НАЦІОНАЛЬНА АКАДЕМІЯ НАУК УКРАЇНИ
ІНСТИТУТ ТЕЛЕКОМУНІКАЦІЙ І ГЛОБАЛЬНОГО ІНФОРМАЦІЙНОГО ПРОСТОРУ

Кваліфікаційна наукова
праця на правах рукопису

СКЛАДАННИЙ ПАВЛО МИКОЛАЙОВИЧ

УДК 004.056.5:681.51

ДИСЕРТАЦІЯ

МОДЕЛІ І МЕТОДИ ЗАБЕЗПЕЧЕННЯ ІМІТОСТІЙКОСТІ ТА КОНФІДЕНЦІЙНОСТІ В СИСТЕМАХ ОБРОБКИ ІНФОРМАЦІЇ

Спеціальність 05.13.06 – Інформаційні технології
Технічні науки

Подається на здобуття наукового ступеня кандидата технічних наук. Дисертація містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело

_____(П.М.Складанний)

(підпис, ініціали та прізвище здобувача)

Науковий керівник
Гулак Геннадій Миколайович
кандидат технічних наук, доцент

Київ – 2021

АНОТАЦІЯ

Складанний П.М. МОДЕЛІ І МЕТОДИ ЗАБЕЗПЕЧЕННЯ ІМІТОСТІЙКОСТІ ТА КОНФІДЕНЦІЙНОСТІ В СИСТЕМАХ ОБРОБКИ ІНФОРМАЦІЇ . – Кваліфікаційна наукова праця на правах рукопису.

Дисертація на здобуття наукового ступеня кандидата технічних наук за спеціальністю 05.13.06 – Інформаційні технології. – Інститут телекомунікацій і глобального інформаційного простору Національної академії наук України, Київ, 2021.

Дисертація присвячена розв`язанню актуального наукового завдання, що полягає у розробленні теоретичних і прикладних засад побудови та забезпечення методами криптографічної обробки інформації імітостійкості та конфіденційності даних в системах обробки інформації з урахуванням множини кіберзагроз та потенційно можливих наслідків їх реалізації.

Об'єктом дослідження в роботі є процеси створення та використання нових моделей та методів забезпечення імітостійкості та конфіденційності криптографічного захисту інформації в системах обробки інформації. *Предметом дослідження* – моделі та методи для забезпечення імітостійкості та конфіденційності даних в системах обробки інформації в умовах зростання потужності кібератак та ймовірності цільового ураження систем. *Методи дослідження*. При вирішенні поставлених задач в дисертаційній роботі було використано методи теорії ймовірностей та математичної статистики, математичного моделювання, синтезу та аналізу криптосистем. *Метою дисертаційної роботи* є підвищення рівня імітостійкості та конфіденційності інформації в системах обробки інформації в умовах кібератак, за рахунок розробки й впровадження адекватних умовам застосування методів і моделей забезпечення надійного криптозахисту таких систем. Для досягнення поставленої мети в роботі:

1) досліджено множину актуальних загроз і уразливостей інформації, що циркулює в системах обробки інформації;

2) побудувано уточнені моделі порушника та загроз, а також автоматну модель безпеки функціонування каналів управління системах обробки інформації в умовах впливу кібератак;

3) розроблено метод генерації потоку підстановок для шифру багатоалфавітної заміни для забезпечення в системах обробки інформації конфіденційності та цілісності інформації;

4) розроблено метод виявлення атак на програмні реалізації засобів криптографічного захисту інформації в системах обробки інформації;

5) розроблено модель функціонування (криптосхему) шифратора багатоалфавітної заміни (модулю криптографічного захисту інформації) в системах обробки інформації;

6) вдосконалено метод оцінки ефективності застосування сучасних криптосистем.

Метод виявлення атак на програмні реалізації засобів криптографічного захисту інформації в системах обробки інформації розроблено на підставі вимог нормативних документів системи криптографічного захисту інформації щодо необхідності забезпечення контролю цілісності програмного коду, виявлення збоїв засобів захисту та помилок операторів. Метод ґрунтується на уточнених моделях порушника і загроз, а також автоматній моделі безпеки функціонування каналів управління СОІ в умовах впливу кібератак та полягає у дослідженні статистики аномальної поведінки (стану) засобу криптографічного захисту інформації та зводиться до виявлення зміни математичного сподівання в деякій новій випадковій послідовності, сформованій з вихідної. При розробці методу було зроблено припущення, що у випадку штатного функціонування системи обробки інформації деякий потік вимірюваних даних від об'єкту контролю є стаціонарним, а після настання деякої події він змінює розподіл своїх значень. Необхідно максимально точно виявити момент настання цій ситуації та прийняти рішення щодо аномальності ситуації. Як результат, застосування методу виявлення атак на програмні реалізації засобів криптографічного захисту інформації в состемі обробки інформації та дозволяє виявити момент настання певної критичної ситуації та прийняти рішення щодо її аномальності, відновити секретний ключ, за допомогою якого створені зашифровані повідомлення та/або розкрити вихідне значення зашифрованої інформації в СОІ, а також знизити вартість системи виявлення атак на програмні реалізації засобів КЗІ приблизно на 25%.

Метод генерації потоку підстановок для шифру БАЗ з метою забезпечення імітостійкого шифрування *в* СОІ в свою чергу ґрунтується на вирішенні задачі швидкого формування підстановок з симетричної групи підстановок S_n. Суть методу полягає у формуванні нижнього рядка підстановки заміни в безповторному наборі підстановок із випадкової (або псевдовипадкової) рівномірно розподіленої послідовності (РРВП). Для цього використовується потік випадкових чисел $j_0, j_1, \dots, j_m, \dots \in \mathbb{Z}_n$ таким чином, що символ, який присутній у сформованій частині рядка, відхиляється та у подальшому не використовується. Таким чином процес може продовжуватися досить

довго. Як результат, застосування методу генерації потоку підстановок з використанням шифру БАЗ для забезпечення імітостійкого шифрування в СОІ (шифру багатоалфавітної заміни на основі оригінального швидкісного алгоритму формування підстановок замін) дозволить підвищити надійність захисту інформації в СОІ, а також знизити ймовірність підробки команди управління до прийнятної для практичного застосування величини порядку 10^{-6}.

Модель функціонування (криптосхему) шифратора БАЗ (модулю КЗІ). Модель включає генератор ПВП, алгоритм генерації підстановок заміни, систему контролю і блокування та вузол шифрування. Базою для моделі стали модель та метод виявлення атак на програмні реалізації засобів КЗІ в СОІ та методу генерації потоку підстановок з використанням шифру БАЗ. Її квінтесенція полягає в тому, що вона передбачає:

по-перше, при ініціалізації ключа – генерацію ПВП у відповідному блоці;

по-друге, перевірку правильності роботи генератора ПВП у блоці «Вузол контролю та блокування»;

по-третє, подачу ПВП сигналу блокування при збої генератора ПВП та його зупинку до відновлення правильної роботи;

по-четверте, при позитивному проходженні перевірки ПВП: – початок роботи алгоритму генерації підстановок заміни з подальшим шифруванням даних; – перевірку в блоці «Вузол контролю та блокування».

Впровадження зазначеної моделі дозволяє, на відміну від існуючих, забезпечити конфіденційність і цілісність інформації, що циркулює в СОІ та в умовах кібератак забезпечити власне функціональну безпеку і живучість самої системи.

Останнім кроком дисертаційних досліджень стало удосконалення методу оцінки ефективності застосування криптосистем в СОІ.

Відомо, що у криптографії в якості оцінки ефективності роботи криптосистем, за звичай, використовують співвідношення p/R, де R – мінімальна складність методу криптоаналізу, що обчислюється в елементарних операціях обчислювальної техніки, p – ймовірність успішної реалізації цього методу. У роботі ця оцінка набула подальшого розвитку у плані практичного застосування для оцінки відносної ефективності системи захисту інформації в СОІ в умовах кібератак. Зважаючи на матеріальний характер збитків власника СОІ у разі успіху несанкціонованого втручання у її роботу внаслідок успішної кібератаки та виходячи з необхідності здійснення зловмисником певних матеріальних витрат для

реалізації цієї атаки, є логічним у якості оцінки рівня відносної ефективності системи захисту (засобу КЗІ) – величини Q – використовувати дещо інше співвідношення

У процесі виконання дисертаційної роботи отримано такі основні результати.

1. Розроблено метод генерації потоку підстановок з використанням шифру БАЗ для забезпечення імітостійкого шифрування в СОІ, впровадження якого дозволяє обрати таку степень підстановок, яка б забезпечувала достатню швидкодію криптоперетворення та була б раціональною для забезпечення захисту повідомлень від підробки.

2. Розроблено метод виявлення атак на програмні реалізації засобів криптографічного захисту інформації в СОІ, впровадження якого дозволяє своєчасно виявити момент настання певної критичної ситуації та прийняти рішення щодо подальших дій.

3. Розроблено модель функціонування (криптосхема) шифратора БАЗ (модулю криптографічного захисту інформації) в СОІ, яка є комплексним рішенням в межах двох попередніх методів та впровадження якої дозволяє забезпечити конфіденційність і цілісність інформації, що циркулює в СОІ та в умовах кібератак забезпечити власне функціональну безпеку та живучість самої системи.

4. Удосконалено метод оцінки ефективності застосування криптосистем, який шляхом урахування матеріального характеру збитків власника СОІ у разі успіху несанкціонованого втручання у її роботу, дозволяє визначити границі для відносної ефективності системи захисту інформації в СОІ в умовах реалізації кіберзагроз.

5. Проведено, шляхом імітаційного моделювання процесу нападу на програмну реалізацію засобу КЗІ, експериментальне дослідження моделі функціонування шифратора БАЗ. Це, по-перше, підтвердило дієздатність крипто схеми та, по-друге, в масштабі часу наближеному до реального дозволило впевнитись, що застосування моделі дозволить зменшити час на виявлення атаки приблизно на 20%.

Ключові слова: система обробки інформації, конфіденційність, ціналісність, автоматизована система, багатоалфавітна заміна, імітостійкість, криптосхема, криптоалгоритм, криптосистема, криптографічний захист.

Skladannyi P.M. MODELS AND METHODS OF ENSURING IMITATION RESISTANCE AND CONFIDENTIALITY IN INFORMATION PROCESSING SYSTEMS. – Manuscript.

The dissertation for the degree of a candidate of technical sciences on the specialty 05.13.06 «Information technologies.» Institute of Telecommunications and Global Information Space of the National Academy of Sciences of Ukraine, Kyiv, 2021.

The dissertation is devoted to the decision of the actual scientific problem in development of theoretical and applied bases of construction and maintenance by methods of cryptographic processing of the information of imitation stability and confidentiality of the data in SOI taking into account set of cyberthreats and potential consequences of their realization.

The object of research in this work are the processes of creating and using new models and methods to ensure imitation and confidentiality of cryptographic protection of information in information processing systems. The subject of research - models and methods to ensure imitation and confidentiality of data in information processing systems in terms of increasing the power of cyberattacks and the probability of target damage to systems. Research methods. In solving the problems in the dissertation the methods of probability theory and mathematical statistics, mathematical modeling, synthesis and analysis of cryptosystems were used. The purpose of the dissertation is to increase the level of imitation and confidentiality of information in information processing systems in cyber attacks, through the development and implementation of adequate conditions for the application of methods and models to ensure reliable cryptosecurity of such systems. To ach 1) the set of actual threats and vulnerabilities of the information circulating in information processing systems is investigated;

2) the specified models of the violator and threats, and also the automatic model of safety of functioning of control channels of information processing systems in the conditions of influence of cyberattacks are constructed;

3) a method for generating a stream of substitutions for a multi-alphabetic substitution cipher has been developed to ensure the confidentiality and integrity of information in information processing systems;

4) the method of detection of attacks on software implementations of means of cryptographic protection of information in information processing systems is developed;

5) the model of functioning (cryptoscheme) of the encoder of multialphabetic replacement (module of cryptographic protection of the information) in information processing systems is developed;

6) improved the method of evaluating the effectiveness of modern cryptosystems. ieve this goal in the work:

The method of detecting attacks on software implementations of cryptographic information protection in information processing systems is developed on the basis of the requirements of regulatory documents of cryptographic information protection system on the need to control the integrity of software code, detecting failures and errors of operators. The method is based on refined models of violators and threats, as well as an automatic model of security of SOI control channels under the influence of cyberattacks and is to study the statistics of abnormal behavior (state) of cryptographic information protection and is to identify changes in mathematical expectation in some new random sequence from the original. When developing the method, it was assumed that in the case of normal operation of the information processing system, some flow of measured data from the object of control is stationary, and after the occurrence of some event, it changes the distribution of its values. It is necessary to identify the moment of occurrence of this situation as accurately as possible and to make a decision on the anomaly of the situation. As a result, the application of the method of detecting attacks on software implementations of cryptographic protection of information in the information processing system and allows to detect the moment of occurrence of a critical situation and decide on its anomaly, restore the secret key with which encrypted messages are created and / or reveal the original value of encrypted information in the SOI, as well as reduce the cost of the system for detecting attacks on software implementations of CCI by about 25%.

The method of generating a flow of substitutions for the BAZ cipher in order to provide imitation-resistant encryption in SOI, in turn, is based on solving the problem of rapid formation of substitutions from a symmetric group of substitutions Sn. The essence of the method is to form the bottom line of the substitution substitution in a unique set of substitutions from a random (or pseudo-random) uniformly distributed sequence (RRVP). To do this, use a stream of random numbers $j_0, j_1,\ldots, j_m,\ldots \in Z_n$ so that the character that is present in the formed part of the line is rejected and not used in the future. Thus, the process can take quite a long time. As a result, the application of

the method of generating a flow of substitutions using the BAZ cipher to provide imitation-resistant encryption in SOI (multi-alphabetic replacement code based on the original speed algorithm for substituting substitutions) will increase the reliability of information protection in SOI, as well as reduce the likelihood of forgery. applying a value of the order of 10^{-6}.

Functioning model (cryptoscheme) of BAZ encoder (CCI module). The model includes a PVP generator, a substitution substitution generation algorithm, a control and lock system, and an encryption node. The basis for the model was the model and method of detecting attacks on software implementations of CCI in SOI and the method of generating the flow of substitutions using the code BAZ. Its quintessence is that it involves:

first, when initializing the key - the generation of PVP in the appropriate block;

secondly, checking the correct operation of the PVP generator in the block "Control and blocking node";

thirdly, the supply of PVP blocking signal in case of failure of the PVP generator and its stop until the restoration of proper operation;

fourth, with a positive pass of the PVP test: - the start of the algorithm for generating substitution substitutions with subsequent data encryption; - check in the block "Control and blocking node".

The implementation of this model allows, in contrast to existing ones, to ensure the confidentiality and integrity of information circulating in the SOI and in the context of cyber attacks to ensure the actual functional security and survivability of the system itself.

The last step of the dissertation research was to improve the method of evaluating the effectiveness of cryptosystems in SOI.

It is known that in cryptography as an assessment of the efficiency of cryptosystems, usually use the ratio p/R, where R is the minimum complexity of the method of cryptanalysis, calculated in the basic operations of computer technology, p is the probability of successful implementation of this method. In this paper, this assessment has been further developed in terms of practical application to assess the relative effectiveness of the information security system in SOI in cyber attacks. Given the material nature of the SOI owner's losses in the event of the success of an unauthorized interference in its work due to a successful cyber attack and based on the need for the

attacker to incur certain material costs to carry out this attack, it is logical to assess the level of relative effectiveness of the defense system. use a slightly different ratio.

In the process of performing the dissertation the following main results were obtained.

1. A method of generating a flow of substitutions using the BAZ cipher to ensure imitation-resistant encryption in SOI, the introduction of which allows to choose a degree of substitutions that would provide sufficient cryptocurrency performance and would be rational to protect messages from forgery.

2. A method of detecting attacks on software implementations of cryptographic information protection in SOI has been developed, the implementation of which allows to identify the moment of occurrence of a certain critical situation and make decisions on further actions.

3. The model of functioning (cryptoscheme) of the BAZ encoder (module of cryptographic protection of information) in SOI which is the complex decision within two previous methods and which implementation allows to provide confidentiality and integrity of the information circulating in SOI and in the conditions of cyberattacks to provide actually functional security and the survivability of the system itself.

4. Improved the method of assessing the effectiveness of cryptosystems, which by taking into account the material nature of the losses of the SOI owner in case of success of unauthorized interference in its work, allows to determine the limits for relative effectiveness of information protection in SOI in cyber threats.

5. An experimental study of the operation model of the BAZ encoder was carried out by simulating the process of attack on the software implementation of the KZI tool. This, firstly, confirmed the viability of the crypto scheme and, secondly, in a time scale close to real allowed to ensure that the application of the model will reduce the time to detect an attack by about 20%.

Keywords: information processing system, confidentiality, value, automated system, multi-alphabetic replacement, imitation resistance, cryptoscheme, cryptoalgorithm, cryptosystem, cryptographic protection.

СПИСОК ОПУБЛІКОВАНИХ ПРАЦЬ ЗА ТЕМОЮ ДИСЕРТАЦІЇ

Колективна монографія в іноземному науковому виданні:

1. Opirskyy Ivan, Ivanchenko Ihor, Platonenko Artem, **Skladannyi Pavlo**, Roshchuk Mariia. Use of near field communication technology for automated profile replication. Bielsko-Biala (Poland) – Wydawnictwo Naukowe Akademii Techniczno-Humanistycznej w Bielsku-Białej. – С. 153–161

Публікації у наукових фахових виданнях України:

2. Семко В. В., Бурячок В. Л., Толюпа С. В., **Складанний П. М.** Модель управління захистом інформації в інформаційно-телекомунікаційній системі. *Вісник Національного університету "Львівська політехніка". Серія: Радіоелектроніка та телекомунікації* : збірник наукових праць. 2015. № 818. С. 151–155.

3. Семко В. В., Бурячок В. Л., Толюпа С. В., **Складанний П. М.** Ситуаційне управління доступом в інформаційно-телекомунікаційній системі. *Проблеми телекомунікацій*. 2015. №2. С. 54–61.

4. Гулак Г. М., Семко В. В., **Складанний П. М.** Модель системи виявлення вторгнень з використанням двоступеневого критерію виявлення мережних аномалій. *Сучасний захист інформації*. 2015. № 4. С. 81–85.

5. Киричок Р. В., **Складанний П. М.**, Бурячок В. Л., Гулак Г.М., Козачок В.А. Проблеми забезпечення контролю захищеності корпоративних мереж та шляхи їх вирішення. *Наукові записки Українського науково-дослідного інституту зв'язку*. 2016. № 3. С. 48–61.

6. Гулак Г.М., Козачок В. А., **Складанний П. М.** та ін. Системи захисту персональних даних в сучасних інформаційно-телекомунікаційних системах. *Сучасний захист інформації*. 2017. № 2. С. 65–71.

7. Гулак Г. М., Бурячок В. Л., **Складанний П. М.** Швидкий алгоритм генерації підстановок багатоалфавітної заміни. *Захист інформації*. 2017. №2. С. 173–177.

8. Гулак Г.М., **Складанний П. М.** Забезпечення гарантоздатності автоматизованих систем управління та передачі даних безпілотних літальних апаратів. *Математичні машини та системи*. 2017. № 3. С. 154-161.

9. Roy Y. V., Mazur N. P., **Skladannyi P. M**. Audit of Information Security is the basis of Effective Protection of the Enterprise. *Cybersecurity: Education, Science, Technique*. 2018. No. 1. P. 86—93. https://doi.org/10.28925/2663-4023.2018.1.8693

10. Гулак Г.М., Кузьменко Л.В., Складаний П.М., Бурячок В.Л. Криптовірологія: загрози безпеки гарантоздатним інформаційним системам і заходи протидії шифрувальним вірусам. *Кібербезпека: освіта, наука, техніка*. Том 2. No10 (2020). С. 6 – 30.

Публікації у наукометричних базах Scopus і Web of Science:

11. Lakhno V., Buriachok V., Parkhuts L., Tarasova H., Kydyralina L., **Skladannyi P**., Skrypnyk M., Shostakovska A. Development of a conceptual model of adaptive access rights management with using the apparatus of Petri nets. International Journal of Civil Engineering & Technology (IJCIET), Volume 9, Issue 11, November 2018. P. 95-104 (**Scopus**).

12. Buriachok V., Sokolov V., **Skladannyi P.** Security Rating Metrics for Distributed Wireless Systems. Proceedings of the 8th International Conference on “Mathematics. Information Technologies. Education” (MoMLeT&DS’2019), June 2–4, 2019: abstracts. —Vol. 2386. —Aachen : CEUR, 2019. — P. 222–233 :(**Scopus**).

13. Solomentsev O., Zaliskyi M., **Skladannyi P.** Operation system for modern unmanned aerial vehicles. International Workshop on Cyber Hygiene, CybHyg 2019. 2019. Vol. 2654. P. 363—374. (**Scopus**).

Тези наукових доповідей:

14. **Складанний П.М**. Аналіз типів атак на державні інформаційні ресурси / Міжнародна науково-технічна конференція “Сучасні інформаційно-телекомунікаційні технології”. Київ: ДУТ, 2014. С.20.

15. **Складанний П.М.** Модель загроз безпеки криптосистем. / І Міжнародна науково-технічна конференція «Актуальні проблеми розвитку науки і техніки» (Київ, 20 жовтня 2015р.). Київ: ДУТ, 2015. С.35-37.

16. **Складанний П.М.** Псевдовипадкові послідовності та методи захисту інформації. / II Міжнародна науково-технічна конференція «Актуальні проблеми розвитку науки і техніки» (Київ, 12 грудня 2015р.). Київ: ДУТ, 2015. С.55.

17. **Складанний П.М.** Визначення критеріїв безпеки криптосистем. / Міжнародна науково-технічна конференція студентства та молоді “Світ телекомунікацій та інформатизації”. Київ: ДУТ, 2015. С.14-15.

18. **Складанний П.М.** Принцип побудови шифра на основі ГОСТ 28147. / III Міжнародна науково-технічна конференція «Актуальні проблеми розвитку науки і техніки» (Київ, 02 березня 2016р.). Київ: ДУТ, 2016 С.74.

19. Гулак Г.М., **Складанний П.М.**Формування вимог щодо забезпечення гарантоздатності автоматизованих систем переробки інформації й управління критично-важливими об’єктами інфраструктури. / II Всеукраїнська науково-практична конференція «Кібербезпека в Україні: правові та організаційні питання» (Одесса, 17 листопада 2017р.). Одеса: ОДУВС, 2017 С.12-14.

20. Гулак Г.М., Кащук В.І., **Складанний П.М.** Уточнена модель порушника та модель реалізації кібератак в системах управління технологічними процесами / IX Всеукраїнська науково-практична конференція «Актуальні проблеми управління інформаційною безпекою держави» (Київ, 30 березня 2018р.): Київ: Нац. акад. СБУ, 2018. С. 47-49

21. Гулак Г.М., **Складанний П.М.** Раціональний вибір степені підстановок шифру багатоалфавітної заміни та джерела рівномірно розподіленої випадкової послідовності / I Міжнародна науково-практична конференція “Проблеми кібербезпеки інформаційно-телекомунікаційних систем” (PCSITS) (Київ, 05-06 квітня 2018 року.) Київ: КНУ імені Т. Шевченка, 2018 С. 27-31.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

АЗ	–	апаратний засіб
АПЗ	–	апаратно-програмний засіб
АС	–	автоматизована система
БАЗ	–	багатоалфавітна заміна
БШ	–	блоковий шифр
ЕЦП	–	електронний цифровий підпис
ДССЗЗІ	–	Державна служба спеціального зв'язку та захисту інформації України
КЗІ	–	криптографічний захист інформації
КС	–	комп’ютерна система
НЖМД	–	накопичувач на жорсткому магнітному диску
ОП	–	оперативна пам’ять
ОС	–	операційна система
ПВП	–	псевдовипадкова послідовність
ПЗ	–	програмний засіб
ПЗРА	–	програмний засіб реалізації атак
СОІ	–	Система обробки інформації
ТЗІ	–	технічний захист інформації
	–	
	–	
	–	
	–	
	–	
	–	
	–	
	–	

ВСТУП

Обґрунтування вибору теми дослідження. У сучасному світі спостерігається постійне зростання сфер застосування різноманітних автоматизованих систем обробки інформації (СОІ). СОІ забезпечують постійне функціонування об'єктів сектору національної безпеки і оборони, банківського сектора, промисловості, транспорту, зв'язку, енергетики. Використання в таких системах у якості транспортної мережі глобальної мережі Інтернет або радіоканалів підвищує, як відомо, ризики погіршення їх гарантоздатності. В умовах кібератак це може привести до порушення в СОІ, як цілісності даних керування (руйнування, зловмисна підміна) внаслідок непередбачуваних змін системи та послуг, так і їх конфіденційності внаслідок отримання неавторизованого доступу до інформації про послуги в них. В свою чергу, це неодмінно призведе до суттєвого зростання ризиків втрат як для органів державного і військового управління, так і для суб'єктів господарювання державного і приватного секторів економіки і, як результат, взагалі може вивести сучасне суспільство на межу як локальних, так і глобальних техногенних катастроф. Проблеми забезпечення імітостійкості та конфіденційності таких систем ускладнюються низкою негативних чинників, а саме:

щорічним зростанням кількості реалізованих кібератак на СОІ, наслідком яких були значні фінансові та матеріальні збитки;

невизначеністю на законодавчому рівні особливостей захисту інформації (даних керування), що циркулює в СОІ;

нормативною неврегульованістю питання застосування технологій криптографічної переробки інформації для забезпечення імітостійкості та конфіденційності;

відсутністю вітчизняних розробок засобів криптографічного захисту інформації, що забезпечують швидкісне імітостійке шифрування.

Зважаючи на те, що в умовах зростання кількості та потужності кібератак суттєво ускладнюється вирішення завдання забезпечення імітостійкості та

конфіденційності каналів передачі даних та/або команд управління (далі – інформації) в СОІ, як їх спроможності надавати послуги і сервіси, яким можна виправдано довіряти, внаслідок того, що саме вони зазнаватимуть найбільш руйнівних кібератак, та враховуючи, що в деяких випадках побудови СОІ, зокрема, в разі застосування мереж загального користування та радіоканалів, не існує більш дієвого механізму забезпечення конфіденційності та імітостійкості каналів передачі команд управління та даних, ніж технології криптографічного захисту інформації (КЗІ), **наукове завдання дисертаційної роботи** полягає у розробленні теоретичних і прикладних засад побудови та забезпечення методами криптографічної обробки інформації імітостійкості та конфіденційності даних в СОІ з урахуванням множини кіберзагроз та потенційно можливих наслідків їх реалізації. В цей час воно залишається невирішеним в повному обсязі й тому потребує свого розв’язання.

Дослідженню проблем забезпечення цілісності та конфіденційності данних в СОІ присвячені роботи A. Avizienis, В.С. Харченка, О.В. Федухіна, В.Л. Бурячка, В.Б. Дудікевіча, J.-C. Laprie, B Randell, C. Landwehr, I.E. Dobson, С.В. Зибіна, О.В.Копійки, О.М.Трофимчука, Д.В. Стефанишина, розв’язку задач імітостійкого шифрування, що не поширює помилок, присвячені роботи А.В. Бабаша, М.М. Глухова, І.Д. Горбенка, А.Ю. Зубова, В.М. Рудницького, В.О. Устименка, Г.П. Шанкина, Shannon C.E., Diffie W., Hellman M. та багатьох інших. Особливість вирішення цих завдань обумовлюється різноманіттям методів побудови СОІ, відсутністю комплексного підходу до забезпечення коректного функціонування СОІ, а також наявністю невизначеностей, викликаних постійно змінними умовами функціонування таких систем. Зазначене, як результат, негативно впливає на загальний рівень імітостійкості та конфіденційності даних в СОІ. Зважаючи на таке, розробка моделей та методів забезпечення імітостійкості КЗІ в СОІ, визначених специфічними умовами функціонування, які за рахунок застосування імітостійкого шифрування, введення системи контролю правильності функціонування та блокування роботи каналу інформаційного обміну, дозволили б

підвищити рівень імітостійкості і конфіденційності та запобігти несанкціонованим діям щодо інформації в СОІ, саме **й визначає актуальність теми дослідження**.

Зв'язок роботи з науковими програмами, планами, темами. Дисертаційна робота пов'язана з вирішенням науково-технічних задач, сформульованих в Стратегії кібербезпеки України, затвердженої Указом Президента України №96/2016 від 27.01.2016 р., в Стратегії національної безпеки України, затвердженої Указом Президента України № 287/2015 від 26.05.2015 р., а також в «Порядку оцінки стану захищеності державних інформаційних ресурсів в інформаційних, телекомунікаційних та інформаційно-телекомунікаційних системах» затвердженому Наказом №660 Адміністрації Державної служби спеціального зв'язку та захисту інформації України від 02.12.2014 р. Частина завдань дисертаційної роботи вирішена автором в рамках науково-дослідної роботи (НДР) на тему «Базис-Наука» (номер державної реєстрації 0119U000042дс), яку виконано у Інституті проблем математичних машин і систем НАН України в 2020 році.

Мета і завдання дослідження. Метою дисертаційної роботи є підвищення рівня імітостійкості та конфіденційності інформації в системах обробки інформації в умовах кібератак, за рахунок розробки й впровадження адекватних умовам застосування методів і моделей забезпечення надійного криптозахисту таких систем. Для досягнення поставленої мети в роботі необхідно розв'язати такі **завдання**:

1) дослідити множину актуальних загроз і уразливостей інформації, що циркулює в СОІ;

2) побудувати уточнені моделі порушника та загроз, а також автоматну модель безпеки функціонування каналів управління СОІ в умовах впливу кібератак;

3) розробити метод генерації потоку підстановок для шифру багатоалфавітної заміни (БАЗ) для забезпечення в СОІ конфіденційності та цілісності інформації;

4) розробити метод виявлення атак на програмні реалізації засобів КЗІ в СОІ;

5) розробити модель функціонування (криптосхему) шифратора БАЗ (модулю криптографічного захисту інформації) в СОІ;

6) вдосконалити метод оцінки ефективності застосування сучасних криптосистем.

Зважаючи на таке, ***об'єктом дослідження*** в роботі є процеси створення та використання нових моделей та методів забезпечення імітостійкості та конфіденційності КЗІ в СОІ. ***Предметом дослідження*** – моделі та методи для забезпечення імітостійкості та конфіденційності даних в СОІ в умовах зростання потужності кібератак та ймовірності цільового ураження систем.

***Методи дослідження*.** При вирішенні поставлених задач в дисертаційній роботі було використано методи теорії ймовірностей та математичної статистики, математичного моделювання, синтезу та аналізу криптосистем.

Наукова новизна одержаних результатів. Новими результатами, отриманими в дисертаційній роботі є:

1) вперше розроблений метод виявлення атак на програмні реалізації засобів криптографічного захисту інформації в СОІ, впровадження якого шляхом *реалізації двоступеневого критерію виявлення аномалій* дозволяє своєчасно виявити момент настання певної критичної ситуації та прийняти рішення щодо подальших дій;

2) вперше розроблена модель функціонування (криптосхема) шифратора БАЗ (модулю криптографічного захисту інформації) в СОІ, впровадження якої *за рахунок методу виявлення атак на програмні реалізації засобів криптографічного захисту інформації в СОІта методу генерації потоку підстановок для шифру БАЗ* дозволяє забезпечити конфіденційність і цілісність інформації, що циркулює в СОІ, та в умовах кібератак підвищити функціональну безпеку та живучість самої системи;

3) вперше розроблений метод генерації потоку підстановок шифру багатоалфавітної заміни для забезпечення в СОІ конфіденційності та цілісності інформації, впровадження якого *за рахунок реалізації імітостійкого шифрування на основі оригінального швидкісного алгоритму формування потоку підстановок замін, критерію вибору степеню таких замін та процедури оцінки якості послідовності підстановок шифру багатоалфавітної заміни* дозволяє обрати таку степень підстановок, яка б забезпечувала достатню швидкодію криптоперетворення та була б раціональною для забезпечення захисту повідомлень від підробки;

4) удосконалений метод оцінки ефективності застосування криптосистем, на базі *врахування співвідношення середнього значення максимальних втрат власника СОІ у випадку успішних кібератак на систему захисту до мінімальної вартості*

реалізації таких атак, що дозволило, на відміну від існуючих, визначити границі для відносної ефективності системи захисту інформації в СОІ в умовах реалізації кіберзагроз.

Практичне значення одержаних результатів у сукупності складає підґрунтя для забезпечення імітостійкості та конфіденційності каналів передачі даних (команд управління) в СОІ.

Одержані результати дозволяють:

визначити границі для відносної ефективності системи захисту інформації в СОІ в умовах кібератак;

знизити ймовірність підробки команди управління до прийнятної для практичного застосування величини, що оцінується величиною 10^{-6};

знизити час на виявлення атаки приблизно на 20%;

знизити вартість системи виявлення атак на програмні реалізації засобів КЗІ приблизно на 25%.

Аналіз можливості застосування запропонованих моделей і методів забезпечення конфіденційності та імітостійкості даних було досліджено на прикладі дистанційно пілотованих літальних апаратів (ДПЛА), під час якого було з'ясовано, що їх застосування дає можливість своєчасного автоматичного прийняття рішення про його перехід в режим автономного виконання завдань в разі виявлення реальних загроз штатному функціонуванню та можливості перехоплення каналу управління.

Отримані науково-практичні результати були перевірені за допомогою створеного в рамках роботи макету моделюючого комплексу інформаційної технології криптографічної обробки інформації. В ході імітаційного моделювання шифру БАЗ з достатньо високим рівнем надійності були підтверджені теоретичні розрахунки дослідження.

Результати дисертаційних досліджень реалізовані й впроваджені в:

Інституті проблем математичних машин і систем НАН України під час виконання НДР «Базис-Наука» (державний номер реєстрації 0119U000042дс) спрямованої на вирішення питань побудови мережі гарантоздатних ситуаційних центрів сектору безпеки і оборони (акт впровадження від 30.11.2020 р.).

Національному центрі управління та випробувань космічних засобів під час виконання НДР «Розробка науково-технічних пропозицій з організації віддаленого управління станціями оптико-електронних спостережень типу 1 та типу 2» (номер держ. реєстрації 0120U105420), що дозволило забезпечити цілісність та конфіденційність команд управління станціями оптико-електронних спостережень в режимі віддаленого доступу (акт впровадження від 30.11.2020 р.).

Київському університеті імені Бориса Грінченка в рамках навчальних дисциплін «Методи побудови та аналізу криптосистем», «Математичні методи криптографії» та впроваджені в програмно-апаратне забезпечення «Центру технологій захисту інформаційних активів» при розгортанні Лабораторії криптографічного та технічного захисту інформації (акт впровадження № 71-н від 30.11.2020 р.).

Особистий внесок здобувача. Всі наукові результати, що виносяться на захист, одержано здобувачем самостійно. У роботах, написаних у співавторстві, здобувачеві належить: в [1] – профіль захищеності систем зв'язку, [2] – модель управління захистом інформації, [3] – формальна модель управління захистом інформації в інформаційно-телекомунікаційній системі; [4] – модель системи виявлення вторгнень; [5] – аналіз методів контролю захищеності корпоративних мереж, [6] – аналіз методів систем захисту персональних даних, [7] – алгоритм генерації підстановок заміни, [8] – криптосхема реалізації шифру БАЗ та блоку виявлення атак, [9] – застосування організаційних і технічних засобів захисту, [10] – модель утворення прихованих каналів, [11] – алгоритм управління правами доступу, [12] – модель порушника та метод оцінки безпеки бездротових систем розподілу; [13] – обгрунтування показників ефективності, часових ресурсів та витрат; [19] – формування вимог щодо гарантоздатності АСУ; [20] – модель порушника та реалізації кібератак, [21] – алгоритм вибору степеню підстановок.

Апробація результатів дисертації. Основні положення і результати досліджень доповідалися та обговорювалися на І Міжнародній науково-технічній конференції «Актуальні проблеми розвитку науки і техніки» (м. Київ, ДУТ, 2015), ІІ Міжнародній науково-технічній конференції «Актуальні проблеми розвитку науки і техніки» (м. Київ, ДУТ, 2015), ІІІ Міжнародній науково-технічній конференції

«Актуальні проблеми розвитку науки і техніки» (м. Київ, ДУТ, 2016), Міжнародній науково-технічній конференції «Сучасні інформаційно-телекомунікаційні технології» (м. Київ, ДУТ, 2015), Міжнародній науково-технічній конференції студентства та молоді «Світ телекомунікацій та інформатизації» (м. Київ, ДУТ, 2015р.), II Всеукраїнській науково-практичній конференції «Кібербезпека в Україні: правові та організаційні питання» (Одеса, 17 листопада 2017р.), IX Всеукраїнській науково-практичній конференції «Актуальні проблеми управління інформаційною безпекою держави» (Київ, 30 березня 2018р), I Міжнародній науково-практичній конференції «Проблеми кібербезпеки інформаційно-телекомунікаційних систем» (PCSITS) (Київ, 5-6 квітня 2018 р.), VIII Міжнародній конференції «Математика. Інформаційні технології, Навчання (м. Львів, 2-4 червня 2019р.), Міжнародній конференції Cyber Hygiene (30 листопада 2019р).

Публікації. За результатами дисертаційної роботи опублікована 21 наукова праця, 1 колективна монографія, 9 наукових статей, написаних у співавторстві й опублікованих у наукових спеціалізованих фахових виданнях України, 3 наукові праці, що входять до наукометричної бази SCOPUS. Разом з тим основні наукові результати додатково відображені у 9 тезах доповідей на семінарах та науково-практичних конференціях. Із праць, що опубліковано в співавторстві, у дисертаційній роботі використано виключно ті результати, які одержано здобувачем особисто.

Обсяг і структура дисертації. Дисертаційна робота складається зі вступу, трьох розділів, висновків, додатків і списку використаних літературних джерел зі 120 найменувань та 3 додатки. Повний обсяг дисертації складає 150 сторінок машинописного тексту. Основний зміст викладений на 130 сторінках. Робота ілюстрована 25 рисунками та 14 таблицями.

Розділ 1
АНАЛІЗ СТАНУ ПРОБЛЕМИ ЗАБЕЗПЕЧЕННЯ ІМІТОСТІЙНОСТІ ТА КОНФІДЕНЦІЙНОСТІ В СИСТЕМАХ ОБРОБКИ ІНФОРМАЦІЇ

Закон України «Про основні засади забезпечення кібербезпеки України» дає таке визначення автоматизованих систем обробки інформації (далі – СОІ): «…це автоматизована або автоматична система, яка є сукупністю обладнання, засобів, комплексів та систем обробки, передачі та приймання, призначена для організаційного управління та/або управління технологічними процесами (включаючи промислове, електронне, комунікаційне обладнання, інші технічні та технологічні засоби) незалежно від наявності доступу системи до мережі Інтернет та/або інших глобальних мереж передачі даних» [1]. Наочно його можна проілюструвати за допомогою рис. 1.1, на якому подана умовна схема дворівневої СОІ із зображенням наступних об’єктів: центр управління (ЦУ) вищого рівня, який отримує технологічну інформацію та на підставі її аналізу формує певні команди управління для підпорядкованих ЦУ, які в свою чергу керують деякою кількістю підпорядкованих об’єктів (технологічним обладнанням).

