DOI: 10.20535/kpi-sn.2020.1.197952
УДК 004.62:004.056.53+004.056.55:004.421.5

Є.С. Сулема*, Є.О. Радченко
КПІ ім. Ігоря Сікорського, Київ, Україна
*corresponding author: sulema@pzks.fpm.kpi.ua

МЕТОД СТЕГАНОГРАФІЧНОГО ЗАХИСТУ МУЛЬТИМЕДІЙНИХ ДАНИХ НА ОСНОВІ ПРОЦЕДУРИ ПСЕВДОВИПАДКОВОГО ВБУДОВУВАННЯ

Проблематика. Класичний стеганографічний захист даних ґрунтується на приховуванні самого факту існування секретних даних. Поки цей факт не розголошено, дані залишаються захищеними. Проте в переважній більшості випадків такого захисту недостатньо для гарантування безпеки даних. Тому звичайною практикою є комбінування шифрування даних і стеганографічного приховування. Останнім часом такий підхід використовується у дедалі ширшому колі застосувань, у т.ч. при захисті операцій електронної комерції. Отже, задача розроблення нових, більш складних способів крипто-стеганографічного захисту даних залишається актуальною.
Мета дослідження. Основною метою є забезпечення захисту мультимедійних даних користувача за рахунок використання складеного приватного ключа для обмеження доступу до цих даних. Для досягнення цієї мети у дослідженні ставиться задача розробити стеганографічний метод захисту мультимедійних даних через їх вбудовування у псевдовипадкові семпли аудіофайлу.
Методика реалізації. Запропонований метод є методом комбінованого, крипто-стеганографічного захисту даних. Він ґрунтується на використанні складеного приватного ключа. У методі використовуються два генератори псевдовипадкових чисел. Перший генератор використовується у процедурі перемішування блоків секретних даних. Другий генератор використовується у процедурі стеганографічного вбудовування для вибору чергового семпла для вбудовування секретних даних. Стеганографічне вбудовування секретних даних виконується за принципом різницевого перетворення бітів із використанням результату логічних операцій. Метод використовує природні властивості аудіоданих, а саме їх надлишковість.
Результати дослідження. Розроблено метод стеганографічного захисту мультимедійних даних на основі процедури псевдовипадкового вбудовування, який включає три основних етапи. На першому етапі відбувається попередня підготовка даних. На другому етапі формується порядок розміщення блоків секретних даних при їх вбудовуванні. На третьому етапі відбувається стеганографічне вбудовування секретних даних у контейнер. Запропоновано процедури оброблення даних, які дають можливість підвищити рівень захисту за рахунок псевдовипадкового перемішування та вбудовування даних.
Висновки. Розроблений стеганографічний метод захисту мультимедійних даних передбачає використання аудіоданих як контейнера для стеганографічного вбудовування секретних даних і застосування процедур оброблення даних, які ґрунтуються на алгоритмах генерації псевдовипадкових чисел. Аналіз стійкості запропонованого методу до несанкціонованого доступу до прихованих даних показав, що ймовірність вгадування стеганографічної схеми є малою. Запропонований метод може використовуватись у випадках, коли стеганографічний захист вважається більш доцільним, ніж криптографічний.

Ключові слова: захист даних; стеганографія; псевдовипадкові послідовності; логічні операції.

Вступ

Класичний стеганографічний захист даних [1, 2] ґрунтується на приховуванні самого факту існування секретних даних. Поки цей факт не розголошено, дані залишаються захищеними. Проте в переважній більшості випадків такого захисту недостатньо для гарантування безпеки даних. Тому звичайною практикою є комбінування шифрування даних і стеганографічного приховування [1–7]. Останнім часом такий підхід використовується у дедалі ширшому колі застосувань, у т.ч. при захисті операцій електронної комерції [4].

