DOI: 10.18372/2410-7840.22.14976
УДК 004.056:351.861

МОДЕЛЬ ЗАХИСТУ КІБЕРПРОСТОРУ CYBERSEC

Юлія Ткач

У статті запропоновано модель захисту кіберпростору CyberSec, що орієнтована на виконання функції «кіберзахисту». Дана модель є функціональною, складається з п'яти етапів, об'єднує в собі низку методів й моделей, є циклічною, а тому дозволяє створити самоналагоджувану систему захисту у кіберпросторі. На першому етапі «Розвідка та виявлення» здійснюється опис середовища безпеки, тобто формується модель загроз, що є повним переліком усіх можливих загроз, які існують або можуть виникнути в даній ситуації. Другим етапом «Озброєння» є вибір засобів захисту та побудова системи захисту кіберпростору (СЗК). Проведення контролю системи (кібепростору) є третім етапом «Контроль». На четвертому етапі «Протидія» побудови захищеного кіберпростору виконується оцінка дієвості запропонованої СЗК. На п'ятому етапі «Активна протидія» здійснюється підготовка нормативних документів, інформування корпорацій щодо інцидентів кібербезпеки активна протидія на рівні держави, тобто відбувається застосування контрзаходів. Модель захисту кіберпростору CyberSec дозволяє на практиці побудувати захищений кіберпростір як окремої корпорації, так і держави в цілому.

Ключові слова: *кіберпростір, модель захисту, національна безпека, захищений кіберпростір.*

ВСТУП

Фундаментом інформаційної безпеки України є безпека кіберпростору держави. Вирішення усіх інших завдань забезпечення національної безпеки країни можливе саме за умов стійкої безпеки в кіберпросторі.

З іншого боку, справжня інформаційна безпека існує тільки за умови надійного захисту національних інтересів України від будь-якого силового чи інформаційного тиску.

Тому серед головних передумов національної безпеки України є кібербезпека держави, в цьому контексті пріоритетного значення набуває побудова захищеного кіберпростору.

МЕТА РОБОТИ

Побудувати модель захисту кіберпростору CyberSec, яка дозволить на практиці побудувати захищений кіберпростір як окремої корпорації, так і держави в цілому.

ПОСТАНОВКА ЗАДАЧІ

Не існує єдиного рішення задля забезпечення кібербезпеки держави.

Розв'язування завдання побудови системи захисту інформації (СЗІ) ускладнюється такими її властивостями [3]:

- складний опосередкований взаємозв'язок показників якості СЗІ з показниками якості інформаційної системи;

- необхідність урахування великої кількості показників (вимог) СЗІ в оцінюванні та виборі її раціонального варіанта;

- переважно якісний характер показників (вимог), що враховуються під час аналізу та синтезу СЗІ;

- істотний взаємозв'язок та взаємозалежність цих показників (вимог), що мають суперечливий характер;

- труднощі, пов'язані з отриманням початкових даних, необхідних для розв'язування задач аналізу та синтезу СЗІ, особливо на ранніх етапах проектування.

Все це значно ускладнює процес аналізу та узагальнення будь-якої інформації щодо системи. Таким чином, існуючі класичні математичні підходи (моделі, статистика, теорія ймовірності, оптимізаційні методи тощо) в умовах некоректної постановки завдання не дають бажаного результату.

Тому виникає необхідність розробки нових підходів, побудови нових моделей, орієнтованих на специфіку процесу забезпечення безпеки інформації в кіберпросторі держави.

ВИКЛАД ОСНОВНОГО МАТЕРІАЛУ

У відповідності до стандарту ISO/IEC 15408-99 [1] розробка моделі захисту припускає виконання наступних операцій:

1. Описати пропоновану сферу, пов'язану з безпекою функціонування в ній інформації.

2. Визначити стратегію протистояння кожній загрозі і сформувати відповідні цілі захисту. На

цьому етапі фактично визначається область дії моделі.

Цілі захисту слід розділити на цілі, досягнення яких покладається на об'єкт оцінки, і меті досягнення яких покладається на середовище.

3. Використати каталог функціональних вимог безпеки з частини 2. Загальні вимоги для специфікації функціональних можливостей, спрямованих на досягнення цілей захисту для кіберпростору держави.

4. Використати каталог вимог довіри і безпеки з частини 3. Загальні вимоги для специфікації компонентів довіри, спрямованих на забезпечення рівня довіри і безпеки, що відповідає цілям безпеки.

5. Розробити логічне обґрунтування того, що вибрані функціональні компоненти і компоненти довіри до захисту підходять для протидії загрозам в кіберпросторі держави.

Зазначимо, що згідно з ISO/IEC 15408-99 частини 2, 3 (ще називають) «Загальні критерії - ЗК» являють собою каталоги вимог безпеки наступних типів:

- функціональні вимоги (Частина 2) - відповідають активному аспекту захисту та висуваються до функцій безпеки ЗК та механізмам, що їх реалізують.

- вимоги довіри (Частина 3) - висуваються до технології та процесу розробки, експлуатації та оцінки ЗК та покликані гарантувати адекватність реалізації механізмів безпеки.

Розглянувши відповідні стандарти та методику Cyber kill chain, нами було запропоновано власну модель захисту кіберпростору, що орієнтована на виконання функції «кіберзахисту» (рис.1). Моделі, орієнтовані на функції систем, прийнято називати функціональними [8].

Для забезпечення виконання даної функції та проектування роботи моделі треба детально проаналізувати роботу їх складових частин і їх взаємодію.

Функціональна модель захисту кіберпростору складається з набору етапів та описує реалізацію переходів від етапу до етапу, а саме порядок та умови переходу починаючи від розвідки і закінчуючи активною протидією на рівні держави.

Вважаємо, що для побудови захищеного кіберпростору треба спочатку проаналізувати та захистити, а потім забезпечувати безпеку створеної СЗІ, виявляючи різноманітні активності зловмисників та реагуючи певним чином на них.

Розглянемо більш детально кожен з етапів нашої моделі.

На першому етапі «Розвідка та виявлення» здійснюється опис середовища безпеки, тобто формується модель загроз, що є повним переліком усіх можливих загроз, які існують або можуть виникнути в даній ситуації, після робляться припущення щодо зловмисника, а отже формується модель порушника.

При складанні моделей враховується середовище в якому функціонує інформація. У нашому випадку це кіберпростір корпорації. Зовнішні порушники за даних умов можуть бути з числа держави.

Формування моделі загроз. Це можна здійснити з використанням теорії ризиків, лінгвістичних термів, теорії графів при одночасній участі експертів, але бажано було б, щоб це був комплексний підхід, оскільки треба визначити не тільки перелік загроз, що існують у кіберпросторі, а і зони ризиків, що в подальшому необхідно для формування вимог довіри до безпеки, міри впливу різних концептів один на одного та на кіберпростір держави в цілому, необхідно також визначити рівень загального ризику.

Складність і трудомісткість аналізу визначається реальними умовами і можливостями її проведення.

Міра впливу трудомісткості обробки експертних даних буде залежить від обсягу і рівня деталізації вхідних даних.

Оскільки переважна більшість впливів на інформацію у випадку технічних й програмних несправностей (відмова, збій й помилка компонентів систем обробки даних, вірусне зараження тощо) носить випадковий характер, можна навіть стверджувати про їх системність, то навмисна причина впливу на інформацію притаманна людині (злочинні дії), тому вона може стати основним джерелом виникнення як суб'єктивних, так і об'єктивних причин.

Аналіз має проводитись за допомогою емпіричного підходу, на основі тривалого збору й

обробки даних про реальні прояви загроз інформації й про розміри того збитку, що при цьому мав би місце. На четвертому етапі «Протидія» побудови захищеного кіберпростору виконується оцінка дієвості запропонованої СЗК.

При рішенні практичних завдань обґрунтування вимог і оцінки дієвості системи захисту кіберпростору виникає природне питання раціонального вибору методу визначення вагових коефіцієнтів з числа існуючих методів.

Принциповими особливостями рішення задачі вибору раціонального варіанту системи захисту, що визначає метод її рішення, є:

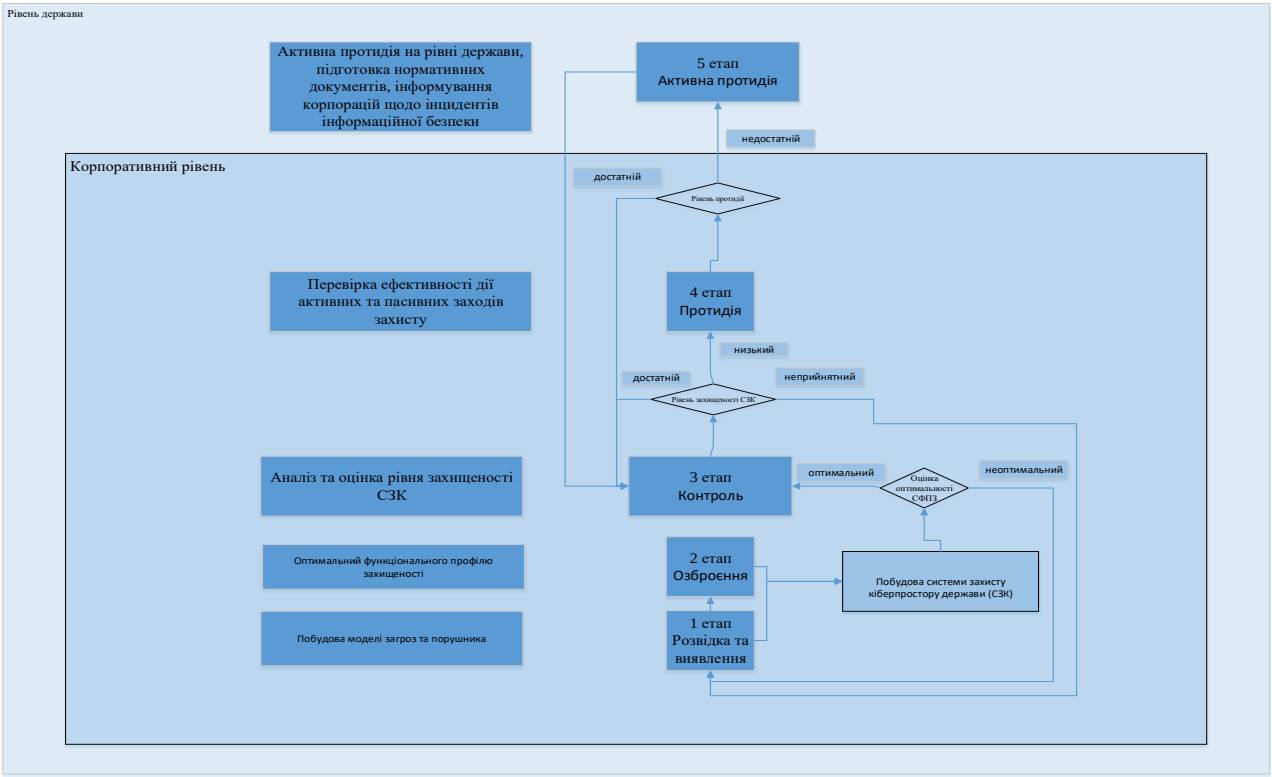


Рис.1. Модель захисту кіберпростору CyberSec

- багатокритеріальність завдання вибору;

- не лише кількісний, але і якісний опис показників якості системи захисту, що задаються у вигляді вимог;

- при якісній постановці завдання вплив на вибір методу її рішення експертної інформації, що визначає перевагу того або іншого показника.

Переважна особливість даного завдання вибору - це якісний характер показників, що трактували їм вимоги, задавалися до системи захисту кіберпростору держави.

Вибір методу рішення задачі залежить від того, в якому виді представлення експертна інформація про перевагу показників, а також від міри їх важливості (рівна і різна важливість вимог).

Відповідно до формування завдання, основними практичними етапами її рішення є:

- розробка методики формування і проведення експертних оцінок;

- розробка принципів і механізмів збору і обробки експертної інформації з характеристики загроз;

- розробка принципів і механізмів збою і обробки експертної інформації з метою визначення важливості виконання функціональних вимог для усунення відпов

Модель порушника. Порушник – це особа, яка помилково, внаслідок необізнаності, цілеспрямовано, за злим умислом або без нього, використовуючи різні можливості, методи та засоби здійснила спробу виконати операції, які призвели або можуть призвести до порушення захищеності інформації. Модель порушника відображає його практичні та потенційні можливості, апріорні знання, час та місце дії тощо. Порушники можуть бути внутрішніми (з числа персона-

лу/користувачів АС), або зовнішніми (з числа сторонніх осіб).

Представники організацій – це особи, що взаємодіють з питань технічного забезпечення (енерго-, водо- теплопостачання, співробітники сервісних центрів тощо) і внаслідок чого мають доступ на територію розташування АС та діють на цій території чи за її межами, або з числа осіб, які зацікавлені в порушенні робіт АС, але доступу у контрольовану зону не мають [2].

На модель порушника впливають такі факти:

- розташування приміщення, в якому розміщено АС, на території, яка має систему надійної фізичної охорони, що практично виключає неконтрольоване проникнення у приміщення сторонніх осіб;

- ретельний підбір співробітників на відповідні посади, що практично виключає виконання ключових функціональних ролей "випадковими" особами; усі співробітники, що задіяні на таких ключових ділянках, повинні мати достатній досвід роботи;

- повсякденний візуальний контроль адміністратора безпеки за станом опечатування системних блоків, що зводить до мінімуму ризик несанкціонованого підключення до ЗОТ;

- обмеження складу встановленого програмного забезпечення та апаратних засобів відповідними затвердженими переліками, ведення паспортів на всі автоматизовані системи, де відбиваються всі відомості щодо складу їх програмного та апаратного забезпечення, а також наявність регулярних звірок паспортних даних з реальними.

Другим етапом «Озброєння» є вибір засобів захисту та побудова системи захисту кіберпростору (СЗК). Для цього нами необхідно розробити підхід, що дозволить зробити оптимальний (який забезпечує найкращі (оптимальні) показники захисту) вибір функціонального профілю кіберзахисту, при цьому витратити на це менше часу, ніж зазвичай.

За умови обрання оптимального функціонального профілю захищеності переходимо до наступного етапу, в іншому випадку повертаємось до початку.

Проведення контролю системи (кібепростору) є третім етапом «Контроль». Для цього здійснюється аналіз основних завдань захисту інформації, визначається реальний рівень зацікавленості суб'єктів, що забезпечують безпеку, у дотриманні всіх вимог до захищеності кожної з властивостей інформації (наприклад, достатній, низький, неприйнятний).ідних загроз (вибір оптимального методу визначення важливості вимог), а також розрахунок взаємозалежних показників; розробка математичної моделі і алгоритму вибору раціонального варіанту побудови системи захисту (раціонального завдання вимог, тобто формування профілю) відповідно до поставленого завдання як завдання математичного моделювання.

Отже, необхідно визначити безпосередньо ефективність застосованих заходів за допомогою деякого кількісного показнику, який відображав би залежність цього показника від ймовірностей запобігання впливу атак на інформацію.

Таким чином, ми зможемо встановити доцільність вживання тих чи тих засобів протидії та зробити висновок про рівень протидії (достатній, недостатній).

На п'ятому етапі «Активна протидія» здійснюється інформування корпорацій щодо інцидентів кібербезпеки та активна протидія на рівні держави, тобто відбувається застосування контрзаходів, також на цьому етапі може відбуватись підготовка відповідних нормативних документів, з метою запобігання повторення реалізації загроз у майбутньому.

ВИСНОВКИ

Отже, для забезпечення представлення міри залежності вимог безпеки заданому рівню якості, необхідно використати низку понять, функцій та математичних моделей. У теорії нечітких великих кількостей є декілька методів побудови функції залежності вимог безпеки заданому рівню якості. Існують методи побудови функції залежності засновані на статистичних даних, на експертних оцінках, на разових оцінках, а також використовуються параметричних підхід. При виборі методу необхідно враховувати, як правило, складність отримання експертної інформації, трудомісткість

алгоритму обробки інформації при побудові функції залежності.

Таким чином, задля побудови захищеного кіберпростору держави має бути застосований цілий комплекс заходів у чітко визначеній послідовності.

ЛІТЕРАТУРА

[1] *ISO/IEC 15408-99* [Електронний ресурс]. – 2020. – Режим доступу до ресурсу: https://infobezlikbez.ru/terminy/standarty/266-mezhdunarodnyj-standart-iso-15408-obshchij-kriterij.

[2] Державна служба спеціального зв'язку та захисту інформації України [Електронний ресурс]. – 2020. – Режим доступу до ресурсу: www.dsszzi.gov.ua.

[3] Домарев В.В. *Безопасность информационных технологий. Системный подход* / Домарев В. В. – К. : ДиаСофт, 2006. – 904 с.

[4] Синенко М.А. Математична модель методів активного захисту інформації / Синенко М.А., Ткач Ю.М. // *Технічні науки та технології : науковий журнал* / Чернігів. нац. технол. ун-т. – Чернігів : ЧНТУ, 2020. – № 2 (20). – С.109-115.

[5] Ткач Ю.М. Моделі систем захисту інформаційної сфери держави // *Сучасна спеціальна техніка*. - 2020. - №2 (61). - С.59-66.

[6] Ткач Ю.М. О развитии киберпространства и его защищенности // Безпека ресурсів інформаційних систем : *збірник тез І Міжнародної науково-практичної конференції* (м. Чернігів 16-17 квітня 2020 р.). – Чернігів : НУЧП, 2020. – С.173-177.

МОДЕЛЬ ЗАЩИТЫ КИБЕРПРОСТРАНСТВА CYBERSEC

В статье предложена модель защиты киберпространства CyberSec, ориентированная на выполнение функции "киберзащиты. Данная модель является функциональной, состоит из пяти этапов, объединяет в себе ряд методов и моделей, является циклической, а потому позволяет создать самоналагоджувану систему защиты в киберпространстве. На первом этапе «Разведка и обнаружения» осуществляется описание среды безопасности, то есть формируется модель угроз, является полным перечнем всех возможных угроз, которые существуют или могут возникнуть в данной ситуации. Вторым этапом «Вооружение» является выбор средств защиты и построение системы защиты киберпространства (СЗК). Проведение контроля системы (кибепростору) является третьим этапом «Контроль». На четвертом этапе «Противодействие» построения защищенного киберпространства выполняется оценка действенности предложенной СЗК. На пятом этапе «Активное противодействие» осуществляется подготовка нормативных документов, информирование корпораций по инцидентов кибербезопасности активное противодействие на уровне государства, то есть происходит применения контрмер. Модель защиты киберпространства CyberSec позволяет на практике построить защищенный киберпространство как отдельной корпорации, так и государства в целом.

Ключевые слова: киберпространство, модель защиты, национальная безопасность, защищенное киберпространство.

CYBER SPACE PROTECTION MODEL CYBERSEC

The article proposes a cyberspace protection model CyberSec, which is focused on performing the function of "cyber protection". This model is functional, consists of five stages, combines a number of methods and models, is cyclical, and therefore allows you to create a self-configuring protection system in cyberspace. At the first stage of "Intelligence and Detection" is a description of the security environment, ie a threat model is formed, which is a complete list of all possible threats that exist or may arise in this situation, then assumptions are made about the attacker, and therefore the violator model is formed.The second stage of "Armament" is the choice of means of protection and construction of a system of cyberspace protection (SSC). To do this, it is necessary to develop an approach that will make the optimal (which provides the best (optimal) protection) choice of the functional profile of cybersecurity, while spending less time than usual. Carrying out control of the system (cyberspace) is the third stage of "Control". To do this, the analysis of the main tasks of information security, determines the real level of interest of security entities in compliance with all requirements for the security of each of the properties of information (for example, sufficient, low, unacceptable). At the fourth stage of "Counteraction" to the construction of secure cyberspace, the evaluation of the effectiveness of the proposed SZK is performed. In the fifth stage of "Active Counteraction", normative documents are prepared, corporations are informed about cybersecurity incidents, active counteraction is carried out at the state level, ie countermeasures are applied. The CyberSec cyberspace protection model allows you to build a secure cyberspace in practice for both an individual corporation and the state as a whole.

Keywords: cyberspace, protection model, national security, protected cyberspace.

Ткач Юлія Миколаївна – доктор педагогічних наук, професор, завідувач кафедри кібербезпеки та математичного моделювання, Чернігівський національний технологічний університет.
E-mail: tkachym79@gmail.com.
ORCID ID: 0000-0002-8565-0525.

Ткач Юлия Николаевна – доктор педагогических наук, профессор, заведующая кафедрой кибербезопасности и математического моделирования, Черниговский национальный технологический университет.

Tkach Yuliia – Doctor of Pedagogical Science, Professor, Head of Department of Cybersecurity and Mathematical Simulation, Chernihiv National University of Technology.

DOI: 10.18372/2410-7840.22.14977
УДК 621.327:681.5

ВНУТРІШНЯ РЕСТРУКТУРИЗАЦІЯ ЯК ЗАСІБ ПІДВИЩЕННЯ ІНФОРМАЦІЙНОЇ БЕЗПЕКИ ПРИ КОДУВАННІ ДАНИХ У СТАТИСТИЧНОМУ ПРОСТОРІ

Володимир Бараннік, Іван Тупиця, Валерій Бараннік, Юрій Бабаенко, Дмитро Жуйков, Валерій Ерошенко

З метою підвищення ефективності статистичного кодування даних з позиції підвищення безпеки даних інформаційного ресурсу запропоновано принципово новий підхід до реструктуризації - внутрішня реструктуризація, суть якої полягає у виявленні закономірностей у внутрішній двійковій структурі даних інформаційного ресурсу за кількісною ознакою. Інструментом для реструктуризації даних виступає кількісна ознака - ознака кількості серій одиниць. Проведено аналіз ефективності застосування розробленого методу внутрішньої реструктуризації з позиції підвищення захисту даних інформаційного ресурсу у статистичному просторі. Інструментом для кодування використовується статистичний підхід на базі класичного алгоритму Хаффмана. Розроблений метод внутрішньої реструктуризації дозволяє вирішувати актуальну науково-прикладну проблему, пов'язану з підвищенням конфіденційності даних з забезпеченням відповідного рівня оперативності доставки інформаційного ресурсу.

Ключові слова: *реструктуризація, кодування, кількісна ознака, безпека даних.*

ВСТУП

Сучасні алгоритми кодування відеоінформації передбачають застосування різного роду трансформацій з метою більш вигідного подання кодованих даних [1-7]. Серед основних, які активно використовуються, слід зазначити наступні:

- перехід з одного колірного простору в інший;

- ортогональні перетворення. Широке поширення отримало дискретно-косинусне перетворення [8-14];

- різного роду перестановки, групування, перетворення даних інформаційних ресурсів (ДІР).

Однак, всі вищевказані види трансформацій є ніщо інше як різновид окремих методів зовнішньої реструктуризації даних. Слід зазначити, що методи зовнішньої реструктуризації даних мають ряд суттєвих недоліків [15-20]. Серед основних недоліків слід зазначити наступні:

- не забезпечують відповідний рівень конфіденційності даних інформаційного ресурсу в умовах дії помилок, в тому числі викликаних внаслідок дії кібератак;

- підвищення складності алгоритмічної реалізації. Це пов'язано з необхідністю виконання значної кількості додаткових математичних (логічних) операцій в залежності від типу трансформації;

- збільшення часу на обробку даних інформаційного ресурсу;

- стратегія позиціонування окремих кодових конструкцій у статистичному просторі кодування (маркерні роздільники, властивість префіксності).

Так існуючі стратегії позиціонування окремих кодових конструкцій у статистичному просторі кодування дозволяють зловмиснику шляхом підбору відповідних законів розподілу ймовірностей появи елементів у повідомленні однозначно декодувати вихідну кодову послідовність. В свою чергу, властивість префіксності дозволяє визначити кодові конструкції, що присвоюються окремим елементам повідомлення у вихідній кодовій послідовності. Тому **актуальним** стає питання пошуку нових підходів до реструктуризації відеоданих, які дозволять мінімізувати (виключити) недоліки методів зовнішньої реструктуризації та підвищити конфіденційність кодованих даних в умовах дії помилок, в тому числі викликаних внаслідок дії кібератак.

У роботах [21-24] представлений принципово новий підхід до реструктуризації даних відеоінформаційного ресурсу - внутрішня реструктуризація. Суть даного підходу полягає у виявленні закономірностей у внутрішній двійковій структурі кодованих даних. Розроблений підхід має ряд переваг в порівнянні з методами зовнішньої рес-

труктуризації [24]. Основні серед них - скорочення часу на обробку відеоданих і виключення необхідності в будь-яких перетвореннях (трансформаціях) ДІР.

З огляду на той факт, що в останні роки спостерігається значне зростання обсягів передаваємих даних, з одного боку, та необхідність забезпечення відповідного рівня конфіденційності - з іншого, основною вимогою, яка пред'являється до ДІР є компактне представлення кодованих даних та підвищення безпеки ДІР з позиції забезпечення відповідного рівня конфіденційності в умовах дії помилок, в тому числі викликаних дією кібератак.

Тому **актуальною науково-прикладною проблемою** є розробка методу реструктуризації для підвищення ефективності статистичного кодування даних відеоінформаційного ресурсу з позиції додаткового скорочення структурної надмірності та забезпечення відповідного рівня конфіденційності передаваємих даних.

