УДК 004.056.55

Демашкевич А. В., здобувач 2 курсу спеціальності 113 Прикладна математика,

Антонов Ю. С., канд. фіз.-мат. наук, доцент, доцент кафедри інформаційних технологій

РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ ПРИХОВУВАННЯ ПОВІДОМЛЕННЯ В ЦИФРОВИХ ЗОБРАЖЕННЯХ ЗА ДОПОМОГОЮ МЕТОДУ LSB

Донецький національний університет імені Василя Стуса, м. Вінниця

У сучасну цифрову епоху важливого значення набуває питання захисту інформації від несанкціонованого використання та розповсюдження, коли потрібен цифровий аналог водяних знаків. У деяких ситуаціях необхідно передавати зашифровані повідомлення відкритими каналами зв'язку, приховуючи їх від інших, або навпаки, потрібно мати можливість виявляти такі повідомлення. Через це виникає необхідність вдосконалення засобів приховування повідомлень. Серед наявних засобів перевагу має стеганографія, оскільки інформація може міститися у файлах мультимедіа, які не сприймаються як носії тексту. В Україні методи стеганографії для захисту конфіденційності можуть використовувати і військові, і цивільні особи.

Стеганографією називають сукупність методів та засобів їх реалізації, які базуються на різних принципах і дають змогу приховувати сам факт існування секретної інформації в тому чи іншому середовищі [4]. Методи комп'ютерної стеганографії використовують комп'ютерну техніку та програмне забезпечення для приховування інформації в потоках оцифрованих сигналів. В. О. Денисюк комп'ютерною стеганографією називає фактичне приховування одного файла в іншому [1]. Метою стеганографії є захист інформації від несанкціонованого використання шляхом вбудовування секретних повідомлень в інші дані, відомі як контейнери [2]. Це забезпечує неможливість візуального або технологічного доступу до повідомлень [3].

Аналіз сучасних стеганографічних програм продемонстрував, що вони лише частково відповідають набору вимог. Були виявлені наступні недоліки: використання одного контейнера для приховування одного повідомлення, неможливість приховати великий об'єм інформації без видимих аномалій у контейнері, розробка програм лише на базі одного методу стеганографії без додаткових заходів захисту. Загалом серед тенденцій у комп'ютерній стеганографії можна виділити розповсюджене використання статичних цифрових зображень в якості контейнерів, нестачу надійних програмних стеганографічних засобів у вітчизняному

просторі. Розробка програмного забезпечення на основі стеганографічного методу для приховування повідомлень є досить актуальним завданням.

Перевагою обраного методу LSB (Least Significant Bit) є універсальність та простота, оскільки його можна використовувати в програмах з різними типами контейнерів, як-от цифрові зображення, аудіофайли та відеофайли. Він є поширеним серед методів заміни в просторовій множині. Сутність цього методу полягає в заміні найменш значимих бітів у контейнері (зображення, аудіо- або відеозапису) на біти приховуваного повідомлення. Різниця між порожнім і заповненим контейнерами непомітна для органів чуття людини [1]. Зважаючи на те, що кожна точка в RGB-зображенні кодується кількома байтами, кожен байт визначає інтенсивність червоного (Red), зеленого (Green) і синього (Blue) кольору. Сукупність інтенсивностей кольору в кожному з 3-х каналів визначає відтінок пікселя. Найменш значимі розряди меншою мірою впливають на підсумкове зображення. З цього можна зробити висновок, що заміна одного або двох молодших, найменш значимих бітів, на інші довільні біти мало спотворить відтінок пікселя (рис. 1).

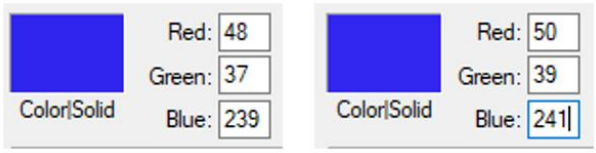

Рисунок 1. Порівняння двох відтінків

Розглянемо ключовий алгоритм програмного засобу, а саме приховування повідомлення в цифрових зображеннях. Колірна компонента кожного пікселя описується одним байтом, модифікації підлягає останній біт. Алгоритм розподілу повідомлення між контейнерами базується на пропорційному підході. У разі використання в одному наборі кількох зображень, різних за розміром, у менший контейнер програма розподілить меншу кількість інформації, відповідно в більший – більший обсяг інформації. Формула (1) описує знаходження коефіцієнта пропорційності, формула (2) визначає довжину підрядка для контейнера:

$$k = \frac{\sum_{i=1}^{n} l_i}{l_t}; \quad (1)$$

$$l_{t_i} = \frac{l_i}{k}, \quad (2)$$

де n – кількість контейнерів; l_i – максимально допустима довжина повідомлення i-му контейнері; l_t – довжина вихідного повідомлення; l_{t_i} – довжина частини повідомлення в i-му контейнері, k – коефіцієнт.

У заповненому контейнері 15 пікселів першого рядка містять системну інформацію, а саме: наявність прихованого повідомлення, порядковий номер частини повідомлення, загальну кількість частин вихідного повідомлення, довжину прихованого повідомлення в контейнері. Максимальний обсяг повідомлення для одного контейнера розраховується за кількістю його пікселів.

Розглянемо алгоритм приховування повідомлень у контейнери (рис. 2). Основними кроками є вибір та завантаження пустих контейнерів та повідомлення. Додатковим рівнем захисту є наявність ключа шифрування. Використовуючи стандартну бібліотеку для криптографічних операцій System.Security.Cryptography, вихідний текст шифрується.