Оскільки немає теоретичних підтверджень того, що відомі способи стеганографічного захисту даних повною мірою вирішили проблему захисту даних від несанкціонованого доступу, задача розроблення нових, більш складних способів крипто-стеганографічного захисту даних залишається актуальною. При розробленні нових методів перевага в частині шифрування надається способам [3–7], що відносяться до так званої "легковагової" криптографії (Lightweight Cryptography) і програмна реалізація яких не вимагає суттєвих обчислювальних ресурсів. Тому в досліджені, результати якого наводяться в цій статті, увагу приділено розробленню саме алго-

ритмів шифрування, що можуть бути віднесені до класу "легковагової" криптографії, та їх комбінуванню зі стеганографічним принципом вбудовування секретних даних у файл-контейнер.

Постановка задачі

Метою роботи є забезпечення захисту мультимедійних даних користувача за рахунок використання складеного приватного ключа для обмеження доступу до цих даних. Для досягнення цієї мети у роботі ставиться задача розробити стеганографічний метод захисту мультимедійних даних через їх вбудовування у псевдовипадкові семпли аудіофайла.

Формулювання методу

Запропонований метод є методом комбінованого, крипто-стеганографічного захисту даних. Він ґрунтується на використанні складеного приватного ключа, який включає п'ять компонентів: K1 – довільне додатне число, що використовується як "зерно" для першого генератора псевдовипадкових чисел; K2 – довільне додатне число, що використовується як "зерно" для другого генератора псевдовипадкових чисел; K3 – довільний файл, що використовується для шифрування даних при їх вбудовуванні; K4 – логічна функція, що використовується для шифрування даних при їх вбудовуванні; K5 – розмір секретних даних у байтах.

У методі використовуються два генератори псевдовипадкових чисел. Перший генератор використовується у процедурі перемішування блоків секретних даних. Цей генератор ґрунтується на алгоритмі методу Mersenne Twister [8]. Другий генератор використовується у процедурі стеганографічного вбудовування для вибору чергового семпла для вбудовування секретних даних. Він ґрунтується на алгоритмі базового методу Linear Congruential Generator [9].

Стеганографічне вбудовування секретних даних виконується за принципом різницевого перетворення бітів [10] із використанням результату логічних операцій [11]. Метод використовує природні властивості аудіоданих, а саме їх надлишковість.

За контейнер у методі береться довільний аудіофайл у форматі WAVE [12]. Можливе також використання формату FLAC [13]. Секретні дані вбудовуються у наймолодші n бітів кожного семпла, де n – кількість байт на один семпл одного каналу. Оскільки найбільш поширеними є wav-файли з двома каналами, де кожний семпл одного каналу описується двома байтами, то при реалізації пропонованого методу було взято $n = 2$, таким чином, стеганографічне вбудовування відбувалось у 2 наймолодших біти семпла.

Розглянемо основні етапи методу.

Перший етап є підготовчим. Вхідними даними на цьому етапі є файл із секретними мультимедійними даними й аудіофайл, який планується використовувати як контейнер.

Спочатку відбувається зчитування метаданих контейнера. У метаданих wav-файлів зберігається інформація про розмір аудіофайла, формат, частоту дискретизації, бітрейт (кількість байт, переданих за одну секунду), кількість байт для одного семпла, ім'я виконавця, назву аудіофайла тощо. Після отримання метаданих контейнер перевіряється на відповідність таким критеріям:

– формат;
– кількість каналів;
– кількість байтів на семпл в одному каналі.

Далі дані контейнера та секретні дані зчитуються та конвертуються у масив байт. Після цього перевіряється можливість вбудовування секретних даних у контейнер. Максимальна кількість байт l, яку можна вбудувати у контейнер, становить $l = (k - m) / 8$, де k – розмір файла-контейнера, m – розмір метаданих. Якщо секретні дані вбудувати неможливо, то має бути вибираний інший контейнер.

Вбудовування секретної інформації в контейнер відбувається у блок даних, що розміщений після метаданих і впорядкований таким чином: перший семпл першого каналу, перший семпл другого каналу, другий семпл першого каналу, другий семпл другого каналу тощо.

На другому етапі методу відбувається формування порядку розміщення блоків секретних даних перед їх вбудовуванням у контейнер.

Вхідними даними на цьому етапі є масив байтів із секретними даними (масив S розмірності s), масив даних контейнера (масив B розмірності b; даними контейнера є семпли; b – це кількість семплів у аудіофайлі, що використовується як контейнер) і значення компонента K1 складеного приватного ключа.

