которая основана на модели интегрированного представления параметров вреда, существующих средств обеспечения данной сферы и метода анализа и оценки величины возможного ущерба как в условных (бальных) единицах, так и в стоимостной (денежной) величине ущерба. Система использует интегрированные базы данных, содержащие основные параметры и показатели средств обеспечения процедуры отнесения информации к государственной тайне, и имеет возможность автоматизированного формирования отчета по результатам проведенного оценивания.

Ключевые слова: разглашение государственной тайны, потеря материальных носителей секретной информации, система оценки ущерба национальной безопасности, сфера охраны государственной тайны, национальная безопасность.

EXPERIMENTAL RESEARCH SYSTEM EVALUATION OF DAMAGE TO NATIONAL SECURITY IN THE SPHERE OF STATE SECRETS

State expert on secrets when classification of information to state secrets is conducted justification and determination of possible damage to national security if its disclosure or loss of material carriers of this information. Existing calculate the damage in most based on a conditional (scoring) estimates of predicted value. The paper presents the algorithm of functioning and experimental research developed evaluation system of damage to national security in the sphere of state secrets, which is based on a model of integrated presentation parameters damage, existing means of this sphere and the method of analysis and estimation of possible damage to both conventional (ball) units and in value (monetary) value of loss. The system uses an integrated database containing basic parameters and performance means of the procedure to ensure classification of information to state secrets and has the capability automated reporting the results of the evaluation.

Index Terms: disclosure of state secrets, the loss material carriers of classified information, system evaluation of damage to national security, the sphere of state secrets, national security.

Дрейс Юрій Олександрович, кандидат технічних наук, доцент кафедри безпеки інформаційних і комунікаційних систем Житомирського військового інституту ім. С.П. Корольова Державного університету телекомунікацій.
E-mail: dr_yr_al@mail.ru
Дрейс Юрий Александрович, кандидат технических наук, доцент кафедры безопасности информационных и коммуникационных систем Житомирского военного института им. С.П. Королева Государственного университета телекоммуникаций.
Dreis Yurii, PhD in Eng., Associate Professor of Academic Department of Security Information and Communication Systems of the Zhytomyr Military Institute named after S.P. Koroleva of the State University of Telecommunication.

УДК 004.932 : 004.056.53

СПОСІБ СТЕГАНОГРАФІЇ ЗОБРАЖЕНЬ НА ОСНОВІ КОМПЛЕМЕНТАРНОГО ОБРАЗУ

Євгенія Сулема, Семен Широчин

Цифрові зображення при їх передачі через Інтернет або збереженні на серверах загального користування можуть легко потрапити у відкритий доступ. Якщо ці зображення мають характер особистої інформації, то задача їх простого та ефективного захисту стає для користувача актуальною. Крім того, для звичайного користувача важливим є час, необхідний для шифрування та дешифрування зображення. Існуючі засоби криптографічного захисту не орієнтовані на звичайного користувача, що потребує простої та швидкої процедури захисту графічних даних при їх передаванні через мережу у реальному часі. В статті пропонується стеганографічний спосіб захисту зображень, який ґрунтується на збереженні замість зображення його комплементарного образу та використанні в ролі ключа рандомізованого латинського квадрату. Розроблений метод забезпечує кращі часові показники процедури кодування та декодування зображення, що приховується, ніж аналогічні засоби, що використовують криптографію. Це досягається використанням паралельних обчислень. Даний спосіб може бути використаний в реальному часі для захисту особистих зображень користувача перед їх передаванням через мережу. Приховані зображення можуть бути збережені у форматі png або jpeg з ущільненням без втрат.

Ключові слова: *стеганографія, комплементарний образ, латинські квадрати*

Вступ. На сьогодні цифрові зображення складають левову частку інформації, що передається через комп'ютерні мережі та зберігається на серверах загального користування, причому більшість цифрових зображень має характер особистої інформації, що не підлягає поширенню без

згоди власника. Отже, постає задача пошуку простого та доступного широкому колу користувачів способу захисту цифрових зображень від несанкціонованого доступу. Таким способом може слугувати стеганографічний захист графічних даних, а саме стеганографія зображень [3], що використовує загальнодоступний графічний файл, так званий *контейнер*, в якому приховуються графічні дані, що не підлягають поширенню – *стегодані*. Стеганографічний захист є простим та доступним для звичайного користувача. Загальні принципи стеганографічного захисту є широко відомими, на відміну від конкретної стеганографічної схеми розміщення даних та алгоритму їх отримання. Тому створення нових способів стеганографічного захисту персональних даних користувача є актуальною задачею.

Мета дослідження. У даній статті представлені результати дослідження, що мало на меті розроблення стеганографічного способу захисту графічних даних користувача, що задовольняє такі умови:

– унеможливлення або ускладнення несанкціонованого доступу до графічних даних без наявності ключа;

– висока швидкодія алгоритму кодування та декодування графічних даних.

Слід зазначити, що для звичайного користувача швидкодія кодування та декодування даних є важливою характеристикою, тому у даному дослідженні цьому питанню приділялась особлива увага.

Опис способу. Спосіб, що пропонується, ґрунтується на LSB-кодуванні стегоданих [4]. Основною особливістю способу є використання зашифрованого образу стегозображення, що утворюється за допомогою *комплементарної функції*, яка задає відображення множини стегоданих на множину даних контейнера. Результатом цього відображення є множина зашифрованих даних, яку будемо називати *комплементарним образом*. Комплементарний образ вбудовується у контейнер замість стегозображення.

