УДК 004.056.5
DOI: 10.31673/2409-7292.2025.041205

Гончаров М. О., Малахов С. В., Жілєнков Д. В.

ДОСЛІДЖЕННЯ СКЛАДНОСТІ ТА ВЛАСТИВОСТЕЙ РІЗНИХ СХЕМ РОЗГОРТКИ ВИХІДНИХ ДАНИХ СТЕГАНОКОНТЕНТА, В УМОВАХ ЗМІНИ СЦЕНАРІЇВ АТАК

В статті розглянуті наслідки моделювання атак та результати оцінки обчислювальної складності різних схем формування масиву серій опорних блоків зображень (тестового контенту) використовуваних, як механізм із протидії спробам нелегітимної екстракції даних при реалізації процедур гібридного стеганографічного алгоритму. Метою роботи є аналіз складності реалізації та дослідження властивостей різних схем формування (далі розгортки) масиву серій опорних блоків (ОБ) для різних умов їх обробки та сценаріїв проведення атак. Проведене моделювання дозволяє візуалізувати наслідки використання різних схем розгортки (тобто, принципу формування масивів серій ОБ) в умовах компрометації інших рівнів захисту. Дослідження різних розгорток показало, що впровадження концепції зміни використовуваної схеми розгортки, є ефективним й обчислювально простим інструментом для протидії спробам неавторизованого доступу до контенту. Отримані результати дозволяють стверджувати, що «вдала» атака (тобто підбір діючих параметрів обробки) інших рівнів захисту, не гарантує успішної зворотної компіляції вихідного стеганографічного контенту. Встановлено, що зигзагоподібні та двоетапні/кратні схеми розгортки, забезпечують вищу захищеність контенту при незначному зростанні обчислювальних витрат. Умовно «прості» розгортки, мають мінімальні часові затрати, але демонструють нижчу стійкість до спроб підбору поточних параметрів схем розгортки. Варіативність налаштувань різних схем розгортки, зумовлює комбінаторність відповідного елементу в структурі ключа екстрактора даних. Зроблено висновок, що суміщення принципу кодування довжин серій із практикою використання різних схем розгорток, дозволяє водночас вирішити два важливих завдання: - забезпечити додатковий рівень захисту та створити умови для зниження обчислювальної складності всього алгоритму на наступних етапах обробки.

Ключові слова: інформаційна безпека, обчислювальна складність, інкапсуляція даних, несанкціонований доступ, стеганографія, контент, кодування довжин серій, кібератака.

Вступ

Стрімке зростання обсягів інформації та неперервний розвиток інформаційно-комунікаційних технологій (ІКТ) формують нові умови функціонування сучасних систем обробки та передачі даних. Поряд із розширенням їх можливостей змінюється й спектр характерних загроз інформаційної безпеки (ІБ) [1-3], що обумовлює необхідність комплексних досліджень, спрямованих, як на створення нових технологій й інструментів захисту інформації, так і неперервного вдосконалення вже існуючих методів і способів протидії сучасним загрозам.

У цьому контексті, стеганографія є важливим напрямом забезпечення ІБ, що діє в парадигмі приховання інформації у цифрових носіях – переносниках даних різного типу, дозволяючи зберігати конфіденційність чутливих даних без привернення зайвої уваги до самого факту існування/обміну прихованої інформації [2-3]. В межах цієї парадигми, актуальним завданням є розробка методів обробки даних, що одночасно поєднують: високу скритність інформації і стійкість до спроб її неавторизованого вилучення (злому), мають високу процедурну продуктивність та забезпечують адаптивність до поточних умов функціонування використовуваних програмно-апаратних платформ. В цієї комбінації чинників, найважливішим аспектом є досягнення балансу між збереженням характерних статистичних властивостей контейнера – переносника даних [4] та мінімальним рівнем його демаскуючих спотворень в умовах впливу спорадичних ресурсних обмежень (наприклад, поточне завантаження CPU та/чи RAM, «вільний» залишковий заряд батареї гаджету, обмеження на смугу частот використовуваних каналів зв’язку тощо).

Постановка проблеми

На шляху розв’язання зазначених вище труднощів й протиріч, перспективним напрямом є розробка гібридних алгоритмів стегановставки, здатних забезпечувати багаторівневий захист даних при мінімізації залучених програмно-апаратних ресурсів [5-7]. Їх використання забезпечує контрольованість основних процесів обробки, а порушення діючих параметрів обробки, що утворюють відповідну структуру ключа екстрактора даних [5], унеможливлює

коректне відновлення прихованих даних (далі, контенту). Це підвищує стійкість даних до спроб їх несанкціонованого вилучення при збереженні вимог підтримки ресурсного консенсусу на виконання передбачених процедур.

