n}(t)$ – поліноми Бернштейна. Кількість опорних точок P_i чітко визначає порядок побудови кривої Без'є, де перша та остання опорні точки належать даній кривій, а решта задають напрям руху побудови кривої. Криві Без'є

володіють наступними властивостями, на основі яких базується метод [4]:

1. будь-яку криву Без'є можна розбити та подати у вигляді візуально однакової сукупності сегментів, де кожний сегмент кривої являтиметься кривою Без'є того ж порядку що і початкова крива;

2. будь-яка крива Без'є є інваріанта відносно афінних перетворень, тобто над кривою Без'є можна здійснити афінне перетворення, при якому положення та відстані між її опорними точками зберігатимуться, але змінюватимуться лише їх координати.

Розбиття кривої Без'є на сегменти здійснюється за алгоритмом де Кастельжо, що описується наступним чином [7]: координати точки $B(t)$ на кривій Без'є при певному значенні t рівні координатам точки P_0^x ($B(t)=P_0^x$, $x=n+1$ – кількість опорних точок кривої), що обчислюється за наступною рекурсивною формулою:

$$P_i^r=(1-t)P_i^{r-1}+tP_{i+1}^{r-1},$$

де $r\in\overline{1,x}$, $i\in\overline{0,x-r}$, $P_i^0=P_i$, а початкові значення P_i^0 – координати опорних точок P_i.

Приховування секретного повідомлення $a=\{a_1,a_2,...,a_{h-1},a_h\}$, $a_i\in\{0,1\}$, $i\in\overline{1,h}$, a_i – біт секретного повідомлення, h – кількість біт повідомлення a, за методом [4] проходитиме за наступним алгоритмом:

1. Визначається секретний параметр (стеганоключ) Δt кроку зміни t, на основі якого будується кожна точка кривої Без'є. Причому, визначений крок Δt не повинен перевищувати максимально допустиме значення $\max\Delta t=1/h$, бо вразі вибору значення $\Delta t>\max\Delta t$ може відбутися вихід параметра t за його допустимі межі.

2. Послідовність a ділиться на частини $a=\{a_1^1,...,a_z^1,a_1^2,...,a_z^2,...,a_1^j,...,a_z^j\}$, $a_i^j\in\{0,1\}$, $i\in\overline{1,z}$, $z=h/m$, $j\in\overline{1,m}$, де m – загальна кількість кривих Без'є векторного зображення в які буде приховуватись секретне повідомлення, а $a^j=\{a_1^j,...,a_z^j\}$ – послідовність біт довжиною z, кожна з яких заноситься в D_j криву Без'є при її поділі на сегменти. Отримана послідовність сегментів позначається як $D_{P_v}^w$, де w – індекс послідовності створених сегментів кривої D_j, $w\in N$, а P_v – координати опорних точок, $v\in\overline{0,3}$. Причому, кожний сегмент з послідовності сегментів $D_{P_v}^w$ є також кривою Без'є того ж порядку, що і початкова крива D_j.

3. Виконується приховування кожної послідовності a_i^j, $i\in\overline{1,z}$ секретного повідомлення в криву D_j, $j\in\overline{1,m}$ шляхом розбиття її на сегменти при різних значеннях параметра $t=t+\Delta t$:

a. Якщо на певному кроці t приховуватиметься біт $a_i^j=0$, то крива Без'є не ділиться на сегменти, а відбувається перехід до наступного значення приховуваної послідовності a_{i+1}^j та наступного кроку t.

b. Якщо на певному кроці t приховуватиметься біт $a_i^j=1$, то в даній точці виконується розбиття кривої Без'є на два сегмента $D_{P_v}^w$ і $D_{P_v}^{w+1}$ за алгоритмом де Кастельжо. Подальше внесення наступного значення приховуваної послідовності a_{i+1}^j відбуватиметься при наступному кроці t в отриманий другий сегмент розбиття $D_{P_v}^{w+1}$. Кожне розбиття кривої Без'є призводить до збільшення кількості сегментів на $w=w+1$.

c. Здійснивши приховування послідовності a_i^j у D_j криву, вона замінюється на отриману послідовність сегментів $D_{P_v}^w$. Причому, візуально крива D_j не буде відрізнятися від послідовності сегментів $D_{P_v}^w: D_j=D_{P_v}^0\cup D_{P_v}^1\cup...\cup D_{P_v}^w$.

Розглядуваний алгоритм згідно методу [4] було програмно реалізовано та експериментально досліджено. Умови проведення експерименту приведені в табл. 1.