Для ускладнення можливого стегоаналізу секретні дані перед вбудовуванням перемішуються. Для цього використовується перший генератор псевдовипадкових чисел, що ґрунтується на алгоритмі методу Mersenne Twister, який уможливлює швидку генерацію псевдо-

випадкових чисел із рівномірним розподілом значень. Ці псевдовипадкові числа використовуються для індексації елементів масиву секретних даних *S*.

Спочатку масив *S* ділиться на блоки. В кожному блоці міститься по $n = 2$ біти. Нехай a – кількість блоків секретних даних. Тоді створюється масив натуральних чисел *A* розмірності *a*, який заповнюється індексами його елементів ($A[0] = 0$, $A[1] = 1$, $A[2] = 2$, ..., $A[a] = a$).

Далі виконується генерація послідовності псевдовипадкових чисел (генератор 1), й елементи масиву *A* подаються відповідно до згенерованих псевдовипадкових порядкових номерів. Для генерації першого псевдовипадкового числа використовується значення компонента K1 складеного приватного ключа (перше "зерно"). Для генерації кожного наступного числа використовується попереднє згенероване число.

У результаті отримуємо масив порядкових номерів *A*, який містить псевдовипадкову послідовність індексів блоків секретних даних, що будуть вбудовані у контейнер саме в такому порядку.

На третьому етапі методу відбувається стеганографічне вбудовування секретних даних у контейнер.

Вхідними даними на цьому етапі є масив байтів із секретними даними (масив *S* розмірності *s*), порядкових номерів (масив *A* розмірності *a*), масив байтів даних контейнера (масив *B* розмірності *b*; даними контейнера є семпли; b – це кількість семплів у аудіофайлі, що використовується як контейнер), значення компонента K2 складеного приватного ключа та масив бітів довільного файла, який є компонентом K3 складеного приватного ключа (масив *F* розмірності *f*). Формат файла-ключа (K3) не є важливим, оскільки будь-який файл у оперативній пам'яті зберігається як масив бітів.

Для подальшого ускладнення можливого стегоаналізу секретні дані вбудовуються у контейнер не послідовно, а чергуються із "шумом" (деякими довільними значеннями), що дає можливість зберігати статистичні характеристики контейнера і, таким чином, запобігає статистичному стегоаналізу. Для визначення того, що саме (блок секретних даних чи "шум") потрібно вбудовувати в поточну позицію контейнера, використовується другий генератор псевдовипадкових чисел, який грунтується на алгоритмі методу Linear Congruential Generator. Вбудовування секретних даних відбувається у циклі проходження по елементах масиву *B*. При вбудовуванні семпли правого та лівого каналів вважаються рівнозначними, тому можлива ситуація, коли блоки секретних даних будуть вбудовані в правий канал семпла, але не будуть вбудовані в лівий канал (тобто у лівий канал буде вбудований "шум") і навпаки. Блок-схема процедури перевірки при вбудовуванні показана на рис. 1.

Стеганографічне вбудовування даних у контейнер відбувається згідно з принципом різницевого перетворення бітів, який полягає у застосування деякої бінарної логічної функції (табл. 1), що уможливлює однозначне декодування секретних даних, до біта файла-ключа (*KeyBit*) та біта файла секретних даних (*SecretBit*). Номер цієї функції є компонентом K4 складеного приватного ключа.

***Таблиця* 1.** Значення істинності логічних функцій

KeyBit	*SecretBit*	F1	F2	F3	F4
0	0	0	0	1	1
0	1	1	1	0	0
1	0	0	1	1	0
1	1	1	0	0	1

Алгоритм вбудовування секретних даних складається з таких кроків:

1. Зчитується черговий блок секретних даних із масиву *S*; прохід по масиву *S* відбувається згідно з порядком слідування індексів у масиві *A*.
2. Зчитується черговий блок даних файла-ключа з масиву *F*.
3. До кожного біта блоків даних п. 1 і 2 застосовується функція-ключ (K4).
4. Перевіряється, що саме – дані чи "шум" – має вбудовуватись у черговий блок даних контейнера.
5. Якщо потрібно вбудовувати дані, то значення, отримане в п. 3, вбудовується у 2 молодших біти чергового блока даних контейнера, вказівники у масивах *A*, *B* і *S* пересуваються на наступний елемент і відбувається перехід на п. 1; інакше вбудовуються 2 випадкових значення біт, вказівник у масиві *B* пересувається на наступний елемент і відбувається перехід на п. 4.