Аналіз останніх досліджень і публікацій

В загальному випадку, оптимізація стеганографічних процедур, тісно пов’язана з особливостями обробки вихідних масивів даних. В межах даної роботи, такими даними є - блоки напівтонових зображень контейнера та контенту [4]. Як свідчить проведені дослідження [8-17], важливим чинником, що значною мірою визначає потрібний результат (в частині простоти й комплексності отримуваного ефекту), є впровадження різних схем розгортки (або сканування) масивів вихідних даних. Застосована схема розгортки визначає порядок доступу/обробки до окремих блоків зображень та/чи вибірки поточних параметрів з формованих масивів довжин серій [4]. Це безпосередньо впливає на обчислювальну складність операцій кодування та декодування, можливість апаратного прискорення, а також форму й візуальну виразність артефактів, що можуть виникати під час атак чи пошкоджень даних [9-11]. Тому вибір потрібної схеми розгортки є не лише суто технічним аспектом реалізації алгоритму, а й важливим чинником із забезпечення стійкості контенту (перш за все через варіативність можливих реалізацій) [8-9] та фактором адаптивності алгоритму до стохастичних умов обробки даних [4].

Метою роботи є: аналіз складності та дослідження властивостей різних схем розгортки (вибірки) діючих параметрів серій опорних блоків (ОБ) напівтонових тестових зображень, в різних умовах їх обробки та сценаріях проведення атак.

Для досягнення зазначеної мети, проведено комп'ютерну імітацію атаки на тестовий стеганоконтент за умови, що атакуючій стороні вдалося визначити поточні параметри обробки контенту, які реалізовані на двох основних рівнях захисту [5,12] дослідного стеганоалгоритму. Під час моделювання використані найбільш показові параметри налаштувань діючого алгоритму, що дозволяє продемонструвати характер і структуру артефактів атакованих зображень та оцінити вплив різних схем розгортки на якість відтворення контенту [8-15].

Отримані результати дозволяють уточнити особливості формування масивів серій ОБ напівтонових зображень, залежно від обраної схеми розгортки та оцінити параметри обчислювальної складності процедур формування масиву серій ОБ. Крім того, проведене моделювання показово демонструє характерні наслідки умовних атак, шляхом підбору діючих схем розгортки для різних типів тестових зображень.

Основна частина

Структура й особливості досліджуваних схем розгорток ОБ

Як вже було зазначено вище, розгортка визначає діючий порядок доступу/обходу /вилучення/обробки потрібних елементів оброблюваних масивів вихідних даних. В межах даної роботи такими елементами виступають:

А) окремі елементи й блоки вихідних зображень;

Б) параметри вже сформованих масивів довжин серій, тобто власне ОБ та значення його клонування/повторів (інакше, «довжина серії»).

Таким чином, реалізація тієї чи іншої схеми розгортки передбачає процес обходу та наступної обробки вказаних вище елементів вихідних даних за певним алгоритмом дій [8-9].В подальшому це дозволяє ефективно застосовувати потрібні перетворення даних, такі як дискретне косинусне перетворення, зсув, інкапсуляція даних тощо [1-4]. Серед іншого, такий підхід (тобто, кодування довжин серій) забезпечує перехід від двовимірної матриці пікселів до лінійного масиву, зберігаючи або з певним наміром [5,12] порушуючи наявні просторові кореляції (статистичні властивості) між сусідніми елементами оброблюваних масивів даних ((А) та (Б)). За принципом обходу/сканування масиву вихідних даних, розгортки слід умовно поділити на: - прості, чи лінійні (див. рис. 1(а-в, д)); - складні або нелінійні (рис. 1(ж, і-к)); - двопрохідні (рис.1(е, з)); - випадкові (рис.1(г)); - комбіновані, тобто ті що одночасно

© Гончаров М. О., Малахов С. В., Жілєнков Д. В. (2025) Дослідження складності та властивостей різних схем розгортки вихідних даних стеганоконтента, в умовах зміни сценаріїв атак. Сучасний захист інформації, 4(64), 44–58. https://doi.org/10.31673/2409-7292.2025.041205

поєднують декілька схем (наприклад, двопрохідний варіант в межах якого на кожному скані (циклу руху), реалізується власна схема вибірки заданих параметрів оброблюваного масиву).

