Вісник ЛДУБЖД

Bulletin of Lviv State University of Life Safety

ISSN 2078-4643
https://journal.ldubgd.edu.ua/index.php/Visnuk/
УДК 004.056.5:003.26:004.415.24
DOI: 10.32447/20784643.20.2019.03

[1]А. Е. Лагун, [2]О. І. Полотай
[1]Національний університет «Львівська політехніка»
[2]Львівський державний університет безпеки життєдіяльності

ОСОБЛИВОСТІ ПРИХОВУВАННЯ ІНФОРМАЦІЇ В ЗОБРАЖЕННЯХ З ВИКОРИСТАННЯМ МОЛОДШОГО ЗНАЧУЩОГО БІТА

У статті розглядаються особливості реалізації стеганографічних алгоритмів приховування інформації в нерухомих зображеннях. Проведено огляд різних алгоритмів вбудовування, що використовують метод молодшого значущого біта. Зокрема використання цифрової фільтрації дозволяє краще вибрати необхідні пікселі для вбудовування, а використання генератора псевдовипадкової послідовності дає змогу більш ефективно приховувати секретну інформацію, ускладнюючи зловмиснику пошук секретної інформації.

З існуючих кольорових палітр для представлення нерухомих зображень вибрано RGB-палітру, яка є найбільш поширеною і містить червоний, зелений та синій кольори різної інтенсивності для створення пікселів зображення. Для формування заповнених стеганографічних контейнерів використовуються кольори, до яких менш чутливе людське око, для забезпечення додаткової візуальної стійкості.

Також в роботі досліджено особливості приховування цифрової текстової інформації в нерухомому зображенні у вигляді BMP-файлу та реалізовано алгоритм, що для файлів зображень різного розміру дозволяє приховати текстовий файл необхідного розміру. Зокрема, у початковий контейнер записується кількість байт секретного повідомлення для отримання необхідної кількості символів при видобуванні. Крім того, враховано особливості формування BMP- файлу, який містить додаткові вирівнювальні байти рядка.

В загальному випадку, алгоритм дозволяє вибрати файл контейнера відповідного розміру для приховування секретної інформації, а також кольори палітри, в які буде вбудовуватися інформація. Видобування секретної інформації відбувається до моменту досягнення кількості байт прихованого повідомлення, яка є значенням, що записане на початку приховування.

Для ускладнення пошуку зрозумілого тексту можна використати перед приховуванням алгоритми шифрування або компресії, які перетворять звичайний текст в незрозумілу форму, і лише ті, хто буде знати про використані алгоритми і, можливо, ключі зможуть правильно прочитати приховану інформацію.

Ключові слова: стеганографічні алгоритми, секретна інформація, нерухоме зображення, контейнер, RGB палітра, BMP файл, молодший значущий біт.

Вступ

Останніми роками інформаційні технології все більше проникають в різні галузі діяльності суспільства. Зрозуміло, що із поширенням цифрової інформації особливо актуальними є завдання захисту цієї інформації від навмисних і ненавмисних впливів. Наприклад, особисту інформацію можна вкрасти і використати у зловмисних цілях, змінити, передавши невірні повідомлення іншому адресату.

Для захисту інформації серед багатьох різних методів використовуються криптографічні та стеганографічні алгоритми [1]. Дослідженням цих алгоритмів присвячено дуже багато праць. У цій роботі будуть розглядатися стеганографічні методи, тому коротко розглянемо їх принципи роботи [2].

В стеганографічних системах приховується факт існування секретної інформації. Як правило, інформація приховується у цифрових носіях, що мають фізичну природу – текстах, зображеннях та

Інформація про авторів:

Лагун Андрій Едуардович, *кандидат технічних наук, доцент*
Національний університет Львівська Політехніка
a.e.lagun@gmail.com
097-457-75-15

Полотай Орест Іванович, *кафедра управління інформаційною безпекою, кандидат технічних наук, доцент*
Львівський державний університет безпеки життєдіяльності
orest.polotaj@gmail.com
097-668-45-62

відеопослідовностях [3]. Надалі розглянемо приховування інформації в нерухомих зображеннях.