Аналіз останніх досліджень і публікацій свідчить про те, що на сьогоднішній день в існуючих стандартах кодування відеоінформації з метою більш вигідного подання кодованих даних активно використовуються методи зовнішньої реструктуризації [16-19]. Слід зазначити, що методи зовнішньої реструктуризації мають ряд суттєвих недоліків, які викладені в працях [20, 21]. Тому з метою усунення недоліків методів зовнішньої реструктуризації був запропонований принципово новий підхід - внутрішня реструктуризація [22]. Використання внутрішньої реструктуризації даних має ряд переваг у порівнянні з зовнішньою з позиції простоти алгоритмічної реалізації і відповідно, скорочення часу на обробки відеоданих [22-24]. Однак актуальним стає питання оцінки розробленого методу реструктуризації даних інформаційного ресурсу як з позиції скорочення структурної надмірності, так і підвищення рівня конфіденційності передаваємих відеоданих в умовах дії помилок, в тому числі викликаних дією кібератак. Тому метою статті є аналіз ефективності застосування методу внутрішньої реструктуризації з позиції підвищення рівня конфіденційності та компактного представлення кодованих даних відеоінформаційного ресурсу.

Аналіз ефективності використання внутрішньої реструктуризації з позиції підвищення рівня конфіденційності та компактного представлення кодованих даних.

В результаті використання методу внутрішньої реструктуризації даних за ознакою кількості серій одиниць (КСО) повідомлення $U(\theta)$ розбивається на множини $U(\lambda_i)$ таким чином, що кожен елемент u_ξ повідомлення $U(\theta)$ може належати тільки одній множині $U(\lambda_i)$ розбиття, тобто множини $U(\lambda_i)$ не перетинаються:

якщо $u_\xi \in U(\lambda_i)$, то $u_\xi \notin U(\lambda_j)$, де $i \neq j$, $\lambda_i, \lambda_j \in \Lambda$ і $U(\lambda_j) \bigcap U(\lambda_i) = \varnothing$.

Таким чином, в результаті проведення кластеризації відбувається групування елементів u_ξ вихідного повідомлення $U(\theta)$ з однаковими значеннями ознаки λ_i КСО у множини $U(\lambda_i)$:

$$U(\theta) \xrightarrow{f_{cl}} \{U(\lambda_1),...,U(\lambda_i),...,U(\lambda_n)\}, \quad (1)$$

де $f_{cl}(u_\xi, \lambda_i)$ - функція групування елементів u_ξ у множини $U(\lambda_i)$ за значенням ознаки λ_i КСО.

Набір різних значень ознаки λ_i КСО, описується наступним виразом:

$$\Lambda = \{\lambda_1,...,\lambda_i,...,\lambda_n\}, \quad (2)$$

де Λ - набір значень ознаки λ_i КСО;

λ_i, λ_n - значення i-ї та n-ї ознаки набору Λ.

Максимальна кількість різних комбінацій η_{λ_i} значень, які може приймати кількісна ознака λ_i КСО, задається наступним виразом:

$$\eta_{\lambda_i \max} = \frac{|u_\xi|_2}{2} + 1, \quad (3)$$

де $\eta_{\lambda_i \max}$ - максимальна кількість комбінацій η_{λ_i} значень, які може приймати ознака λ_i КСО.

Таким чином, набір Λ можливих значень ознаки λ_i КСО, матиме такий вигляд:

$$\Lambda = \left\{ \lambda_i \middle| 0 \le \lambda_i \le \frac{|u_\xi|_2}{2} \right\}, \ \lambda_i \in Z^{\geq} \quad (4)$$

де $Z^{\geq}$ - множина цілих позитивних чисел, включаючи 0.

Максимальна потужність $|\Lambda|$ ознаки λ_i КСО, залежить від довжини $|u_\xi|_2$ послідовності $[u_\xi]_2$ двійкових розрядів $q_{\xi,\alpha}$, $\alpha = \overline{1,|u_\xi|_2}$, якими описується елемент u_ξ повідомлення $U(\theta)$ і задається наступним виразом:

$$|\Lambda|_{max} = \frac{|u_\xi|_2}{2} + 1, \quad (5)$$

де $|\Lambda|_{max}$ - максимальна потужність $|\Lambda|$ ознаки λ_i кількості серій одиниць.

У свою чергу, максимальна кількість множин $U(\lambda_i)$, яка може формуватися в процесі кластеризації елементів u_ξ повідомлення $U(\theta)$, обмежується максимальною потужністю $|\Lambda|_{max}$ ознаки λ_i КСО, і задається наступним виразом:

$$N(U(\lambda_i))_{max} = |\Lambda|_{max}. \quad (6)$$

Як було зазначено вище, кількість різних комбінацій $\eta_{|u_\xi|_2}$, які може приймати окремий елемент u_ξ повідомлення $U(\theta)$, задається наступним виразом:

$$\eta_{|u_\xi|_2} = 2^{|u_\xi|_2}, \quad (7)$$

де $\eta_{|u_\xi|_2}$ - кількість комбінацій, які може приймати елемент u_ξ.

Можлива кількість $\eta_{|u_\xi|_2,\lambda_i}$ комбінацій значень, які може приймати окремий елемент u_ξ повідомлення $U(\theta)$ довжиною $|u_\xi|_2$ при значенні ознаки λ_i КСО, так само [20-21]:

$$\eta_{|u_\xi|_2,\lambda_i} = \frac{(|u_\xi|_2 + 1)!}{(2\lambda_i)!\,(|u_\xi|_2 + 1 - 2\lambda_i)!}, \quad (8)$$

де $\eta_{|u_\xi|_2,\lambda_i}$ - кількість комбінацій значень, які може приймати елемент u_ξ повідомлення $U(\theta)$ при значенні ознаки λ_i.

Аналізуючи результати кластеризації елементів u_ξ повідомлення $U(\theta)$ за ознакою λ_i КСО можна зробити наступні висновки:

1. В процесі кластеризації повідомлення $U(\theta)$ розбивається на множини $U(\lambda_i)$ таким чином, що елемент u_ξ повідомлення $U(\theta)$ може належати тільки одній множині $U(\lambda_i)$ розбиття, тобто множини не перетинаються.

2. Кількість множин $N(U(\lambda_i))$, які формуються в процесі кластеризації елементів u_ξ повідомлення $U(\theta)$, залежить від кількості двійкових розрядів $q_{\xi,\alpha}$, $\alpha = \overline{1,|u_\xi|_2}$, якими задається елемент u_ξ, тобто від його довжини $|u_\xi|_2$. Чим більше довжина $|u_\xi|_2$ елемента u_ξ, тим більше множин $U(\lambda_i)$ може сформуватися в процесі кластеризації.

3. Кластеризація елементів u_ξ повідомлення $U(\theta)$ буде неефективною для випадку, коли набір Λ ознаки λ_i обмежується одним значенням, тобто:

якщо $\Lambda = \{\lambda_i | i = 1\}$, то $|U(\theta)| = |U(\lambda_i)|$.

Це можливо в тому випадку, якщо в результаті кластеризації всі елементи u_ξ повідомлення $U(\theta)$ сформують тільки одному множину $U(\lambda_i)$, тобто якщо буде виконуватися така умова:

$\forall u_\xi \in U(\lambda_i)$, де $i = 1$.

Для такого варіанту розвитку подій потужність повідомлення $U(\theta)$ буде дорівнює потужності множини $U(\lambda_i)$, тобто:

$$|U(\theta)| = |U(\lambda_i)|,$$

де $|U(\lambda_i)|$ - довжина множини $U(\lambda_i)$, тобто кількість елементів у множині;

$|U(\theta)|$ - довжина вхідного повідомлення.

Для оцінки ефективності проведення кластеризації з використанням внутрішньої реструктуризації елементів u_ξ повідомлення $U(\theta)$ за кількісною ознакою λ_i КСО, пропонується розглянути варіант розвитку подій, коли елемент u_ξ має довжину $|u_\xi|_2$ 8 біт, тобто $|u_\xi|_2 = 8$ біт.

На першому етапі визначається кількість можливих комбінацій $\eta_{|u_\xi|_2}$, які може приймати елемент u_ξ повідомлення $U(\theta)$:

$$\eta_{|u_\xi|_2} = 2^{|u_\xi|_2} = 2^8 = 256.$$

Таким чином елемент u_ξ повідомлення $U(\theta)$ може приймати значення від 0 до 255:

$$U(\theta)=\{u_1,...,u_\xi,...,u_\theta\},\theta=\overline{0,255}.$$

На наступному етапі визначається набір $|\Lambda|$ значень ознаки λ_i КСО. Для випадку коли елемент u_ξ має довжину 8 біт, набір $|\Lambda|$ значень ознаки λ_i КСО буде мати наступний вигляд:

$$\Lambda=\{\lambda_1,...,\lambda_i,...\lambda_n\},i=\overline{1,n},\ \lambda_i=\overline{0,4}. \quad (9)$$

Результати кластеризації елементів u_ξ повідомлення $U(\theta)$ з однаковими значеннями ознаки λ_i КСО представлені наступною системою виразів:

$$U(\theta)\xrightarrow{f_{cl}}\begin{cases}\lambda_1=0\to U(\lambda_1),|U(\lambda_1)|=1;\\ \lambda_2=1\to U(\lambda_2),|U(\lambda_2)|=36;\\ \lambda_3=2\to U(\lambda_3),|U(\lambda_3)|=126;\\ \lambda_4=3\to U(\lambda_4),|U(\lambda_4)|=84;\\ \lambda_5=4\to U(\lambda_5),|U(\lambda_5)|=9.\end{cases} \quad (10)$$

Якщо припустити, що повідомлення $U(\theta)$ має вид:

$$U(\theta)=\{u_1,...,u_\xi,...,u_\theta\},\theta=\overline{0,255},$$

а значенням елементів u_ξ відповідає вся можлива кількість $\eta_{|u_\xi|_2,\lambda_i}$ комбінацій, тобто виникає ситуація рівноймовірної появи всіх елементів u_ξ для аналізованого прикладу, то в результаті проведення кластеризації відбувається групування елементів u_ξ вихідного повідомлення $U(\theta)$ (де $\theta=\overline{0,255}$) з однаковими значеннями ознаки λ_i КСО в п'ять множин $U(\lambda_i)$ ($i=\overline{1,5}$), тобто:

$$U(\theta)\xrightarrow{f_{cl}}\{U(\lambda_1),...,U(\lambda_i),...,U(\lambda_5)\}.$$

Процентне співвідношення розподілу елементів повідомлення у множинах за значенням ознаки КСО представлене на рис. 1.

Аналізуючи отримані результати кластеризації елементів u_ξ повідомлення $U(\theta)$ за кількісною ознакою λ_i можна зробити висновок,що основна частина комбінацій $\eta_{|u_\xi|_2,\lambda_i}$ значень елемента u_ξ належить множинам зі значеннями ознаки λ_i, що дорівнює 2 і 3 (49% і 33% від можливої кількості комбінацій), тобто множини $U(\lambda_3)$ і $U(\lambda_4)$.

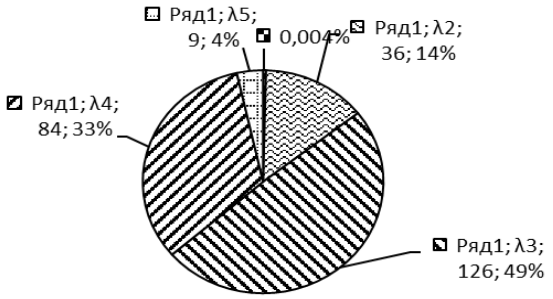


Рис. 1 Результати кластеризації елементів повідомлення

На рис. 1. прийняті наступні позначення:

- множина комбінацій значень елемента зі значенням ознаки, що дорівнює 0;
- множина комбінацій значень елемента зі значенням ознаки, що дорівнює 1;
- множина комбінацій значень елемента зі значенням ознаки, що дорівнює 2;
- множина комбінацій значень елемента зі значенням ознаки, що дорівнює 3;
- множина комбінацій значень елемента зі значенням ознаки, що дорівнює 4.

Таким чином, кластеризація комбінацій значень елементів u_ξ повідомлення $U(\theta)$ за значенням ознаки λ_i, дозволяє знизити потужність $|U(\theta)|$ повідомлення $U(\theta)$ мінімум в 2 рази (для множини $U(\lambda_3)$).

Далі пропонується проаналізувати результати використання запропонованого підходу до кластеризації даних інформаційного ресурсу для тестових слабо-, середньо- та сильнонасиченого зображень (рис. 2 а, б, в). Результати моделювання процесу кластеризації за ознакою кількості серій одиниць для тестових зображень представлені на рис. рис. 4.а)-4.в).

На рис. 3.а)-3.в) прийняті наступні позначення:

- чорним кольором позначені елементи повідомлення зі значенням ознаки КСО що дорівнює 0;
- синім - елементи повідомлення зі значенням ознаки КСО, що дорівнює 1;
- червоним - елементи повідомлення зі значенням ознаки КСО, що дорівнює 2;
- жовтим - елементи повідомлення зі значенням ознаки КСО, що дорівнює 3;
- зеленим - елементи повідомлення зі значенням ознаки КСО, що дорівнює 4.

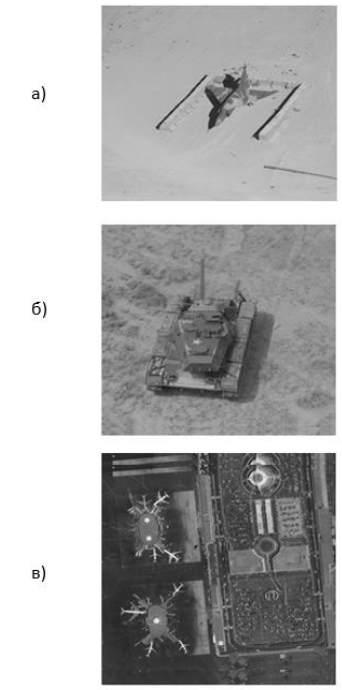


Рис. 2 Тестові слабо-, середньо- та сильнонасичені напівтонові зображення

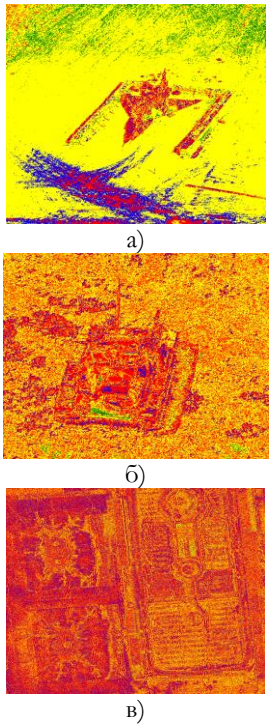


Рис. 3 Результати кластеризації елементів тестових слабо-, середньо- та сильнонасиченого зображень

На рис. 4.а) - 4.в) представлені у вигляді відповідних діаграм кількісні оцінки розподілу елементів u_ξ тестових зображень за кількісною ознакою λ_i КСО.

Аналізуючи результати кластеризації елементів u_ξ повідомлення $U(\theta)$, які представлені на рис. 4.а) - 4.в) можна зробити наступні висновки:

1) В результаті кластеризації елементів u_ξ повідомлення $U(\theta)$ за ознакою λ_i КСО для слабонасиченого зображення формуються 4 множини $U(\lambda_i)$ (рис. 4.а). Більша частина елементів u_ξ повідомлення $U(\theta)$ увійшла в множину $U(\lambda_4)$ зі значенням ознаки $\lambda_i = 3$, що склало 79%.

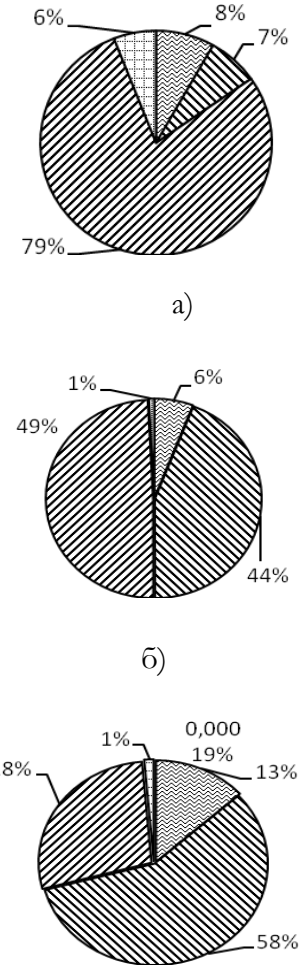


Рис. 4 Процентне співвідношення розподілу елементів тестових зображень у множини за значенням ознаки

На рис. 4.а) - 4.в) прийняті наступні позначення:

- множина елементів зі значенням ознаки, що дорівнює 0;
- множина елементів зі значенням ознаки, що дорівнює 1;
- множина елементів зі значенням ознаки, що дорівнює 2;
- множина елементів зі значенням ознаки, що дорівнює 3;
- множина елементів зі значенням ознаки, що дорівнює 4.

Таким чином, кластеризація елементів u_ξ повідомлення $U(\theta)$ за ознакою λ_i КСО для слабонасиченого зображення дозволяє знизити потужність $|U(\theta)|$ повідомлення $U(\theta)$ від 22% (для множини $U(\lambda_3)$) до 95% (для множини $U(\lambda_4)$).

2) Кластеризація елементів u_ξ повідомлення $U(\theta)$ за ознакою λ_i КСО для середньонасиченого зображення дозволяє знизити потужність $|U(\theta)|$ повідомлення $U(\theta)$ від 51% (для множини $U(\lambda_3)$) до 99% (для множини $U(\lambda_4)$).

3) Кластеризація елементів повідомлення за ознакою КСО для сильнонасиченого зображення дозволяє знизити потужність повідомлення від 42% (для множини $U(\lambda_3)$) до 99,99% (для множини $U(\lambda_1)$).

4) В результаті кластеризації елементів u_ξ повідомлення за ознакою КСО для слабо-, середньо- та сильнонасиченого зображень формуються від 4 до 5 множин.

5) Кластеризація елементів повідомлення $U(\theta)$ за ознакою λ_i КСО дозволяє знизити потужність $|U(\theta)|$ повідомлення $U(\theta)$ від 22% до 99,99% для кожного з аналізованих прикладів.

Використання розробленого методу реструктуризації даних дозволяє підвищити ефективність статистичного кодування з позиції додаткового скорочення структурної надмірності кодової послідовності. Результати статистичного кодування тестових зображень наведені на рис.5.

Таким чином, для досліджуваних прикладів обсяг вихідних даних скорочується від 1,77 разів - для слабонасиченого до 3,01 разів - для сильнонасиченого зображень. Результатом використання розробленого методу реструктуризації для кодування даних у статистичному просторі є трансформація стратегії позиціонування окремих кодових конструкцій у загальній кодовій послідовності.

Результатом трансформації є:

- присвоєння різним за значенням елементам повідомлення однакових кодових конструкцій;

- відсутність властивості префіксності для кодових конструкцій вихідної кодової послідовності.

Таким чином, однозначне декодування даних інформаційного ресурсу можливо лише при наявності інформації, якій з множин належить кодований елемент.

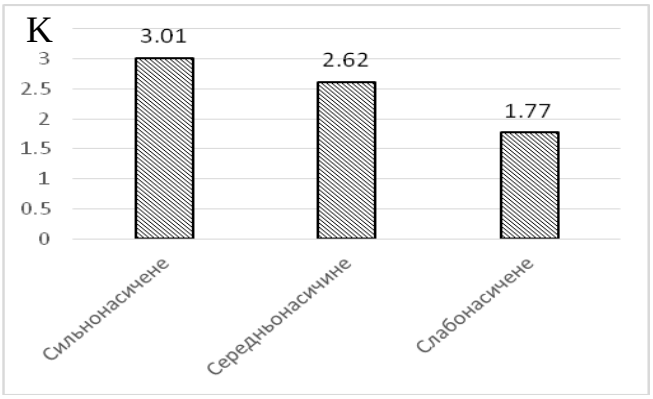
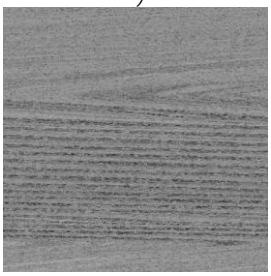


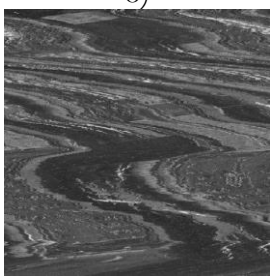
Рис. 5 Діаграма залежності коефіцієнта скорочення обсягу вихідної кодової послідовності від ступеня насиченості зображень

На рис. 6.а) - 6.в) представлені результати реконструкції тестових зображень в умовах дії помилок при декодуванні кодової послідовності.

а)



б)



в)

Рис. 6 Результати реконструкції тестових слабо-, середньо- та сильнонасиченого зображень

Аналіз результатів декодування тестових зображень, представлений на рис.6 свідчить про те, що помилки, які виникають при реконструкції зображень призводять до значного спотворення (руйнування) вихідних даних.

Таким чином внутрішня реструктуризація дозволяє підвищити конфіденційність передаваємих даних. Це пов'язано з тим, що для однозначного декодування кодової послідовності зловмиснику необхідно володіти інформацією якій з множин належить той чи інший елемент.

Порушення властивості префіксності кодової послідовності, що формується у статистичному просторі множини не дозволяє однозначно відокремити кодові констукції, що присвоюються окремим елементам.

ВИСНОВКИ

Аналізуючи результати ефективності використання розробленого методу внутрішньої реструктуризації даних за кількісною ознакою з позиції більш компактного представлення та забезпечення відповідного рівня конфіденційності передаваємих даних можна зробити наступні висновки:

- використання розробленого методу реструктуризації ДІР дозволяє підвищити ефективність статистичного кодування з позиції скорочення довжини вихідної кодової послідовності. Так для аналізованих прикладів обсяг вихідних даних скорочується від 1,77 - для слабонасичених до 3,01 разів - для сильнонасичених зображень;

- результатом кластеризації є підвищення рівня конфіденційності передаваємих даних. Це пов'язано з тим, що різним за значенням елементам повідомлення присвоюються однакові кодові конструкції.

Тому однозначне декодування зловмисником внаслідок несанкціонованого доступу можливе лише за наявності інформації якій з множин належить той чи інший елемент.

Також для зловмисника ускладнюється процес декодування і тим фактором, що кодова послідовність, що формується в процесі кодування елементів повідомлення з використанням кластеризації за кількісною ознакою не володіє властивістю префіксності.

ЛІТЕРАТУРА

[1] Gonzalez R., Woods R., "Digital Image Processing", M .: Technosphere, 2005, 1073 p.

[2] A. Skodras, C. Christopoulos, and T. Ebrahimi. The jpeg 2000 still image compression standard. IEEE Signal processing magazine, 18 (5): 36–58, 2001. J. Miano, "Formats and image compression algorithms in action", K .: Triumph, 2013, 336 p.

[3] Pratt W.K., Chen W.H., Welch L.R., "Slant transform image coding. Proc. Computer Processing in communications. " New York: Polytechnic Press, 1969, pp. 63-84.

[4] J. Miano. Formats and image compression algorithms in action [Text] K.: Triumph, 2013. — 336 p.

[5] D. Taubman and M. Marcellin, JPEG2000 Image Compression Fundamentals Standards and Practice, Boston: Kluwer:Springer, pp. 777, 2002.

[6] Ming Huwi. Horng, "Vector quantization using the firefly algorithm for image compression", Expert Systems with Applications, vol. 39, no. 1, pp. 1078-1091, 2012.

[7] Sindeev M., Konushin A., Rother C., Alpha-flow for video matting, "Technical Report, 2012. pp. 41–46.

[8] Wallace GK, "The JPEG Still Picture Compression Standard," Communication in ACM, vol. 34., No. 4, 1991. - pp. 31-34.

[9] S. Wang, X. Zhang, X. Liu, J. Zhang, S. Ma and W. Gao, "Utility Driven Adaptive Preprocessing for Screen Content Video Compression," in IEEE Transactions on Multimedia, vol. 19, no. 3, 2017 pp. 660-667.

[10] Y. Zhang, S. Negahdaripour and Q. Li, "Error-resilient coding for underwater video transmission," OCEANS 2016 MTS / IEEE Monterey, CA, 2016, pp. 1-7.

[11] O. Stankiewicz, K. Wegner, D. Karwowski, J. Stankowski, K. Klimaszewski and T. Grajek, "Encoding mode selection in HEVC with the use of noise reduction," International Conference on Systems, Signals and Image Processing (IWSSIP) , Poznan, 2017, pp. 1-6.

[12] Xuan Zhu, Li Liu, Peng Jin, Na Ai, "Morphological component decomposition combined with compressed sensing for image compression", 2016 IEEE International Conference on Information and Automation (ICIA), Ningbo, China.

[13] Arnob Paul, Tanvir Zaman Khan, Prajoy Podder, Rafi Ahmed, M. Muktadir Rahman, Mamdudul Haque Khan, "Iris image compression using wavelets transform coding", 2015 2nd International Conference on Signal Processing and Integrated Networks (SPIN), Noida, India, pp. 544-548.

[14] Okuwobi Idowu Paul and YH Lu, "A New Approach in Digital Image Compression Using Unequal Error Protection (UEP)", Applied Mechanics & Materials, no. 704, pp. 403-407, 2015.

[15] Zhu Shuyuan, B. Zeng and M. Gabbouj, "Adaptive sampling for compressed sensing based image

compression", Journal of Visual Communication & Image Representation, no. 30, pp. 94-105, 2015.

[16] Barannik V.V. Fundamentals of the theory of structurally combinatorial steganographic coding: monograph / V.V. Barannik, D.V. Barannik, A.E. Bekirov. - X .: Publisher "Leader", 2017. - 256 p.

[17] Vladimir.V. Barannik ; M.P. Karpinski ; V.V. Tverdokhleb ; Dmitry.V. Barannik ; V.V. Himenko ; Marek Aleksander The technology of the video stream intensity controlling based on the bit-planes recombination. 2018 IEEE 4th International Symposium on Wireless Systems within the International Conferences on Intelligent Data Acquisition and Advanced Computing Systems (IDAACS-SWS), 20-21 Sept. 2018, Lviv, Ukraine.

[18] Ghadah Al-Khafaji and H. Al-Khafaji, "Medical Image Compression using Wavelet Quadrants of Polynomial Prediction Coding & Bit Plane Slicing", vol. 4, no. 6, 2014.