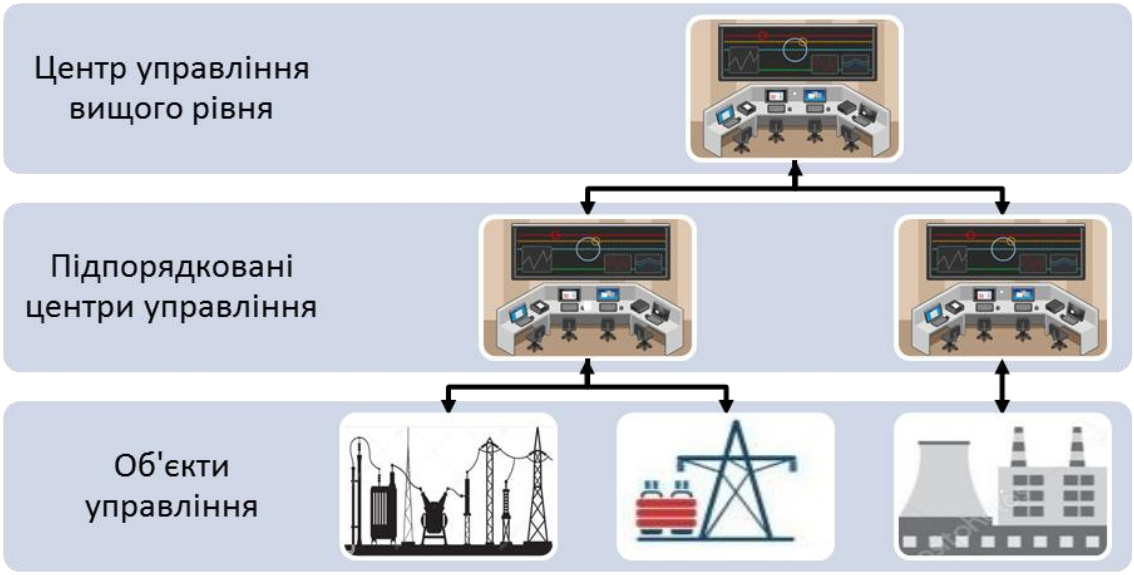


Рис. 1.1. Схема дворівневої СОІ

Слід зауважити, що використання у якості транспортної мережі СОІ глобальних мереж або радіоканалів захист локальних мереж центрів управління та

кінцевого обладнання не виключає потенційної можливості реалізації несанкціонованого втручання в їх роботу з боку каналів зв’язку.

Доцільно також звернути увагу, що вимоги поточної редакції закону про забезпечення кібербезпеки [1], інших законів або підзаконних актів не поширюються на відносини та послуги, пов’язані із змістом інформації, що обробляється (передається, зберігається) в технологічних системах. Тобто відповідні питання криптографічного та технічного захисту інформації в таких системах залишаються неунормованими, що створює передумови до неналежного ставлення до питань забезпечення імітостійкості та конфіденційності, як наслідок, до підвищення ризиків порушення сталого функціонування СОІ.

Про необхідність унормування криптографічного та технічного захисту інформації в СОІ писали майже багато дослідників цієї теми, наприклад, в роботі [2] зазначається, що нагальною сьогодні є проблема підвищення ефективності протидії шифрувальним вірусам та відновлення зашифрованих файлів шляхом застосування активних дій у комп’ютерному середовищі і утворення каналів впливу на програмні засоби, що реалізують зловмисні криптографічні перетворення, в роботі [3] актуалізуються питання сертифікації та вибору серед існуючого розмаїття ПЗ однакового функціонального призначення такого, яке за параметрами і функціями відповідало б вимогам та відбивало б усі потреби організацій-замовників, дослідження [4] зосереджується на розробці методу оцінювання стану захищеності ІР за рахунок визначення найбільш значимих загроз та структуризації пар загроза-уразливість, автори колективної монографії [5] обгрунтовують необхідність криптографічного та технічного захисту за допомогою технології бездротового зв’язк ближнього радіусу дії.

Лояльне ставлення чинного законодавства щодо безпеки СОІ, способів та методів захисту інформації, яка не становить державної таємниці або захист якої не гарантується державою, з одного боку, надає власникам відповідних систем простору для вибору більш ефективних або більш придатних для конкретних застосувань рішень. З іншого боку, існуючий вакуум методичних рекомендацій щодо забезпечення в СОІ таких важливих складових імітостійкість та

конфіденційність команд управління та даних, що обробляються, може суттєво підвищувати ризики реалізації багатьох загроз [4].

При цьому, що дотепер співвідношення понять імітостійкості та конфіденційності діючими нормативними документами не врегульовані, хоча нормативні документи системи технічного захисту інформації у разі побудови комплексних систем захисту інформації висувають вимоги щодо забезпечення кондиціювання серверних приміщень, застосування безперебійного живлення [6].

Загалом, проблеми сучасного етапу забезпечення імітостійкості та конфіденційності можуть бути подані у вигляді органіграми (рис.1.2), яка поєднує низку об'єктивних і суб'єктивних факторів, згрупованих нами на підставі [7], а саме:

слабка ефективність заходів щодо протидії кіберзагрозам, включаючи наукове забезпечення відповідних процесів;

безсистемність заходів щодо захисту кіберпростору;

недостатній рівень координації та взаємодії між суб'єктами забезпечення кібербезпеки, а також суспільством;

невідповідність стану заходів щодо убезпечення електронних комунікацій рівню їх розвитку, невизначеність на законодавчому рівні статусу інформації, що циркулює у СОІ;

недостатній розвиток організаційно-технічного забезпечення кібербезпеки, низька готовність певних служб забезпечити безпеку СОІ в умовах кібернетичного впливу;

низький рівень захищеності кіберпростору, недостатня увага до рівня захищеності інформаційної інфраструктури СОІ. Про останнє свідчить:

статистика сталого зростання кількості успішних атак у кіберпросторі та спроб несанкціонованого втручання в роботу СОІ;

відсутність конкретних вимог та норм щодо технічного захисту інформації, що циркулює у СОІ та особливостей застосування засобів криптографічного захисту такої інформації (КЗІ);

недосконаліть існуючої нормативно-правової бази, яка регламентує лише впровадження технологій та методів забезпечення безпеки переважно в тих АС, де обробляється інформація, щодо якої законодавчо визначена відповідальність тощо [7, с. 13].

Рис. 1.2. Органіграма основних проблем забезпечення імітостійкості та конфіденційності даних в СОІ

Вивчення та аналіз різноманітних джерел про інциденти з СОІ [8-14] дозволяє зробити висновок, що на складність розв'язання проблем забезпечення імітостійкості та конфіденційності СОІ суттєво впливають такі негативні тренди:

швидке впровадження мережевих технологій, що випереджає розвиток регуляторних механізмів, нормативних вимог до систем та засобів захисту;

підвищення складності й досконалості атак в кіберпросторі, що зменшує імовірність їх своєчасного виявлення;

впровадження в дію нових, надзвичайно уніфікованих шкідливих кодів і технологій анонімізації, що робить атаки успішними й дозволяє порушникам долати певні захисні бар'єри;

швидке старіння систем спостереження за проникненням та баз даних антивірусів, що не дозволяє забезпечувати необхідний рівень безпеки;

використання порушниками вразливостей програмного забезпечення (на рівні операційних систем та прикладних програм) та різного роду хакерських «інновацій» раніше, аніж розробники програмних систем усувають ці вразливості або створюють необхідні інструменти убезпечення.

Зважаючи, що й нині питання щодо одночасного забезпечення імітостійкості та конфіденційності безпеки СОІ в нормативно-правових актах нашої держави не розглядається, а в умовах ведення гібридної війни СОІ можуть зазнавати руйнівних кібернетичних атак, внаслідок яких держава може отримувати значних матеріальних втрат та фінансових збитків, видається доцільним посилити дослідження у сфері розробки та створення нових способів та методів забезпечення імітостійкості та конфіденційності.

1.1. Загрози та уразливості автоматизованих систем обробки даних

Виходячи з наведеного у попередньому розділі визначення, можливо стверджувати, що за своєю сутністю СОІ є сервіс-орієнтованою комп’ютерною системою, тому доцільно скористатися науковим доробком [6; 14-15] для систем цього типу.

Згідно з поширеним визначенням [6], гарантоздатність – це властивість комп’ютерної системи надавати потрібні послуги, щодо яких є підстави для довіри. Гарантоздатність є комплексною властивістю, яка включає наступні первинні характеристики (складові):

безвідмовність, що ототожнюється зі здатністю безперервно надавати коректні (потрібні) послуги;

готовність, яка характеризує доступність ресурсів комп’ютерної системи надавати потрібні послуги;

живучість, як здатність мінімізувати зниження і зберігати у припустимих межах якість та обсяг послуг, що надаються у випадку відмов;

функціональна безпека, як здатність виключати або мінімізувати шкідливі (катастрофічні) наслідки у разі відмов для користувачів, інших систем, оточуючого середовища;

цілісність, яка визначається здатністю виключати несанкціоновані зміни системи та послуг, які надаються;

конфіденційність, як здатність попереджати можливість неавторизованого доступу до інформації про послуги;

достовірність, що визначає можливість вірно оцінювати коректність послуг, що надаються, тобто визначати ступінь довіри до послуги;

обслуговуваність, що характеризує придатність до модифікацій та ремонту.

Систематизація поняття обов’язково передбачає розгляд загроз, наслідком реалізації яких може бути виникнення ситуації з відмовою надання потрібних послуг. Про це зокрема зазначається в роботах [14-17]. Звичайно при цьому розглядаються дефекти системи або її несправності різної природи походження, але в рамках цього дослідження розглядаються лише дефекти, що є наслідком зовнішнього несанкціонованого втручання або інформаційних атак.

У загальному випадку реалізації загроз інформаційної безпеки можуть відбуватися її порушення. При цьому загрози класифікують за результатом їх впливу на інформацію. В комп’ютерних системах, відповідно до стандарту «Загальні положення щодо захисту інформації в комп'ютерних системах від несанкціонованого доступу» [18] розрізняють наступні типи загроз безпеки інформації:

порушення конфіденційності внаслідок несанкціонованого доступу до інформації;

порушення цілісності, зокрема шляхом несанкціонованої модифікації або викрадення інформації; а також порушення автентичності

порушення доступності ресурсів (послуг та інформації), зокрема у разі відмови в обслуговуванні.

Джерела загроз інформаційної безпеки поділяють на:

антропогенні (навмисні, що породжені діями терористичних угрупувань, спеціальних служб і технічних розвідок, інсайдерів, а також ненавмисні, що обумовлені помилками обслуговуючого персоналу);

техногенні (аварії, відмови обладнання);

природні (стихійні лиха, кліматичні умови тощо) [19].

Зважаючи на те, що інформація (команди управління та дані), яка циркулює в СОІ, нормативно не класифікована, в рамках дослідження будемо використовувати термін «технологічна інформація», розуміючи при цьому дані з відносно низьким рівнем конфіденційності та високими вимогами що цілісності під час їх передавання каналами зв’язку.

Відповідно до Положення про порядок розроблення, виробництва та експлуатації засобів криптографічного захисту інформації (КЗІ) [20] залежно від запланованих умов експлуатації засобів КЗІ та відповідно до цінності інформації, що захищається, розрізняють чотири рівні можливостей порушника.

нульовий рівень – ненавмисне порушення конфіденційності, цілісності та підтвердження авторства інформації;

перший рівень – порушник має обмежені кошти та самостійно створює засоби, розробляє методи атак на засоби КЗІ, а також інформаційно-телекомунікаційні системи із застосуванням поширених програмних засобів та електронно-обчислювальної техніки;

другий рівень – порушник корпоративного типу має змогу створення спеціальних технічних засобів, вартість яких співвідноситься з можливими фінансовими збитками, що виникатимуть від порушення конфіденційності, цілісності та підтвердження авторства інформації, зокрема при втраті, спотворенні та знищенні інформації, що захищається. У цьому разі для розподілу обчислень при проведенні атак можуть застосовуватися локальні обчислювальні мережі;

третій рівень – порушник має науково-технічний ресурс, який прирівнюється до науково-технічного ресурсу спеціальної служби економічно розвинутої держави.

Одразу же зазначимо, що нульовий рівень можливостей порушника, який ненавмисно порушує конфіденційність, цілісність або автентичність інформації, не

становить науково-практичного інтересу. Інша справа, порушники першого - третього рівня, які можуть розглядатися в якості потенційних порушників, що намагаються здійснити відповідні деструктивні дії.

Зауважимо, що вимоги вказаного положення поширюються лише на засоби криптографічного захисту державних інформаційних ресурсів або інформації, вимога щодо захисту якої встановлена законами України "Про Державну службу спеціального зв'язку та захисту інформації України" [21], "Про інформацію" [22], "Про електронний цифровий підпис" [23], "Про стандарти, технічні регламенти та процедури оцінки відповідності" [24]. Тобто, як було зазначено раніше, воно не поширюється на технологічну інформацію, але застосування деяких його вимог в рамках цього дослідження доречно з метою забезпечення узгодженості із загальнодержавними підходами.

Найбільш поширеними інструментами, що використовують порушники для досягнення поставлених цілей, є наступні.

1. Ботнет (англ. *botnet= roBOT + NETwork*) – комп'ютерна мережа, що включає декілька хостів, на яких приховано інстальоване шкідливе програмне забезпечення - бот, що дозволяє порушнику виконувати несанкціовані власником зараженого комп'ютера дії з використанням його ресурсів. Зазвичай, ботнет використовують для підбору (розкриття) паролів та іншої критичної інформації систем захисту, побудови атак на відмову в обслуговуванні тощо.

2. Технологія «фішинг» (англ. *phishing*) – вид інтернет-шахрайства, метою якого є отримання конфіденційних даних користувачів, включаючи його логіни і паролі. «Фішинг»-атака реалізується шляхом проведення масових розсилок фіктивних електронних листів від імені реальних суб’єктів інформаційного обміну. У листі міститься пряме посилання на фіктивний сайт - пастку. Після доступу на цей сайт користувача різними психологічними прийомами спонукають ввести свої логін і пароль.

3. Технологія «сніфінг» (англ. *to sniff*) – поширений вид атак за допомогою шкідливого коду – сніферу (англ. *sniffer*), який перехоплює усі пакети від мережевої карти. У підсумку роботи сніферу порушник може отримати великий обсяг

службової інформації щодо маршруту руху перехоплених пакетів, логінів і паролів легальних користувачів.

4. Технології сканування вразливостей системи захисту використовуються для підготовки атак. Мережеві сканери (англ. *network scanners*) - програми, які аналізують топологію мережі і виявляють доступні сервіси. У якості прикладу такої програми можна назвати систему *nmap*. Сканери вразливостей це програми, які здійснюють пошук вразливостей на вузлах мережі і які можуть бути використані для реалізації атак Приклади: система SATAN або *ShadowSecurityScanner*.

Залежно від об’єкта атаки [25], їх можна розділити на наступні дві групи:

атаки на інфраструктуру мережі: інформаційний обмін, протоколи і використовувані в мережі сервісні служби;

атаки на телекомунікаційні служби: служби швидкісної комутації та маршрутизації даних, наприклад, SMDS (стандарт IEEE 802.6) пакетний сервіс для передачі даних в LAN, WAN.

На рис. 1.3 наведена уточнена класифікація комп'ютерних атак, більш детально охарактеризована нами в роботах [26-27] залежно від умов їх реалізації.

	Фактор класифікації	***Тип атаки***
Атаки на комп’ютерні системи	**1.** За характером впливу	**1.1** Пасивна
		1.2 Активна
	2. За умовою початку впливу	**2.1** За запитом від об’єкту атаки
		2.2 За виникненням деякої події
		2.3 З безумовним початком
	3. За наявністю зворотного зв’язку з об’єктом атаки	**3.1** Зі зворотнім зв’язком
		3.2 Односпрямована
	4. По відношенню суб’єкта атаки до об’єкта атаки	**4.1** Всередині сегмента
		4.2 Поза сегментом
	5. За рівнем стека протоколів TCP/IP	**5.1** Рівня додатків (*application layer*)
		5.2 Транспортного рівня (*transport layer*)
		5.3 Мережного рівня (*Internet layer*)
		5.4 Канального рівня (*link layer*)

Рис. 1.3. Класифікація атак на комп’ютерні системи

Комп'ютерні віруси в сенсі їх призначення є програмним засобом реалізації кібератак (ПЗРА) [20]. Їх можна класифікувати залежно від місця розміщення

(рис. 1.4). Зокрема, в залежності від їх місця розташування (виникнення) їх можна поділити на певні типи [28]:

Завантажувальні – віруси, які проникають в завантажувальний сектор пристроїв зберігання даних, таких як жорсткі диски, дискети тощо.

Файлові – тип вірусів, які впроваджуються у виконувані файли (файли з розширенням COM і EXE) і негативно впливають на їх функціональність. Файлово-завантажувальні віруси – віруси, які об'єднують в собі функції відповідних типів вірусів;

Макровіруси (документні) це тип вірусів, які заражають файли офісних систем. Такий вид називають «макровірусами», оскільки зараження системи відбувається за допомогою зараження макросів програми.

Мережеві – тип вірусів, які поширюються за рахунок використання комп'ютерної мережі, тобто мережевих служб і протоколів.

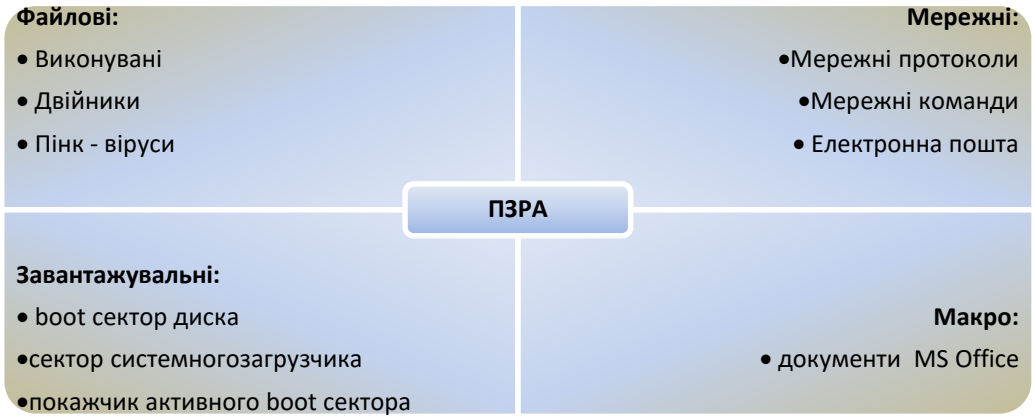


Рис. 1.4. Класифікація ПЗРА (вірусів) за місцем розташування

Практика свідчить, що успішній реалізації атак можуть сприяти різноманітні фактори [29-30], деякі з них узагальнені нами в таблиці 1.1. Характеристика ймовірності успішної реалізації атак для різних типів реалізації засобів КЗІ отримана методом голосування за результатами опитування експертів з питань інформаційної безпеки.

Таблиця 1.1

Фактори що сприяють реалізації атак та оцінка ймовірності їх успішної реалізації для різних типів засобів захисту інформації

2№	Фактори що сприяють реалізації атак (загрози)	Характеристика* ймовірності успішної реалізації атак для різних типів реалізації засобів КЗІ								
		Класи** безпеки ПЗ			Класи** безпеки АПЗ			Класи** безпеки АЗ		
		1	2	3	1	2	3	1	2	3
1.	Ненавмисні помилки операторів обслуговування	В	В	С	С	С	С	С	Н	Н
2.	Несанкціоновані дії операторів безпеки (інсайдери), включаючи вхід до системи в обхід засобів захисту з метою забезпечення в подальшому доступу порушника.	В	В	В	В	С	С	С	С	Н
3.	Збої та відмови технічних засобів та носіїв інформації	В	В	В	С	С	С	С	Н	Н
4.	Навмисне пошкодження або крадіжка обладнання	В	В	В	С	С	С	С	Н	Н
5.	Руйнування даних та команд, нав'язування хибної інформації у кінцевому обладнанні за допомогою спеціальних засобів.	В	В	В	С	С	С	С	С	Н
6.	Відмова в обслуговуванні внаслідок впливу на канал управління	В	В	В	С	В	В	Н	С	Н
7.	Збої та відмови програмного забезпечення внаслідок помилок етапу проектування	С	С	С	С	С	Н	Н	Н	Н
8.	Некоректна інсталяція та конфігурування програмного забезпечення (ОС, прикладне програмне забезпечення)	В	В	В	С	С	С	С	С	С
9.	Пошкодження файлів (у т.ч. системних журналів ОС) внаслідок зовнішніх випадкових факторів	Н	Н	Н	Н	Н	Н	Н	Н	Н
10.	Ураження системи шкідливими кодами (вірусами)	В	В	В	В	С	С	С	Н	Н
11.	Незаконне одержання паролів та інших реквізитів розмежування доступу з наступним маскуванням під зареєстрованого користувача	В	В	В	В	С	С	С	Н	Н
12.	Несанкціоноване відключення засобів захисту	В	В	В	В	В	В	С	С	Н
13.	Підбір та злам паролів	В	С	С	С	С	С	С	С	Н

* - характеристика ймовірності: В – висока, С – середня, Н – низька.

** - клас безпеки засобу криптографічного захисту інформації визначений відповідно до рівня можливостей потенційного порушника безпеки (рівні 1-3).

Найчастіше атака має успіх, якщо в комп'ютерній системі, системі захисту або в організації контролю за станом безпеки існують певні недоліки, які отримали

назву вразливостей. Як показано у [31], вразливості інформаційної безпеки СОІ можуть бути обумовлені:

1. Недоліками (відсутністю) політики й процедур безпеки. Аналіз свідчить, що уразливості політики і процедур в промислових автоматизованих системах управління виникають через відсутність або неповну, неадекватну документацію в галузі безпеки, у тому числі політики і керівництва (процедур), адміністрування аудиту, відновлення.

2. Помилками (вразливостями) апаратних і програмних платформ. Вразливості платформ СОІ можуть виникати через недоліки, помилки, або неякісне обслуговування обладнання та несвоєчасне оновлення програмного забезпечення.

3. Вразливостями адміністрування мереж. Ці помилки можуть бути усунені або нівельовані за допомогою правильного проектування мережі, шифрування мережевих з'єднань, забезпечення контролю фізичного доступу до мережевих компонентів.

Виходячи з визначеної моделі порушника, можливо спрогнозувати наступні цілі зловмисних дій (потенційні загрози):

1) модифікація дійсної команди або реальної інформації про внутрішній стан вузлів мережі (ЦУ) СОІ або керованих пристроїв;

2) формування та надсилання керованому вузлу неприпустимої команди або фальшивої інформації про внутрішній стан від керованого вузла;

3) перехоплення в транспортній мережі окремих команд або частки інформації щодо внутрішніх станів задля їх вилучення.

Підсумовуючі викладене, можна зробити висновок, що для забезпечення безпеки СОІ у плані цілісності та конфіденційності команд управління та даних потрібно вжити адекватних заходів щодо надійного виявлення аномальної поведінки засобів КЗІ та вузлів її мережі (ЦУ) та блокуванню відповідних загроз. Дослідженню цього присвячений наступний підрозділ.

1.2. Шляхи забезпечення імітостійкості та конфіденційності програмних реалізацій засобів криптографічного захисту інформації, що циркулює в СОІ

1.2.1. Методи та механізми криптографічного захисту даних, що передаються в СОІ мережами зв'язку, від спроб підробки

На практиці для захисту від спроб підробки даних, що передаються мережами зв'язку, використовуються різні механізми, включаючи використання електронного цифрового підпису, внесення надлишковості у дані, наприклад, за допомогою кодування, застосування кодів автентифікації повідомлень (англ. *message authentication code - MAC*), позначок часу, «плаваючого» графіка зміни ключів, а також імітостійкого шифрування (табл. 1.2).

Таблиця 1.2

Порівняльний аналіз методів протидії маніпулюванню даними та/ або контролю цілісності

	Метод	Переваги	Недоліки
1.	Електронний цифровий підпис (ЕЦП)	Практично неможливо підробити ЕЦП, що побудоване на основі сучасних алгоритмів. Дозволяє будувати системи з юридично значущим доведенням авторства повідомлення.	Відносно тривалий час формування та перевіряння ЕЦП. ЕЦП для коротких повідомлень може мати довжину, що перевищує їх розмір на два порядки. Для захисту конфіденційності за допомогою симетричних шифрів потрібно мати три ключі: секретний для шифрування, секретний для формування ЕЦП та відкритий для його перевірки. Потребує наявності в системі третьої сторони – центру сертифікації ключів.
2.	Коди автентифікації повідомлень (MAC)	Перевірка цілісності інформації за допомогою лише секретного ключа. Мала ймовірність підробки MAC, що не перевищує величини 2-m, де m – довжина MAC	У випадку забезпечення конфіденційності повідомлень за допомогою симетричних БШ потрібно мати два ключі: шифрування та формування MAC; Генерація MAC за допомогою алгоритму блокового шифрування потребує значного часу.
3.	Використання позначок часу.	Не суттєво збільшує довжину повідомлення. Не потребує суттєвих витрат часу процесора.	Потребує шифрування повідомлень та процедури синхронізації годинників приймача та передавача; Для забезпечення низької ймовірності підробки потрібна велика розрядність цифрового годинника.
4.	«Плаваючий» графік зміни ключів.	Не змінює швидкості виробничої потужності системи.	Не суттєво підвищує імітостійкість; Ускладнює процедури управління ключами.

Продовження табл. 1.2

5.	Імітостійке шифрування	Має значно більшу швидкодію серед інших криптографічних методів захисту. Не втрачає стійкості в разі повторення ключу шифрування.	Генерація псевдовипадкової послідовності (ПВП) підстановок заміни: X1, X2, ..., Xm, ... потребує певного часу, що зменшує пропускну здатність; Відмова блоку генерації ПВП внаслідок несанкціонованого втручання у роботу системи може зруйнувати систему безпеки.

1. Найбільш поширеним методом убезпечення інформаційного обміну та протидії загрозам конфіденційності та автентичності інформації є застосування механізмів формування та перевірки електронного цифрового підпису (ЕЦП) [32].

За визначенням авторів роботи [33], ЕЦП – це дані, які логічно поєднані з інформацією, що захищається? та отримані за допомогою асиметричного криптографічного перетворення, здійсненого над електронним документом його власником. Застосування криптографічного перетворення практично унеможливлює підробку технологічної інформації та забезпечує незаперечність факту перетворення даних конкретною особою.

Існує декілька схем цифрового підпису, найбільш поширеною з них у комерційній сфері є ЕЦП (підтримується багатьма операційними системами) із застосуванням алгоритму RSA.

Припустимо, що абонент А і Б мають спільну криптосистему RSA з параметрами e, d, n.

Для забезпечення цілісності повідомлення M, яке передається від абонента А к Б, абонент А, використовуючи свій секретний ключ d_A і відкритий ключ e_B абонента Б, «підписує» повідомлення за допомогою перетворення: $E_{e_B,n}\left(E_{d_A,n}(M)\right)$.

Для перевірки підпису абонент Б спочатку розшифровує підписане повідомлення за допомогою свого секретного ключа d_B і onhbvyє: $E_{d_A,n}(M)$. Потім, використовуючи відкритий ключ абонента А, він отримає M, виходячи з $E_{d_A,n}(M)$. Істинність повідомлення, а отже і підпису, забезпечується використанням секретного ключа d_A.

Абонент А не може відмовитися від свого повідомлення, якщо він визнає, що секретний ключ відомий тільки йому. Порушник без знання секретного ключа не може ані сформувати, ані зробити зміну повідомлення, що передається в мережі (каналом зв'язку).

Суттєвим недоліком забезпечення цілісності даних за допомогою ЕЦП, як зазначають автори роботи [34], є досить повільна швидкість процедур його формування та перевірки, яка може не відповідати встановленим вимогам щодо часу реакції вузлів СОІ на вхідні команди, а додавання цифрового підпису до відносно коротких команд непропорційно зменшує ефективність використання пропускної здатності транспортної мережі.

Крім того, для забезпечення конфіденційності технологічної інформації потрібен ще симетричний шифр, тоді для захисту буде потрібно мати три ключі: секретний для шифрування, пара секретний/відкритий для формування та перевірки ЕЦП. Також цей підхід потребує наявності в системі третьої сторони – центру сертифікації ключів.

2. Процедура використання коду автентифікації повідомлень (MAC) потребує наявності у абонентів спільного секретного ключа k та алгоритму блокового шифрування (БШ) E_k, за допомогою якого створюється цей код.

Зокрема, криптографічний алгоритм ДСТУ ГОСТ 28147:2009 в одному з режимів роботи дозволяє отримувати код MAC довжиною від 1 до 32 біт.

Його формування відбувається за наступною схемою. Спочатку, відкритий текст M розбивається на блоки довжини 64 бита. Якщо останній блок має меншу довжину, то він доповнюється нулями до довжини 64 біта:

$$M=M_1M_2 \dots M_N$$

Далі, перший блок M_1 зашифровується в режимі простої заміни зазначеного алгоритму (еквівалент режиму ECB) тим самим ключем, що і повідомлення тільки із застосуванням 16 циклів перетворень замість 32.

Результат по бітам за модулем 2 додається до другого блоку M_2, після чого зашифровується. Отриманий блок даних додається до третього блоку відкритого тексту і так далі. Тоді це рівняння можливо записати:

$$I=E_k(M_N\oplus \dots E_k\,(M_3\oplus E_k(M_2\oplus E_k(M_1)) \dots))$$

Де перші 32 бита (або менша кількість біт) отриманого блоку I утворюють код MAC. Код MAC може обчислюватися в процесі шифрування повідомлення або окремо, а передається за звичай у кінці повідомлення.

Найпростішим способом створити незалежний від ключа код MAC – є шифрування повідомлення блоковим алгоритмом в режимах CBC або CFB. Кодом MAC є останній шифрований блок, зашифрований в цих режимах.

Потенційна проблема, пов'язана з безпекою цього методу, полягає в тому, що одержувач повинен знати секретний ключ, і цей ключ дозволяє йому генерувати повідомлення з тим же значенням коду MAC, що і у надісланого повідомлення. Таким чином, код MAC на основі симетричного шифру не дає знання, хто сформував цей код MAC. Звідси випливає, що код MAC на основі симетричного шифру не може повністю заміняти ЕЦП. Крім того, генерація MAC за допомогою алгоритму БШ потребує значного часу, а її застосування в режимі потокового шифрування ускладнено через необхідність дроблення потоку на окремі фрагменти [35].

Перевагами цього методу є можливість перевірки цілісності інформації за допомогою лише секретного ключу, а також мала ймовірність підробки коду MAC, що не перевищує величини 2^{-m}, де m – довжина MAC.

3. Використання позначок часу передбачає, що кожен самостійний фрагмент M технологічної інформації до початку його шифрування за допомогою БШ в режимі OFB доповнюється цифровим значенням часу T його створення: $\langle MT\rangle$. Під час розшифрування отримувач звіряє отримане значення T з поточним часом T_n, у випадку перевищення припустимої величини:

$$\Delta T \le |T - T_n|\,,$$

відповідний фрагмент відхиляється, як такий, що з нез'ясовних причин передавався занадто довго [35].

Перевагою цього методу є несуттєве збільшення початкової довжини відповідного фрагменту та висока швидкодія. Потребує шифрування повідомлень та процедури синхронізації годинників приймача та передавача;

Недоліки методу полягають в тому, що при відносно великої ймовірності підробки потрібна велика розрядність цифрового годинника, а також необхідність забезпечення високої степені синхронізму годинників відправника та отримувача.

4. Застосування поточного шифрування технологічної інформації одночасно з «плаваючим» графіком зміни ключів частково вирішує проблему підробки інформації у мережі, не знижуючи при цьому швидкості роботи СОІ. У той же час, для незначного підвищення імітостійкості потрібно суттєво ускладнити процедури управління ключами. За визначенням [36-38], імітостійкість (англ. *imitation resistance*) це здатність шифру протистояти спробам порушника підміни у мережі (каналі зв'язку) зашифрованої істинної команди (інформації) фіктивною або створення несправжньої інформації, яку з деякою ймовірністю отримувач розшифровує та сприймає як істинну. Несправжня інформація, що сприйнята до виконання отримувачем, вважається нав'язаною.

1.2.2. Імітостійке шифрування, як метод криптографічного захисту технологічної інформації в СОІ

Загальновідомий шлях захисту технологічної інформації від підміни – це її імітостійке шифрування за допомогою БШ [36] у режимі *CBC* (англ. *Cipher Block Chaining*). Це рішення може забезпечити захист від маніпулювання даними у каналі зв'язку, але воно може бути застосоване лише у випадку достатньої надлишковості у технологічній інформації, наприклад, завдяки застосуванню завадостійкого кодування. У той же час, внаслідок слабкої завадозахищеності цього режиму спрощується задача простого (не спрямованого) викривлення порушником інформації в системі.

Для подолання останньої проблеми для розв'язку задачі захисту цілісності замість БШ може бути застосований потоковий шифр багатоалфавітної заміни (БАЗ) [37] на алфавіті $Z = \{0, \dots, 2^{k-1}\}$). Загалом, порівняльний аналіз властивостей блокових та потокових шифрів наведений у таблиці 1.3 [38-39].

Таблиця 1.3

Порівняльний аналіз криптографічних методів забезпечення конфіденційності

Криптографічне перетворення	**Переваги**	**Недоліки**
Блокове шифрування	• Забезпечує імітостійке шифрування в режимі CBC; • Достатньо складні співвідношення між вихідним і шифрованим текстом для того, щоб аналітичні і (або) статистичні методи визначення вихідного тексту і (або) ключа на основі відповідності вихідного і шифрованого тексту були б по можливості такими, що не реалізуються.	•Викривлення одного біту в зашифрованому блоці призводить до викривлення всього блоку у режимі шифрування ECB, або залишкової частини шифротексту в режимі CBC, що вимагає додаткового застосування потужних кодів виправляють помилки. •З двох однакових блоків вихідного тексту виходять однакові блоки шифрованого, що підвищує ризики підробки відповідних повідомлень.
Потокове шифрування	• Висока швидкість шифрування та розшифрування • Потік ключів генерується з короткого основного ключа за допомогою однозначних певних детермінованих алгоритмів.	• Практична реалізація «наддовгих» ключових послідовностей і їх зберігання занадто важке і незручне • Вставка або випадання одного двійкового символу в шифрограмі призводить до неправильного розшифрування інших символів через втрату синхронізації • Число параметрів, що впливають на їх стійкість, істотно більше, ніж для блокових шифрів

Згадаємо, що за визначенням [36] шифром БАЗ з періодом n є криптографічне перетворення $\tilde{X}$, яке визначається ключем $\tilde{k} = (X_1, X_2, \dots X_n)$, де кожна підстановка заміни належить до групи підстановок степеню m: $X_i \in S(Z_m)$ і існує хоча б одна пара $i \neq j$, для якої $X_i \neq X_j$, якщо система рівнянь шифрування послідовних n-грам вихідного тексту $(t_1, t_2, \dots, t_n) \to (s_1, s_2, \dots, s_n)$ має вигляд: $s_i = X(t_i)$, де $i = 1, 2, \dots, n$.

У разі неперіодичного шифру БАЗ використовується деяка псевдовипадкова послідовність (ПВП): $\{\gamma_i \in Z_n,\ i = 1,2, \dots\}$, довжина якої не менше довжини даних, що підлягають шифруванню.

Перевагою цього методу є його достатньо висока швидкодія порівняно з методами ЕЦП та коду МАС. Слід також підкреслити, що шифр БАЗ на відміну від шифру гамування не втрачає стійкості навіть в разі неодноразового використання секретного ключа [40]. «Вузьким місцем» методу є дві проблеми:

а) генерація псевдовипадкової послідовності (ПВП) підстановок заміни: $X_1, X_2, \dots, X_n, \dots$ потребує певного часу, що зменшує загальну пропускну здатність СОІ;

б) відмова блоку генерації ПВП внаслідок несанкціонованого втручання в його роботу може зруйнувати систему безпеки.

На підставі викладеного вважаємо, що у випадку розв'язання «вузьких місць» БАЗ в визначених умовах може вважатися найкращим для реалізації методом забезпечення конфіденційності послуг і протидії загрозам маніпулювання даними в мережах СОІ. Надалі ми розглядатимемо лише випадкові величини з дискретним розподілом, тобто для всіх i випадкові величини γ_i приймають значення на множині $Z_n = \{0,1,\dots,n-1\}$, що називається алфавітом послідовності. За найбільш поширеним визначенням [13], дискретна рівномірна розподілена випадкова послідовність (РРВП) $\{\gamma_i \in Z_n,\ i = 1,2, \dots\}$ - це послідовність незалежних у сукупності випадкових величин з рівномірним розподілом, тобто: $P(\gamma_i = l) = n^{-1}$ *для* $\forall l \in Z_n$, де n - об'єм алфавіту Z_n.

Випадкова послідовність має наступні властивості [41]:

1. Будь-яка підпослідовність послідовності $\{\gamma_i \in Z_n,\ i = 1,2, \dots\}$ також є РРВП;

2. Сума за модулем n РРВП $\{\gamma_i \in Z_n,\ i = 1,2, \dots\}$ та будь-якої послідовності $\{m_i \in Z_n,\ i = 1,2, \dots\}$, що від неї не залежить:

$$\{c_i = m_i + \gamma_i \mathrm{mod} n,\ i = 1,2, \dots\},$$

також є РРВП.

Послідовність називають псевдовипадковою (ПВП), якщо не існує поліноміального (імовірнісного) алгоритму, який може відрізнити цю послідовність від суто випадкової.

1.2.3. Способи реалізації засобів криптографічного захисту даних, що передаються в СОІ мережами зв'язку

Методи та механізми криптографічного захисту даних, що описані в попередніх підрозділах роботи, можуть бути реалізовані в програмних (ПЗ), апаратно-програмних (АПЗ) та апаратних (АЗ) засобах, а саме: публічних криптографічних ключах; сертифікатах відкритих ключів; самостійно підписаних сертифікатах; довірчих точках; одноразових паролях, пов'язаних з лічильником, внутрішньою датою й часом.

Їх основні переваги та недоліки, про які більш детально – в роботі [41], узагальнені нами в табл.1.4.

Таблиця 1.4

Порівняльна таблиця властивостей реалізації засобів КЗІ

Тип реалізації засобу	**Переваги**	**Недоліки**
ПЗ	1. Низька вартість засобу 2. Гнучкість	1. Слабка захищеність від атак на реалізацію 2. Невисока продуктивність
АПЗ	1. Підвищений рівень безпеки 2. Підвищена продуктивність	1. Середній рівень захисту від атак на реалізацію 2. Підвищена вартість
АЗ	1. Висока продуктивність 2. Відомі механізми блокування побічних каналів витоку	1. Надто висока вартість 2. Орієнтованість на конкретні протоколи 3. На мобільних об'єктах може висувати додаткові масо-габаритні вимоги

Засоби КЗІ можуть бути класифіковані за чотирма рівнями безпеки, які відрізняються критичними (англ.. *critical security parameter; CSP*) та чутливими (*sensitive security parameters; SSP*) параметрами безпеки.