Вимогою до комплементарної функції є збереження відповідності всіх значень байтів стегозображення з відповідними значеннями байтів контейнера. Для виконання цієї вимоги в даній статті пропонується використання в ролі комплементарної функції заповнених псевдовипадковим чином латинських квадратів [1]. Причому латинський квадрат виступає в ролі ключа. Окрім цього, для відновлення даних користувача необхідно мати додаткову інформацію, таку як ширина і довжина комплементарного образу та зони накладання, а також кількість стегобіт, що використовуються при LSB-кодуванні. Ця інформація зберігається у контейнері за принципом LSB-кодування.

Генерування латинського квадрату відбувається наступним чином. Для всіх варіантів значень байту (від 0 до 255) зображення S, що приховується, та зони накладання C формується таблиця розміром 255×255, при заповненні якої мають виконуватись наступні правила:

– кожне значення може зустрічатись в рядку лише один раз;

– кожен рядок має містити всі варіанти значень;

– послідовність значень в рядках має бути непередбачуваною.

Генерування латинського квадрату розміром 255x255 відбувається наступним чином:

Створюється матриця 255x255, значення кожного її елементу дорівнює його номеру в рядку.

Для кожного рядка генерується псевдовипадкове число кількості перестановок у діапазоні від 128 до 255.

Для кожної перестановки генеруються два псевдовипадкових числа у діапазоні від 0 до 255.

Елементи з номерами двох отриманих чисел міняються місцями.

Отриманий латинський квадрат використовується для створення комплементарного образу. Створення комплементарного образу відбувається одночасно з його вбудовуванням в контейнер.

Перед початком процедури кодування стегоданих відбувається аналіз можливості розміщення комплементарного образу в контейнері: оскільки спосіб вимагає, щоб зона накладання не використовувалась для запису комплементарного образу, це збільшує вимоги до розміру контейнера. Розмір зони накладання дорівнює розміру зображення, що передається, а розмір комплементарного образу залежить від кількості бітів, що використовуються і може бути у діапазоні $S \le I \le 8S$, де S – розмір стегозображення, I – розмір комплементарного образу.

Для кожного байту комплементарного образу обчислюється значення комплементарної функції, що визначається латинським квадратом:

$$X_k = f(S_i, C_j), \qquad (1)$$

де X_k – k-й байт комплементарного образу; S_i – i-й байт зображення, що приховується; C_j – відповідний йому j-й байт контейнера; $f(S_i, C_j)$ – відповідний байт масиву значень ключа.

Тобто аргументами комплементарної функції є значення певного байта зображення S і відповідного йому байта зони накладання контейнера, а

значенням комплементарної функції є відповідне значення комірки у латинському квадраті.

Отриманий блок даних комплементарного образу вбудовується за принципом LSB-кодування в зображення таким чином, щоб забезпечити уникнення його накладання на область зображення, що використовується для його створення, а отже буде використовуватись для відновлення зображення, що приховується (рис. 1). Після вбудовування даних у зображення-контейнер його графічні дані ущільнюються. Для збереження зображення-контейнеру можуть бути використані графічні формати з ущільненням без втрат, такі як PNG та lossless JPEG. Утворений ключ зберігається і передається окремо згідно зі звичайними правилами передачі секретного ключа.

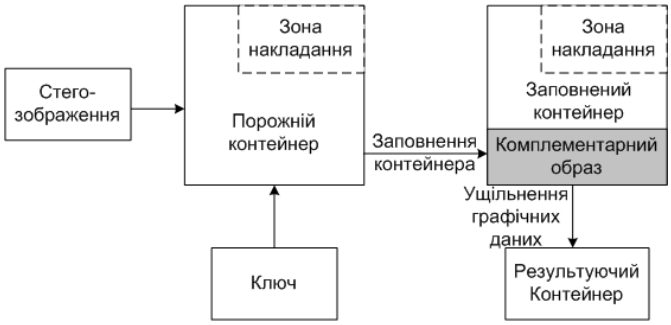


Рис. 1. LSB-кодування з використанням комплементарного образу

Відновлення даних. Для відновлення даних зі заповненого контейнера, що містить комплементарний образ, необхідно мати:

– адресу початку та розмір комплементарного образу і зони накладання;

– латинський квадрат, що є ключем.

Для зручності роботи з комплементарним образом під час декодування латинський квадрат *K* перетворюється у латинський квадрат *D* за формулою:

$$D\left[i, K\left[i, j\right]\right] = j, \qquad (2)$$

де i, j– індекси матриць латинських квадратів.

Декодування складається з двох етапів. Першим етапом є зчитування комплементарного образу, що для біту *n* байту *k* комплементарного образу формулюється наступним чином:

$$X_k = X_k \mid ((C_i \;\&\; (255 >> (8-b))) << (8-n-b)), \qquad (3)$$

де X_k – k -й байт комплементарного образу; C_i – і-й байт контейнера; b – кількість стегобіт в байті контейнера; n – порядковий номер стегобіта в послідовності стегобіт, $n \le b$.