Дії алгоритму повторюються, доки не буде вбудовано всі секретні дані. При цьому важливим фактом є те, що спотворення молодших бітів семплів аудіо-файла, який є контейнером, не впливає на суб'єктивне враження від прослуховування вмісту цього аудіофайла завдяки природній надлишковості аудіоданих.

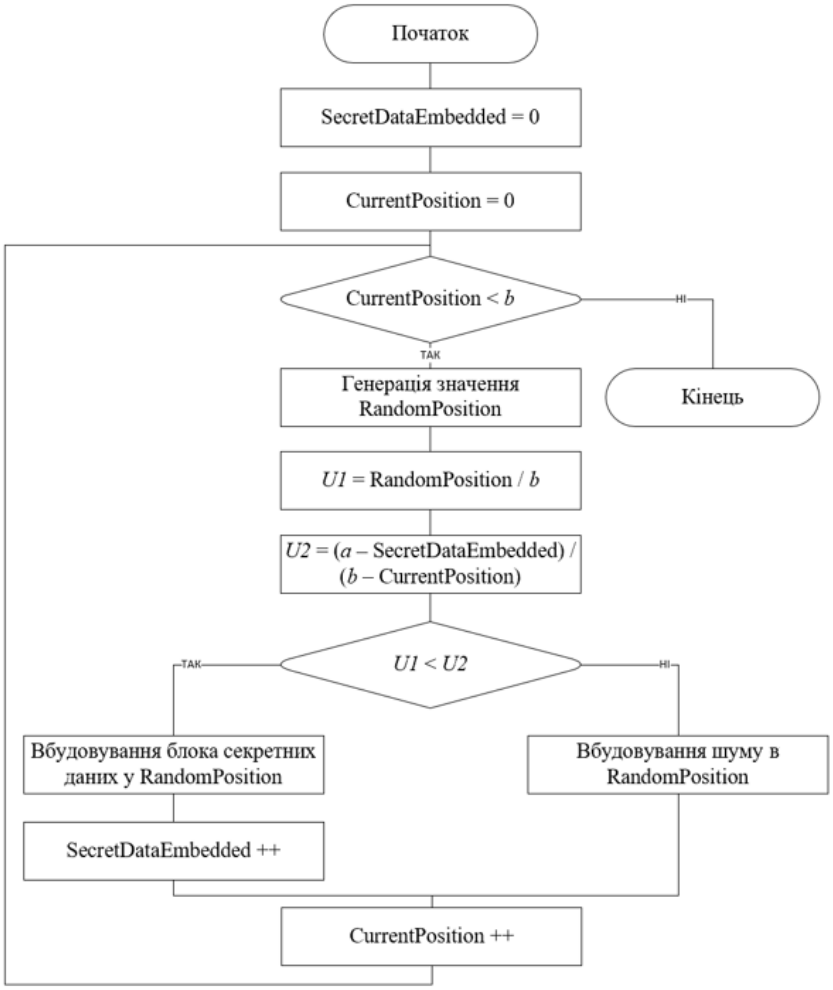

Рис. 1. Процедура перевірки при вбудовуванні

При виконанні алгоритму вбудовування враховується, що якщо файл-ключ має менший за файл секретних даних розмір, то після досягнення кінця файла-ключа вказівник на поточний елемент цього файла переміщується на початок файла, тобто можливий багатократний прохід по файлу-ключу. Крім того, слід зауважити, що адресація чергового елемента масиву секретних даних S відбувається як $S[A[i]]$, де i – індекс чергового елемента масиву A.

Схему вбудовування та відновлення секретних даних показано на рис. 2.

