Вісник ЛДУБЖД

Bulletin of Lviv State University of Life Safety

ISSN 2078-4643 (print), ISSN 2708-1389 (online)
https://journal.ldubgd.edu.ua/index.php/Visnuk/
УДК 004.056
DOI: 10.32447/20784643.23.2021.04

О.І. Полотай, О.І. Белей, Н.А. Мальцева

ФІЗИЧНИЙ ЗМІСТ КОМП'ЮТЕРНОЇ СТЕГАНОГРАФІЇ

Постановка проблеми. Розвиток засобів обчислювальної техніки дав новий поштовх для застосування комп'ютерної стеганографії. Однак важливих є розуміння фізичного змісту цього виду стеганографії.

Мета. Метою роботи є описати практичне використання і фізичний зміст явища комп'ютерної стеганографії, представити результати виконаного дослідження із приховування файлів у стегоконтейнері.

Результати. Описано основні на теперішній час методи комп'ютерної стеганографії активно використовуються для вирішення таких завдань: захист конфіденційної інформації від несанкціонованого доступу; подолання систем моніторингу і управління мережевими ресурсами; камуфляж програмного забезпечення; захист авторських прав, що проявляється в технології використання цифрових водяних знаків, що є одним з найбільш перспективних напрямів комп'ютерної стеганографії. Серед методів приховування інформації в зображеннях найпоширенішими є категорія алгоритмів з використанням молодших бітів даних зображення. Саме вони і розглядається у цій роботі. Ці алгоритми засновані на тому факті, що в деяких форматах файлів молодші біти значень хоча і присутні у файлі, але не впливають на сприйняття людиною звуку або зображення. Для дослідження у роботі було обрано стеганографічний програмний засіб S-Tools. З метою аналізу особливостей розміщення стегоданих у файлах-контейнерах було створено два тестові монотонні зображення розміром 50×50 пікселів у 24-бітному форматі bmp. Для дослідження було обрано зображення чорного та білого кольорів. У кожному із зображень було приховано текстовий файл, після чого виконано обернену дію – видобування файлу. В результаті приховування було одержано два стегофайли. У роботі було порівняно двійковий вміст початкових зображень та файлів, що містять приховані дані. Для порівняння наведено двійковий вміст зображення чорного квадрата та вміст стегоконтейнера із прихованим текстовим файлом. Зазначимо, що вміст контейнера та стегофайлу наведено лише частково, проте адреси комірок пам'яті було обрано відповідні. У правому стовпці наведено вміст комірок пам'яті у шістнадцятковому форматі. Байти, що відображають колір квадрата, мають значення «00», оскільки початкове зображення містить лише чорний колір. Було відзначено закономірність, що вміст комірок, які відповідають за зображення, змінився після приховання додаткових даних (це відображають комірки із значеннями «01»). Також в роботі описано процедуру приховування групи різнотипних файлів. Під час проведеного дослідження було встановлено, що у файлі зображення (1920×1080 пікселів) обсягом 6 220 854 байт можна приховати 777 584 байт інформації

Висновки. При застосуванні стеганографії програми використовують певні алгоритми, які приховують секретні дані серед вмісту контейнера: біти початкового файлу у випадкових позиціях замінюються на біти приховуваного файлу. Таким чином, розмір початкового файлу і файлу-контейнера (що містить вкладену інформацію) є однаковим, навіть за умови приховуванні різної кількості файлів або різного обсягу даних.

Ключові слова: комп'ютерна стеганографія, секретний ключ, стегоконтейнер, приховування інформації.

O.I Polotai, O.I Belej, N.A. Maltseva

PHYSICAL CONTENT OF COMPUTER STEGANOGRAPHY

Introduction. The development of computer technology has given a new impetus to the use of computer steganography. However, it is important to understand the physical content of this type of steganography.

Purpose. The work aims to describe the practical use and physical content of the phenomenon of computer steganography, the results of the study on the hiding of files in the stegocontainer.