Рисунок 2. Блок-схема приховування повідомлення

Процес вилучення повідомлення будується в аналогічний спосіб. Спочатку проводиться завантаження контейнерів, далі вводиться ключ шифрування, частини повідомлення з різних контейнерів об’єднуються, текст дешифрується, отримуються дані з контейнерів. Якщо під час перевірки контейнерів програма

не виявила жодного заповненого з наявних, тобто завантажені пусті зображення, користувач отримує повідомлення про відсутність прихованої інформації.

На основі викладеного алгоритму розроблено програмне забезпечення на мові програмування C# (рис. 3). У процесі проєктування використана модель багатошарової архітектури з урахуванням особливостей впровадження та масштабування у майбутньому. У програмі під час вибору контейнерів бажано використовувати зображення форматів PNG та BMP.

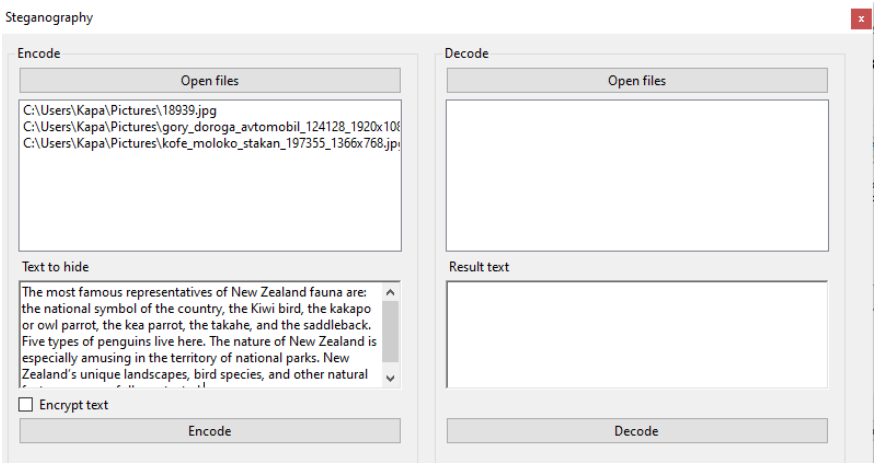

Рисунок 3. Головне вікно програми

На рис. 4 показані контейнери для зрівняння, ліворуч – незаповнений контейнер, праворуч – заповнений. Як бачимо, заповнений контейнер візуально майже не відрізняється від пустого.

Рисунок 4. Порівняння пустого та заповненого контейнерів

Отже, проблему оптимального приховування повідомлень у серії зображень за допомогою методу LSB можна вважати вирішеною.

Список використаних джерел

1. Хорошко В. О., Яремчук Ю. Є., Карпінець В. В. Комп’ютерна стеганографія: навч. посіб. Вінниця: ВНТУ, 2017. 155 с.

2. Денисюк В. О. Стеганографічний алгоритм захисту даних з використанням файлів зображень. *Ефективна економіка.* 2017. № 5. URL: http://www.economy.nayka.com.ua/?op=1&z=5584 (дата звернення: 30.10.2023).

3. Кошкіна Н. В. Стегоаналіз цифрових зображень із застосуванням контрольного вкраплення. *Захист інформації і безпека інформаційних систем:* матеріали III міжнародної науково-технічної конференції. Львів, 2014. С. 98–100.

4. Римар П. В., Крохмалюк В. В. Атаки на стеганосистеми. Криптографічні атаки. *Матеріали наукової конференції професорсько-викладацького складу, наукових працівників і здобувачів наукового ступеня за підсумками науково-дослідної роботи за період 2019–2020 рр.* (квітень–травень 2021 р.). Вінниця: ДонНУ імені Василя Стуса, 2021. С. 344–346.

УДК 004.9+005.5

Колосова К. К., здобувач 1 курсу ОС «Магістр» спеціальності 122 Комп'ютерні науки,

Потапова Н. А., канд. екон. наук, доцент, доцент кафедри інформаційних технологій

РОЛЬ SEO У ПРОСУВАННІ ТА ПІДВИЩЕННІ ВІДВІДУВАНОСТІ ВЕБСАЙТІВ ІТ-КОМПАНІЙ

Донецький національний університет імені Василя Стуса, м. Вінниця

SEO-просування – це сукупність методів оптимізації вебсайта з метою підвищення його видимості в пошукових системах. Це означає, що відповідні запити користувачів, які стосуються сайта, будуть відображатися на вищих позиціях у результатах пошуку. Оптимізація вебсайта включає внутрішні та зовнішні елементи, які впливають на його рейтинг у пошукових системах.

Просування пошукових систем (SEO) дає змогу збільшити кількість відвідувачів сайта, залучити більше потенційних клієнтів і підвищити конверсію. Оскільки він підвищує вплив бренду, збільшує продажі та покращує конкурентоспроможність на ринку, це важливий інструмент для будь-якого бізнесу [1].

Основні аспекти SEO для ІТ-компаній включають технічну оптимізацію вебсайта (швидкість завантаження, мобільна сумісність, правильна структура URL), вибір правильних ключових слів та створення відповідного контенту, побудову якісних посилань з авторитетних джерел, а також постійний аналіз і вдосконалення стратегій SEO на основі вебаналітики та даних про користувачів. Ці аспекти є важливими для підвищення видимості в пошукових системах та привертання цільового трафіка на вебсайт ІТ-компанії.