Нерухоме цифрове зображення є масивом пікселів. Кожен піксел характеризується місцем розташування і кольором. Колір піксела залежить від типу кольорової палітри. Наприклад, будь-які зображення можуть бути створені в палітрі на основі градацій сірого або кольоровій. Якщо використовується кольорова палітра, то важливим є використання певних видів кольорів. Найбільш використовуваною є RGB палітра, яка складається з червоного, зеленого та синього кольорів. Значення інтенсивностей кожного з кольорів визначають колір піксела нерухомого зображення. Нехай інтенсивність кольору кодується одним байтом, а саме значенням від 0 до 255. Відомо, що людське око не здатне помітити зміну молодшого значущого біта в кольорі, тому молодші значущі біти можна замінити на якусь інформацію – в нашому випадку секретну.

В стеганографії існують різні алгоритми, що використовують молодший значущий біт.

Під час приховування найпростіше записати секретну інформацію в кожен піксел зображення, починаючи з верхнього лівого кутка до правого нижнього. Тоді секретну інформацію можна відносно легко знайти і прочитати.

Якщо секретну інформацію розташовувати в зображенні згідно з якоюсь псевдовипадковою послідовністю, яка залежить від ключа, то знайти приховані біти буде важко. Зрозуміло, що ключ має використовуватися і на передавальній, і на приймальній сторонах [4].

Існує також спосіб приховування, який використовує пошук пікселів порожнього контейнера найменш помітних для людського ока, в які буде записуватися секретна інформація. Тоді можна використати не один, а кілька молодших значущих біт для одного байта кольору.

Поєднання алгоритму пошуку пікселів і використання псевдовипадкової послідовності дозволяє досягти ще кращого результату.

Дослідження та реалізація алгоритму приховування секретної інформації

В даній статті розглянемо особливості приховування цифрової інформації в нерухомому зображенні у вигляді BMP-файлу. Алгоритм, в загальному випадку, буде складатися з таких кроків:

- відкривання файлу контейнера для приховування інформації;
- вибір кольору палітри контейнера (червона, зелена, синя), в якому буде міститися прихована інформація (за умови недостатності розміру зображення для зберігання в одному кольорі, потрібно вибрати наступний колір);
- відкривання файлу з інформацією, яка буде приховуватися;
- переведення інформації для приховування в двійкову форму; для текстових файлів кожен символ перетворюється в масив з 8 біт – 1 байт; для нерухомих зображень кожна із характеристик зображення (висота, ширина, кількість пікселів і т.д.), а також інтенсивності кольорів пікселів, перетворюються в двійкові масиви по 8 біт;
- заміна молодшого значущого біта піксела контейнера бітом прихованої інформації згідно із величиною вибраного ключа;
- зберігання зміненого файлу (заповненого контейнера) у вигляді BMP- файлу згідно з правилами формування файлу у форматі BMP.

При видобуванні, в першу чергу, необхідно знати ключ, згідно з яким прихована інформація була розміщена в контейнері. Спочатку потрібно отримати значення кольорів пікселів заповненого контейнера, відкривши файл з прихованою інформацією, як і під час приховування. Потім вибираються молодші значущі біти кольорів пікселів з використанням потрібного ключа і з них формується прихована інформація.

Одним з важливих моментів є знаходження місця завершення видобування прихованої інформації. Якщо не встановити жодних вказівників завершення прихованого повідомлення, то видобування буде тривати, доки не переглянеться весь заповнений контейнер. Більше того, видобута інформація, яка буде міститися після прихованої інформації, може бути незрозумілою і тому виділити корисну інформацію може бути важко. Тому найпростішим способом для вдалого розпізнавання прихованої інформації є використання якихось міток для точної ідентифікації секретного повідомлення, наприклад запис кількості символів або пікселів прихованої інформації на початку або наприкінці прихованого повідомлення. Тоді кількість біт контейнера для обробки під час видобування буде наперед відомою.