Results. Describes the main ns currently computer steganography methods are actively used to solve the following tasks: Protection of confidential information from unauthorized access, overcoming monitoring and management of network resources, software camouflage, copyright protection, which is manifested in the use of digital watermarks, is one of the most promising areas of computer steganography. Among the methods of hiding information in images, the most common is the category of algorithms using the lower bits of the image data. They are considered in this paper. These algorithms are based on the fact that in some file formats, the lower bits of the values, although present in the file, but do not affect a person's perception of sound or image. The steganographic software S-Tools was chosen for the study. We created two test monotonous images with the size of 50 × 50 pixels in 24-bit bmp format to analyze the peculiarities of the placement of stego-data in container files. We chose black and white images for the study. A text file was hidden in

each of the images, after which the reverse action was performed - extracting the file. As a result of hiding, two stego files were obtained. The paper compared the binary content of the original images and files containing private data. For comparison, the binary content of the black square image and the contents of the stegocontainer with a latent text file are given. Note that the contents of the container and the stego file are only partially listed, but the addresses of the memory cells have selected accordingly. The right column shows the contents of the memory cells in hexadecimal format. The bytes that display the colour of the square are set to "00" because the original image contains only black. We noted that the contents of the cells responsible for the image changed after hiding additional data (this reflected by cells with values of "01"). The paper also describes the procedure for hiding a group of different types of files. During the study, we found that the image file (1920 × 1080 pixels) with a volume of 6,220,854 bytes can hide 777,584 bytes of information.

Conclusion. When using steganography, the program uses some algorithms that hide confidential data among the contents of the container: bits of the hidden file replace the bits of the original file at random positions. Thus, the size of the source file and the container file (containing the attached information) is the same, even if you hide a different number of files or different amounts of data.

Keywords: computer steganography, secret key, stegocontainer, information hiding.

Вступ. Розвиток засобів обчислювальної техніки дав новий поштовх для застосування комп'ютерної стеганографії. Повідомлення, що мають аналогову природу, вбудовуються у цифрові дані (мова, аудіозаписи, зображення, відео тощо). Наразі методи комп'ютерної стеганографії активно використовуються для вирішення таких завдань [5]:

1. Захист конфіденційної інформації від несанкціонованого доступу. Ця область використання комп'ютерної стеганографії є найбільш ефективною при вирішенні проблем захисту конфіденційної інформації. Так, наприклад, об'єм таємного повідомлення в звукових і графічних файлах може становити до 25 – 30 % від розміру файлу. Причому, аудіовізуальні зміни не виявляються при прослуховуванні чи перегляді файлів більшістю людей, навіть якщо факт приховування відомий.

2. Подолання систем моніторингу і управління мережевими ресурсами. Стеганографічні методи дозволяють протистояти спробам контролю над інформаційним простором при проходженні інформації через сервери управління локальних і глобальних комп’ютерних мереж.

3. Камуфляж програмного забезпечення. Застосовується в тих випадках, коли використання програмного забезпечення незареєстрованими користувачами є небажаним. Відповідне програмне забезпечення може бути закамуфльоване під стандартні програмні продукти (наприклад, текстовий редактор) або приховано у файлах мультимедіа і використовуватися лише особами, що мають на це право.

4. Захист авторських прав. Одним з найбільш перспективних напрямів комп'ютерної стеганографії є технологія використання цифрових водяних знаків (ЦВЗ) – створення невидимих оку знаків захисту авторських прав у графічних і аудіо файлах. Такі ЦВЗ, поміщені у файлі, можуть бути розпізнані спеціальними програмами, які зчитуватимуть з файлу додаткову інформацію: коли створений файл, хто володіє авторськими правами тощо.

Методи досліджень. Існують два основні напрями в комп'ютерній стеганографії: пов'язані з цифровою обробкою сигналів і не пов'язані з нею [6]. В останньому випадку повідомлення можуть бути вбудовані в заголовки файлів, заголовки пакетів даних. Цей напрям має обмежене застосування у зв'язку з відносною легкістю розкриття та знищення прихованої інформації. Більшість сучасних досліджень в області стеганографії так чи інакше пов'язана з цифровою обробкою сигналів.