[19] Vladimir Barannik ; Dmitry Barannik ; Vadym Fustii ; Maksym Parkhomenko Evaluation of Effectiveness of Masking Methods of Aerial Photographs. 2019 3rd International Conference on Advanced Information and Communications Technologies (AICT), 2-6 July 2019, Lviv, Ukraine, Ukraine.

[20] J. Lee, S. Cho, and S.-K. Beack. Context-adaptive entropy model for end-to-end optimized image compression. arXiv preprint arXiv: 1809.10452, 2018.

[21] Y. Patel, S. Appalaraju, and R. Manmatha. Human perceptual evaluations for image compression. arXiv preprint arXiv: 1908.04187, 2019.

[22] O. Rippel and L. Bourdev. Real-time adaptive image compression. In Proceedings of the 34th International Conference on Machine Learning-Volume 70, JMLR. org, 2017. – pp. 2922-2930.

[23] S. Santurkar, D. Budden, and N. Shavit. Generative compression. In 2018 Picture Coding Symposium (PCS), IEEE, 2018. – pp. 258-262.

[24] Barannik V.V., Ryabukha Yu. N., Tverdokhleb V.V., Barannik D.V.: Methodological basis for constructing a method for compressing of transformants bit representation, based on non-equilibrium positional encoding. In: Advanced Information and Communication Technologies (AICT), 2017 2nd International Conference, 2017. - pp.188-192.

[25] Li Ji, Zhang Zhi-Guo, Xiao Bin, Yang Ze-Lin and Wang Dun, "Based on discrete orthogonalt chebichef transform for image compression", Classification No. of Chinese Library Classification: TP391 [A], 2013. - pp. 12-4261-06

[26] Vladimir Barannik, Valeriy Barannik, Dmytro Havrylov, Anton Sorokun.: Development Second and Third Phase of the Selective Frame Processing Method. In.: 2019 3rd International Conference on Advanced Information and Communications Technologies (AICT), 2019. - pp. 54-57.

[27] Barannik, V.V., Ryabukha, Yu.N. and Kulitsa, O.S.: The method for improving security of the remote video information resource on the basis of intellectual processing of video frames in the telecommunication systems. In: Telecommunications and Radio Engineering. Vol. 76. No 9. , 2017. - pp. 785-797.

[28] Barannik, V. and Barannik, N. and Ryabukha, Yu. and Barannik, D.: Indirect Steganographic Embedding Method Based On Modifications of The Basis of the Polyadic System. In.: 15th IEEE International Conference on Modern Problems of Radio Engineering, Telecommunications and Computer Science (TCSET'2020), 2020. - pp. 699-702.

[29] Barannik, V. and Barannik, V.: Binomial-Polyadic Binary Data Encoding by Quantity of Series of Ones. In.: 15th IEEE International Conference on Modern Problems of Radio Engineering, Telecommunications and Computer Science (TCSET'2020), 2020. - pp. 775-780.

[30] Vladimir Barannik, Tatyana Belikova, Pavlo Gurzhii.: The model of threats to information and psychological security, taking into account the hidden information destructive impact on the subconscious of adolescents. In 2019 IEEE International Conference on Advanced Trends in Information Theory (ATIT), 2019. - pp. 656 – 661.

[31] Vladimir Barannik, Denys Tarasenko.: Method coding efficiency segments for information technology processing video. In.: 2017 4th International Scientific-Practical Conference Problems of Infocommunications. Science and Technology (PIC S&T), 2017. - pp. 551-555.

[32] Barannik V.V., Krasnoruckiy A., Hahanova A. The positional structural-weight coding of the binary view of transformants, Proceedings of the International Conference on East-West Design and Test Symposium (EWDTS), September 2013. - pp. 1-4.

[33] Volodymyr Barannik; S.S. Shulgin.: The method of increasing accessibility of the dynamic video information resource. In.: 2016 13th International Conference on Modern Problems of Radio Engineering, Telecommunications and Computer Science (TCSET), 2016. - pp. 621-623.

ВНУТРЕННЯЯ РЕСТРУКТУРИЗАЦИЯ КАК СРЕДСТВО ПОВЫШЕНИЯ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ ПРИ КОДИРОВАНИИ ДАННЫХ В СТАТИСТИЧЕСКОМ ПРОСТРАНСТВЕ

С целью повышения эффективности статистического кодирования данных с позиции повышения безопасности данных информационного ресурса предложен принципиально новый подход к реструктуризации - внутренняя реструктуризация, суть которой заключается в выявлении закономерностей во внутренней двоичной структуре данных информационного ресурса по количественному признаку. Инструментом для реструктуризации данных выступает количе-

ственный признак - признак количества серий единиц. Проведен анализ эффективности применения разработанного метода внутренней реструктуризации с позиции повышения защиты данных информационного ресурса в статистическом пространстве. Инструментом для кодирования используется статистический подход на базе классического алгоритма Хаффмана. Разработанный метод внутренней реструктуризации позволяет решать актуальную научно-прикладную проблему, связанную с повышением эффективности статистического кодирования с позиции сокращения структурной избыточности и повышения уровня безопасности данных информационного ресурса.

Ключевые слова: реструктуризация, кодирование, количественный признак, безопасность данных.

INTERNAL RESTRUCTURING AS A MEANS OF INCREASING INFORMATION SECURITY WHEN CODING DATA IN A STATISTICAL SPACE

In order to improve the efficiency of statistical data coding from the standpoint of increasing the data security of an information resource, a fundamentally new approach to restructuring is proposed - internal restructuring, the essence of which is to identify patterns in the internal binary data structure of an information resource by a quantitative criterion. The tool for data restructuring is a quantitative feature - a feature of the number of series of units. The analysis of the effectiveness of the application of the developed method of internal restructuring from the standpoint of increasing the data protection of an information resource in the statistical space is carried out. The coding tool uses a statistical approach based on the classical Huffman algorithm. The developed method of internal restructuring allows us to solve an urgent scientific and applied problem related to increasing the efficiency of statistical coding from the standpoint of reducing structural redundancy and increasing the level of data security of an information resource.

Keywords: restructuring, coding, quantitative sign, data security.

Бараннік Володимир Вікторович, доктор технічних наук, професор, професор кафедри штучного інтелекту і програмування, Харківського національного університету імені В.Н. Каразіна.
E-mail: vvbar.off@gmail.com.
Orcid ID: 0000-0002-2848-4524.
Баранник Владимир Викторович, доктор технических наук, профессор, профессор кафедры искусственного интеллекта и программирования, Харьковского национального университета имени В.Н. Каразина.
Barannik Vladimir, Doctor of Technical Sciences, Professor, Professor Department, V.N. Karazin Kharkiv National University.

Тупиця Іван Михайлович, викладач, Харківський національний університет Повітряних Сил імені Івана Кожедуба.
E-mail: ivan20081982@gmail.com.
Orcid ID: 0000-0001-6806-4914.
Тупица Иван Михайлович, преподаватель, Харьковский национальный университет Воздушных Сил имени Ивана Кожедуба.
Tupitsya Ivan, Combat use of ASC department, Ivan Kozhedub Kharkiv National Air Force University.

Бараннік Валерій Володимирович, студент Харківського національного університету радіоелектроніки.
E-mail: valera462000@gmail.com.
Orcid ID: 0000-0003-3516-5553.
Баранник Валерий Владимирович, студент Харьковского национального университету радіоелектроніки, Харків, Україна.
Barannik Valery, student, Kharkov National University of Radio Electronics, Kharkiv, Ukraine.

Бабенко Юрій Михайлович, аспірант кафедри кібербезпеки та захисту інформації Київського національного університету імені Тараса Шевченко.
E-mail: babenkomahalych@gmail.com.
orcid.org/0000-0002-8115-3329.
Бабенко Юрий Михайлович, аспирант кафедры кибербезопасности и защиты информации Киевского национального университета имени Тараса Шевченко.
Babenko Yurii, ostgraduate of Department of cybersecurity and information hijacking Taras Shevchenko National University of Kyiv.

Жуйков Дмитро Борисович, доцент кафедри заочної підготовки Харківського національного університету Повітряних Сил.
E-mail: 1967dbz@gmail.com.
Orcid ID: 0000-0002-7064-1343.
Жуйков Дмитрий Борисович, доцент кафедры заочной подготовки Харьковского национального университета Воздушных Сил.
Dmytro Zhuikov, Ivan Kozhedub Kharkiv National Air Force University, Ukraine, Kharkiv.

Ерошенко Валерій Петрович, кандидат технічних наук, викладач кафедри Харківського національного університету Повітряних Сил ім. І. Кожедуба, Харків, Україна.
E-mail: wpEroshenko59@gmail.com.
Orcid ID: 0000-0003-3175-6444
Ерошенко Валерий Петрович, кандидат технических наук, преподаватель Харьковского национального университета Воздушных Сил имени Ивана Кожедуба, Харьков
Yroshenko Valerii, Candidate of Technical Science, Teacher of Ivan Kozhedub Kharkiv National Air Force University Kharkiv, Ukraine.

DOI: 10.18372/2410-7840.22.14977
УДК 654.071

АНАЛІЗ АТАК, ЩО ВИКОРИСТОВУЮТЬСЯ КІБЕРЗЛОЧИНЦЯМИ ПІД ЧАС ПАНДЕМІЇ COVID 19

Віталій Сусукайло, Іван Опірський, Андріян Піскозуб, Ростислав Волошин, Олег Друзюк

За даними Всесвітньої організації охорони здоров'я, пандемія визначається як „поширення нової хвороби у всьому світі ". З точки зору кібербезпеки це означає - катастрофа. Під час катастроф кількість кіберзлочинців зростає щодня. По мірі того як все більше висококваліфікованих фахівців з кібербезпеки долучається до блакитної команди, щодня запускається все більше шкідливих аплікацій, приблизно 230000 нових зразків шкідливих програм на день, згідно з інформацією дослідників з PandaLabs. Пандемію можна розглядати як подію, яка може призвести до виконання планів безперервності бізнесу або реалізації заходів з аварійного відновлення. Протягом цього часу слід аналізувати зростаючу кількість загроз кібербезпеки та визначати застосовні заходи безпеки. Існує також багато проблем з адміністративною інформаційною безпекою, які також слід враховувати. У цій статті розкрито основні питання щодо моніторингу інфраструктури, а також забезпечення високого рівня управління вразливостями та реагування на інциденти. Наведено заходи управління, які необхідно використовувати у SOC центрах, а також представлено поглиблений аналіз векторів атак та заходів безпеки, які можна застосувати для їх запобігання. Визначено, що віддалений моніторинг безпеки повинен бути зосереджений на аналізі подій з кінцевих точок за допомогою хост-систем виявлення вторгнень, рішень для виявлення і реагування на кінцеві точки, а також програмного забезпечення для забезпечення безпеки кінцевих точок, яке дозволяє дистанційно керувати і агрегувати події в центральній консолі.

***Ключові слова**: вектор атаки, шкідливе програмне забезпечення, реагування на інциденти, безперервність бізнесу, Cyber Kill Chain.*

Вступ. Ситуація з пандемією COVID-19 змінила принципи роботи. Люди були стурбовані, і з занепокоєнням вони хотіли отримати інформацію, відчуття безпеки та підтримки. У той же час організовані злочинні групи використовують страх, невпевненість і сумніви, пов'язані з COVID-19, роблячи вразливими людей та цілі компанії. Через цю ситуацію приватні організації та державні установи почали забезпечувати дистанційну оперативну діяльність, що спричиняє багато проблем із безпекою.

У той же час критично важливі об'єкти інфраструктури, такі як експлуатаційні, інфраструктурні стаціонарні активи та IoT, стали першочерговими об'єктами для суб'єктів загрози. Поглиблений аналіз поверхні атаки та стану кібербезпеки під час COVID-19 визначив чотири основні виклики кібербезпеки, описані нижче.

- Недостатній захист критичної інфраструктури. Наступна проблема може призвести до збільшення кількості приватних та державних SOC центрів. Крім того, це може вплинути на якість послуг, які надають SOC центри, експерти з кібербезпеки повинні забезпечити високий рівень моніторингу безпеки, управління вразливістю та реагування на інциденти, щоб гарантувати відсутність перебоїв у роботі критичної інфраструктури та захист персональних даних клієнтів.
- На сьогодні недостатня кількість фахівців з кібербезпеки є актуальною проблемою, і ситуація з пандемією показує, наскільки важливо мати кваліфікованих експертів, що забезпечує захист інформації в державних та приватних організаціях. Наступна проблема може призвести до вдосконалення системи освіти в галузі кібербезпеки та збільшення кількості закладів, які можуть підготувати кваліфікованих фахівців [2].
- Необхідно вдосконалити процеси безперервності бізнесу та заходи з аварійного відновлення. Ситуація з COVID-19 показала, що багато оперативних заходів з безпеки неможливо виконувати віддалено в організаціях, які будують свої рішення

щодо безпеки на локальних активах інфраструктури. Очікується, що кінцевим користувачам, що використовують ПЗ в якості моделі обслуговування, буде надаватися більше вирішень з кібербезпеки, а також буде застосовуватися більш агентоорієнтований підхід.

- Недостатній рівень обізнаності в галузі соціальної інженерії. Хакери озброїлись картою COVID-19, маніпулювали інформацією ВООЗ та інших медичних організацій, створили багато заражених фішинг-сайтів, що поставило під загрозу мільйони користувачів. Наступна проблема може призвести до збільшення кількості державних програм інформування про соціальну інженерію.

Аналіз векторів атак. Найпоширенішими та найпопулярнішими нападами під час пандемії є група атак соціальної інженерії, яка використовує страхи людей або співчуття (рис.1).

Для аналізу були відібрані атаки соціальної інженерії під час COVID-19, в яких використовувалося шкідливе програмне забезпечення Alozurt.

Рис. 1 Атаки з використанням соціальної інженерії

На етапі розвідки зловмисник досліджував найпоширеніші страхи кінцевих користувачів - необхідність інформації про COVID-19. Найпопулярнішими ресурсами, які користувачі відвідали під час пандемії, були інтерактивні карти COVID-19, які передавали інформацію про стан зараження в різних країнах.

Тож зловмисники використовували страхи людей і створили тисячі підроблених карт COVID 19 та веб-сайти з неправдивими новинами.

Після вибору способу зараження зловмисники вибирали шкідливе програмне забезпечення.

Популярним був Azolurt - це шкідливе програмне забезпечення для крадіжки інформації, яке націлене на викрадення облікових даних та облікових записів[5].

Для доставки Azolurt було використано зловмисне програмне забезпечення для підроблених онлайн-карт COVID-19 або фішинг-листів. Шкідливе програмне забезпечення було доставлено у вигляді документа Microsoft Office, що полегшило зловмисникам маніпуляції зі страхами кінцевих користувачів. Коли користувач відкрив файл, шкідливе програмне забезпечення використовує вразливість CVE-2017-11882 Microsoft Office Equation Editor і завантажує шкідливий файл. Потім зловмисний файл вносить зміни в реєстр, що відповідає за автозапуск.

На завершальному етапі зловмисне програмне забезпечення запускається самостійно, а потім приступає до викрадення персональних даних та підключення до серверів керування та управління. По-

тім зловмисний файл запускає cmd.exe, щоб видалити себе через 3-секундний тайм-аут.

Перший вектор атаки показує, як суб'єкти загрози використовують страхи кінцевих користувачів. Другий вектор атаки, який складається з методів соціальної інженерії під час пандемії, покладається на людське співчуття. Зловмисники створили кілька підроблених благодійних веб-сайтів, щоб обдурити людей та заробити кошти або скомпрометувати активи кінцевих користувачів шкідливим програмним забезпеченням.

Третім вектором атаки, який використовує методи соціальної інженерії, є неправдиві інтернет-магазини, які продають ліки або медичні товари. Метою таких веб-сайтів є фінансова вигода. Для запобігання цьому типу атак засоби управління інформаційною безпекою повинні посилатися на стратегію оборонного захисту інформації. Першим засобом контролю, який можна застосувати для уникнення атак соціальної інженерії, є сеанси інформування або тренінги, які містять приклади підроблених COVID 19 ресурсів, таких як coronavirus-map.com та офіційні ресурси, які необхідно використовувати, такі як www.who.int. Результати обізнаності можуть бути протестовані шляхом проведення фальшивої фішинг-атаки на вибрану групу осіб, які відвідували тренінг. Другим контролем може бути включення модулів виявлення фішингу та сканування зловмисного програмного забезпечення електронної пошти в програмному забезпеченні безпеки кінцевої точки та в налаштуваннях системи доставки електронної пошти - GSuite, Office 365 вже має можливості виявлення та запобігання фішингу [7]. Також, щоб уникнути цього типу атак, можна встановити розширення браузера VirusTotal, що забезпечить фільтрацію веб-сайтів.

Атаки, які переривають критично важливі бізнес-функції. Щоб забезпечити безперервність бізнесу під час пандемії, комерційним організаціям потрібно надати високоякісні можливості дистанційної роботи де це можливо. Ось чому найпоширенішими службами під час віддаленої роботи є послуги VPN, SaaS, хмарні технології, програмне забезпечення для проведення конференцій тощо. Для аналізу векторів атак на важливий бізнес були обрані функції атак на програмне забезпечення для конференцій Zoom.

Протягом березня-квітня 2020 року програмне забезпечення Zoom було основною ціллю зловмисників. Щодня джерела інформації про загрози інформували про проблеми безпеки та конфіденційності програмного забезпечення Zoom, близько 500000 облікових записів Zoom, проданих на хакерських форумах, у настільних додатках Zoom виявлялись численні критичні вразливості, хмарна служба, яка зберігала записані конференції, була скомпрометована. Але найпоширенішою проблемою було Zoombombing, яке дозволяє зловмисникам приєднуватися до незахищених зустрічей Zoom. Наступна проблема могла призвести до компрометації вмісту зустрічі, переривання критичної для бізнесу зустрічі, зараження кінцевих користувачів шкідливим програмним забезпеченням.

Найгірший сценарій, який може вплинути на організацію, - це зараження кінцевих користувачів шкідливим програмним забезпеченням через обмін ним в чаті Zoom. Коли зловмисне програмне забезпечення надається під законним псевдонімом користувача, інші користувачі можуть завантажити його та встановити на свої активи. На рис. 2. представлена атака через програмне забезпечення для проведення конференцій Zoom, що описує можливий вектор атаки, який використовується суб'єктом загрози для компрометації організації через незахищених користувачів.

Першим контролем інформаційної безпеки, щоб уникнути атак через програмне забезпечення для проведення конференцій Zoom, має бути встановлений захист паролем для конференц-залів. Постійні оновлення підписів шкідливого програмного забезпечення та глибоке сканування файлів можуть допомогти виявити шкідливі файли, якими користуються Zoom. Крім того, рекомендується використовувати програмне забезпечення виявлення та реагування на кінцеві точки, таке як

Wazuh, для виявлення аномалій потенційно зараженого активу.

Напади на критично важливу інфраструктуру. Ще однією основною мішенню для нападників під час пандемії стали лікарні та організації охорони здоров'я. Зловмисники використовували стандартні методи для компрометації інфраструктури та IoT-пристроїв через вразливості в програмному забезпеченні і службах рис.3. Початковий вектор атаки був здійснений через сторонні служби. У квітні 2020 року було зламано 25 000 акаунтів ВООЗ, Фонду Гейтса, NIH. Як повідомили наступні організації, атака проводилася на ресурси, не пов'язані з організаціями охорони здоров'я, а на сервіси, де ці облікові записи використовувалися для реєстрації. Щоб уникнути атак через сторонні сервіси, організації, які інтегрують кілька сервісів в одну інфраструктуру, повинні встановити процес перевірки безпеки постачальників послуг.

Профілактичні заходи. Пандемія ускладнює оперативну діяльність по забезпеченню безпеки. Фахівцям з кібербезпеки стає все важче виявляти загрози на ранній стадії і оперативно реагувати на них. Процеси забезпечення безпеки в організаціях під час пандемії повинні бути трансформовані. Для операційних центрів безпеки стало неможливим відслідковувати мережеві загрози, які можуть впливати на їх операційні активи і кінцевих користувачів. Ще одна проблема, викликана пандемічної ситуацією, контроль робочих станцій кінцевих користувачів. Для організацій, які використовують локальну інфраструктуру, стає неможливим управляти робочою станцією без активного VPN-з'єднання.

Під час епідемії необхідно забезпечити дистанційне керування кінцевими точками. Фахівці з кібербезпеки повинні мати можливість віддалено впроваджувати конфігурації безпеки. Повинні застосовуватися такі політики безпеки, як політика паролів, політика блокування облікових записів і тощо. Операційна система і служби повинні оновлюватися автоматично. Віддалене підключення до ресурсів кінцевого користувача також має бути забезпечено для виконання дій з діагностики системи. Необхідно вдосконалювати стратегії зміцнення кінцевих точок. Для забезпечення безпечних умов роботи користувачів, що виконують свої повсякденні обов'язки з домашнього кінцевого програмного забезпечення безпеки повинні бути налаштовані так, щоб ними можна було керувати з "хмари" або як програмне забезпечення через центр управління послугами. Додатки для захисту від шкідливого ПО повинні отримувати сигнатури безпосередньо від джерел інформації про загрози постачальників. Програмне забезпечення для виявлення кінцевих точок і реагування на них з "хмарним" центром управління повинно бути встановлено на кінцевих точках кінцевих користувачів. Правила брандмауера кінцевих точок повинні бути налаштовані на дозвіл віддалених підключень до робочих станцій тільки з IP-адресу організації; непотрібні порти повинні бути відключені. Необхідно встановити рішення веб-фільтрації та управління додатками, які можуть мінімізувати ризик компрометації кінцевого користувача невідомими ресурсами. Це дозволить фахівцям з кібербезпеки здійснювати постійний моніторинг безпеки і виявляти аномалії на робочих станціях кінцевих користувачів.

Пандемія є проблемою для державних і приватних організацій, яка може привести до активації плану забезпечення безперервності бізнесу. Цифрове перетворення і міграція в хмару повинні бути розглянуті організаціями, які покладаються на локальну інфраструктуру. Вектор атак, що описує дії, які ставлять під загрозу критично важливі бізнес-функції, показує, наскільки важливо застосовувати конфігурації безпеки, що надаються виробником. У випадку з додатком Zoom необхідно було застосувати захист паролем для зборів, застосувати конфігурацію, яка вимагає, щоб учасники зборів були прийняті хостом, тимчасово не зберігати записані відео в хмарі Zoom і застосувати MFA.

Рис. 2 Атака за допомогою програмного забезпечення для проведення конференцій Zoom.

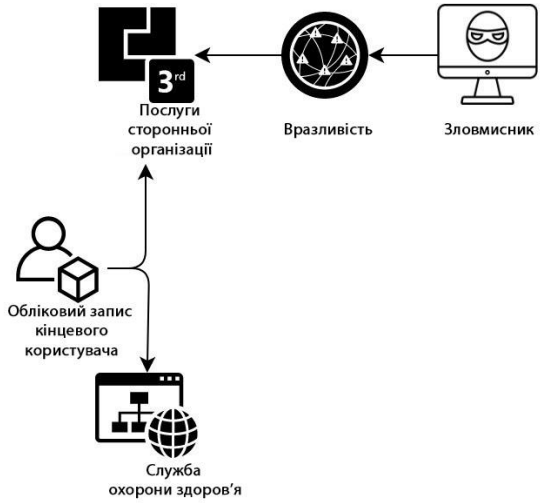


Рис.3 Атака на медичні сервіси

Ці прості заходи зміцнення захисту можуть бути використані для запобігання компрометації користувачів. Крім того, для захисту активів організації необхідно забезпечити безпечне підключення до локальних мереж організації.Щоб уникнути юридичних проблем, організаціям слід переглянути свої договірні угоди з клієнтами, які містять вимоги щодо інформаційної безпеки. Це дозволить організації запобігти порушенням контрактної угоди.

Віддалений моніторинг безпеки повинен бути зосереджений на аналізі подій з кінцевих точок за допомогою хост-систем виявлення вторгнень, рішень для виявлення і реагування на кінцеві точки, а також програмного забезпечення для забезпечення безпеки кінцевих точок, яке дозволяє дистанційно керувати і агрегувати події в центральній консолі. Крім того, аналітики з безпеки повинні звертати увагу на події та журнали, що надходять від служб організації і активів "хмарної" інфраструктури. Розподілена система управління інформацією про безпечність та подіями (SIEM), побудована на локальній інфраструктурі, є загальноприйнятим підходом.

Однак під час пандемічної ситуації було б більш доцільно проводити моніторинг безпеки з використанням SaaS SIEM, або SIEM, розгорнутої в "хмарній" інфраструктурі.

ЛІТЕРАТУРА

[1] "Рекомендації щодо посилення боротьби за кібербезпеку під час COVID -19", https://home.kpmg/ua/uk/home/insights/2020/04/covid-19-cyber-security.html

[2] Dubov, Covid-19: основні тенденції в області кібербезпеки, NSS 2019.

[3] John Wileym. Carbon Black Special Edition, "Полювання на загрози для манекенів". Inc. 111 River St. Hoboken, 2017, pp 9.-10.

[4] O. Milov, A.Voitko, I. Husarova, I. Opirskyy, O. Fraze-Frazenko, et al., Development of methodology for modeling the interaction of antagonistic agents in cybersecurity systems Eastern-European Journal of Enterprise Technologies, 2019.

[5] "Alozurt - аналіз шкідливого ПЗ", https://any.run/malware-trends/azorult

[6] "Реагування на інциденти і усунення їх наслідків при віддаленій роботі", https://www.crowdstrike.com/resources/crowdcasts/conducting-incident-response-and-remediation-remotely

[7] Іван Опірський, Андрій Винар. «Аналіз використання хмарних сервісів для фішингових атак»// Кібербезпека: освіта, наука, техніка, вип. 1, вип. 9, 2020. -С. 59-68.