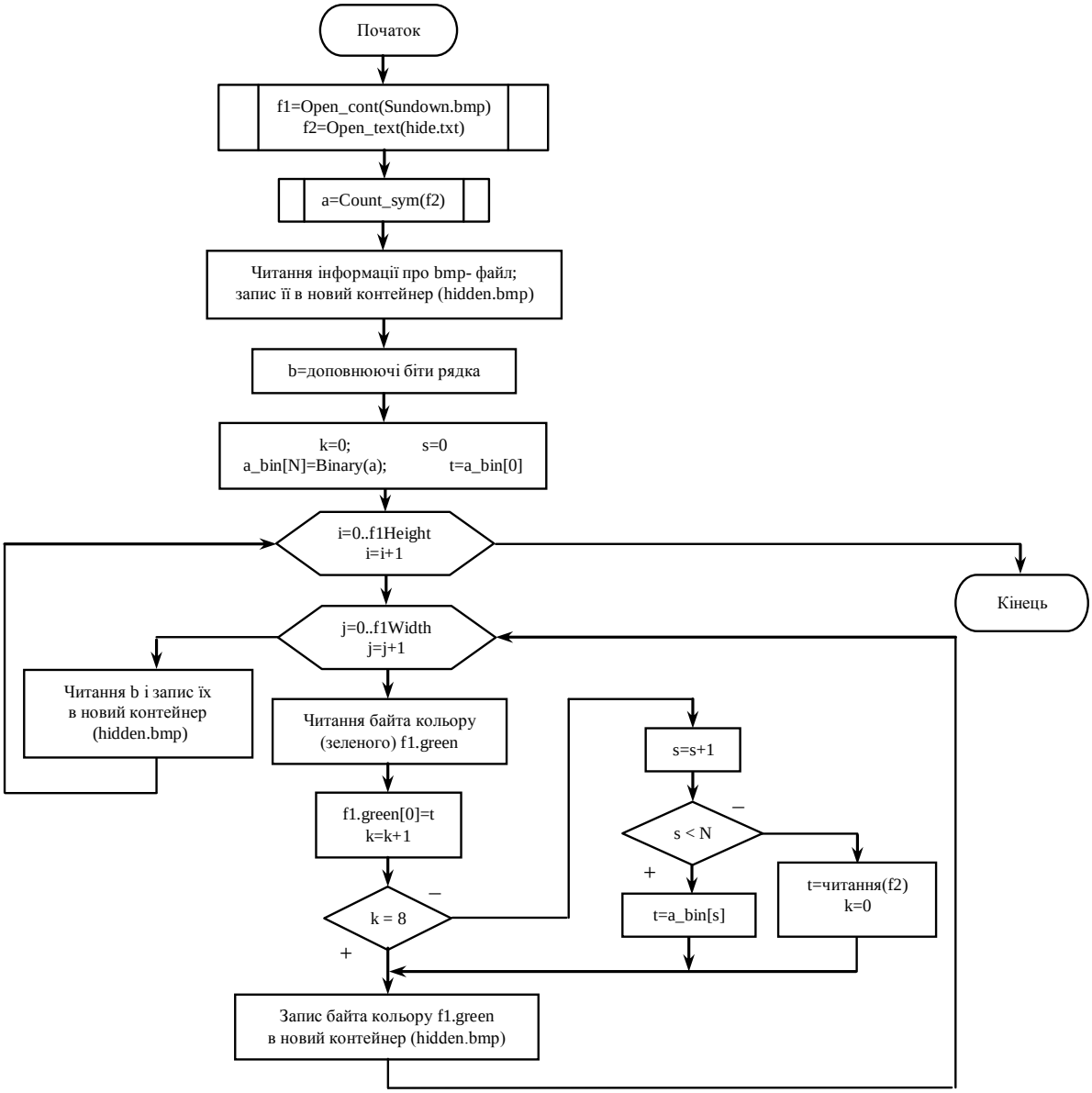

Рисунок 1 - Алгоритм формування заповненого контейнера стеганографічного зображення

Структуру алгоритму приховування наведено на рис. 1.

Контейнером вибираємо зображення у вигляді BMP-файлу, який складається з трьох матриць кольорів – червоного, зеленого та синього – ‘Sundown.bmp’ (рис. 2). Кожен колір представлений своєю інтенсивністю, а саме числами в діапазоні від 0 до 255. Комбінація червоного, зеленого та синього кольорів формує колір пікселя зображення.

Для приховування вибираємо текст, який записаний у текстовому файлі ‘hide.txt’ (рис. 3). Як було сказано вище, перед початком процесу приховування обчислюється кількість символів, які потрібно приховати. Потім кількість символів (**a**) записується у вигляді “\ **a** \” перед прихованим повідомленням. Це потрібно для того, щоб під час видобування точно було відомо кількість символів, які потрібно видобути із заповненого контейнера.

Під час формування заповненого контейнера – ‘hidden.bmp’ у нього записується зчитана з порожнього контейнера інформація про структуру BMP-файлу.

