«Системні технології» 6 (131) 2020 «System technologies»
DOI 10.34185/1562-9945-6-131-2020-05
УДК 004.492.2

Є.О. Радченко, Є.С. Сулема

МАСШТАБОВАНА ПРОГРАМНА СИСТЕМА СТЕГАНОГРАФІЧНОГО ЗАХИСТУ МУЛЬТИМЕДІЙНИХ ДАНИХ КОРИСТУВАЧІВ ІНТЕРНЕТ

Анотація. У статті запропоновано універсальну об'єктно-орієнтовану архітектуру програмної системи стеганографічного захисту мультимедійних даних, що ґрунтується на використані поліморфізму для мінімізації роботи програміста при необхідності масштабування та підтримки стеганографічної системи.
Ключові слова: архітектура програмної системи, захист мультимедійних даних, стеганографія.

Постановка проблеми. Зростання кількості задач, пов'язаних з реєстрацією, обробкою, передачею та збереженням мультимедійних даних, вимагає забезпечення захисту даних при їх передаванні через мережу Інтернет, віддаленій обробці та збережені у хмарних сховищах. Крім того, останнім часом спостерігається тенденція до збільшення популярності хмарних сервісів, які надають стрімко зростаючому колу користувачів Інтернет можливість зберігати віддалено великі об'єми особистих даних. Таким чином, постає науково-практична задача створення нових, більш надійних та швидких стеганографічних методів захисту мультимедійних даних. При цьому важливою задачею є ефективна реалізація методів захисту мультимедійних даних у вигляді програмного продукту із забезпеченням ефективної обробки мультимедійних даних.

Аналіз останніх досліджень і публікацій. Аналіз існуючих стеганографічних методів [1] не дозволяє виділити один метод стеганографічного захисту даних як найкращий, оскільки кожен підхід, що пропонується, має свої переваги та недоліки. Вибір того чи іншого методу для програмної реалізації залежить від вимог до програмного забезпечення, що розробляється, зокрема, галузі використання програмного продукту, виду конфіденційних даних, характеру загроз їх цілісності, вимог до швидкодії програмного забезпечення тощо. Так, деякі методи [2] доцільно використовувати,

коли часом вбудовування та декодування стегоданих можна знехтувати, проте важливою є стійкість до стегоатак. Водночас, в деяких випадках важливим є швидкість обробки даних, а ймовірність стегоатак є мінімальною. Також, у певних випадках, велике значення має наявність ключа та його формат. Таким чином, постає проблема об'єднання окремих стеганографічних методів в одній програмній системі, що може легко масштабуватись, підтримуватись та доповнюватись новими стеганографічними методами захисту мультимедійних даних.

У відкритому доступі є досить багато програмних реалізацій засобів стеганографічного захисту даних [3]. Найбільш популярними є S-Tools, EzStego, OutGuess, JSteg, Contraband, Steganos, Hide4PGP [4]. Проаналізувавши ці та інші програмні засоби стеганографічного захисту персональних мультимедійних даних користувачів, можна сформувати основні вимоги до програмних систем на основі стеганографічного захисту:

- підтримка роботи з графічними та аудіо-контейнерами [5];
- шифрування стегоданих перед їх вбудовуванням [6];
- використання особливостей мультимедійних даних;
- стійкість до статистичних атак;
- можливість масштабування та додавання нових стеганографічних методів.

Серед усіх розглянутих програмних засобів жоден не відповідає цим вимогам у повній мірі, тому задача створення вдосконалених стеганографічних систем є актуальною.

Мета досліджень. Метою статті є розроблення універсальної архітектури програмної системи захисту конфіденційних даних користувачів, що дозволяє мінімізувати час роботи програміста у процесі підтримки та масштабування системи.

Викладення основного матеріалу досліджень. Компоненти архітектури програмної системи, що пропонується, можна розділити на чотири групи (рис. 1).

Перша група компонентів реалізує інтерфейс користувача (UI). Передбачається, що графічний інтерфейс користувача розроблений засобами QML [7], проте існує можливість створення нестандартного UI;

Друга група компонентів реалізує модуль логіки (Logical Unit), що являє собою набір класів для обміну даними між користувачем та методами вбудовування / декодування.