Рівень безпеки 1 забезпечує базовий рівень безпеки. Базові вимоги безпеки вказані для криптографічного модуля (наприклад, має використовуватися щонайменше одна схвалена функція безпеки або схвалений метод створення конфіденційного параметра безпеки). Модулі програмного забезпечення або вбудованого програмного забезпечення можуть працювати в незмінному, обмеженому або змінному операційному середовищі. За межами основних вимог для промислових компонентів для рівня безпеки 1 немає конкретних механізмів забезпечення фізичної безпеки апаратного криптографічного модуля. Задокументовано реалізовані методи пом'якшення неінвазивних атак або пом'якшення інших атак.

Прикладами криптографічного модуля із рівнем безпеки 1 є апаратне шифрування у персональному комп'ютері (ПК) або криптографічний інструментарій портативного пристрою, або комп'ютера загального призначення.

Такі реалізації ідеально підходять для застосувань безпеки, де контроль фізичної безпеки, мережевої безпеки та адміністративних процедур надаються за межами модуля, але в середовищі, в якому він функціонує. Наприклад, реалізація криптографічного модуля із рівнем безпеки 1 може бути економічно ефективнішою, ніж відповідні модулі на більш високих рівнях довіри, які забезпечують вищій рівень безпеки SSP модулів, що дає можливість організаціям вибирати альтернативні криптографічні рішення для задоволення вимог безпеки, де увага до модульного середовища має вирішальне значення для забезпечення загальної безпеки.

Рівень безпеки 2 посилює фізичні механізми безпеки рівня безпеки 1, додавши вимогу наявності доказів спроб злому, що охоплює використання відповідного покриття або печаток чи стійких замків до знімних кришок або дверей.

Покриття для виявлення спроб злому або печатки розташовані на модулі, потребують їх пошкодження для отримання фізичного доступу до загальних послуг модуля. Пломби або стійкі замки розміщені на кришках або дверях для захисту від несанкціонованого фізичного доступу.

Рівень безпеки 2 потребує аутентифікації, що базується на ролі, якій криптографічний модуль довіряє виконувати відповідну множину послуг.

Рівень безпеки 2 дає можливість програмному криптографічному модулю працювати в модифікованому середовищі, яке імплементує контроль доступу на основі ролей, або, як мінімум, вибірковий контроль доступу з надійним механізмом визначення нових груп і призначення обмежених прав доступу через списки контролю доступу (наприклад, ACL), та можливість присвоювати кожному користувачеві більше ніж однієї групи, яка захищає від несанкціонованого виконання, модифікації і читання криптографічного програмного забезпечення.

Рівень безпеки 3: На додаток до механізмів фізичної безпеки, які надають докази спроб злому, потрібних у рівні безпеки 2, рівень безпеки 3 забезпечує додаткові вимоги для пом’якшення наслідків несанкціонованого доступу до SSP криптографічного модуля. Фізичні механізми безпеки, потрібні на рівні безпеки 3, призначені для виявлення та реагування на спроби прямого фізичного доступу, використання або модифікації криптографічного модуля і зондування через вентиляційні отвори або прорізи. Механізми фізичної безпеки можуть охоплювати використання сильних корпусів і схем виявлення/протидії злому, які анулюють CSP у разі зняття кришки/дверей криптографічного модуля.

Рівень безпеки 3 потребує, щоб механізми аутентифікації, які ґрунтуються на ідентифікації особи, посилювали безпеку механізмів аутентифікації на основі ролей, визначених для рівня безпеки 2. Криптографічний модуль аутентифікує особистість оператора і верифікує, що ідентифікований оператор має визначену роль і уповноважений виконувати відповідну множину послуг.

Рівень безпеки 3 потребує шифрування, використання надійного каналу або використання процедур поділу знань для введення чи виводу простого CSP.

Рівень безпеки 3 також захищає криптографічний модуль від компрометації безпеки, через вимоги до середовища, до нормальних робочих діапазонів напруги і температури. Умисні дії за межами нормальних діапазонів можуть бути використані порушником, щоб перешкодити захисту криптографічного модуля. Криптографічний модуль має містити спеціальні функції захисту середовища, призначені для виявлення

недопустимих значень напруги і температури, які анулюють CSP, або виконувати ретельне тестування середовища для забезпечення впевненості у надійності модуля, який працює за межами нормального робочого діапазону.

Модулі рівня безпеки 3 потребують додаткових гарантій життєвого циклу, таких як автоматизоване управління конфігурацією, детальне проектування, тестування низького рівня і аутентифікація оператора із використанням інформації аутентифікації постачальника.

Рівень безпеки 4 забезпечує найвищій рівень безпеки, визначений у цьому стандарті. Цей рівень охоплює всі необхідні функції безпеки на більш низьких рівнях, а також розширені можливості.

На рівні безпеки 4 фізичні механізми безпеки забезпечують повний всеохоплюючий захист криптографічного модуля з метою виявлення та реагування на всі спроби несанкціонованого фізичного доступу до SSP, які містяться в модулі, із врахуванням чи без застосовної зовнішньої сили.

Проникнення у корпус криптографічного модуля має дуже високу ймовірність виявлення, що призводить до миттєвого анулювання всіх незахищених SSP. Криптографічні модулі із рівнем безпеки 4 доцільні для роботи у фізично незахищеному середовищі.

Рівень безпеки 4 вводить вимогу багатофакторної аутентифікації оператора. Як мінімум, для цього потрібно два з трьох атрибутів:

— знання, наприклад, секретного пароля;

— володіння, наприклад, фізичним ключем або токеном;

— фізичну властивість, наприклад, біометрику.

Криптографічний модуль із рівнем безпеки 4 має містити спеціальні функції захисту середовища, призначені для виявлення недопустимих значень напруги і температури, які анулюють CSP для забезпечення впевненості у надійності модуля, який працює за межами нормального робочого діапазону.

1.3 Формулювання і постановка наукової задачі дослідження

Виходячи з результатів проведеного аналізу можна стверджувати, що:

питання щодо забезпечення імітостійкості та конфіденційності СОІ нині не розглядається в жодних нормативно-правових актах нашої держави, хоча в умовах ведення гібридної війни саме СОІ зазнаватимуть найбільш руйнівних кібератак, а держава в цілому може понести значні фінансові та матеріальні втрати;

в деяких випадках побудови СОІ, зокрема в разі застосування мереж загального користування та радіоканалів, не існує більш дієвого механізму забезпечення конфіденційності та цілісності каналів передачі команд управління та даних, аніж технології КЗІ.

Виходячи з цього, наукова задача дисертаційної роботи полягає у розробленні теоретичних і прикладних засад побудови та забезпечення методами криптографічної обробки інформації імітостійкості та конфіденційності даних в СОІ з урахуванням множини кіберзагроз та потенційно можливих наслідків їх реалізації. В цей час вона в повному обсязі залишається невирішеною й тому потребує свого розв’язання. Для цього необхідно розробити:

1) метод виявлення атак на програмні реалізації засобів криптографічного захисту інформації в СОІ, впровадження якого шляхом реалізації двоступеневого критерію виявлення аномалій дозволяє своєчасно виявити момент настання певної критичної ситуації та прийняти рішення щодо подальших дій;

2) модель функціонування (криптосхема) шифратора БАЗ (модулю криптографічного захисту інформації) в СОІ, впровадження якої за рахунок методу контролю стану гарантоздатності програмних реалізацій засобів КЗІ та методу генерації потоку підстановок для шифру БАЗ дозволяє забезпечити конфіденційність і цілісність інформації, що циркулює в СОІ, та в умовах кібератак підвищити функціональну безпеку та живучість самої системи;

3) метод генерації потоку підстановок шифру багатоалфавітної заміни для забезпечення в СОІ конфіденційності та цілісності інформації, впровадження якого за рахунок реалізації імітостійкого шифрування на основі оригінального швидкісного алгоритму формування потоку підстановок замін, критерію вибору степеню таких

замін та процедури оцінки якості послідовності підстановок шифру багатоалфавітної заміни дозволяє обрати таку степень підстановок, яка б забезпечувала достатню швидкодію криптоперетворення та була б раціональною для забезпечення захисту повідомлень від підробки;

4) метод оцінки ефективності застосування криптосистем, на базі врахування співвідношення середнього значення максимальних втрат власника СОІ у випадку успішних кібератак на систему захисту до мінімальної вартості реалізації таких атак, що дозволило, на відміну від існуючих, визначити границі для відносної ефективності системи захисту інформації в СОІ в умовах реалізації кіберзагроз.

Структурно-логічна схема проведення дослідження наведена на рис. 1.5.

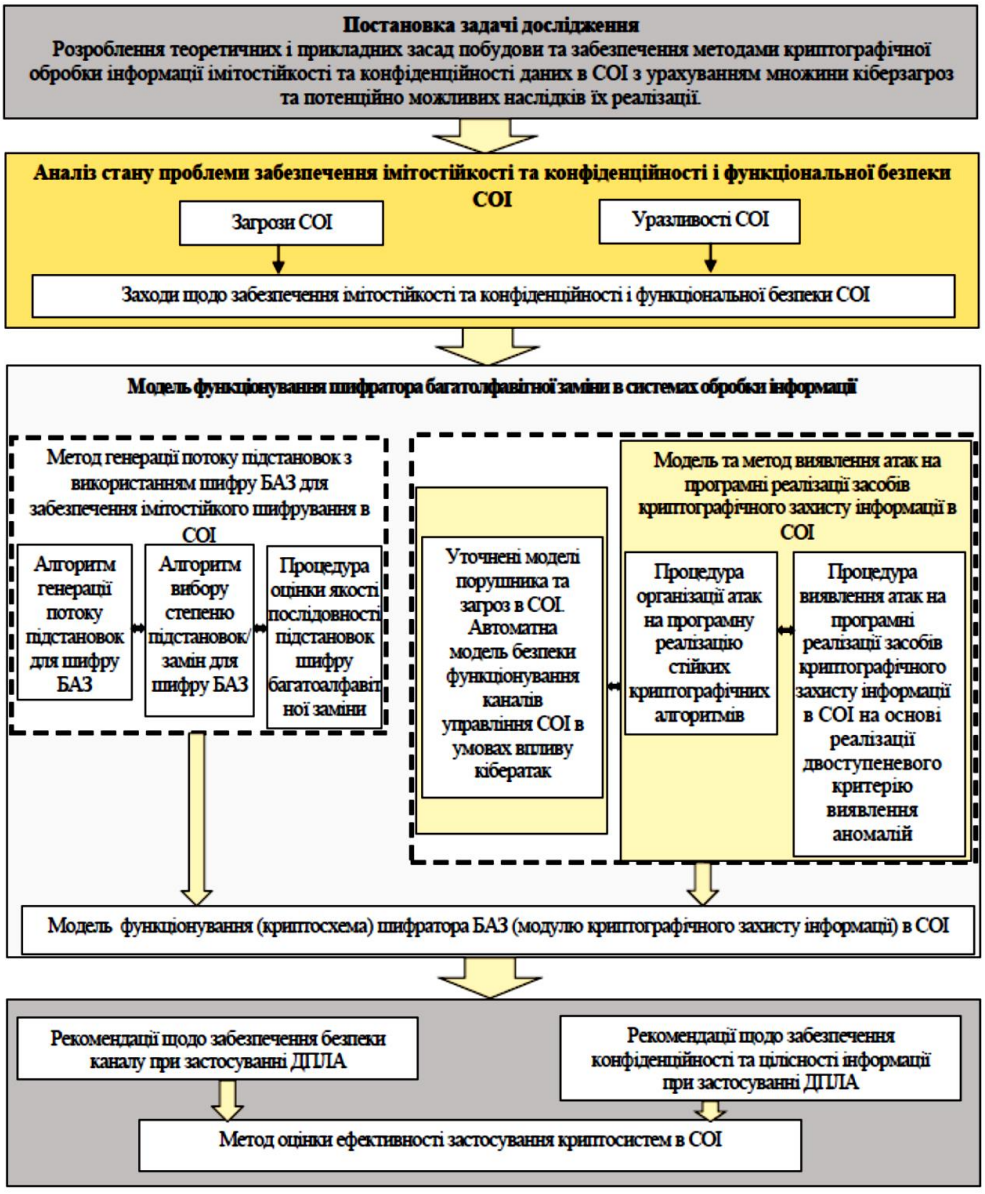


Рис. 1.5. Структурно-логічна схема проведення досліджень

Висновки до першого розділу

На основі аналізу сучасних підходів щодо вирішення завдань гарантоздатності у частині імітостійкості та конфіденційності в розділі констатовано, що універсальної методики, яка б дозволила вирішити ці завдання на практиці не існує.

Зважаючи на те, що прийняття адекватних і обґрунтованих заходів із забезпечення імітостійкості та конфіденційності даних в СОІ не повинно суттєво впливати на виконання нею функціональних завдань за призначенням та суттєво уповільнювати її швидкодію, їх реалізація повинна узгоджуватися із загальною архітектурою системи, а їх працездатність має належним чином контролюватися.

Враховуючи таке, за результатами виконання першого розділу було обрано напрям дослідження, що ґрунтується на необхідності вдосконалення побудови програмних засобів імітостійкого шифрування та розробці нових, ефективних методів виявлення атак на програмні реалізації системи безпеки СОІ.

Список джерел, використаних у першому розділі

1. Про основні засади забезпечення кібербезпеки України: Закон України / Відомості Верховної Ради, 2017, № 45, ст.403.

2. Гулак Г.М., Бурячок В.Л., Складаний П.М,. Кузьменко Л.В. Криптовірологія: загрози безпеки гарантоздатним інформаційним системам і заходи протидії шифрувальним вірусам. *Кібербезпека: освіта, наука, техніка*. Том 2. №10 (2020). С. 6 – 28.

3. Бурячок В.Л., Бурячок Л.В., Костюк Т.Я. Обґрунтування вибору раціональної системи електронного документообігу для державних структур спеціального призначення. *Вісник воєнної розвідки*. 2011. № 24. С. 67–74.

4. Невойт Я. В. Метод оцінювання стану захищеності інформаційних ресурсів на основі дослідження джерел загроз інформаційній безпеці : автореф. дис. … канд. техн. наук : 21.05.01. Київ, 2016. 170 с.

5. Opirskyy Ivan, Ivanchenko Ihor, Platonenko Artem, Skladannyi Pavlo, Roshchuk Mariia Use of near field communication technology for automated profile replication/. // Wydawnictwo Naukowe Akademii Techniczno-Humanistycznej w Bielsku-Białej, Bielsko-Biala, Poland. p.153–161.

6. Харченко В.С. Гарантоспособность и гарантоспособные системы: элементы методологии. *Радіоелектронні і комп'ютерні системи*. 2006. № 5. С. 7-19.

7. Гулак Г.М., Складанний П.М. Формування вимог щодо забезпечення гарантоздатності автоматизованих систем переробки інформації й управління критично-важливими об'єктами інфраструктури. / II Всеукраїнська науково-практична конференція «Кібербезпека в Україні: правові та організаційні питання» (Одесса, 17 листопада 2017р.). Одесса: ОДУВС, 2017. С.12-14.

8. Бурячок В.Л., Гулак Г.М., Хорошко В.О. Завдання, форми та способи ведення воєн у кібернетичному просторі. *Наука і оборона*. 2011. № 3. С. 35–42.

9. Медведовский И.Д., Семьянов П.В., Платонов В.В. Атака через Internet. Содержание. URL :http://citforum.vision.am/internet/attack/toc.shtml.

10. Ільяшов О.А., Бурячок В.Л. До питання захисту інформаційно-телекомунікаційної сфери від стороннього кібернетичного впливу. Наука і оборона. 2010. № 4. С. 35–40.

11. Складанний П.М. Визначення критеріїв безпеки криптосистем. / Складанний П.М. // Міжнародна науково-технічна конференція студентства та молоді «Світ телекомунікацій та інформатизації». – Київ: ДУТ, 2015. – С.14-15.

12. Пермяков О.Ю., Вернер І.Є. Інформаційне протиборство: реалії і тенденції. Арсенал XXI століття. 2002. №2. С. 17 -20.

13. A Solution-based Examination of Local, State, and National Government Groups Combating Terrorism and Cyberterrorism.By: Matusitz, Jonathan; Breen, GeraldMark. Journal of Human Behavior in the Social Environment, Feb2011, Vol. 21 Issue 2, p109-129, 21p. URL : http://search.ebscohost.com/login.aspx?direct=true&db=aph&AN.

14. Гулак Г.М., Козачок В. А., Складанний П. М. та ін. Системи захисту персональних даних в сучасних інформаційно-телекомунікаційних системах. *Сучасний захист інформації*. 2017. № 2. С. 65–71.

15. Roy Y. V., Mazur N. P., Skladannyi P. M. Audit of Information Security is the basis of Effective Protection of the Enterprise. Cybersecurity: Education, Science, Technique. 2018. No. 1. P. 86—93. https://doi.org/10.28925/2663-4023.2018.1.8693

16. Харченко В.С. Гарантоздатні системи та багатоверсійні обчислення: аспекти еволюції. *Радіоелектронні і комп'ютерні системи*, 2009, № 7 (41). С. 46-59. URL: http://nti.khai.edu:57772/csp/nauchportal/Arhiv/REKS/-2009/RE-KS709/Harch.pdf

17. Лисенко, Сергій & Харченко, Вячеслав & Бобровнікова, Кіра & Щука, Роман. Резильєнтність комп'ютерних систем в умовах кіберзагроз: таксономія та онтологія. *Radioelectronic and computer systems*. 2020. С. 17-28. DOI: 10.32620/reks.2020.1.02.

18. НД ТЗІ 1.1-002-99 Загальні положення щодо захисту інформації в комп'ютерних системах від несанкціонованого доступу. Затверджено наказом ДСТСЗІ СБ України від 28.04.1999 № 22

19. Ленков С.В., Перегудов Д.А., Хорошко В.А. Методы и средства защиты информации. Киев: Арий, 2008. Том II. Информационная безопасность, 344 с.

20. Положення про порядок розроблення, виробництва та експлуатації засобів криптографічного захисту інформації. Наказ Адміністрації Державної служби спеціального зв'язку та захисту інформації України від 20.07.2007 № 141, із змінами, внесеними згідно з Наказами Адміністрації ДССЗЗІ від 04.12.2009 № 254, від 02.03.2012 № 90, від 14.12.2015 № 767.

21. Про Державну службу спеціального зв'язку та захисту інформації України: Закон України № 3475-IV від 23 лютого 2006 року. URL: https://zakon.rada.gov.ua/laws/show/3475-15#Text (дата звернення 11.05.2020)

22. Про інформацію: Закон України № 2657-XII від 2 жовтня 1992 року. URL: https://zakon.rada.gov.ua/laws/show/2657-12#Text (дата звернення 11.05.2020)

23. Про електронний цифровий підпис: Закон України № № 852-IV від 22 травня 2003 року. URL: https://zakon.rada.gov.ua/laws/show/852-15#Text (дата звернення 11.05.2020)

24. Про технічні регламенти та процедури оцінки відповідності: Закон України № № 124-VIII від 15 січня 2015 року. URL: https://zakon.rada.gov.ua/laws/show/124-19#Text (дата звернення 11.05.2020)

25. Шабуров А.С. О разработке модели обнаружения компьютерных атак на объекты критической информационной инфраструктуры. *Вестник ПНИПУ. Электротехника, информационные технологии, системы управления*. 2018. №26. URL: https://cyberleninka.ru/article/n/o-razrabotke-modeli-obnaruzheniya-kompyuternyh-atak-na-obekty-kriticheskoy-informatsionnoy-infrastruktury (дата обращения: 25.02.2021).

26. Roy Y. V., Mazur N. P., Skladannyi P. M. Audit of Information Security is the basis of Effective Protection of the Enterprise. *Cybersecurity: Education, Science, Technique.* 2018. No. 1. P. 86-93. https://doi.org/10.28925/2663-4023.2018.1.8693

27. Складанний П.М. Аналіз типів атак на державні інформаційні ресурси. / Міжнародна науково-технічна конференція "Сучасні інформаційно-телекомунікаційні технології". Київ: ДУТ, 2014. С.20.

28. Карачка А.Ф. Технології захисту інформації : текст лекцій. Тернопіль: ТНЕУ, 2017. URL: http://dspace.wunu.edu.ua/bitstream/316497/26564/1/lekzii.pdf

29. Іваненко Д. В. Методи протидії атакам спеціального виду на схеми направленого шифрування у кільцях зрізаних поліномів. *Радиотехника*. 2012. Вип. 171. С. 90-98. URL: http://nbuv.gov.ua/UJRN/rvmnts_2012_171_13.

30. Дудикевич В.Б., Томашевський Б.В., Сергієнко Р.В. Протоколи і механізми безпеки інформації в комп'ютерних системах і мережах. Захист інформації. 2009. № 2. С. 39-54. URL: http://jrnl.nau.edu.ua/index.php/ZI/article/download/4043/4191

31. Guide to Industrial Control Systems (ICS) Security: NIST Special Publication 800-82. – Recommendations of the National Institute of Standards and Technology.

32. Гулак Г.М., Мухачов В.А., Хорошко В.О., Яремчук Ю.Є. Основи криптографічного захисту інформації: підручник. Вінниця: ВНТУ, 2011. 198 с.

33. Довгань О.Д., Гулак Г.М., Грінь А.К., Мельник С.В. Методологія захисту інформації: навч.-метод. посіб. Київ: Наук.-вид. центр НА СБ України, 2012. 186 с.

34. Ільєнко А. В., Тригуб Д. А. Практичні підходи використання криптографічних алгоритмів симетричного та асиметричного шифрування для забезпечення захисту електронних платіжних систем. – MATERIÁLY XIV MEZINÁRODNÍ VĚDECKO-PRAKTICKÁ KONFERENCE AKTUÁLNÍ VYMOŽENOSTI VĚDY, 2018.

35. Тарнавський Ю.А. Технології захисту інформації: підручник. Київ: КПІ ім. Ігоря Сікорського, 2018. URL: https://ela.kpi.ua/bitstream/-123456789/23896/1/TZI_book.pdf

36. Диффи У., Хеллмэн М. Э. Защищенность и имитостойкость. Введение в криптографию. *ТИИЭР*, 1979. Т. 67, № 3. С. 71-109.

37. Алферов А.П., Зубов А.Ю., Кузьмин А.С., Черемушкин А.В. Основы криптографии : Учебное пособие. Москва : Гелиос АРВ, 2002. 480 с.

38. Бабаш А.В., Шанкин Г.П. Криптография / Под ред/ В.П. Шерстюка, Э.А. Применко. Москва : СОЛОН-Р, 2002. 512 с.

39. Гулак Г., Ковальчук Л. Різні підходи до визначення випадкових послідовностей / Науково-технічний збірник «Правове, нормативне та метрологічне забезпечення системи захисту інформації в Україні». Вип. 3. Київ, 2001. С.127-133.

40. Кулажський В.І., Берестов Д.С., Кульчицький О.С., Тернавський І.О. Вибір засобів криптографічного захисту інформації для захисту ERP-системи від несанкціонованого доступу до її інформаційних ресурсів. Збірник наукових праць Центру воєнно-стратегічних досліджень Національного університету оборони України імені Івана Черняховського. 2014. №2 (11). С. 57-60. URL: http://www.irbis-nbuv.gov.ua/cgi-bin/irbis_nbuv/cgiirbis_64.exe?C21COM=2&I21-DBN=UJRN&P21DBN=UJRN&IMAGE_FILE_DOWNLOAD=1&Image_file_name=PDF/Znpcvsd_2014_2_11.pdf

41. Харин Ю.С., Берник В.И., Матвеев Г.В. Математические основы криптологии. Минск : БГУ, 1999. 319 с.

Розділ 2
МОДЕЛІ І МЕТОДИ ЗАБЕЗПЕЧЕННЯ ІМІТОСТІЙКОСТІ ТА КОНФІДЕНЦІЙНОСТІ В СИСТЕМАХ ОБРОБКИ ІНФОРМАЦІЇ

Не зважаючи на існуючі переваги імітостійкого шифрування на основі шифру багатоалфавітної заміни для забезпечення в СОІ імітостійкості та конфіденційності інформації, зазначений метод як й інші подібні методи, що були розглянуті в розділі 1, має ряд недоліків, які суттєво обмежують можливості його застосування. Основним із таких недоліків є низька швидкодія процедури генерації послідовностей підстановок. Для усунення цього недоліку доцільно скористатися ідеєю формування підстановок з послідовності випадкових чисел [1], а саме з симетричної групи підстановок S_n.

2.1. Шляхи забезпечення імітостійкості криптозахисту інформації в СОІ

Як було зазначено раніше, суть методу безповторного набору підстановок із випадкової (або псевдовипадкової) рівномірно розподіленої послідовності (ВРРП) полягає у поступовому формуванні нижнього рядку підстановки [2; 3]. Для цього використовується потік випадкових чисел $j_0, j_1, \dots, j_m, \dots \in \mathbb{Z}_n$ таким чином, що символ, який присутній у вже сформованій частині рядка, відхиляється та у подальшому не використовується. Таким чином процес може продовжуватися досить довго. Оскільки результатом вибірки з ВРРП є також ВРРП, то тільки частина векторів $\bar{J} = (j_0, j_1, \dots, j_{n-1})$ без корегування може утворювати нижній рядок підстановки внаслідок наявності повторювань в $\bar{J}$, тому ймовірність π_n отримати підстановку за методом безповторного набору за n кроків одразу без додаткового коригування дорівнює:

$$\pi_n = P\left(\begin{pmatrix} 0 & \dots & n-1 \\ j_0 & \dots & j_{n-1} \end{pmatrix} \in S_n\right) = \frac{|S_n|}{n^n} = \frac{n!}{n^n}. \qquad (2.1)$$

Для оцінки ймовірності π_n у рівнянні (2.1) можливо скористатися формулою Стірлінга для факторіалу [4]:

$$n! = \sqrt{2\pi n} \cdot n^n \cdot e^{-n}\big(1 + o(1)\big), \text{ де } o(1) \to 0 \text{ при } n \to \infty.$$

Звідки отримуємо оцінку для вказаної ймовірності:

$$\pi_n = \frac{\sqrt{2\pi n}\cdot n^n \cdot e^{-n}}{n^n} = \sqrt{2\pi n}\cdot e^{-n}. \quad (2.2)$$

Оскільки обсяг вихідних випадкових даних, що необхідні для формування однієї підстановки асимптотично оцінюється величиною $C \cdot n \cdot \ln n$, де C – деяка константа, практичне застосування методу безповторного набору для побудови шифрів багатоалфавітної заміни (БАЗ) для швидкісних систем управління постає проблематичним.

2.1.1. Метод генерації потоку підстановок з використанням шифру БАЗ для забезпечення імітостійкого шифрування в СОІ

2.1.1.1. Швидкісний алгоритм генерації потоку підстановок для шифру багатоалфавітної заміни

З урахуванням якостей та недоліків методу безповторного набору підстановок із випадкової (або псевдовипадкової) рівномірно розподіленої послідовності постає актуальне завдання щодо розробки швидкісного *алгоритму генерації потоку підстановок заміни для шифру багатоалфавітної заміни*, застосування якого забезпечуватиме конфіденційність та імітостійкість інформаційного обміну в СОІ. Передамбулою розробки алгоритму є дві леми.

Лема 1. Якщо найбільший спільний дільник чисел $n, l \in \mathbb{N}, n > 1$: $(n, l) = 1$, то для фіксованого $\forall \delta \in \mathbb{Z}_n$ у рівнянні

$$\delta + k \cdot l = x_k \bmod n, \quad (2.3)$$

для різних $k \in \mathbb{Z}_n$ усі значення $x_k \in \mathbb{Z}_n$ різні.

Дійсно, нехай $k_1 \neq k_2$, але $x_1 = x_2$. Тоді з рівняння (2.3) маємо:

$$\delta + k_1 \cdot l = \delta + k_2 \cdot l \bmod n,$$

або:

$$(k_1 - k_2) \cdot l = 0 \bmod n.$$

Останнє суперечить вимозі взаємної простоти чисел $(n, l) = 1$. З цього слідує $\forall k_1 \neq k_2 \implies x_1 \neq x_2$, що потрібно було довести.

Головною особливістю *алгоритму генерації потоку підстановок заміни для шифру багатоалфавітної заміни*, виходячи з леми 1, є: випадковість вибору початкової позиції для заповнення нижнього рядка у підстановці, що забезпечує, як буде показано далі, вирівнювання статистичних характеристик шифрованого повідомлення, а також коректність роботи методу в сенсі генерації підстановок. При цьому важливими характеристиками схеми генерації підстановок (рис.2.1) з точки зору забезпечення необхідних криптографічних якостей шифру багатоалфавітної заміни є:

кількість різних підстановок, що генеруються;

ймовірності їх зустрічаємості;

матриця перехідних ймовірностей.

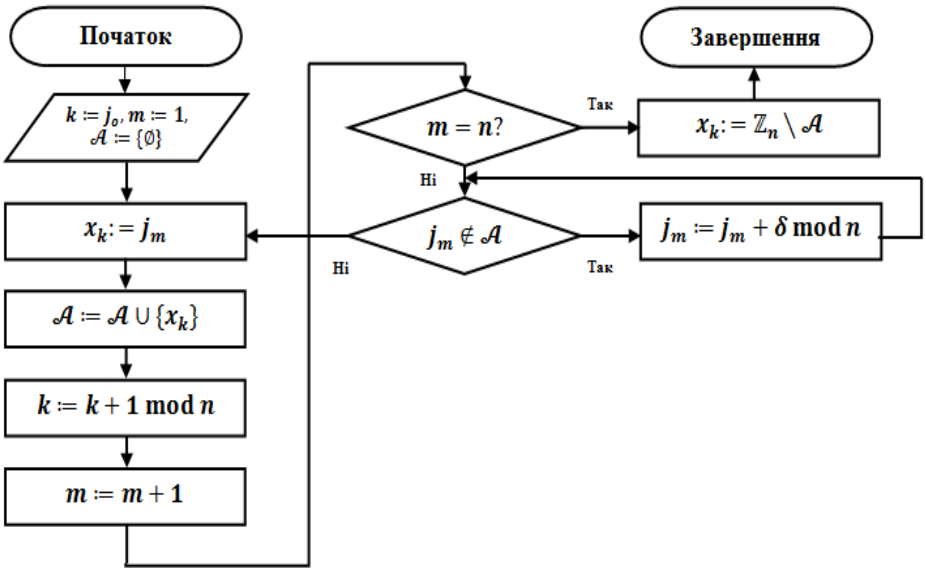


Рис. 2.1 Блок-схема алгоритму генерації потоку підстановок

Сформулюємо зміст розробленого алгоритму генерації потоку підстановки степені n: $X = \begin{pmatrix} 0 & \dots & n-1 \\ x_0 & \dots & x_{n-1} \end{pmatrix}$ за допомогою послідовності випадкових чисел $j_0, j_1, \dots, j_m, \dots \in \mathbb{Z}_n$. Для формального опису алгоритму генерації скористаємось оператором Бекуса присвоювання значення деякої змінної «:=». На кожному кроці алгоритму визначається один перехід у підстановці $\begin{pmatrix} k \\ x_k \end{pmatrix}$ та загальна множина переходів $\mathcal{A}$, що сформовані після виконання чергового кроку.

Тоді формалізований опис послідовність кроків виконання алгоритму генерації потоку підстановок у випадку $(l = 1, \delta)$ матиме вигляд, наведений в табл.2.1.

Таблиця 2.1

Формалізований опис алгоритму генерації потоку підстановок $(l = 1, \delta)$

Крок	**Формалізований опис**	**Коментар**
0.	$\mathrm{t} \coloneqq 0.$	t - початок поточного вектору РРВП.
1.	$\mathrm{k} \coloneqq \mathrm{j_t}, \mathrm{m} \coloneqq \mathrm{t}+1,\ \mathcal{A} \coloneqq \{\emptyset\}.$	k - номер переходу; m - номер кроку підстановки.
2.	$\mathrm{x_k} \coloneqq \mathrm{j_m}.$	Визначення значення для нижнього рядка $\binom{\mathrm{k}}{\mathrm{x_k}}$.
3.	$\mathcal{A} \coloneqq \mathcal{A} \cup \{\mathrm{x_k}\}.$	Доповнюємо множину вже визначених переходів.
4.	$\mathrm{k} \coloneqq \mathrm{k}+1 \bmod \mathrm{n}.$	Збільшуємо номер позиції у верхньому рядку для формування наступного переходу.
5.	$\mathrm{m} \coloneqq \mathrm{m}+1.$	Номер чергового випадкового числа.
6.	Якщо $\mathrm{m} = \mathrm{n}-1$, то виконуємо крок 10, якщо НІ – виконуємо крок 7.	Умовний перехід на завершення підстановки.
7.	Якщо має місце $\mathrm{j_m} \notin \mathcal{A}$ - виконуємо крок 2, якщо НІ – виконуємо наступний крок 8.	Умовний перехід до способу обчислення наступного значення для нижнього рядка.
8.	$\mathrm{j_m} \coloneqq \mathrm{j_m}+\delta \bmod \mathrm{n}.$	Модифікація випадкового числа.
9.	Далі виконуємо крок 7	Перехід до перевірки у полі визначених переходів.
10.	Останньому переходу присвоюємо значення $\mathrm{x_{n-1}} \coloneqq \mathbb{Z}_\mathrm{n} \setminus \mathcal{A}.$	Завершення формування підстановки;.
11.	Чи існує ще нешифрований символ? Якщо ТАК – виконуємо крок 12, якщо НІ – завершуємо	Перевірка вхідного потоку відкритих символів
12.	$\mathrm{t} \coloneqq \mathrm{t}+\mathrm{n}.$	Початок чергового вектору.
13.	Перехід до кроку 1.	Перехід до початку формування нової підстановки.

Лема 2. Якщо послідовність випадкових чисел $j_0, j_1, \dots, j_{n-1} \in \mathbb{Z}_n$ є незалежною у сукупності та має рівномірний розподіл, тоді алгоритм генерації підстановок забезпечує формування S_n - симетричної групи підстановок з n [5]. Це пояснюється тим, що ймовірність появи будь-якої послідовності $j_0, j_1, \dots, лі j_{n-1} \in \mathbb{Z}_n$, у тому

числі такої, що утворює нижній рядок підстановки без коригування послідовності визначається із залежності:

$$P(j_0, j_1, \dots, j_{n-1} \in \mathbb{Z}_n) = n^{-n} \neq 0.$$

Зауважимо, що порівняно з методом безповторного набору запропонований алгоритм: по-перше, завжди виконується за фіксовану кількість кроків й, по-друге, він є більш швидким в середньому на:

$$S \approx \ln n \times 100\%. \quad (2.4)$$

Таким чином вибір величини n - розміру підстановки БАЗ повинен здійснюватися, виходячи з критерію забезпечення достатньої швидкодії. Крім того, на вибір вказаної величини, зважаючи, що у разі застосування БАЗ стійкість криптографічного перетворення зберігається у випадку навіть $N-$кратного повтору ключа [6]

$$N \leq {}^{n}/_{2}. \quad (2.5)$$

може впливати необхідність забезпечення достатнього запасу криптостійкості.

2.1.1.2 Алгоритм вибору степеня підстановок/замін шифру БАЗ для забезпечення захисту повідомлень від підробки

Відомо, що криптографічно нестійкий шифр не може забезпечити імітостійкість. Наприклад, шифр простої заміни не є криптографічно стійким при достатній довжині відкритого повідомлення, а шифр гамування втрачає свою стійкість в разі випадкового повторення ключу, що породжує загрозу підробки інших повідомлень. З іншого боку імітостійкість шифру не може однозначно свідчити про його криптографічну стійкість. В роботі [7] наведено деякі оцінки імітостійкості шифрів, зокрема доведено, що для будь-якого шифру для ймовірності $p_{\text{п}}$ підміни повідомлення в каналі зв'язку справедлива нерівність:

$$p_{\text{п}} \geq \frac{|\mathcal{M}|-1}{|\mathcal{C}|-1}, \quad (2.6)$$

де $|\mathcal{M}|$ - потужність множини припустимих відкритих повідомлень, $|\mathcal{C}|$ - потужність множини шифрованих повідомлень.

Зважаючи на таке для підвищення криптографічної стійкості шифру БАЗ необхідно обрати таку кількість підстановок, яка була б раціональною для забезпечення захисту повідомлень від підробки. Посилити вимогу щодо криптографічної стійкості шифру БАЗ (2.6) можливо за рахунок використання бітової ентропії технологічної (незашифрованої) інформації.

$$H_0 = -p_0 \log_2 p_0 - p_1 \log_2 p_1. \tag{2.7}$$

Для цього доведемо наступну лему.

Лема 3. У випадку практичної криптографічної стійкості шифру та джерела повідомлень без пам'яті для ймовірності $p_{\text{п}}$ підміни в каналі зв'язку шифрованого повідомлення довжини L справедлива оцінка знизу:

$$p_{\text{п}} \geq 2^{-(1-H_0)L}. \tag{2.8}$$

Слід зазначити, що для інформаційного обміну в рамках СОІ зазвичай характерний високий ступінь формалізації повідомлень, які переважно мають деяку фіксовану довжину, тому обмеження щодо довжини можна вважати припустимим. При цьому як теоретично, так і практично стійкі шифри забезпечують розподіл ймовірностей знаків шифрованого тексту, який не відрізняється від випадкового та рівномірного. Тому потужність множини шифрованих текстів $|\mathcal{C}|$ дорівнює величині 2^L. Оскільки для джерела повідомлень виконані умови другої теореми Шеннона [8], то для $|\mathcal{M}|$ потужності множини припустимих відкритих повідомлень справедлива оцінка:

$$|\mathcal{M}| = 2^{H_0 L}.$$

Далі скористаємось нерівністю (2.6) і отримаємо $p_{\text{п}} \geq \frac{|\mathcal{M}|-1}{|\mathcal{C}|-1} = \frac{2^{H_0 L}-1}{2^L-1} \approx 2^{-(1-H_0)L}$, для достатньо великих L, що й було потрібно довести.

Доцільно звернути увагу, що, виходячи з нерівності (2.7), нескладно отримати наступну нерівність:

$$L \geq \frac{\log_2 {}^1/_{p_{\text{п}}}}{1-H_0}, \tag{2.9}$$

яку можливо трактувати наступним чином. Для заданої імовірності підміни $p_{\text{п}}$ довжина технологічної інформації L не може бути меншою ніж та, що визначена нерівністю (2.9).

Для різних природних мов величина ентропії приймає значення в діапазоні $0{,}2 \leq H_0 \leq 0.4$, для технологічної інформації такі довідкові дані відсутні, але відповідну оцінку можливо отримати з рівняння (2.7). В таблиці 2.2 та на рис. 2.2 приведені розрахункові дані та сімейство графіків функції $p = 2^{-(1-H_0)L}$ для випадків значень бітової ентропії $H_0 = 0{,}1 \ldots 0{,}6$.