Зразки тестових зображень та досліджувані схеми розгорток, що були використані в ході проведеного циклу імітаційного моделювання, представлені на рис. 1 (де скорочення «CG» – комп'ютерна графіка, тобто штучні зображення).

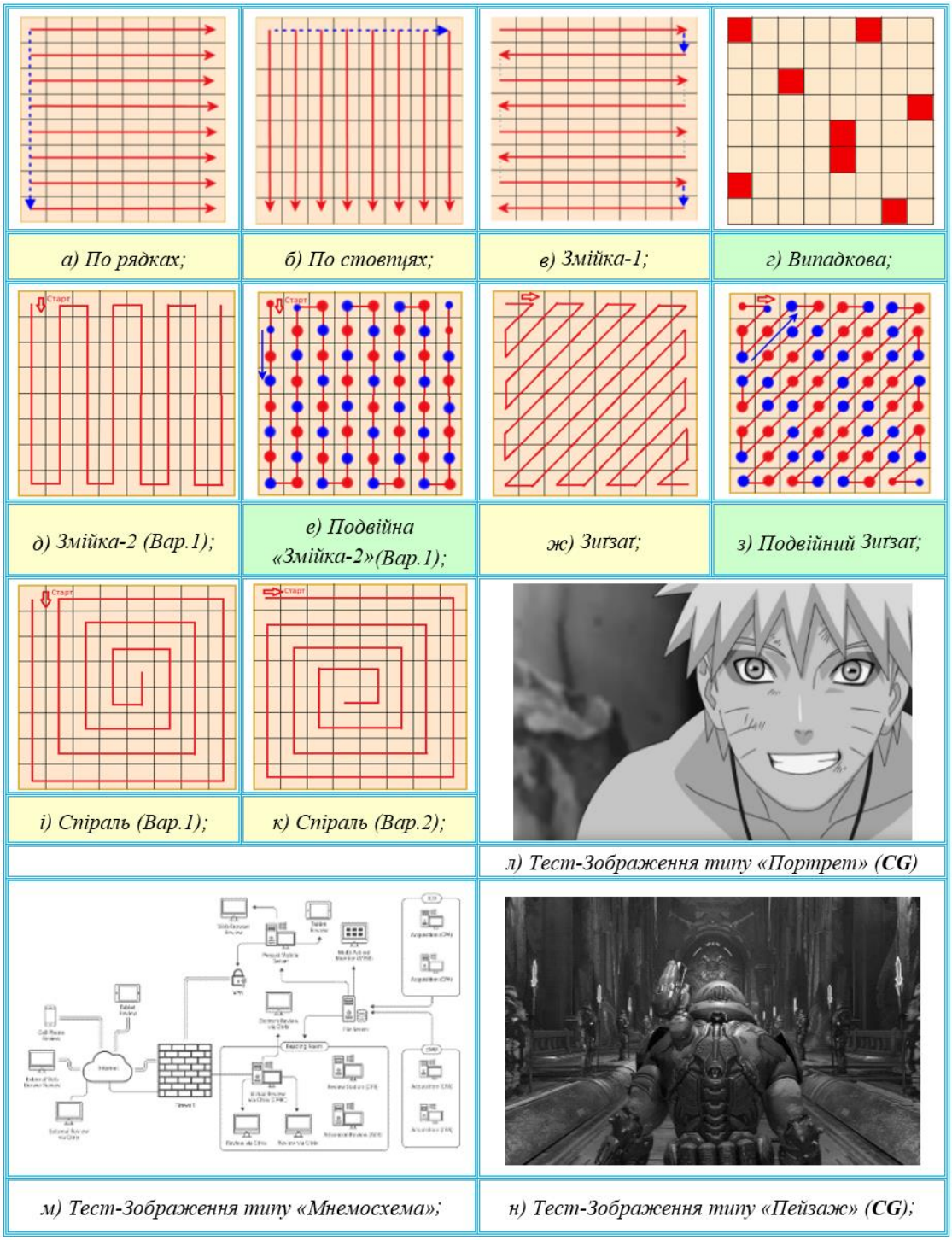


Рис. 1. Досліджувані розгортки *(а-к)* та зразки тестових зображень *(л-м)* різного типу

Отримані результати підтверджують загальну реалізованість розглянутої концепції стеганоалгоритму [5] та ефективність відповідних схем розгортки, як інструменту з протидії спробам неавторизованого вилучення контенту, у різних сценаріях їх використання [8-11,13].