ISSN 1562-9945 (Print)
ISSN 2707-7977 (Online)

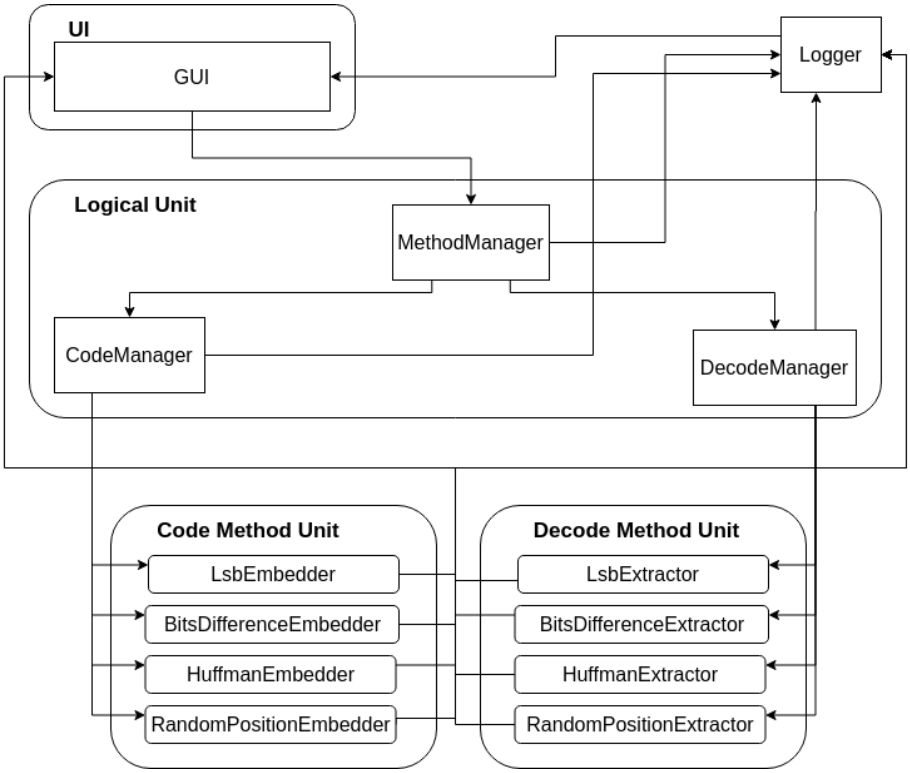

Рисунок 1 - Узагальнена архітектура програмного забезпечення стеганографічного захисту мультимедійних даних

Третя група компонентів реалізує модуль кодування (Code Method Unit). Цей модуль реалізує процедури вбудовування стегоданих.

Четверта група компонентів реалізує модуль декодування (Decode Method Unit). Цей модуль реалізує процедури відновлення даних.

Таким чином, основними модулями програмної системи, які забезпечують безпосередню роботу з даними, є модуль кодування та модуль декодування. Ці модулі реалізують певний набір методів стеганографічного захисту мультимедійних даних. Пропонується використовувати наступні методи як базовий набір методів захисту:

- LSB-метод [8];
- метод на основі схеми відповідності бітів [9];
- метод на основі дерева Гаффмана [2];
- метод на основі процедури псевдовипадкового вбудовування [10].

LSB-метод є базовим методом, який дозволяє досягти максимальної швидкодії обробки даних. Його доцільно застосовувати, якщо ймовірність стегоатаки є мінімальною. Метод на основі схеми відповідності бітів легко об’єднується з більшістю існуючих модифікацій LSB-методів для підвищення їх стеганографічної стійкості використовуючи логічні операції. Метод на основі

дерева Гаффмана дозволяє досягти високої стеганографічної стійкості при вбудовуванні графічних даних та є стійким до певних видів стегоатак. Метод на основі процедури псевдовипадкового вбудовування доцільно використовувати у випадку вбудовування конфіденційних даних у WAV-контейнер та коли існує ймовірність, що у стегоаналітика може бути оригінальний контейнер для порівняння, оскільки даний метод додатково має можливість вбудовувати шум.

Об'єктно-орієнтований дизайн системи, на основі запропонованої архітектури, ґрунтується на використанні поліморфізму. У результаті отримуємо набір споріднених класів із загальним інтерфейсом і різними реалізаціями стеганографічних методів. Пропонується використовувати шаблон проєктування "Стратегія" [11].

Для перевірки архітектури у дослідженні було розроблено програмну систему на мові програмування C++ з використанням бібліотеки Qt та QML для розроблення GUI. QML є декларативною мовою програмування для розроблення інтерфейсу користувача, що заснована на JavaScript.