АНАЛИЗ АТАК, ИСПОЛЬЗУЕМЫХ КИБЕРПРЕСТУПНИКАМИ ВО ВРЕМЯ COVID 19

По данным Всемирной организации здравоохранения, пандемия определяется как "распространение новой болезни во всем мире". С точки зрения кибербезопасности это значит - катастрофа. Во время катастроф количество киберпреступников растет каждый день. По мере того, как все больше высококвалифицированных специалистов по кибербезопасности вступает в голубый команды, ежедневно запускается все больше вредных аппликаций, примерно 230000 новых образцов вредоносных программ в день, по информации исследователей из PandaLabs. Пандемию можно рассматривать как событие, которое может привести к выполнению планов непрерывности бизнеса или реализации мер аварийного восстановления. В течение этого времени следует анализировать растущее количество угроз кибербезопасности и определять применимы меры безопасности. Существует также много проблем с административной информационной безопасностью, которые также следует учитывать. В этой статье раскрыты основные вопросы мониторинга инфраструктуры, а также обеспечения высокого уровня управления уязвимостями и реагирования на инциденты. Приведены меры управления, которые необходимо использовать в SOC центрах, а также представлено углубленный анализ векторов атак и мер безопасности, которые можно применить для их предотвращения. Представленная атака через программное обеспечение для проведения конференций Zoom, описывающая возможный вектор атаки, используемый субъектом угрозы для компрометации организации через незащищенных пользователей. Когда вредоносное программное обеспечение предоставляется под законным псевдонимом пользователя, другие пользователи могут скачать его и установить на свои активы. Определено, что первым средством контроля, которое можно применить во избежание атак социальной инженерии, является сеансы информирования или тренинги, которые содержат примеры поддельных COVID 19 ресурсов, а вторым - может быть включение модулей обнаружения фишинга и сканирования вредоносных программ электронной почты в программном обеспечении безопасности конечной точки и в настройках системы доставки электронной почты. Определено, что удаленный мониторинг безопасности должен быть сосредоточен на анализе событий из конечных точек с помощью хост-систем обнаружения вторжений, решений для выявления и реагирования на конечные точки, а также программного обеспечения для обеспечения безопасности конечных точек, которое позволяет дистанционно управлять и агрегировать события в центральной консоли. Кроме того, аналитики по безопасности должны обращать внимание на события и журналы, поступающих от служб организации и активов "облачной" инфраструктуры.

Ключевые слова: вектор атаки, вредоносное программное обеспечение, реагирование на инциденты, непрерывность бизнеса, Cyber Kill Chain.

ANALYSIS OF ATTACKS USED BY CYBER CRIMINALS DURING COVID 19

According to the World Health Organization, a pandemic is defined as "the spread of a new disease throughout the world." From a cybersecurity perspective, this means disaster. During disasters, the number of cybercriminals grows every day. As more and more highly qualified cybersecurity professionals join the blue team, more and more malicious applications are launched daily, with an estimated 230,000 new malware samples per day, according to researchers at PandaLabs. A pandemic can be viewed as an event that can lead to the fulfillment of business continuity plans or the implementation of disaster recovery measures. During this time, the growing number of cybersecurity threats should be analyzed and the applicable security measures determined. There are also considermany administrative information security issues. This article covers the main issues of infrastructure monitoring, as well as ensuring a high level of vulnerability management and incident response. Presented the management measures that must be used in SOC centers, as well as an in-depth analysis of attack vectors and security measures that can be applied to prevent them. A presented attack via Zoom conferencing software that describes the possible attack vector used by a threat actor to compromise an organization through unprotected users. When malware is provided under the user's legitimate pseudonym, other users can download it and install it on their assets. It has been determined that the first control that can be applied to avoid social engineering attacks is informative sessions or training sessions that contain examples of fake COVID 19 resources and the second could be the inclusion of phishing detection and email malware scanning modules in the endpoint security software and in the settings of the e-mail delivery system. It has been determined that remote security monitoring should focus on analyzing events from endpoints with host intrusion detection systems, endpoint detection and response solutions, and endpoint security software that allows remote control and aggregation of events across center console. In addition, security analysts should pay attention to events and logs from the organization's services and cloud infrastructure assets.

Keywords: attack vector, malware, incident response, business continuity, Cyber Kill Chain.

Сусукайло Віталій Андрійович аспірант кафедри захисту інформації Національного університету «Львівська політехніка».
E-mail: vitalii.a.susukailo@lpnu.ua.
Orcid ID: 0000-0003-4431-9964.
Сусукайло Виталий Андреевич аспирант кафедры защиты информации Национального университета «Львовская политехника».
Vitalii Susukailo, Postgraduate Student of the Department of Information Protection of the National University "Lviv Polytechnic".

Опірський Іван Романович, д.т.н., доц., професор кафедри захисту інформації Національного університету «Львівська політехніка».
E-mail: ivan.r.opirskyi@lpnu.ua.
Orcid ID: 0000-0002-8461-8996.
Опирский Иван Романович, д.т.н., доц., профессор кафедры защиты информации Национального университета «Львовская политехника».
Opirskyy Ivan, Doctor of Technical Sciences, Associate Professor, Professor of the Department of Information Protection of the National University "Lviv Polytechnic".

Піскозуб Андріян Збігнєвич, к.т.н., доц., доцент кафедри захисту інформації Національного університету «Львівська політехніка».
E-mail: azpiskozub@gmail.com.
Orcid ID: 0000-0002-3582-2835.
Пискозуб Андриян Збигневич, к.т.н., доц., доцент кафедры защиты информации Национального университета «Львовская политехника».
Piskozub Andrian Zbignievycz, Ph.D., Associate Professor, Associate Professor of the Department of Information Protection of the National University "Lviv Polytechnic".

Волошин Ростислав Ярославович студент кафедри захисту інформації Національного університету «Львівська політехніка».
E-mail: rostyslav.voloshyn.kb.2017@lpnu.ua.
Orcid ID:
Волошин Ростислав Ярославович студент кафедры защиты информации Национального университета «Львовская политехника».
Voloshyn Rostyslav, Student of the Department of Information Protection of the National University "Lviv Polytechnic".

Друзюк Олег Сергійович студент кафедри захисту інформації Національного університету «Львівська політехніка».E-mail: oleh.druziuk.kb.2017@lpnu.ua.
Orcid ID:
Друзюк Олег Сергеевич студент кафедры защиты информации Национального университета «Львовская политехника».
Druziuk Oleg, Student of the Department of Information Protection of the National University "Lviv Polytechnic".

DOI: 10.18372/2410-7840.22.14976
УДК 004.056

СТАНДАРТИЗАЦІЯ СИСТЕМ, КОМПЛЕКСІВ ТА ЗАСОБІВ КРИПТОГРАФІЧНОГО ЗАХИСТУ ІНФОРМАЦІЇ ДЛЯ ЗАСТОСУВАННЯ У ПОСТ-КВАНТОВОМУ СЕРЕДОВИЩІ

Анна Корченко, Євгенія Іванченко, Наталія Кошкіна, Олександр Кузнецов, Олена Качко, Олександр Потій, Віктор Онопрієнко, Всеволод Бобух

Криптографічний захист інформації (КЗІ) є важливою складовою інформаційної безпеки держави, безпосередньо пов'язаною з подоланням сучасних проблем та викликів в кібернетичному просторі України, нових загроз інформаційній безпеці в критичних інфраструктурах в оборонній та сфері безпеки, промисловості, банківському секторі, економіці тощо. Особливу небезпеку в цьому змісті становлять нові ризики, пов'язані з розробкою та стрімким впровадженням сучасних та перспективних інформаційних технологій, здатних докорінно змінити архітектуру інформаційних систем, існуючі парадигми, сталі принципи побудови та математичні основи сучасних засобів КЗІ. Зокрема, поява та стрімке удосконалення нових обчислювальних засобів, заснованих на принципах та ефектах квантової фізики (т.з. універсальних квантових комп'ютерів) ставить під загрозу саме існування діючих нині та стандартизованих на національному та міжнародному рівнях механізмів (протоколів, алгоритмів та засобів) асиметричної криптографії.

Ключові слова: *квантові обчислювачі, криптографічні алгоритми, інформаційна безпека, інформаційні системи.*

ВСТУП

Найближчим часом через можливість ефективного застосування квантових обчислювачів для вирішення задач криптографічного аналізу, переважна більшість існуючих сьогодні асиметричних механізмів стане вразливою та безпорадною щодо забезпечення навіть найнижчого рівня безпеки, а математичні перетворення, на яких базуються такі криптографічні засоби, відійдуть в історію криптографічної науки. Така перспектива є реальним викликом сьогодення, що змусило національний інститут стандартів і технологій (NIST) США оголосити відкритий конкурс пост-квантових криптографічних перетворень (Post-Quantum Cryptography, PQC), тобто таких, що будуть надійними та безпечними, навіть за умови застосування квантового криптографічного аналізу. Але уже попередній аналіз показує, що навіть у нових міжнародних рішеннях можуть бути закладені вразливості та системні недоліки.

Проєкти, що були подані на конкурс PQC криптографами усього світу, представляють декілька напрямків розвитку пост-квантової криптографії: криптографія на решітках, кодова криптографія, мультиваріативна криптографія, криптографія, що базується на геш-функціях, та симетрична криптографія. При цьому статистика конкурсу демонструє, що найбільше проєктів представлено у контексті перших чотирьох напрямків та згідно з результатами другого туру подібна тенденція зберігається. Таким чином, математичні перетворення, засновані на використанні решіток, завадостійких кодів, багатовимірної криптографії та геш-функцій, потребують поглибленого вивчення та дослідження з метою синтезу криптографічних примітивів, які будуть конкурентоспроможними у контексті перспективних пост-квантових механізмів з урахуванням міжнародних вимог та елементної базі сучасної мікроелектроніки.

За поглядами Агентства Національної Безпеки (NSA) США [1], NIST США [2], Європейського інституту телекомунікаційних стандартів (ETSI) [3] та провідних світових учених [4] повномасштабні універсальні квантові комп'ютери можуть стати доступними для кіберзловмисників у найближчі 10-15 років. Для упередження цих наявних загроз безпеки наприкінці 2016 року NIST США оголосив всесвітній конкурс пост-квантових криптоалгоритмів [5], в якому задіяні найбільш досвідчені та авторитетні наукові установи, зокрема, Інститут квантових обчислень (IQC), Європейський інститут телекомунікаційних стандартів ETSI, міжнародний проект PQCrypto, тощо.

На сьогоднішній день опубліковані (у липні 2020 р.) результати другого раунду конкурсу [6] та оголошено черговий, третій етап. Всі дослідження зосереджено за чотирма напрямками (перетворення на решітках; збиткових кодах; хеш-функціях та мультиваріативні перетворення) та за трьома механізмами (електронний підпис (ЕП); направлене шифрування (НШ) та інкапсуляція ключів (ІК)). Проміжні результати досліджень за другим етапом конкурсу найбільш докладно викладено в [7]. Результати досліджень та порівняльного аналізу кандидатів другого раунду з використанням ПЛІС представлено в [8]. Докладний опис алгоритмів-фіналістів 3-ого раунду наведено в [9]. Окремим напрямком досліджень є симетрична криптографія та побудова квантових криптоалгоритмів, тобто криптографічних перетворень із використанням квантово-механічних властивостей [10].

Всі зазначені напрямки досліджень у безпосередній співпраці із організаторами конкурсу NIST США розробляються та супроводжуються провідними українськими вченими. Зокрема, авторським колективом цієї роботи протягом останніх 5 років було проведено низку пошукових НДР та ДКР з теоретичного обґрунтування та розробки сучасних моделей, методів та механізмів криптографічного перетворення для пост-квантового застосування. Розроблено та впроваджено 4 національні стандарти, зокрема, асиметричного шифрування та інкапсуляції ключів, що відповідає більш жорстким вимогам щодо надійності та безпеки, ніж встановлені NIST США до пост-квантових криптографічних алгоритмів. Закладене теоретичне підґрунтя, виконане моделювання (прототипування) пост-квантових криптографічних систем і технологій, які потребують подальшого продовження досліджень, доповнення та вдосконалення.

МЕТА РОБОТИ.

Метою роботи є розробка основних елементів загальнонаціональної системи КЗІ, а саме стандартизація та безпосереднє впровадження в Україні нових моделей, методів та механізмів криптографічного перетворення інформації в умовах можливого застосування квантових засобів криптографічного аналізу, ведення інформаційних та гібридних війн.

ПОСТАНОВКА ЗАДАЧІ.

Для досягнення поставленої мети необхідно провести дослідження стану КЗІ стосовно забезпечення інформаційної безпеки держави та механізмів криптографічного перетворення для пост-квантового застосування.

На виконання рішення Уряду України протягом останніх років Державною службою спеціального зв'язку та захисту інформації (Держспецзв'язку) було організовано розробку, дослідження та прийняття трьох нових стандартів симетричного криптографічного перетворення (ДСТУ 7624:2014; ДСТУ 7564:2014; ДСТУ 8845:2019).

Національний стандарт ДСТУ 7564:2014 «Інформаційні технології. Криптографічний захист інформації. Функція хешування» установлює алгоритм обчислення хеш-значення для послідовностей двійкових символів, що застосовується в криптографічних методах захисту, забезпечення цілісності та автентичності інформації під час її передачі, обробки і зберігання, в тому числі при використанні електронного цифрового підпису, що визначений ДСТУ 4145-2002. Цей стандарт рекомендується використовувати під час розробки засобів криптографічного захисту інформації в інформаційно-телекомунікаційних системах, а також при модернізації діючих систем для заміни функції хешування, що визначена у міждержавному стандарті ГОСТ 34.311-95.

Національний стандарт ДСТУ 7624:2014 «Інформаційні технології. Криптографічний захист інформації. Алгоритм симетричного блокового перетворення» установлює криптографічний алгоритм симетричного блокового перетворення для забезпечення конфіденційності та цілісності (як додаткової послуги) інформації під час її обробки. Стандарт використовується під час розробки засобів криптографічного захисту інформації в інформаційних, телекомунікаційних та інформаційно-телекомунікаційних системах, а також при модерні-

зації діючих систем для заміни ДСТУ ГОСТ 28147:2009. Для забезпечення конфіденційності і цілісності послідовностей двійкових символів можливо використання цього стандарту сумісно з ДСТУ 7624:2014 «Інформаційні технології. Криптографічний захист інформації. Функція хешування», при цьому повинні використовуватись різні ключі шифрування і автентифікації.

Національний стандарт ДСТУ 8845:2019 «Інформаційні технології. Криптографічний захист інформації. Алгоритм симетричного потокового перетворення» установлює криптографічний алгоритм симетричного потокового перетворення для забезпечення конфіденційності та цілісності (як додаткової послуги) інформації під час її оброблення. У стандарті описано алгоритм симетричного потокового криптографічного перетворення, який використовує ключовий потік для шифрування відкритого тексту побітовим або поблоковим чином. Ключовий потік генерується лише із секретного ключа та вектора ініціалізації (синхропосилки), отже стандарт визначає синхронний потоковий шифр (за класифікацією з ДСТУ ISO/IEC 18033-4:2015). Стандарт використовують під час розроблення засобів криптографічного захисту інформації в інформаційних, телекомунікаційних та інформаційно-телекомунікаційних системах, а також в разі модернізації наявних систем для заміни потокових режимів згідно з ДСТУ ГОСТ 28147:2009. Для забезпечення конфіденційності та цілісності послідовностей двійкових символів можна застосовувати цей стандарт сумісно з ДСТУ ISO/IEC 18033-4:2015 (ISO/IEC 18033-4:2011, IDT) та ДСТУ 7564:2014.

Розглянемо основні результати з розробки та дослідження зазначених криптографічних алгоритмів, зосереджуючи увагу на перевагах їх практичного застосування в пост-квантовому середовищі, в умовах ведення інформаційних та гібридних війн.

Національний стандарт ДСТУ 7564:2014 «Інформаційні технології. Криптографічний захист інформації. Функція хешування». Наприкінці 2014 року до Переліку прийнятих і затверджених національних стандартів України внесено ДСТУ 7564:2014 «Інформаційні технології. Криптографічний захист інформації. Функція хешування», який був розроблений на замовлення Держспецзв'язку на виконання наказу Мінекономрозвитку від 02 грудня 2014 року № 1431 «Про прийняття національних стандартів України, гармонізованих з європейськими стандартами, міжнародних стандартів як національних стандартів України…» та вводиться в дію 01 квітня 2015 року [11, 12].

Загальні положення. Під функцією гешування H розуміється залежне від вектора ініціалізації $IV \in V_l$, $l \in \{512,1024\}$ відображення повідомлення $M \in V_N$, $N \in \{0,1,..,2^{96}-1\}$ у геш-значення $H(IV,M) \in V_n$, $n \in \{8 \cdot s \mid s = 1,2,...,64\}$, таке що $H^{(IV)} : V_N \to V_n$. Режим роботи функції гешування для $n \in \{8 \cdot s \mid s = 1,2,...,64\}$ позначається «Купина-n». Основними режимами роботи функції гешування, що рекомендуються до застосування, є «Купина-256», «Купина-384» і «Купина-512».

Загальна структура перетворення. При формуванні геш-значення повідомлення M завжди доповнюється до довжини, кратної розміру блоку, та поділяється на блоки $m_1, m_2, \ldots, m_k$, кожен з яких має довжину l біт. Вибір l здійснюється відповідно до розміру геш-значення n:

$$l = \begin{cases} 512 \text{ для } 8 \le n \le 256, \\ 1024 \text{ для } 256 < n \le 512, \end{cases}$$

де $n \in \{8 \cdot s \mid s = 1,2,...,64\}$.

Обчислення геш-значення здійснюється за наступною ітеративною процедурою:

$$h_0 = IV,$$
$$h_i = T_l^{\oplus}(h_{i-1} \oplus m_i) \oplus T_l^{+}(m_i) \oplus h_{i-1}, \quad i = 1,2,...,k,$$
$$H(IV,M) = R_{l,n}\left(T_l^{\oplus}(h_k) \oplus h_k\right),$$

де $IV = \begin{cases} 0x4000...00 \text{ для } l = 512, \\ 0x8000...00 \text{ для } l = 1024 \end{cases}$ – вектор ініціалізації довжиною l біт,

$T_l^{\oplus}$, T_l^{+} – бієктивні перетворення, що виконують відображення вхідного блоку довжиною l біт у вихідний такої ж довжини,

– функція, що повертає n старших біт з вхідного блоку x довжиною l біт ($n<l$). При обробці l-бітових слів представляється у вигляді $R_{l,n}(x)=\left(x>>(l-n)\right)\&\sim\left(0xFF...F<<n\right)$, де результат записується в молодші n біт обчисленого значення. Структурна схема функції гешування "Купина" наведена на рисунку 1.

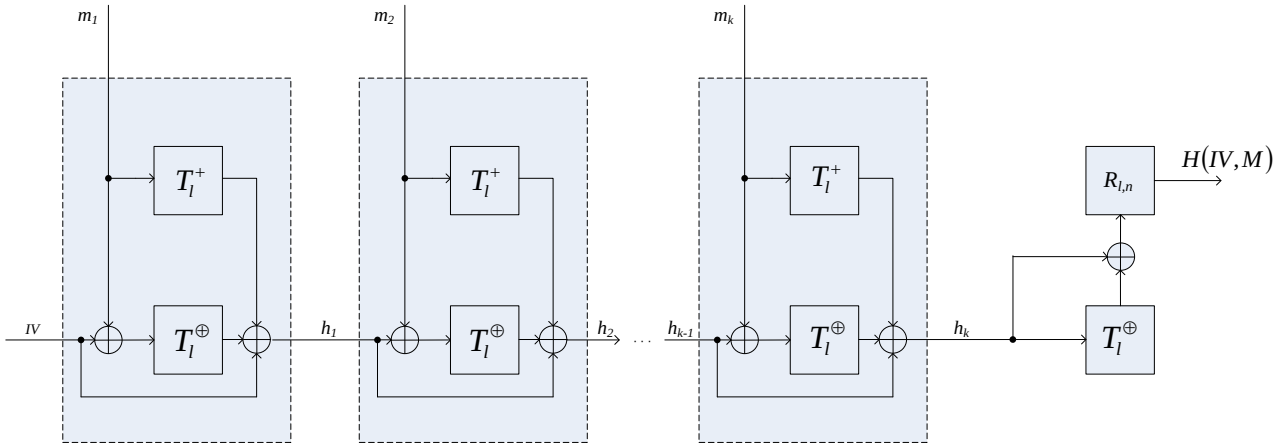


Рис. 1. Структурна схема функції гешування "Купина"

Доповнення повідомлення. На вхід функції гешування подається повідомлення (бітова послідовність) M довжини N, $N\in\left\{0,1,...,2^{96}-1\right\}$, яка задана в бітах. Кожне повідомлення доповнюється, незалежно від його довжини. У кінець повідомлення додається допоміжна інформація, яка містить одиничний біт, необхідну кількість нульових бітів (див. нижче) та довжину повідомлення на вході функції гешування, таким чином, щоб доповнена бітова послідовність мала довжину, кратну розміру внутрішнього стану l, $l\in\{512,1024\}$.

При доповненні спочатку у кінець повідомлення додається одиничний біт «1», потім додаються d нульових бітів, де $d=(-N-97)\bmod l$. Після цього додаються ще 96 біт, в яких записано значення N (найменш значущі байти мають менший номер, тобто використовується формат little endian). Максимальна довжина повідомлення, що може бути оброблено, становить $2^{96}-1$ біт.

Перетворення $T_l^{\oplus}$ та T_l^{+}. Перетворення $T_l^{\oplus}$ та T_l^{+} є бієктивними відображеннями $T_l^{\oplus},T_l^{+}:V_l\to V_l$, $l\in\{512,1024\}$, кожне з яких реалізоване у вигляді ітеративного застосування низки функцій, що обробляють вхідний аргумент $x\in V_l$ як матрицю розміром $8\times c$ байтів, що містить елементи поля GF(2^8). Залежність розміру внутрішнього стану (l), кількості ітерацій (t) та розмірності матриці (c) від розміру геш-значення n наведено у табл. 1.

Таблиця 1

Процес перетворення

Розмір геш-значення	Розмір внутрішнього стану (l)	Кількість ітерацій перетворення (t)	Кількість стовпців в матриці (c)
$8\le n\le 256$	512	10	8
$256<n\le 512$	1024	14	16

Матриця внутрішнього стану позначається як $G=(g_{i,j})$, $g_{i,j}\in GF(2^8)$. Запис байтів $B_1,B_2,...,B_{l/8}$ перетворень $T_l^{\oplus}$ та T_l^{+} до матриці і зчитування з неї здійснюється по стовпцях (приклад для $l=512$ та $c=8$ див. на рис. 2).

$T_l^{\oplus}$ та T_l^{+} визначені наступним чином:

$$T_l^{\oplus}=\prod_{i=0}^{t-1}\left(\psi\circ\tau^{(l)}\circ\pi'\circ\kappa_i^{(l)}\right),$$

$$T_l^{+}=\prod_{i=0}^{t-1}\left(\psi\circ\tau^{(l)}\circ\pi'\circ\eta_i^{(l)}\right),$$

де $\kappa_i^{(l)}$ – функція додавання констант ітерацій за модулем 2,

$\eta_i^{(l)}$ – функція додавання констант ітерацій за модулем 2^{64},

π' – шар нелінійного бієктивного відображення, який виконує обробку векторів, заданих на V_8 (байтову підстановку);

$\tau^{(l)}$ – перестановка елементів $g_{i,j}\in GF(2^8)$ внутрішнього стану (циклічний зсув при матричному поданні);

ψ – лінійне перетворення (множення вектору на матрицю над скінченним полем).

В функціях $\kappa_i^{(l)}$, $\eta_i^{(l)}$, π', $\tau^{(l)}$ і ψ вхідний аргумент $x\in V_l$ та вихідне значення $\chi(x)\in V_l$, $\chi\in\left\{\kappa_i^{(l)},\eta_i^{(l)},\pi',\tau^{(l)},\psi\right\}$ розглядається як матриця розміром $8\times c$ байтів (див. табл.1).