Таблиця 1

Апаратні та програмні засоби проведення експерименту

Назва	Опис характеристики
Процесор	*AMD A10-4600M 2.3GHz*
Оперативна пам'ять	*4Gb*
Операційна система	*Windows 7 SP1 x86-64*
Середа розробки	*Visual Studio 2012*
Мова реалізації	*Visual C++*

Приховування даних здійснювалося в структурний елемент *Path* векторного зображення у *SVG* форматі, а саме в кубічні криві (типу *Curveto*, що задаються командами *C* та *c*) [8]. Внесення даних різного розміру (від 100 байт до 1000 байт) в *SVG* зображення відбувалося тільки в одну криву Без'є з наступними ключовими параметрами:

– $CP_1 = 20$, де CP_1 – максимально допустима кількість десяткових знаків дробової частини координат опорних точок кривих Без'є;

– $CP_2 = 6$, де CP_2 – допустима мінімальна довжини кривої Без'є відносно відстаней між її опорними точками;

– $CP_3 = 6$, де CP_3 – максимально допустима похибка в кількості останніх десяткових знаків дробової частини координат опорних точок, що виникає при зворотному процесі відтворення початкової кривої D_j з послідовності сегментів $D_{P_v}^w$.

Результати експерименту з середніми значеннями часу виконання приховування і відтворення інформації наведені в табл. 2.

Таблиця 2

Результати експериментального дослідження методу побітового приховування даних

№	Розмір прихованої інформації, байт	Час приховування, с	Час відтворення, с	Розмір контейнера «до», КБайт	Розмір контейнера «після», КБайт	Збільшення розміру контейнера «після» відносно контейнера «до», %
1	100	12,16	25,77	45,28	98,90	118,44
2	200	23,93	50,23	45,28	152,60	237,05
3	300	46,10	59,50	45,28	208,28	360,03
4	400	59,71	81,14	45,28	264,49	484,19
5	500	77,93	99,16	45,28	318,61	603,71
6	600	92,81	119,85	45,28	374,24	726,58
7	700	101,74	140,61	45,28	428,57	846,57
8	800	123,37	163,28	45,28	482,24	965,13
9	900	128,66	183,88	45,28	534,61	1080,80
10	1000	143,81	202,46	45,28	588,31	1199,41

Подальше зменшення розміру одержаного стеганоконтейнера можна досягти завдяки його стисненню за допомогою програм архіваторів. В табл. 3 наведено дослідження стосовно стиснення стеганоконтейнера за допомогою *ZIP* архіватора.

Таблиця 3

Результати стиснення стеганоконтейнерів отриманих за побітовим методом

№	Розмір контейнера «після», КБайт	Розмір стиснутого контейнера «після», КБайт	Зменшення розміру стиснутого контейнера відносно контейнера «після», %	Збільшення розміру стиснутого контейнера «після» відносно контейнера «до», %
1	98,90	45,41	54,09	0,29
2	152,60	69,21	54,65	52,86
3	208,28	93,29	55,21	106,04
4	264,49	116,82	55,83	158,02
5	318,61	139,40	56,25	207,89
6	374,24	161,08	56,96	255,77
7	428,57	186,11	56,57	311,07
8	482,24	206,76	57,13	356,66
9	534,61	226,26	57,68	399,73
10	588,31	244,77	58,39	440,63

З отриманих результатів можна зробити висновок, що при приховуванні даних в структурі *SVG* зображення побітовим методом суттєво збільшується розмір стеганоконтейнера, навіть після стиснення *ZIP* архіватором. Для зменшення розмірів одержуваного стеганоконтейнера, авторами запропоновано шаблонний метод приховування даних, який буде розглянуто нижче.

Метод шаблонного приховування даних у векторні зображення. Для зменшення розмірів стеганоконтейнера, пропонується метод шаблонного приховування даних у векторні зображення, який оперує вже не бітами інформації, а цілими блоками (серіями) біт. Основною відмінністю шаблонного методу, є можливість визначати наперед різні кроки зміни параметра t відповідно до кожного значення елемента шаблону, на відміну від побітового, в якому для приховування одного біту даних використовувався сталий крок Δt зміни параметра t. В свою, чергу це дозволить приховувати цілий блок бітів лише за одне розбиття кривої Без'є. Таблиця значень шаблону буде відігравати роль стеганок-

люча, необхідного для приховування та відтворення даних з стеганоконтейнера. Значення елементів шаблону, де кожному її елементу ставитиметься у відповідність свій крок зміни параметра t, будуть задаватися наступним співвідношенням:

$$TV_l^k = T\Delta t^k,$$

де k – індекс значень елементів шаблону, $k \in \overline{1, 2^l}$, TV_l^k – значення одного елементу шаблону, l – кількість біт одного значення елементу шаблону, $T\Delta t^k$ – один крок зміни параметра t.