Другим етапом є відновлення зображення, що приховане:

$$S_i = f(C_{j \cdot h+l}, X_i), \qquad (4)$$

де S_i -й байт зображення, що приховане; C– контейнер; X_i-й байт комплементарного образу; j – адреса початку зони накладання; h – ширина зони накладання; l – порядковий номер елемента в рядку зони накладання; $f(C_{j \cdot h+l}, X_i)$-відповідний байт масиву значень ключа.

В результаті повністю відновлюється зображення, приховане в комплементарному образі.

Розпаралелювання. Для прискорення роботи алгоритму було застосоване розпаралелювання між ядрами багатоядерного процесора. Для визначення частини алгоритму, що має бути розпаралелена, була проаналізована схема роботи, що не використовує розпаралелювання (рис. 2). Було вирішено застосувати розпаралелювання до запису даних в контейнер. Аналогічним чином було розпаралелене зчитування даних з контейнеру при відновленні даних.

Зовнішній цикл проходить по рядках контейнеру, починаючи від початку зони накладання *compzonestartindex* і проходячи кількість рядків стегозображення *secret.Height*. Середній цикл проходить по кожному байту стегозображення від 0 до значення *Stride*, а внутрішній цикл вбудовує стегодані у відповідності до налаштувань алгоритму, а саме – кількості стегобіт.

Оскільки розпаралелювання орієнтоване на персональний комп'ютер з багатоядерним процесором, кількість ядер якого обмежена, найбільш раціональним шляхом буде розпаралелювання тільки зовнішнього циклу. Таким чином, всі процеси всередині будуть незалежними, відповідно маючи незалежні змінні та лічильники. Це потребує більших ресурсів для забезпечення незалежності процесів (окремі змінні і лічильники в кожній паралельній ітерації), проте, як буде видно з результатів, навіть при 2 ядрах це дає перевагу в часі роботи. Подальше розпаралелювання всіх внутрішніх циклів є недоцільним, оскільки потребує значного збільшення необхідної кількості ресурсів, а переваги розпаралелювання будуть відчутні тільки у випадку, коли кількість процесорів набагато більша за кількість рядків стегозображення. Дане розпаралелювання є адаптивним до кількості ядер процесора, схема роботи розпаралеленого алгоритму зображена на рис. 3.

Оскільки запропонований спосіб з використанням розпаралелювання вимагає використання більшої кількості ресурсів, ніж спосіб без розпаралелювання, то використання цього способу на одноядерному процесорі, є недоцільним, оскільки час роботи буде більший, ніж у способу без розпаралелювання.

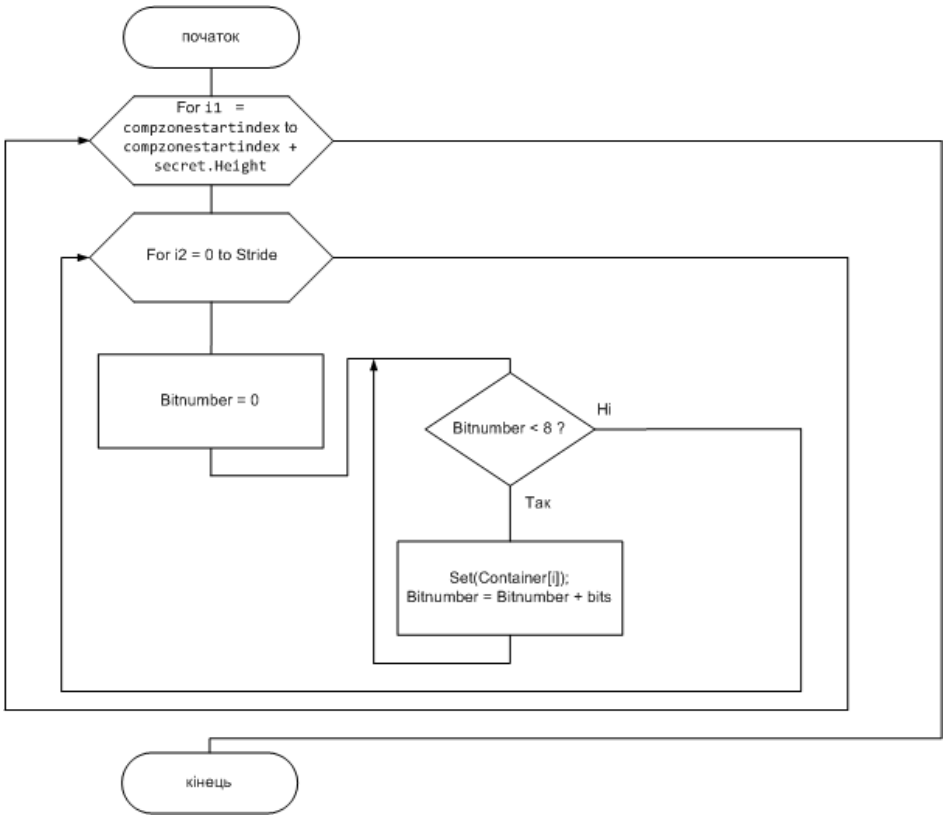


Рис. 2. Блок-схема алгоритму запису даних в контейнер без використання розпаралелювання

Тому в проведеній серії експериментів порівнюється час роботи способу з розпаралелюванням на двоядерному процесорі та способу без розпаралелювання.

Результати. Для оцінки запропонованого способу була проведена серія експериментів із заміром часу. Експерименти проводились на тестовій стегосистемі, що на даному етапі не забезпечує захисту від статистичного стегоаналізу [2].