Таблиця 2.2

Нижня оцінка ймовірності підробки
для практично стійкого шифру

L	Ентропія на один знак H0				
	0.1	0.3	0.4	0.5	0.6
1	0,5359	0,6156	0,6598	0,7071	0,7579
2	0,2872	0,3789	0,4353	0,5000	0,5743
3	0,1539	0,2333	0,2872	0,3536	0,4353
4	0,0825	0,1436	0,1895	0,2500	0,3299
5	0,0442	0,0884	0,1250	0,1768	0,2500
6	0,0237	0,0544	0,0825	0,1250	0,1895
7	0,0127	0,0335	0,0544	0,0884	0,1436
8	0,0068	0,0206	0,0359	0,0625	0,1088
9	0,0036	0,0127	0,0237	0,0442	0,0825
10	0,0020	0,0078	0,0156	0,0313	0,0625
11	0,0010	0,0048	0,0103	0,0221	0,0474
12	0,0006	0,0030	0,0068	0,0156	0,0359

З таблиці та графіків можливо з’ясувати, що для малих довжин команд (повідомлень), коли $L \leq 5$ біт ймовірність підміни в каналі зв’язку для більшості значень має досить велике значення $0{,}044 \leq p_{\text{п}} \leq 0{,}25$.

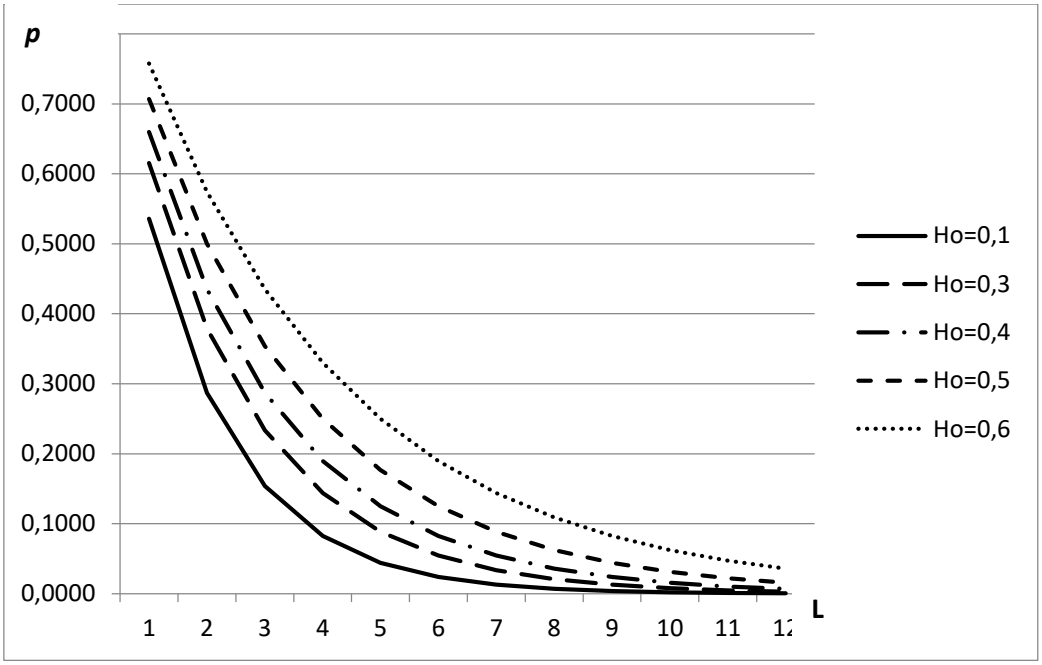


Рис. 2.2 - Сімейство графіків функції $p_n = 2^{-(1-H_0)L}$ $(H_0 = 0.1,0.3,0.5,0.6,0.7)$.

Водночас, у випадку $L \geq 12$ біт ймовірність підміни в каналі зв'язку для більшості значень має більш придатні для практичного застосування значення $0{,}0006 \leq p_{\text{п}} \leq 0{,}0359$. Тобто у найгіршому випадку ймовірність підробки не перевищила 5%. Разом з тим, слід звернути увагу, що разі наближення ентропії повідомлень до одиниці $H_0 \to 1$ права частина нерівності (2.8) також наближається до одиниці, а це означає зростання ймовірності підміни $p_{\text{п}} \to 1$. У цьому випадку, для розв'язку проблеми пропонується ввести у відкриті повідомлення надлишковість шляхом формування його контрольної суми (наприклад, за алгоритмом CRC32), яка також підлягатиме за шифруванню. Застосування у цьому випадку шифру багатоалфавітної заміни забезпечить як завгодно малу величину ймовірності підробки.

Знання конкретного виду криптографічного стійкого шифру дає змогу більш точно оцінити ймовірність цільового нав'язування. Зокрема, у випадку шифру багатоалфавітної заміни його імітостійкість можливо характеризувати за допомогою складності $C_{\text{п}}$ цільового нав'язування порушником повідомлення, яку можна оцінити наступним чином.

Лема 4. У випадку практично стійкого шифру багатоалфавітної заміни з підстановками степеню n для ймовірності $q_{r\text{п}}$ підміни в каналі зв'язку істинного

повідомлення $M = (m_1, m_2, \dots, m_l)$ на фіктивне $\widetilde{M} = (\widetilde{m}_1, \widetilde{m}_2, \dots, \widetilde{m}_l)$ таке, що M та $\widetilde{M}$ відрізняються лише на r місцях, справедлива оцінка:

$$q_{r\text{п}} = (n-1)^{-r}. \tag{2.10}$$

Доведемо це. Нехай, за допомогою шифру багатоалфавітної заміни з ключем K легальним користувачем системи було зашифроване та передане в канал зв'язку деяке істинне повідомлення $M = (m_1, m_2, \dots, m_l)$ із множини припустимих відкритих повідомлень: $M \in \mathcal{M}$:

$$C = E_K(M).$$

Порушник, який перехопив шифротекст $C = (c_1, c_2, \dots, c_l)$, намагається провести атаку з відомим відкритим повідомленням для нав'язування фіктивного повідомлення $\widetilde{M} = (\widetilde{m}_1, \widetilde{m}_2, \dots, \widetilde{m}_l)$. Вважаємо, що дані M та $\widetilde{M}$ відрізняються лише на r місцях, тобто маємо:

$$|\{i: m_i \neq \widetilde{m}_i\}| = r$$

Тоді порушник на $(l-r)$ місцях співпадаючих знаків $m_i = \widetilde{m}_i$ використовує відомі йому символи шифротексту, оскільки:

$$E_K(m_i) = E_K(\widetilde{m}_i) = \begin{pmatrix} \dots & m_i & \dots \\ & c_i & \end{pmatrix}.$$

Оскільки шифр є практично стійкім, то він відповідає вимозі статистичної безпеки [9]. Це означає, що будь який його шифротекст не можливо відрізнити від випадкової послідовності за допомогою жодного поліноміального алгоритму [10]. Тому на інших місцях порушник буде вимушений обирати один з невідомих $(n-1)$ варіантів переходів навмання, а загальна кількість варіантів підбору шифротексту $\tilde{C}$, який після розшифрування може буде сприйнятий як

$$\widetilde{M} = E_K^{-1}(\tilde{C}),$$

становить величину:

$$C_n = (n-1)^r.$$

Звідси отримуємо ймовірність вгадування порушником r символів з першого разу, яка може слугувати оцінкою для рівня імітостійкості шифру багатоалфавітної заміни:

$$q_{rn} = C_n^{-1} = (n-1)^{-r}.$$

Що і було потрібно довести.

Таким чином, складність задачі підробки повідомлень, які зашифровані за допомогою багатоалфавітної заміни, зростає за експоненціальним законом залежно від кількості символів, що підробляються та, відповідно, зменшується ймовірність цільового нав'язування. Типовим заходом протидії послідовному перебору є обмеження кількості невдалих спроб. Загалом оцінка (2.4), нерівності (2.5, 2.8), а також рівняння (2.10) дозволяють раціонально обрати величину степеня підстановок.

Зважаючи на необхідність врахування вимоги щодо зручності застосування запропонованого алгоритму у вигляді програмної реалізації, значення степеня підстановки доцільно обирати такі, що узгоджуються з форматами даних у мікропроцесорах, а саме: $n = 2^m$, де $m = 2,4,8$ (зауважимо, що для $m = 1$ шифр багатоалфавітної заміни перетворюється на звичайний шифр гамування по $\mathrm{mod}\ 2$).

В таблиці 2.3 наведені характеристики методики генерації послідовності підстановок для випадків $n = 4,16,256$.

Таблиця 2.3

Розрахункові характеристики розробленої методики генерації підстановок

№№	Характеристики методики		Степінь підстановки		
			4	16	256
1.	$q_{1п}$	Імовірність підробки 1 символу	0.333	0.067	0.008
2.	N	Стійкість при повторі ключа	2	8	128
3.	S	Швидкодія порівняно з МБН (%)	138.6	277.2	554.5
4.	V	Обсяг двійкової РРВП для генерації однієї підстановки (біт)	8	64	4096

На підставі аналізу даних таблиці можливо дійти висновку, що характеристики імітостійкості та криптографічної стійкості при повторі ключа є найгіршими для $n = 2^2$, а у випадку $n = 2^8 = 256$ - найкращими. Але останній варіант внаслідок занадто великого обсягу вихідної РРВП призведе до суттєвого уповільнення процесу шифрування порівняно з іншими варіантами. Тому для практичного застосування пропонується обирати степінь підстановок $n = 2^4 = 16$.

2.1.1.3 Процедура перевірки послідовності підстановок шифру багатоалфавітної заміни та оцінки їх якості

Для оцінки якості послідовності підстановок в удосконаленому за рахунок впровадження алгоритмів генерації потоку та вибору степені підстановок, методі генерації потоку підстановок з використанням шифру БАЗ [5; 11] доцільно скористатися матрицею перехідних ймовірностей.

Припустимо, що з урахуванням леми 2, підстановки $X_1, X_2, \dots, X_{n!}$ – це усі підстановки, які можливо отримати за допомогою удосконаленого методу їх генерації на основі ВРРП (табл.2.1):

$$\{X_1, X_2, \dots, X_{n!}\} = S_n.$$

Позначимо $\tilde{X}_i = \left\| x_{kl}^{(i)} \right\|, x_{kl}^{(i)} = \begin{cases} 1, & k = X_i(l) \\ 0, & k \neq X_i(l) \end{cases}, k, l \in \{\overline{1,n}\}$ - двійкова матриця розміру $n \times n$, в кожному рядку і стовпці якої знаходиться рівно один одиничний елемент, що має назву матриця підстановки X_i.

Відмітимо, якщо вектор $\bar{q} = (q_1, \dots, q_n),\ q_j = P(m_i = j),\ j \in \{\overline{1,n}\}$ є вектором розподілу ймовірностей зустрічаємості символів відкритого повідомлення, то у підсумку його зашифрування за допомогою деякої підстановки $X_i \in S_n$, тоді вектор розподілу ймовірностей зустрічаємості символів зашифрованого повідомлення $\bar{Q} = (Q_1, \dots, Q_n)$ може бути обчислений [12] за допомогою наступного рівняння:

$$\bar{Q} = \bar{q} \cdot \tilde{X}_i. \tag{2.11}$$

Рівняння (2.9) легко узагальнюється на випадок використання множини підстановок $\{X_1, X_2, \dots, X_{n!}\}$, які застосовуються з ймовірністю p_{X_i} (варіант додавання шифрів по Шеннону) маємо [12]:

$$\bar{Q} = \bar{q} \cdot \mathcal{P}, \tag{2.12}$$

де матриця $\mathcal{P}$ обчислюється за формулою:

$$\mathcal{P} = \begin{pmatrix} p_{00} & \cdots & p_{0,n-1} \\ \vdots & \ddots & \vdots \\ p_{n-1,0} & \cdots & p_{n-1,n-1} \end{pmatrix} = \sum_{X_i \in S_n} p_{X_i} \cdot \bar{X}_i, \tag{2.13}$$

де $p_{ij} = P(j/i),\ i,j \in Z_n$ - умовна ймовірність появи на виході вузла зашифрування знаку j в разі надходження знаку i; $\bar{X}_i$ – підстановочна матриця, що відповідає генерованій підстановці $X_i \in S_n$, p_{X_i} – ймовірність генерації підстановки X_i.

Матриця $\mathcal{P}$ по суті є матрицею перехідних ймовірностей шифру багатоалфавітної заміни. Якщо матриця $\mathcal{P}$ має вигляд:

$$\mathcal{P} = \begin{pmatrix} 1/n & \cdots & 1/n \\ \vdots & \ddots & \vdots \\ 1/n & \cdots & 1/n \end{pmatrix}, \tag{2.14}$$

то із (2.12) слідує, що $\bar{Q} = (1/n, \ldots, 1/n)$, тобто шифрований текст у цьому випадку має рівномірний розподіл зустрічаємості різних символів. А це є однією з ознак статистичної безпеки шифру.

Таким чином, матриця перехідних ймовірностей, яка визначена формулою (2.13), є однією з основних характеристик удосконаленого методу генерації підстановок. Зауважимо, що його особливістю є випадковий вибір початкової позиції для заповнення нижнього рядка у підстановці. При цьому перший визначений перехід $\binom{j_0}{j_1}$ з ймовірністю $1/n^2$ може приймати будь-яке значення із множини:

$$\mathcal{R} = \left\{\binom{0}{0}, \ldots, \binom{0}{n-1}, \binom{1}{0}, \ldots, \binom{1}{n-1}, \ldots, \binom{n-1}{0}, \ldots, \binom{n-1}{n-1}\right\},$$

На другому кроці починається розгалужений процес, який передбачає, що:

<u>*по-перше*</u>, з ймовірністю $(n-1)\cdot n^{-1}$ будуть отримані переходи $\begin{pmatrix} j_0 & j_0+1 \\ j_1 & j_2 \end{pmatrix}$, якщо випадкове число j_2 не співпадає з раніше отриманим числом j_1;

<u>*по-друге*</u>, з ймовірністю n^{-1} будуть отримані переходи $\begin{pmatrix} j_0 & j_0+1 \\ j_1 & l\cdot j_1+\delta \end{pmatrix}$, якщо випадкове число j_2 співпадає з раніше отриманим числом j_1.

Перший варіант формування нового переходу назвемо не модифікованим, інший модифікованим. Тоді метод формування підстановки можливо подати у вигляді процесу, зображеного на рисунку 2.3 й стверджувати, що розгалуження процесу створення підстановки відбувається з певною ймовірністю після кожного його кроку.

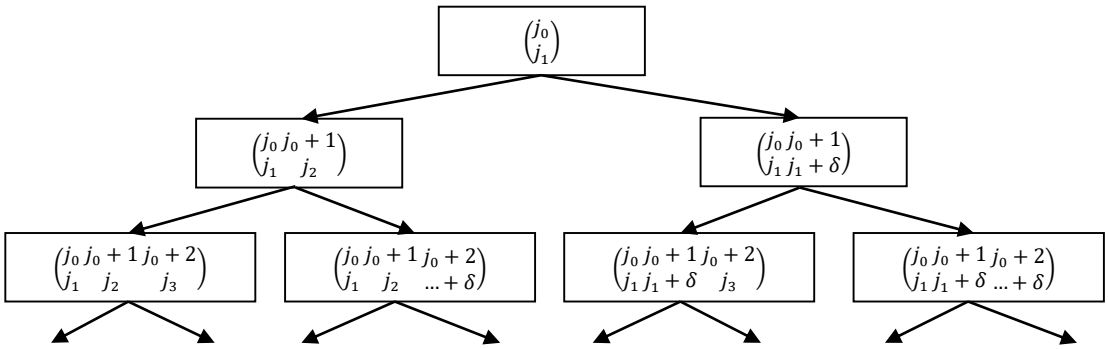


Рис.2.3. Схема процесу створення підстановки

З наведеної схеми нескладно бачити, що існує 2^{n-1} варіантів розгалуження процесу. При цьому деякі варіантів можуть співпадати. Наприклад, після другого кроку можуть співпадати $\binom{j_0\ j_0+1}{j_1\quad j_2}$ та $\binom{j_0\ j_0+1}{j_1\ j_1+\delta}$, якщо $\delta + j_1 = j_2 \bmod n$. Тобто, n^2 різних варіантів першого переходу мають наслідком не більше $n^2 \cdot 2^{n-1} \leq n!$ різних підстановок, але за рахунок симетрії процесу відбувається часткове вирівнювання ймовірностей різних підстановок. А це призводить до наближення матриці перехідних ймовірностей до рівномірної.

У ході статистичного експерименту було досліджено цей факт, результати дослідження викладено у третьому розділі [13].

2.2 Шляхи забезпечення цілісності програмних реалізацій засобів криптографічного захисту інформації в СОІ

2.2.1 Уточнені моделі порушника і загроз в СОІ та автоматна модель безпеки функціонування каналів управління системи

Порушник – це особа, яка робить спроби виконання заборонених операцій (дій) помилково, від незнання або усвідомлено зі злим умислом (можливо, з корисливих мотивів) або без такого (заради гри або задоволення, з метою самоствердження тощо) і використовує для цього різні можливості, методи й засоби. В ІТС, як це показано в роботах [14-15], в залежності від локалізації джерела загрози, порушники

поділяються на зовнішніх та внутрішніх. Категорії осіб, які можуть бути зовнішніми та внутрішніми прушниками, представлено на рис. 2.4.

Інші критерії класифікації порушників в ІТС пов'язані з: а) мотивами порушень, б) рівнем знань з ІТС; в) часом та місцем дії; г) рівнем можливостей порушника [16]. На рис. 2.4 нами узагальнено всі названі вище види класифікації.

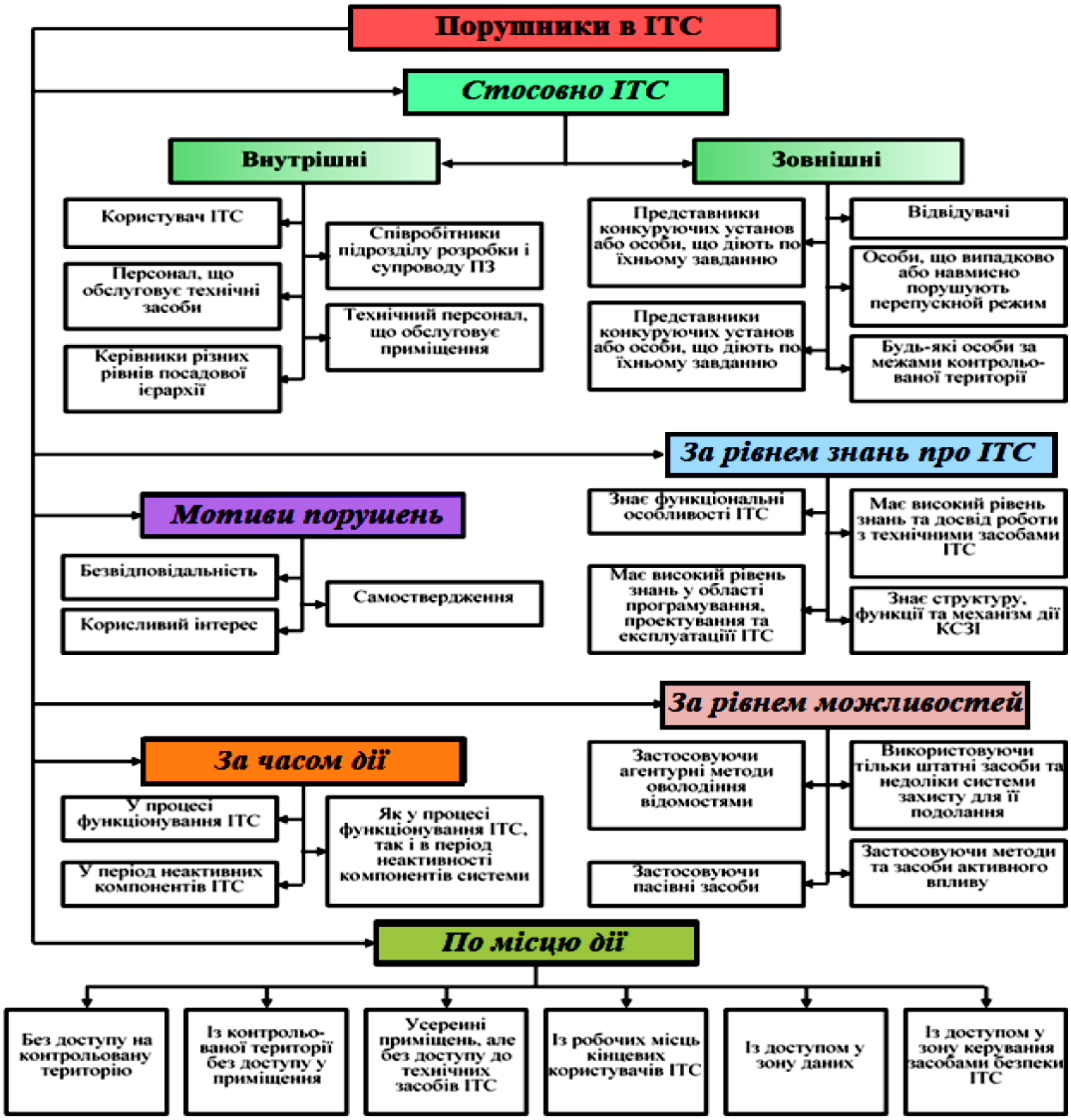


Рис.2.4. Порушники в ІТС (СОІ)

Дуже важливим видається описати профіль порушника з урахуванням всіх наведених вище класифікаційних критеріїв. Наші міркування з цього приводу ми узагальнили в таблиці 2.4.

Як бачимо з таблиці, модель порушника відображає його практичні та потенційні можливості, мотиви, апріорні знання та обладнання, яке він може використовувати.

Таблиця 2.4

Характеристика порушника за типами

Тип порушника	**Характеристика**	**Знання про ТЗО**	**Оснащення**	**Примітки**
Професіонал	Дуже небезпечний. Може добувати інформацію про СОІ, планувати та готувати вторгнення в систему. Немає інтересу до матеріальних ресурсів.	Високі	Спеціальний набір засобів для несанкціонованого доступу	Тренується державою та використовується в її інтересах
Найманець (любитель)	Небезпечний. Має ознаки плановості дій. Може збирати інформацію про СОІ. Цікавить весь спектр цілей вторгнення в систему	Хороші	Набір саморобних або доступних засобів для несанкціонованого доступу	Його послуги вартують дорого. Ціль НСД має виправдовувати його наймання
Безробітний (дилетант)	Помірно небезпечний. Вторгнення в систему планується на дилетантському рівні по сценаріям фільмів. Цікавлять матеріальні та інформаційні ресурси.	Слабкі	Побутові засоби, легко доступні для придбання	Послуги дешеві. Потреба в збагаченні.
Побутовий (хуліган, наркоман)	Слабко небезпечний. Діє імпульсивно. Цікавлять матеріальні цінності.	Відсутні	Підручні засоби	Здійснює несанкціонований доступ з хуліганських мотивів, або з метою збагачення
Співробітник підприємства	Небезпечний. Можливе видавання інформації про СОІ, планувати та готувати вторгнення в систему, проводити саботаж в СОІ. Цікавлять матеріальні та інформаційні ресурси.	Високі	Набір саморобних або доступних засобів для несанкціонованого доступу	Потреба в збагаченні. Його послуги вартують дорого. Ціль НСД має виправдовувати його наймання. Може переслідувати свої цілі

Далі слід визначити такі ключові для нашого дослідження поняття, як цілі порушника та методи його роботи. Навіть загальне визначення цих позицій дозволяє оцінювати небезпеки і вибудовувати різні моделі рішень, які можуть їй запобігти.

Цілі порушника: розкриття інформації або факту її існування; виклик відмови в обслуговуванні; переривання коректної операції користувача; отримання необмеженого контролю над обчислювальною системою тощо.

Методи роботи порушника:

Перехоплення – пасивний вплив, при якому порушник прослуховує канал зв’язку.

Модифікація – активний вплив – з метою фальсифікації даних або відмови в обслуговуванні або отриманні контролю над системою.

Маскарад – підробка даних, які ідентифікують активну сутність.

Хибне спростування авторства (одна з форм фальсифікації).

Затримка обслуговування – тимчасове припинення обслуговування.

Відмова в обслуговуванні – форма монополізації контролю над СОІ.

Використання цих неправомірних методів роботи зумовлює вибір способу кібератаки і досягнення цілей порушника стосовно інформації або інформаційної інфраструктури, яка цій атаці піддається. З боку СОІ, відповідно, відбувається порушення гарантоздатності, під якою прийнято розуміти комплексну властивість, що поєднує аспекти надійності, функціональної та кібернетичної безпеки й забезпечує здатність таких систем надавати необхідні довірчі послуги та виконувати потрібні функції в заданих режимах та умовах застосування [17]. Способи порушення гарантоздатності СОІ подано в табл.2.5.

Таблиця 2.5

Способи порушення гарантоздатності СОІ

Спосіб порушення гарантоздатності СОІ	**Характеристика інформації та інформаційної інфраструктури, яка порушується**
Впровадження програм-вірусів і програмних закладок на стадії проектування або експлуатації ІТС, що призводять до компрометації системи захисту інформації	Цілісність, Аутентичність Безвідмовність, Підзвітність Конфіденційність
Вплив на обслуговуючий персонал і користувачів ІТС для створення сприятливих умов для реалізації загроз ІБ	Конфіденційність Доступність, Підзвітність
Радіоелектронне придушення ліній зв'язку та систем керування	Доступність, Цілісність
Порушення технології обробки даних та інформаційного обміну	Конфіденційність, доступність, цілісність
Перехоплення інформації технічними каналами її витоку	Конфіденційність
Перехоплення і дешифрування інформації в мережах передачі даних і лініях зв'язку	Конфіденційність, Цілісність
Впровадження електронних пристроїв перехоплення інформації в технічні засоби і приміщення	Конфіденційність,Підзвітність
Нав'язування неправдивої інформації по мережах передачі даних і лініях зв'язку	Достовірність, Безвідмовність Аутентичність, Цілісність
Маніпулювання інформацією (дезінформація, приховування або перекручення інформації)	Цілісність, Достовірність Доступність
Незаконне копіювання, знищення, розкрадання даних і програм, знищення носіїв інформації	Конфіденційність, Доступність
Розкрадання ключів (ключових документів) засобів КЗІ, програмних або апаратних ключів засобів захисту інформації від НСД	Конфіденційність, Цілісність Аутентичність, Безвідмовність

Обсяг інформації, що останнім часом надходить до користувачів із зовнішнього середовища та безперервно зростає, а також потреба підвищення вимог до захисту такої інформації від несанкціонованого доступу зумовлюють постійний аналіз специфіки процесів злому ІТ систем потенційними порушниками [18]. В загальному вигляді послідовність дій порушника представлена в діаграмі на рис.2.5.

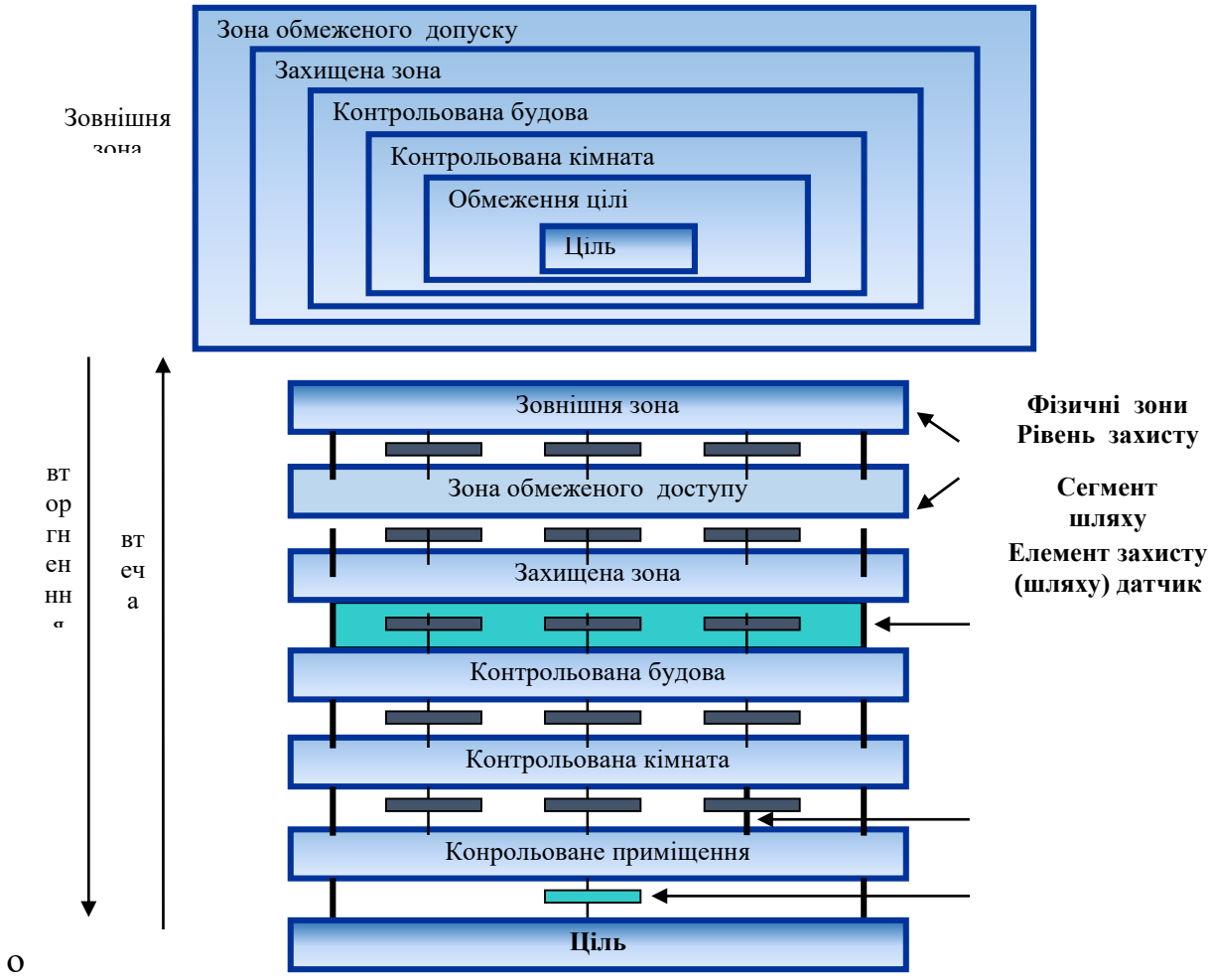


Рис.2.5. Діаграма послідовності дій порушника в ході кібератаки

Для цього, *по-перше*, порушник має достатньо високу кваліфікацію та необхідний фінансовий ресурс, технічне і програмне оснащення, які дозволяють йому створювати складні програмно комплекси для реалізації кібератак. *По-друге*, згідно з принципом Керкхофса [19], він знає алгоритми функціонування засобів захисту, включаючи засоби КЗІ, але до початку атаки знає діючих ключів. *По-третє*, для досягнення поставлених цілей порушник має можливість перехоплення будь якої інформації у транспортній мережі, модифікацію або створення

неприпустимої команди за відносно невеликий час. На рис. 2.6 представлено ієрархію можливостей порушника за рівнями, скадену на підставі роботи [20].

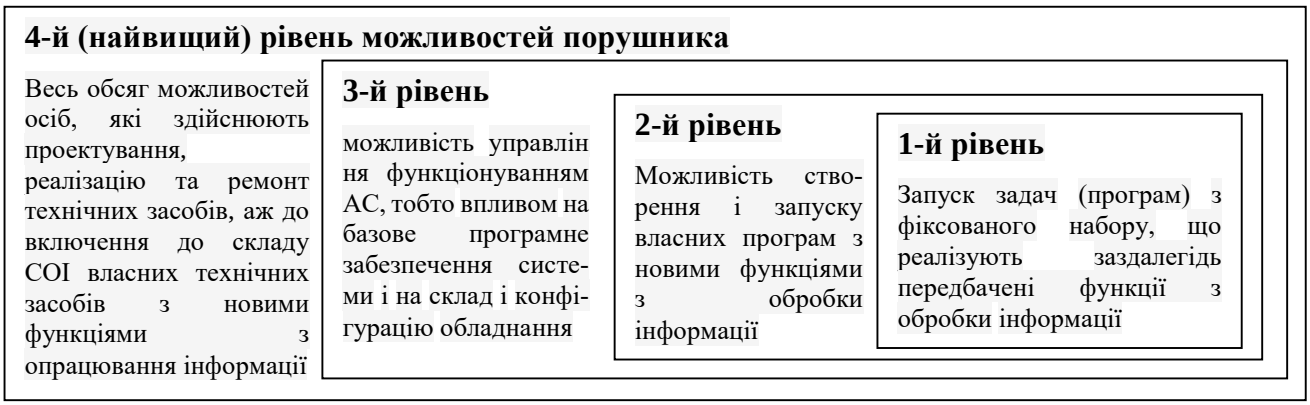


Рис.2.6. Рівні можливостей порушника в ході кібератаки

Виходячи з уточненої моделі порушника в СОІ можливо спрогнозувати наступні варіанти зловмисних дій (потенційні загрози):

<u>стосовно СОІ в цілому</u>:

1) модифікація дійсної команди $x_t \in X_W$ або реальної інформації про внутрішній стан $s_t \in S_W$;

2) формування та надсилання керованому вузлу неприпустимої команди $x_t \in X \setminus X_W$ або фальшивої інформації про внутрішній стан $s_t \in S \setminus S_W$ від керованого вузла;

3) перехоплення в транспортній мережі окремих команд або частки інформації щодо внутрішніх станів задля їх вилучення;

4) крадіжка конфіденційної інформації щодо сервісів, які надаються;

5) модифікація або руйнування програмного коду СОІ;

<u>стосовно засобів програмних реалізацій засобів КЗІ</u>:

1) зміна, знищення або крадіжка критичних параметрів CSP;

2) модифікація програмного коду (криптосхеми) засобу КЗІ.

На підставі аналізу наукових публікацій щодо реалізації кібератак [18; 20-22] та наведених у першому розділі роботи даних про технології шкідливих кодів у поєднанні з відомими методами криптоаналізу за побічними каналами [23-24] була сформована наступна модель кібератак на гарантоздатність в СОІ (у т.ч. на засоби КЗІ [25-26]), яка включає вісім фаз активних дій порушника (рис. 2.7):

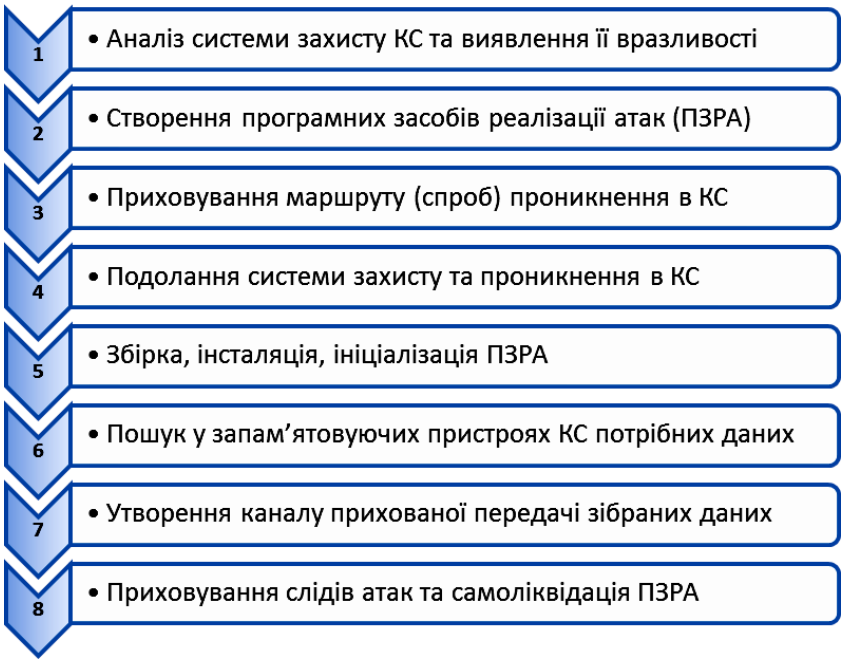


Рис. 2.7. Модель реалізації кібератак на засоби КЗІ (ЦУ) в СОІ

1. Розвідка. На першому етапі порушник, використовуючи всі доступні методи, здійснює приховане вивчення вразливостей комп'ютерної системи (КхщС), яка є технологічною базою функціонування СОІ, а також виявлення слабких місць наявної системи захисту, включаючи програмну реалізацію засобів КЗІ.

2. Розробка. На наступному кроці, який потребує певного часу, здійснюється вивчення отриманої інформації та розробка програмних засобів для реалізації атак (ПЗРА).

3. Маскування. Порушник здійснює заходи щодо усунення ознак, які пов'язують ПЗРА та спосіб його застосування з реальним розробником, та/або створює фіктивні ознаки, що ототожнюються з непричетним до кібератаки суб'єктом. Також він визначає тактику приховування реального маршруту (адрес проміжних вузлів глобальної мережі) спроб проникнення в КС.

4. Проникнення. Використовуючи створені засоби і технології, а також можливості інсайдерського впливу порушник забезпечує подолання системи захисту та проникнення ядра ПЗРА в програмне середовище КС.

5. Підготовка. Далі в автоматичному або автоматизованому режимі реалізується збирання ПЗРА з окремих модулів, його інсталяція та ініціалізація.

6. Реалізація. Ініціалізоване ПЗРА з використанням певної апріорної інформації про підсистеми (елементи) КС, що виконують конкретні функції COI, а також про потрібні порушнику дані, зокрема, чутливі параметри безпеки криптографічних модулів *SSP* виявляє та ідентифікує зазначені об’єкти у запам’ятовуючих пристроях КС. У якості відповідної апріорної інформації можуть виступати розмір файлів, формати даних, певні ключові слова, програмні переривання/ звернення до деяких ресурсів системи тощо. Залежно від цілей кібератаки виявлені об’єкти можуть бути знищені, модифіковані або використані для розкриття конфіденційної інформації.

7. Витік. За необхідності, створюється канал прихованої передачі зібраних даних з використанням методів стеганографії, процедури стискання даних з наступним шифруванням, фізичного переносу під час підключення зовнішніх пристроїв тощо.

8. Самоліквідація. На завершальному етапі ПЗРА включає механізми самознищення та приховування слідів (критичних параметрів) кібератаки. Цей етап реалізується автоматично, в разі настання в КС певних обставин (наприклад, визначеного часу), або автоматизовано на підставі отримання команди ззовні.

Запропонована модель читко відстежується на прикладі шкідливого криптографічного вірусу WannaCry [28] та Petya [29]. WannaCry, як відомо, шукає в мережі комп'ютери з відкритим портом TCP з номером 445, що використовується мережним протоколом прикладного рівня SMBv.1 (*Server Message Block*) для віддаленого доступу до мережних ресурсів та для міжпроцесної взаємодії. У випадку успіху вірус намагається скористатись вразливістю EternalBlue для встановлення «лазівки» DoublePulsar, яка забезпечує завантаження та запуск виконуваного коду WannaCry. WannaCry перевіряє наявність на комп'ютері – мішені «лазівки» DoublePulsar, за допомогою якої завантажується. Математична модель WannaCry її шифрувальної частини може бути описана у вигляді наступної послідовності процедур (PWC1-PWC6).