Вхідна послідовність							
B_1	B_9	B_{17}	B_{25}	B_{33}	B_{41}	B_{49}	B_{57}
B_2	B_{10}	B_{18}	B_{26}	B_{34}	B_{42}	B_{50}	B_{58}
B_3	B_{11}	B_{19}	B_{27}	B_{35}	B_{43}	B_{51}	B_{59}
B_4	B_{12}	B_{20}	B_{28}	B_{36}	B_{44}	B_{52}	B_{60}
B_5	B_{13}	B_{21}	B_{29}	B_{37}	B_{45}	B_{53}	B_{61}
B_6	B_{14}	B_{22}	B_{30}	B_{38}	B_{46}	B_{54}	B_{62}
B_7	B_{15}	B_{23}	B_{31}	B_{39}	B_{47}	B_{55}	B_{63}
B_8	B_{16}	B_{24}	B_{32}	B_{40}	B_{48}	B_{56}	B_{64}

↓

Внутрішній стан							
$g_{0,0}$	$g_{0,1}$	$g_{0,2}$	$g_{0,3}$	$g_{0,4}$	$g_{0,5}$	$g_{0,6}$	$g_{0,7}$
$g_{1,0}$	$g_{1,1}$	$g_{1,2}$	$g_{1,3}$	$g_{1,4}$	$g_{1,5}$	$g_{1,6}$	$g_{1,7}$
$g_{2,0}$	$g_{2,1}$	$g_{2,2}$	$g_{2,3}$	$g_{2,4}$	$g_{2,5}$	$g_{2,6}$	$g_{2,7}$
$g_{3,0}$	$g_{3,1}$	$g_{3,2}$	$g_{3,3}$	$g_{3,4}$	$g_{3,5}$	$g_{3,6}$	$g_{3,7}$
$g_{4,0}$	$g_{4,1}$	$g_{4,2}$	$g_{4,3}$	$g_{4,4}$	$g_{4,5}$	$g_{4,6}$	$g_{4,7}$
$g_{5,0}$	$g_{5,1}$	$g_{5,2}$	$g_{5,3}$	$g_{5,4}$	$g_{5,5}$	$g_{5,6}$	$g_{5,7}$
$g_{6,0}$	$g_{6,1}$	$g_{6,2}$	$g_{6,3}$	$g_{6,4}$	$g_{6,5}$	$g_{6,6}$	$g_{6,7}$
$g_{7,0}$	$g_{7,1}$	$g_{7,2}$	$g_{7,3}$	$g_{7,4}$	$g_{7,5}$	$g_{7,6}$	$g_{7,7}$

Рис. 2. Заповнення внутрішнього стану

Функція $\kappa_i^{(l)}$ здійснює додавання за модулем 2 до кожного стовпця матриці внутрішнього стану $G=(g_{i,j})$ вектора $\omega_{i,j}\in V_{64}$. Для j-го стобця на i-й ітерації перетворення $T_l^{\oplus}$ вектор $\omega_{i,j}=\left((j<<4)\oplus i,0,0,0,0,0,0,0\right)^T$. Функція $\eta_i^{(l)}$ здійснює додавання за модулем 2^{64} до кожного стовпця матриці внутрішнього стану $G=(g_{i,j})$ вектора $\varsigma_{i,j}\in V_{64}$. Для j-го стовпця на i-й ітерації перетворення T_l^{+} вектор $\varsigma_{i,j}$ визначається як

$$\varsigma_{i,j}=\left(0xF3,0xF0,0xF0,0xF0,0xF0,0xF0,0xF0,\left((c-1-j)<<4\right)\oplus i\right)^T,$$

де $0xF3$ – молодші 8 біт (при виконанні операції

додавання) вектору $\varsigma_{i,j}$.

Функція π' виконує заміну кожного елементу $g_{i,j}$ матриці внутрішнього стану $G=(g_{i,j})$ на $\pi_{i \bmod 4}\left(g_{i,j}\right)$, де $\pi_s : V_8 \to V_8, s \in \{0,1,2,3\}$ - підстановки, наведені у додатку А. Наприклад, нехай $g_{0,0}=0x22$, тоді $\pi_0\left(0x22\right)=0xA3$.

Функція $\tau^{(l)}$ виконує циклічний зсув вправо рядків матриці стану $G=(g_{i,j})$. Рядки з номерами i=0,1,2, ..,6 матриці зсуваються на i елементів, а рядок з номером 7 зсувається на 7 елементів для $l=512$ і на 11 елементів для $l=1024$.

При обчисленні результату функції ψ кожен елемент $g_{i,j}$ матриці внутрішнього стану $G=(g_{i,j})$ розглядається як елемент скінченного поля $GF(2^8)$, що утворене незвідним поліномом $g(x) = x^8 + x^4 + x^3 + x^2 + 1$, або $0x11d$ у шістнадцятковому поданні. Кожен елемент результуючої матриці стану $U=\left(u_{i,j}\right)$ отримується як результат множення стовпця матриці стану $G=(g_{i,j})$ на вектор над скінченним полем $GF(2^8)$, що утворює циркулянтну матрицю МДР-коду: $u_{i,j}=\left(v>>>i\right)\times G_j$, де $v=\left(0x01,0x01,0x05,0x01,0x08,0x06,0x07,0x04\right)$ – вектор, що складається з послідовності байтових констант у шістнадцятковому поданні, які інтерпретуються як елементи поля $GF(2^8)$; G_j – j–й стовпець матриці стану $G=(g_{i,j})$, $G_j=\left(g_{0,j},g_{1,j},g_{2,j},g_{3,j},g_{4,j},g_{5,j},g_{6,j},g_{7,j}\right)^T$.

Алгоритм ДСТУ 7564:2014 є результатом багаторічної плідної співпраці Державної служби спеціального зв'язку та захисту інформації України та провідних українських науковців і враховує досвід та результати проведення міжнародних і відкритого національного конкурсів криптографічних алгоритмів. Криптографічний алгоритм, що визначаються ДСТУ 7564:2014, є гнучким, підтримує розмір блока від 128 до 512 бітів, що є унікальним у світі. Підвищена довжина блоку внутрішнього стану унеможливлює ефективне застосування квантових засобів криптографічного аналізу, що робить його придатним для практичного застосування в постквантовому середовищі, в умовах ведення інформаційних та гібридних війн. Практичне впровадження ДСТУ 7564:2014 дозволить суттєво вдосконалити показники ефективності захисту систем, засобів і протоколів криптографічного захисту інформації, що розробляються в Україні, і у деяких випадках зробити їх суттєво кращими ніж наявні та перспективні світові рішення.

Національний стандарт ДСТУ 7624:2014 «Інформаційні технології. Криптографічний захист інформації. Алгоритм симетричного блокового перетворення

Цей національний стандарт був розроблений на замовлення Держспецзв'язку та внесений до Переліку прийнятих та затверджених національних стандартів України на виконання наказу Мінекономрозвитку від 29 грудня 2014 року № 1484 «Про прийняття європейських стандартів як національних стандартів України та скасування національних стандартів України» [11, 12].

Стандарт ДСТУ 7624:2014 розроблено задля поступової заміни міждержавного стандарту ДСТУ ГОСТ 28147:2009 (на базі ГОСТ 28147-89, який визначає симетричний блочний алгоритм криптографічного перетворення), а ДСТУ 7564:2014 – для поступової заміни міждержавного стандарту ДСТУ ГОСТ 34.311:2009 (визначає функцію хешування та має посилання на ГОСТ 28147-89), які не відповідають сучасним вимогам до швидкодії і потенційним викликам щодо криптографічної стійкості.

Стандарт ДСТУ 7624:2014 "Інформаційні технології. Криптографічний захист інформації. Алгоритм симетричного блокового перетворення" визначає сучасний алгоритм симетричного блокового перетворення для забезпечення конфіденційності та цілісності інформації під час її обробки і встановлює режими його роботи (застосування). Криптографічні перетворення, що застосовуються в алгоритмі, відповідають сучасним вимогам щодо рівня криптографічної стійкості та швидкодії.

Алгоритм розроблено з урахуванням існуючих і потенційних загроз, подальшого інтенсивного розвитку інформаційних технологій та необхідності активного використання протягом декількох наступних десятиліть.

Призначення. Під алгоритмом симетричного блокового перетворення $\mathfrak{I}_{l,k}^{(K)}$ у режимі шифрування розуміється пряме і обернене відображення відкритого тексту $M \in V_N$ у шифртекст $C \in V_N$ (і навпаки), що залежить від ключа шифрування $K \in V_k$ і (для деяких режимів) синхропосилки $S \in V_l$: $\mathfrak{I}_{l,k}^{(K)} : V_N \times V_k \to V_N$ або $\mathfrak{I}_{l,k}^{(K)} : V_l \times V_N \times V_k \to V_N$. Параметри l і k визначають розмір блоку та довжину ключа базового блокового алгоритму.

Під алгоритмом симетричного блокового перетворення $\phi_{l,k}^{(K)}$ у режимі забезпечення цілісності (вироблення імітовставки) розуміється відображення повідомлення $M \in V_N$ в імітовставку (код автентифікації повідомлення), що залежить від ключа автентифікації $K \in V_k$ і (для деяких режимів) синхропосилки $S \in V_l$: $\phi_{l,k}^{(K)}(S, M) \in V_q$, $64 \le q \le l$ таке що $\phi_{l,k}^{(K)} : V_l \times V_N \times V_k \to V_q$.

Криптографічний алгоритм, визначений у цьому стандарті, передбачає можливість одночасного забезпечення конфіденційності та цілісності повідомлення шляхом послідовного застосування відповідних перетворень.

Режими роботи. Криптографічний алгоритм симетричного блокового перетворення використовує базове перетворення як основний елемент при забезпеченні конфіденційності та (або) цілісності.

Режими роботи криптографічного алгоритму, визначеного в цьому стандарті, їх позначення та послуги безпеки, які забезпечує відповідний режим, визначені у табл. 2.

Додаткові параметри використання кожного режиму наведені при його описі.

Таблиця 2

Режими роботи криптографічного алгоритму

№ режиму	Назва режиму	Позначення	Послуга безпеки
1	Проста заміна (базове перетворення)	ECB	Конфіденційність
2	Гамування	CTR	Конфіденційність
3	Гамування зі зворотнім зв'язком за шифртекстом	CFB	Конфіденційність
4	Вироблення імітовставки	CMAC	Цілісність
5	Зчеплення шифрблоків	CBC	Конфіденційність
6	Гамування зі зворотнім зв'язком за шифргамою	OFB	Конфіденційність
7	Вибіркове гамування із прискореним виробленням імітовставки	GCM, GMAC	конфіденційність і цілісність (GCM), тільки цілісність (GMAC)
8	Вироблення імітовставки і гамування	CCM	цілісність і конфіденційність
9	Індексованої заміни	XTS	конфіденційність
10	Захисту ключових даних	KW	конфіденційність і цілісність

Режим роботи криптографічного алгоритму, визначеного у цьому стандарті, позначається на-

ступним чином: „Калина-l/k-позначення режиму-параметри режиму” (для деяких режимів параметри відсутні), де l – розмір блоку базового перетворення, k – довжина ключа.

Наприклад, Калина-256/512-CCM-32,128 визначає використання базового перетворення з розміром блоку 256 бітів, довжиною ключа 512 бітів, застосування у режимі вироблення імітовставки і гамування, довжина конфіденційної (та відкритої) частини повідомлення завжди менша 2^{32} байтів, довжина імітовставки дорівнює 128 бітам. Режим простої заміни збігається з базовим перетворенням, тому крім позначення «Калина-l/k-ECB» може використовуватись позначення «Калина-l/k».

Проста заміна (базове перетворення). Режим простої заміни є компонентом усіх інших режимів роботи криптографічного алгоритму симетричного блокового перетворення. Без додаткових перетворень, визначених іншими режимами, використання простої заміни для захисту повідомлень не рекомендується. Базове перетворення реалізує пряме перетворення (зашифрування) та обернене перетворення (розшифрування). Базове перетворення зашифрування $T_{l,k}^{(K)}$ є параметризованим ключем шифрування K відображенням $T_{l,k}^{(K)}: V_l \to V_l$, $K \in V_k$, $l,k \in \{128,256,512\}$ при цьому $k = l$ або $k = 2 \cdot l$, що реалізоване у вигляді ітеративного застосування низки функцій, які обробляють вхідний аргумент $x \in V_l$ як матрицю внутрішнього стану розміром $8 \times c$ байтів, що містить елементи поля GF(2^8). Базове перетворення розшифрування $U_{l,k}^{(K)}$ є параметризованим ключем шифрування K відображенням, оберненим до $T_{l,k}^{(K)}$, також реалізованим у вигляді ітеративного перетворення. Залежність кількості ітерацій (t) при реалізації перетворень $T_{l,k}^{(K)}$ та $U_{l,k}^{(K)}$, кількості стовпців матриці внутрішнього стану (c) від розміру блоку (l) і довжини ключа шифрування (k) наведено у табл. 3.

Таблиця 3

Базове перетворення

№ з/п	Розмір блоку (l)	Довжина ключа (k)	Кількість ітерацій перетворення (t)	Кількість стовпців в матриці (c)
1	128	128	10	2
2		256	14	
3	256	256	14	4
4		512	18	
5	512	512	18	8

Базове перетворення виконує обробку вхідного блоку даних довжиною l бітів (відкритий текст при зашифруванні або шифртекст при розшифруванні). Матриця внутрішнього стану позначається як $G = (g_{i,j})$, $g_{i,j} \in GF(2^8)$, де $i = \overline{0,7}$, $j = \overline{0,c-1}$.. Запис байтів $B_1, B_2, ..., B_{l/8}$ для перетворень $T_{l,k}^{(K)}$ та $U_{l,k}^{(K)}$ до матриці і зчитування з неї здійснюється по стовпцях.

Приклад запису байтів до внутрішнього стану для $l = 512$ ($k = 512$, $c = 8$) див. на рис. 1.

Базове перетворення зашифрування $T_{l,k}^{(K)}$ визначено наступним чином:

$$T_{l,k}^{(K)} = \eta_l^{(K_t)} \circ \psi_l \circ \tau_l \circ \pi_l' \circ \left(\prod_{v=1}^{t-1} \left(\kappa_l^{(K_v)} \circ \psi_l \circ \tau_l \circ \pi_l' \right) \right) \circ \eta_l^{(K_0)},$$

де l – розмір внутрішнього стану блокового шифру (у бітах),

K – ключ шифрування,

k – довжина ключа шифрування (у бітах),

$\eta_l^{(K_v)}$ – функція додавання циклового ключа K_v ($v \in \{0,t\}$) за модулем 2^{64},

π'_l – шар нелінійного бієктивного відображення, який виконує обробку векторів, заданих над V_8 (байтова підстановка);

τ_l – перестановка елементів $g_{i,j} \in GF(2^8)$ внутрішнього стану (циклічний зсув рядків вправо при матричному поданні);

ψ_l – лінійне перетворення (множення матриці лінійного перетворення на матрицю внутрішнього стану над скінченним полем);

$\kappa_l^{(K_v)}$ – функція додавання циклового ключа K_v ($v \in \{1,2,...,t-1\}$) за модулем 2 (інволютивне перетворення).

В функціях π'_l, τ_l і ψ_l вхідний аргумент $x \in V_l$ та вихідне значення $\chi(x) \in V_l$, $\chi \in \{\pi'_l, \tau_l, \psi_l\}$ розглядаються як матриці розміром $8 \times c$ байтів (див. табл. 3). Функції $\eta_l^{(K_v)}$ і $\kappa_l^{(K_v)}$ залежать від параметра $K_v \in V_l$ (циклового ключа v-ї ітерації), мають вхідний аргумент $x \in V_l$ (внутрішній стан шифру), та вихідне значення $\chi(x, K_v) \in V_l$, $\chi \in \{\eta_l^{(K_v)}, \kappa_l^{(K_v)}\}$, при цьому вхідні аргументи та вихідне значення розглядаються як матриці розміром $8 \times c$ байтів.

Функція додавання циклового ключа K_v за модулем 2^{64} $\eta_l^{(K_v)}$ здійснює додавання за модулем 2^{64} стовпців матриці внутрішнього стану $G = (g_{i,j})$ і стовпців матриці циклового ключа $K_v = (k_{i,j}^v)$, при цьому результат також є матрицею розміром $8 \times c$ байтів (внутрішнім станом після додавання). При виконанні додавання менші значущі байти мають менші індекси, тобто використовується формат little endian.

Функція π'_l виконує заміну кожного елементу $g_{i,j}$ матриці внутрішнього стану $G = (g_{i,j})$ на $\pi_{i \bmod 4}(g_{i,j})$, де $\pi_s : V_8 \to V_8, s \in \{0,1,2,3\}$ - підстановки, які наведені у додатку А.

Наприклад, нехай $g_{0,0} = 0x23$, тоді $\pi_0(0x23) = 0x4F$. Для здійснення перетворення може використовуватися інший набір підстановок, відмінний від наведеного у додатку А. У цьому випадку набір підстановок має постачатися і застосовуватися в установленому порядку.

Функція τ_l виконує циклічний зсув вправо рядків матриці стану $G = (g_{i,j})$. Кількість елементів зсуву залежить від номеру рядку $i \in \{0,1,..,7\}$, розміру блоку $l \in \{128,256,512\}$, та обчислюється за формулою $\delta_i = \left\lfloor \frac{i \cdot l}{512} \right\rfloor$. Наприклад, 5-й рядок матриці стану шифра з 256-бітовим блоком зсувається вправо на 2 елемента.

При обчисленні результату функції ψ_l кожен елемент $g_{i,j}$ матриці внутрішнього стану $G = (g_{i,j})$ розглядається як елемент скінченного поля $GF(2^8)$, що утворене незвідним поліномом $g(x) = x^8 + x^4 + x^3 + x^2 + 1$, або *0x11d* у шістнадцятковому поданні. Кожен елемент результуючої матриці стану $W = (w_{i,j})$ отримується як результат множення векторів довжини 8 над скінченним полем $GF(2^8)$ за формулою

$$w_{i,j} = (\upsilon >>> i) \otimes G_j,$$

де $\upsilon = (0x01, 0x01, 0x05, 0x01, 0x08, 0x06, 0x07, 0x04)$ – вектор, що утворює циркулянтну матрицю МДР-коду і складається з послідовності байтових констант у шістнадцятковому поданні, які інтерпретуються як елементи поля $GF(2^8)$, при цьому циклічний зсув виконується відносно елементів вектора над скінченним полем; G_j – j–й стовпець матриці стану $G = (g_{i,j})$.

Функція $\kappa_l^{(K_v)}$ має вхідний аргумент $x \in V_l$ (внутрішній стан шифру) і залежить від параметра $K_v \in V_l$ (циклового ключа v-ї ітерації), кожен з яких поданий як матриця розміром $8 \times c$ байтів.

Функція $\kappa_l^{(K_v)}$ здійснює побітове додавання (за модулем 2) стовпців матриці внутрішнього стану $G=(g_{i,j})$ і стовпців матриці циклового ключа $K_v=(k_{i,j}^v)$, при цьому результат також є матрицею розміром $8\times c$ байтів (внутрішнім станом після додавання).

Базове перетворення розшифрування $U_{l,k}^{(K)}$ визначено наступним чином:

$$U_{l,k}^{(K)}={}_{-1}\eta_l^{(K_0)}\circ\left(\prod_{v=t-1}^{1}\left({}_{-1}\pi_l'\circ{}_{-1}\tau_l\circ{}_{-1}\psi_l\circ\kappa_l^{(K_v)}\right)\right)\circ{}_{-1}\pi_l'\circ{}_{-1}\tau_l\circ{}_{-1}\psi_l\circ{}_{-1}\eta_l^{(K_t)},$$

де l – розмір внутрішнього стану блокового шифру (у бітах), K – ключ шифрування, k – довжина ключа шифрування (у бітах), ${}_{-1}\eta_l^{(K_v)}$ – функція віднімання циклового ключа K_v ($v\in\{0,t\}$) за модулем 2^{64} (обернена до $\eta_l^{(K_v)}$); ${}_{-1}\psi_l$ – обернене лінійне перетворення (множення матриці оберненого лінійного перетворення на матрицю внутрішнього стану над скінченним полем); ${}_{-1}\tau_l$ – обернена перестановка елементів $g_{i,j}\in GF(2^8)$ внутрішнього стану (циклічний зсув рядків вліво при матричному поданні); ${}_{-1}\pi_l'$ – шар оберненого нелінійного бієктивного відображення, який виконує обробку векторів, заданих над V_8 (обернена байтова підстановка); $\kappa_l^{(K_v)}$ – інволютивна функція додавання циклового ключа K_v ($v\in\{1,2,...,t-1\}$) за модулем 2 (однакова для зашифрування і розшифрування).

Як і при зашифруванні, в функціях ${}_{-1}\pi'$, ${}_{-1}\tau_l$ і ${}_{-1}\psi_l$ вхідний аргумент $x\in V_l$ та вихідне значення $\chi(x)\in V_l$, $\chi\in\{{}_{-1}\pi',{}_{-1}\tau_l,{}_{-1}\psi_l\}$ розглядаються як матриці розміром $8\times c$ байтів.

Функція ${}_{-1}\eta_l^{(K_v)}$ має два вхідних аргументи $x\in V_l$ (внутрішній стан шифру) і $K_v\in V_l$ (циклові ключ v-ї ітерації) та вихідне значення ${}_{-1}\eta_l^{(K_v)}(x,K_v)\in V_l$, при цьому вхідні аргументи та вихідне значення розглядаються як матриці розміром $8\times c$ байтів.

Функція віднімання циклового ключа K_v за модулем 2^{64} ${}_{-1}\eta_l^{(K_v)}$ є оберненою до $\eta_l^{(K_v)}$ і здійснює віднімання за модулем 2^{64} стовпців матриці циклового ключа $K_v=(k_{i,j}^v)$ від стовпців матриці внутрішнього стану $G=(g_{i,j})$, при цьому результат також є матрицею розміром $8\times c$ байтів (внутрішнім станом після віднімання). При виконанні віднімання найменш значущі байти мають менший індекс, тобто використовується формат little endian.

Функція ${}_{-1}\pi_l'$ виконує заміну кожного елементу $g_{i,j}$ матриці внутрішнього стану $G=(g_{i,j})$ на ${}_{-1}\pi_{i \bmod 4}(g_{i,j})$, де ${}_{-1}\pi_s:V_8\to V_8, s\in\{0,1,2,3\}$ - підстановки, які наведені у додатку А. Наприклад, нехай $g_{0,0}=0xA3$, тоді ${}_{-1}\pi_0(0xA3)=0x22$. У разі використання підстановок, відмінних від наведених у додатку А, застосовуються відповідні їм обернені.

Функція ${}_{-1}\tau_l$ виконує циклічний зсув вліво рядків матриці стану $G=(g_{i,j})$. Кількість елементів зсуву залежить від номеру рядку $i\in\{0,1,..,7\}$ розміру блоку $l\in\{128,256,512\}$, та обчислюється за формулою $\delta_i=\left\lfloor\frac{i\cdot l}{512}\right\rfloor$. Наприклад, 4-й рядок матриці стану шифру з 128-бітовим блоком зсувається вліво на 1 елемент.

При обчисленні результату функції ${}_{-1}\psi_l$ кожен елемент $g_{i,j}$ матриці внутрішнього стану $G=(g_{i,j})$ розглядається як елемент скінченого поля $GF(2^8)$, що утворене незвідним поліномом $\vartheta(x)=x^8+x^4+x^3+x^2+1$, або *0x11d* у шістнадцятковому поданні.

Кожен елемент результуючої матриці стану ${}_{-1}W=({}_{-1}w_{i,j})$ отримується як результат множення векторів довжини 8 над скінченним полем $GF(2^8)$ за формулою ${}_{-1}w_{i,j}=({}_{-1}\upsilon>>>i)\otimes G_j$, де

$_{-1}v =$
$(0xAD, 0x95, 0x76, 0xA8, 0x2F, 0x49, 0xD7, 0xCA)$
– вектор, що утворює циркулянтну матрицю МДР-коду і складається з послідовності байтових констант у шістнадцятковому поданні, які інтерпретуються як елементи поля $GF(2^8)$, при цьому циклічний зсув виконується відносно елементів вектора над скінченним полем; G_j – j–й стовпець матриці $G = (g_{i,j})$.

Гамування. Режим забезпечує конфіденційність повідомлення шляхом шифрування. Шифрування виконує пряме відображення повідомлення M ($|M| \geq 1$) у шифртекст C, $|C| = |M|$, та обернене відображення шифртексту C в повідомлення M. Вимоги на кратність довжини повідомлення розміру блоку базового перетворення не накладаються. Параметрами режиму є ключ шифрування K, $|K| = k$ та синхропосилка S, $|S| = l$. Додаткові вимоги щодо синхропосилки не накладаються. Режим гамування позначається як Калина-l/k-CTR.

Повідомлення M ($|M| \geq 1$) подається у вигляді послідовності блоків: $M = m_1 \| m_2 \| ... \| m_n$, $|m_i| = l$ для $i = 1,2,..,n-1$, $1 \leq |m_n| \leq l$. Початкове значення лічильника s_0 ($|s_0| = l$) обчислюється як $s_0 = T_{l,k}^{(K)}(S)$. Кожен з блоків шифртекста обчислюється відповідно до співвідношення $c_i = m_i \oplus L_{l,|m_i|}\left(T_{l,k}^{(K)}\left(L_{l,l/2}(s_0 + i) \| R_{l,l/2}(s_0)\right)\right)$ для $i = 1,2,..,n$, $|c_i| = |m_i|$. Результатом зашифрування повідомлення є шифртекст $C = c_1 \| c_2 \| ... \| c_n$.

Шифртекст C ($|C| \geq 1$) подається у вигляді послідовності блоків: $C = c_1 \| c_2 \| ... \| c_n$, $|c_i| = l$ для $i = 1,2,..,n-1$, $1 \leq |c_n| \leq l$. Початкове значення лічильника s_0 обчислюється як $s_0 = T_{l,k}^{(K)}(S)$. Кожен з блоків повідомлення обчислюється відповідно до співвідношення $m_i = c_i \oplus L_{l,|c_i|}\left(T_{l,k}^{(K)}\left(L_{l,l/2}(s_0 + i) \| R_{l,l/2}(s_0)\right)\right)$ для $i = 1,2,..,n$. Результатом розшифрування є повідомлення $M = m_1 \| m_2 \| ... \| m_n$.

Гамування зі зворотнім зв'язком за шифртекстом. Режим забезпечує конфіденційність повідомлення шляхом шифрування. Шифрування виконує пряме відображення повідомлення M ($|M| \geq 1$) у шифртекст C, $|C| = |M|$, та обернене відображення шифртексту C в повідомлення M. Вимоги на кратність довжини повідомлення розміру блоку базового перетворення не накладаються. Параметрами режиму є ключ шифрування K, $|K| = k$, синхропосилка S, $|S| = l$ та додаткове значення q, яке визначає кількість бітів повідомлення, що обробляються за допомогою одного застосування базового перетворення, $q \in \{1,8,64,128,256,512 \mid q \leq l\}$. Рекомендованим значенням параметра є $q = l$. Додатковою вимогою до синхропосилки в цьому режимі є випадковість, в тому числі непередбачуваність значення, яке буде застосовано для будь-якого повідомлення, до його формування. Режим гамування позначається як Калина-l/k-CFB-q.