Результати експериментів представлені на рис. 4 і рис. 5 та згруповані за розмірами контейнерів і зображень, що приховуються. Для експерименту було реалізовано алгоритм LSB без захисту, алгоритм шифрування AES, а також запропонований алгоритм на основі комплементарного образу в двох варіантах: звичайному та розпаралеленому. Заміри відбувались на комп'ютері з двоядерним процесором Intel Centrino, що дає можливість побачити переваги паралельної обробки даних.

На рис. 4 наведені дані для запису LSB без захисту, з шифруванням AES, з використанням комплементарного образу і комплементарного образу з розпаралелюванням. Результуючий час обробки даних при вбудовуванні включає в себе: час генерації ключа; час збереження ключа; час запису даних в контейнер відповідно до алгоритму.

Аналіз отриманих експериментальних даних дозволяє зробити наступні висновки:

Час виконання алгоритму на основі комплементарного образу для всіх випадків випереджає алгоритм шифрування AES при запису, хоча створення ключа займає більше часу, ніж у випадку алгоритму AES.

Розпаралелювання алгоритму на основі комплементарного образу дозволяє досягнути збільшення швидкодії, проте не дозволяє випередити алгоритм LSB без захисту.

Збільшення кількості стегобіт прискорює запис при всіх способах.

Інша група експериментів пов'язана зі зчитуванням стегоданих. Результуючий час для зчитування включає в себе:

– час зчитування ключа;

– час зчитування стегоданих / комплементарного образу;

– час розшифрування / відновлення зображення з образу.

На рис. 5 наведені дані для зчитування алгоритму LSB без захисту, алгоритму шифрування AES, алгоритму з використанням комплементарного образу у звичайній реалізації та з розпаралелюванням. Аналіз отриманих експериментальних даних дозволяє зробити такі висновки:

Алгоритм на основі комплементарного образу показує кращі часові результати, ніж алгоритм шифрування AES, а також не гірші, ніж демонструє алгоритм LSB без захисту.

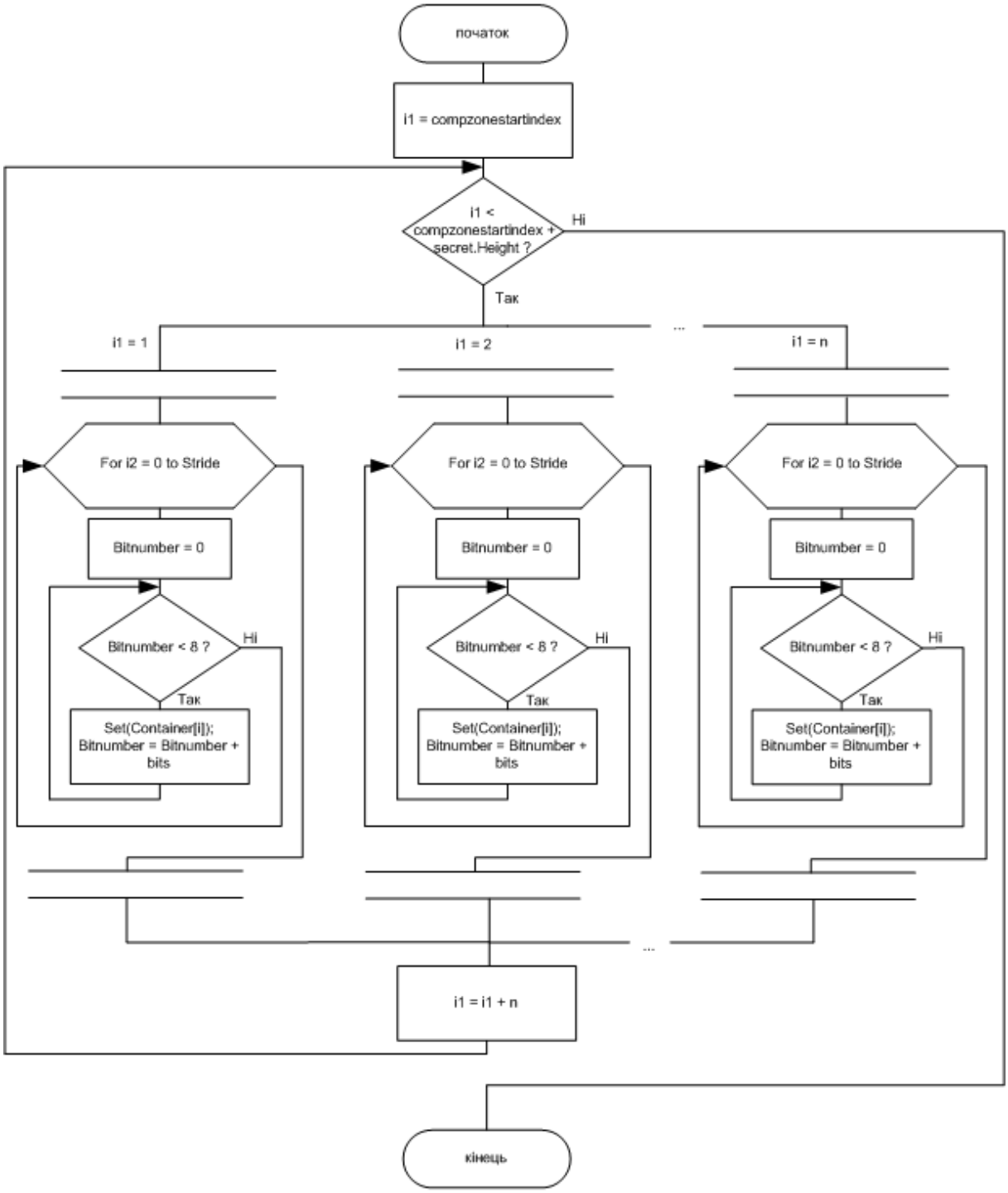
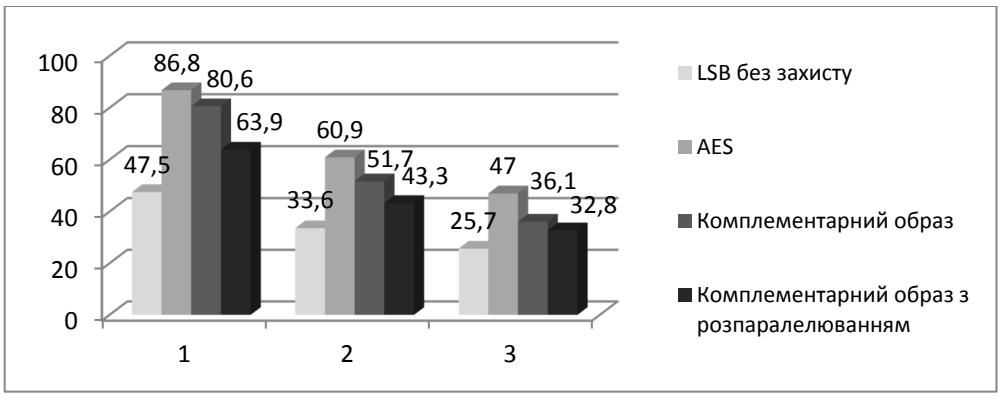
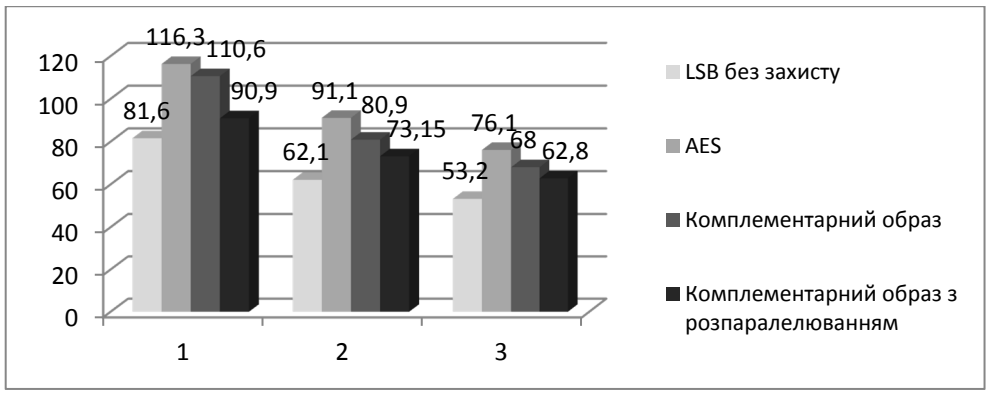