PWC1. Після запуску WannaCry генерується унікальна для конкретного комп'ютера пара ключів алгоритму RSA: (e, d), де e – відкритий ключ, d – секретний ключ порушника:

$$ed = 1 mod \varphi(N),$$

де $\varphi(N) = (P-1)(Q-1)$, *фукція Ейлера*,

P, Q – випадкові прості числа, $N = P \cdot Q \geq 2^{2048}$.

Для цього необхідно здійсни декілька звернень до функцій генерації випадкових чисел та тестування чисел на простоту. У загальному випадку вказані функції по відношенню до можуть бути внутрішніми або зовнішніми, що використовують можливості криптографічних DLL бібліотек операційної системи [4; 30];

PWC2. Для чергового файлу комп'ютера F_i (певного типу, зокрема, з розширеннями $.docx, .pdf, jpeg, .pptx$ тощо) генерується унікальний ключ довжиною 128 біт $K_i = (k_{i1}, \dots, k_{i128})$, *де* $k_{ij} \in \{0,1\}$ *для* $j = \overline{1,n}$. Виняток становлять файли, які потрібні вірусу для продовження функціонування:

PWC3. Шифрування зазначених файлів відбувається за допомогою симетричного блокового алгоритму AES в режимі CBC:

$$\bar{F}_i = AES_{CBC}(F_i, K_i).$$

PWC4. Кожен ключ симетричного алгоритму K_i шифрується відкритим ключем RSA, результат шифрування $\bar{K}_i$ зберігається в заголовку зашифрованого файлу $\bar{F}_i$:

$$\bar{K}_i \equiv K_i^e mod N, \;\; \bar{F}_i^* = \bar{K}_i \parallel \bar{F}_i.$$

PWC5. Кожен зашифрований файл отримує розширення *.wncry*.

PWC6. Пара ключів RSA ураженої системи шифрується відкритим ключем зловмисника і відправляється на сервери управління, що розташовані в мережі *Tor*, після чого всі ключі з пам'яті інфікованої машини видаляються, а на моніторі з'являється повідомлення з вимогою виплати певної суми у криптовалюті.

Аналіз зазначеної криптографічної схеми свідчить, що у випадку правильного використання криптографічних примітивів, якісної генерації ключів за допомогою фізичних генераторів випадкових чисел, правильного їх знищення на інфікованому комп'ютері відновити зашифровані файли в сучасних умовах майже неможливо. У той же час, на операційних системах (ОС) Windows XP и Windows

Server 2003, якщо комп’ютер не був завантажений наново після його ураження, внаслідок особливостей реалізації ОС алгоритму генерації псевдовипадкових чисел існує можливість відновлення секретних ключів алгоритму RSA і розшифрувати усі перетворені файли. Така можливість досліджена у випадку ОС Windows 7 французькими експертами з компанії Comae Technologies та практично реалізована у вигляді відкритої утиліти WanaKiwi, що надає можливість відновити зашифровані файли [31; 32].

За результати аналізу шифрувального коду Petya експертами міжнародної компанії Positive Technologies, яка спеціалізується на розробці програмного забезпечення в галузі інформаційної безпеки [29] встановлено, що після запуску шкідливого файлу створюється відкладене на декілька хвилин завдання на перезапуск комп’ютера, яке після виконання фактично унеможливлює відновлення даних. Вірус Petya для кожного диска генерує свій ключ до криптографічного алгоритму AES-128, який існує в пам’яті комп’ютера до завершення шифрування, а потім знищується.

Побачимо, як розвиваються в часі відповідні етапи (рис. 2.8).

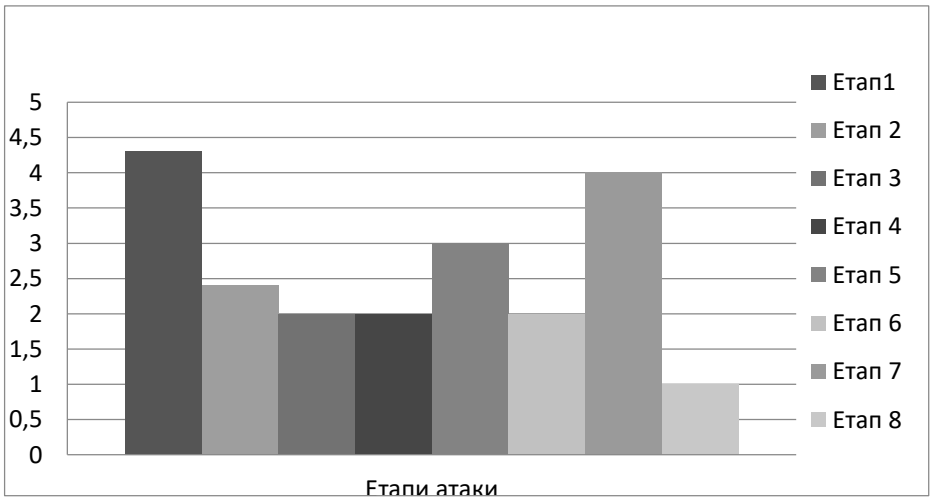


Рис. 2.8. Тривалість та зовнішня активність кібератак

З наведеної діаграми можливо з’ясувати, що серед інших етапів реалізації атаки найбільш тривалим в часі є другій, який в абсолютних одиницях може продовжуватися декілька місяців, найкоротшим є останній етап що виконується у лічені хвилини. Найбільша мережна активність порушника та ПЗРА може

спостерігатися на першому, четвертому та сьомому етапах, тривалість яких може змінюватися залежно від особливостей побудови СОІ, її системи захисту та наявності та ефективності системи контролю безпеки. Саме на цих етапах існує можливість виявити та нейтралізувати атаку.

Як результат, серед основних принципів формування моделей загроз та порушника доцільно відмітити принципи:

адекватності моделей реальним умовам функціонування СОІ;

повноти опису можливих загроз та дій порушника, в тому числі, надання моделям та описам необхідного рівня деталізації;

обґрунтованості побудови моделей, в тому числі щодо визначення кількісних та якісних показників які використані в них. Наприклад із застосуванням спеціалізованих методів та засобів, виконання опису моделей із використанням формалізованих підходів.

Дотримання цих принципів на теперішній час, як правило, пов'язується з використанням засобів, що спрощують або забезпечують: проведення регулярного моніторингу рівня захищеності цінних ресурсів СОІ; отримання (обчислення) показників захищеності ресурсів; побудову актуальних (зміну існуючих) моделей порушника та загроз. Процес функціонування кожного засобу КЗІ в мережі СОІ може бути описаний при цьому, як відомо [7], у вигляді кінцевого автомату Мілі:

$$A = \{X, Y, F, f, S\}, \tag{2.15}$$

де $S = \{s_1, \dots, s_p\}$ – множина внутрішніх станів, $X = \{x_1, \dots, x_q\}$ - множина вхідних сигналів (інформація про поточні стани СОІ або її команд управління), $Y = \{y_1, \dots, y_r\}$ – множина вихідних сигналів, $F: S \times X \to S$ – функція переходів, $f: S \times X \to Y$ – функція виходів.

Автомат Мілі працює, змінюючи внутрішні стани в дискретні моменти часу - такти: $t = 1,2, \dots$. Якщо в момент часу t автомат A перебував у стані $s_t \in S$ і отримав вхідний сигнал $x_t \in X$, то він формує вихідний сигнал $y_t = f(s_t, x_t) \in Y$ та змінює свій внутрішній стан наступним чином: $s_{t+1} = F(s_t, x_t)$. Якщо зміна внутрішніх станів відбувається незалежно від вхідних сигналів $s_{t+1} = F(s_t)$, то

маємо варіант автономного автомату Мілі. У випадку шифруючого автомату система може включати шостий елемент $s_0 \in S$ – початковий стан автомату.

Враховуючи, що технологічною основою побудови сучасних засобів КЗІ є обчислювальна (мікропроцесорна) техніка, будемо вважати, що множини X, Y, S є підмножинами просторів двійкових векторів, а їх елементи – суть двійкові вектори відповідної розмірності, що обумовлена конкретним практичним застосуванням:

$$\begin{cases} x_t = (\alpha_{1t}, \dots, \alpha_{lt}) \in X \subseteq V_l \\ y_t = (\beta_{1t}, \dots, \beta_{mt}) \in Y \subseteq V_m. \\ s_t = (\gamma_{1t}, \dots, \gamma_{nt}) \in S \subseteq V_n \end{cases} \quad (2.16)$$

Звернемо увагу, що в системі (2.16) потужність множин така, що:

$$|X| \leq 2^l, |Y| \leq 2^m, |S| \leq 2^n.$$

Зауважимо, що множина внутрішніх станів може включати деякі підмножини, що не перетинаються та відповідають, зокрема, штатному робочому режиму функціонування - S_W, режиму тестування (проведення профілактичних робіт) - S_T, режиму локального (ручного) управління - S_M тощо:

$$S = S_W \cup S_T \cup S_M \cup \dots \text{ , де } S_i \cap S_j = \emptyset. \quad (2.17)$$

За умов штатного функціонування шифруючого автомату (засобу КЗІ) в режимі зашифрування залежно від введеного ключу та поточного його стану на основі відкритої команди x_t формується зашифрована команда y_t, яка передається другому вузлу СОІ або керованому кінцевому (виконавчому) пристрою. В режимі розшифрування з відповідним ключем відновлюється початковий вигляд команди. Деякі з вхідних команд можуть бути не припустимі у штатному робочому режимі функціонування, тобто під час перебування шифруючого автомату у станах $s_t \in S_W$. Їх надходження може призводити до певних збоїв в роботі засобу КЗІ, наприклад, незапланований перехід у режим тестування або ручного управління.

Несанкціонована зміна КС та кожен збій у роботі засобу КЗІ можуть мати наслідком тимчасову або постійну втрату керованості СОІ, що в свою чергу потребує часу на її повернення у нормальний режим та призведе до витрат

фінансових та матеріальних ресурсів. Тому доцільно розглядати множину вхідної інформації у вигляді об’єднання припустимих та неприпустимих команд:

$$X = X_W \cup (X \setminus X_W). \quad (2.18)$$

Слід також зазначити, що для синхронізації дій підпорядкований вузол для підтвердження виконання отриманих команд у відповідь надсилає старшому з них інформацію про новий стан роботи.

2.2.2. Метод виявлення атак на програмні реалізації засобів криптографічного захисту інформації в СОІ

Викладене у попередніх розділах підтверджує той факт, що в умовах кібератак, випадкових відмов та збоїв, а також помилкових дій обслуговуючого персоналу та засобів КЗІ, що використовуються для забезпечення конфіденційності та цілісності технологічної інформації, необхідно вжити поточні заходи щодо дієвого контролю за їх станом. Враховуючи, що контроль цілісності програмного коду СОІ або засобів КЗІ [30] потребує певного часу, він може бути реалізований переважно на етапі ініціалізації системи та засобів, або завершення їх роботи. Разом з цим слід враховувати, що такий контроль не може бути поширений на критичні параметри безпеки, які зазнають змін своїх значень у процесі виконання відповідних процедур.

Надалі в цьому розділі для простоти викладення під поняттям «об’єкт контролю» розуміємо як СОІ в цілому, так і її компоненти, включаючи вбудовані засоби захисту.

2.2.2.1 Процедура виявлення прихованих каналів в ході атак на програмну реалізацію стійких криптографічних алгоритмів

Сучасний етап розвитку криптографічного захисту інформації і стандартизації у галузі криптографії характеризується широкою доступністю так званої «сильної» криптографії, тобто відкритих стандартів практично стійких алгоритмів шифрування, функцій хешування, алгоритмів цифрового підпису, протоколів генерації та управління ключами. Існують вихідні тексти програм, що

реалізують відповідні процедури, доступні в мережі Інтернет та у спеціалізованих науково-практичних виданнях [4; 33]. Відповідно, складність задач атакуючої сторони щодо створення криптографічних ПЗРА суттєво спрощується, а ефективність вирішення задач по відновленню (дешифруванню) даних КС, які зашифровані за допомогою ПЗРА, має об'єктивну тенденцію до постійного з року в рік зниження. При цьому мають місце такі групи загроз, що пов'язані з надійністю [20; 27-28]:

помилки програмування криптографічних примітивів;

помилки використання криптографічних примітивів у криптопровайдерах;

помилки передачі параметрів у криптопровайдер і повернення результатів обробки;

помилки і відмови апаратної платформи;

випадкові та навмисні порушення цілісності програм і даних криптопровайдерів.

Ці фактори можуть бути передумовами утворення прихованих каналів витоку інформації про роботу криптографічних ПЗРА. Розглянемо два варіанти проведення атак на реалізацію ПЗРА, що не передбачають наявності критичної інформації про їх роботу.

Варіант 1. Існує певний криптографічний ПЗРА (криптосистема) з секретними ключами на основі стійкого блокового симетричного криптографічного алгоритму, який забезпечує шифрування у режимі *OFB* (рис.2.9) або інакше - гамування зі зворотнім зв'язком [33-35]. Необхідно: побудувати атаку на реалізацію вказаної криптосистеми, щоб забезпечити можливість часткового або повного відновлення інформації (дешифрування).

Такий режим роботи криптоалгоритму досить широко використовується для побудови повно зв'язаних мереж зв'язку, у яких кожен абонент має надіслати повідомлення будь якому іншому абоненту цієї мережі. Для цього режиму характерним є наявність вектору ініціалізації IV - випадкового числа, що забезпечує на припустимому рівні ймовірність не перекриття шифру у разі використання одного ключа K протягом певного часу [33].

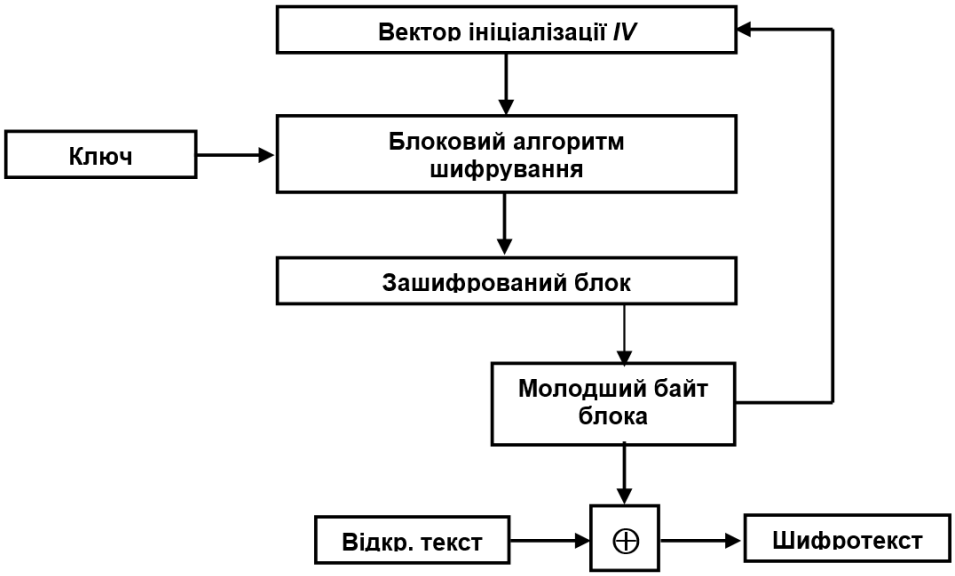


Рис. 2.9. Застосування блокового симетричного алгоритму в режимі *OFB*

Ідея атаки на цю криптосистему полягає у реалізації фіктивного генератора випадкових даних криптоалгоритму, що включається замість реального PRNG (pseudorandom number generator) забезпечує приховане повторення векторів ініціалізації або ключів. Внаслідок цього деякі повідомлення будуть зашифровані однаково:

$$\begin{cases} \tilde{S}_1 = \tilde{T}_1 + \tilde{\Gamma}_j \\ \ldots \quad \ldots \quad \ldots \\ \tilde{S}_{m_j} = \tilde{T}_{m_j} + \tilde{\Gamma}_j \end{cases}, \tag{2.19}$$

де $\tilde{T}_1$, ..., $\tilde{T}_{m_j}$ – відкриті повідомлення;

$\tilde{\Gamma}_j$, $j = \overline{0.2^k - 1}$ – послідовність знаків гами, що утворені з одного вектораініціалізації та ключа алгоритму;

$\tilde{S}_1$, ..., $\tilde{S}_{m_j}$ – відповідні зашифровані повідомлення.

У цьому випадку, залежно від кількості однаково зашифрованих повідомлень m_j та надлишковості вихідних текстів повідомлень відповідні зашифровані повідомлення можуть бути частково, або повністю дешифровані [5]. Для повторення векторів ініціалізації IV генератор випадкових біт доцільно використовувати таким чином:

$$IV = \langle \varphi(\alpha_1, \alpha_2, ..., \alpha_k) \rangle = \langle \beta_1, \beta_2, ..., \beta_b \rangle, \tag{2.20}$$

де $\alpha_1, \alpha_2, ..., \alpha_k$ - послідовність випадкових біт; $\varphi(...)$ - деяка однозначна функція, що забезпечує розширення випадкової послідовності до заданого розміру b; $\beta_1, \beta_2, ..., \beta_b$ - координати вектору ініціалізації.

З метою рішення задачі спочатку розрахуємо припустиму кількість випадкових біт для забезпечення необхідної кількості повторів однаково зашифрованих повідомлень. Нехай у мережі присутні M абонентів, кожен з яких у середньому щодоби відправляє $\overline{\mu}$ повідомлень за кожним напрямом з'єднань. При цьому T - термін дії спільного ключу (діб).

Тоді середня кількість відправлених повідомлень $\overline{N}$ становить величину:

$$\overline{N} = \binom{M}{2} \cdot \overline{\mu} \cdot T \,. \tag{2.21}$$

Протягом вказаного періоду середня кількість повторів $\overline{R}$ кожного із 2^k значень векторів випадкових біт $\langle \alpha_1, \alpha_2, ..., \alpha_k \rangle$ у випадку їх рівномірного розподілу становитиме:

$$\overline{R} = \frac{\overline{N}}{2^k} = \binom{M}{2} \cdot \overline{\mu} \cdot T \Big/ 2^k \,. \tag{2.22}$$

Виходячи з рівняння (3.8) отримуємо оцінку припустимого числа випадкових біт:

$$k \le \left\lfloor \log_2 \binom{M}{2} \cdot \overline{\mu} \cdot T - \log_2 \overline{R} \right\rfloor. \tag{2.23}$$

За допомогою нерівності у кожному конкретному випадку мережі, що атакується, можливо розрахувати припустиму кількість випадкових біт, для забезпечення середньо статичної кількості $\overline{R}$ однаково зашифрованих повідомлень. Звичайно, випадкові дані, що використовуються для криптографічних перетворень підлягають статистичному тестуванню, але досить мала довжина вектору IV (довжина блоку для більшості стандартних криптоалгоритмів дорівнює *64* або *128* біт) суттєво обмежує можливості щодо застосування статистичних критеріїв. Для перевірки рівномірності розподілу послідовностей такої довжини переважно застосовуються критерії частот знаків та біграм [10], що висуває відповідні вимоги до функції розширення. Тому зрозуміло, що для забезпечення рівної зустрічаємості біграмм у двійковому векторі

$\langle \alpha_1, \alpha_2, ..., \alpha_k \rangle$ доцільно величину k обрати кратною *4* (всього існує чотири варіанти біграм 00, 01, 10, 00).

Наприклад, у разі мережі спеціального зв'язку що включає $M = 10$ абонентів із середньою інтенсивністю відправлення повідомлень протягом терміну дії одного мережного ключу $\overline{\mu} \cdot T = 10$ для отримання середньої кількості повторів шифрування $\overline{R} = 2$ необхідно мати кількість випадкових біт не більш $k \le 7$.

Для приховування статистичних залежностей в векторі *IV* та отримання необхідної кількості біт у відповідному виразі у якості функції φ слід скористатися безумовно стійкою хеш–функцією, наприклад *MD5*, яка входить до стандартного набору алгоритмів ОС *Windows,* або аналогічною іншою. При цьому зі 128 біт дайджесту, що отримуємо за допомогою хеш-функції, скористаємося необхідною кількістю.

Варіант 2. Нехай спостерігається ПЗРА (криптосистема), що побудований на основі стійкого симетричного блокового криптографічного алгоритму шифрування у режимі *OFB* з використанням відкритого розподілу ключів на основі протоколу Ель-Гамаля [33].

Задача полягає у організації атаки на реалізацію криптосистеми таким чином, щоб забезпечити повне дешифрування.

Спочатку роботи криптосистеми у протоколі Ель-Гамаля за допомогою циклічного елемента *g* деякого поля та секретного ключу асиметричного алгоритму *x* формується відкритий ключ *y*:

$$y = g^x \bmod p \qquad (2.24)$$

де *p* – деяке велике просте число, довжиною від 1024 біт.

Для зашифрування сеансового ключу симетричного алгоритму генерується випадкове число *k*, обчислюється величина:

$$y_1 = g^k \bmod p$$

та шифрується секретний ключ симетричного алгоритму *K*:

$$\delta = K \cdot y^k \bmod p \qquad (2.25)$$

Пара (δ,y_1) разом із зашифрованим повідомленням передається власнику секретного ключу x, який на основі отриманої пари (δ,y_1) обчислює секретний ключ симетричного алгоритму K:

$$K = \delta \cdot y_1^{-x} \bmod p \tag{2.26}$$

Атаку на вказаний протокол організуємо шляхом підміни випадкового числа k псевдовипадковим таким чином, щоб система тестування не виявила цього факту. Нехай $k \in \{\xi_t.,\ t=1,2,\ \dots\ ,\Psi\}$, при цьому потужність Ψ множини припустимих значень псевдовипадкового числа k виберемо досить великою, але достатньо меншою ніж продуктивність спеціалізованої обчислювальної системи щодо перебору ключів за припустимий проміжок часу.

Обчислення можливих варіантів ключа K у цьому випадку виконаємо за допомогою МОПП виходячи з рівняння (2.23) на підставі множини припустимих значень псевдовипадкового числа $\{\xi_t\}$.

У підсумку слід зауважити, що доступ к інформації «істинного» генератора випадкових даних та методи його підміни у кожному конкретному випадку залежать від апаратної та програмної платформ, на яких функціонує ПЗРА, особливостей його реалізації, наявності в складі автоматизованої системи засобів захисту від несанкціонованого втручання в її роботу тощо. Для створення фіктивного генератора випадкових простих чисел можливо скористатись послідовністю чисел Мерсенна $M_p = 2^p - 1$, якщо p – просте число, M_p також просте.

У зв’язку з цим, особливий інтерес становлять сучасні технології, що дають можливість підвищити ефективність класичних методів дешифрування [37], які використовуючи вразливості в певному криптографічному алгоритмі та/або протоколі забезпечують відновлення секретного ключа, за допомогою якого створені зашифровані повідомлення, та/або розкриття вихідного значення зашифрованої інформації. При цьому відновлення вихідної інформації може бути повним або частковим, тому говорять про повне або часткове дешифрування. Одною з таких технологій є процедура виявлення атак на програмні реалізації засобів КЗІ на основі реалізації двоступеневого критерію виявлення аномалій.

2.2.2.2 Метод виявлення атак на програмні реалізації засобів КЗІ на основі реалізації двоступеневого критерію виявлення аномалій

Стосовно виявлення кібернетичних атак, збоїв об’єкту контролю можливо зробити висновок щодо доцільності побудови системи поточного контролю, яка відповідала б наступним вихідним вимогам:

1. Універсальний характер методу контролю, можливість застосування для різної природи об’єктів, стан функціонування яких контролюється.

2. Простота програмної реалізації, невибагливість щодо архітектури та ресурсів мікроконтролеру, за допомогою якого здійснюється обробка даних.

3. Максимально можлива швидкодія.

4. Мінімальна затримка з прийняттям рішення щодо аномальної поведінки об’єкта контролю. Це обумовлене необхідністю виключення можливості настання ситуації з катастрофічними наслідками для об’єктів управління в СОІ (перш за все, рухомих, із швидкими технологічними процесами тощо).

5. Мінімально можливий рівень помилкових спрацювань, обумовлених тимчасовими несуттєвими відхиленнями у роботі об’єкта контролю.

З урахуванням наведених вимог, можливо зробити припущення, щодо доцільності оцінки правильності функціонування об’єкт контролю за допомогою статистичних методів. Пояснимо фізичний зміст задачі. Вважаємо, що у випадку штатного функціонування СОІ деякий потік вимірюваних даних від об’єкту контролю є стаціонарним, а після настання деякої події він змінює розподіл своїх значень. Необхідно максимально точно виявити момент настання цієї ситуації та прийняти рішення щодо аномальності ситуації. Формалізуємо задачу.

Нехай стосовно випадкової послідовності, що аналізується:

$$X = \{x_t, x_t \in R,\ t = \overline{1, N}\},$$

розглядаються дві складних гіпотези: H_0: послідовність X є стаціонарною з єдиною функцією розподілу ймовірностей, H_1: послідовність X є конкатенацією - результатом об’єднання - двох стаціонарних випадкових послідовностей з різними функціями розподілу:

$$X = X_1 || X_2, \text{ де } X_1 = \{x_t, t = \overline{1, n^*}\},$$
$$X_2 = \{x_t, t = \overline{n^* + 1, N}\}, n^* = [\theta N], 0 < \theta < 1. \quad (2.27)$$

Потрібно оцінити точку об'єднання n^*.

Вважається, що послідовності X_1 і X_2 відрізняються між собою однією з двовимірних функцій розподілу, а саме, розподілу ймовірностей вектору (x_t, x_{t+2}):

$$F(u_0, u_1) = P\{x_t \le u_0, x_{t+2} \le u_1\} \quad (2.28)$$

до моменту $t_1^* = n^* - 2$ включно дорівнює $F_1(u)$, а при $t \ge t_2^* = n^* + 1$ дорівнює $F_2(u)$, причому

$$||F_1(u) - F_2(u)|| \ge \varepsilon > 0, \text{ де } ||\ldots|| \text{ - звичайна sup-норма.} \quad (2.29)$$

Відомо що функція розподілу кінцевомірного випадкового вектора може бути наближена рівномірно з будь-якою точністю функцією розподілу ймовірностей випадкового вектора з кінцевим числом значень. Звідси випливає, що якщо подати множину R у вигляді поєднання досить великої кількості областей $\{Aj,\ j = \overline{1, r}\}$, що не перетинаються $A_i \cap A_j = \emptyset$ для $i \ne j$, то вектор (x_t, x_{t+2}) можна апроксимувати по розподілу вектором з кінцевим числом значень.

Тому, якщо ввести нові випадкові послідовності

$$V_t^{ij} = I\left(x_t \in A_i, x_{t+2} \in A_j\right), \text{ де } 1 \le i \le r, 1 \le j \le r, \quad (2.30)$$

де *I(A)* - індикатор множини *A*, то хоча б в одній з них відбувається зміна математичного очікування.

Отже, якщо скористатися алгоритмом, який виявляє зміну математичного очікування, то цей же алгоритм виявить і зміну функції розподілу. Ця обставина дозволила в роботі обмежитися розробкою тільки одного, базового алгоритму, який може виявляти зміну математичного очікування. Для цього з метою виявлення моментів "розладнань" запропоновано сімейство статистик виду:

$$Y_N(n, \delta) = \left[\frac{n}{N}\left(1 - \frac{n}{N}\right)\right]^{\delta} \cdot \left[\frac{1}{n}\sum_{k=1}^{n} x_k - \frac{1}{N-n}\sum_{k=n+1}^{N} x_k\right], \quad (2.31)$$

де $0 < \delta \le 1,\ 1 \le n \le N - 1,\ X = \{x_k,\ k = \overline{1, N}\}$ – послідовність, що досліджується.

Наведене сімейство статистик у випадку фіксованого n є узагальненим варіантом статистики Колмогорова - Смірнова, що використовується для перевірки гіпотез збігу або відмінності функцій розподілу у двох вибірках. Статистика виду (2.27) в разі $\delta = 1$ при $N \to \infty$ та збереженні співвідношення між обсягами "склеєних" реалізацій мінімізує максимально можливу ймовірність помилки оцінювання моменту "розладнання" (мінімаксна по порядку).

При цьому:

$$P\{\max_{1\le n\le N-1}\sqrt{N}|Y_N(n,1)| > C^{(1)}\} \to 2\sum_{k=1}^{\infty}(-1)^{k+1}\exp(-2k^2\left(\frac{C^{(1)}}{\sigma_*}\right)^2) \equiv f(C^{(1)}), \quad (2.32)$$

де параметр σ_* є стандартним відхиленням, $C^{(1)}$ – границя критерію, перевищення якої буде сприйматися, як виникнення «розладнання», значення n_*, для якого це відбулося, є шуканим моментом «розладнання».

Крок другий. Зафіксувавши рівень ймовірності α "помилкової тривоги" про розладнання, під час статистичної обробки реальних даних визначимо рівень порога першого рівня $C^{(1)}$:

$$\alpha = f\left(C^{(1)}\sqrt{N}/\bar{\sigma}_*\right), \quad (2.33)$$

де $\bar{\sigma}_*$ є оцінкою параметра σ_* (стандартне відхилення), а N є обсягом вибірки - послідовності, що досліджується.

У випадку гіпотези H_1 про подання вихідної послідовності вимірювань у вигляді конкатенації декількох стаціонарних випадкових послідовностей з різними функціями розподілу ймовірностей, застосуємо критерій припустимої кількості спрацювань («розладнань») Z. Тобто, вважаємо що має місце наступне рівняння:

$$n_1 + n_2 + \ldots + n_Z = N, \text{де } 2 \le Z < N - 2. \quad (2.34)$$

Для невеликої кількості послідовностей з метою визначення границі критерію можна скористатися нерівністю Чебишева, якщо випадкова величина Z має математичне очікування μ та стандартне відхилення σ для заданого $\varepsilon > 0$:

$$P\{|Z - \mu| \ge \varepsilon\} \le \frac{\sigma^2}{\varepsilon^2}. \quad (2.35)$$

Якщо Z є випадковою величиною з одномодовим розподілом ймовірностей з математичним очікуванням μ та стандартним відхиленням $0 < \sigma < \infty$, то для будь якого $\lambda > \sqrt{8/3} \approx 1.63299$ ..., має місце нерівність Височанського – Петунина, що покращує оцінку ймовірності відхилення:

$$P\{|Z - \mu| \geq \lambda\sigma\} \leq \frac{4}{9\lambda^2}\,. \tag{2.36}$$

Зокрема, для типового відхилення у 3σ (три сигма) рівень значущості критерію - ймовірність "помилкової тривоги" обчислюється як:

$$\alpha = \frac{4}{9\lambda^2} \approx 0.0494.$$

Більш точний результат можливо отримати для випадку випадкової величини Z, що є сумою досить великої кількості незалежних випадкових величин ($m \to \infty$):

$$Z = z_1 + z_2 + \ \ldots + z_m.$$

Згідно інтегральної граничної теореми ймовірність відхилення можливо апроксимувати за допомогою функції нормального розподілу ймовірностей що має нульове математичне очікування та одиничну дисперсію - $N(0,1)$:

$$P\left\{\left|\frac{Z-\mu}{\sigma}\right| \geq t_{1-\alpha}\right\} = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{t_{1-\alpha}} \exp(-\frac{t^2}{2})dt\,. \tag{2.37}$$

Розраховуємо границю критерію другого рівня:

$$C^{(2)} = \mu + t_{1-\alpha}\sigma\,.$$

Таким чином, алгоритм обчислення критерію включає наступні кроки (рис. 2.10):

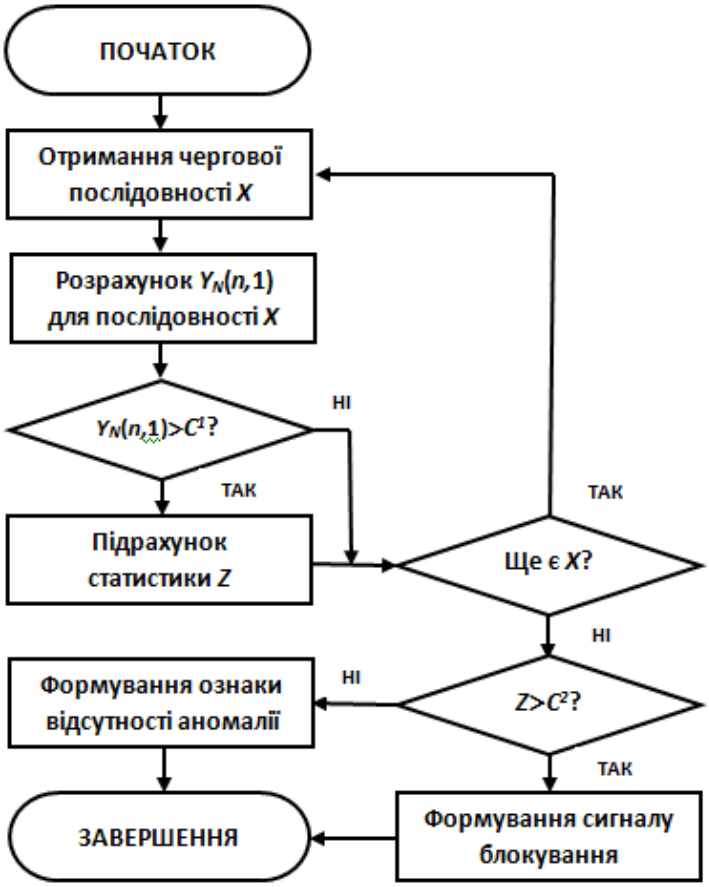


Рис. 2.10 Блок – схема алгоритму реалізації двоступеневого критерію виявлення аномалії у поведінці (стані) засобу КЗІ

Можливо бачити, що складність алгоритму, який реалізує запропонований критерій, оцінюється величиною O(N). Таким чином, його реалізація за принципом «ковзаюче вікно» не викличе суттєвого навантаження на обчислювальну систему [38].

2.3. Модель функціонування шифратора БАЗ (модуля криптографічного захисту інформації) в СОІ

З метою забезпечення конфіденційності та цілісності інформації при прийомі та передачі в СОІ на третьому кроці другого розділу дистертаціної роботи було побудовано модель функціонування (криптосхему) шифратора БАЗ (модулю КЗІ). Модель включає генератор ПВП, алгоритм генерації підстановок заміни, систему контролю і блокування та вузол шифрування (рис. 2.11).

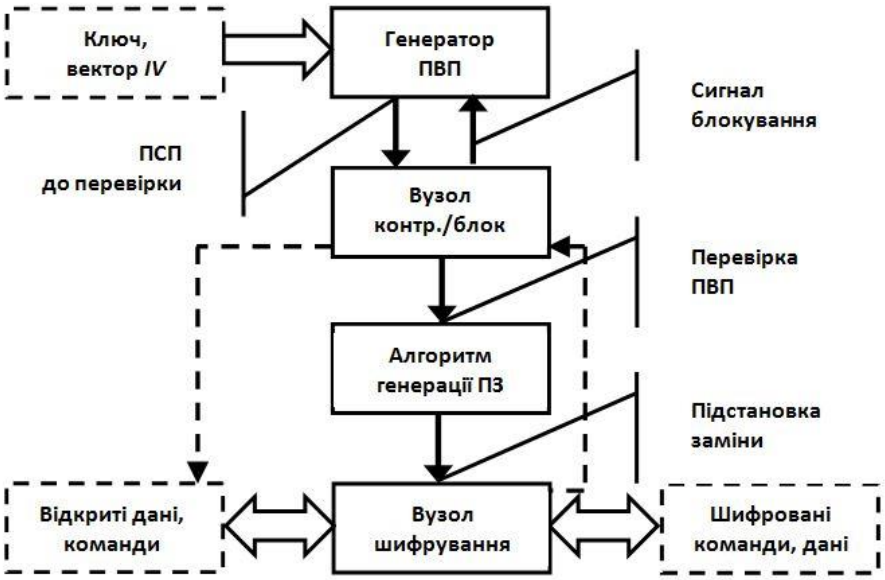


Рис. 2.11. Модель функціонування шифратора БАЗ

Базою для моделі стали модель та метод виявлення атак на програмні реалізації засобів КЗІ в СОІ та методу генерації потоку підстановок з використанням шифру БАЗ. Її квінтесенція полягає в тому, що вона передбачає:

по-перше, при ініціалізації ключа – генерацію ПВП у відповідному блоці;

по-друге, перевірку правильності роботи генератора ПВП у блоці «Вузол контролю та блокування»;

по-третє, подачу ПВП сигналу блокування при збої генератора ПВП та його зупинку до відновлення правильної роботи;

по-четверте, при позитивному проходженні перевірки ПВП: – початок роботи алгоритму генерації підстановок заміни з подальшим шифруванням даних; – перевірку в блоці «Вузол контролю та блокування» [39].

Висновки до другого розділу

Виходячи з можливості застосування для цілей генерації рівномірно розподілених випадкових (РРВП) та/або псевдовипадкових (ПВП) послідовностей, статистично безпечних алгоритмів блокового шифрування (БШ) у другому розділі розроблено моделі і методи забезпечення імітостійкості та конфіденційності в системах обробки інформації в умовах загроз (кібератак). Головним елементом є

метод генерації потоку підстановок з використанням шифру багатоалфавітної заміни (БАЗ), удосконалений за рахунок алгоритмів генерації потоку та вибору степеню підстановок заміни для шифру багатоалфавітної заміни. Його застосування, дозволяє:

по-перше, обрати степінь підстановок замін та оцінити якість послідовності підстановок;

по-друге, підвищити надійність імітостійкого шифрування та знизити ймовірність підробки команди управління до прийнятної для практичного застосування величини порядку 10^{-6}.

З використанням методу може бути генерована будь-яка підстановка з S_n - симетричної групи підстановок степені n.

В другому розділі враховуючи перелік загроз, що сформовані у першому розділі роботи, а також виходячи з уточнених моделей порушника і загроз та автоматної моделі безпеки функціонування каналів управління системи розроблено метод виявлення атак на програмні реалізації засобів криптографічного захисту інформації в СОІ.

Головними елементами методу є:

1) процедура виявлення прихованих каналів в ході атак на програмну реалізацію стійких криптографічних алгоритмів;

2) модель виявлення атак на програмні реалізації засобів криптографічного захисту інформації в СОІ на основі реалізації двоступеневого критерію виявлення аномалій.

Їх застосування в умовах загроз і обмежень дозволяє, виявити момент настання певної критичної ситуації та прийняти рішення щодо її аномальності, за допомогою якого створені зашифровані повідомлення та/або розкрити вихідне значення зашифрованої інформації в СОІ.