Серед методів приховування інформації в зображеннях найпоширенішими є категорія алгоритмів з використанням молодших бітів даних зображення. Вони засновані на тому факті, що в деяких форматах файлів молодші біти значень хоча і присутні у файлі, але не впливають на сприйняття людиною звуку або зображення. На цьому ж принципі засновано і стиснення з втратами (формати JPEG, MP3, MP4 та інші). Саме у таких місцях файлів можна зберігати повідомлення. Найчастіше контейнерами служать графічні формати з прямим кодуванням в 24 і більше бітів на піксель (формати BMP, TIFF). Рідше – звукові файли з абсолютним кодуванням амплітуди аудіосигналу (формат WAV). Щодо кольорових графічних зображень, замінювати молодші біти можна у кожній із складових кольору пікселя: R, G, B або C, M, Y, K. При цьому необхідно уникати зображень з великими яскравими областями, адже байти приховуваних файлів можуть відрізнятися від фону. Тож, для більшої надійності приховування слід використовувати зображення з великою кількістю півтонів та відтінків.

Результати досліджень. Для дослідження ми обрали стеганографічний програмний засіб S-Tools. З метою аналізу особливостей розміщення стегоданих у файлах-контейнерах було створено два тестові монотонні зображення розміром 50×50 пікселів у 24-бітному форматі bmp. Для дослідження ми обрали зображення чорного та білого кольорів. У кожному із зображень було приховано текстовий файл (рис. 1), після чого виконано обернену дію – видобування файлу (рис.2). В результаті приховування одержали два стего-файли (рис. 3).

Зауважимо, що для успішного видобування інформації, окрім володіння файлом із прихованим вмістом, необхідно знати пароль, що було вказано під час створення стегоконтейнера.

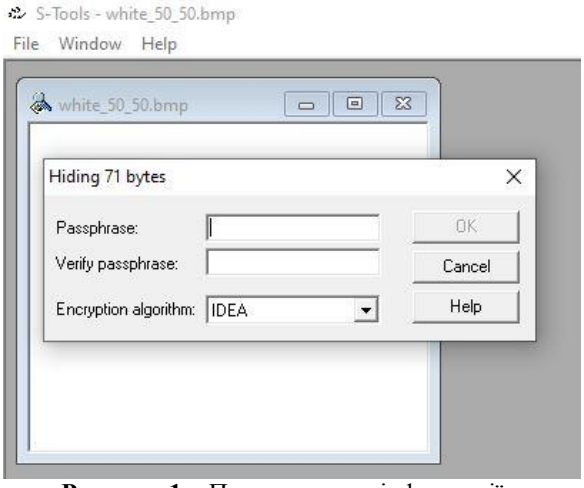

Рисунок 1 – Приховування інформації у файлі-контейнері

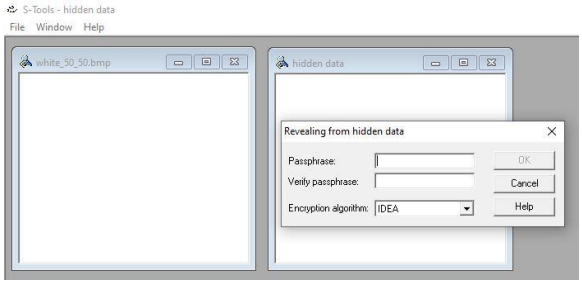

Рисунок 2 – Процедура видобування прихованого документа

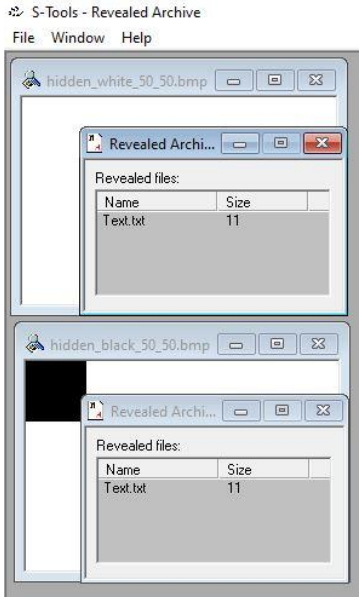

Рисунок 3 – Приховуване повідомлення у файлах монотонних зображень

Зазначимо, що процедура видобування прихованого файлу засвідчила збереження даних у стегофайлі без втрат (рис. 4). У верхній частині рисунка наведено вміст файлу, що приховувався, а нижче – вміст файлу, що було отримано із зображення.