Усі реалізовані методи успадковуються від базового класу `BaseMethod` та зберігають шлях та ім'я заповненого контейнеру та ключа як параметри. Клас `BaseMethod` має один віртуальний метод `virtual void initialize() = 0` для ініціалізації параметрів методу. Від класу BaseMethod успадковуються класи `BaseCodeMethod` та `BaseDecodeMethod`. Клас `BaseCodeMethod` має два допоміжні поля: шлях та ім'я порожнього контейнера та стегоданих. Також даний клас має віртуальний метод `virtual bool code() = 0`, що перевизначається у дочірніх класах. Клас `BaseDecodeMethod` має поля, що визначають шлях та ім'я потенціального декодованого файлу. Як і `BaseCodeMethod`, цей клас має один віртуальний метод `virtual bool decode() = 0`.

Загальну схему успадкування класів у розробленому програмному забезпеченні представлено на рис. 2.

ISSN 1562-9945 (Print)
ISSN 2707-7977 (Online)

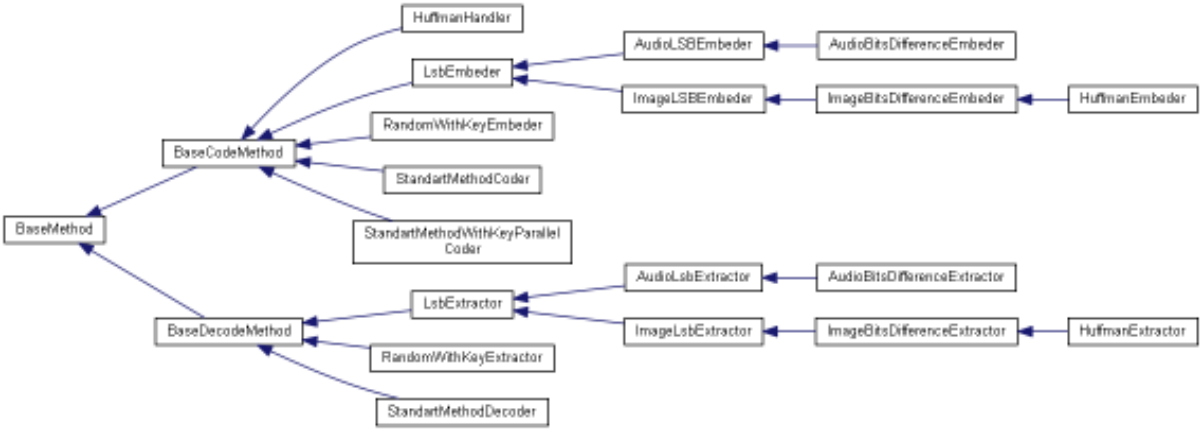

Рисунок 2 - Схема успадкованості класів

Кожен метод реалізований у двох класах: для вбудовування (`Embedder`) та для декодування (`Extractor`). На кожен тип контейнера (зображення або аудіодані), що підтримує метод, створено окремий клас. Наприклад, вбудовування конфіденційних мультимедійних даних класичним LSB-методом представлено абстрактним класом `LsbEmbeder`, від якого успадковані класи `AudioLsbEmbeder` та `ImageLsbEmbeder`, що використовуються залежно від типу контейнера. Для декодування даних реалізований абстрактний клас `LsbExtractor`, від якого успадковані `AudioLsbExtractor` та `ImageLsbExtractor` відповідно.

За вбудовування стегодних відповідає класс `CodeManager`, а за їх зворотне декодування – `DecodeManager`. Розроблені стеганографічні методи у програмному коді представленні наступнми класами:

1. LSB-метод – `LsbEmbeder` та `LsbExtractor` для вбудовування та декодування конфіденційних даних відповідно з префіксом `Audio` або `Image` залежно від типу контейнера.

2. Метод на основі схеми відповідності бітів – `BitsDifferenceEmbeder` та `BitsDifferenceExtractor` для вбудовування та декодування конфіденційних даних відповідно з префіксом `Audio` або `Image` залежно від типу контейнера.

3. Метод на основі дерева Гаффмана – `HuffmanEmbeder` та `HuffmanExtractor` для вбудовування та декодування конфіденційних даних відповідно у графічні контейнери.