Для співставлення значенням шаблону TV_l^k кроків $T\Delta t^k$ зміни параметра t потрібно дотримуватись виконання наступних вимог:

1. Визначення кроків $T\Delta t^k$ зміни параметра t не повинно перевищувати максимально допустимого $\max \Delta t = 1/g$, де g – сумарна кількість значень елементів шаблону в приховуваному повідомленні. Вразі вибору, хоч одного, значення $T\Delta t^k > \max \Delta t$ може відбутися вихід параметра t за його допустимі межі.

2. Значення кроків $T\Delta t^k$ зміни параметра t не повинні повторюватися між собою.

В табл. 4 наведено приклад співставлення кожному елементу TV_l^k, $l = 4$, що являється послідовністю з чотирьох біт, власного кроку $T\Delta t^k$ зміни параметра t.

Таблиця 4

Приклад співвідношення значень елементів шаблону з різними кроками побудови кривої Без'є

Індекс k	Шаблон TV_4^k, біт	Крок зміни $T\Delta t^k$
1	0000	0,00095
2	0001	0,0009
3	0010	0,00085
4	0011	0,0008
5	0100	0,00075
6	0101	0,0007
7	0110	0,00065
8	0111	0,0006
9	1000	0,00055
10	1001	0,0005
11	1010	0,00045
12	1011	0,0004
13	1100	0,00035
14	1101	0,0003
15	1110	0,00025
16	1111	0,0002

Приховування даних за шаблонним методом відбуватиметься за наступним алгоритмом:

1. Обраховується сумарна кількість значень елементів шаблону в приховуваному повідомленні g та максимально допустиме значення $\max \Delta t$ кроку зміни параметра t.

2. Встановлюється кожному елементу TV_l^k шаблону власний крок $T\Delta t^k$ зміни параметра t відповідно до описаних вище вимог.

3. Представлення приховуваного повідомлення a у вигляді послідовності блоків $a = \{a_1^z, a_2^z, ..., a_{m-1}^z, a_m^z\}$, $z = n/m$, де z – розмір блоку a_i^z, $i \in \overline{1, m}$, n – розмір приховуваного повідомлення, а m – загальна кількість кривих Без'є векторного зображення в які буде приховуватись секретне повідомлення, кожний з яких заноситься в D_i криву Без'є. Кожний блок повідомлення подається у вигляді $a_i^z = \{a_i^1, ..., a_i^q\}$, a_i^y – частина повідомлення з довжиною l біт, $y \in \overline{1, q}$, $z = q \cdot l$, q – кількість значень елементів шаблону в блоці. Отримана послідовність сегментів позначається як $D_{P_v}^w$, де w – індекс послідовності створених сегментів кривої D_i, $w \in N$, а P_v – координати опорних точок, $v \in \overline{0, 3}$.

4. Виконується приховування кожного блоку a_i^z секретного повідомлення в криву D_i, $i \in \overline{1, m}$ шляхом розбиття її на сегменти при різних значеннях параметра t:

a. Приховування кожного елементу шаблону TV_l^k з блоку a_i^z при певному значенні t відбуватиметься шляхом розбиття кривої Без'є на два сегмента $D_{P_v}^w$ і $D_{P_v}^{w+1}$ за алгоритмом де Кастельжо. Подальше внесення наступних елементів шаблону з блоку a_i^z буде відбуватися при наступному значенні $t = t + T\Delta t^k$, де значення $T\Delta t^k$ відповідає приховуваному елементу TV_l^k на даному кроці t, в отриманий другий сегмент $D_{P_v}^{w+1}$, що буде розпочинатися з координат точки розбиття початкової кривої. Кожне розбиття кривої Без'є призводить до збільшення кількості кривих Без'є в послідовності сегментів на $w = w + 1$.

b. Здійснивши приховування блоку послідовності a_i^z у D_i криву Без'є, вона змінюється на візуально однакову послідовність сегментів $D_{P_v}^w : D_i = D_{P_v}^0 \cup D_{P_v}^1 \cup ... \cup D_{P_v}^w$.