Відновлення секретних даних відбувається у зворотному порядку. Для відновлення секретних даних потрібні всі компоненти приватного ключа. Оскільки процедура відновлення даних використовує алгоритми, що є аналогічними алгоритмам процедури приховування даних, розглянемо лише ключові відмінності та особливості.

На першому етапі відновлення даних зчитується та пропускається блок із метаданими. На другому етапі формується масив $A1$ на основі “зерна” для першого генератора псевдовипадкових чисел і розміру секретних даних. Побудова масиву $A1$ відбувається таким же чином, як і масиву A у процедурі приховування даних. За однакового значення “зерна”, яке є компонентом K1 складеного приватного ключа, генератор псевдовипадкових чисел буде генерувати такі ж самі індекси семплів, як і в процедурі приховування даних.

На третьому етапі процедури відновлення за відомим компонентом K4 ключа (ключ-функція) та черговим бітом файла-ключа (*KeyBit*) відповідно до таблиці істинності ключ–функції відбувається визначення чергового біта файла секретних даних (*SecretBit*).

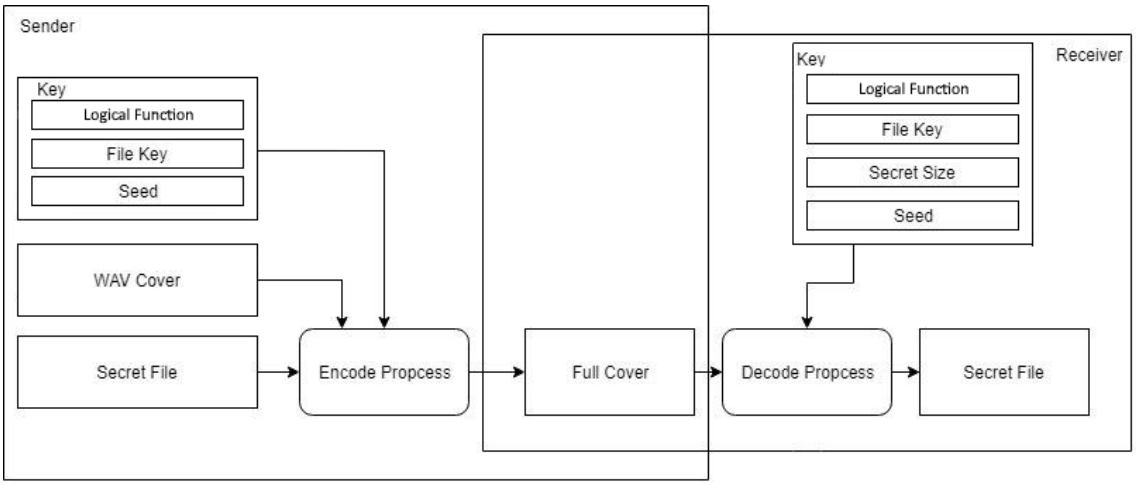

Рис. 2. Схема вбудовування та відновлення секретних даних

Після виконання всіх потрібних ітерацій (визначається компонентом K5 приватного складеного ключа) процедури відновлення секретних даних результуючий масив конвертується в масив байтів і зберігається у файл, який і буде відновленим секретним мультимедійним файлом.

Це метод можливо використовувати для вбудовування секретних даних у файл-контейнер формату FLAC. Існує можливість конвертації wav-файла у flac-файл і навпаки без втрат. Таким чином, якщо є необхідність вбудовування секретних даних у контейнер формату FLAC, то достатньо спочатку конвертувати цей flac-файл у wav-файл, далі застосувати всі етапи методу та виконати конвертацію заповненого секретними даними контейнера у форматі WAVE у формат FLAC. При відновленні даних конвертацію потрібно виконати у зворотному порядку.

Аналіз розробленого методу

Проаналізуємо стійкість і часові характеристики запропонованого методу. При аналізі стійкості будемо використовувати методику та результати, отримані в роботі [14]. Для оцінки стеганографічної стійкості методу визначимо ймовірність несанкціонованого доступу до прихованих даних у контейнері.

Нехай зловмисник знає такі факти про заповнений контейнер:

– контейнер містить секретні дані;

– загальний метод, за яким вбудовані секретні дані;

– формат секретних даних.