Квінтесенцією розділу є розроблена модель функціонування (криптосхема) шифратора БАЗ (модулю криптографічного захисту інформації) в СОІ, впровадження якої за рахунок методу контролю стану гарантоздатності програмних реалізацій засобів КЗІ та методу генерації потоку підстановок для шифру БАЗ дозволяє забезпечити

конфіденційність і цілісність інформації, що циркулює в СОІ, та в умовах кібератак підвищити функціональну безпеку та життєзданість самої системи;

Список джерел, використаних у другому розділі

1. Гулак Г. М., Бурячок В. Л., Складанний П. М. Швидкий алгоритм генерації підстановок багатоалфавітної заміни. *Захист інформації*. 2017. № 2. С. 173-177.

2. Семко В. В. Бурячок В. Л., Толюпа С. В., Складанний П. М. Модель управління захистом інформації в інформаційно-телекомунікаційній системі. *Вісник Національного університету «Львівська політехніка»*. Серія: Радіоелектроніка та телекомунікації : збірник наукових праць. 2015. № 818. С. 151–155.

3. Buriachok V., Sokolov V., Skladannyi P. Security Rating Metrics for Distributed Wireless Systems. Proceedings of the 8th International Conference on “Mathematics. Information Technologies. Education” (MoMLeT&DS’2019), June 2–4, 2019: Vol. 2386. Aachen : CEUR, 2019. P. 222-233.

4. Феллер В. Введение в теорию вероятностей и ее применение. Т.1. Москва: Мир, 1964. 498 с.

5. Бабаш А., Шанкин Г. Криптография. Москва: СОЛОН-Р, 2002. 512 с.

6. Матов О.Я., Василенко В.С., Василенко М.Ю. Криптографічна стійкість методів шифрування на основі перетворень з використанням лишкових класів. *Реєстрація, зберігання і оброб. даних*. 2012. Т. 14, № 4. С. 81-87.

7. Зубов А.Ю. Криптографические методы защиты информации. Совершенные шифры : Учебное пособие. Москва : Гелиос АРВ, 2005. 160 с.

8. Духин А.А. Теория информации : Учебное пособие. Москва : Гелиос АРВ, 2007. 248 с.

9. Горбенко І.Д., Горбенко Ю.І. Прикладна криптологія: Теорія. Практика. Застосування: Монографія. (Вид. 2-ге, перероб. і доп.). Харків: Форт, 2012. 880 с.

10. Гулак Г., Ковальчук Л. Різні підходи до визначення випадкових послідовностей / Науково-технічний збірник *«Правове, нормативне та*

метрологічне забезпечення системи захисту інформації в Україні», вип. 3. Київ, 2001. С.127-133.

11. . Konheim A.G. Cryptography: A primer. New York: John Wiley & Sons, Inc., 1981. 432 p.

12. Гулак Г.М., Мухачов В.А., Хорошко В.О., Яремчук Ю.Є. Основи криптографічного захисту інформації : підручник. Вінниця: ВНТУ, 2011. 198 с.

13. Семко В.В. Бурячок В.Л., Толюпа С.В., Складанний П.М. Ситуаційне управління доступом в інформаційно-телекомунікаційній системі. *Проблеми телекомунікацій*. 2015. № 2. С. 54-61.

14. Зубок В.Ю. Особливості моделі порушника при аналізі атак на глобальну маршрутизацію в Інтернеті. *Електрон. Моделювання*. 2019. Т.41, № 5. С. 59-69.

15. Складанний П.М. Модель загроз безпеки криптосистем. / І Міжнародна науково-технічна конференція «Актуальні проблеми розвитку науки і техніки» (Київ, 20 жовтня 2015р.). Київ: ДУТ, 2015. С.35-37.

16. Воскобойніков С.О., Кащук В.Ф. Формування професійної компетентності сучасного фахівця з кібербезпеки для реалізації компетенцій комп'ютерної криміналістики / Актуальні проблеми управління інформаційною безпекою держави : зб. тез наук. доп. наук.-практ. конф. (Київ, 30 березня 2018 р.). Київ : Нац. акад. СБУ, 2018. С. 38-40.

17. Харченко В. С. Гарантоздатність комп'ютерних систем: межа універсальності в контексті інформаційно-технічного стану. Радіоелектронні і комп'ютерні системи. 2007. № 8. С. 8-16.

18. Бурячок В.Л. Модель формування дерева атак для одержання інформації в інформаційно-телекомунікаційних системах і мережах при вилученому доступі. *Інформатика та математичні методи в моделюванні*. 2013, Том 3, №2. С. 123-131.

19. Алферов А.П., Зубов А.Ю., Кузьмин А.С., Черемушкин А.В. Основы криптографии : Учебное пособие. Москва : Гелиос АРВ, 2002. 480 с.

20. Бойченко О., Зюбіна Р. Метод розрахунку ймовірності реалізації загроз інформації з обмеженим доступом від внутрішнього порушника. *Безпека інформаційних систем і технологій*, 2019. № 1(1). С. 19-26.

21. Олійников Р., Горбенко І., Казимиров О., Руженцев В., Горбенко Ю. Принципи побудови і основні властивості нового національного стандарту блокового шифрування України. *Захист інформації*, 2015, том 17, №2, квітень-червень. С.142-157.

22. Бурячок В.Л., Гулак Г.М., Хорошко В.О. Завдання форми та способи ведення воєн у кібернетичному просторі. Наука та оборона, 2011, № 3. С. 35-42.

23. Жуков А.Е. Криптоанализ по побочным каналам (side channel attacks).

24. Kelsey J., Schneier B, Wagner D., Hall C. Side Channel Cryptanalysis of Product Ciphers / 5th European Symposium on Research in Computer Security Louvain-la-Neuve, Belgium September 16–18, 1998 Proceedings, Berlin, Springer, 1998, С.97-111.

25. Гулак Г.М. Забезпечення безпеки засобів КЗІ у кіберпросторі / Матеріали науково-технічної конференції «Сучасні інформаційно-телекомунікаційні технології», том ІУ Сучасні технології інформаційної безпеки. Київ, 2015. С. 100-102.

26. Бурячок В.Л., Гулак Г.М., Мельник С.В. Метод оцінювання ефективності кібернетичного озброєння з подолання засобів криптографічного захисту інформації. *Інформаційна безпека людини, суспільства, держави*, 2011. № 1(8). С. 100-106.

27. Складанний П.М. Модель загроз безпеки криптосистем / І Міжнародна науково-технічна конференція «Актуальні проблеми розвитку науки і техніки» (Київ, 20 жовтня 2015р.). Київ: ДУТ, 2015. С.35-37.

28. Гулак Г.М., Бурячок В.Л., Складаний П.М,. Кузьменко Л.В. Криптовірологія: загрози безпеки гарантоздатним інформаційним системам і заходи протидії шифрувальним вірусам. *Кібербезпека: освіта, наука, техніка*. 2020. Том 2. №10. С. 6-28.

29. Особенности Petya, URL https://xakep.ru/2017/06/28/petya-write-up/

30. Положення про порядок розроблення, виробництва та експлуатації засобів криптографічного захисту інформації. Наказ Адміністрації Державної служби спеціального зв'язку та захисту інформації України від 20.07.2007 № 141, із змінами, внесеними згідно з Наказами Адміністрації ДССЗЗІ від 04.12.2009 № 254, від 02.03.2012 № 90, від 14.12.2015 № 767.

31. Suiche M., 2017. WannaCry Decrypting files with WanaKiwi + Demos. [online] Comae Technologies. Available at: https://blog.comae.io/wannacry-decrypting-files-with-wanakiwi-demo-86bafb81112d. [Accessed 5 October 2020]

32. Складанний П.М. Псевдовипадкові послідовності та методи захисту інформації. / II Міжнародна науково-технічна конференція «Актуальні проблеми розвитку науки і техніки» (Київ, 12 грудня 2015р.). Київ: ДУТ, 2015. С.55.

33. Хопкрофт Д., Мотвани Р., Ульман Д. Введение в теорию автоматов, языков и вычислений. Москва: Вильямс, 2002. 528 с.

34. Шнайер Б. Прикладная криптография. Протоколы, алгоритмы, исходные тексты на языке Си. – Москва: Триумф, 2002. 816 с.

35. ДСТУ 7624:2014 Информационные технологии. Криптографическая защита информации. Алгоритм симметричного блочного преобразования.

36. Складанний П.М. Принцип побудови шифра на основі ГОСТ 28147. / III Міжнародна науково-технічна конференція «Актуальні проблеми розвитку науки і техніки» (Київ, 02 березня 2016р.). Київ: ДУТ, 2016 С.74.

37. Семко В.В., Гулак Г.М. Екологія кібернетичного простору. *Інформаційна безпека людини, суспільства, держави*. 2014, № 2(10). С.100-108.

38. Гулак Г.М., Семко В.В., Складанний П.М. Модель системи виявлення вторгнень з використанням двоступеневого критерію виявлення мережних аномалій. Сучасний захист інформації. 2015. № 4. С. 81-85.

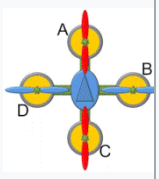
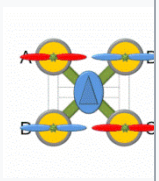
39. Гулак Г.М., Складанний П.М. Забезпечення гарантоздатності автоматизованих систем управління та передачі даних безпілотних літальних апаратів. Математичні машини та системи. 2017. № 3. С. 154-161.

Розділ 3
ШЛЯХИ ЗАБЕЗПЕЧЕННЯ ІМІТОСТІЙКОСТІ КРИПТОЗАХИСТУ ТА ЦІЛІСНОСТІ ПРОГРАМНИХ РЕАЛІЗАЦІЙ ЗАСОБІВ КРИПТОЗАХИСТУ ІНФОРМАЦІЇ НА ПРИКЛАДІ ДИСТАНЦІЙНО ПІЛОТОВАНИХ ЛІТАЛЬНИХ АПАРАТІВ

Під дистанційно пілотованим літальним апаратом (ДПЛА) в дисертаційній роботі будемо розуміти квадрокоптери – літальні апарати з довільною кількістю несучих гвинтів, розміщених в одній площині, що обертаються діагонально в протилежних напрямках. Їх можлива конфігурація наведена в табл.3.1.

Таблиця 3.1

Конфігурації квадрокоптерів та схеми керування [1]

Конфігурація	зліт	зниження	обертання проворуч	обертання ліворуч	нахил вперед	нахил назад	нахил вправо	нахил вліво
	Оберти ABCD пропорційно збільшити	Оберти ABCD пропорційно зменшити	Оберти BD збільшити та AC пропорційно зменшити	Оберти AC збільшити та BD пропорційно зменшити	Оберти C збільшити та A пропорційно зменшити	Оберти A збільшити та C пропорційно зменшити	Оберти D збільшити та B пропорційно зменшити	Оберти B збільшити та D пропорційно зменшити
	Оберти ABCD пропорційно збільшити	Оберти ABCD пропорційно зменшити	Оберти BD збільшити та AC пропорційно зменшити	Оберти AC збільшити та BD пропорційно зменшити	Оберти CD збільшити та/або AB зменшити	Оберти AB збільшити та/або CD зменшити	Оберти AD збільшити та/або BC зменшити	Оберти BC збільшити та/або AD зменши

Квадрокоптер керується мікропроцесором («мозок» апарату), який перетворює сигнали з гіроскопів та передавача радіокерування в команди двигунам. Щоб забезпечити рівновагу в повітрі та стабільне зависання, квадрокоптер оснащений гіроскопами. Вони контролюють положення апарата по крену, тангажу та обертанню навколо вертикальної осі. Кожен гіроскоп миттєво реагує на мінімальне відхилення від своєї осі налаштування та подає сигнал на процесор, який в свою чергу, через регулятори ходу контролює оберти електродвигунів, що забезпечує рівновагу апарата в повітрі.

3.1 Особливості застосування ДПЛА для моніторингу периметру контрольованої зони

Передбачається, що впровадження квадрокоптерів та супутніх їм програмних рішень може суттєво підвищити ефективність забезпечення передусім фізичного захисту шляхом здійснення моніторингу території в межах периметру контрольованої зони [2] (рис.3.1).

Рис. 3.1. Використання квадрокоптера для обстежень

Літальний апарат може бути відправлений на «місію» – заздалегідь визначені й зафіксовані в програмному забезпеченні точки, що містять інформацію про координати, висоту, орієнтацію в просторі, кут направлення камери, та складають маршрут слідування квадрокоптера, що прокладається між ними. Після здійснення запуску у повітря, повітряне судно може бути одразу відправлене на виконання такої місії, тобто автоматично пролетить весь маршрут від однієї точки до наступної, аж до кінця. В цей час оператор може спостерігати за дроном та зображенням, що він передає у реальному часі через бездротовий зв’язок, а може, наприклад, здійснити запуск ще кількох квадрокоптерів на патрулювання інших ділянок, які вони виконуватимуть в автоматичному режимі. Відеозаписи кожного з таких польотів зберігаються для подальшого аналізу.

Завдяки тому, що місії заздалегідь записані і можуть виконуватись повторювано із певним часовим інтервалом в автоматизованому режимі, сам час, необхідний на

охоплення певної території значно менший, ніж при піших або моторизованих патрулях. Окрім того, точка зору згори стає доступна лише завдяки квадрокоптеру, а також буде залучена до моніторингу вся площа визначеної території, а не лише ділянки біля її периметру. Це значно економить робочу силу особового складу, що, в свою чергу, дозволяє забезпечити виконання інших завдань, або зменшити кількість персоналу, необхідного для фізичного захисту [3].

У випадку застосування квадрокоптерів алгоритм реагування на дії порушника полягатиме в такому [4-5]:

1) автоматична система, або оператор зафіксували факт або підозру на несанкціоноване перетинання периметру контрольованої зони;

2) мобільна група у складі щонайменше двох осіб, однією з яких є оператор ДПЛА, отримують вказівки щодо місця, напрямку перетинання меж КЗ, можливого радіусу пошуку та інших відомостей, що стосуються даного інциденту і є необхідними та корисними для прийняття рішення про пошук ймовірного порушника. Мобільна група висувається у визначену точку;

3) після прибуття до зони пошуків, оператор виконує розгортання апарату та здійснює його запуск у повітря. Керуючись інформацію, отриманою з повітряної перспективи, за використання засобів зв’язку, інший член групи може більш ефективно виконувати пошук ймовірного порушника, завдяки тому, що оператор ДПЛА може підтвердити відсутність порушника або слідів порушника за перешкодами, які інакше необхідно було би обходити, або витрачати час на пересування між об’єктами, тоді як квадрокоптер забезпечує широкий кут огляду і високу швидкість пересування найкоротшим маршрутом, без урахування наземних споруд та інших перешкод для руху;

4) після виявлення ймовірного порушника оператором дрону, виконується безперервний його супровід до затримання співробітниками, або до того моменту, як він покине межі контрольованої зони. Відеозапис польоту, виконаний на камеру дрону, може бути передано співробітникам внутрішньої служби безпеки або правоохоронних органів .

Автоматизований контроль периметру здійснюється шляхом штатного програмного забезпечення в режимі польоту за наперед визначеними точками, що складають маршрут. Точка містить відомості про положення апарату, його висоту, сам маршрут може бути подоланий із заданою швидкістю. Для такої задачі найбільш раціональними будуть два способи реалізації [6].

Перший – режим «вільної камери», в такому випадку оператору необхідно виконати наступні дії:

1) Підготувати безпілотний комплекс до запуску;
2) Виконати запуск дрону в ручному режимі;
3) Обрати маршрут та підтвердити виконання;
4) Зліт, політ за точками, та посадка буде виконано автоматично. Оператор може керувати камерою в двох площинах та ставити виконання прольоту на паузу, із можливістю подальшого відновлення.

Такий підхід максимально звільняє оператора від керування та дозволяє сконцентруватися на керуванні та аналізі зображення з камери.

Другий – автоматичний, в якому оператор має лише розмістити безпілотник на злітному майданчику, а всі наступні дії, включно з подальшою посадкою, відбуватимуться автоматично. Для такого сценарію, налаштування камери задаються попередньо, раціональною буде зйомка під кутом -90°, тобто строго вертикально вниз. Висота підбирається так, щоб відповідати вимогам зі створюваного квадрокоптером шуму, його видимості, та розмірам ділянки, що буде потрапляти до вихідного файлу.

Попри те, що оператор може стежити за перебігом польоту в режимі реального часу, такий підхід рекомендується для подальшого перегляду фото чи відеозапису польоту, обробки, аналізу, порівняння з попередніми польотами за тим же маршрутом. Крім того, такий політ підійде для сценаріїв, в яких довжина периметру дозволяє здійснити повний його обліт на одному заряді акумулятора, але відстань чи наявність перешкод для сигналу між пультом, на який відбувається трансляція зображення у реальному часі, та дроном спричиняють припинення з’єднання між ними.

Для того, щоб запускати літальний апарат на автоматизований політ за маршрутом, такий маршрут має бути попередньо записаний. Чинне програмне забезпечення допускає це зробити одним із двох способів [7-8] (рис.3.2):

«Віртуальний» – точки встановлюються у програмному забезпеченні, поверх мапи із територією.

«Запис маршруту» – у цьому випадку необхідно виконати проліт у ручному режимі, записуючи дані точки, такі як широта, довгота, висота відносно точки зльоту в тій позиції, де знаходиться дрон в момент запису такої точки.

Рис.3.2. Місія автоматичного польоту

Оскільки електронні мапи мають похибки у позиціонуванні, а також територія контрольованої зони може бути відсутня на супутникових знімках – фізичний проліт і запис точок «за фактом» є бажаним [9].

3.1.1 Рекомендації щодо забезпечення безпеки каналу та інформаційного обміну з ДПЛА від активних і пасивних атак

Разом із беззаперечними перевагами ДПЛА, викладеними у попередньому підрозділі, їм притаманна й низка недоліків. Один з них пов'язаний із забезпеченням безпеки каналу управління та/або каналу передавання трафіку. Особливо важливо це, наприклад, при намаганні порушника (сторонніх осіб) перехопити керування такими засобами (ДПЛА).

Відомо, що серед безлічі загроз безпеці інформації в безпілотних системах можна виділити низку найбільш небезпечних. Це: перехоплення в радіоканалі (контроль трафіку); вплив навмисних перешкод; несанкціоноване декодування і дешифрування інформації; інформаційне перевантаження за рахунок передачі великої кількості фрагментів неправдивої інформації; передачу неправдивої інформації, постановку імітуючих перешкод; фізичний вплив на кінцеві пристрої [10].

Всі перелічені вище атаки можна розділити на пасивні та на активні. Так, наприклад, пасивна атака на передачу даних з борту безпілотника може реалізовуватися по-різному, залежно від типу лінії передачі даних. Приміром, при передачі даних на частоті 2,4 ГГц (стандарт 802.11) для її реалізації необхідний програмний сніфер, що перехоплює трафік, а також мережева плата бездротового зв’язку з набором мікросхем Prism [11]. Іншим способом передачі даних є супутниковий канал зв’язку. Існує певне програмне забезпечення, що дозволяє перехоплювати трафік з супутника і зберігати його на персональний комп’ютер, приміром, продукт SkyGrabber, знову ж, при використанні відповідного апаратного забезпечення. Саме його, за твердженнями командування американської армії в Іраку, використовували повстанці для отримання зображення з розвідувальних ДПЛА [12].

Для передачі відеопотоку більш низької якості може використовуватися канал бездротового зв’язку на частоті 2.4 ГГц. При використанні даного виду зв’язку на канал можуть бути проведені як пасивні, так і активні атаки. Програмне забезпечення, що використовується для проведення такого роду атак, також

широко поширене в Інтернеті. Приміром, може використовуватися сніфер Wireshark, що дозволяє перехоплювати пакети, передані по бездротовій мережі, а також зберігати їх на персональний комп'ютер для подальшої обробки. Одним з плюсів даної програми є можливість провести атаку на зашифрований бездротовий канал зв'язку [13].

Перше, що можна зробити для нівелювання кібератак на канали управління квадрокоптерами – спробувати виявити джерело перешкод. Це можна зробити методом тріангуляції, тобто шляхом заміру рівня сигналу в декількох точках. На підставі отриманих даних можна спробувати визначити місце знаходження придушуючого пристрою та його знешкодити [14]. У випадку, що розглядається в дисертаційному дооослідженні, це може дати позитивний результат. В інших випадках, наприклад, при застосуванні ДПЛА в районі АТО – це не дасть позитивного результату.

3.1.2 Рекомендації щодо забезпечення захисту інформації від несанкціонованого доступу до її смислового змісту

Враховуючи просторову доступність радіоканалів управління та інформаційного обміну з ДПЛА, значну увагу необхідно приділяти питанням захисту інформації від несанкціонованого доступу до її смислового змісту, наприклад, при намаганні підмінити відео в процесі його передавання до центру управління із застосування кібератаки «men of the middle» [15]. Основним інструментом захисту в цьому випадку є криптографічний захист, побудований на основі різних алгоритмів шифрування. Крім того, при здійсненні передачі конфіденційної інформації по зв'язку ДПЛА принципово необхідно виконання процедури аутентифікації – підтвердження дійсності абонента. У безпілотних системах доцільніше використовувати асиметричні криптосистеми. Асиметричні криптосистеми дозволяють будувати ефективні алгоритми аутентифікації. Використання для захисту інформації тільки симетричних криптосистем вимагає поширення великої кількості ключової інформації, а асиметричні криптосистеми вільні від даного недоліку [16].

Серед представлених на ринку квадрокоптерів, зокрема, розглянутих у рамках цього розділу, поширені системи зв’язку Lightbridge 2 [17] та OcuSync 2.0 [18].

Найновіші літальні апарати використовують OcuSync другого покоління та OcuSync Enterprise, в якій використовується шифрування AES-256, але в наявності досі багато моделей із старшою системою Lightbridge 2 (DJI Matrice 200, Inspire 2), де шифрування сигналу реалізовано за допомогою 3DES.

AES (Advanced Encryption Standard) і 3DES, або також відомий як Triple DES (Data Encryption Standard), є двома чинними стандартами шифрування даних. Хоча AES – це абсолютно нове шифрування, яке використовує SP-мережу (мережу замін-перестановок), 3DES – це лише адаптація до старого шифрування DES, стандарт якого був офіційно визнаний неактуальним для використання у державному секторі через знижений показник надійності в 1981 р., порівняно з 1976 роком, що спиралася на збалансовану мережу Фейстеля. Технічно, 3DES – це просто DES, який тричі застосовується до інформації, що шифрується [19].

AES використовує три загальні довжини ключа шифрування, 128, 192 та 256 біт. Що стосується 3DES, ключ шифрування все ще обмежений 56 бітами, як це продиктовано стандартом DES. Але оскільки він застосовується три рази, реалізатор може вибрати 3 однакових 56-бітові ключі, або 2 однакові та 1 дискретний, або навіть три однакові ключі. Це означає, що 3DES може мати довжину ключів шифрування 168, 112 або 56-бітний ключ шифрування відповідно. Але через втрату ефективності при повторному застосуванні одного і того ж шифрування тричі, використання 168 біт має знижений рівень безпеки, еквівалентний 112 бітам, а використання 112 біт має знижену безпеку, еквівалентну 80 бітам [20].

3DES також використовує однакову довжину блоку в 64 біти, наполовину меншу, ніж AES, де це значення складає 128 біт. Використання AES забезпечує додаткове страхування, що важче аналізувати дані трафіку щодо витоку з однакових блоків. Під час використання 3DES користувачеві необхідно змінювати ключі шифрування кожні 32 ГБ переданих даних, щоб мінімізувати можливість витоків; ідентично до використання стандартного шифрування DES.

Зрештою, повторення одного і того ж процесу тричі потребує певного часу. Завдяки тому, що AES набагато швидше порівняно з 3DES. Ця відмінність зменшується, якщо до процесу включити програмне, апаратне та складність розробки апаратних засобів. Отже, якщо наявне апаратне забезпечення, що прискорює 3DES, перехід на AES, реалізований лише програмним забезпеченням, може призвести до меншої переваги у часі, або навіть уповільнення часу обробки. У цьому аспекті немає кращого рішення, ніж тестування кожного та вимірювання їх швидкості. Але якщо мова йде про безпеку, AES – це впевнений переможець, оскільки він все ще вважається непорушним у практичному використанні [21-22].

Підсумок: 3DES використовує ідентичне шифрування DES, тоді як AES використовує абсолютно інше; 3DES має більш короткі та слабкі ключі шифрування порівняно з AES; 3DES використовує повторювані ключі шифрування, тоді як AES цього не робить; 3DES також використовує меншу довжину блоку порівняно з AES.

Шифрування 3DES займає більше часу, ніж шифрування AES.

3.1.3 Рекомендації щодо організації віддаленого доступу до ДПЛА

Разом з тим, впровадження процедур шифрування трафіку шляхом встановлення додаткового шифрувального обладнання, в свою чергу обтяжує квадрокоптер й може привести до несвоєчасного отримання рим керуючого сигналу. Як результат, це може привести до того, що квадрокоптер буде втрачено через зіткнення з перешкодою. Зважаючи на таке необхідно настільки спростити супроводжувані обчислювальними ресурсами квадрокоптера методи шифрування, щоб ефективність забезпечення конфіденційності та цілісності повідомлень на зменшилась. Саме вирішенню цього завдання й були присвячені дослідження, описані в другому розділі дисертаційної роботи. Результатом цього стало моделі функціонування шифратора БАЗ в ДПЛА, які вирішують завдання захисту фізичного периметру СОІ, при одночасному забезпеченні припустимого рівня криптографічної стійкості.

В таблиці 3.2 наведено результати обчислення часу на виконання задач кодування/декодування інформації, отримані при застосуванні створеної швидкодіючої криптосхеми [23].

Таблиця 3.2

Середній час тестів для криптографічних перетворень

Вид тестування	Результати
Швидкість виконання шифрування (сек)	0,215 с
Швидкість виконання дешифрування (сек)	0,219 с
Частотність (%)	13%
Актуальність переданої інформації	58 с
Криптостійкість	6 днів

3.2 Дослідження розроблених методів і моделей на макеті моделюючого комплексу інформаційної технології криптографічної обробки інформації

З метою практичної перевірки моделей і методів, розроблених в ході дисертаційного дослідження, а саме якості методу генерації потоку підстановок з використанням шифру БАЗ для забезпечення імітостійкого шифрування в СОІ та модель функціонування шифратора БАЗ на мові Java було створено макет моделюючого комплексу інформаційної технології криптографічної обробки інформації (МК ІТ КОІ) [24]. Логічну схему макету наведено рис. 3.3.

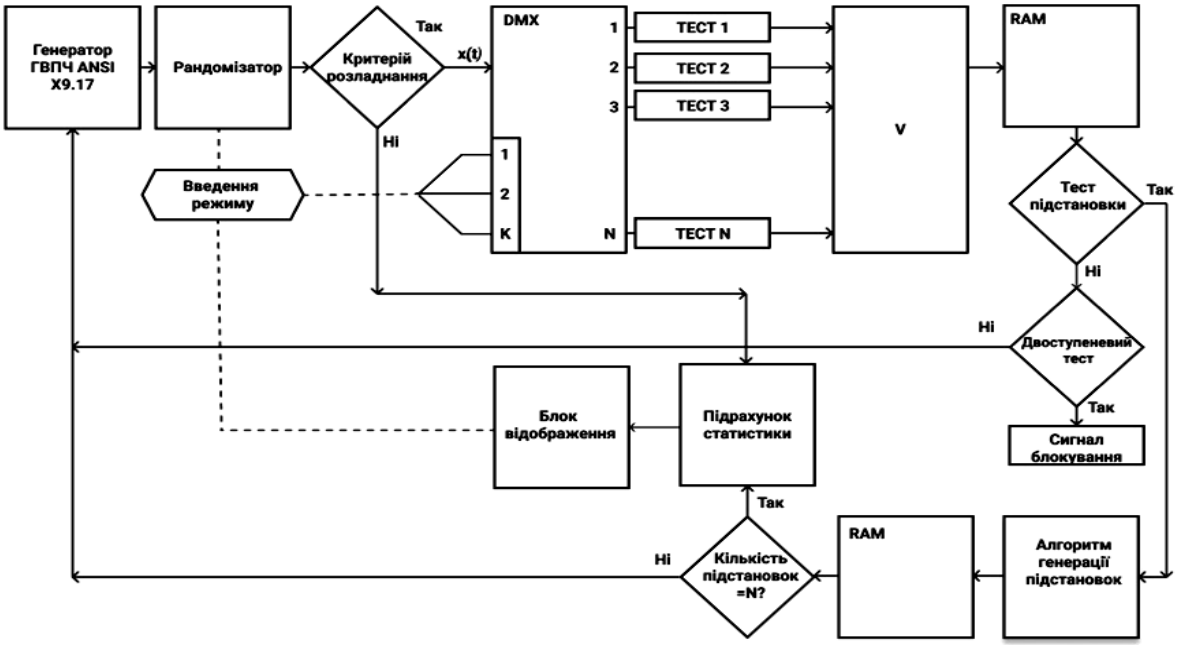


Рис. 3.3. Логічна схема макету моделюючого комплексу інформаційної технології криптографічної обробки інформації в СОІ

Комплекс забезпечує генерацію двійкових випадкових даних, їх рандомізацію відповідно до заданого режиму, вибір схеми тестування, генерацію заданої кількості підстановок визначеного розміру та відображення отриманих результатів за заданою діаграмою. Панель управління зазначеного комплексу, яку наведено на рис. 3.4, відображає заданий режим генерації кількості підстановок N = 5000, розміру m = 16 з двійкової послідовності з імовірністю зустрічаємості P(1) = 0,5001. Для перевірки використано критерій Пірсона χ^2, вид діаграми - кільцева.

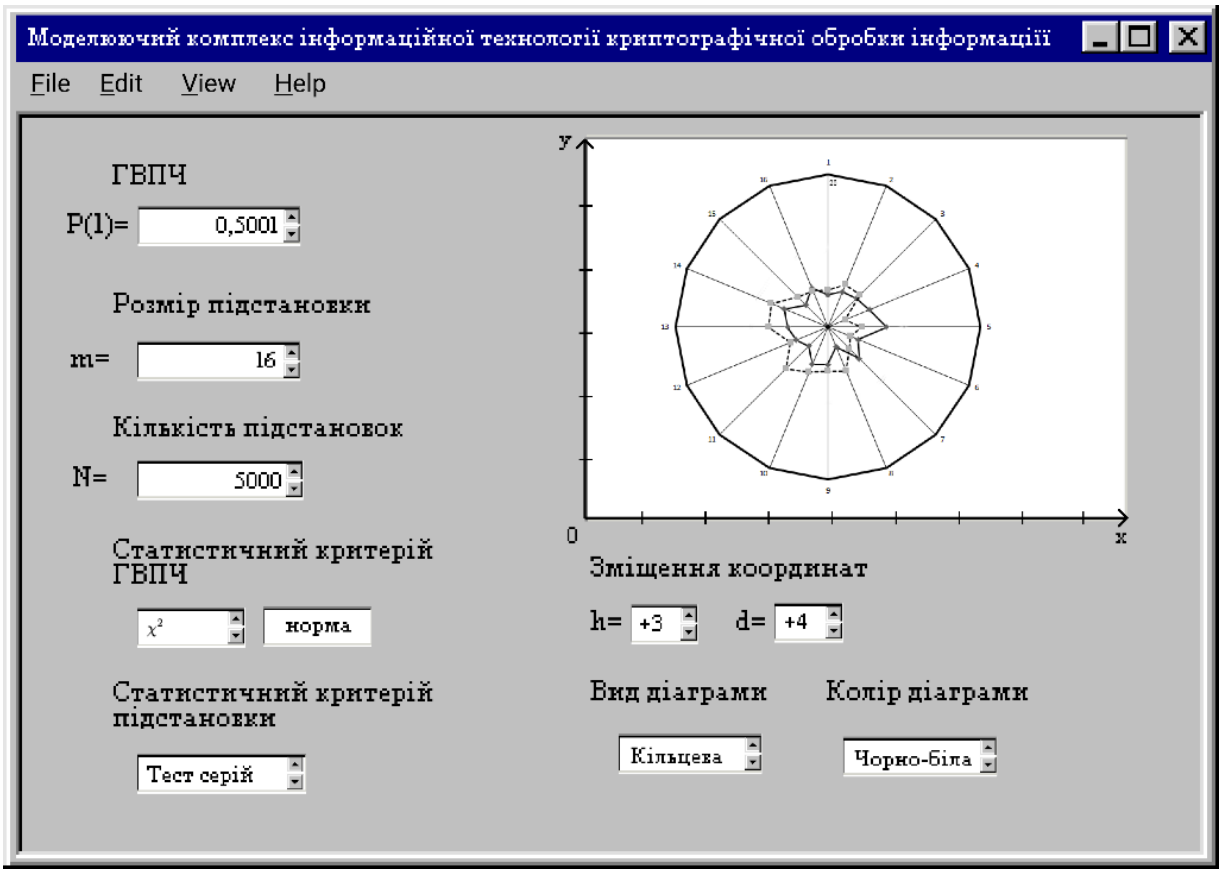


Рис. 3.4. Скриншот панелі управління моделюючого комплексу інформаційної технології криптографічної обробки інформації в СОІ

3.2.1. База статистичних критеріїв моделюючого комплексу

Для побудови моделюючого комплексу і оцінки ймовірнісних властивостей випадкових даних від генератора псевдовипадкових даних (ГПВД) та генератора

підстановок були використані модифіковані статистичні тести з переліку рекомендованих NIST для відповідних досліджень [25] для випадків довільної довжини вихідної послідовності та ймовірності одиниці.

1. Зокрема, для тестування всіх бітових послідовностей у якості першого був застосований ***частотний монобітовий тест***, який у загальному випадку використовується для перевірки простої гіпотези $H_0 : P\left(x(t)=1\right)= p_1$, проти складної альтернативи $H_1 : P\left(x(t)=1\right)\neq p_1$.

З урахуванням теореми Муавра-Лапласа [26], для достатньо великих M та вірної гіпотези H_0 має місце наближення:

$$P\left(\left|\frac{\sum_{t=1}^{M} x(t)-Mp_1}{\sqrt{Mp_1(1-p_1)}}\right| < t_\alpha\right) \approx \frac{1}{\sqrt{2\pi}}\int_{-t_\alpha}^{t_\alpha} e^{-\frac{u^2}{2}}\,du \qquad (3.1)$$

де t_α - квантіль стандартного нормального розподілу ймовірностей $N(0,1)$ з рівнем надійності α, M – кількість біт в послідовності. Тоді з (1) отримуємо, що з рівнем надійності α у випадку гіпотези H_0 виконується нерівність:

$$-\left(Mp_1 + t_\alpha\sqrt{Mp_1(1-p_1)}\right) < \sum x(t) < \left(Mp_1 + t_\alpha\sqrt{Mp_1(1-p_1)}\right).$$

2. Для двійкових послідовностей у разі проходження монобітового тесту виконувався ***тест покеру*** для якого підраховані у вихідній послідовності кількості $\{\delta_{00},\delta_{01},\delta_{10},\delta_{11}\}, де \sum\delta_i = M$ комбінацій виду $\{00,01,10,11\}$. При цьому підрахунок здійснюється ланцюжком.

Ймовірності появи різних комбінацій розраховуються наступним чином:

$$\begin{cases} \pi_0 = P(00) = (1-p_1)(1-p_1) \\ \pi_1 = P(01) = (1-p_1)\,p_1 \\ \pi_2 = P(10) = p_1(1-p_1) \\ \pi_3 = P(11) = p_1^2 \end{cases}.$$

Далі застосовувався критерій Пірсона [26] а саме, для гіпотези H_0 з рівнем надійності α має місце нерівність:

$$\sum_{i=0}^{3}\frac{(\delta_i - \pi_i M)^2}{\pi_i M} < \chi^2_{\alpha,3},$$

де $\chi^2_{\alpha,4}$ – квантіль розподілу Пірсона з 3-ма степенями свободи і рівнем надійності α.

3. Критерій Пірсона також був використаний для тестування розподілу блоків біт довжини $2^k = m > 2,$ а також оцінки якості генерованих підстановок довжини m наступним чином.

Нехай вектор $\bar{\delta} = (\delta_0, \delta_1, \ldots, \delta_{m-1})$ визначає частоти зустрічаємості бітових блоків або кількість отриманих варіантів переходу підстановках деякого символу з множини $\{1, 2, \ldots, m\}$ у відповідний символ з цієї ж множини. При цьому, має місце представлення $0_{10} \leftrightarrow (0, \ldots, 0, 0)_2, 1_{10} \leftrightarrow (0, ..., 0, 1)_2, 2_{10} \leftrightarrow (0, ..., 1, 0), \ldots$ тощо. Тоді для рівномірного розподілу повинна виконуватись нерівність:

$$\sum_{i=0}^{m-1} \frac{\left(\delta_i - m^{-1}M\right)^2}{m^{-1}M} < \chi^2_{\alpha,(m-1)},$$

де $\chi^2_{\alpha,(m-1)}$ – квантіль розподілу Пірсона з $(m-1)$ степенями свободи і рівнем надійності α.

Інші статистичні критерії застосовувались згідно з методологією, що запропонована в [27].

Також варто зазначити, що наступні тести теж можливі для використання у ході роботи моделюючого комплексу інформаційної технології криптографічної обробки інформації в СОІ [28]:

1. Частотний побітовий тест

Суть даного тесту полягає у визначенні співвідношення між нулями і одиницями у всій двійковій послідовності. Мета – з'ясувати, чи дійсно число нулів і одиниць в послідовності приблизно однакові, як це можна було б припустити в разі істинно випадкової бінарної послідовності. Тест оцінює, наскільки близька частка одиниць до 0,5. Таким чином, число нулів і одиниць має бути приблизно однаковим. Якщо обчислене в ході тесту значення ймовірності p <0,01, то дана двійкова послідовність не є істинно випадковою. В іншому випадку послідовність

носить випадковий характер. Варто відзначити, що всі наступні тести проводяться за умови, що пройдено даний тест.

2. Частотний блоковий тест

Суть тесту – визначення частки одиниць всередині блоку довжиною m біт. Мета – з'ясувати, чи дійсно частота повторення одиниць в блоці довжиною m біт приблизно дорівнює $m/2$, як можна було б припустити в разі абсолютно випадкової послідовності. Обчислення в ході тесту значення ймовірності p повинно бути не менше 0,01. В іншому випадку (p <0,01) двійкова послідовність не носить істинно випадковий характер. Якщо прийняти m = 1, даний тест переходить в тест № 1 (частотний побітовий тест).

3. Тест на послідовність однакових бітів

Суть полягає в підрахунку повного числа рядів у вихідній послідовності, де під словом «ряд» мається на увазі безперервна підпослідовність однакових бітів. Ряд довжиною k біт складається з k абсолютно ідентичних бітів, починається і закінчується з біта, що містить протилежне значення. Мета даного тесту – зробити висновок про те, чи дійсно кількість рядів, що складаються з одиниць і нулів з різними довжинами, відповідає їх кількості в випадковій послідовності. Зокрема, визначається швидко або повільно чергуються одиниці і нулі в вихідній послідовності. Якщо обчислене в ході тесту значення ймовірності p <0,01, то дана двійкова послідовність не є істинно випадковою. В іншому випадку вона носить випадковий характер.