Це, у свою чергу, відкриває перспективи для подальшої інтеграції розроблених рішень у реальні алгоритми стеганографічного захисту інформації.

Схема розгортки блоків «по рядках» (рис.1(а)) передбачає послідовне розбиття масиву вихідного зображення на блоки заданого розміру N×M у напрямку рядків [8,11-12]. Іншими словами, зчитування і групування блоків відбувається лінійно, спочатку по елементах одного рядка даних, після чого алгоритм покроково переходить до наступного тощо.

Такий підхід характеризується високою ефективністю виконання та мінімальними витратами на обробку. Алгоритм простий у реалізації і не потребує додаткових умов чи складних перевірок та зайвого кешування виконуваних процедур. Лінійний зсув зумовлює передбачувану швидкодію навіть для великих обсягів вихідних даних, а структура отримуваного масиву довжин серій ОБ, зручна для подальшої обробки й аналізу. Цю і наступну (по стовбцях) схеми розгорток можна розглядати, як умовний «еталон» для порівняння з іншими, більш складними схемами.

Схема розгортки блоків «по стовпцях» (рис.1(б)) передбачає послідовне розбиття зображення на блоки заданого розміру N×M у напряму стовпців. Тобто групування елементів вибірки здійснюється не за рядками, як у попередньому випадку, а за вертикальним принципом, з покроковим зміщенням до наступного стовпця. Формування масиву серій за такою схемою розгортки відзначається прямолінійною логікою та високою ефективністю виконання. Вона проста в реалізації, а варіативність її реалізацій передбачає можливість умовного «старту» з будь-якого стовпця вихідного масиву даних, в одному з 2-х напрямків (вниз чи вгору).

Схема «Змійка-1» (рис.1(в)) передбачає послідовне сканування/обхід вихідного масиву за принципом «змійки». Варіативність її реалізацій передбачає можливість умовного «старту» з будь-якої строки вихідного масиву, в будь-якому з 2-х напрямків Такий підхід забезпечує безперервний обхід з мінімальною кількістю переходів і додаткових перевірок. Алгоритм простий та добре враховує особливості локальної статистики вихідного масиву, але є більш складний для атакуючого, з точки зору підбору діючих параметрів розгортки. Процедурна простота гарантує стабільну швидкодію навіть на великих зображеннях.

Схема «Змійка-2» (рис.1(д)) впроваджує аналогічну для «Змійка-1» концепцію дій, але тільки відносно стовпців [8-11]. Варіативність її реалізацій передбачає можливість умовного «старту» з будь-якого стовбцю вихідного масиву, в одному з двох напрямків (вниз чи вгору) з цикловим поверненням до умовної точки початку скану. Так, наприклад, в ході проведеного моделювання було реалізовано два варіанти зсуву умовної точки «Старту»: - початок з лівого верхнього кута (див. рис.1(д)) та початок із лівого нижнього кута. В обох випадках отримувані результати практично однакові (різниця обумовлена лише статистичними особливостями структури зображень, що обробляються). Алгоритм простий у реалізації та не перевантажує обчислення, що дозволяє оперативно працювати з великими блоками.

Ця схема добре підходить для обробки зображень де превалюють вертикально орієнтовані структури (перш за все з точки зору обліку статистичних особливостей вихідних даних та порядку їх вибірки).В цілому, усі «прості» схеми розгортки (Рис. 1(а-в, д)) мають лінійну складність O(N×M) на блок (де, O це асимптотична оцінка складності алгоритму, що описує верхню межу кількості операцій), не потребують генерації координат відливків та легко інтегруються в подальшу концепцію обробки [5], роблячи їх вдалим варіантом для подальших порівнянь з більш складними релізами розгорток.

Складні або нелінійні варіанти розгорток (рис. 1(ж, і-к)) дозволяють підвищити рівень стійкість до атак, так як пропонують складніший механізм обробки вихідних масивів, зберігаючи при цьому принцип послідовної вибірки статистично пов'язаних «сусідніх» блоків. В даному випадку важливо, що атакуюча сторона не має інформації стосовно параметрів вихідного масиву даних, орієнтуючись тільки по вмісту й структурі, невідомим образом дефрагментованих даних масиву довжин серій.