При цьому зловмиснику не відомий складений приватний ключ, а саме невідомими є:

– значення “зерен”, що використовуються для генерації псевдовипадкових чисел;

– файл-ключ;

– логічна функція;

– розмір секретних даних.

Тоді, виходячи з імовірнісної оцінки вгадування компонентів приватного ключа, ймовірність несанкціонованого доступу до секретних даних у запропонованому методі становить

$$P = \frac{1}{2^{64} \cdot 4 \cdot s^2 \cdot (k-m)} \approx 10^{-20} \cdot \frac{1}{s^2 \cdot (k-m)},$$

де k – розмір контейнера, m – розмір метаданих, s – розмір секретних даних.

Результати оцінювання характеристик базового методу LSB-стеганографії [1, 2], методу на основі різницевого перетворення біт [10] і запропонованого методу наведено в табл. 2. Аналіз отриманих результатів дає змогу зробити висновок про високий ступінь захисту даних і порівняно невисокий рівень складності.

Для оцінки швидкодії програмної реалізації запропонованого методу було проведено серію експериментів із заміром часу. Заміри виконувались на одному комп’ютері з процесором Intel Core i7-3610QM під операційною системою Windows 10×64.

***Таблиця* 2.** Порівняння методів

Характеристика	Базовий метод LSB-стеганографії	Метод на основі різницевого перетворення бітів	Запропонований метод
Імовірність розкриття	0,25	$\frac{1}{72 \cdot n \cdot (24 - 3 \cdot n)^3}$	$10^{-20} \cdot \frac{1}{s^2 \cdot (k - m)}$
Мінімальний розмір контейнера	$8 \cdot s$	$8 \cdot s$	$8 \cdot s$
Надлишковість	8	8	8
Складність алгоритму	Constant	Cubic	Quadratic

Було проведено серії експериментів та сформовано 9 пар "контейнер"—"секретні графічні дані". За контейнери були взяті 3 wav-файли з музичним контентом розмірами 51,8; 45,9; 26,9 Мб. У ролі секретних даних було використано 3 кольорових зображення у форматі PNG розмірами 65, 254, 1097 Кб відповідно. Вбудовування повторювалось 100 разів для кожної пари "контейнер"—"секретні графічні дані". Усереднені часові показники процесу вбудовування даних наведено в табл. 3.

***Таблиця* 3.** Усереднені часові показники процесу вбудовування даних

Контейнер	Розмір секретних графічних даних, Кб	Час вбудовування, мс	Швидкодія, Кб/с
Контейнер 1 (51,8 Мб)	65	1046	62
	254	1293	196
	1097	2106	521
Контейнер 2 (45,9 Мб)	65	919	71
	254	968	262
	1097	1906	576
Контейнер 3 (26,9 Мб)	65	559	117
	254	669	380
	1097	1269	864

Отже, з табл. 3 можна зробити висновок, що при збільшенні розміру секретних даних і зменшенні розміру контейнера зростає швидкодія процедури вбудовування (відношення розміру секретних даних до часу їх вбудовування у контейнер). Додатково поліпшити часові показники можна за рахунок паралельних обчислень.

Висновки

У цій статті наведено результати дослідження, в якому розв'язувалась задача розроблення стеганографічного методу захисту мультимедійних даних, що передбачає використання аудіоданих як контейнера для стеганографічного вбудовування секретних даних і застосування процедур оброблення даних, які ґрунтуються на алгоритмах генерації псевдовипадкових чисел.

Аналіз стійкості запропонованого методу до несанкціонованого доступу до прихованих даних показав, що ймовірність вгадування стеганографічної схеми є малою. Запропонований метод може використовуватись у випадках, коли стеганографічний захист вважається більш доцільним, ніж криптографічний.

У подальших дослідженнях доцільно виконати паралельну реалізацію запропонованих процедур і алгоритмів методу для прискорення вбудовування великих об'ємів секретних даних.