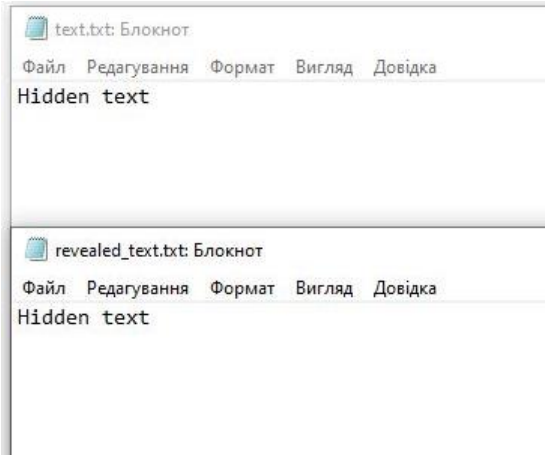

Рисунок 4 – Вміст текстового файлу після видобування

Переглянемо двійковий вміст початкових зображень та файлів, що містять приховані дані. Отже, для порівняння наведено двійковий вміст зображення чорного квадрата (рис. 5) та вміст стегоконтейнера із прихованим текстовим файлом (рис. 6). Зазначимо, що вміст контейнера та стегофайлу наведено лише частково, проте адреси комірок пам'яті було обрано відповідні (рис. 5-6). У правому стовпці наведено вміст комірок пам'яті у шістнадцятковому форматі. Байти, що відображають колір квадрата, мають значення «00», оскільки початкове зображення містить лише чорний колір. Можна помітити, що вміст комірок, які відповідають за зображення, змінився після приховання додаткових даних (це відображають комірки із значеннями «01» на рисунку 6).

Повторимо цю ж процедуру для зображення білого квадрата розміром 50 на 50 пікселів. Відповідно, в оригінальному файлі байти, що відображають колір, міститимуть значення «FF». Відмінність між файлом контейнером (рис. 7) та стегофайлом (рис. 8) у випадку білого кольору є подібною до попереднього випадку (модифіковані комірки містять значення «FE»).

Таким чином, для файлів контейнерів із зображенням чорного та білого квадратів розміром 50 на 50 пікселів було виконано приховування 11 байт даних. Розмір початкових файлів та стего-файлів є однаковим і становить 7654 байт. Максимальна кількість інформації для приховування у цьому випадку становить 922 байти. З наведеного двійкового вмісту зображень можна зробити висновки, що значення комірок пам'яті змінюються з мінімальним приростом значення (значення чорного кольору було змінене із «00» на «01», а білого – із «FF» на «FE»).

Також, на цьому прикладі ми переконались, що однотонні зображення не варто використовувати для приховування в них інших файлів,

адже факт присутності додаткового вмісту досить легко виявити, на відміну від зображень, що містять багато кольорів та тонів.

Offset(h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
00000450	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000460	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000470	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000480	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000490	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000004A0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000004B0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000004C0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000004D0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000004E0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000004F0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000500	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000510	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000520	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000530	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000540	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000550	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000560	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000570	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000580	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000590	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000005A0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000005B0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000005C0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000005D0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000005E0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000005F0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000600	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000610	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000620	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000630	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000640	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000650	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000660	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000670	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00

Рисунок 5 – Двійковий вміст файлу-контейнера

Offset(h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
00000450	00	00	00	00	00	00	00	00	00	00	00	00	00	00	01	00
00000460	00	00	00	00	01	00	00	00	00	00	00	00	00	00	00	01
00000470	00	00	00	00	00	00	00	00	00	00	00	00	00	00	01	01
00000480	00	00	00	00	00	00	00	00	00	00	01	00	00	00	00	00
00000490	00	00	00	00	00	00	00	00	01	00	00	00	00	00	00	00
000004A0	00	00	00	01	00	00	00	00	00	00	00	00	00	00	00	00
000004B0	00	00	00	00	01	00	00	00	00	00	01	00	00	00	00	00
000004C0	00	00	00	00	00	00	00	00	00	00	00	00	00	01	00	00
000004D0	00	01	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000004E0	00	00	01	00	00	00	00	00	00	00	00	00	00	00	00	00
000004F0	00	00	00	00	00	00	00	00	00	00	01	00	00	00	00	00
00000500	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000510	00	00	00	00	00	00	01	00	00	00	00	00	00	00	00	00
00000520	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000530	00	00	00	00	00	00	00	00	00	00	00	00	00	01	00	00
00000540	00	00	01	00	00	00	00	00	00	01	00	00	00	00	00	00
00000550	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000560	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000570	00	00	01	00	00	00	00	00	00	00	00	00	00	01	00	00
00000580	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000590	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000005A0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000005B0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000005C0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000005D0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	01
000005E0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000005F0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000600	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000610	00	00	00	00	00	01	00	00	00	00	00	00	01	00	00	00
00000620	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000630	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	01
00000640	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000650	00	00	00	00	00	00	00	00	00	00	00	01	00	00	00	00
00000660	00	00	00	00	00	00	00	00	01	00	00	00	00	00	00	01
00000670	00	00	00	00	00	00	00	00	00	00	00	01	00	00	00	00