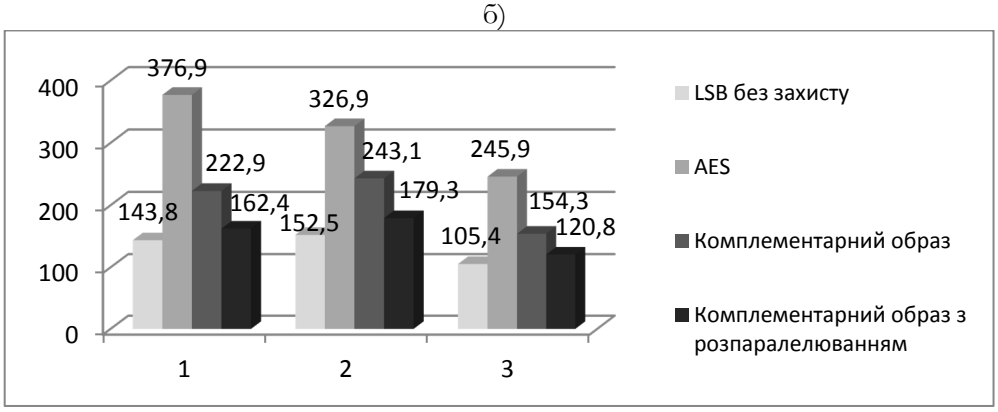
Рис. 3. Блок-схема алгоритму запису даних в контейнер з використанням розпаралелювання



а)