References

[1] M.S. Taha *et al.*, "Combination of steganography and cryptography: A short survey", *Inform. Technol. Commun.*, vol. 518, pp. 1–13, 2019. doi: 10.1088/1757-899X/518/5/052003

[2] M. Douglas *et al.*, "An overview of steganography techniques applied to the protection of biometric data", *Multimedia Tools and Applications*, vol. 77, no. 13, pp. 17333–17373, 2018. doi: 10.1007/s11042-017-5308-3

[3] M. Hashim *et al.*, "An extensive analysis and conduct comparative based on statistical attach of LSB substitution and LSB matching", *Int. J. Eng. Technol.*, vol. 7, no. 4, pp. 4008–4023, 2018. doi: 10.14419/ijet.v7i4.16004

[4] N. Devadiga *et al.*, "E-banking security using cryptography, steganography and data mining", *Int. J. Comp. Appl.*, vol. 164, no. 9, pp. 26–30, 2017. doi: 10.5120/ijca2017913746

[5] S. Mishra *et al.*, "Audio steganography techniques: A survey", *Adv. Comp. Comput. Sci.*, vol. 554, pp. 581–589, 2018. doi: 10.1007/978-981-10-3773-3_56

[6] M. Hussain *et al.*, "Image steganography in spatial domain: A survey", *Signal Processing: Image Communication*, vol. 65, pp. 46–66, 2018. doi: 10.1016/j.image.2018.03.012

[7] A. Kumar and K. Pooja, "Steganography – A data hiding technique", *Int. J. Comp. Appl.*, vol. 9, no. 7, pp. 19–23, 2010.

[8] M. Matsumoto and T. Nishimura, "Mersenne twister: A 623-dimensionally equidistributed uniform pseudo-random number generator", *ACM Trans. Model. Comput. Simul.*, vol. 8, no. 1, pp. 3–30, 1998. doi: 10.1145/272991.272995

[9] P. L'ecuyer, "Maximally equidistributed combined tausworthe generators", *Math. Comput.*, vol. 65, no. 213, pp. 203–213, 1996. doi: 10.1090/s0025-5718-96-00696-5

[10] Z. Hu *et al.*, "Graphical data steganographic protection method based on bits correspondence scheme", *Int. J. Intell. Syst. Appl.*, vol. 9, no 8, pp. 34–40, 2017. doi: 10.5815/ijisa.2017.08.04

[11] Ye. Sulema, "Image protection method based on binary operations", in *Proc. XXIII IEEE Int. Conf. Systems, Signals and Image Processing*, IWSSIP2016, Bratislava, Slovakia, 2016, pp. 295–298. doi: 10.1109/IWSSIP.2016.7502760

[12] *WAVE PCM Soundfile Format* [Online]. Available: http://soundfile.sapp.org/doc/WaveFormat/

[13] *Free Lossless Audio Codec* [Online]. Available: https://xiph.org/flac/

[14] Ye. Radchenko *et al.*, "Steganographic protection method based on Huffman tree", *Advances in Artificial Systems for Medicine and Education*, vol. 902, pp. 283–292, 2019. doi: 10.1007/978-3-030-12082-5_26

Е.С. Сулема, Е.А. Радченко

МЕТОД СТЕГАНОГРАФИЧЕСКОЙ ЗАЩИТЫ МУЛЬТИМЕДИЙНЫХ ДАННЫХ НА ОСНОВЕ ПРОЦЕДУРЫ ПСЕВДОСЛУЧАЙНОГО ВСТРАИВАНИЯ

Проблематика. Классическая стеганографическая защита данных основывается на сокрытии самого факта существования секретных данных. Пока этот факт не разглашен, данные остаются защищенными. Однако в подавляющем большинстве случаев такой защиты недостаточно для обеспечения безопасности данных. Поэтому обычной практикой является комбинирование шифрования данных и стеганографического встраивания. В последнее время такой подход используется во многих сферах применения, в т.ч. при защите операций электронной коммерции. Поэтому задача разработки новых, более сложных способов крипто-стеганографической защиты данных остается актуальной.

Цель исследования. Основной целью является обеспечение защиты мультимедийных данных пользователя за счет использования составного закрытого ключа для ограничения доступа к этим данным. Для достижения этой цели в исследовании ставится задача разработать стеганографический метод защиты мультимедийных данных путем их встраивания в псевдослучайные сэмплы аудиофайла.