На основі шаблонного методу був проведений експеримент, аналогічний до експерименту для побітового методу, тільки з використанням наперед заданої таблиці кроків $T\Delta t^k$. Відповідно якій кожному елементу значення TV_l^k шаблону ставилося різне значення $T\Delta t^k$ кроку зміни параметра t.. Результати експерименту з використовуваними наступними параметрами $l = 8$, $k \in \overline{1, 256}$, $CP_1 = 20$, $CP_2 = 6$ та $CP_3 = 6$ наведені в табл. 5.

Таблиця 5

Результати експериментального дослідження методу шаблонного приховування даних

№	Розмір прихованої інформації, байт	Час приховування, с	Час відтворення, с	Розмір контейнера «до», КБайт	Розмір контейнера, «після», КБайт	Збільшення розміру контейнера «після» відносно контейнера «до», %
1	100	3,60	21,59	45,28	60,36	33,32
2	200	7,05	57,47	45,28	75,22	66,14
3	300	10,73	65,80	45,28	90,09	98,98
4	400	15,01	92,17	45,28	104,99	131,88
5	500	18,70	108,28	45,28	119,89	164,81
6	600	22,65	150,92	45,28	134,78	197,68
7	700	27,24	154,71	45,28	149,65	230,53
8	800	29,59	177,97	45,28	164,39	263,09
9	900	33,65	203,60	45,28	179,28	295,97
10	1000	38,42	228,60	45,28	194,10	328,72

Провівши стиснення одержаних стеганоконтейнерів *ZIP* архіватором отримаємо наступне (див. табл. 6):

Таблиця 6

Результати стиснення стеганоконтейнерів отриманих за шаблонним методом

№	Розмір контейнера «після», КБайт	Розмір стиснутого контейнера «після», КБайт	Зменшення розміру стиснутого контейнера відносно контейнера «після», %	Збільшення розміру стиснутого контейнера «після» відносно контейнера «до», %
1	60,36	27,63	54,23	-38,98
2	75,22	34,69	53,88	-23,38
3	90,09	41,65	53,77	-8,01
4	104,99	48,54	53,76	7,21
5	119,89	55,43	53,77	22,43
6	134,78	62,14	53,89	37,25
7	149,65	68,91	53,95	52,19
8	164,39	75,52	54,06	66,81
9	179,28	82,24	54,13	81,63
10	194,10	88,94	54,18	96,43

Результати порівняння розмірів стеганоконтейнерів, одержаних при застосуванні розглядуваних методів та стиснення їх *ZIP* архіватором, наведено на діаграмі рис. 1.