4. Метод на основі процедури псевдовипадкового вбудовування – `RandomWithKeyEmbeder` та `RandomWithKeyExtractor` для вбудовування та декодування конфіденційних даних відповідно у аудіоконтейнери.

Таким чином, систему досить легко підтримувати, оскільки всі методи перенесено в окрему ієрархію класів.

Також, даний підхід дозволяє зміну одного методу на інший у процесі виконання програми шляхом заміни вказівників на об'єкти, що описують необхідний метод у `CodeManager` та `DecodeManager`. Разом з тим, дана архітектура дозволяє досить просто додати нові стеганографічні методи як для вбудовування стегоданих, так і для їх зворотного декодування.

Оскільки кожен стеганографічний метод має свій власний формат ключа, було реалізовано абстрактний клас `BaseKey` (лістинг 1). Після додавання нового методу до системи, потрібно створити новий клас, що буде реалізовувати ключ для даного методу, та успадкувати його від `BaseKey`. Після цього необхідно перевизначити метод `toBytes()`, де кожен ключ конвертується у масив байтів та послідовно записується до результуючого масиву байтів, що повертає метод. Це зроблено для уніфікації інтерфейсу.

Лістинг 1. Абстрактний клас BaseKey.

```
class BaseKey
{
public:
   BaseKey() = default;
   virtual ~BaseKey() = default;
   virtual std::vector<uint8_t> toBytes() const = 0;
};
```

Окрім програмного додавання нових стеганографічних методів, існує необхідність використання цих методів у різних програмних середовищах.

Клас `DataProvider` є абстрактним класом для обміну даними між користувачем та логічним модулем, що відповідає за вбудовування та декодування стегоданих. Основною задачею цього класу є робота з MethodManager для передачі даних від користувача у методи. У поточній реалізації клас `QmlDataProvider` успадкований від `DataProvider`, оскільки програмна система має GUI, що реалізований засобами QML. Отже, архітектура побудована таким чином, щоб мінімізувати роботу програміста у випадку, якщо програмний продукт треба розширити для постачання даних не через вікно програми, а наприклад, через деякий сервіс. Для цього потрібно створити новий клас успадкований від `DataProvider` та перевизначити необхідні методи.

Для логування повідомлень програмного забезпечення реалізовано класс `Logger`. Цей клас є одиначкою (Singleton), тож усі компоненти програмної сис-

ISSN 1562-9945 (Print)
ISSN 2707-7977 (Online)

теми передають важливі повідомлення користувачеві саме за допомогою об’єкту цього класу. Це дає змогу контролювати доступ до єдиного екземпляру класу.

Оскільки у програмному застосунку використовується велика кількість конвертувань даних, для даних цілей було реалізовано шаблон проєктування “адаптер”. Він дає змогу організувати використання методів об’єкта, що недоступні для модифікації, шляхом створення спеціального інтерфейсу. Прикладом використання може бути конвертування зображення у масив бітів для подальшого вбудовування у контейнер.

Оскільки інтерфейс користувача для вводу вхідних даних у поточній архітектурі легко змінюється, виникає необхідність забезпечити уніфікований інтерфейс до підсистеми з реалізованими стеганографічними методами (Code Method Unit та Decode Method Unit). Разом з тим, необхідно приховати складність системи шляхом зведення усіх можливих викликів до одного об'єкту, що буде делегувати ці виклики відповідним об’єктам системи. Дану задачу пропонується вирішувати за допомогою шаблону проєктування “фасад”. Так, у розробленій системі було створено клас `MethodManager`, об’єкт якого є фасадним об’єктом, через який UI викликає відповідні методи для кодування та декодування стегоданих. Даний об’єкт виступає у ролі API для стеганографічних методів.

Таким чином, при розробленні програмного забезпечення рекомендується використовувати наступні шаблони проєктування: “Singleton”, “Стратегія”, “Адаптер” та “Фасад”. Використання цих шаблонів проєктування дозволяє полегшити розробку, масштабованість та оптимізувати програмну систему.

Висновки. Запропоновано нову універсальну архітектуру програмної системи LSB-стеганографії, що дозволяє легко масштабувати та підтримувати систему, оскільки всі реалізації стеганографічних методів знаходяться у окремій ієрархії класів. Даний факт надає можливість досить просто додавати нові процедури як для вбудовування стегоданих, так і для відновлення даних користувача. Разом з тим, ця архітектура дозволяє зміну одного методу на інший у процесі виконання програми шляхом заміни вказівників на необхідні об'єкти, що описують певний стеганографічний метод у `CodeManager` та `DecodeManager`.