4. Тест на найдовшу послідовність одиниць в блоці

В даному тесті визначається найдовший ряд одиниць всередині блоку довжиною m біт. Мета – з'ясувати, чи дійсно довжина такого ряду відповідає очікуванням довжини найдовшим ряду одиниць в разі з абсолютно випадковою послідовністю. Якщо вирахувані в ході тесту значення ймовірності p <0,01, вважається, що вихідна послідовність не є випадковою. В іншому випадку робиться висновок про її випадковість. Слід зауважити, що з припущення про приблизно однакову частоту появи одиниць і нулів (тест № 1) випливає, що точно такі ж

результати даного тесту будуть отримані при розгляді найдовшого ряду нулів. Тому вимірювання можна проводити тільки з одиницями.

5. Тест рангів бінарних матриць

Тут проводиться розрахунок рангів непересічних підматриць, побудованих з вихідної двійкової послідовності. Метою цього тесту є перевірка на лінійну залежність підрядків фіксованої довжини, що становлять первісну послідовність. У разі, якщо обчислене в ході тесту значення ймовірності p <0,01, робиться висновок про невипадковий характер вхідної послідовності біт. В іншому випадку вважаємо її абсолютно випадковою. Даний тест так само присутній в пакеті DIEHARD.

6. Спектральний тест

Суть тесту полягає в оцінці висоти піків дискретного перетворення Фур'є вихідної послідовності. Мета – виявлення періодичних властивостей вхідної послідовності, наприклад, близько розташованих один до одного повторюваних ділянок. Тим самим це явно демонструє відхилення від випадкового характеру досліджуваної послідовності. Ідея полягає в тому, щоб число піків, що перевищують граничне значення в 95% по амплітуді, було значно більше 5%. Якщо обчислене в ході тесту значення ймовірності p <0,01, то дана двійкова послідовність не є абсолютно випадковою. В іншому випадку вона носить випадковий характер.

7. Тест на збіг шаблонів, які не перекриваються

В даному тесті підраховується кількість заздалегідь визначених шаблонів, знайдених у вихідній послідовності. Мета – виявити генератори випадкових або псевдовипадкових чисел, що формують занадто часто задані неперіодичні шаблони. Як і в тесті № 8 на збіг перекриття шаблонів для пошуку конкретних шаблонів довжиною m біт, використовується вікно також довжиною m біт. Якщо шаблон не виявлений, вікно зміщується на один біт. *Якщо ж шаблон знайдений, вікно переміщається на біт, наступний за знайденим шаблоном, і пошук триває далі.* Обчислення в ході тесту значення ймовірності p повинно бути не менше 0,01. В іншому випадку (p <0,01), двійкова послідовність не є абсолютно випадковою.

8. Тест на збіг шаблонів, які перекриваються

Суть даного тесту полягає в підрахунку кількості заздалегідь визначених шаблонів, знайдених у вихідній послідовності. Як і в тесті № 7 «збіг шаблонів, які не перекриваються», для пошуку конкретних шаблонів довжиною *m* біт використовується вікно також довжиною *m* біт. Сам пошук здійснюється аналогічним чином. Якщо шаблон не виявлений, вікно зміщується на один біт. Різниця між цим тестом і тестом № 7 полягає лише в тому, що якщо шаблон знайдений, вікно переміщається тільки на біт вперед, після чого пошук триває далі. Обчислення в ході тесту значення ймовірності p повинно бути не менше 0,01. В іншому випадку (p <0,01), двійкова послідовність не є абсолютно випадковою.

9. Універсальний статистичний тест Маурера

Тут визначається число біт між однаковими шаблонами у вихідній послідовності (міра, що має безпосереднє відношення до довжини стислої послідовності). Мета тесту – з'ясувати, чи може дана послідовність бути значно стиснута без втрат інформації. У разі, якщо це можливо зробити, то вона не є істинно випадковою. В ході тесту обчислюється значення ймовірності p. Якщо p <0,01, то вважається, що вихідна послідовність не є випадковою. В іншому випадку робиться висновок про її випадковість.

10. Тест на лінійну складність

В основі тесту лежить принцип роботи лінійного регістра зсуву зі зворотним зв'язком (англ. Linear Feedback Shift Register, LFSR). Мета – з'ясувати, чи є вхідна послідовність досить складною для того, щоб вважатися абсолютно випадковою. Абсолютно випадкові послідовності характеризуються довгими лінійними регістрами зсуву зі зворотним зв'язком. Якщо ж такий регістр занадто короткий, то передбачається, що послідовність не є в повній мірі випадковою. В ході тесту обчислюється значення ймовірності p. Якщо p <0,01, то вважається, що вихідна послідовність не є випадковою. В іншому випадку робиться висновок про її випадковість.

11. Тест на періодичність

Даний тест полягає в підрахунку частоти всіх можливих перекриттів шаблонів довжини m біт протягом вихідної послідовності бітів. Метою є визначення, чи дійсно кількість появ 2^m перекриваючихся шаблонів довжиною m біт, приблизно така ж, як у випадку абсолютно випадкової вхідної послідовності біт. Остання, як відомо, має одноманітність, тобто кожен шаблон довжиною m біт з'являється в послідовності з однаковою ймовірністю. Якщо обчислене в ході тесту значення ймовірності p <0,01, то дана двійкова послідовність не є абсолютно випадковою. В іншому випадку, вона носить випадковий характер. Варто відзначити, що при m = 1 тест на періодичність переходить в частотний побітовий тест (№ 1).

12. Тест приблизної ентропії

Як і в тесті на періодичність, в даному тесті акцент робиться на підрахунку частоти всіх можливих перекриттів шаблонів довжини m біт протягом вихідної послідовності бітів. Мета тесту – порівняти частоти перекривання двох послідовних блоків вихідної послідовності з довжинами m і m + 1 з частотами перекривання аналогічних блоків в абсолютно випадковій послідовності. Обчислюване в ході тесту значення ймовірності p повинно бути не менше 0,01. В іншому випадку (p <0,01), двійкова послідовність не є абсолютно випадковою.

13. Тест кумулятивних сум

Тест полягає в максимальному відхиленні (від нуля) при довільному обході, що визначаються кумулятивною сумою заданих (-1, +1) цифр в послідовності. Мета даного тесту – визначити, чи є кумулятивна сума часткових послідовностей, які виникають у вхідній послідовності, занадто великою або надто маленькою в порівнянні з очікуваною поведінкою такої суми для абсолютно випадкової вхідної послідовності. Таким чином, кумулятивна сума може розглядатися як довільний обхід. Для випадкової послідовності відхилення від свавільного обходу мають бути поблизу нуля. Для деяких типів послідовностей, які не є в повній мірі випадковими, подібні відхилення від нуля при довільному обході будуть досить істотними. Якщо обчислене в ході тесту значення

ймовірності p <0,01, то вхідна двійкова послідовність не є абсолютно випадковою. В іншому випадку вона носить випадковий характер.

14. Тест на довільні відхилення

Суть даного тесту полягає в підрахунку числа циклів, що мають строго k відвідувань при довільному обході кумулятивної суми. Довільний обхід кумулятивної суми починається з часткових сум після послідовності (0,1), перекладеної у відповідну послідовність (-1, +1). Цикл довільного обходу складається з серії кроків одиничної довжини, що здійснюються у випадковому порядку. Крім того, такий обхід починається і закінчується на одному й тому ж елементі. Мета даного тесту – визначити, чи відрізняється число відвідувань певного стану всередині циклу від аналогічного числа в разі абсолютно випадкової вхідної послідовності. Фактично даний тест є набором, що складається з восьми тестів, проведених для кожного з восьми станів циклу: -4, -3, -2, -1 і +1, +2, +3, +4. У кожному такому тесті приймається рішення про ступінь випадковості вихідної послідовності у відповідності з наступним правилом: якщо обчислене в ході тесту значення ймовірності p <0,01, то вхідна двійкова послідовність не є абсолютно випадковою. В іншому випадку вона носить випадковий характер.

15. Інший тест на довільні відхилення

У цьому тесті підраховується загальна кількість відвідувань певного стану при довільному обході кумулятивної суми. Метою є визначення відхилень від очікуваного числа відвідувань різних станів при довільному обході. Насправді цей тест складається з 18 тестів, проведених для кожного стану: -9, -8, ..., -1 і +1, +2, ..., +9. На кожному етапі робиться висновок про випадковість вхідної послідовності. Якщо обчислене в ході тесту значення ймовірності p <0,01, то вхідна двійкова послідовність не є абсолютно випадковою. В іншому випадку вона носить випадковий характер.

3.2.2 Перевірка якостей генератора підстановок БАЗ

Під час дослідження якостей генератора підстановок БАЗ були досліджені наступні завдання, що суттєво впливають на криптографічні якості шифратора БАЗ, а саме:

1. Як впливає зміна розподілу частот зустрічаємості знаків вхідної двійкової послідовності на розподіл частот переходів у підстановках?
2. Чи змінюється розподіл частот переходів у підстановках залежно від кількості генерованих підстановок, якщо він наближається до рівномірного розподілу?
3. Чи змінюється розподіл знаків залежно від місця у підстановці?

На рис. 3.5–3.15 наведені деякі результати досліджень за першим питанням, у підсумку якого можливо стверджувати, що навіть у випадку певного відхилення розподілу вхідної двійкової послідовності рівномірного розподілу ($P(1) = 0.5$, $0.50098, 0.50195, 0.50292, 0.50391$, $N = 10000$, $m = 16$), розподіл частот переходів у підстановках практично узгоджується з гіпотезою про їх рівномірний розподіл. Кроки зміни ймовірності одиниці обумовлені обраною схемою рандомізатора в МК ІТКП.

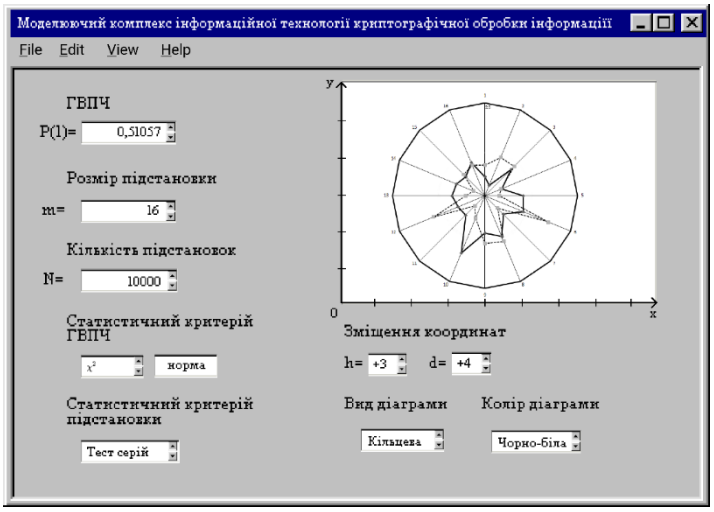


Рис. 3.5. Розподіл значень переходів послідовності підстановок $N = 10000,\quad m = 16,\quad p(1) = 0.50391$

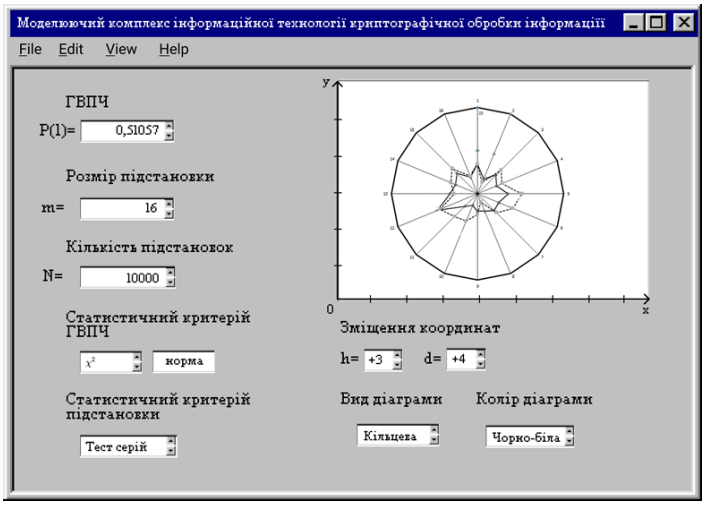


Рис. 3.6. Розподіл значень переходів послідовності підстановок $N = 10000,\quad m = 16,\quad p(1) = 0.5029$

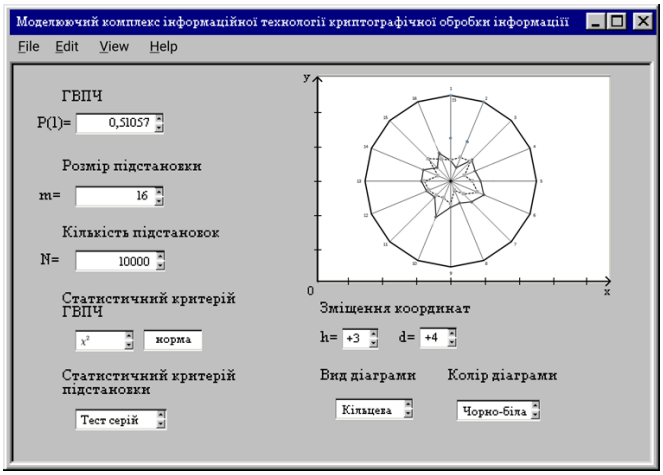


Рис. 3.7. Розподіл значень переходів послідовності підстановок $N = 10000, \quad m = 16, \quad p(1) = 0.50195$

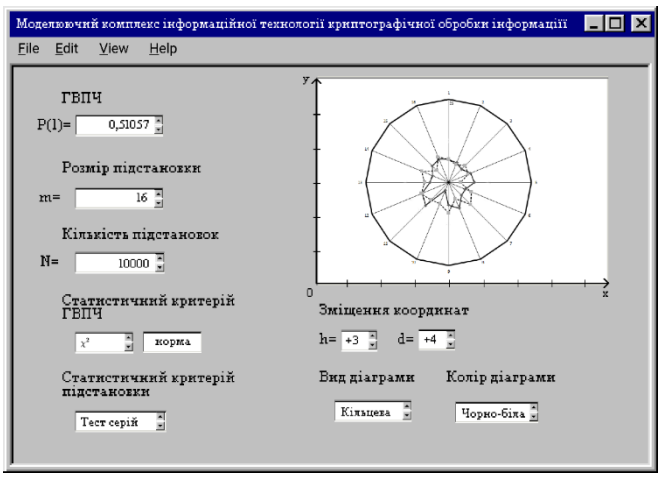


Рис. 3.8. Розподіл значень переходів послідовності підстановок $N = 10000, \quad m = 16, \quad p(1) = 0.50098$

Рис. 3.9. Розподіл значень переходів послідовності підстановок $N = 10000, \quad m = 16, \quad p(1) = 0.50$

Рис. 3.10. Розподіл значень переходів послідовності підстановок
$N = 10000,\ m = 16,\ p(1) = 0.50$

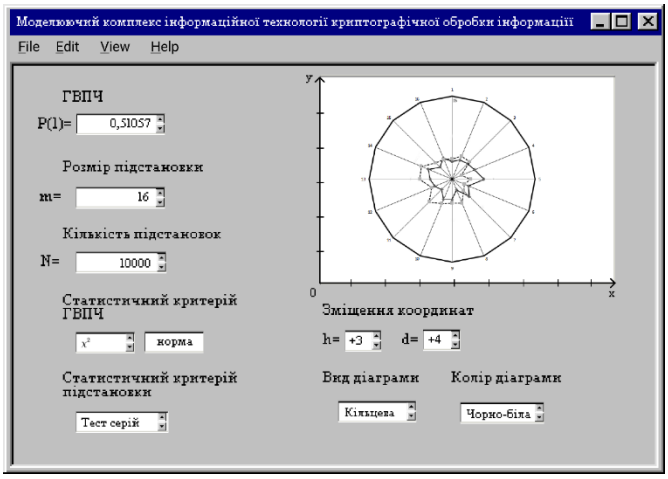


Рис. 3.11. Розподіл значень переходів послідовності підстановок
$N = 10000, \quad m = 16, \quad p(1) = 0.5023$

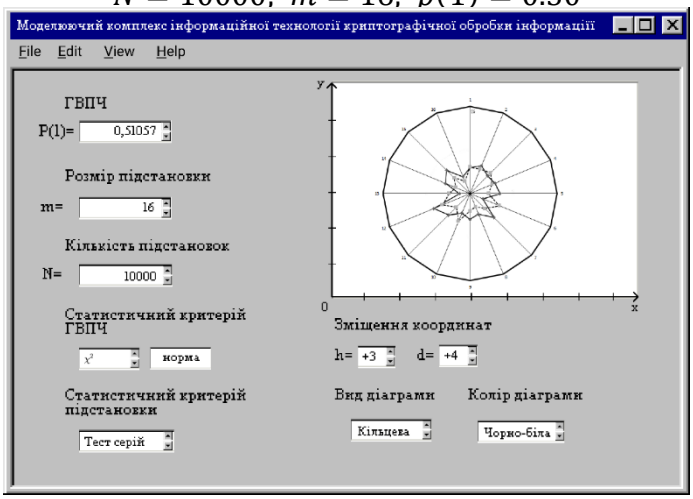


Рис. 3.12. Розподіл значень переходів послідовності підстановок
$N = 10000,\ m = 16,\ p(1) = 0.5023$

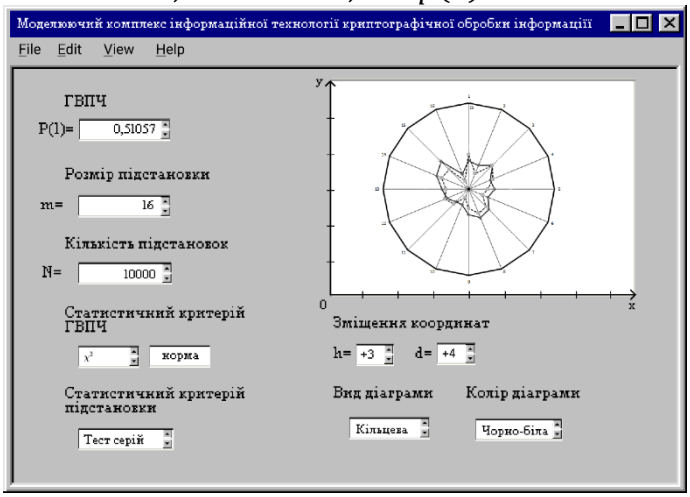


Рис. 3.13. Розподіл значень переходів послідовності підстановок
$N = 10000, \quad m = 16, \quad p(1) = 0.5023$

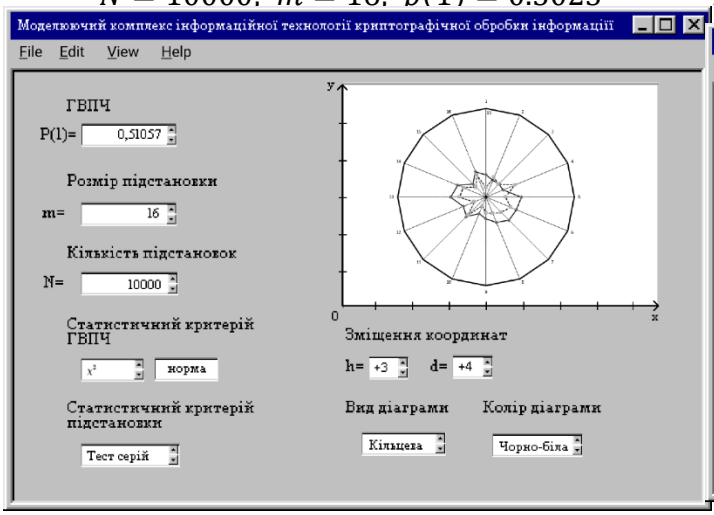


Рис. 3.14. Розподіл значень переходів послідовності підстановок
$N = 10000, \quad m = 16, \quad p(1) = 0.5023$

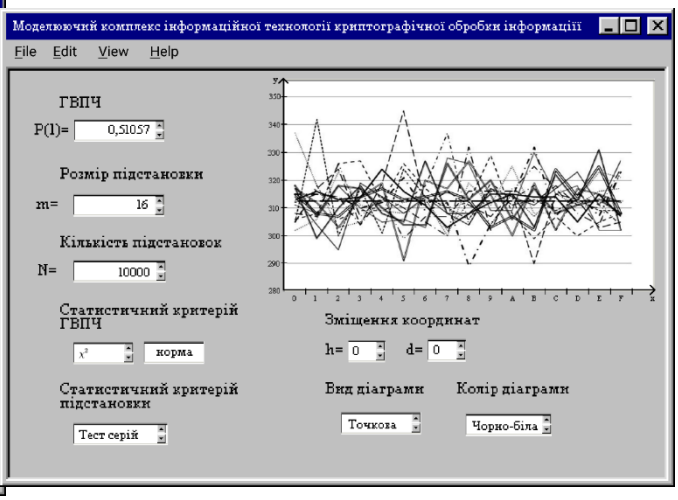


Рис. 3.15. Розподіл значень переходів у аналогу матриці перехідних ймовірностей шифру БАЗ
$N = 10000, \quad m = 16, \quad p(1) = 0.5023$

3.2.3 Перевірка випадковості та рівномірності розподілу потоку підстановок

Одним із напрямів застосування МК ІТ КОІ було проведено перевірку випадковості та рівномірності розподілу потоку підстановок, які утворювалися за допомогою генератора псевдовипадкових послідовностей на основі сучасних блокових криптоалгоритмів (ДСТУ 7624-2014, AES) [29]. Зокрема за формулою 6 було сформовано статистичний аналог матриці перехідних ймовірностей $\mathcal{P}$ перехідних ймовірностей для вузла накладання шифру [30] (табл.3.3):

Таблиця 3.3

Статистичний аналог матриці перехідних ймовірностей

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	315	308	307	314	317	313	314	316	313	303	307	302	324	314	325	308
1	313	316	313	314	324	316	311	303	308	312	315	314	314	305	315	307
2	317	312	303	314	319	291	320	317	327	315	300	319	302	317	315	312
3	318	307	318	317	310	304	304	326	313	320	306	317	308	311	313	308
4	315	315	314	310	312	321	309	312	289	304	314	332	307	316	307	323
5	306	308	326	327	312	326	317	310	312	329	308	299	307	300	305	308
6	302	306	323	311	309	308	313	300	319	311	325	308	308	313	323	321
7	305	342	300	317	301	324	314	310	306	317	311	313	309	313	309	309
8	311	308	318	310	305	306	303	328	326	309	309	330	315	318	302	302
9	305	317	324	304	319	345	311	316	311	307	315	290	315	313	303	305
10	308	307	326	307	314	299	307	307	316	316	316	325	317	308	304	323
11	312	315	314	304	314	310	305	300	332	302	317	312	318	311	310	324
12	305	318	306	309	309	312	318	337	304	319	313	318	302	315	311	304
13	337	318	303	305	313	312	313	302	312	304	315	307	315	311	316	317
14	314	304	295	319	314	308	314	308	304	317	314	311	317	323	311	327
15	317	299	310	318	308	305	327	308	308	315	315	303	322	312	331	302

Враховуючи складний аналітичний вираз для обчислення цієї матриці, було поставлено числовий експеримент, за допомогою якого було з’ясовано що, якщо на вході алгоритму генерації підстановок застосована РРВП, то матриця перехідних ймовірностей наближається до рівноймовірної [31]. У таблиці 3.3 наведено значення частот елементів статистичного аналога матриці для обсягу вхідних

даних $N = 5 \cdot 10^3$ для $n = 2^4, \delta = 3$ (для суттєво більшого обсягу даних зменшується наочність подання матриці внаслідок її вельми великого розміру).

Розподіл зустрічаємості частот у стовпчиках та рядках таблиці перевірявся за допомогою критерію Пірсона χ^2. Зокрема, для наведених у таблиці 3.4 даних, значення статистики Пірсона щодо рівномірного розподілу частот у рядках становлять:

Таблиця 3.4

Значення статистики Пірсона згоди для рядків статистичного аналога матриці $\mathcal{P}$

	Номер рядка															
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
зχ^2	2,04	1,2	3,99	1,94	4,15	4,49	2,81	4,73	4,03	6,69	2,96	3,15	3,58	3,32	2,91	3,88

Графіки перехідних ймовірностей наведено на рис.3.16.

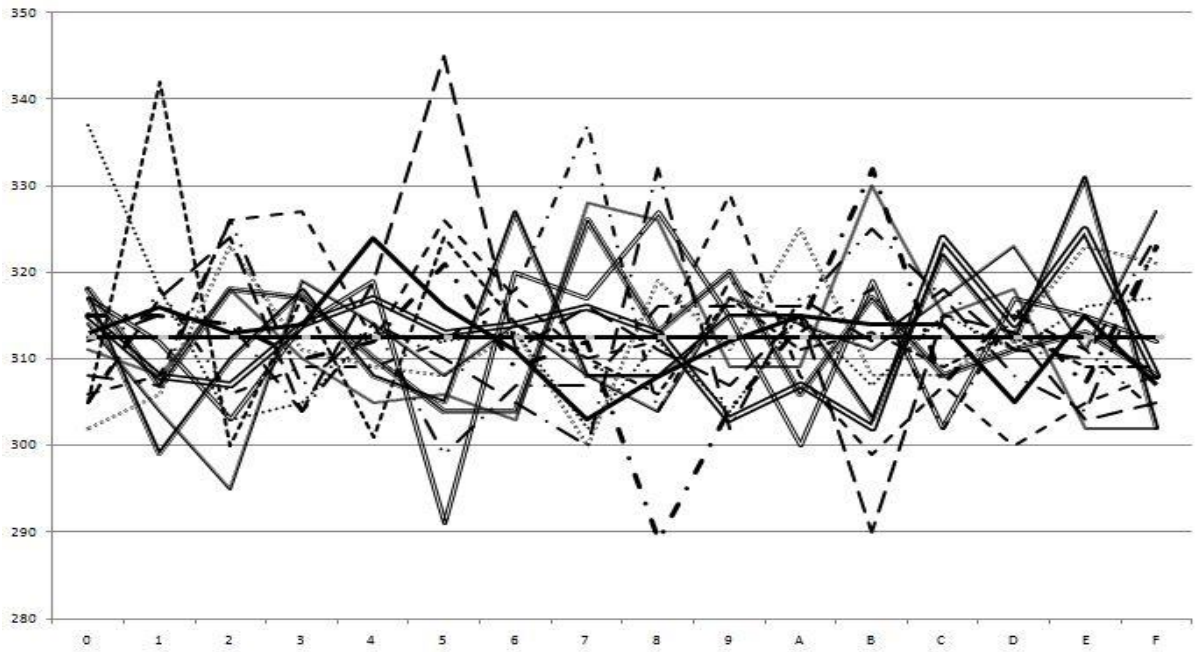


Рис. 3.16. Графік перехідних ймовірностей.

Порівнюючи наведені у таблиці 3.4 значення з квантілем статистики Пірсона з 15 степенями свободи з рівнем значущості $\alpha = 0{,}05$, $\chi^2_{15,0.05} = 25{,}0$ можливо зробити висновок, що це добре узгоджується з гіпотезою про рівномірний розподіл частот [31] у статистичному аналозі матриці табл.3.3.

На другому етапі було проведено перевірку методу виявлення атак на програмні реалізації засобів криптографічного захисту інформації [32], який дозволив сформувати двоступеневий критерій виявлення аномалій в СОІ та подолати прив'язку статистичних методів до моделі звичайної поведінки користувачів.

Провівши імітаційне моделювання процесу нападу на програмну реалізацію засобу КЗІ (рис. 3.4), можна зробити висновок, що модель підтвердила свою дієздатність, оскільки час виявлення атаки досить низький, що в свою чергу дозволить оперативно прийняти рішення щодо подальших дій системи. Так, в масштабі часу, наближеному до реального, застосування даного методу дозволить на 20% зменшити час на виявлення атаки.

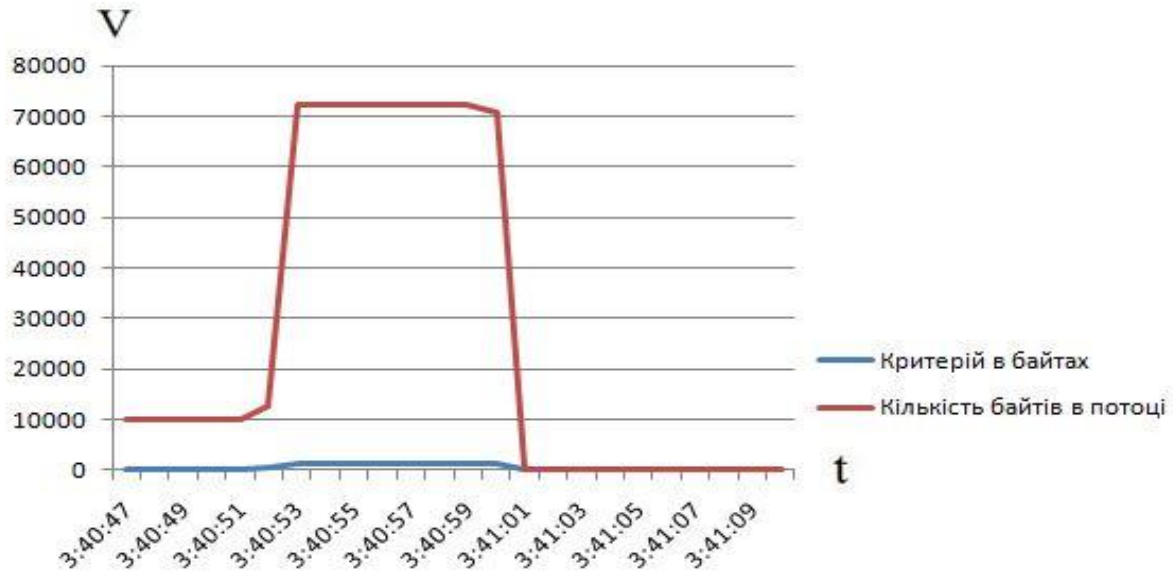


Рис. 3.4. Графік виявлення атаки

Отримані результати дозволили зробити висновок щодо високої конкурентоздатності методів, розроблених в процесі проведення дисертаційного дослідження [31, 33-35], що дозволило сформувати рекомендації щодо їх застосування. Результати досліджень визначають ефективність і практичну значимість швидкодіючої криптосхеми модуля КЗІ для досягнення належної швидкості обміну закодованими повідомленнями та надання їй відповідної криптостікості та дозволяють сформувати вимоги до віддаленого доступу до ДПЛА.

Вони полягатимуть у такому:

1.1. Загальні вимоги до віддаленого доступу:

1.1.1. Віддалений доступ має бути організований тільки з використанням міжмережевого екрану.

1.1.2. Необхідно забезпечити шифрування усього потоку даних (IPSec, SSL тощо).

1.1.3. Необхідно забезпечити двох факторну аутентифікацію.

1.1.4. Заборона створення з'єднання з «периферійними» пристроями (RTU, КП телемеханіки, терміналами МП РЗА, PLC та іншими інтелектуальними пристроями).

1.2. Використовувати Dial UP модем (комутовану лінію) для віддаленого доступу до системи допускається тільки у виняткових випадках при неможливості створення інших видів з'єднання та з дотриманням таких вимог:

1.2.1. Доступ повинен здійснюватися тільки з використанням телефонного міжмережевого екрану.

1.2.2. З обмеженням (лімітом) за часом.

1.2.3. З обмежених точок входу (список дозволених номерів).

1.2.4. Повинні бути змінені або відключені параметри конфігурації, які не використовуються для експлуатації.

1.2.5. Автоматичне блокування при несанкціонованому доступі та автоматичному огляді журналу.

1.3. Вимоги організації з'єднання на основі стека протоколів TCP/IP:

1.3.1. Використання IP Security (IPsec).

1.3.2. Сумісність з Системою Виявлення Вторгнень.

1.3.3. Сумісність з IPv6.

1.4. Вимоги до віддаленого доступу засобами WEB - інтерфейсу:

1.4.1. Повинні бути видалені або заблоковані усі налаштування авторизації, встановлені для надання доступу за замовчуванням.

1.4.2. Необхідно забезпечити захист програмного коду від всіх відомих уразливостей, в тому числі:

а) Переповнення буферу.

б) Ін’єкція командного рядка.

в) SQL уразливостей.

г) Уразливостей Cross-site Scripting (XSS).

д) Уразливостей Remote File lnclude (RFI).

1.5. Вимоги до віддаленого доступу з використанням VPN (Virtual Private Network):

1.5.1. Повинна бути забезпечена перевірка достовірності сертифікатів або ключів безпеки.

1.5.2. Повинна бути забезпечена сумісність роботи VPN сервера з міжмережевим екраном, IPS та IDS щодо виконання функцій маршрутизації та контролю трафіку.

При застосуванні групи квадрокоптерів доволі проблемними стають питання щодо управління ними [36]. Нині існує два можливих методи управління: мультиагентний та ройовий [37-38] (табл.3.5).

Таблиця 3.5

Особливості застосування мультигенного і ройового методів управління групою ДПЛА

Критерій	**Мультиагентний підхід**	**Ройовий підхід**
Автономність групи	Група може діяти автономно	Група може діяти автономно
Автономність окремого ДПЛА	Окремий ДПЛА може діяти цілком автономно	Окремому ДПЛА необхідна постійна комунікація з членами групи
Розміри групи	Від одного ДПЛА	Група має складатись з кількох ДПЛА, що дадуть змогу їй ефективно функціонувати
Кількість виконуваних групою задач	Фактично кожен ДПЛА в групі можевиконувати окрему задачу	Група виконує одну розподілену або невелику кількість схожих задач
Розміри окремого ДПЛА	ДПЛА повинен мати достатні розміри, аби нести на собі всі необхідні для автономногопольоту датчики і засоби комунікації	ДПЛА може мати мінімальні розміри, проте має нести на собі деякі датчики і засоби комунікації
Комунікаційні можливості	Кожен ДПЛА повинен мати можливість комунікації з координаційним центром	Кожен ДПЛА повинен мати можливість комунікації з членами групи
Можливість використання ДПЛА різних модифікацій	Можуть бути використані ДПЛА різних модифікацій для виконання різних окремих задач	Можуть бути використані ДПЛА різних модифікацій для забезпечення ефективності функціонування групи
Вартість окремого ДПЛА	Вартість окремого ДПЛА висока	Вартість окремого ДПЛА невисока

Таким чином, використання мультиагентного підходу дозволяє управляти групою ДПЛА умовно незалежних, що виконують задачі, які істотно відрізняються одна від одної. При використанні ройових методів всі ДПЛА покликані виконувати одну загальну задачу і діють як розподілений об’єкт.

3.3 Метод оцінки ефективності застосування криптосистем в СОІ

У криптографії у якості оцінки ефективності роботи криптосистем, зазвичай, використовують співвідношення $Q = {}^{p}/_{R}$, де R – мінімальна складність методу криптоаналізу, що обчислюється в елементарних операціях обчислювальної техніки, p - ймовірність успіху атакуючої сторони у випадку реалізації цього методу [39-40]. У роботі ця оцінка набула подальшого розвитку у плані

практичного застосування для оцінки відносної ефективності системи захисту інформації в АС переробки інформації та управління технологічними процесами в умовах кібернетичних атак.

Зважаючи на матеріальний характер збитків власника СОІ у разі успіху несанкціонованого втручання у її роботу внаслідок успішної кібератаки та виходячи з необхідності здійснення порушником певних матеріальних витрат для реалізації цієї атаки, є логічним у якості оцінки рівня відносної ефективності системи захисту (засобу КЗІ) – величини Q – використовувати дещо інше співвідношення, а саме:

$$Q = \frac{p^* \cdot \max(C_{\text{ш}})}{\min\{C_{A1}, C_{A2}, \ldots, C_{Ak}\}}, \tag{3.1}$$

де: $C_{\text{ш}}$ – очікувана вартість втрат внаслідок атаки; $C_{A1}, C_{A2}, \ldots, C_{Ak}$ – вартість реалізації відомих атак на систему захисту та їх комбінацій, включаючи криптоаналітичні атаки, атаки на реалізацію та атаки побічними каналами. Всі вартості розраховуються в однакових одиницях; p^* – ймовірність успішної реалізації кращої атаки.

Логічним кроком постає вибір наступних границь для відносної ефективності системи захисту:

$Q \ll 1$ – високий рівень безпеки, коли очікувана шкода власника СОІ суттєво менше витрат порушника на реалізацію атаки;

$Q \approx 1$ – середній рівень безпеки, коли очікувана шкода власника СОІ практично дорівнює сумі витрат порушника на реалізацію атаки, але сукупність атак може мати більшу ефективність;

$Q \gg 1$ – занадто низький рівень безпеки, коли очікувана шкода власника СОІ суттєво перевищує витрати порушника на реалізацію однієї атаки.

Висновки до третього розділу

Квадрокоптери – це сучасна портативна техніка, яка може бути перевезена у будь-якому автомобілі, або переноситись однією людиною, здатна розгортатися із транспортного стану у готовий до польоту дуже швидко, залежно від моделі пристрою та навичок пілота, запуск може бути здійснений менш, ніж за хвилину. Висока завадостійкість сигналу забезпечує стабільний потік відеоданих високої роздільної здатності у каналі зв’язку, завдяки чому оператор в реальному часі може помічати дрібні деталі без необхідності знижуватись, щоб зафіксувати їх.

Однією з основних переваг використання ДПЛА для забезпечення фізичного захисту є забезпечення перспективи «виду згори» та відсутність сліпих зон для спостереження, адже квадрокоптери здатні переміщуватись у просторі із високою швидкістю та моментально змінювати свій напрямок.

Можливість встановлення на борт безпілотника комп’ютера із програмним забезпеченням спеціального значення розширює можливості їх використання для виконання польотів з метою збору інформації для подальшого використання.

Системи зв’язку підтримують криптографічний захист каналу передачі керуючого сигналу та сигналу відеопотоку, шифрування вбудованих сховищ за стандартом AES та 3DES, відмінності яких було розглянуто у цьому розділі в рамках вироблення рекомендацій щодо безпеки каналу передачі даних.

Метод оцінки ефективності роботи криптосистем, застосування якого за рахунок врахування співвідношення середнього значення максимальних втрат власника СОІ у випадку успішних атак на систему захисту до мінімальної вартості реалізації таких атак дозволяє визначити границі для відносної ефективності системи захисту інформації в АС переробки інформації та управління технологічними процесами в умовах кібернетичних атак.

Список джерел, використаних у третьому розділі

1. Конфігурації квадрокоптерів та схеми керування / Мультикоптер. Вікіпедія. URL: https://uk.wikipedia.org/wiki/Мультикоптер]

2. Осипов Ю.Н, Ершов В.И., Иванов А.В. Проблемы выбора рационального типа и оснащения комплексов с БЛА / Доклады и статьи ежегодной научно-практической конференции «Перспективы развития и применения комплексов с беспилотными летательными аппаратами», г. Коломна, 2016. – С. 75-81.

3. Запорожець А. Безпілотні літальні апарати для систем моніторингу в енергетиці та екології. ResearchGate, 2020. https://doi/org/10.13140/-RG.2.2.22137.29280 .