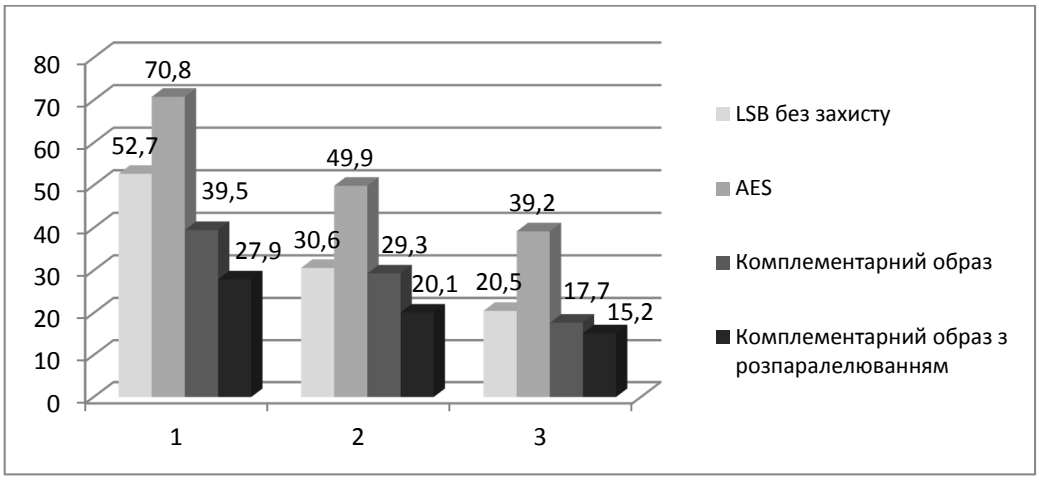


б)

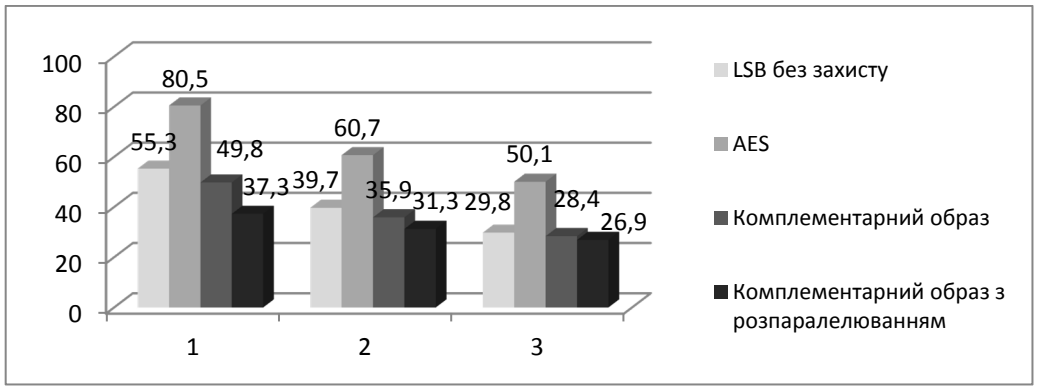


в)

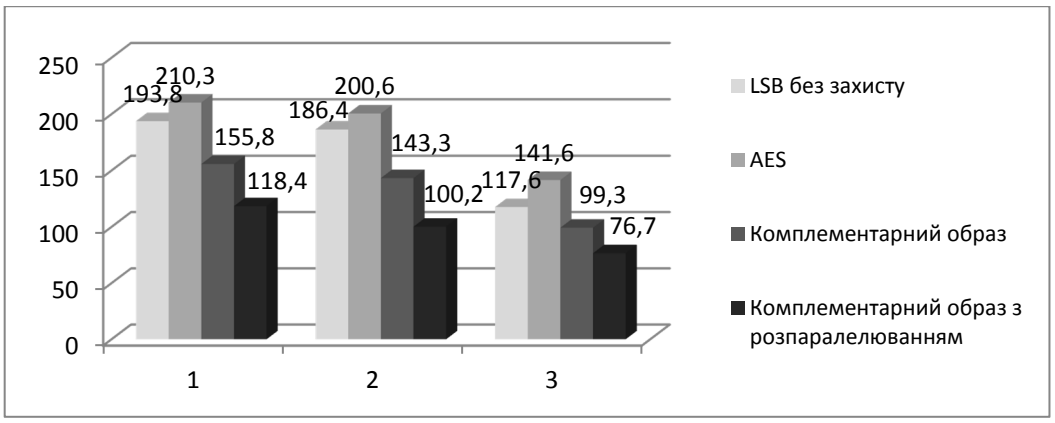
Рис. 4. Час запису стегоданих (мс): а) Контейнер 800x1204, зображення 320x213. 1 – 1 біт, 2 – 2 біта, 3 – 4 біта; б). Контейнер 1324x2048, зображення 320x213. 1 – 1 біт, 2 – 2 біта, 3 – 4 біта; в). 1 – контейнер 1324x2048, зображення 1024x685, 4 біта; 2 – контейнер 1324x2048, зображення 800x524, 2 біта; 3 – контейнер 1324x2048, зображення 800x524, 4 біта



а)



б)



в)

Рис. 5. Час зчитування стегоданих (мс): а). Контейнер 800x1204, зображення 320x213 1 – 1 біт, 2 – 2 біта, 3 – 4 біта; б). Контейнер 1324x2048, зображення 320x213. 1 – 1 біт, 2 – 2 біта, 3 – 4 біта; в). 1 – контейнер 1324x2048, зображення 1024x685, 4 біта; 2 – контейнер 1324x2048, зображення 800x524, 2 біта; 3 – контейнер 1324x2048, зображення 800x524, 4 біта

Таблиця 1

Ступінь прискорення при збільшенні кількості стегобіт

Алгоритм	Запис		Зчитування	
	2 біти	4 біти	2 біти	4 біти
LSB без захисту	1,41	1,84	1,72	2,57
AES	1,42	1,84	1,41	1,84
Комплементарний образ	1,55	2,23	1,34	2,23
Комплементарний образ з розпаралелюванням	1,47	1,94	1,38	1,83

Розпаралелювання алгоритму на основі комплементарного образу дозволяє для всіх випадків досягнути кращих часових показників, ніж LSB без захисту. Збільшення кількості стегобіт прискорює всі способи, оскільки модифікується менша кількість біт контейнера. Також було проаналізовано вплив кількості стегобіт на швидкодію. Значення в табл. 1 вказують, наскільки швидше спосіб працює у випадку 2 та 4 стегобіт, ніж при 1 стегобіті.