Схема «Зиґзаґ» (рис.1(ж)) аналогічна послідовності Z – упорядкування коефіцієнтів перетворення в стандарті стиску зображень JPEG, та за своєю суттю впроваджує послідовне й рівномірне охоплення всіх сусідніх блоків, враховуючи існуючі кореляційні взаємозв'язки між ними. Алгоритм детермінований, де кожен елемент обробляється лише один раз, що робить обчислювальні витрати помірними. Контроль зміни напрямків руху додає невелику складність, однак це дозволяє успішно протидіяти спробам нелегітимної екстракції контенту. Ця схема корисна для випадків обробки реалістичних зображень із рівномірною структурною складністю відображуваних сцен (наприклад, зображення типу «аерофотозйомка» чи пейзаж). За своєю суттю, є універсальним варіантом обробки, що в рівній мірі враховує всі напрямки градієнтів для наявних деталей зображення (горизонтальні, вертикальні та похилі). Варіативність реплік «Зиґзаґ», передбачає можливість умовного «старту» з будь-якого куту вихідного зображення та вибір напрямку руху на його 1-му кроці (тобто, по рядку або по стовпцю). Так, на рис.1(ж) представлений варіант зі «стартом» руху в лівому верхньому кутку і першим кроком по рядку.

Схема «Спіраль» (рис.1(і-к)) є гібридом двох лінійних схем сканування: - по рядках і стовпцям (рис.1(а-б)). За своєю сутністю є більш складною їх реалізацією, істотно розширюючи доступну комбінаторику реалізованих варіантів [10-11]. Варіативність доступних реплік «Спіралі» базується на виборі точки «старту» (периферія або центр) та початковому напрямку руху (тобто, першому кроці здійснюваного скану). Т.ч., забезпечується 12 різних комбінацій. Такий принцип розгортки враховує природну локальну структуру вихідного зображення та посилює внутрішні можливості алгоритму до протистояння спробам неавторизованого вивчення контенту, зберігаючи при цьому малу обчислювальну складність. Незалежно від варіанта, спіральна схема забезпечує безперервне покриття всього масиву вихідних даних (зображення) без повернення до початкових координат. Алгоритм складніший у реалізації через контроль меж і напрямків, але залишається структурованим. Складність залишається близькою до лінійної, хоча включає додаткові перевірки для зміни напрямку. Схема безумовно корисна для цілей стеганографії, оскільки дозволяє уникнути регулярних повторів у структурі зображень при формуванні масиву серій ОБ. Всі варіанти «Спіралі» додають контроль меж та напрямків (O(1) на крок), зберігають лінійну складність O(N×M) на блок та ускладнюють реалізацію порівняно з лінійними розгортками. Однак, при цьому, «Спіраль» забезпечує кращу фрагментацію вихідних даних при мінімальному зростанні обчислювальних витрат.

Схема «Подвійний Зиґзаґ» (рис.1(з)) є подальшим розширенням класичної зиґзаґу, що має на меті посилення можливостей, стосовно парирування спроб підбору діючих параметрів розгортки. Це здійснюється за рахунок додаткового розділення вихідного зображення на дві умовні підмножини - парні та непарні. Елементи кожної підмножини обробляються в двох незалежних циклах сканування за однією і тією ж схемою вибірки – «зиґзаґ» [8-9,11]. Т.ч., подвійний зиґзаґ дозволяє розділити елементи вихідного масиву на два масиву парних й непарних елементів та обробляти їх незалежно один від одного, що дозволяє зменшити взаємну кореляцію між сусідніми елементами одержуваного (результуючого) масиву довжин серій ОБ. Алгоритм потребує використання «внутрішніх» етапів класифікації та об’єднання результатів, але зберігає загальну циклічність дій.

Обчислювальна складність є помірною (розглянуто нижче). Варіативність модифікацій «Подвійний Зиґзаґ», подвоює кратність реплік, які характерні для випадку схеми «Зиґзаґ» (за умови збереження однакових схем вибірки у кожній підмножині, парних та непарних елементів). Так, на рис.1(з) представлений варіант зі «стартом» в лівому верхньому куті та з однаковою схемою руху для обох підмножин, тобто «зиґзаґ з першим кроком по рядку». Схема є універсальною, щодо типу оброблюваного контенту (тобто, ймовірності перепаду яскравості сусідніх елементів зображень) [5] та ефективною, з точки зору протидії спробам нелегітимного вилучення контенту.