Рисунок 2 - Файл-зображення порожнього контейнера

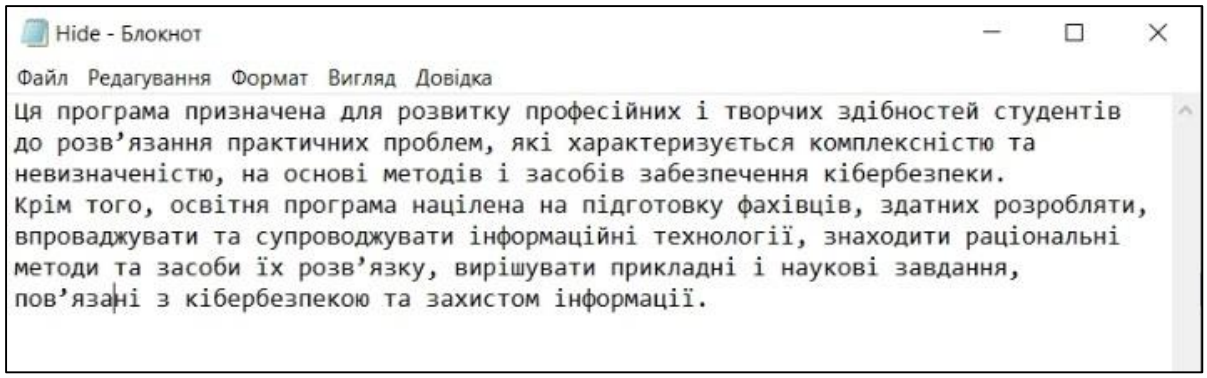

Рисунок 3 – Вміст текстового файлу з прихованим повідомленням

При читанні і записі BMP-файлу необхідно враховувати вирівнюючі байти [5]. А саме довжина кожного рядка пікселів має бути кратна 4 байтам. Оскільки зображення обробляється у 24-бітному форматі BMP, то у файлі за потреби формуються додаткові порожні байти (**b**) для кратності 4. Для цього використовується формула

$$b = 4 - (Height * 3) \% 4 \qquad (1)$$

де Height – ширина зображення в пікселах, % – операція остачі цілочисельного ділення.

Наступним кроком є переведення значення кількості символів в двійкову форму і вбудовування цього двійкового значення у молодші значущі біти одного з кольорів порожнього контейнера. Наприклад, вибираємо зелений колір. Після вбудовування кількості символів прихованої інформації необхідно вбудувати двійкові значення символів прихованого повідомлення в кожен молодший значущий біт порожнього контейнера. Всі значення в першому випадку будемо вбудовувати по рядках порожнього контейнера, починаючи з лівого верхнього кутка зображення, тобто стеганографічним ключем буде місцезнаходження лівого верхнього кутка.

Після формування кожного рядка заповненого контейнера читаються з порожнього контейнера і записуються у заповнений контейнер вирівнюючі байти, кількість яких обчислюється за формулою (1).

Алгоритм видобування інформації з молодших значущих бітів

Структуру алгоритму видобування прихованої інформації наведено на рис. 4. Коротко опишемо суть цього алгоритму.

Спочатку читається інформація про bmp-файл заповненого контейнера і обчислюється кількість доповнюючих бітів рядка. Після загальної інформації про зображення знаходяться байти кольорів зображення, в яких в молодших значущих бітах міститься прихована інформація. Як було зазначено вище, приховані біти розміщувалися в порядку зліва направо і зверху вниз без використання спеціального ключа. Тому читання bmp-файлу і видобування прихованої інформації буде відбуватися в такому ж порядку.

Відомо, що на першому місці між двома символами зворотних слешів у порожній контейнер записувалася кількість символів прихованої інформації для точного підрахунку довжини видобутого повідомлення, Тому спочатку визначається ця кількість. Взагалі кажучи, кожне десяткове значення прихованого символу обчислюється за формулою:

$$t = \sum_{i=0}^{7} LSB_i * 2^i, \qquad (2)$$

після чого це значення переводиться в символ згідно з ASCII таблицею кодування.

Кожен отриманий символ записується у текстовий файл ‘result.txt’. Після читання визначеної кількості бітів рядка bmp- файлу за потреби читаються ще доповнюючі біти. Зрозуміло, що в алгоритмі використовується допоміжна змінна, в якій міститься кількість перетворених символів. Якщо це значення збігається із числом, що міститься до початку прихованого повідомлення і вказує на кількість символів в повідомленні, то алгоритм припиняє роботу.