Методика реализации. Предложенный метод является методом комбинированной крипто-стеганографической защиты данных. Он основывается на использовании составного закрытого ключа. В методе используются два генератора псевдослучайных чисел. Первый генератор используется в процедуре перемешивания блоков секретных данных. Второй генератор используется в процедуре стеганографического встраивания для выбора очередного сэмпла для встраивания секретных данных. Стеганографическое встраивание секретных данных выполняется по принципу разностного преобразования бит с использованием результата логических операций. Метод использует природные свойства аудиоданных, а именно их избыточность.

Результаты исследования. Разработан метод стеганографической защиты мультимедийных данных на основе процедуры псевдослучайного встраивания, который включает три основных этапа. На первом этапе происходит предварительная подготовка данных. На втором этапе формируется порядок расположения блоков секретных данных при их встраивании. На третьем этапе происходит стеганографическое встраивание секретных данных в контейнер. Предложены процедуры обработки данных, которые позволяют повысить уровень защиты за счет псевдослучайного перемешивания и встраивания данных.

Выводы. Разработанный стеганографический метод защиты мультимедийных данных предусматривает использование аудиоданных как контейнера для стеганографического встраивания секретных данных и применения процедур обработки данных, основанных на алгоритмах генерации псевдослучайных чисел. Анализ устойчивости предложенного метода к несанкционированному доступу к скрытым данным показал, что вероятность угадывания стеганографической схемы является малой. Предложенный метод может использоваться в случаях, когда стеганографическая защита считается более целесообразной, чем криптографическая.

Ключевые слова: защита данных; стеганография; псевдослучайные последовательности; логические операции.

Ye.S. Sulema, Ye.O. Radchenko

MULTIMEDIA DATA STEGANOGRAPHIC PROTECTION METHOD BASED ON PSEUDORANDOM EMBEDDING PROCEDURE

Background. Classical steganographic data protection is based on hiding the fact of secret data existence. Until this fact is disclosed, the data remains protected. However, in most of the cases, such protection is insufficient for ensuring data security. Therefore, it is common practice to combine data encryption and steganographic embedding. Recently, this approach has been used in many fields of application, including the protection of e-commerce transactions. Thus, the task of developing new, more complex methods of crypto-steganographic data protection remains topical.

Objective. The main goal is to protect the user's multimedia data by using a composite private key to limit access to this data. To achieve this goal, the research objective is to develop a steganographic method for protecting multimedia data by embedding it in pseudo-random samples of an audio file.

Methods. The proposed method is a method of combined crypto-steganographic data protection. It is based on the use of a composite private key. The method uses two pseudo-random number generators. The first generator is used in the procedure of secret data block mixing. The second generator is used in the steganographic embedding procedure to select the next sample for embedding secret data. The steganographic embedding of secret data is carried out according to the principle of bits correspondence using the result of logical operations. The method uses the natural properties of audio data, namely, their redundancy.

Results. A method for steganographic protection of multimedia data based on the pseudo-random embedding procedure has been developed. It includes three main stages. At the first stage, preliminary data preparation is fulfilled. At the second stage, the secret data block arrangement order is formed to be used while embedding. At the third stage, the steganographic embedding of secret data into the container is carried out. The proposed data processing procedures enable increasing the level of protection due to pseudo-random mixing and embedding the data.

Conclusions. The developed steganographic method for protecting multimedia data uses audio data as a container for secret data steganographic embedding. It includes data processing procedures based on pseudo-random number generation algorithms. An analysis of the proposed method resistance to unauthorized access to hidden data has shown that the probability of the steganographic scheme guessing is small. The proposed method can be used in cases where steganographic protection is considered as more appropriate than cryptographic protection.

Keywords: data protection; steganography; pseudo-random sequences; logical operations.

Рекомендована Радою
факультету прикладної математики
КПІ ім. Ігоря Сікорського

Надійшла до редакції
14 січня 2020 року

Прийнята до публікації
04 лютого 2020 року