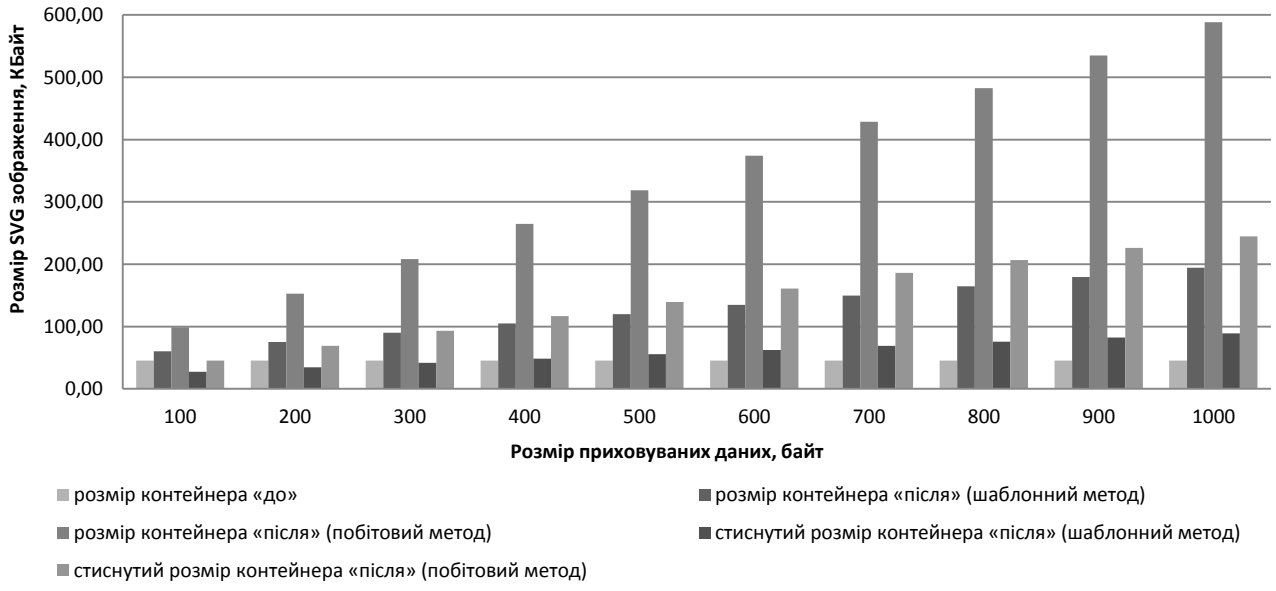

Рис. 1. Порівняння розмірів одержаних стеганоконтейнерів

Результати процентного порівняння збільшення розмірів стеганоконтейнерів, одержаних при застосуванні розглядуваних методів, наведено на діаграмі рис. 2.

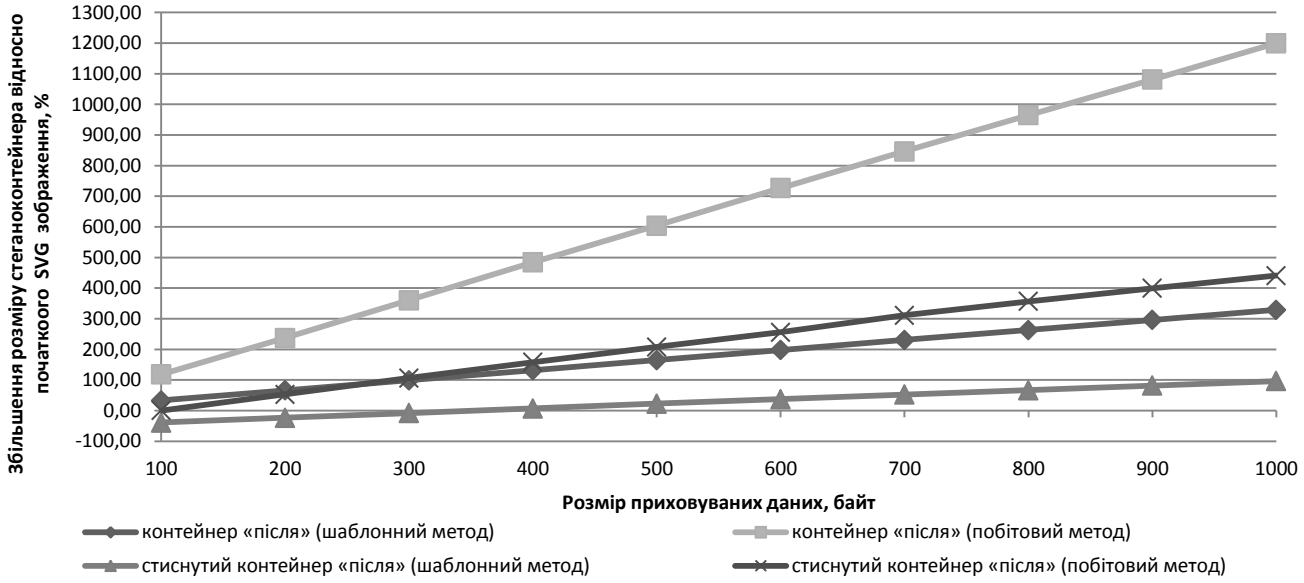

Рис. 2. Процентне збільшення розмірів одержаних стеганоконтейнерів відносно початкового контейнера

Результати порівняння часових затрат на приховування та вилучення даних за шаблонним та побітовим методів представлені на діаграмах рис. 3 та рис. 4.