Рисунок 6 – Двійковий вміст стегофайлу

Offset(h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
00000450	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
00000460	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
00000470	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
00000480	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
00000490	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
000004A0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
000004B0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
000004C0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
000004D0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
000004E0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
000004F0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
00000500	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
00000510	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
00000520	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
00000530	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
00000540	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
00000550	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
00000560	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
00000570	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
00000580	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
00000590	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
000005A0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
000005B0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
000005C0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
000005D0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
000005E0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
000005F0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
00000600	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
00000610	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
00000620	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
00000630	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
00000640	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
00000650	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
00000660	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
00000670	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF

Рисунок 7 – Двійковий вміст файлу-контейнера

Offset(h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
00000450	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FE	FF
00000460	FF	FF	FF	FF	FE	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
00000470	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FE
00000480	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FE	FF	FF	FF	FF	FF
00000490	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
000004A0	FF	FF	FF	FF	FF	FF	FF	FF	FE	FF	FF	FF	FF	FF	FF	FF
000004B0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
000004C0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FE	FF	FF
000004D0	FF	FE	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
000004E0	FF	FF	FE	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
000004F0	FF	FF	FF	FF	FF	FF	FF	FF	FE	FF	FF	FF	FF	FF	FF	FF
00000500	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
00000510	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
00000520	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
00000530	FF	FF	FF	FF	FF	FE	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
00000540	FF	FF	FE	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
00000550	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
00000560	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
00000570	FF	FF	FF	FF	FE	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
00000580	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FE	FF	FF	FF	FF
00000590	FF	FF	FF	FF	FE	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
000005A0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
000005B0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
000005C0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
000005D0	FF	FF	FF	FF	FE	FF	FF	FF	FF	FF	FF	FF	FE	FF	FF	FE
000005E0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
000005F0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FE	FF	FF	FF	FF	FF
00000600	FF	FF	FF	FF	FF	FF	FE	FF	FF	FF	FF	FF	FF	FF	FF	FF
00000610	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FE	FF	FF	FF
00000620	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
00000630	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FE
00000640	FE	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
00000650	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
00000660	FF	FF	FF	FE	FF	FF	FF	FE	FF	FF	FF	FF	FF	FF	FF	FF
00000670	FF	FF	FF	FF	FF	FF	FE	FF	FF	FF	FF	FF	FF	FF	FF	FF

Рисунок 8 – Двійковий вміст стегофайлу

Приховування групи різнотипних файлів. Виконаємо одночасне приховування групи різнотипних файлів. Зокрема у файл-контейнер зображення було успішно поміщено інший файл зображення (.bmp), аудіо файл (.wav), виконуваний файл (.exe). Ці результати засвідчують можливість приховування будь-яких двійкових даних (незалежно від формату) стеганографічним засобом S-Tools. Порівнюючи візуальний вигляд початкового зображення (рис. 9) та стегоконтейнера (рис. 10), можемо переконатися, що подібні

зображення доцільніше застосовувати у якості стегофайлів, аніж монотонні. Також варто зазначити, що двійковий вміст зображення не повинен однозначно вказувати на наявність додаткового вмісту, тому доцільніше обирати оригінальні та нетривіальні зображення.