4. Гребенюк А.М., Рибальченко Л.В. Використання безпілотників для потреб поліції / Використання сучасних інформаційних технологій в діяльності Національної поліції України: матеріали Всеукраїнського науково-практичного семінару (28 листопада 2019 р., м. Дніпро). – Дніпро: Дніпропетровський державний університет внутрішніх справ, 2019. С. 26-28.

5. Ле Ч.А., Литвинов Ю.В., Бушуев А.Б. Алгоритм управления парой квадрокоптеров. ResearcherGate. 2019. DOI: 10.18411/lj-05-2019-35

6. Скляров А.А., Скляров С.А. Синергетический подход к управлению квадрокоптером в среде с внешними возмущениями. ИЗВЕСТИЯ ЮФУ. ТЕХНИЧЕСКИЕ НАУКИ. 2014. № 8 (157). С. 159-170.

7. Мясіщев О. А., Швець В.В. Режими польоту контролерів польоту APM 2.6 і Pixhawk БПЛА. *Вісник Хмельницького національного університету*, 2018. С. 78-82.

8. ПИД для квадрокоптеров. 2015. URL : https://blog.rcdetails.info/pid-dlya-kvadrokopterov-perevod

9. В Украине показали возможности ПО DJI Terra для квадрокоптеров серии Phantom 4. *hiTech.ua*. 17.05.2019. URL: https://hi-tech.ua/v-ukraine-pokazali-vozmozhnosti-po-dji-terra-dlya-kvadrokopterov-serii-phantom-4/

10. Довбня С.Я., Кошиченко А.В., Нікірін А.В., Седько В.О. Аналіз використання узгоджених сигналів при проектуванні захищеної інформаційно-телекомунікаційної системи з безпілотними летальними апаратами / Інформаційна безпека України: Зб. наук. доп. та тез науково-технічної конференції; м. Київ, 12-13 березня 2015р., Київський національний університет імені Тараса Шевченка. 156 с.

11. Щекотихин О.В. Сметанин И.В., Піза Д.М. Пассивные оптические сети доступа: монографія. Запоріжжя: ЗНТУ, 2016. 276 с.

12. Иракским повстанцам удалось взломать беспилотники США. LB.ua. 2009, 18 грудня. URL: https://lb.ua/world/2009/12/18/17407_irakskim_-povstantsam_udalos_vzlo.html (дата звернення 19.01.2019)

13. Solomentsev O., Zaliskyi M., Skladannyi P. Operation system for modern unmanned aerial vehicles. International Workshop on Cyber Hygiene, CybHyg 2019. 2019. Vol. 2654. P. 363-374.

14. Карпуков Л.М., Лізунов С.І. Захист інформації в радіоканалах БПЛА / Використання сучасних інформаційних технологій в діяльності Національної поліції України: Всеукр. науково-практ. семінар, 25 листопада 2016 р.: матеріали семінару – Дніпро, 2016. – С.17-22.

15. Костромин В. А. Кибератаки в подробностях: атаки с применением снифферов URL : http://rus-linux.net/MyLDP/sec/cyber-attacks-network-sniffing.html

16. Мороз Б.І., Антіпов О.А., Журавльов В.С. Автоматизована система доставки медикаментів за допомогою безпілотних літальних апаратів (мультикоптерів) за запитом споживача. *Комп'ютерно-інтегровані технології: освіта, наука, виробництво*. 2019. № 35.

17. DJI Lightbridge 2 Информация о продукте. URL: https://www.dji.com/lightbridge-2/info

18. Воздушная система DJI Ocusync. URL: https://4vision.ru/articles/vozdushnaya-sistema-dji-ocusync-air-system.html

19. Сравнение DES, Triple DES, AES, шифрование blowfish для данных. QAStack. URL: https://qastack.ru/programming/5554526/comparison-of-des-triple-des-aes-blowfish-encryption-for-data

20. Тарнавський Ю.А. Технології захисту інформації: підручник. Київ: КПІ ім. Ігоря Сікорського, 2018.

21. Lakhno V., Buriachok V., Parkhuts L., Tarasova H., Kydyralina L., Skladannyi P., Skrypnyk M., Shostakovska A. Development of a conceptual model of adaptive access rights management with using the apparatus of Petri nets. International Journal of Civil Engineering & Technology (IJCIET), Volume 9, Issue 11, November 2018. P. 95-104.

22. Семко В.В., Бурячок В.Л., Толюпа С.В., Складанний П.М. Ситуаційне управління доступом в інформаційно-телекомунікаційній системі. *Проблеми телекомунікацій*. 2015. №2. С. 54–61.

23. Гулак Г.М. Складанний П.М. Забезпечення гарантоздатності автоматизованих систем управління та передачі даних безпілотних літальних апаратів. *Математичні машини та системи*. 2017. № 3. С. 154-161.

24. Красиленко В.Г., Нікітовіч Д.В. Моделювання протоколів узгодженнясекретного матричного ключа для криптографічних перетворень та систем матричного типу. *Системи обробки інформації*. 2017. Вип. 3. С. 151-157.

25. SP 800-22 Rev. 1a A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications. URL: https://csrc.nist.gov/publications/detail/sp/800-22/rev-1a/final

26. Ширяев А.Н. Вероятность (4-е изд., переработ. и доп). Москва: МЦНМО, 2007. 552 с.

27. Гулак Г., Ковальчук Л. Різні підходи до визначення випадкових послідовностей // Науково-технічний збірник «Правове, нормативне та метрологічне забезпечення системи захисту інформації в Україні», вип. 3, Київ, 2001. С. 127-133.

28. Статистические тесты NIST. URL: https://www.nist.gov/

29. ДСТУ 7624:2014 Інформаційні технології. Криптографічний захист інформації. Алгоритм симетричного блокового перетворення.URL: http://online.budstandart.com/ua/catalog/doc-page?id_doc=65314

30. Авраменко В. І., Карімов І. К.. Теорія ймовірностей і математична статистика : навч. посібник (2-ге вид., перероб. і доп.) Дніпродзержинськ : ДДТУ, 2013. 245 с.

31. Гулак Г.М., Бурячок В.Л., Складанний П.М. Швидкий алгоритм генерації підстановок багатоалфавітної заміни. *Захист інформації*. 2017. №2. С. 173–177.

32. Остапов С. Е., Євсеєв С. П., Король О. Г. Технології захисту інформації : навчальний посібник. Харків: Вид. ХНЕУ, 2013. 476 с.

33. Гулак Г. М., Семко В. В., Складанний П. М. Модель системи виявлення вторгнень з використанням двоступеневого критерію виявлення мережних аномалій. *Сучасний захист інформації*. 2015. № 4. С. 81–85.

34. Гулак Г. М., Бурячок В. Л., Складанний П. М. Швидкий алгоритм генерації підстановок багатоалфавітної заміни. *Захист інформації*. 2017. №2. С. 173–177.

35. Модель загроз безпеки криптосистем. / І Міжнародна науково-технічна конференція «Актуальні проблеми розвитку науки і техніки» (Київ, 20 жовтня 2015р.). Київ: ДУТ, 2015. С.35-37.

36. Лысенко А.И., Чумаченко С.Н., Тачинина Е.Н. Математическая постановка задачи оптимизации движения группы квадрокоптеров. *Техническая механика*. 2016. № 1. С. 74-83.

37. Савченко В.А., Савченко В.В., Мацько О.Ю., Кізяк Я.О., Лаптєв О.А., Лазаренко С.В. Мультиагентна технологія пошуку цифрових радіозакладних пристроїв на основі кластеризації за методом бджолиної колонії. *Захист інформації*, (2019). 21(3), 194-202. doi:10.18372/2410-7840.21.13955.

38. Парасюк И.Н., Ершов С.В. Методы взаимодействия и координации в мультиагентных системах на основе нечеткой логики высшего типа. *Проблеми програмування*. 2014. № 2-3. С. 242-252.

39. Кузнецов О.О., Пушкарьов А.І., Горбенко Ю.І. Кодові криптосистеми для постквантового застосування. Математичне та комп'ютерне моделювання. Серія: Технічні науки: зб. наук. пр. Кам’янець-Подільський: Кам'янець-Подільськ. нац. ун-т, 2017. Вип. 15. С. 109-115.

40. Миронюк Т. В., Дуда Л. Т. Дослідження методів захисту онлайн спілкування. *Вісник Черкаського державного технологічного університету. Серія : Технічні науки*. 2017. № 4. С. 138-143.

ВИСНОВКИ

У дисертації вирішено актуальне наукове завдання*, яке полягає у розробленні теоретичних і прикладних засад побудови та забезпечення методами криптографічної обробки інформації імітостійкості та конфіденційності даних в СОІ з урахуванням множини кіберзагроз та потенційно можливих наслідків їх реалізації.*

У процесі виконання дисертаційної роботи отримано такі основні результати.

1. Розроблено метод генерації потоку підстановок з використанням шифру БАЗ для забезпечення імітостійкого шифрування в СОІ, впровадження якого дозволяє обрати таку степень підстановок, яка б забезпечувала достатню швидкодію криптоперетворення та була б раціональною для забезпечення захисту повідомлень від підробки.

2. Розроблено метод виявлення атак на програмні реалізації засобів криптографічного захисту інформації в СОІ, впровадження якого дозволяє своєчасно виявити момент настання певної критичної ситуації та прийняти рішення щодо подальших дій.

3. Розроблено модель функціонування (криптосхема) шифратора БАЗ (модулю криптографічного захисту інформації) в СОІ, яка є комплексним рішенням в межах двох попередніх методів та впровадження якої дозволяє забезпечити конфіденційність і цілісність інформації, що циркулює в СОІ та в умовах кібератак забезпечити власне функціональну безпеку та живучість самої системи.

4. Удосконалено метод оцінки ефективності застосування криптосистем, який шляхом урахування матеріального характеру збитків власника СОІ у разі успіху несанкціонованого втручання у її роботу, дозволяє визначити границі для відносної ефективності системи захисту інформації в СОІ в умовах реалізації кіберзагроз.

5. Проведено, шляхом імітаційного моделювання процесу нападу на програмну реалізацію засобу КЗІ, експериментальне дослідження моделі функціонування шифратора БАЗ. Це, по-перше, підтвердило дієздатність крипто

схеми та, по-друге, в масштабі часу наближеному до реального дозволило впевнитись, що застосування моделі дозволить зменшити час на виявлення атаки приблизно на 20%.

Як результат, в цілому застосування розроблених методів та моделі дозволить:

оперативно приймати рішення щодо подальших дій системи.

знизити ймовірність підробки команди управління до прийнятної для практичного застосування величини, що оцінюється величиною 10^{-6};

знизити вартість системи виявлення атак на програмні реалізації засобів КЗІ приблизно на 25%;

забезпечити можливість автоматичного переходу ДПЛА в режим автономного виконання завдань в разі виявлення загроз безпеки функціонування системи захисту його радіоканалу управління та передачі даних.

Таким чином, поставлене актуальне наукове завдання розв’язане у повному обсязі. Усі визначені часткові завдання вирішено, мету досліджень досягнуто.

ДОДАТКИ

Додаток А.
Акти впровадження результатів дисертаційної роботи

ЗАТВЕРДЖУЮ
Заступник директора Інституту проблем
математичних машин і систем НАН України
д.ф.-м.н., професор
_____ В.П. Клименко
"30" листопада 2020 року

АКТ
про впровадження результатів дисертаційного дослідження
СКЛАДАННОГО Павла Миколайовича

Даним актом засвідчується, що нижчеперелічені наукові положення, а саме:

модель виявлення атак на програмні реалізації засобів криптографічного захисту інформації в автоматизованих системах критичного застосування;

метод генерації потоку підстановок з використанням шифру багатоалфавітної заміни для забезпечення імітостійкого шифрування в автоматизованих системах критичного застосування;

метод контролю стану гарантоздатності програмних реалізацій засобів криптографічного захисту інформації в автоматизованих системах критичного застосування;

модель формування криптосхеми модулю криптографічного захисту інформації в автоматизованих системах критичного застосування, -

що розроблені Складанним П.М., використано в Інституті проблем математичних машин і систем НАН України під час виконання у 2019 – 2020 роках за Державним контрактом від 09.08.2018 № 183/18/2 науково-дослідної роботи «Базис-Наука» (державний номер реєстрації 0119U000042дс), спрямованої на вирішення питань побудови мережі гарантоздатних ситуацних центрів (СЦ) сектору безпеки і оборони (СБО)

Отримані П.М. Складанним результати було використано для прийняття обґрунтованих технічних рішень щодо побудови системи кібербезпеки СЦ, а також застосування засобів кіберзахисту та раціонального управління ними в контексті забезпечення конфіденційності та цілісності інформації, що циркулює в СЦ СБО в умовах кібератак та функціональної безпеки і живучості самої системи.

Запропоновані автором рішення дозволяють підвищити ефективність функціонування системи кібербезпеки мережі гарантоздатних СЦ. В подальшому їх буде використано для розв'язання комплексних проблем щодо захисту критичної інфраструктури та забезпечення національної безпеки в цілому.

Завідувач відділу 235 ІПММС
к.т.н. _____ В.Ф. Гречанінов.

ЗАТВЕРДЖУЮ
Начальник Національного центру управління
та випробувань космічних засобів
канд. техн. наук, старш. наук. співр.

Володимир ПРИСЯЖНИЙ

«27» листопада 2020 року

А К Т

про впровадження результатів дисертаційної роботи здобувача наукового ступеня кандидата технічних наук Складанного Павла Миколайовича на тему «Моделі і методи забезпечення імітостійкості та конфіденційності в системах обробки інформації»

Комісія Національного центру управління та випробувань космічних засобів у складі:

голови комісії - начальника відділу науково-дослідної та випробувальної роботи, канд. техн. наук Мамарєва В.М;

членів комісії:
- заступника начальника відділу науково-дослідної та випробувальної роботи, канд. техн. наук, старш. наук. співр. Козуба А.М.;
- головного фахівця відділу науково-дослідної та випробувальної роботи, канд. техн. наук Кутового О.М.;
- начальника відділу технічного захисту інформації Шпиталя О.О.

у період з «23» по «26» листопада 2020 року розглянула матеріали та результати дисертаційного дослідження Складанного П.М. на тему «Моделі і методи забезпечення імітостійкості та конфіденційності в системах обробки інформації».

КОМІСІЯ ВСТАНОВИЛА ТА ДАНИМ АКТОМ ЗАСВІДЧУЄ:

1) наукові положення, отримані особисто Складанним П.М., зокрема:

- удосконалений метод генерації потоку підстановок з використанням шифру багатоалфавітної заміни для забезпечення конфіденційності та цілісності інформації в АСУ;

- удосконалений метод оцінки ефективності застосування криптосистем;

впроваджено Національним центром управління та випробувань космічних засобів в НДР «Розробка науково-технічних пропозицій з організації віддаленого управління станціями оптико-електронних спостережень типу 1 та типу 2» (номер держ. реєстрації 0120U105420).

2) впровадження зазначених наукових положень забезпечило гарантовану цілісність та конфіденційність команд управління станціями оптико-електронних спостережень в режимі віддаленого доступу з наступними показниками:

- зниження ймовірності порушення цілісності команд управління до прийнятної для практичного використання величини - порядку 10^{-6};
- до 25% зниження вартості системи виявлення атак на програмні засоби криптографічного захисту інформації.

Економічний ефект від впровадження не розраховувався у зв'язку з науковим призначенням результатів.

Акт складено для представлення у спеціалізовану вчену раду та не є підставою для виплати винагороди за впровадження та інших авторських винагород.

Голова комісії: Віктор МАМАРЄВ

Члени комісії: Андрій КОЗУБ

Олександр КУТОВИЙ

Олександр ШПИТАЛЬ

КИЇВСЬКИЙ УНІВЕРСИТЕТ
ІМЕНІ БОРИСА ГРІНЧЕНКА

Вул. Бульварно-Кудрявська, 18/2, м. Київ,
Україна, 04053, тел./факс: +380 44 272-19-02
e-mail: kubg@kubg.edu.ua, www.kubg.edu.ua

BORYS GRINCHENKO
KYIV UNIVERSITY

18/2 Bulvarno-Kudriavska St., Kyiv,
Ukraine, 04053, tel./fax: +380 44 272-19-02
e-mail: kubg@kubg.edu.ua, www.kubg.edu.ua

30.11.2020р. № 71-н

На № _____ *від* _____

АКТ
про впровадження результатів дисертаційного дослідження
Складанного Павла Миколайовича
на тему «Моделі і методи забезпечення імітостійкості та конфіденційності в системах обробки інформації»,
поданої на здобуття наукового ступеня кандидата технічних наук
зі спеціальності 05.13.06 – інформаційні технології

Цим Актом, ґрунтуючись на рішенні кафедри інформаційної та кібернетичної безпеки Факультету інформаційних технологій та управління Київського університету імені Бориса Грінченка (протокол №12 від 04 листопада 2020 р.), засвідчуємо, що *нижчеперелічені наукові положення, а саме:*

а) модель виявлення атак на програмні реалізації засобів криптографічного захисту інформації;

б) метод контролю стану гарантоздатності програмних реалізацій засобів криптографічного захисту інформації;

в) метод оцінки ефективності застосування криптосистем;

г) модель функціонування (криптосхема) шифратора БАЗ (модулю криптографічного захисту інформації).

Розроблені особисто Складанним Павлом Миколайовичем у ході проведення ним дисертаційних досліджень та *отримали високу оцінку* при обговоренні на засіданнях кафедри інформаційної та кібернетичної безпеки Київського університету імені Бориса Грінченка, а також за результатами озвучення відповідних доповідей у ході міжнародних і всеукраїнських наукових конференцій та семінарів.

Зазначені наукові результати:

по-перше, впроваджені в освітній процес Університету у робочих програмах навчальних дисциплін «Методи побудова та аналізу криптосистем» і «Математичні методи криптографії» другого (магістерського) рівня вищої освіти;

по-друге, впроваджені в програмно-апаратне забезпечення «Центру дослідження технологій захисту інформаційних активів» Університету Грінченка при розгортанні Лабораторії систем криптографічного та технічного захисту інформації (навчальний кіберполігон).

Це дозволило знизити вартість системи виявлення атак на програмні реалізації засобів КЗІ приблизно на 25%, а також визначити границі для відносної ефективності системи захисту інформації в умовах кібернетичних атак, суттєво підвищити якість підготовки фахівців із кібербезпеки за спеціалізацією «Безпека інформаційно-комунікаційних систем».

Акт видано для подання до спеціалізованої вченої ради.

Проректор з наукової роботи Наталія ВІННІКОВА

Додаток Б

Лістинг програми генерації потоку підстановок шифру багатоалфавітної заміни

#include <linux/types.h>
lfsr113 version:

```
	 *
	 * x_n = (s1_n ^ s2_n ^ s3_n ^ s4_n)
	 *
	 * s1_{n+1} = (((s1_n & 4294967294) << 18) ^ (((s1_n <<  6) ^ s1_n)
>> 13))
	 * s2_{n+1} = (((s2_n & 4294967288) <<  2) ^ (((s2_n <<  2) ^ s2_n)
>> 27))
	 * s3_{n+1} = (((s3_n & 4294967280) <<  7) ^ (((s3_n << 13) ^ s3_n)
>> 21))
	 * s4_{n+1} = (((s4_n & 4294967168) << 13) ^ (((s4_n <<  3) ^ s4_n)
>> 12))
	 *
	 * The period of this generator is about 2^113 (see erratum paper).
	 *
	 * From: P. L'Ecuyer, "Maximally Equidistributed Combined Tausworthe
	 * Generators", Mathematics of Computation, 65, 213 (1996), 203--
213:
	 * http://www.iro.umontreal.ca/~lecuyer/myftp/papers/tausme.ps
	 *
ftp://ftp.iro.umontreal.ca/pub/simulation/lecuyer/papers/tausme.ps
	 *
	 * There is an erratum in the paper "Tables of Maximally
Equidistributed
	 * Combined LFSR Generators", Mathematics of Computation, 68, 225
(1999),
	 * 261--269:
http://www.iro.umontreal.ca/~lecuyer/myftp/papers/tausme2.ps
	 *
	 *      ... the k_j most significant bits of z_j must be non-zero,
	 *      for each j. (Note: this restriction also applies to the
	 *      computer code given in [4], but was mistakenly not mentioned
	 *      in that paper.)
	 *
	 * This affects the seeding procedure by imposing the requirement
	 * s1 > 1, s2 > 7, s3 > 15, s4 > 127.
	 */


	#include <linux/types.h>
	#include <linux/percpu.h>
	#include <linux/export.h>
	#include <linux/jiffies.h>
	#include <linux/random.h>
	#include <linux/sched.h>
	#include <linux/bitops.h>
	#include <asm/unaligned.h>
```

```
	#include <trace/events/random.h>


	/**
	 *	prandom_u32_state - seeded pseudo-random number generator.
	 *	@state: pointer to state structure holding seeded state.
	 *
	 *	This is used for pseudo-randomness with no outside seeding.
	 *	For more random results, use prandom_u32().
	 */
	u32 prandom_u32_state(struct rnd_state *state)
	{
	#define TAUSWORTHE(s, a, b, c, d) ((s & c) << d) ^ (((s << a) ^ s)
>> b)
	 state->s1 = TAUSWORTHE(state->s1,  6U, 13U, 4294967294U, 18U);
	 state->s2 = TAUSWORTHE(state->s2,  2U, 27U, 4294967288U,  2U);
	 state->s3 = TAUSWORTHE(state->s3, 13U, 21U, 4294967280U,  7U);
	 state->s4 = TAUSWORTHE(state->s4,  3U, 12U, 4294967168U, 13U);


	 return (state->s1 ^ state->s2 ^ state->s3 ^ state->s4);
	}
	EXPORT_SYMBOL(prandom_u32_state);


	/**
	 *	prandom_bytes_state - get the requested number of pseudo-
random bytes
	 *
	 *	@state: pointer to state structure holding seeded state.
	 *	@buf: where to copy the pseudo-random bytes to
	 *	@bytes: the requested number of bytes
	 *
	 *	This is used for pseudo-randomness with no outside seeding.
	 *	For more random results, use prandom_bytes().
	 */
	void prandom_bytes_state(struct rnd_state *state, void *buf, size_t
bytes)
	{
	 u8 *ptr = buf;


	 while (bytes >= sizeof(u32)) {
		put_unaligned(prandom_u32_state(state), (u32 *) ptr);
		ptr += sizeof(u32);
		bytes -= sizeof(u32);
	 }
```

```
 if (bytes > 0) {
        u32 rem = prandom_u32_state(state);
        do {
                *ptr++ = (u8) rem;
                bytes--;
                rem >>= BITS_PER_BYTE;
        } while (bytes > 0);
 }
}
EXPORT_SYMBOL(prandom_bytes_state);


static void prandom_warmup(struct rnd_state *state)
{
 /* Calling RNG ten times to satisfy recurrence condition */
 prandom_u32_state(state);
 prandom_u32_state(state);
 prandom_u32_state(state);
 prandom_u32_state(state);
 prandom_u32_state(state);
 prandom_u32_state(state);
 prandom_u32_state(state);
 prandom_u32_state(state);
 prandom_u32_state(state);
 prandom_u32_state(state);
}


void prandom_seed_full_state(struct rnd_state __percpu *pcpu_state)
{
 int i;


 for_each_possible_cpu(i) {
        struct rnd_state *state = per_cpu_ptr(pcpu_state, i);
        u32 seeds[4];


        get_random_bytes(&seeds, sizeof(seeds));
        state->s1 = __seed(seeds[0],   2U);
        state->s2 = __seed(seeds[1],   8U);
        state->s3 = __seed(seeds[2],  16U);
        state->s4 = __seed(seeds[3], 128U);


        prandom_warmup(state);
 }
```

```
}
EXPORT_SYMBOL(prandom_seed_full_state);


#ifdef CONFIG_RANDOM32_SELFTEST
static struct prandom_test1 {
 u32 seed;
 u32 result;
} test1[] = {
 { 1U, 3484351685U },
 { 2U, 2623130059U },
 { 3U, 3125133893U },
 { 4U,  984847254U },
};


static struct prandom_test2 {
 u32 seed;
 u32 iteration;
 u32 result;
} test2[] = {
 /* Test cases against taus113 from GSL library. */
 {  931557656U, 959U, 2975593782U },
 { 1339693295U, 876U, 3887776532U },
 { 1545556285U, 961U, 1615538833U },
 {  601730776U, 723U, 1776162651U },
 { 1027516047U, 687U,  511983079U },
 {  416526298U, 700U,  916156552U },
 { 1395522032U, 652U, 2222063676U },
 {  366221443U, 617U, 2992857763U },
 { 1539836965U, 714U, 3783265725U },
 {  556206671U, 994U,  799626459U },
 {  684907218U, 799U,  367789491U },
 { 2121230701U, 931U, 2115467001U },
 { 1668516451U, 644U, 3620590685U },
 {  768046066U, 883U, 2034077390U },
 { 1989159136U, 833U, 1195767305U },
 {  536585145U, 996U, 3577259204U },
 { 1008129373U, 642U, 1478080776U },
 { 1740775604U, 939U, 1264980372U },
 { 1967883163U, 508U,   10734624U },
 { 1923019697U, 730U, 3821419629U },
 {  442079932U, 560U, 3440032343U },
 { 1961302714U, 845U,  841962572U },
 { 2030205964U, 962U, 1325144227U },
 { 1160407529U, 507U,  240940858U },
 {  635482502U, 779U, 4200489746U },
 { 1252788931U, 699U,  867195434U },
```

```
        { 1961817131U, 719U,  668237657U },
        { 1071468216U, 983U,  917876630U },
        { 1281848367U, 932U, 1003100039U },
        {  582537119U, 780U, 1127273778U },
        { 1973672777U, 853U, 1071368872U },
        { 1896756996U, 762U, 1127851055U },
        {  847917054U, 500U, 1717499075U },
        { 1240520510U, 951U, 2849576657U },
        { 1685071682U, 567U, 1961810396U },
        { 1516232129U, 557U,    3173877U },
        { 1208118903U, 612U, 1613145022U },
        { 1817269927U, 693U, 4279122573U },
        { 1510091701U, 717U,  638191229U },
        {  365916850U, 807U,  600424314U },
        {  399324359U, 702U, 1803598116U },
        { 1318480274U, 779U, 2074237022U },
        {  697758115U, 840U, 1483639402U },
        { 1696507773U, 840U,  577415447U },
        { 2081979121U, 981U, 3041486449U },
        {  955646687U, 742U, 3846494357U },
        { 1250683506U, 749U,  836419859U },
        {  595003102U, 534U,  366794109U },
        {   47485338U, 558U, 3521120834U },
        {  619433479U, 610U, 3991783875U },
        {  704096520U, 518U, 4139493852U },


      struct siprand_state {
        unsigned long v0;
        unsigned long v1;
        unsigned long v2;
        unsigned long v3;
      };


      static DEFINE_PER_CPU(struct siprand_state, net_rand_state)
__latent_entropy;
      DEFINE_PER_CPU(unsigned long, net_rand_noise);
      EXPORT_PER_CPU_SYMBOL(net_rand_noise);


      /*
       * This is the core CPRNG function.  As "pseudorandom", this is not
used
       * for truly valuable things, just intended to be a PITA to guess.
       * For maximum speed, we do just two SipHash rounds per word.  This
is
       * the same rate as 4 rounds per 64 bits that SipHash normally uses,
```

```
       * so hopefully it's reasonably secure.
       *
       * There are two changes from the official SipHash finalization:
       * - We omit some constants XORed with v2 in the SipHash spec as
irrelevant;
       *   they are there only to make the output rounds distinct from the
input
       *   rounds, and this application has no input rounds.
       * - Rather than returning v0^v1^v2^v3, return v1+v3.
       *   If you look at the SipHash round, the last operation on v3 is
       *   "v3 ^= v0", so "v0 ^ v3" just undoes that, a waste of time.
       *   Likewise "v1 ^= v2".  (The rotate of v2 makes a difference, but
       *   it still cancels out half of the bits in v2 for no benefit.)
       *   Second, since the last combining operation was xor, continue
the
       *   pattern of alternating xor/add for a tiny bit of extra non-
linearity.
       */
      static inline u32 siprand_u32(struct siprand_state *s)
      {
       unsigned long v0 = s->v0, v1 = s->v1, v2 = s->v2, v3 = s->v3;
       unsigned long n = raw_cpu_read(net_rand_noise);



       v3 ^= n;
       PRND_SIPROUND(v0, v1, v2, v3);
       PRND_SIPROUND(v0, v1, v2, v3);
       v0 ^= n;
       s->v0 = v0;  s->v1 = v1;  s->v2 = v2;  s->v3 = v3;
       return v1 + v3;
      }




      /**
       *       prandom_u32 - pseudo random number generator
       *
       *       A 32 bit pseudo-random number is generated using a fast
       *       algorithm suitable for simulation. This algorithm is NOT
                              *       considered safe for cryptographic use.
       */
      u32 prandom_u32(void)
      {
               struct siprand_state *state = get_cpu_ptr(&net_rand_state);
       u32 res = siprand_u32(state);
```

```
                                                        trace_prandom_u32(res);
        put_cpu_ptr(&net_rand_state);
        return res;
                                                                              }
      EXPORT_SYMBOL(prandom_u32);


      /**
       *       prandom_bytes - get the requested number of pseudo-random
bytes
       *       @buf: where to copy the pseudo-random bytes to
       *       @bytes: the requested number of bytes
       */
      void prandom_bytes(void *buf, size_t bytes)
      {
        struct siprand_state *state = get_cpu_ptr(&net_rand_state);
                                                              u8 *ptr = buf;


        while (bytes >= sizeof(u32)) {
               put_unaligned(siprand_u32(state), (u32 *)ptr);
               ptr += sizeof(u32);
                                                    bytes -= sizeof(u32);
        }


                                                         if (bytes > 0) {
               u32 rem = siprand_u32(state);


                                                                        do {
                      *ptr++ = (u8)rem;
                      rem >>= BITS_PER_BYTE;
               } while (--bytes > 0);
                                                                              }
        put_cpu_ptr(&net_rand_state);
      }
      EXPORT_SYMBOL(prandom_bytes);


                                                                     /**
       *       prandom_seed - add entropy to pseudo random number
generator
       *       @entropy: entropy value
                                                                            *
       *       Add some additional seed material to the prandom pool.
       *       The "entropy" is actually our IP address (the only caller
is
```

```
   *       the network code), not for unpredictability, but to ensure
                                                                    that
 *       different machines are initialized differently.
 */
                                        void prandom_seed(u32 entropy)
{
 int i;


 add_device_randomness(&entropy, sizeof(entropy));


 for_each_possible_cpu(i) {
          struct siprand_state *state = per_cpu_ptr(&net_rand_state,
                                                                     i);
         unsigned long v0 = state->v0, v1 = state->v1;
         unsigned long v2 = state->v2, v3 = state->v3;


         do {
                 v3 ^= entropy;
                 PRND_SIPROUND(v0, v1, v2, v3);
                 PRND_SIPROUND(v0, v1, v2, v3);
                 v0 ^= entropy;
                        } while (unlikely(!v0 || !v1 || !v2 || !v3));


         WRITE_ONCE(state->v0, v0);
         WRITE_ONCE(state->v1, v1);
         WRITE_ONCE(state->v2, v2);
         WRITE_ONCE(state->v3, v3);
 }
}
EXPORT_SYMBOL(prandom_seed);


/*
 *       Generate some initially weak seeding values to allow
 *       the prandom_u32() engine to be started.
 */
static int __init prandom_init_early(void)
{
 int i;
 unsigned long v0, v1, v2, v3;


                                      if (!arch_get_random_long(&v0))
         v0 = jiffies;
```

```
 if (!arch_get_random_long(&v1))
        v1 = random_get_entropy();
 v2 = v0 ^ PRND_K0;
 v3 = v1 ^ PRND_K1;


                                          for_each_possible_cpu(i) {
        struct siprand_state *state;


        v3 ^= i;
                                      PRND_SIPROUND(v0, v1, v2, v3);
        PRND_SIPROUND(v0, v1, v2, v3);
        v0 ^= i;


        state = per_cpu_ptr(&net_rand_state, i);
        state->v0 = v0;  state->v1 = v1;
        state->v2 = v2;  state->v3 = v3;
 }


 return 0;
}
core_initcall(prandom_init_early);




/* Stronger reseeding when available, and periodically thereafter.
*/
static void prandom_reseed(struct timer_list *unused);


static DEFINE_TIMER(seed_timer, prandom_reseed);


static void prandom_reseed(struct timer_list *unused)
{
 unsigned long expires;
 int i;


                                                                    /*
  * Reinitialize each CPU's PRNG with 128 bits of key.
  * No locking on the CPUs, but then somewhat random results are,
  * well, expected.
  */
```

```
        for_each_possible_cpu(i) {
                struct siprand_state *state;
                unsigned long v0 = get_random_long(), v2 = v0 ^ PRND_K0;
                    unsigned long v1 = get_random_long(), v3 = v1 ^ PRND_K1;
                                                        #if BITS_PER_LONG == 32
                int j;


                /*
                 * On 32-bit machines, hash in two extra words to
                 * approximate 128-bit key length.  Not that the hash
                 * has that much security, but this prevents a trivial
                 * 64-bit brute force.
                 */
                for (j = 0; j < 2; j++) {
                        unsigned long m = get_random_long();


                                                                    v3 ^= m;
                        PRND_SIPROUND(v0, v1, v2, v3);
                        PRND_SIPROUND(v0, v1, v2, v3);
                        v0 ^= m;
                }
        #endif
                                                                            /*
                 * Probably impossible in practice, but there is a
                 * theoretical risk that a race between this reseeding
                 * and the target CPU writing its state back could
                 * create the all-zero SipHash fixed point.
                 *
                 * To ensure that never happens, ensure the state
                 * we write contains no zero words.
                 */
                state = per_cpu_ptr(&net_rand_state, i);
                                        WRITE_ONCE(state->v0, v0 ? v0 : -1ul);
                WRITE_ONCE(state->v1, v1 ? v1 : -1ul);
                WRITE_ONCE(state->v2, v2 ? v2 : -1ul);
                WRITE_ONCE(state->v3, v3 ? v3 : -1ul);
        }


          /* reseed every ~60 seconds, in [40 .. 80) interval with slack */
        expires = round_jiffies(jiffies + 40 * HZ + prandom_u32_max(40 *
HZ));
        mod_timer(&seed_timer, expires);
                                                                            }
```

```
	/*
	 * The random ready callback can be called from almost any
interrupt.
	 * To avoid worrying about whether it's safe to delay that interrupt
	 * long enough to seed all CPUs, just schedule an immediate timer
event.
	 */
	static void prandom_timer_start(struct random_ready_callback
*unused)
	{
	                                        mod_timer(&seed_timer, jiffies);
	}


	#ifdef CONFIG_RANDOM32_SELFTEST
	/* Principle: True 32-bit random numbers will all have 16 differing
bits on
	 * average. For each 32-bit number, there are 601M numbers differing
by 16
	 * bits, and 89% of the numbers differ by at least 12 bits. Note
that more
	 * than 16 differing bits also implies a correlation with inverted
bits. Thus
	 * we take 1024 random numbers and compare each of them to the other
ones,
	 * counting the deviation of correlated bits to 16. Constants report
32,
	 * counters 32-log2(TEST_SIZE), and pure randoms, around 6 or lower.
With the
	 * u32 total, TEST_SIZE may be as large as 4096 samples.
	 */
	                                                #define TEST_SIZE 1024
	static int __init prandom32_state_selftest(void)
	                                                                        {
	 unsigned int x, y, bits, samples;
	 u32 xor, flip;
	 u32 total;
	 u32 *data;


	 data = kmalloc(sizeof(*data) * TEST_SIZE, GFP_KERNEL);
	 if (!data)
	        return 0;


	 for (samples = 0; samples < TEST_SIZE; samples++)
	        data[samples] = prandom_u32();
```

```
        flip = total = 0;
        for (x = 0; x < samples; x++) {
                for (y = 0; y < samples; y++) {
                        if (x == y)
                                continue;
                        xor = data[x] ^ data[y];
                        flip |= xor;
                                                bits = hweight32(xor);
                        total += (bits - 16) * (bits - 16);
                }
        }


        /* We'll return the average deviation as 2*sqrt(corr/samples),
which
         * is also sqrt(4*corr/samples) which provides a better
resolution.
         */
        bits = int_sqrt(total / (samples * (samples - 1)) * 4);
                                                        if (bits > 6)
                pr_warn("prandom32: self test failed (at least %u bits"
                        " correlated, fixed_mask=%#x fixed_value=%#x\n",
                        bits, ~flip, data[0] & ~flip);
        else
                pr_info("prandom32: self test passed (less than %u bits"
                        " correlated)\n",
                                                                bits+1);
        kfree(data);
        return 0;
                                                                }
core_initcall(prandom32_state_selftest);
#endif /*  CONFIG_RANDOM32_SELFTEST */


/*
                    * Start periodic full reseeding as soon as strong
 * random numbers are available.
 */
static int __init prandom_init_late(void)
{
                 static struct random_ready_callback random_ready = {
                .func = prandom_timer_start
        };
        int ret = add_random_ready_callback(&random_ready);


                                                if (ret == -EALREADY) {
```

```
        prandom_timer_start(&random_ready);
        ret = 0;
    }
    return ret;
}
late_initcall(prandom_init_late);
```

Додаток В.
Відомості про апробацію результатів дисертаційної роботи

№ з/п	Назви конференції, конгресу, симпозіуму, семінару, школи	Місце та дата проведення	Форма участі	Де опубліковано
1	Міжнародна науково-технічна конференція «Сучасні інформаційно-телекомунікаційні технології».	м. Київ, Державний університет телекомунікацій, 2014	Очна, тези доповіді	Доповіді та тези доповідей конференції
2	І Міжнародна науково-технічна конференція «Актуальні проблеми розвитку науки і техніки»	м. Київ, Державний університет телекомунікацій, 20 жовтня 2015р	Очна, тези доповіді	Матеріали доповідей конференції
3	ІІ Міжнародна науково-технічна конференція «Актуальні проблеми розвитку науки і техніки»	м. Київ, Державний університет телекомунікацій, 12 грудня 2015р.	Очна, доповідь	Тезиси доповідей конференції
4	Міжнародна науково-технічна конференція студентства та молоді «Світ телекомунікацій та інформатизації»	м. Київ, Державний університет телекомунікацій, 02 березня 2016р	Очна, тези доповіді	Матеріали доповідей конференції
5	ІІ Всеукраїнська науково-практична конференція «Кібербезпека в Україні: правові та організаційні питання»	м. Одеса, Одеський державний університет внутрішніх справ, 17 листопада 2017р	Очна, доповідь	Тезиси доповідей конференції
6	ІХ Всеукраїнська науково-практична конференція «Актуальні проблеми управління інформаційною безпекою держави»	м. Київ; Нац. акад. СБУ 30 березня 2018р	Очна, доповідь	Тезиси доповідей конференції
7	І Міжнародна науково-практична конференція «Проблеми кібербезпеки інформаційно-телекомунікаційних систем» (PCSITS)	м. Київ Київський національний університет імені Тараса Шевченка 05-06 квітня 2018 року	Очна, доповідь	Тезиси доповідей конференції