Зрозуміло, що в алгоритмі стеганографічного приховування-видобування можна використовувати різноманітні ключі, які будуть вказувати на місцезнаходження бітів прихованого повідомлення і, таким чином, ускладнювати пошук прихованої інформації зловмисником. Також для ускладнення пошуку зрозумілого тексту можна використати перед приховуванням алгоритми шифрування або компресії, які перетворять звичайний текст в незрозумілу форму, і лише ті, хто будуть знати про використані алгоритми і, можливо, ключі, зможуть правильно прочитати приховану інформацію.

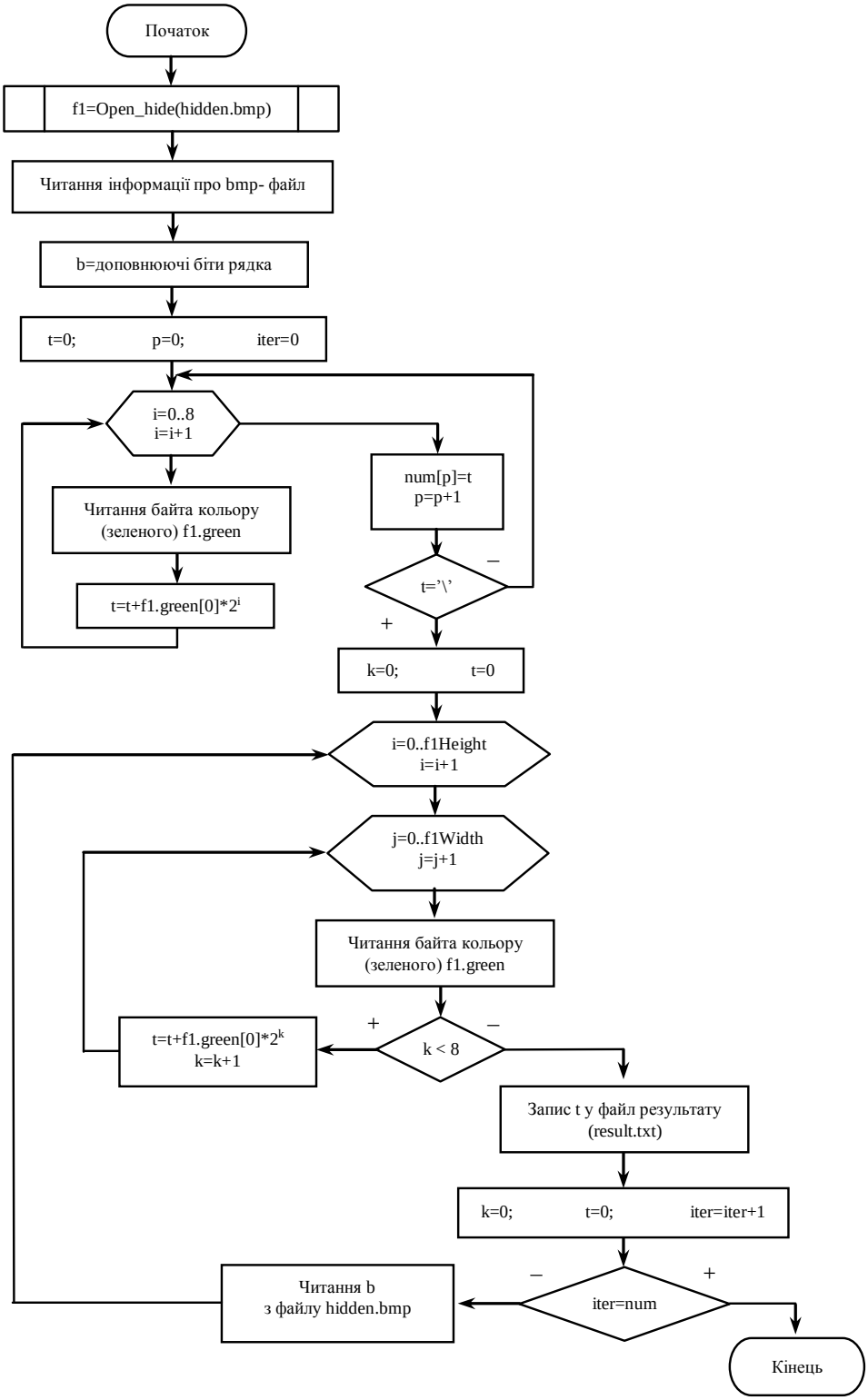

Рисунок 4 – Стеганографічний алгоритм видобування прихованої інформації

Висновки

В статті розглянуто особливості використання та реалізації алгоритму молодшого значущого біта для приховування секретної інформації в нерухомих зображеннях. Цей алгоритм має такі переваги, як простота і можливість приховування значної кількості інформації. Проте основним недоліком стеганографічних алгоритмів молодшого значущого біта є мала стійкість до різноманітних геометричних та інших атак. Тому основною задачею є забезпечення стійкості до таких атак.

Список літератури:

1. Schneier B. Applied Cryptography: Protocols, Algorithms, and Source Code in C. 2nd ed. / B. Schneier. – New York : John Wiley and Sons, 1996.

2. Конахович Г. Ф. Компьютерная стеганография. Теория и практика / Г. Ф. Конахович, А. Ю Пузыренко. – К. : МК-Пресс, 2006. – 249 с.

3. Лагун А. Атаки на сучасні стеганографічні системи і методи захисту / А. Лагун // Збірник наукових праць «Вісник ЛДУ БЖД». – Львів : ЛДУ БЖД, 2016. – № 13. – С. 6-12.

4. Lagun A. Embedding of the hidden information with the use of Discrete Fourier Transform [Electronic resource] / A. Lagun, N. Kukharska // Automatic Control and Information Technology (ICACIT'17) : 4th International Conference, 14-16 December 2017 : Proceedings. – Cracow, 2017. – 1 electr. opt. disk (CD-ROM).

5. BMP file format. Wikipedia, the free encyclopedia. Retrieved from https://en.wikipedia.org/wiki/BMP_file_format.

References:

1. Schneier, B. (1996). *Applied Cryptography: Protocols, Algorithms, and Source Code in C*. John Wiley and Sons, New York, (USA).

2. Konakhovych, G. F. and Puzyrenko, A. Yu. (2006*). Computer steganography*. MK-PRESS, Kyiv (in Russ.)

3. Lagun, A. (2016). Attacks on modern steganographic systems and security methods. *Bulletin of Lviv State University of Life Safety*, *13*, 6-12 (in Ukr.)