Рисунок 9 – Файл зображення (контейнер)

Рисунок 10 – Файл зображення з прихованим документом (стегофайл)

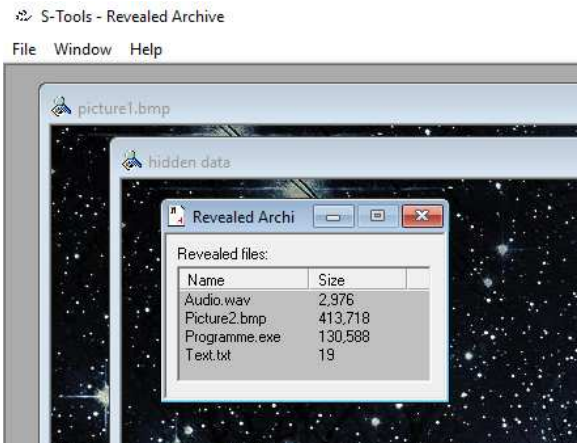

Рисунок 11 – Процедура видобування групи різнотипних файлів

Розглянемо відмінність двійкового вмісту файлу-контейнера та стегофайлу у випадку приховування великої кількості даних у немонотонному зображенні. Для порівняння наведено вміст початкового файлу (рис. 12) та вміст вихідного файлу (рис. 13). Можемо зробити висновки, що приховані дані, як і у попередньому випадку, розміщуються рівномірно по всьому файлу і вміст комірок модифікується мінімально.

Offset(h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
0033C280	01	01	00	00	00	00	00	00	00	01	00	01	04	02	02	03
0033C290	01	02	00	00	04	01	00	08	01	00	20	19	16	39	2D	2B
0033C2A0	31	26	22	1B	0F	0B	09	01	00	19	11	0A	25	1B	14	2A
0033C2B0	20	19	2F	23	1F	2C	20	1C	30	21	1E	3C	2D	2A	2E	21
0033C2C0	1F	1C	0F	0D	1A	10	10	22	18	18	14	0F	0E	0B	06	05
0033C2D0	0C	08	07	10	0B	08	18	12	0D	1B	12	0E	18	14	0F	15
0033C2E0	10	0D	0A	06	05	04	00	00	06	04	04	11	0F	0F	13	11
0033C2F0	11	16	14	14	07	04	00	0D	0A	05	16	13	0B	20	1B	12
0033C300	12	0F	01	32	29	1F	45	3A	36	38	2B	29	2F	23	21	30
0033C310	24	22	2D	21	1F	29	1D	1B	2F	26	23	3F	36	33	1E	17
0033C320	14	1D	16	13	14	0F	0C	0C	07	04	0E	09	06	18	13	10
0033C330	1E	1B	17	1F	1A	17	15	11	0C	19	13	0E	1E	18	13	07
0033C340	01	00	0D	04	00	0A	01	00	1E	14	0D	1A	10	09	1E	14
0033C350	0D	22	18	11	27	1B	15	2D	21	1B	37	29	23	3C	2E	28
0033C360	37	29	23	2E	20	1A	38	27	24	2C	1B	18	30	1D	18	37
0033C370	23	1E	2A	16	11	34	1E	19	9F	89	84	3A	24	1F	34	20
0033C380	1B	3D	29	24	42	31	2E	3E	2F	2C	2D	21	1D	1B	10	0C
0033C390	0A	03	00	07	00	00	0A	00	04	0E	01	09	0D	06	0B	0C
0033C3A0	06	07	0E	0A	09	1D	19	14	34	2D	24	48	3C	32	5B	4B
0033C3B0	3E	65	52	45	8C	77	68	45	32	23	24	14	07	E8	DD	CF
0033C3C0	B5	B2	A3	14	12	07	09	06	01	10	0B	0A	1B	17	12	36
0033C3D0	2D	29	22	18	11	39	2B	25	3E	2E	27	3D	2B	24	3D	2B
0033C3E0	24	2E	1E	17	36	28	22	4B	41	3A	43	3A	36	1E	17	14
0033C3F0	0A	07	03	13	0F	0E	12	0D	0C	1D	18	17	2E	29	28	2D
0033C400	28	27	17	12	11	07	02	01	07	02	01	0B	06	05	1C	17
0033C410	16	19	14	13	26	21	20	2E	29	28	23	1B	1B	17	0F	0F
0033C420	19	11	11	1C	14	14	16	12	0D	13	0F	0A	1A	14	0F	19
0033C430	13	0E	10	0A	05	15	0F	0A	25	1C	18	29	20	1C	39	30
0033C440	2C	23	1A	16	12	0C	07	15	0F	0A	15	0F	0A	0D	07	02
0033C450	07	03	00	0C	08	03	1B	15	0E	1F	19	12	1A	14	0D	14
0033C460	0E	07	2E	2B	23	11	0E	06	34	31	29	36	33	2B	1B	18
0033C470	10	1F	1C	14	09	05	00	05	01	00	05	01	00	0C	08	03
0033C480	26	22	1D	29	22	1F	36	2F	2C	27	1B	1B	25	1A	16	38
0033C490	29	26	38	2A	24	2A	1A	14	2F	1D	16	42	30	29	31	1F
0033C4A0	18	2B	19	12	25	15	0F	28	18	12	1C	10	0C	0C	00	00
0033C4B0	1A	11	0E	48	3E	3E	0E	0C	0C	02	00	00	0C	0A	0A	0C