Отримані результати дозволяють зробити висновок, що збільшення швидкодії при записі спостерігається для всіх способів, особливо у способі на основі комплементарного образу. При цьому реалізація способу на основі комплементарного образу з розпаралелюванням має вищі показники приросту швидкодії, аніж спосіб без захисту або алгоритм AES. Отже, при збільшенні кількості стегобіт ефективність розпаралелювання тільки збільшується.

Аналіз приросту швидкодії при зчитуванні даних дає протилежні результати. Найбільший приріст швидкодії спостерігається за відсутності захисту, найменший – при способі на основі комплементарного образу з паралельною реалізацією. Але це свідчить не про неефективність використання розпаралелювання при збільшенні кількості стегобіт, а про максимальну ефективність розпаралеленого алгоритму на основі комплементарного образу саме при мінімальній кількості стегобіт, що є найбільш частим випадком, відносно якого відбувались обчислення всіх відношень. Оскільки у всіх випадках спосіб на основі комплементарного образу показує найкращі значення, в тому числі при 1 стегобіті, то і відносний приріст швидкодії буде меншим.

Висновки. Запропонований спосіб дозволяє забезпечити захист графічних даних користувача та при цьому зменшити час, необхідний на обробку даних шляхом розпаралелювання алгоритму для виконання на багатоядерних процесорах.

Незважаючи на те, що частина часу при записі витрачається на генерування ключа, розпаралелена реалізація цього способу дозволяє досягти кращих часових показників, ніж при шифруванні. При зчитуванні даний спосіб у його розпаралеленій реалізації забезпечує навіть кращі результати, ніж за відсутності захисту. Розпаралелювання на два ядра, що виконувалось в експериментах, не дає двократного приросту швидкодії, що пов'язане з необхідністю організації взаємодії роботи паралельних потоків, а також з необхідністю перебудови програмного коду для забезпечення паралельної

роботи. Проте даний спосіб, порівняно з використанням шифрування, має низку недоліків: великий розмір ключа (64 кб); збільшення часових витрат на генерування ключа; зменшення місткості контейнера.

Необхідність створювати ключ великого розміру погіршує часові показники даного способу при захисті, що більше відчувається при використанні зображень і контейнерів невеликого розміру, де відносна частка часу генерування ключа є більшою. Проте, незважаючи на наявні недоліки, даний спосіб забезпечує кращі часові показники і захист від отримання стегоданих, а отже може бути застосований у стегосистемах зі захистом від статистичного стегоаналізу [5].

ЛІТЕРАТУРА

[1]. Латинские квадраты [Електронний ресурс] / Режим доступу: http://www.e-olimp.com/en/problems/2105 – [02.03.2012].

[2]. Cummins, J. Steganography And Digital Watermarking / Jonathan Cummins, Patrick Diskin, Samuel Lau and Robert Parlett. – School of Computer Science, The University of Birmingham, 2004.

[3]. Kharrazi, Mehdi. Image Steganography: Concepts and Practice / Mehdi Kharrazi, Husrev T. Sencar, Nasir Memon. – Polytechnic University of Brooklyn, USA.

[4]. Owens, M. A discussion of covert channels and steganography / M. Owens. – SANS Institute, 2002.

[5]. Zaidan, B. B. Stego-Image Vs Stego-Analysis System. / B. B Zaidan, A. A Zaidan, Alaa Taqa, Fazida Othman // International Journal of Computer and Electrical Engineering. – Vol. 1, No. 5 December, 2009.

REFERENCES

[1]. Latin squares (2012). Available from: http://www.e-olimp.com/en/problems/2105. [Accessed: 2nd March 2012].

[2]. Cummins, J. (2004). Steganography And Digital Watermarking. School of Computer Science, The University of Birmingham.

[3]. Kharrazi, Mehdi et al. Image Steganography: Concepts and Practice. Polytechnic University of Brooklyn, USA.

[4]. Owens, M. (2002). A discussion of covert channels and steganography. SANS Institute.

[5]. Zaidan, B.B. et al (2009). Stego-Image Vs Stego-Analysis System. International Journal of Computer and Electrical Engineering, Vol. 1, No. 5.

СПОСОБ СТЕГАНОГРАФИИ ИЗОБРАЖЕНИЙ НА ОСНОВЕ КОМПЛЕМЕНТАРНОГО ОБРАЗА

Цифровые изображения при их передаче через Интернет или сохранении на серверах общего пользования могут легко попасть в открытый доступ. Если эти изображения носят характер личной информации, то задача их простой и эффективной защиты становится для пользователя актуальной. Кроме того, для обычного пользователя важным является время, необходимое для шифрования и дешифрования изображения. Существующие средства криптографической защиты не ориентированы на обычного пользователя, которому нужен простой и быстрый способ защиты графических данных при их передаче по сети в реальном времени. В статье предлагается стеганографический способ защиты изображений, который основан на сохранении вместо изображения его комплементарного образа и использовании в роле ключа рандомизированного латинского квадрата. Разработанный метод обеспечивает лучшие временные показатели процедуры кодирования и декодирования скрываемого изображения, чем аналогичные способы, использующие криптографию. Это достигается путем использования параллельных вычислений. Данный способ может быть использован в реальном времени для защиты личных изображений пользователя перед их передачей по сети. Скрытые изображения могут быть сохранены в формате png или jpeg со сжатием без потерь.