Повідомлення M ($|M| \geq 1$) подається у вигляді послідовності блоків: $M = m_1 \| m_2 \| ... \| m_n$, $|m_i| = q$ для $i = 1,2,..,n-1$, $1 \leq |m_n| \leq q$. Встановлюється значення $c_0^{\#} = T_{l,k}^{(K)}(S)$, $|c_0^{\#}| = l$. Кожен з блоків шифртекста c_i ($|c_i| = q$) обчислюється відповідно до співвідношення $c_i = m_i \oplus R_{l,|m_i|}\left(c_{i-1}^{\#}\right)$ для $i = 1,2,..,n$ та $c_i^{\#} = T_{l,k}^{(K)}\left(L_{l,l-q}\left(c_{i-1}^{\#}\right) \| c_i\right)$ для $i = 1,2,..,n-1$, $|c_i^{\#}| = l$. Результатом зашифрування повідомлення є шифртекст $C = c_1 \| c_2 \| ... \| c_n$.

Шифртекст C ($|C| \geq 1$) подається у вигляді послідовності блоків: $C = c_1 \| c_2 \| ... \| c_n$, $|c_i| = q$ для $i = 1,2,..,n-1$, $1 \leq |c_n| \leq q$. Встановлюється значен-

ня $c_0^{\#} = T_{l,k}^{(K)}(S)$. Кожен з блоків повідомлення обчислюється відповідно до співвідношення $m_i = c_i \oplus R_{l,|c_i|}(c_{i-1}^{\#})$ для $i = 1,2,..,n$ та $c_i^{\#} = T_{l,k}^{(K)}(L_{l,l-q}(c_{i-1}^{\#}) \| c_i)$ для $i = 1,2,..,n-1$. Результатом розшифрування шифртекста є повідомлення $M = m_1 \| m_2 \| ... \| m_n$.

Вироблення імітовставки. Режим забезпечує цілісність повідомлення шляхом обчислення та перевірки імітовставки. Режим виконує відображення повідомлення O ($|O| = l \cdot n_o$, де n_o – додатне ціле) в імітовставку h, $|h| \in \{64,128,256,384,512\}$ при $|h| \le l$, а при перевірці цілісності додатково виконується порівняння обчисленої імітовставки із тією, що була отримана разом із повідомленням. Синхропосилка в цьому режимі не використовується. Параметрами режиму є ключ автентифікації K, $|K| = k$ та q – довжина імітовставки, $64 \le q \le l$. Рекомендоване значення є $q = l$. Режим вироблення імітовставки позначається як Калина-l/k-CMAC-q.

Повідомлення O ($|O| = l \cdot n_o$, де n_o – додатне ціле) подається у вигляді послідовності блоків: $O = o_1 \| o_2 \| ... \| o_{n_o}$, $|o_i| = l$ для $i = 1,2,..,n_o$. Встановлюється значення $c_0 = 0^l$. Якщо повідомлення було доповнене, то встановлюється $K_\delta = T_{l,k}^{(K)}(0x00..01)$, де $0x00..01$ – l-бітове подання 1 у форматі little endian; у іншому випадку $K_\delta = T_{l,k}^{(K)}(0^l)$. Для $i = 1,2,..,n_o - 1$ обчислюються $c_i = T_{l,k}^{(K)}(c_{i-1} \oplus o_i)$. Для $i = n_o$ задається $c_i = T_{l,k}^{(K)}(c_{i-1} \oplus o_i \oplus K_\delta)$. Результатом обчислення є імітовставка $h = L_{l,q}(c_{n_o})$.

Для повідомлення M застосовується алгоритм обчислення імітовставки.

У разі, якщо обчислена імітовставка не співпадає із тою, що була отримана разом із повідомленням, цілісність повідомлення є порушеною.

У іншому випадку цілісність повідомлення підтверджена.

Зчеплення шифрблоків. Режим забезпечує конфіденційність повідомлення шляхом шифрування. Якщо довжина повідомлення не є кратною розміру блоку базового перетворення, то застосовується алгоритм доповнення. Шифрування виконує пряме відображення повідомлення M ($|M| = l \cdot n$, де n – додатне ціле) у шифртекст C, $|C| = |M|$, та обернене відображення шифртексту C в повідомлення M. Параметрами режиму є ключ шифрування K, $|K| = k$, синхропосилка S, $|S| = l$. Додатковою вимогою до синхропосилки в цьому режимі є випадковість (непередбачуваність значення, яке буде застосовано для будь-якого повідомлення). Режим зчеплення шифр блоків позначається як Калина-l/k-CBC.

Повідомлення M подається у вигляді послідовності блоків: $M = m_1 \| m_2 \| ... \| m_n$, $|m_i| = l$ для $i = 1,2,..,n$. Встановлюється значення $c_0 = S$. Для $i = 1,2,..,n$ обчислюються $c_i = T_{l,k}^{(K)}(c_{i-1} \oplus m_i)$. Результатом зашифрування повідомлення є шифртекст $C = c_1 \| c_2 \| ... \| c_n$.

Шифртекст C ($|C| = l \cdot n$, де n – додатне ціле) подається у вигляді послідовності блоків: $C = c_1 \| c_2 \| ... \| c_n$, $|c_i| = l$ для $i = 1,2,..,n$. Встановлюється значення $c_0 = S$. Для $i = 1,2,..,n$ обчислюються $m_i = c_{i-1} \oplus U_{l,k}^{(K)}(c_i)$. Результатом розшифрування шифртекста є повідомлення $M = m_1 \| m_2 \| ... \| m_n$.

Гамування зі зворотнім зв'язком за шифргамою. Режим забезпечує конфіденційність повідомлення шляхом шифрування. Шифрування виконує пряме відображення повідомлення M ($|M| \ge 1$) у шифртекст C, $|C| = |M|$, та обернене відображення шифртексту C в повідомлення M. Вимоги на кратність довжини повідомлення розміру блоку базового перетворення не накладаються.

Параметрами режиму є ключ шифрування K, $|K| = k$ та синхропосилка S, $|S| = l$. Додаткові вимоги щодо синхропосилки не накладаються. Режим гамування позначається як Калина-l/k-OFB.

Повідомлення M ($|M| \geq 1$) подається у вигляді послідовності блоків: $M = m_1 \| m_2 \| ... \| m_n$, $|m_i| = l$ для $i = 1,2,..,n-1$, $1 \leq |m_n| \leq l$. Початкове значення блока гами γ_0 ($|\gamma_0| = l$) обчислюється як $\gamma_0 = T_{l,k}^{(K)}(S)$. Кожен з блоків шифртекста обчислюється відповідно до співвідношення $c_i = m_i \oplus L_{l,|m_i|}(\gamma_{i-1})$ для $i = 1,2,..,n$, та $\gamma_i = T_{l,k}^{(K)}(\gamma_{i-1})$ для $i = 1,2,..,n-1$. Результатом зашифрування повідомлення є шифртекст $C = c_1 \| c_2 \| ... \| c_n$. Шифртекст C ($|C| \geq 1$) подається у вигляді послідовності блоків: $C = c_1 \| c_2 \| ... \| c_n$, $|c_i| = l$ для $i = 1,2,..,n-1$, $1 \leq |c_n| \leq l$. Початкове значення блока гами γ_0 ($|\gamma_0| = l$) обчислюється як $\gamma_0 = T_{l,k}^{(K)}(S)$. Кожен з блоків повідомлення обчислюється відповідно до співвідношення $m_i = c_i \oplus L_{l,|m_i|}(\gamma_{i-1})$ для $i = 1,2,..,n$, та $\gamma_i = T_{l,k}^{(K)}(\gamma_{i-1})$ для $i = 1,2,..,n-1$. Результатом розшифрування шифртекста є повідомлення $M = m_1 \| m_2 \| ... \| m_n$.

Вибіркове гамування із прискореним виробленням імітовставки. Режим забезпечує конфіденційність і цілісність повідомлення шляхом шифрування і обчислення та перевірки імітовставки. Шифрування (гамування) є вибірковим, тобто конфіденційність забезпечується для обраної частини повідомлення (довжина цієї частини обирається в залежності від вимог до засобу криптографічного захисту: від шифрування всього повідомлення до відсутності шифрування взагалі). Повідомлення складається з двох частин: відкритої O (для якої буде забезпечена лише цілісність) та конфіденційної M (для якої буде забезпечена конфіденційність та цілісність), $|O| + |M| \geq 1$. Режим забезпечує цілісність відкритої частини повідомлення O та шифртекста C (зашифрованої частини повідомлення M) шляхом обчислення та перевірки імітовставки. Шифрування виконує пряме відображення повідомлення M ($|M| \geq 1$) у шифртекст C, $|C| = |M|$, та обернене відображення шифртексту C в повідомлення M. Крім того, виконується відображення криптограми (відкритої частини повідомлення O і шифртексту) в імітовставку h, $64 \leq |h| \leq l$, а при перевірці цілісності додатково виконується порівняння обчисленої імітовставки із тією, що була отримана разом із повідомленням. Вимоги на кратність довжини повідомлення (відкритої або конфіденційної частини) розміру блоку базового перетворення не накладаються. Параметрами режиму є ключ шифрування K, $|K| = k$, синхропосилка S, $|S| = l$ та q – довжина імітовставки, $64 \leq q \leq l$. Рекомендоване значення $q = l$. Додаткові вимоги щодо синхропосилки не накладаються. Режим вибіркового гамування із прискореним виробленням імітовставки для $|M| \geq 1$ позначається як Калина-l/k-GCM-q (забезпечується конфіденційність та цілісність), для $|M| = 0$ режим позначається як Калина-l/k - GMAC - q (забезпечується тільки цілісність).

Вироблення імітовставки є допоміжним алгоритмом, який використовується при прямому та оберному криптографічному перетворенні для обробки відкритої частини повідомлення O та шифртексту C. Якщо довжина шифртексту C не кратна розміру блоку базового перетворення ($|C| \neq n \cdot l$, $n \in \{0,1,2,...\}$), до нього застосовується алгоритм доповнення: $C^* = c_1^* \| c_2^* \| ... \| c_n^*$, де $|c_i^*| = l$ для $i = 1,2,..,n$. У іншому випадку (доповнення не потрібне) $C^* = C$. Коли довжина відкритої частини повідомлення O не кратна розміру блоку базового перетворення ($|O| \neq n_o \cdot l$,

$n_o \in \{0,1,2,...\}$), до неї застосовується алгоритм доповнення, $O^* = o_1^* \| o_2^* \| ... \| o_{n_o}^*$, де $|o_i^*| = l$ для $i = 1,2,..,n_o$. У іншому випадку (доповнення не потрібне) $O^* = O$. Значення параметризованої змінної автентифікації H ($|H| = l$) обчислюється як $H = T_{l,k}^{(K)}(0^l)$. Встановлюється значення $b_0 = 0^l$. Обчислюються значення $b_i = (o_i^* \oplus b_{i-1}) \bullet_l H$ для $i = 1,2,..,n_o$, $|b_i| = l$. Встановлюється значення $b_0' = b_{n_0}$, $|b_i'| = l$ для $i = 1,2,..,n$. Обчислюються значення $b_i' = (c_i^* \oplus b_{i-1}') \bullet_l H$ для $i = 1,2,..,n$. Встановлюється $B = b_n'$. Довжина відкритої та конфіденційної частини повідомлення (задана у бітах) подається у вигляді бітових послідовностей довжиною $l/2$ бітів кожна (формат little endian): $\lambda_o = |O|$, $\lambda_C = |C|$, $|\lambda_o| = |\lambda_C| = l/2$. Імітовставка h обчислюється як $h = L_{l,q}(T_{l,k}^{(K)}(B \oplus (\lambda_o \| \lambda_C)))$. Цей пункт визначає пряме перетворення Калина-l/k-GCM-q, коли присутня конфіденційна частина повідомлення M ($|M| \geq 1$).

M подається у вигляді послідовності блоків: $M = m_1 \| m_2 \| ... \| m_n$, $|m_i| = l$ для $i = 1,2,..,n-1$, $1 \leq |m_n| \leq l$. Початкове значення лічильника s_0 ($|s_0| = l$) обчислюється як $s_0 = T_{l,k}^{(K)}(S)$. Кожен з блоків шифртекста обчислюється відповідно до співвідношення

$$c_i = m_i \oplus L_{l,|m_i|}\left(T_{l,k}^{(K)}\left(L_{l,l/2}(s_0 + i) \| R_{l,l/2}(s_0)\right)\right)$$ для $i = 1,2,..,n$, $|c_i| = |m_i|$. Результатом зашифрування конфіденційної частини повідомлення є шифртекст $C = c_1 \| c_2 \| ... \| c_n$.

Результатом роботи прямого перетворення режиму Калина-l/k-GCM-q є шифртекст C та імітовставка h. Цей пункт визначає обернене перетворення Калина-l/k-GCM-q, коли у складі вхідних даних присутній шифртекст C ($|C| \geq 1$).

У разі, якщо обчислена імітовставка не співпадає із тою, що була отримана разом із вхідними даними, цілісність є порушеною. Обробка переривається та повертається повідомлення про порушення цілісності. Якщо цілісність підтверджена, то виконується розшифрування конфіденційної частини повідомлення.

Шифртекст C ($|C| \geq 1$) подається у вигляді послідовності блоків: $C = c_1 \| c_2 \| ... \| c_n$, $|c_i| = l$ для $i = 1,2,..,n-1$, $1 \leq |c_n| \leq l$. Початкове значення лічильника s_0 обчислюється як $s_0 = T_{l,k}^{(K)}(S)$. Кожен з блоків повідомлення обчислюється відповідно до співвідношення

$$m_i = c_i \oplus L_{l,|m_i|}\left(T_{l,k}^{(K)}\left(L_{l,l/2}(s_0 + i) \| R_{l,l/2}(s_0)\right)\right)$$ для $i = 1,2,..,n$. Результатом розшифрування шифртекста є повідомлення $M = m_1 \| m_2 \| ... \| m_n$. Результатом роботи оберненого перетворення режиму Калина-l/k-GCM-q є $M = m_1 \| m_2 \| ... \| m_n$ або повідомлення про порушення цілісності. Цей пункт визначає перетворення Калина-l/k-GMAC-q, коли відсутня конфіденційна частина повідомлення M ($|M| = 0$).

Якщо довжина відкритої частини повідомлення O не кратна розміру блоку базового перетворення ($|O| \neq n_o \cdot l$, $n_o \in \{1,2,...\}$), до неї застосовується алгоритм доповнення, $O^* = o_1^* \| o_2^* \| ... \| o_n^*$, де $|o_i^*| = l$ для $i = 1,2,..,n$. У іншому випадку (доповнення не потрібне) $O^* = O$. Значення параметризованої змінної автентифікації H ($|H| = l$) обчислюється як $H = T_{l,k}^{(K)}(0^l)$. Встановлюється значення $b_0 = 0^l$. Обчислюються значення $b_i = (o_i^* \oplus b_{i-1}) \bullet_l H$ для $i = 1,2,..,n_o$, $|b_i| = l$. Встановлюється $B = b_{n_o}$, $|B| = l$. Довжина відкритої та конфіденційної частини повідомлення подається у вигляді бітових послідовностей довжиною $l/2$ бітів кожна (формат little endian): $\lambda_o = |O|$,

$\lambda_M = 0^{l/2}$, $|\lambda_o| = |\lambda_M| = l/2$. Імітовставка h обчислюється як $h = L_{l,q}\left(T_{l,k}^{(K)}\left(B \oplus (\lambda_o \| \lambda_M)\right)\right)$. Результатом роботи режиму Калина-l/k - GMAC -q є імітовставка h. Цей пункт визначає перетворення Калина-l/k - GMAC -q, коли відсутня конфіденційна частина повідомлення M ($|M| = 0$), а для відкритої частини повідомлення вже обчислена імітовставка.

У разі, якщо обчислена імітовставка не співпадає із тою, що була отримана разом із повідомленням, цілісність повідомлення є порушеною. У іншому випадку цілісність повідомлення підтверджена.

Вироблення імітовставки і гамування. Режим забезпечує цілісність і конфіденційність повідомлення шляхом вироблення імітовставки та шифрування. Шифрування (гамування) є вибірковим, тобто конфіденційність може забезпечуватися лише для обраної частини повідомлення. На вхід режиму для прямого перетворення подається повідомлення, що складається з двох частин: відкритої O (для якої буде забезпечена лише цілісність), та конфіденційної M (для якої буде забезпечена конфіденційність та цілісність), бітова довжина обох частин є кратною 8: $|O| = 8 \cdot r'$, $r' \in \{0,1,2,...\}$ і $|M| = 8 \cdot r$, $r \in \{1,2,...\}$. Режим забезпечує цілісність обох частин повідомлення (O та M) і конфіденційність M.

Для оберненого перетворення на вхід подається відкрита частина O повідомлення та шифртекст, що був сформований при виконанні прямого перетворення. Параметрами режиму є ключ шифрування K, $|K| = k$, синхропосилка S, $|S| = l$, $N_{\max}$ – найбільша можлива довжина відкритої або конфіденційної частини повідомлення (в бітах), яке повинно бути оброблене засобом криптографічного захисту, та q – довжина імітовставки, яке обирається як $q \in \{64,128,256,384,512 | q \le l\}$. У якості $N_{\max}$ рекомендується обирати найменше значення, яке задовольняє практичним потребам (наприклад, коли довжина повідомлення завжди менша 4 ГБ, тобто не перевищує $2^{32}-1$ байтів, $N_{\max} = 2^{35} - 8$). Мінімальна необхідна кількість байтів $N_Б$ для збереження довжини повідомлення у байтах (тобто $8 \cdot N_Б$ бітів) обчислюється за формулою $N_Б = \left\lfloor \frac{1}{8}\left(-3 + log_2 N_{\max}\right) + 1 \right\rfloor$. Для прикладу $N_{\max} = 2^{35} - 8$ обчислене $N_Б = 4$.

Режим вироблення імітовставки і гамування позначається як Калина-l/k-CCM-$(8 \cdot N_Б), q$ (наприклад, Калина-256/512-CCM-32,128 визначає режим CCM з використанням базового перетворення з розміром блоку 256 бітів, довжиною ключа 512 бітів, довжина конфіденційної частини повідомлення завжди менша 2^{32} байтів і довжина імітовставки дорівнює 128 бітам).

Вироблення імітовставки є допоміжним алгоритмом, який використовується при прямому та оберненому криптографічному перетворенні для обробки відкритої частини повідомлення O та конфіденційної частини M. Формується заголовок автентифікації G_1, $|G_1| = l$, який складається із молодших байтів синхропосилки, до яких додане поле довжиною $N_Б$ байтів, що містить запис довжини конфіденційної частини повідомлення у форматі little endian, та байт прапорців таким чином, щоб загальний розмір заголовку дорівнював розміру блоку базового перетворення.

Якщо наявна відкрита частина повідомлення ($|O| > 0$), то довжина цієї частини подається у вигляді бітової послідовності довжиною $8 \cdot N_Б$ бітів (формат little endian): $\lambda_o = |O|/8$, $|\lambda_o| = 8 \cdot N_Б$. Формується блок довжини відкритої частини повідомлення G_2 шляхом додавання $\left(l - \left(|O| \bmod l\right) - 8 \cdot N_Б\right) \bmod l$ нульових бітів до довжини відкритої частини: $G_2 = \left(\lambda_o \| 0^{\left(l - \left(|O| \bmod l\right) - 8 \cdot N_Б\right) \bmod l}\right)$ таким чином, щоб

довжина послідовності $(G_1 \| G_2 \| O)$ була кратною довжині блоку базового перетворення. Послідовність $(G_1 \| G_2 \| O)$ подається у вигляді блоків $(G_1 \| G_2 \| O) = (g_1 \| g_2 \| g_3 \| ... \| g_{n_g})$, де $n_g = |(G_1 \| G_2 \| O)| / l$. Встановлюється значення $b_0 = 0^l$. Для $i = 1,2,..,n_g$ обчислюються $b_i = T_{l,k}^{(K)}(b_{i-1} \oplus g_i)$, $|b_i| = l$. Встановлюється $B = b_{n_g}$, $|B| = l$.

Якщо відкрита частина повідомлення відсутня ($|O| = 0$), то встановлюється $B = G_1$, $|B| = l$. У разі, коли довжина конфіденційної частини повідомлення M не є кратною розміру блоку базового перетворення, то застосовується алгоритм доповнення для формування доповненої конфіденційної частини $M^* = m_1^* \| m_2^* \| ... \| m_n^*$, де $|m_i^*| = l$ для $i = 1,2,..,n$. Встановлюється значення $b_0' = B$, $|b_0'| = l$. Для $i = 1,2,..,n$ обчислюються $b_i' = T_{l,k}^{(K)}(b_{i-1}' \oplus m_i^*)$, $|b_i| = l$. Імітовставка h обчислюється як $h = L_{l,q}(b_n')$. Для відкритої частини повідомлення O та конфіденційної частини повідомлення M обчислюється імітовставка h. Повідомлення для зашифрування M'' складається з конфіденційної частини повідомлення (M) та імітовставки h: $M'' = M \| h$, $M'' = m_1'' \| m_2'' \| ... \| m_{n_m}''$, $|m_i''| = l$ для $i = 1,2,..,n_m - 1$, $1 \le |m_{n_m}''| \le l$. Початкове значення лічильника s_0 ($|s_0| = l$) обчислюється як $s_0 = T_{l,k}^{(K)}(S)$.

Кожен з блоків шифртекста обчислюється відповідно до співвідношення $c_i = m_i'' \oplus L_{l,|m_i''|}\left(T_{l,k}^{(K)}\left(L_{l,l/2}(s_0 + i) \| R_{l,l/2}(s_0)\right)\right)$ для $i = 1,2,..,n_m$, $|c_i| = |m_i''|$. Результатом роботи режиму є шифртекст $C = c_1 \| c_2 \| ... \| c_n$.

При розшифруванні шифртекст C ($|C| \ge 1$) подається у вигляді послідовності блоків: $C = c_1 \| c_2 \| ... \| c_n$, $|c_i| = l$ для $i = 1,2,..,n-1$, $1 \le |c_n| \le l$. Початкове значення лічильника s_0 обчислюється як $s_0 = T_{l,k}^{(K)}(S)$. Кожен з блоків отриманої бітової послідовності обчислюється відповідно до співвідношення $m_i'' = c_i \oplus L_{l,|m_i|}\left(T_{l,k}^{(K)}\left(L_{l,l/2}(s_0 + i) \| R_{l,l/2}(s_0)\right)\right)$ для $i = 1,2,..,n$. Результатом розшифрування шифртекста є $M'' = m_1'' \| m_2'' \| ... \| m_{n_m}''$, $|m_i''| = l$ для $i = 1,2,..,n_m - 1$, $1 \le |m_{n_m}''| \le l$.

З M'' отримується імітовставка $h' = R_{|M''|,q}(M'')$ та конфіденційна частина повідомлення $M = L_{|M''|,|M''|-q}(M'')$. Для відкритої частини повідомлення O та розшифрованої конфіденційної частини повідомлення M обчислюється імітовставка h. У разі, якщо обчислена імітовставка не співпадає із тою, що була отримана разом із повідомленням ($h \ne h'$), обробка переривається та повертається повідомлення про порушення цілісності. У іншому випадку цілісність повідомлення підтверджена, і результатом роботи режиму є конфіденційна частина повідомлення M.

Індексована заміна. Режим забезпечує конфіденційність повідомлення шляхом шифрування. Це перетворення не забезпечує криптографічну послугу збереження цілісності повідомлення, але у випадку модифікації будь-якого блоку шифртексту відповідний блок відкритого тексту після розшифрування буде мати псевдовипадкове значення (його зміст буде цілком зіпсований), а інші блоки залишаться непошкодженими. Шифрування виконує пряме відображення повідомлення M ($|M| \ge l$) у шифртекст C, $|C| = |M|$, та обернене відображення шифртексту C в повідомлення M. Вимоги на кратність довжини повідомлення розміру блоку базового перетворення не накладаються. Параметрами режиму є ключ шифрування K, $|K| = k$ та синхропосилка S, $|S| = l$.

Додаткові вимоги щодо синхропосилки не на-

кладаються. У разі, коли розмір повідомлення є кратним розміру блоку базового перетворення, виконується шифрування без доповнення.

У іншому випадку застосовується модифікований алгоритм із доповненням. Режим індексованої заміни позначається як Калина-l/k-XTS (без доповнення) або Калина-l/k-XTS-p (із доповненням).

Повідомлення M ($|M| = l \cdot r$, де r – додатне ціле) подається у вигляді послідовності блоків: $M = m_1 \| m_2 \| ... \| m_n$, $|m_i| = l$ для $i = 1,2,..,n$. Початкове значення лічильника s_0 ($|s_0| = l$) обчислюється як $s_0 = T_{l,k}^{(K)}(S)$. Кожен з блоків шифртекста обчислюється відповідно до співвідношення $c_i = \left(T_{l,k}^{(K)}\left(m_i \oplus \left(\alpha_l^i \bullet_l s_0\right)\right)\right) \oplus \left(\alpha_l^i \bullet_l s_0\right)$ для $i = 1,2,..,n$, $|c_i| = |m_i|$. Результатом зашифрування повідомлення є шифртекст $C = c_1 \| c_2 \| ... \| c_n$.

Шифртекст C ($|C| = l \cdot r$, де r – додатне ціле) подається у вигляді послідовності блоків: $C = c_1 \| c_2 \| ... \| c_n$, $|c_i| = l$ для $i = 1,2,..,n$.

Початкове значення лічильника s_0 обчислюється як $s_0 = T_{l,k}^{(K)}(S)$. Кожен з блоків повідомлення обчислюється відповідно до співвідношення $m_i = \left(U_{l,k}^{(K)}\left(c_i \oplus \left(\alpha_l^i \bullet_l s_0\right)\right)\right) \oplus \left(\alpha_l^i \bullet_l s_0\right)$ для $i = 1,2,..,n$. Результатом розшифрування є повідомлення $M = m_1 \| m_2 \| ... \| m_n$.