Рисунок 12 – Двійковий вміст файлу-контейнера

Offset(h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
0033C280	01	00	00	00	01	01	01	00	00	01	01	01	05	03	03	02
0033C290	00	02	00	00	04	00	00	08	01	00	20	18	16	38	2C	2A
0033C2A0	31	26	22	1A	0E	0A	09	00	00	19	10	0A	25	1B	14	2A
0033C2B0	20	19	2E	23	1E	2C	21	1C	31	20	1F	3C	2D	2A	2E	21
0033C2C0	1F	1C	0E	0D	1B	10	11	23	18	19	15	0E	0E	0B	07	04
0033C2D0	0C	08	07	10	0B	08	18	13	0C	1A	13	0E	18	14	0E	14
0033C2E0	10	0C	0A	06	05	04	00	01	06	04	04	10	0F	0F	13	11
0033C2F0	11	16	14	14	06	05	00	0C	0B	05	17	13	0B	20	1B	12
0033C300	12	0F	01	33	28	1F	44	3B	37	38	2A	28	2F	22	20	30
0033C310	24	23	2C	21	1E	28	1D	1A	2F	27	23	3F	36	33	1F	17
0033C320	14	1D	17	13	14	0F	0C	0D	07	05	0E	09	07	18	13	10
0033C330	1E	1B	17	1F	1B	16	14	11	0D	18	12	0E	1E	18	13	07
0033C340	01	00	0D	04	00	0A	00	01	1F	14	0D	1A	11	09	1F	14
0033C350	0D	23	19	11	27	1B	15	2D	20	1A	37	28	22	3C	2E	28
0033C360	37	28	23	2F	20	1A	38	27	24	2D	1A	18	30	1D	18	36
0033C370	22	1E	2A	17	10	35	1E	19	9F	89	84	3A	25	1E	35	21
0033C380	1A	3C	28	25	42	30	2F	3E	2F	2C	2C	20	1D	1B	10	0D
0033C390	0A	02	00	07	00	00	0B	00	04	0E	01	09	0D	06	0A	0D
0033C3A0	07	07	0F	0A	09	1C	19	14	35	2C	25	48	3C	33	5B	4A
0033C3B0	3F	65	53	44	8C	77	69	44	33	23	25	15	07	E8	DD	CE
0033C3C0	B5	B2	A3	15	12	06	09	06	01	10	0B	0B	1A	17	12	37
0033C3D0	2C	28	22	18	10	38	2A	25	3E	2F	26	3C	2A	25	3C	2A
0033C3E0	24	2E	1E	17	36	29	22	4A	41	3A	42	3B	36	1E	16	15
0033C3F0	0A	07	02	13	0F	0E	12	0C	0C	1D	18	17	2E	28	29	2C
0033C400	29	26	16	12	11	06	03	00	06	02	01	0B	06	05	1C	17
0033C410	16	18	14	13	27	20	20	2F	29	29	22	1A	1A	16	0E	0E
0033C420	18	11	11	1C	15	15	16	13	0D	12	0F	0B	1B	14	0F	19
0033C430	12	0E	10	0B	05	15	0E	0A	24	1C	18	28	21	1D	39	31
0033C440	2C	22	1A	16	12	0C	07	15	0F	0A	15	0F	0A	0C	06	02
0033C450	07	03	01	0C	09	02	1B	14	0E	1E	18	12	1A	14	0D	14
0033C460	0E	06	2E	2B	22	11	0F	07	34	30	28	36	32	2A	1B	18
0033C470	10	1F	1C	15	09	05	01	05	01	01	04	01	00	0D	08	03
0033C480	26	22	1C	29	23	1E	37	2F	2C	26	1B	1B	24	1A	17	39
0033C490	28	26	39	2A	24	2B	1B	15	2E	1D	16	43	30	28	30	1E
0033C4A0	19	2B	19	13	25	14	0F	29	18	12	1C	11	0C	0D	00	00
0033C4B0	1B	10	0F	48	3E	3F	0E	0D	0C	03	01	00	0C	0B	0A	0D