Ключевые слова: стеганография, комплементарный образ, латинские квадраты

IMAGE STEGANOGRAPHY METHOD BASED ON COMPLEMENTARY IMAGE

Abstract. Digital images while either transferring via Internet or storing in common-used servers can easily get into open access. If these images are private, the task of their simple and effective protection becomes topical for a user. Furthermore, time of image processing while protecting is important for an ordinary user. The existing cryptographic methods are not oriented to an ordinary user who needs a simple and fast method for graphical data security while transferring them via network in real time. This paper proposes the steganographic method for digital images protection based on storage of a complementary image instead of the stego-image as well as on using a randomized Latin square as a key. The developed method provides better time performance in encoding and decoding of user data than similar cryptographic methods. It is achieved by using parallel computing. The proposed method can be used in real-time for the protection of private images of a user before their transfer via network. Hidden images can be saved in either png or lossless jpeg formats.

Index Terms: steganography, complementary image, Latin squares

Сулема Євгенія Станіславівна, кандидат технічних наук, доцент, Національний технічний університет України «Київський політехнічний інститут», доцент кафедри програмного забезпечення комп'ютерних систем.
E-mail: sulema@pzks.fpm.kpi.ua

Сулема Евгения Станиславовна, кандидат технических наук, доцент, Национальный технический университет Украины «Киевский политехнический институт», доцент кафедры программного обеспечения компьютерных систем.
Sulema Yevgeniya, Ph.D., Assoc. Prof., National Technical University of Ukraine «Kyiv Polytechnic Institute», Assoc. Prof. of Systems Software Department.

Широчин Семен Станіславович, Національний технічний університет України «Київський політехнічний інститут», аспірант кафедри програмного забезпечення комп'ютерних систем.
E-mail: semenstsh@mail.ru
Широчин Семён Станиславович, Национальный технический университет Украины «Киевский политехнический институт», аспирант кафедры программного обеспечения компьютерных систем.
Shyrochyn Semen, National Technical University of Ukraine «Kyiv Polytechnic Institute», Ph.D. student of Systems Software Department.

УДК 004.056.53(045)

СИСТЕМА ФОРМИРОВАНИЯ ЭВРИСТИЧЕСКИХ ПРАВИЛ ДЛЯ ОЦЕНИВАНИЯ СЕТЕВОЙ АКТИВНОСТИ

Анна Корченко

На основе известного метода выявления аномалий порожденных кибератаками разработана соответствующая система, для поддержки функционирования которой необходима реализация этапа формирования множества эвристических правил. Они предназначены для создания соответствующих решающих правил, направленных на проверку истинности взаимосвязей эталонных и текущих параметров при оценивании сетевой активности в определенной среде окружения. Для решения такой задачи предложено новое структурное решение соответствующей системы, основанной на базе правил и содержащей блоки коммутации, формирования логико-лингвистических связок, ранжирования и инициализации правил, а также регистры эталонов, текущих значений, лингвистических идентификаторов и правил. Предложенное решение может быть реализовано программно или программно-аппаратно и использоваться в качестве основы систем выявления аномалий.

***Ключевые слова**: кибератаки, аномалии, системы обнаружения вторжений, системы обнаружения аномалий, системы обнаружения атак, обнаружение аномалий в компьютерных сетях, эвристические правила, оценка сетевой активности.*

Современная теоретическая и практическая база, которая используется для обнаружения атак в компьютерных сетях, имеет определенные ограничения по идентификации новых и несигнатурных типов кибератак. Применение математического аппарата теории нечетких множеств для построения средств обнаружения аномалий, порожденных атакующими действиями, позволит усовершенствовать существующие системы обнаружения вторжений. С этой целью разработана базовая модель параметров для нечетко определенной слабоформализованой среды [1] и универсальная модель эталонов лингвистических переменных [2], позволяющие формализовать процесс построения эталонных значений и устанавливать соответствие между типом атаки и необходимыми для ее идентификации атрибутами. Также построена модель эвристических правил (ЭП) [3], которая за счет множества эталонных параметров, логико-лингвистических связок и лингвистических идентификаторов позволяет формализовать процесс формирования множеств ЭП для выявления аномального состояния.

В работе [4] разработан метод выявления аномалий, который за счет указанных моделей [1-3, 5] и сформированных текущих параметров, позволяет строить средства обнаружения несигнатурных и новых типов кибератак. На основе этого метода предложено новое структурное решение системы выявления аномального состояния в компьютерных сетях [6]. Она состоит из подсистем первичной обработки, формирования нечетких эталонов [7] и формирования ЭП, а также модулей нечеткой арифметики, логического вывода и визуализации.

Это решение используется для совершенствования систем сетевой безопасности, которое основывается на реализации указанного метода обнаружения аномалий [4] и ориентировано на осуществление контроля активности в определенной среде окружения.