Уніфіковано інтерфейс до підсистеми з реалізованими стеганографічними методами, що дозволяє адаптувати програмну систему до постачання вхідних

даних для роботи стеганографічних методів не лише через головне вікно програмного продукту, а і через інші засоби, наприклад Інтернет-сервіси. Уніфіковано формат ключа для можливості використання у якості ключа різних структур даних (зображення, текст, файл, число тощо).

Запропонований підхід дозволяє спростити розроблення, масштабування та підтримку програмного забезпечення стеганографічного захисту мультимедійних даних користувачів Інтернет.

ЛІТЕРАТУРА / ЛИТЕРАТУРА

1. Широчин С.С. Методи комбінованого стеганографічного захисту мультимедійних даних в хмарних сховищах : дис. канд. техн. наук : 05.13.05 / Широчин С. С. – 2015.

2. Steganographic Protection Method Based on Huffman Tree / [Y. Radchenko, I. Dychka, Y. Sulema та ін.]. // Springer. – 2019. – № 902. – P. 283–292.

3. Johnson N.F. Steganalysis of Images Created Using Current Steganography Software / N.F. Johnson, S. Jajodia. // Information Hiding, LNCS 1525, Springer-Verlag Berlin Heidelberg. – 1998. – P. 32–47.

4. Липка Т.Б. Модифікація методу стеганографії з використанням матриці судоку / Лиипка Т. Б. // ст., фак-т приклад. матем., НТУУ “КПІ ім. Ігоря Сікорського”. – 2018.

5. E-Banking Security using Cryptography, Steganography and Data Mining / [N.Devadiga, H. Kothari, H. Jain, S. Sankheта ін.] // International Journal of Computer Applications. – 2017. – №164. – P. 26–30.

6. Combination of Steganography and Cryptography: A short Survey / [M. Taha, M. Rahim, S. Lafta та ін.]. // Information Technology and Communication. – 2019. – №518. – P. 1–13.

7. Боровский А.Н. Qt 4.7+ Практическое программирование / Боровский А. Н. // Санкт-Петербург, Россия: БХВ-Петербург. – 2012.

8. Roy S. Audio Steganography Using LSB Encoding Technique with Increased Capacity and Bit Error Rate Optimization / [Roy S. та ін.] // Proceedings of the Second International Conference on Computational Science, Engineering and Information Technology. – 2017. – P. 372-376.

9. Hu Z. Graphical Data Steganographic Protection Method Based on Bits Correspondence Scheme / Z. Hu, I. Dychka, Y. Sulema, Y. Radchenko. // China : Hong Kong MECS Press.. – 2017. – P. 34–40.

ISSN 1562-9945 (Print)
ISSN 2707-7977 (Online)

10. Сулема Є. С. Метод стеганографічного захисту мультимедійних даних на основі процедури псевдовипадкового вбудовування / Є. С. Сулема, Є. О. Радченко. // Наукові вісті КПІ ім. Ігоря Сікорського. – 2020. – №1. – С. 40–47.

11. Pikus F. G. Hands-On Design Patterns with C++: Solve common C++ problems with modern design patterns and build robust applications / Pikus F.G. // Birmingham, UK: Packt Publishing Ltd. – 2019. – P. 328.

REFERNCES

1. Shyrochyn S.S. Methods of combined steganographic protection of multimedia data in cloud storage : dis. PhD : 05.13.05 / Shyrochyn S. S. – 2015.

2. Steganographic Protection Method Based on Huffman Tree / [Y. Radchenko, I. Dychka, Y. Sulema та ін.]. // Springer. – 2019. – № 902. – P. 283–292.

3. Johnson N.F. Steganalysis of Images Created Using Current Steganography Software / N.F. Johnson, S. Jajodia. // Information Hiding, LNCS 1525, Springer-Verlag Berlin Heidelberg. – 1998. – P. 32–47.

4. Lypka T. B. Modification of the steganography method using the Sudoku matrix / Lypka T. B. // st., Faculty of Applied Mathematics, National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute". – 2018.

5. E-Banking Security using Cryptography, Steganography and Data Mining / [N.Devadiga, H. Kothari, H. Jain, S. Sankheта ін.] // International Journal of Computer Applications. – 2017. – №164. – P. 26–30.