Рисунок 13 – Двійковий вміст стегофайлу

Обговорення результатів досліджень. Отже, під час проведеного дослідження було встановлено, що у файлі зображення (1920×1080 пікселів) обсягом 6 220 854 байт можна приховати 777 584 байт інформації. Процедура видобування прихованих файлів (рис. 11) засвідчила збереження даних у стегофайлі без втрат. Співвідношення між розміром файлу із зображенням і розміром файлу, який можна приховати, залежить від

конкретного випадку, однак при правильному виборі файлу-контейнера, факт використання стеганографічних засобів (не знаючи секретний ключ) встановити і довести практично неможливо. Якщо ж скористатись компресією зображення і приховувати не сам файл, а його архів, то у зображенні меншого розміру можна приховати зображення більшого розміру.

Висновки. При виконанні цієї роботи ми переконались, що при застосуванні стеганографії програми використовують певні алгоритми, які приховують секретні дані серед вмісту контейнера: біти початкового файлу у випадкових позиціях замінюються на біти приховуваного файлу. Таким чином, розмір початкового файлу і файлу-контейнера (що містить вкладену інформацію) є однаковим, навіть за умови приховування різної кількості файлів або різного обсягу даних.

Тим часом, для людини не є можливим визначити, чи були використані засоби стеганографії під час створення певного файлу. При вдалому наповненні файлу стегоданими незаповнений контейнер від заповненого не зможе відрізнити без спеціального аналізу навіть досвідчений фахівець. Адже для цього необхідне застосування спеціалізованого програмного забезпечення, проте з причини низької швидкодії воно не може бути використане в промислових об'ємах, а антивірусні програми не виявляють факту застосування стеганографічних засобів.

Список літератури

1. Грибунин В.Г., Оков И.Н., Туринцев И.В. Цифровая стеганография. Москва : СОЛОН-Пресс, 2002. 272 с.

2. Конахович Г.Ф., Прогонов Д.О., Пузиренко О.Ю. Комп'ютерна стеганографічна обробка й аналіз мультимедійних даних. Київ : Центр навчальної літератури, 2018. 558 с.

3. Конахович Г.Ф., Пузиренко А.Ю. Комп'ютерна стеганографія. Теорія і практика. Київ : МК-Пресс, 2006. 288 с.

4. Лагун А.Е., Полотай О.І. Особливості приховування інформації в зображеннях з використанням молодшого значущого біта. *Вісник ЛДУБЖД*, Львів, 2019, Вип. 20. С. 17-22.

5. Мельник С.В., Кащук В.І. Методи цифрової стеганографії: стан та напрями розвитку. *Інформаційна безпека людини, суспільства, держави,* Київ, 2013. Вип. 3. С. 65–70.

6. Шелест М.Є., Андреєв В.І. Комп'ютерна стеганографія та її можливості. *Сучасна спеціальна техніка*. Київ, 2011. Вип. 24. С. 97–104.

References

1. Gribunin V.G., Okov I.N., Turintsev I.V. Digital steganography. Moscow : SOLON-Press, 2002. 272 p.

2. Konakhovich G.F., Progonov D.O., Puzirenko O.Y. Computer steganographic processing and analysis of multimedia data. Kyiv : Center for Educational Literature, 2018. 558 p.

3. Konakhovich G.F., Puzirenko A.Yu. Computer steganography. Theory and practice. Kyiv : MK-Press, 2006. 288 c.

4. Lagun A.E., Polotai O.I. Features of hiding information in images using the least significant bit. Bulletin of the LDUBZhD, Lviv, 2019, Issue. 20. pp. 17-22.

5. Melnik S.V., Kashchuk V.I. Methods of digital steganography: state and directions of development. Information security of man, society, state, Kyiv, 2013. Issue. 3. S. 65–70.

6. Shelest M.E., Andreev V.I. Computer steganography and its possibilities. Modern special equipment. Kyiv, 2011. Issue. 24. pp. 97–104.

*** Науково-методична стаття**