Повідомлення M ($|M| > l$) подається у вигляді послідовності блоків: $M = m_1 \| m_2 \| ... \| m_n$, $|m_i| = l$ для $i = 1,2,..,n-1$, $1 \le |m_n| < l$. Блоки $m_1, m_2, ..., m_{n-1}$ обробляються для отримання фрагменту шифртексту $C^* = c_1 \| c_2 \| ... \| c_{n-2} \| c_{n-1}$. Обчислюється $c_n = \left(T_{l,k}^{(K)}\left(\left(m_n \| R_{l,l-|m_n|}(c_{n-1})\right) \oplus \left(\alpha_l^n \bullet_l s_0\right)\right)\right) \oplus \left(\alpha_l^n \bullet_l s_0\right)$, де $s_0 = T_{l,k}^{(K)}(S)$.

Результатом зашифрування повідомлення є шифртекст $C = c_1 \| c_2 \| ... \| c_{n-2} \| c_n \| L_{l,|m_n|}(c_{n-1})$.

Шифртекст C ($|C| > l$) подається у вигляді послідовності блоків: $C = c_1 \| c_2 \| ... \| c_n$, $|c_i| = l$ для $i = 1,2,..,n-1$, $1 \le |c_n| < l$. Блоки $c_1, c_2, ..., c_{n-2}$ обробляються для отримання фрагменту відкритого тексту $M^* = m_1 \| m_2 \| ... \| m_{n-2}$. Обчислюються $m_n^* = \left(U_{l,k}^{(K)}\left(c_{n-1} \oplus \left(\alpha_l^n \bullet_l s_0\right)\right)\right) \oplus \left(\alpha_l^n \bullet_l s_0\right)$ і

$$m_{n-1} = \left(U_{l,k}^{(K)}\left(\left(c_n \| R_{l,l-|c_n|}(m_n^*)\right) \oplus \left(\alpha_l^{n-1} \bullet_l s_0\right)\right)\right) \oplus \left(\alpha_l^{n-1} \bullet_l s_0\right)$$

, де $s_0 = T_{l,k}^{(K)}(S)$. Результатом розшифрування є повідомлення $M = m_1 \| m_2 \| ... \| m_{n-1} \| L_{l,|c_n|}(m_n^*)$.

Захист ключових даних. Режим забезпечує конфіденційність та цілісність повідомлення. Шифрування виконує пряме відображення повідомлення M ($|M| \ge l$) у шифртекст C, $|M| < |C| < |M| + 2 \cdot l$, та обернене відображення шифртексту C в повідомлення M. Вимоги на кратність довжини повідомлення розміру блоку базового перетворення не накладаються. Параметром режиму є ключ шифрування K, $|K| = k$. У разі, коли розмір повідомлення є кратним розміру блоку базового перетворення, виконується шифрування без доповнення.

У іншому випадку застосовується алгоритм із доповненням. Режим захисту ключових даних позначається як Калина-l/k-KW (без доповнення) або Калина-l/k-KW-p (із доповненням). До повідомлення M ($|M| = l \cdot r$, де r – додатне ціле) додається 0^l для отримання M^*: $M^* = M \| 0^l$. M^* подається у вигляді послідовності напівблоків: $M^* = m_1^* \| m_2^* \| ... \| m_n^*$, $|m_i^*| = l/2$ для $i = 1,2,..,n$ та $n = 2 \cdot (r+1)$. Встановлюється $V = (n-1) \cdot 6$ і $B^0 = m_1^*$, де $|B^j| = l/2$ для $j = 0,1,...,V$. Задається $b_i^0 = m_i^*$ для $i = 2,..,n$, де $|b_i^j| = l/2$ для

$j = 0,1,...,V$. Для $j = 1,...,V$ обчислюється $B^j = R_{l,l/2}\left(T_{l,k}^{(K)}\left(B^{j-1} \| b_2^{j-1}\right)\right) \oplus \mu_{l/2}^{(j)}$, $b_n^j = L_{l,l/2}\left(T_{l,k}^{(K)}\left(B^{j-1} \| b_2^{j-1}\right)\right)$ та $b_i^j = b_{i+1}^{j-1}$ для $i = 2,..,n-1$. Задається $c_1 = B^V$ і $c_i = b_i^V$ для $i = 2,..,n$. Результатом є шифртекст $C = c_1 \| c_2 \| ... \| c_n$.

Шифртекст C ($|C| = l \cdot r$, де r – додатне ціле) подається у вигляді послідовності напівблоків: $C = c_1 \| c_2 \| ... \| c_n$, $|c_i| = l/2$ для $i = 1,2,..,n$ та $n = 2 \cdot r$. Встановлюється $V = (n-1) \cdot 6$ і $B^V = c_1$, де $|B^j| = l/2$ для $j = 0,1,...,V$. Задається $b_i^V = c_i$ для $i = 2,..,n$, де $|b_i^j| = l/2$ для $j = 0,1,...,V$. Для $j = V, V-1,...,1$ обчислюється $B^{j-1} = L_{l,l/2}\left(U_{l,k}^{(K)}\left(b_n^j \| \left(B^j \oplus \mu_{l/2}^{(j)}\right)\right)\right)$, $b_2^{j-1} = R_{l,l/2}\left(U_{l,k}^{(K)}\left(b_n^j \| \left(B^j \oplus \mu_{l/2}^{(j)}\right)\right)\right)$ і $b_{i+1}^{j-1} = b_i^j$ для $i = 2,..,n-1$. Задається $m_1^* = B^0$ і $m_i^* = b_i^0$ для $i = 2,..,n$. Формується $M^* = m_1^* \| m_2^* \| ... \| m_n^*$. У разі, коли $R_{n \cdot l/2,l}\left(M^*\right)$ не дорівнює 0^l, повертається повідомлення про порушення цілісності. У іншому випадку повертається розшифроване повідомлення $M = L_{n \cdot l/2, n \cdot l/2-l}\left(M^*\right)$. До повідомлення M ($|M| > l$) додається бітове подання довжини $\mu_{l/2}^{(|M|)}$, після чого до результату $\left(M \| \mu_{l/2}^{(|M|)}\right)$ застосовується алгоритм доповнення для отримання доповненого повідомлення $M'' = m_1'' \| m_2'' \| ... \| m_{n_m}''$, $|m_i''| = l$ для $i = 1,2,..,n_m$.

Доповнене повідомлення M'' обробляється для отримання шифртексту $C = c_1 \| c_2 \| ... | c_{n_m}$, який є результатом роботи режиму. Шифртекст C ($|C| = l \cdot r$, де r – додатне ціле) обробляються для отримання доповненого повідомлення M''. Якщо результатом роботи є повідомлення про порушення цілісності, подальша обробка припиняється із повертанням повідомлення про порушення цілісності. У іншому випадку до доповненого повідомлення M'' застосовується алгоритм зняття доповнення повідомлення. Якщо результатом є помилка оберненого перетворення, подальша обробка припиняється із повідомленням про порушення цілісності. У разі, коли $\mu_{l/2}^{(|M''|-l/2)} \neq R_{|M''|,l/2}\left(M''\right)$, повертається повідомлення про порушення цілісності. При $\mu_{l/2}^{(|M''|-l/2)} = R_{|M''|,l/2}\left(M''\right)$ результатом роботи режиму є відкритий текст $M = L_{|M''|,|M''|-l/2}\left(M''\right)$.

Таким чином, ДСТУ 7624:2014 визначає десять різних режимів роботи (застосування), які широко поширені відповідно до міжнародного стандарту ISO/IEC 10116:2006. Це спрямовано на забезпечення широкої застосовності ДСТУ 7624:2014, у тому числі для захисту інформації, що передається комп'ютерними мережами (Інтернет), прозорого шифрування жорстких дисків та знімних носіїв, електронних документів, ключових даних тощо. Наявність такої кількості режимів роботи надає можливість ефективної реалізації систем, засобів і протоколів криптографічного захисту інформації в інформаційно-телекомунікаційних системах різноманітного призначення.

Криптографічний алгоритм блокового шифрування, що визначається ДСТУ 7624:2014, є гнучким, підтримує розмір блока та довжину ключа від 128 до 512 бітів. Порівняно з відомим міжнародним стандартом AES (ISO/IEC 18033-3:2010), алгоритм ДСТУ 7624:2014 забезпечує вищий рівень криптографічної стійкості (із можливістю застосування блока та ключа шифрування включно до 512 бітів) і аналогічну або вищу швидкодію на сучасних і перспективних програмних і програмно-апаратних платформах.

Підвищена довжина внутрішнього стану унеможливлює ефективне застосування квантових засобів криптографічного аналізу, що робить цей алгоритм найбільш придатним для практичного застосування в пост-квантовому середовищі, в умовах ведення інформаційних та гібридних війн.

Національний стандарт ДСТУ 8845:2019

«Інформаційні технології. Криптографічний захист інформації. Алгоритм симетричного потокового перетворення»

Потоковий симетричний шифр, що описаний у національному стандарті України ДСТУ 8845:2019 «Інформаційні технології. Криптографічний захист інформації. Алгоритм симетричного потокового перетворення», визначає алгоритм генерації ключового потоку «СТРУМОК» (англ. STRUMOK), відповідний стандарт набрав чинності з 1 жовтня 2019 року наказом ДП «УкрНДНЦ» від 2 квітня 2019 року № 85 [14, 15].

Основними структурними компонентами генератору ключових потоків *СТРУМОК* є регістр зсуву з лінійним зворотнім зв'язком (*РЗЛЗЗ*) та скінченний автомат (*СА*), в якому виконується нелінійне перетворення. Криптоалгоритм орієнтований на 64-розрядні обчислювальні системи, отже розмір слова визначено рівним 64 бітам. Для запису байтів використовується подання від старшого до молодшого.

Вхідні дані використовуються для ініціалізації змінної стану $S_i (i \geq 0)$, яка складається з вісімнадцяти 64-бітових блоків, до складу яких входить дві компоненти: 16 змінних $s^{(i)}$ – комірок регістра зсуву з лінійним зворотнім зв'язком: $s^{(i)} = (s_{15}^{(i)}, s_{14}^{(i)}, ..., s_0^{(i)})$; два регістри скінченного автомату $r^{(i)}$: $r^{(i)} = (r_2^{(i)}, r_1^{(i)})$.

На виході отримуємо ключовий потік (гаму шифру), який формується з 64-бітових слів Z_i. Схематичне зображення генератору ключових потоків *СТРУМОК* у режимі генерації гами шифру наведено на рисунку 3.

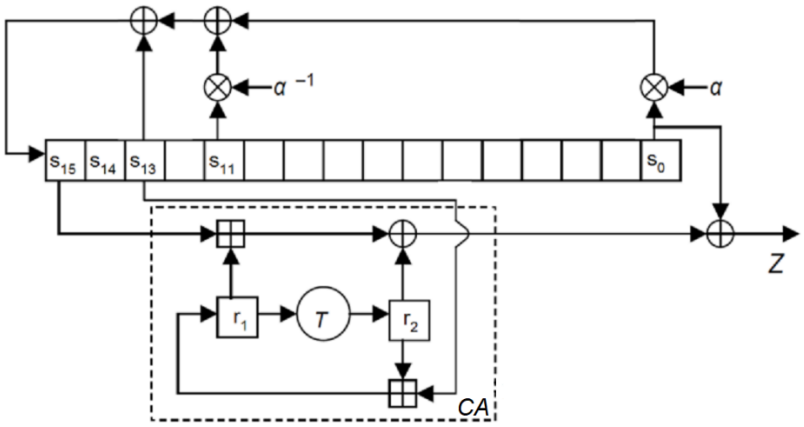


Рис. 3. Схематичне зображення генератору ключових потоків *СТРУМОК* у режимі генерації гами шифру

На рис. 3 зображено функціонування генератора в довільний момент часу *i*. Змінну часової залежності *i* не наведено. Відводи зворотного зв'язку у *РЗЛЗЗ* будується за примітивним над полем $GF(2^{64})$ поліномом $f(x) = x^{16} + x^{13} + \alpha^{-1}x^{11} + \alpha$, де α є коренем примітивного над полем $GF(2^8)$ поліному

$g(z) = z^8 + \beta^{170}z^7 + \beta^{166}z^6 + \beta^2 z^5 + \beta^{224}z^4 + \beta^{70}z^3 + \beta^2$

. В свою чергу поле $GF(2^8)$ будується за примітивним над полем $GF(2)$ поліномом $p(y) = y^8 + y^4 + y^3 + y^2 + 1$, а коефіцієнти поліному $g(z)$ подаються через ступінь примітивного елементу β поля $GF(2^8)$, тобто β – корінь поліному $p(y)$. Таким чином, маємо вежу полів: $GF(2) \subset GF(2^8) \subset GF(2^{64}) \subset GF(2^{1024})$, де: поле $GF(2^{1024})$ задається відводами зворотного зв'язку *РЗЛЗЗ* як факторкільце $GF(2^{64})[x]/(f(x))$, поле $GF(2^{64})$ задається як факторкільце

$GF(2^8)[z]/(g(z))$, поле $GF(2^8)$ задається як факторкільце $GF(2)[y]/(p(y))$. Отже період вихідної послідовності *РЗЛЗЗ* є максимальним і дорівнює $2^{1024}-1$.

Структурно в алгоритмі *СТРУМОК* можна виділити три основні функції: функція ініціалізації $Init$, яка приймає в якості вхідних даних ключ K (256 біт або 512 біта) і вектор ініціалізації IV (256 біт), і виробляє початкове значення змінної стану $S_0 = (s^{(0)}, r^{(0)})$; функція наступного стану *Next*, яка приймає на вхід змінну стану $S_i = (s^{(i)}, r^{(i)})$ і виробляє наступне значення змінної стану $S_{i+1} = (s^{(i+1)}, r^{(i+1)})$. Функція *Next* може виконуватися в двох режимах, в залежності від способу виконання ітерації – як частини реалізації або як частини нормального режиму генерації вихідних даних; функція ключового потоку *Strm*, що приймає на вході змінну стану $S_i = (s^{(i)}, r^{(i)})$ і виробляє на виході 64-бітний ключовий потік Z_i.

Функція ініціалізації внутрішнього стану *Init* описується наступним чином.

Вхід: 256 або 512-бітний ключ K, 256-бітний вектор ініціалізації IV.

Вихід: початкове значення змінної стану $S_0 = (s^{(0)}, r^{(0)})$. Ключ для режиму *СТРУМОК-256* можна представити у вигляді чотирьох 64-бітних слів $K = (K_3, K_2, K_1, K_0)$, а для 512-бітного ключа – у вигляді восьми 64-бітних слів $K = (K_7, K_6, K_5, K_4, K_3, K_2, K_1, K_0)$ де K_3 та K_7, відповідно для 256 и 512 біт, найбільш значущі слова, а K_0 – найменш значущі. Вектор ініціалізації можна представити у вигляді чотирьох 64-бітних слів $IV = (IV_3, IV_2, IV_1, IV_0)$ де IV_3 – найбільш значуще слово, а IV_0 – найменш значуще.

1. В 16 комірок РЗЛЗЗ заноситься значення ключа.

Для версії з 256-бітним ключем виконуються операції:

$$s_{15}^{(-33)} = \neg K_0, \quad s_{14}^{(-33)} = K_1, \quad s_{13}^{(-33)} = \neg K_2,$$
$$s_{12}^{(-33)} = K_3, \quad s_{11}^{(-33)} = K_0, \quad s_{10}^{(-33)} = \neg K_1, \quad s_9^{(-33)} = K_2,$$
$$s_8^{(-33)} = K_3, \quad s_7^{(-33)} = \neg K_0, \quad s_6^{(-33)} = \neg K_1,$$
$$s_5^{(-33)} = K_2 \oplus IV_3, \quad s_4^{(-33)} = K_3, \quad s_3^{(-33)} = K_0 \oplus IV_2,$$
$$s_2^{(-33)} = K_1 \oplus IV_1, \; s_1^{(-33)} = K_2, \; s_0^{(-33)} = K_3 \oplus IV_0.$$

Для версії з 512-бітним ключем K виконуються операції:

$$s_{15}^{(-33)} = K_0, \; s_{14}^{(-33)} = \neg K_1, \; s_{13}^{(-33)} = K_2, \; s_{12}^{(-33)} = K_3,$$
$$s_{11}^{(-33)} = \neg K_7, \quad s_{10}^{(-33)} = K_5, \quad s_9^{(-33)} = \neg K_6,$$
$$s_8^{(-33)} = K_4 \oplus IV_3, \quad s_7^{(-33)} = \neg K_0, \quad s_6^{(-33)} = K_1,$$
$$s_5^{(-33)} = K_2 \oplus IV_2, \quad s_4^{(-33)} = K_3, \quad s_3^{(-33)} = K_4 \oplus IV_1,$$
$$s_2^{(-33)} = K_5, \; s_1^{(-33)} = K_6, \; s_0^{(-33)} = K_7 \oplus IV_0.$$

2. Виконується 32 ініціюючих такти без генерації ключового потоку, тобто 4 повних циклів. Формально це подається наступним чином:

$$S_{-1} = Next^{32}(S_{-33}, INIT),$$

що означає 32 ітерації з виконання функції *Next* у режимі ініціалізації *INIT*, $S_{-33} = (s^{(-33)}, r^{(-33)})$ –значення змінної стану: обраховані на попередньому кроці 16 комірок регістра зсуву $s^{(-33)}$ та початкові нульові значення двох регістрів $r^{(-33)} = (r_2^{(-33)}, r_1^{(-33)})$ скінченного автомату, що у шістнадцятковому поданні мають вигляд $r^{(-33)} = (0000000000000000, 0000000000000000)$.

3. Розраховується початкове значення змінної стану $S_0 = (s^{(0)}, r^{(0)})$ за правилом:

$$S_0 = Next(S_{-1}),$$

тобто шляхом виконання функції *Next* у звичайному режимі.

4. Виводиться вихідне значення $S_0 = (s^{(0)}, r^{(0)})$.

Функція наступного стану *Next* описується наступним чином.

Вхід: Змінна стану $S_i = (s^{(i)}, r^{(i)})$, обраний режим (звичайний, або режим ініціалізації). За замовчуванням використовується звичайний режим.

Вихід: Наступне значення змінної стану $S_{i+1} = (s^{(i+1)}, r^{(i+1)})$.

1. Виконується нелінійна підстановка для оновлення значення регістру $r_2^{(i+1)}$ скінченного автома-

ту. Для цього розраховується значення функції T: $r_2^{(i+1)} = T(r_1^{(i)})$.

2. Оновлюється значення регістру $r_1^{(i+1)}$ скінченного автомату. Для цього розраховується значення: $r_1^{(i+1)} = r_2^{(i)} +_{64} s_{13}^{(i)}$, де $+_{64}$ позначає операцію додавання цілих чисел за модулем 2^{64} (у схемі на рисунках 1 та 2 цю операцію позначено як $\boxplus$).

3. Оновлюється значення 15 комірок *РЗЛЗЗ*

$s_j^{(i+1)} = s_{j+1}^{(i)}$

для всіх $j = 0, 1, \ldots, 14$.

4. Оновлюється значення 16-ї комірки *РЗЛЗЗ*. Якщо встановлено звичайний режим функції *Next* значення цієї комірки обчислюється за правилом:

$$s_{15}^{(i+1)} = (s_0^{(i)} \otimes \alpha) \oplus (s_{11}^{(i)} \otimes \alpha^{-1}) \oplus s_{13}^{(i)}.$$

Якщо встановлено режим ініціалізації функції *Next* значення обчислюється за правилом:

$$s_{15}^{(i+1)} = FSM(s_{15}^{(i)}, r_1^{(i)}, r_2^{(i)}) \oplus (s_0^{(i)} \otimes \alpha) \oplus (s_{11}^{(i)} \otimes \alpha^{-1}) \oplus s_{13}^{(i)}.$$

Операції множення $\otimes$ на α та на α^{-1}, а також сутність функції *СА* пояснюються далі.

5. Обчислюється та виводиться значення змінної стану $S_i = (s^{(i)}, r^{(i)})$.

Функція ключового потоку *Strm* описується таким чином.

Вхід: Змінна стану $S_i = (s^{(i)}, r^{(i)})$.

Вихід: 64-бітовий ключовий потік Z_i.

1. Обчислюється значення $Z_i = FSM(s_{15}^{(i)}, r_1^{(i)}, r_2^{(i)}) \oplus s_0^{(i)}$.

2. Виводиться вихідне значення Z_i.

Функція скінченного автомату позначається як $FSM(x, y, z)$ та описується наступним чином.

Вхід: три 64-бітових рядка x, y і z.

Вихід: 64-бітовий рядок *q*.

1. Обчислюється значення $q = (x +_{64} y) \oplus z$.

2. Виводиться вихідне значення *q*.

Функція нелінійної підстановки *T* реалізує перестановку елементів скінченного поля $GF(2^{64})$ за допомогою компонентів національного стандарту блокового симетричного криптоперетворення ДСТУ 7624:2014.

Вхід: 64-бітовий рядок *w*.

Вихід: 64-бітовий рядок $T = T(w)$.

1. Вхідний 64-бітовий рядок *w* розбивається на підблоки w_j по 8 біт:

$$w = (w_7, w_6, w_5, w_4, w_3, w_2, w_1, w_0),$$

2. Для кожного підблоку w_j виконується підстановка з алгоритму ДСТУ 7624:2014 за допомогою чотирьох табличних перетворень $\pi_0, \pi_1, \pi_2, \pi_3$.

Підстановка π_0:

A8 43 5F 06 6B 75 6C 59 71 DF 87 95 17 F0 D8 09
6D F3 1D CB C9 4D 2C AF 79 E0 97 FD 6F 4B 45
39
3E DD A3 4F B4 B6 9A 0E 1F BF 15 E1 49 D2 93
C6
92 72 9E 61 D1 63 FA EE F4 19 D5 AD 58 A4 BB
A1
DC F2 83 37 42 E4 7A 32 9C CC AB 4A 8F 6E 04 27
2E E7 E2 5A 96 16 23 2B C2 65 66 0F BC A9 47 41
34 48 FC B7 6A 88 A5 53 86 F9 5B DB 38 7B C3 1E
22 33 24 28 36 C7 B2 3B 8E 77 BA F5 14 9F 08 55
9B 4C FE 60 5C DA 18 46 CD 7D 21 B0 3F 1B 89
FF
EB 84 69 3A 9D D7 D3 70 67 40 B5 DE 5D 30 91 B1
78 11 01 E5 00 68 98 A0 C5 02 A6 74 2D 0B A2 76
B3 BE CE BD AE E9 8A 31 1C EC F1 99 94 AA F6
26
2F EF E8 8C 35 03 D4 7F FB 05 C1 5E 90 20 3D 82
F7 EA 0A 0D 7E F8 50 1A C4 07 57 B8 3C 62 E3 C8
AC 52 64 10 D0 D9 13 0C 12 29 51 B9 CF D6 73 8D
81 54 C0 ED 4E 44 A7 2A 85 25 E6 CA 7C 8B 56 80

Підстановка π_1:

CE BB EB 92 EA CB 13 C1 E9 3A D6 B2 D2 90 17
F8
42 15 56 B4 65 1C 88 43 C5 5C 36 BA F5 57 67 8D
31 F6 64 58 9E F4 22 AA 75 0F 02 B1 DF 6D 73 4D
7C 26 2E F7 08 5D 44 3E 9F 14 C8 AE 54 10 D8 BC
1A 6B 69 F3 BD 33 AB FA D1 9B 68 4E 16 95 91
EE
4C 63 8E 5B CC 3C 19 A1 81 49 7B D9 6F 37 60 CA
E7 2B 48 FD 96 45 FC 41 12 0D 79 E5 89 8C E3 20

30 DC B7 6C 4A B5 3F 97 D4 62 2D 06 A4 A5 83 5F
2A DA C9 00 7E A2 55 BF 11 D5 9C CF 0E 0A 3D
51
7D 93 1B FE C4 47 09 86 0B 8F 9D 6A 07 B9 B0 98
18 32 71 4B EF 3B 70 A0 E4 40 FF C3 A9 E6 78 F9
8B 46 80 1E 38 E1 B8 A8 E0 0C 23 76 1D 25 24 05
F1 6E 94 28 9A 84 E8 A3 4F 77 D3 85 E2 52 F2 82
50 7A 2F 74 53 B3 61 AF 39 35 DE CD 1F 99 AC
AD
72 2C DD D0 87 BE 5E A6 EC 04 C6 03 34 FB DB
59
B6 C2 01 F0 5A ED A7 66 21 7F 8A 27 C7 C0 29 D7

Підстановка π_2:

93 D9 9A B5 98 22 45 FC BA 6A DF 02 9F DC 51
59
4A 17 2B C2 94 F4 BB A3 62 E4 71 D4 CD 70 16 E1
49 3C C0 D8 5C 9B AD 85 53 A1 7A C8 2D E0 D1
72
A6 2C C4 E3 76 78 B7 B4 09 3B 0E 41 4C DE B2 90
25 A5 D7 03 11 00 C3 2E 92 EF 4E 12 9D 7D CB 35
10 D5 4F 9E 4D A9 55 C6 D0 7B 18 97 D3 36 E6 48
56 81 8F 77 CC 9C B9 E2 AC B8 2F 15 A4 7C DA
38
1E 0B 05 D6 14 6E 6C 7E 66 FD B1 E5 60 AF 5E
33
87 C9 F0 5D 6D 3F 88 8D C7 F7 1D E9 EC ED 80
29
27 CF 99 A8 50 0F 37 24 28 30 95 D2 3E 5B 40 83
B3 69 57 1F 07 1C 8A BC 20 EB CE 8E AB EE 31
A2
73 F9 CA 3A 1A FB 0D C1 FE FA F2 6F BD 96 DD
43
52 B6 08 F3 AE BE 19 89 32 26 B0 EA 4B 64 84 82
6B F5 79 BF 01 5F 75 63 1B 23 3D 68 2A 65 E8 91
F6 FF 13 58 F1 47 0A 7F C5 A7 E7 61 5A 06 46 44
42 04 A0 DB 39 86 54 AA 8C 34 21 8B F8 0C 74 67

Підстановка π_3:

68 8D CA 4D 73 4B 4E 2A D4 52 26 B3 54 1E 19 1F
22 03 46 3D 2D 4A 53 83 13 8A B7 D5 25 79 F5 BD
58 2F 0D 02 ED 51 9E 11 F2 3E 55 5E D1 16 3C 66
70 5D F3 45 40 CC E8 94 56 08 CE 1A 3A D2 E1
DF
B5 38 6E 0E E5 F4 F9 86 E9 4F D6 85 23 CF 32 99
31 14 AE EE C8 48 D3 30 A1 92 41 B1 18 C4 2C 71
72 44 15 FD 37 BE 5F AA 9B 88 D8 AB 89 9C FA
60
EA BC 62 0C 24 A6 A8 EC 67 20 DB 7C 28 DD AC
5B
34 7E 10 F1 7B 8F 63 A0 05 9A 43 77 21 BF 27 09
C3 9F B6 D7 29 C2 EB C0 A4 8B 8C 1D FB FF C1
B2
97 2E F8 65 F6 75 07 04 49 33 E4 D9 B9 D0 42 C7
6C 90 00 8E 6F 50 01 C5 DA 47 3F CD 69 A2 E2 7A
A7 C6 93 0F 0A 06 E6 2B 96 A3 1C AF 6A 12 84 39
E7 B0 82 F7 FE 9D 87 5C 81 35 DE B4 A5 FC 80
EF
CB BB 6B 76 BA 5A 7D 78 0B 95 E3 AD 74 98 3B
36
64 6D DC F0 59 A9 4C 17 7F 91 B8 C9 57 1B E0 61

У результаті формується вихідний вектор $r = (r_7, r_6, r_5, r_4, r_3, r_2, r_1, r_0)$: $r_j = \pi_{j \bmod 4}\left[w_j\right]$, де $j = 0, 1, \ldots, 7$.