6. Combination of Steganography and Cryptography: A short Survey / [M. Taha, M. Rahim, S. Lafta та ін.]. // Information Technology and Communication. – 2019. – №518. – P. 1–13.

7. Borovsky A. N. Qt 4.7+ Practical programming / Borovsky A. N. // St. Petersburg, Russia: BHV-Petersburg. – 2012.

8. Roy S. Audio Steganography Using LSB Encoding Technique with Increased Capacity and Bit Error Rate Optimization / [Roy S. та ін.] // Proceedings of the Second International Conference on Computational Science, Engineering and Information Technology. – 2017. – P. 372-376.

9. Hu Z. Graphical Data Steganographic Protection Method Based on Bits Correspondence Scheme / Z. Hu, I. Dychka, Y. Sulema, Y. Radchenko. // China : Hong Kong MECS Press.. – 2017. – P. 34–40.

10. Sulema Y. S. Method of steganographic protection of multimedia data based on the procedure of pseudo-random embedding / Sulema Y. S., Radchenko Y. O. // KPI Science News. – 2020. – №1. – C. 40–47.

11. Pikus F. G. Hands-On Design Patterns with C++: Solve common C++ problems with modern design patterns and build robust applications / Pikus F. G. // Birmingham, UK: Packt Publishing Ltd. – 2019. – P. 328.

Received 11.11.2020.
Accepted 13.11.2020.

Масштабируемая программная система стеганографической защиты мультимедийных данных пользователей интернет

Предложено универсальную объектно-ориентированную архитектуру программной системы стеганографической защиты мультимедийных данных, основанной на использовании полиморфизма для минимизации работы программиста при необходимости масштабирования и поддержки стеганографической системы. Все программные реализации стеганографических методов защиты мультимедийных данных имеют одинаковый интерфейс для использования, что дает возможность быстрого добавления новых стеганографических методов. Унифицирован интерфейс для работы со стеганографическими методами в программной системе через единый объект, что позволяет адаптировать программное приложение к поставке входных данных не только через главное окно программного продукта, а и через Интернет-сервисы, мобильные приложения и другие способы поставки входных данных. Унифицирован формат ключа для возможности использования ключей различных структур и типов данных.

Scalable software system for internet user's multimedia data steganographic protection

The universal object-oriented architecture for the programmatic system was designed for steganographic protection of multimedia data. The architecture is based on the usage of polymorphism in order to minimize the amount of work needed to scale and support the steganographic system. All software implementations of steganographic methods of multimedia data protection have the same interface for use, which allows you to quickly add new steganographic methods for embedding stegodata and for restoring confidential user data. However, the proposed universal architecture allows the replacement of one steganographic method to another while using the software product by replacing the pointers to the necessary objects of implementation of steganographic methods of multimedia data protection. Unified interface for working with implemented steganographic methods in the software system through a single object in the software product, which allows you to adapt the software application to supply input necessary for the correct operation of steganographic methods of multimedia data protection, not only through the main software window, but also through Internet services, mobile applications and other ways of providing input data for the usage of steganographic methods in other software environments. The key format (array of bytes) was unified for the possibility of using keys of different structures and types of data (number, text, file, image, etc.) in the universal architecture of the software system of steganographic protection of multimedia data of users. The usage of structural, concurrency and behavioral design templates in a universal architecture can minimize the development time of the programmer, facilitate the maintenance, scalability and optimization of the software system.

ISSN 1562-9945 (Print)
ISSN 2707-7977 (Online)

Радченко Євген Олександрович – аспірант кафедри програмного забезпечення комп'ютерних систем, Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського», Київ.

Сулема Євгенія Станіславівна – канд. техн. наук, доцент, доцент кафедри програмного забезпечення комп'ютерних систем. Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського», Київ.

Радченко Евгений Александрович – аспирант кафедры программного обеспечения компьютерных систем, Национальный технический университет Украины «Киевский политехнический институт имени Игоря Сикорского», Киев.

Сулема Евгения Станиславовна – канд. техн. наук, доцент, доцент кафедры программного обеспечения компьютерных систем. Национальный технический университет Украины «Киевский политехнический институт имени Игоря Сикорского», Киев.

Yevhen Radchenko – Post-Graduate Student of Computer Systems Software Department, National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Kyiv.

Yevgeniya Sulema – Cand. Sc. (Eng.), Associate Professor, Associate Professor of Computer Systems Software Department. National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Kyiv.