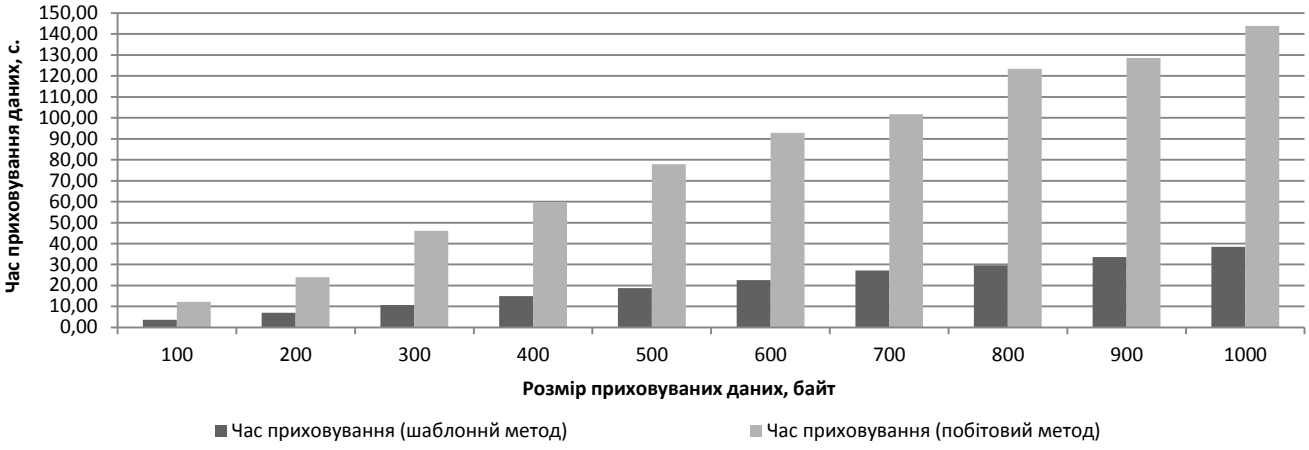

Рис. 3. Порівняння необхідних затрат часу на приховування даних в структуру *SVG* зображення

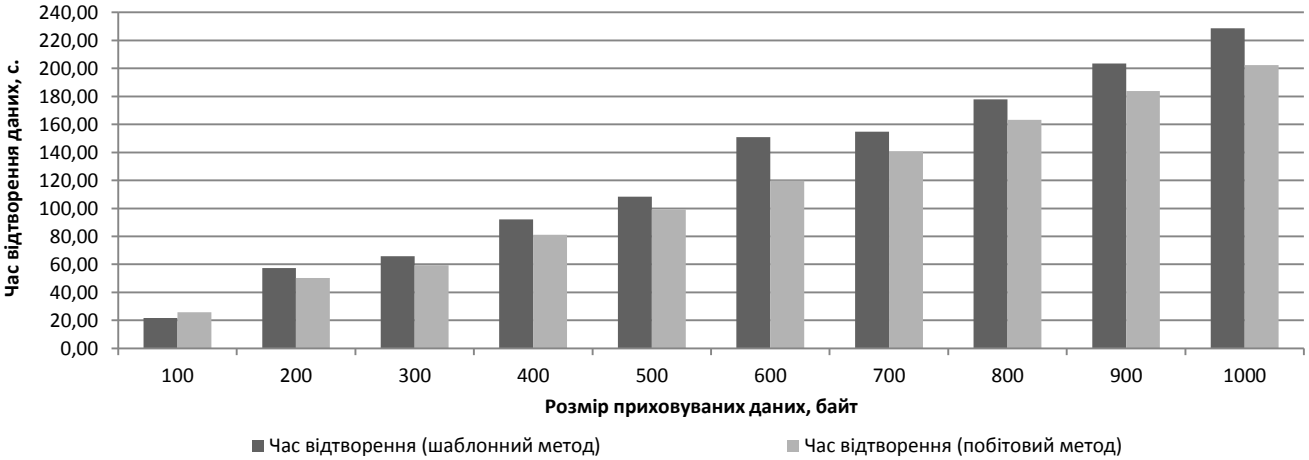

Рис. 4. Порівняння необхідних затрат часу на вилучення даних з стеганоконтейнера

Висновки. Для підвищення ефективності приховування інформації в структуру векторного зображення в роботі запропоновано шаблонний метод приховування даних. Експериментальне дослідження показало, що метод шаблонного приховування даних створює стеганоконтейнер більш ніж в 2-а рази меншого розміру та з меншими затратами часу на приховування інформації ніж побітовий метод, оскільки за одне розбиття кривої Без'є приховується цілий блок бітів секретного повідомлення. Однак, застосування шаблонного методу призводить до збільшення часу на вилучення приховуваних даних з стеганоконтейнера у порівнянні з побітовим методом, оскільки при зворотному процесі поступового відтворення початкової кривої з сукупності сегментів постійно потрібно перебирати всі значення $T\Delta t^k$ шаблону для відшукання необхідного значення, при якому буде здійснюватися одне об'єднання двох останніх сегментів послідовності.

За допомогою додаткового дослідження по стисненню одержуваних стеганоконтейнерів встановлено, що стеганоконтейнери на основі векторних зображень досить гарно піддаються стисненню ZIP архіватором, а отримуваний коефіцієнт стиснення становить більше 50%.