4. Lagun, A. (2017). "Embedding of the hidden information with the use of Discrete Fourier Transform". *Automatic Control and Information Technology (ICACIT'17) : 4th International Conference*, 14-16 December 2017, Cracow, available at http://icacit2017.weebly.com.

5. BMP file format. Wikipedia, the free encyclopedia. Retrieved from https://en.wikipedia.org/wiki/BMP_file_format.

A.E. Lagun, O.I. Polotai

FEATURES OF HIDING INFORMATION IN IMAGES WITH USING THE LEAST SIGNIFICANT BIT

In the article has considered the peculiarities of steganographic algorithms implemenation for hiding information in inmoveable images. Authors has described different embedding algorithms which use the method of least significant bit. In particular, the use of digital filtering allows you to better select the necessary pixels for embedding, and the use of a pseudorandom sequence generator allows you to more effectively hide secret information, complicating the search for secret information to the attacker.

From the existing color palettes to represent inmoveable images have been selected the most common RGB palette, which contains red, green, and blue intensities to produce image pixels. Colors that are less sensitive to the human eye are used to form the filled steganographic containers to provide additional visual stability.

Also, in the paper authors have investigated the features of hiding digital text information in a inmoveable image as a BMP file and have realized an algorithm that for images of different size allows you to hide a text file of the necessary size. In particular, the number of bytes of the secret message is written to the original container to retrieve the required number of characters during searching. In addition, it takes into account the peculiarities of forming a BMP file that contains additional alignment bytes of the string.

In general, the algorithm allows you to select a container file of the appropriate size to hide the secret information, as well as the colors of the palette in which the information will be embedded. The extracting of secret information occurs until the number of bytes of the hidden message is reached. This value has recorded at the beginning of the hiding text.

You can use encryption or compression algorithms to complication searching of clear text by attacker. Only users those who are aware of the algorithms used and perhaps the keys will be able to read the hidden information correctly.

Keywords: steganographic algorithms, secret information, inmoveable image, container, RGB palette, BMP file, least significant bit.

***Науково-методична стаття**