3. Обчислюється вектор $q = (q_7, q_6, q_5, q_4, q_3, q_2, q_1, q_0)$ за правилом:

$$\begin{pmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \\ q_4 \\ q_5 \\ q_6 \\ q_7 \end{pmatrix} = \begin{pmatrix} 01 & 01 & 05 & 01 & 08 & 06 & 07 & 04 \\ 04 & 01 & 01 & 05 & 01 & 08 & 06 & 07 \\ 07 & 04 & 01 & 01 & 05 & 01 & 08 & 06 \\ 06 & 07 & 04 & 01 & 01 & 05 & 01 & 08 \\ 08 & 06 & 07 & 04 & 01 & 01 & 05 & 01 \\ 01 & 08 & 06 & 07 & 04 & 01 & 01 & 05 \\ 05 & 01 & 08 & 06 & 07 & 04 & 01 & 01 \\ 01 & 05 & 01 & 08 & 06 & 07 & 04 & 01 \end{pmatrix} \cdot \begin{pmatrix} r_0 \\ r_1 \\ r_2 \\ r_3 \\ r_4 \\ r_5 \\ r_6 \\ r_7 \end{pmatrix},$$

де елементи матриці (подано у шітнадцятковому вигляді) та векторів r і q інтерпретуються як елементи скінченного поля $GF(2^8)$, яке задане як факторкільце $GF(2)[y]/(p(y))$.

4. Виводиться вихідне значення q, яке інтерпретується як 64-бітовий рядок.

Швидке обчислення вектору q можна реалізувати за правилом:

$$\begin{pmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \\ q_4 \\ q_5 \\ q_6 \\ q_7 \end{pmatrix} = \begin{matrix} T_0[w_0] \oplus T_1[w_1] \oplus T_2[w_2] \oplus T_3[w_3] \oplus T_4[w_4] \oplus \\ \oplus T_5[w_5] \oplus T_6[w_6] \oplus T_7[w_7] \end{matrix},$$

де $T_0[a] = \begin{pmatrix} 01 \\ 04 \\ 07 \\ 06 \\ 08 \\ 01 \\ 05 \\ 01 \end{pmatrix} \cdot \pi_0[a]$, $T_1[a] = \begin{pmatrix} 01 \\ 01 \\ 04 \\ 07 \\ 06 \\ 08 \\ 01 \\ 05 \end{pmatrix} \cdot \pi_1[a]$,

$T_2[a] = \begin{pmatrix} 05 \\ 01 \\ 01 \\ 04 \\ 07 \\ 06 \\ 08 \\ 01 \end{pmatrix} \cdot \pi_2[a]$, $T_3[a] = \begin{pmatrix} 01 \\ 05 \\ 01 \\ 01 \\ 04 \\ 07 \\ 06 \\ 08 \end{pmatrix} \cdot \pi_3[a]$,

$T_4[a] = \begin{pmatrix} 08 \\ 01 \\ 05 \\ 01 \\ 01 \\ 04 \\ 07 \\ 06 \end{pmatrix} \cdot \pi_0[a]$, $T_5[a] = \begin{pmatrix} 06 \\ 08 \\ 01 \\ 05 \\ 01 \\ 01 \\ 04 \\ 07 \end{pmatrix} \cdot \pi_1[a]$,

$T_6[a] = \begin{pmatrix} 07 \\ 06 \\ 08 \\ 01 \\ 05 \\ 01 \\ 01 \\ 04 \end{pmatrix} \cdot \pi_2[a]$, $T_7[a] = \begin{pmatrix} 04 \\ 07 \\ 06 \\ 08 \\ 01 \\ 05 \\ 01 \\ 01 \end{pmatrix} \cdot \pi_3[a]$.

Застосування таблиць-констант $T_i[a]$, $i = 0,1,...,7$ дозволяє значно зменшити кількість операцій, зокрема, функція нелінійної підстановки обчислюється за сім операцій XOR над 64-бітовими рядками.

Множення на α в арифметиці поля $GF(2^{64})$ реалізується за допомогою таблиці передобчислень Mul_α з 256 рядків по 64 бітів в кожному.

Вхід: 64-бітовий рядок w, що представляє елемент поля $GF(2^{64})$.

Вихід: 64-бітовий рядок $w' = w \otimes \alpha$, що представляє елемент поля $GF(2^{64})$.

1. Обчислюється значення $w' = (w << 8) \oplus Mul_\alpha[w >> 56]$, де: $w << 8$ є результатом зсуву ліворуч (в бік старших розрядів) 64-бітового рядка w на 8 розрядів із заповненням молодших розрядів нульовими значеннями; $w >> 56$ є результатом зсуву праворуч (в бік молодших розрядів) 64-бітового рядка w на 56 розрядів із заповненням старших розрядів нульовими значеннями. Вісім молодших розрядів вектору $w >> 56$ інтерпретуються як елемент поля $GF(2^8)$ для індексації таблиці передобчислень Mul_α; Mul_α – таблиця-константа з 256 рядків по 64 бітів в кожному (таблиця передобчислень), $Mul_\alpha[c]$ – 64-бітне значення таблиці передобчислень у рядку з індексом c, де c представляє елемент поля $GF(2^8)$, $Mul_\alpha[c]$ представляє елемент поля $GF(2^{64})$.

2. Виводиться вихідне значення w'.

Множення на α^{-1} в арифметиці поля $GF(2^{64})$ реалізується за допомогою таблиці передобчислень з 256 рядків по 64 бітів в кожному.

Вхід: 64-бітовий рядок w, що представляє елемент поля $GF(2^{64})$.

Вихід: 64-бітовий рядок $w' = w \otimes \alpha^{-1}$, що представляє елемент поля $GF(2^{64})$.

1. Обчислюється значення $w' = (w >> 8) \oplus Mul_{\alpha^{-1}}[w \& \gamma]$, де: $w >> 8$ є результатом зсуву праворуч (в бік молодших розрядів) 64-бітового рядка w на 8 розрядів із заповненням старших розрядів нульовими значеннями; $w \& \gamma$ є результатом побітової кон'юнкції 64-бітового рядка w та 64-бітового рядка γ, який у шістнадцятковому поданні має вигляд $\gamma = 00000000000000FF$. Вісім молодших розрядів вектору $w \& \gamma$ інтерпретуються як елемент поля $GF(2^8)$ для індексації таблиці передобчислень $Mul_{\alpha^{-1}}$;

2. Виводиться вихідне значення w'. Таким чином, генератор СТРУМОК побудовано за SNOW-2.0-подібною схемою підсумовуючого генератора (генератор SNOW-2.0 визначено в ДСТУ ISO/IEC 18033-4:2015).

Він використовує 256-бітний вектор ініціалізації та 256-бітний або 512-бітний секретний ключ.

Генератор забезпечує високий та надвисокий рівні стійкості (див. табл. 3) та призначений для забезпечення конфіденційності інформації під час її оброблення із врахуванням можливого застосування квантового криптографічного аналізу.

Це є унікальним українським рішенням, яке забезпечує високу швидкість криптоперетворення (див. табл 4) та надзвичайно високі показники безпеки, навіть з урахуванням можливого застосування в пост-квантовому середовищі, в умовах ведення інформаційних та гібридних війн.

Таблиця 3

Встановлені рівні захисту

Генератор ключових потоків, встановлений рівень захисту	Довжина ключа	Обчислювальна складність для найкращої відомої атаки	Обчислювальна складність квантового криптографічного аналізу
СТРУМОК-256, високий	256	2^{256}	2^{128}
СТРУМОК-512, надвисокий	512	2^{512}	2^{256}

Таблиця 4

Швидкість генерації ключових потоків

	Intel Core i3-4005U 1.7 GHz, Windows 10 x64	Intel Core i5-7200U 2.5 GHz, Windows 10 x64	Intel Core i7-7700 3,6 GHz, Windows 10 x64
SNOW 2.0-128	4,2 Гбіт/с	8,0 Гбіт/с	10,6 Гбіт/с
SNOW 2.0-256	4,2 Гбіт/с	8,0 Гбіт/с	10,6 Гбіт/с
СТРУМОК-256	6,8 Гбіт/с	12,8 Гбіт/с	17,4 Гбіт/с
СТРУМОК-512	6,8 Гбіт/с	12,8 Гбіт/с	17,4 Гбіт/с

Основні результати з розробки, дослідження та впровадження в Україні національних стандартів асиметричного КЗІ для пост-квантового застосування.

Для побудови надійних та безпечних моделей, методів, протоколів та алгоритмів КЗІ, стійких як до класичних, так і до квантових технологій криптоаналізу, необхідно застосовувати нові криптографічні примітиви, стійкість яких базується на застосуванні односторонніх математичних функцій, складність обернення яких не зменшується навіть у разі використання відомих квантових алгоритмів вирішення складних теоретико-обчислювальних задач.

Національний стандарт ДСТУ 8961:2019 «Інформаційні технології. Криптографічний захист інформації. Алгоритми асиметричного шифрування та інкапсуляції ключів» установлює криптографічний алгоритм асиметричного шифрування та інкапсуляції ключів для забезпечення конфіден-

ційності, цілісності, доступності, неспростовності та криптоживучості (як додаткової послуги) інформації та ключів під час їх оброблення [16]. У стандарті описано алгоритм асиметричного шифрування та інкапсуляції ключів, який використовує перетвореннях у кільці та скінченому полі для асиметричного поблокового шифрування та інкапсуляції ключів з використанням асиметричних пар ключів – для зашифрування з використанням відкритого ключа отримувача та інкапсуляції з використання секретного ключа сеансу інкапсуляції відправника. З використанням відповідних відкритого та секретного ключів отримувача та відправника обчислюється та встановлюється секретний ключ для блокового чи потокового симетричного шифрування інформації, отже стандарт визначає асиметричний блоковий шифр та протокол інкапсуляції ключів, які застосовуються для подальшого вироблення та встановлення ключів блокового чи потокового симетричного шифрування (за класифікацією ДСТУ ISO/IEC 18033-2:2015).

В алгоритмі асиметричного шифрування використовується асиметрична пара ключів – відкритий ключ для зашифрування блоків інформації (даних) відправником, а особистий (секретний) ключ – для розшифрування зашифрованих блоків отримувачем.

В алгоритмі (протоколі) інкапсуляції ключів також використовується асиметрична пара ключів – особистий (секретний) ключ сеансу – для інкапсуляції ключа сеансу відправником, а відкритий ключ сеансу – для декапсуляції сеансового ключа отримувачем.

Отримувач на основі свого секретного (особистого) ключа розшифрування та декапсульованого ключа сеансу відправника, а відправник на основі відкритого ключа зашифрування отримувача та свого секретного ключа сеансу, виробляють спільну таємницю (спільний ключ). В подальшому спільний ключ може використовуватись для шифрування інформації (даних) в каналах зв'язку при обміні інформацією.

Інкапсуляція ключа представляє процес криптографічного перетворення ключа сеансу та інших даних з метою забезпечення їх конфіденційності, цілісності (справжності) та криптоживучості, а також узгодження ключа симетричного шифрування даних між відправником та отримувачем в подальшому. Декапсуляція представляє собою процес перевірки цілісності та справжності інкапсульованого ключа сеансу зв'язку та узгодження ключа захисту даних між отримувачем та відправником.

Шифрування та інкапсуляція ключів здійснюється на основі математичних перетворень в кільці поліномі над скінченим полем.

При розробленні стандарту враховані вимоги щодо забезпечення криптографічної стійкості проти спеціальних атак на основі витоку по технічним каналам, а також потенційних класичних та квантових атак, в тому числі у перехідний та постквантовий періоди. Стандарт розроблено з урахуванням досвіду створення та застосування стандартів ДСТУ ISO/IEC 18033-2:2015, ANSI X9.98-2010 та результатів, що представлені в [1-10]. Цей стандарт може використовуватись під час розробки систем, комплексів та засобів криптографічного захисту інформації при наданні користувачам послуг конфіденційності, цілісності, неспростовності, доступності та криптоживучості ключів, що узгоджуються відправником та отримувачем, в інформаційних, телекомунікаційних та інформаційно-телекомунікаційних системах, в тому числі для захисту від спеціальних атак, а також у перехідний та постквантовий періоди.

Режими роботи та функції криптографічного захисту. Стандарт, у залежності від рівня криптографічної стійкості проти класичних та квантових атак, яку необхідно забезпечити, може застосовуватись в трьох режимах роботи:

- режим Скеля – KEM 256/128 – 256 біт захисту від класичних атак та 128 біт захисту від квантових атак, а також захисту від спеціальних атак;
- режим Скеля – KEM 384/192 – 384 біт захисту від класичних атак та 192 біт захисту від квантових атак, а також захисту від спеціальних атак;
- режим Скеля – KEM 512/256 – 512 біт захи-

сту від класичних атак та 256 біт захисту від квантових атак, а також захисту від спеціальних атак.

Також в кожному із режимів роботи можуть використовуватись окремо такі криптографічні перетворення:

- незалежний алгоритм (функція) асиметричного шифрування;
- протокол інкапсуляції ключів (функція), що ґрунтується на використанні функції асиметричного шифру;
- механізм (функція) симетричного шифрування та автентифікації, що ґрунтується на функціях асиметричного шифрування та інкапсуляція ключів.

В кожному із режимів роботи за рахунок застосування криптографічних перетворень в кільцях поліномів та скінчених полях забезпечується надання послуг конфіденційності, цілісності, справжності, доступності та криптографічної живучості ключа сеансу зв'язку та його узгодження між відправником та отримувачем.

Прийнятий національний стандарт ДСТУ 8961:2019 «Інформаційні технології. Криптографічний захист інформації. Алгоритми асиметричного шифрування та інкапсуляції ключів» заплановано до впровадження протягом 2021 року із наданням чинності з 01 січня 2021 року, отже в цій роботі докладний опис відповідних алгоритмів не наводиться. Стандарт планується використовувати під час розробки комплексів, систем та засобів криптографічного захисту інформації в інформаційних, телекомунікаційних та інформаційно-телекомунікаційних системах, а також в разі модернізації наявних систем для заміни асиметричних режимів шифрування та інкапсуляції ключів згідно з ДСТУ ISO/IEC 18033-2:2015, в тому числі у постквантовий період, при використанні стандарту забезпечується захист від спеціальних атак сторонніми каналами, в тому числі криптографічна стійкість у постквантовий період.

ВИСНОВКИ

Протягом 2014-2020 р. в Україні розроблено, досліджено, організовано прийняття та впроваджено низку національних криптографічних стандартів (ДСТУ 7624:2014; ДСТУ 7564:2014; ДСТУ 8845:2019; ДСТУ 8961:2019), які відповідають найжорсткішим вимогам надійності та безпеки, встановленим NIST США до пост-квантових криптографічних алгоритмів, в тому числі до асиметричного шифрування та інкапсуляції ключів (ДСТУ 8961:2019).

Безпосередньо за участю авторів цієї роботи виконано більше 100 НДР та ДКР, в тому числі за державним оборонним замовленням, господарчими договорами та за держбюджетним замовленням. За отриманими результатами розроблено та впроваджено основні елементи загальнонаціональної системи КЗІ: програмно-технічні комплекси акредитованих центрів сертифікації ключів (АЦСК) Міністерства зборів та податків, Укрзалізниці, Державної митної служби, Державної автомобільної інспекції, МОН України, тощо, АЦСК центрального засвідчувального органу (ЦЗО), тощо. Авторами розроблено та впроваджено в дію на базі державного підприємства Інформресурс систему захисту інформації МОН України. Розроблено та функціонують АЦСК та системи КЗІ більше 10 комерційних банків (Укрсіббанк, Укрсоцбанк, Приватбанк та інші). Розроблено та введено в дію Інтегровану систему електронної ідентифікації ID.GOV.UA, підсистему КЗІ Єдиного порталу державних послуг Дія, тощо. За результатами досліджень підготовлено серію наукових та навчально-методичних робіт (підручників, навчальних посібників та монографій), які вже впроваджено в навчальний процес провідних ВНЗ України за спеціальністю 125 - кібербезпека.

ЛІТЕРАТУРА

[1] Post-Quantum Cybersecurity Resources. https://www.nsa.gov/what-we-do/cybersecurity/post-quantum-cybersecurity-resources/

[2] NISTIR 8105 Report on Post-Quantum Cryptography. https://nvlpubs.nist.gov/nistpubs/ir/2016/NIST.IR.8105.pdf

[3] Quantum Safe Cryptography and Security; An introduction, benefits, enablers and challenges. https://portal.etsi.org/Portals/0/TBpages/QSC/Docs/Quantum_Safe_Whitepaper_1_0_0.pdf

[4] Neal Koblitz and Alfred J. Menezes. A Riddle Wrapped in an Enigma. https://eprint.iacr.org/2015/1018.pdf
[5] Announcing Request for Nominations for Public-Key Post-Quantum Cryptographic Algorithms. https://csrc.nist.gov/News/2016/Public-Key-Post-Quantum-Cryptographic-Algorithms
[6] NISTIR 8309 Status Report on the Second Round of the NIST Post-Quantum Cryptography Standardization Process. https://nvlpubs.nist.gov/nistpubs/ir/2020/NIST.IR.8309.pdf
[7] 11th International Conference, PQCrypto 2020, Paris, France, April 15–17, 2020, Proceedings. https://link.springer.com/book/10.1007/978-3-030-44223-1
[8] Kris Gaj. Implementation and Benchmarking of Round 2 Candidates in the NIST Post-Quantum Cryptography Standardization Process Using FPGAs. https://csrc.nist.gov/CSRC/media/Projects/post-quantum-cryptography/documents/round-3/seminars/oct-2020-gaj-kris-presentation.pdf
[9] Round 3 Submissions. https://csrc.nist.gov/projects/post-quantum-cryptography/round-3-submissions
[10] Rotem Arnon-Friedman, Frédéric Dupuis, Omar Fawzi, Renato Renner and Thomas Vidick. Practical device-independent quantum cryptography via entropy accumulation. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5792631/
[11] Держспецзв'язку впроваджує нові стандарти криптографічного захисту інформації. https://www.kmu.gov.ua/news/247952015
[12] Національні стандарти на які є посилання у нормативно-правових актах. http://uas.org.ua/ua/services/standartizatsiya /109-2/
[13] Про прийняття національних стандартів України, гармонізованих з європейськими стандартами, міжнародних стандартів як національних стандартів України, затвердження національних стандартів України, скасування міждержавних стандартів в Україні та внесення зміни до наказу Державного комітету стандартизації, метрології та сертифікації України від 12.06.2002 № 357. https://zakon.rada.gov.ua/rada/show/v1431731-14#Text
[14] Про прийняття та скасування національних стандартів, прийняття змін до національних стандартів, скасування міждержавного стандарту. https://zakon.rada.gov.ua/rada/show/v0085774-19#Text
[15] Каталог НД України. http://csm.kiev.ua/nd/nd.php?z= %D0%94%D0%A1%D0%A2%D0%A3+8845%3A2019&st=0&b=1
[16] Про прийняття та скасування національних стандартів. https://zakon.rada.gov.ua/rada/show/v0465774-19#Text

СТАНДАРТИЗАЦИЯ СИСТЕМ, КОМПЛЕКСОВ И СРЕДСТВ КРИПТОГРАФИЧЕСКОЙ ЗАЩИТЫ ИНФОРМАЦИИ ДЛЯ ПРИМЕНЕНИЯ В ПОСТ-КВАНТОВОЙ СРЕДЕ.

Криптографическая защита информации (КЗИ) является важной составляющей информационной безопасности государства, непосредственно связанной с преодолением современных проблем и вызовов в киберне-тическом пространстве Украины, новых угроз информационной безопасности в критических инфраструктурах в оборонной и сфере безопасности, промышленности, банковском секторе, экономике и т. Особую опасность в этом смысле представляют новые риски, связанные с разработкой и стремительным внедрением современных и перспективных информационных технологий, способных в корне изменить архитектуру информационных систем, существующие парадигмы, стали принципы построения и математические основы современных средств криптографической защиты. В частности, появление и стремительное совершенствование новых вычислительных средств, основанных на принципах и эффектах квантовой физики (т.н. универсальных квантовых компьютеров) ставит под угрозу само существование действующих ныне и стандартизированных на национальном и международном уровнях механизмов (протоколов, алгоритмов и средств) асимметричной криптографии.

Ключевые слова: квантовые вычислители, крипографические алгоритмы, информационная безопасность, информационная безопасность, информационные системы.

STANDARDIZATION OF SYSTEMS, COMPLEXES AND MEANS OF CRYPTOGRAPHIC PROTECTION OF INFORMATION FOR APPLICATION IN POST-QUANTUM ENVIRONMENT

Cryptographic information protection (CRC) is an important component of information security of the state, directly related to overcoming modern problems and challenges in the cyberspace of Ukraine, new threats to information security in critical infrastructures in defense and security, industry, banking, economy, etc. Particularly dangerous in this sense are the new risks associated with the development and rapid implementation of modern and advanced information technologies that can radically change the architecture of information systems, existing paradigms, stable principles of construction and mathematical foundations of modern CCI tools. In particular, the emergence and rapid improvement of new computing tools based on the principles and effects of quantum physics (so-called universal quantum computers) threatens the very existence of existing and standardized at

the national and international levels mechanisms (protocols, algorithms and tools) asymmetric cryptography.
Keywords: quantum computers, cryptographic algorithms, information security, information systems.

КОРЧЕНКО Анна Олександрівна, професор кафедри безпеки інформаційних технологій факультету кібербезпеки, комп'ютерної та програмної інженерії Національного авіаційного університету, доктор технічних наук, доцент.
E-mail: annakor@ukr.net
ORCID: *0000-0003-0016-1966*
ІВАНЧЕНКО Євгенія Вікторівна, професор кафедри безпеки інформаційних технологій факультету кібербезпеки, комп'ютерної та програмної інженерії Національного авіаційного університету, кандидат технічних наук, доцент.
E-mail: evivancenko@gmail.com.
ORCID: *0000-0003-3017-5752*
КОШКІНА Наталія Василівна, старший науковий співробітник відділу оптимізації чисельних методів, Інститут кібернетики імені В.М. Глушкова НАН України, доктор технічних наук, старший науковий співробітник.
E-mail: nata.koshkina@gmail.com
ORCID: 0000-0001-5180-2255
КУЗНЕЦОВ Олександр Олександрович, професор кафедри безпеки інформаційних систем і технологій факультету комп'ютерних наук Харківського національного університету імені В. Н. Каразіна, заступник головного конструктора приватного акціонерного товариства «Інститут інформаційних технологій» доктор технічних наук, професор.
E-mail: kuznetsov@karazin.ua
ORCID: 0000-0003-2331-6326
КАЧКО Олена Григорівна, професор кафедри програмної інженерії факультету комп'ютерних наук Харківського національного університету радіоелектроніки, заступник головного конструктора приватного акціонерного товариства «Інститут інформаційних технологій» кандидат технічних наук, професор.
E-mail: ekachko@gmail.com
ORCID: 0000-0001-9249-0497
ПОТІЙ Олександр Володимирович, заступник Голови Державної служби спеціального зв'язку та захисту інформації України доктор технічних наук, професор.
E-mail: *potav@ua.fm*
ОНОПРІЄНКО Віктор Васильович, генеральний директор приватного акціонерного товариства «Інститут інформаційних технологій», кандидат технічних наук, старший науковий співробітник
E-mail: kuznetsov@karazin.ua
БОБУХ Всеволод Анатолійович, начальник відділу апаратних засобів захисту інформації приватного акціонерного товариства «Інститут інформаційних технологій», кандидат технічних нау
E-mail: *bobukhv@iit.kharkov.ua*
ORCID: 0000-0002-1175-5092