ЛИТЕРАТУРА

[1]. Маценко В.Г. Комп'ютерна графіка : [навчальний посібник] / В.Г. Маценко. – Ч. : Рута, 2009. – 343 с.

[2]. Копитова О.М. Курс лекцій з комп'ютерної графіки : [для студентів всіх форм навчання] / О.М. Копитова. – Д. : ДонНТУ, 2011. – 81 с.

[3]. Кінзерявий О.М. Систематизація сучасних методів комп'ютерної стеганографії / О.М. Кінзерявий, В.Ю. Ковтун, С.О. Гнатюк // науковий журнал «Безпека інформації». – Т.19, №3, –2013. – С. 209-217.

[4]. Кінзерявий О.М. Стеганографічний метод приховування даних у векторних зображеннях / О.М. Кінзерявий, В.Ю. Ковтун, С.О. Гнатюк, В.М. Кінзерявий // теоретичний і науково-практичний журнал «Вісник Інженерної академії України». – 2013. – №3-4. – С. 66-68.

[5]. Роджерс Д. Математические основы машинной графики : [пер. с англ.] / Д. Роджерс, Дж. Адамс. – М. : Мир, 2001. – 604 с.

[6]. Голованов Н.Н. Геометрическое моделирование / Н.Н. Голованов. – М. : издательство Физико-математической литературы, 2002. – 472 с.

[7]. Кунву Ли Основы САПР (CAO/CAM/CAE) / Ли Кунву. – СПб. : Питер, – 2004. – 560 с.

[8]. Дунаев В.В. HTML, скрипты и стили : [3-е издание] / В.В. Дунаев. – СПб. : БХВ-Петербург, 2011. – 816 с.

[9]. Dailey D. Building Web Applications with SVG / D. Dailey, J. Frost, D. Strazzullo. – O'Reilly Media, 2012. – 268 p.

REFERENCES

[1]. Matsenko V.G. (2009) "Computer Graphics", *Ch. : Ryta*, 343 p.

[2]. Kopytova O.M. (2011) "Lectures on Computer Graphics" : [for students of all learning], *D.: DonNTU*, 81 p.

[3]. Kinzeryavyy O.M., Kovtun V.Y., Gnatyuk S.O. (2013) "Systematization of modern methods of computer steganography", Information Security, VOL. 19 №3, pp. 209-217.

[4]. Kinzeryavyy O.M., Kovtun V.Y., Gnatyuk S.O., Kinzeryavyy V.M. (2013) "Steganography method of hiding data in vector images", Bulletin of Engineering Academy of Ukraine, №3-4. — pp. 66-68.

[5]. Rodjers D., Adams J. (2001) "Mathematical Foundations of Computer Graphics", *M. : Mir*, 604 p.

[6]. Golovanov N.N. (2002) "Geometric modeling", *M. : publishing house of physicomathematical literature*, 472 p.

[7]. Kunvu Lee (2004) "Basics CAD (CAO/CAM/CAE)", *St.P.:Peter*, 560 p.

[8]. Dunaev V.V. (2011) "HTML, scripts, and styles" : [3rd edition], *St.P. : BHV-Petersburg*, 816 p.

[9]. Dailey D., Frost J., Strazzullo D. (2012) "Building Web Applications with SVG", *O'Reilly Media*, 268 p.

МЕТОД ШАБЛОННОГО СОКРЫТИЯ ДАННЫХ В ВЕКТОРНЫЕ ИЗОБРАЖЕНИЯ

В роботе авторами предложен новый стеганографический метод шаблонного сокрытия информации в структуру векторного изображения. Сокрытие данных, согласно предложенному методу, осуществляется путём постепенного разбиения кривых Безье на визуально одинаковые совокупности сегментов при использовании предустановленной таблицы соотношений различных значений элементов шаблона с разными шагами построения кривых Безье. Проведено экспериментальное исследование программной реализации предложенного метода по сокрытию информации в векторные изображения формата SVG. Полученные результаты эксперимента были сравнены с результатами существующего побитового метода сокрытия информации в кривые Безье. Предложенный метод показал выигрыш (более чем в 2-а раза) в уменьшении размеров получаемого стеганоконтейнера и времени, необходимого для сокрытия данных в структуру SVG изображения.

Ключевые слова: защита информации, стеганография, векторные изображения, метод шаблонного сокрытия данных, метод побитового сокрытия данных, SVG изображения, кривые Безье, алгоритм де Кастельжо.

METHOD OF TEMPLATE HIDING DATA IN VECTOR IMAGES

In this work authors propose the new steganographic method of template hiding data in vector image structure. According to the proposed method, hiding data is performed by the gradually separation of Bezier curves into visually identical pluralities of segments with using preassigned correlation table of the different values of template elements with the different steps of Bezier curves structure. There was conducted the software implementation of proposed method of hiding information in vector images of SVG format. The obtained results of experiment were compared with the results of existing bitwise method of hiding data in Bezier curves. The proposed method showed the profit (in more than 2 times) in reduction of steganocontainer size and time, required for hiding data in SVG image structure.

Key words: information security, steganography, vector images, method of template hiding data, method of bitwise hiding data, SVG images, Bezier curves, algorithm de Casteljau.

Ковтун Владислав Юрійович, кандидат технічних наук, доцент, доцент кафедри безпеки інформаційних технологій Національного авіаційного університету.
E-mail: vladislav.kovtun@gmail.com.
Ковтун Владислав Юрьевич, кандидат технических наук, доцент, доцент кафедры безопасности информационных технологий Национального авиационного университета.
Vladislav Kovtun, Ph.D., docent of Academic Department of IT-Security, National Aviation University.

Кінзерявий Олексій Миколайович, аспірант кафедри безпеки інформаційних технологій Національного авіаційного університету.
E-mail: oleksiykinzeryavyy@gmail.com.
Кинзерявый Алексей Николаевич, аспирант кафедры безопасности информационных технологий Национального авиационного университета.
Kinzeryavyy Oleksiy, postgraduate student of Academic Department of IT-Security, National Aviation University.

Стокіпний Олександр Леонідович, кандидат технічних наук, доцент кафедри інформаційних систем факультету економічної інформатики, Харківського національного економічного університету ім. С. Кузнеця (ХНЕУ).
E-mail: a.stokipny@gmail.com
Стокипный Александр Леонидович, кандидат технических наук, доцент, доцент кафедры информационных систем факультета экономыческой информатики, Харьковского национального экономического университета им. С. Кузнеца (ХНЭУ).
Stokipnyy Oleksandr, Ph.D., docent of Academic Department of Information systems, Simon Kuznets Kharkiv National University of Economics.

УДК 004.056:007

МЕТОД ІНФОРМАЦІЙНО-АНАЛІТИЧНОЇ ПІДТРИМКИ УПРАВЛІННЯ ІНФОРМАЦІЙНОЮ БЕЗПЕКОЮ НА ОСНОВІ СИСТЕМНОГО ПІДХОДУ

Дмитро Домарєв

Об'єднання розрізнених засобів захисту та зусиль фахівців різних профілів для виконання стратегічних завдань забезпечення інформаційної безпеки (ІБ) України потребує інформаційно-аналітичної підтримки управління ІБ на основі системного підходу. Відсутність узгодженої обробки та зберігання оперативних задач, знань та ризиків ІБ в умовах неповноти інформації, а також відсутність застосування системного підходу в процесі управління ІБ зменшує адаптивність та мобільність систем інформаційної безпеки. Вдосконалено модель логічних і функціональних зв'язків між складовими системи управління інформаційною безпекою (СУІБ), в якій за рахунок надання множині складових «напрямки» змінної розмірності забезпечено гнучкість процесів аналізу, прогнозування та інформаційно-аналітичної підтримки прийняття рішень щодо забезпечення ІБ. Вперше розроблено модель даних СУІБ, в якій за рахунок структуризації даних за моделлю логічних і функціональних зв'язків між складовими СУІБ забезпечено узгоджену обробку та зберігання оперативних задач, знань та ризиків ІБ в умовах неповноти інформації. Вперше розроблено метод інформаційно-аналітичної підтримки управління ІБ, який за рахунок використання вдосконаленої моделі зв'язків між складовими СУІБ, розробленої моделі даних СУІБ та розробленої методики оцінки поточного стану ІБ забезпечує застосування принципів системного підходу в управлінні ІБ. Наведено приклад застосування розробленого методу для банківської системи України. Надано рекомендації щодо наукового та практичного використання розроблених моделей та методу.

***Ключові слова**: системний підхід до інформаційної безпеки, модель зв'язків між складовими СУІБ, модель даних управління інформаційною безпекою, метод управління інформаційною безпекою, система управління інформаційною безпекою, СУІБ.*