Як й у попередньому випадку, схема «Подвійна змійка-2» (рис.1(е)) є подальшим розширенням розгортки типу «Змійка-2», що посилена додатковими можливостями із протидії спробам підбору діючих параметрів розгортки при збереженні компромісних значень обчислювальної складності. Ці можливості зумовлені розширенням комбінаторики можливих налаштувань: - точка старту (початку руху), напрям руху, кратність скану (двопрохідний цикл). Всі ці чинники важко коректно ідентифікувати та виділити із структури отримуваного масиву серій ОБ. Тому вочевидь, що ця схема у порівнянні з варіантами «Змійка 1» й «Змійка 2» (рис.1 (в,д)), є більш складною для атакуючого, перш за все з точки зору підбору діючих параметрів розгортки. При цьому відносна процедурна «простота» гарантує стабільну швидкодію навіть на великих зображеннях. В цілому, формування результуючого масиву серій, виконується аналогічно до концепції схеми «Подвійний зигзаг». Тобто здійснюється поділ вихідного зображення на дві умовних підмножини (одновимірні масиви), що містять дані з урахуванням специфічного порядку руху - подвійного змійкоподібного сканування. Зрозуміло, що при тенденції до збільшення розмірності блоків, використання концепції «кратності сканів» (рис.1 (е,з)), зі зрозумілої причини, призводить до ослаблення кореляційних зв'язків між блоками, котрі потрапляють у різні вибірки (підмножини), що відбиватиметься на зменшенні середньої довжини серій ОБ, які формуються (що тотожно збільшенню загальної кількості серій).

В цілому алгоритм «Подвійна змійка-2» має помірну лінійну складність та є зручний для реалізації. Додаткові етапи класифікації (підтримуваних релізів цієї схеми) та об’єднання, збільшують обчислювальні витрати вдвічі порівняно з базовими схемами, загальна складність алгоритму залишається лінійною O(W×H) (де, W - ширина матриці зображення (пікселів), H - висота матриці зображення (пікселів)). Інакше кажучи, всі репліки подвійної змійки, зберігають лінійну складність, але додають етапи класифікації та об’єднання, забезпечуючи більшу гнучкість при мінімальному зростанні обчислювальних витрат [8].

Схема розгортки типу «Random» (рис.1(г)) передбачає реалізацію механізму випадкової вибірки елементів вихідного масиву даних. Вочевидь, що на відміну від детермінованих схем («по рядках», «по стовпцях», «змійка-1» тощо), порядок руху сканування та умовна точка «старту», визначаються за допомогою генератора випадкових чисел. В даному разі відновлення початкового масиву здійснюється за нумерацією індексів блоків, що зберігає відомості про вихідне розташування блоків, тому така схема носить суто лабораторний (дослідницькій) характер. Генерація та зберігання координат додає невелике, більш-менш рівномірне навантаження O(1), але основна процедурна частина залишається відносно простою, на рівні O(W×H). При цьому гнучкість реалізації дозволяє комбінувати цю схему з іншими для синтезу умовно псевдовипадкових варіантів. В цілому, ця схема забезпечує максимальне ігнорування початкових кореляційних зв'язків між сусідніми елементами вихідних зображень. Як наслідок, для більшості зображень (за виключенням зображень типу мнемосхема (рис.1(м)) випадкова розгортка зумовлює відповідну надмірність у частині кількості формованих серій опорних блоків [8]. З одного боку, це добре: - для розширення комбінаторики мультиплексування (перемішування) поточних параметрів довжин серій. А з іншого боку, різко зростає обчислювальне навантаження, перш за все, через збільшення кількості ОБ, що вимагають реалізації кодування з перетворенням (у даному випадку, DCT) [4-5]. Таким чином, цій тип розгортки слід розглядати виключно в контексті завдань взаємного порівняння різних схем вибірки, та як інструмент для забезпечення максимальної дефрагментації й руйнування існуючих кореляційних зв'язків у масиві вихідних даних.

Усі розглянуті схеми формують блоки розміром N×M, які зберігаються у вигляді лінійного масиву 1×X, де X - загальна кількість блоків на які поділяється вихідне зображення для реалізації подальших етапів стеганографічого алгоритму [5-6]. Обчислювальна складність кожної схеми, визначається кількістю елементарних операцій у межах кожного блоку та загальною кількістю блоків у всьому вихідному масиві. Якщо зображення має розмір

© Гончаров М. О., Малахов С. В., Жілєнков Д. В. (2025) Дослідження складності та властивостей різних схем розгортки вихідних даних стеганоконтента, в умовах зміни сценаріїв атак. Сучасний захист інформації, 4(64), 44–58. https://doi.org/10.31673/2409-7292.2025.041205

«Width×Height», а блок має розмірність N×M, то загальна кількість блоків, на які розбивається зображення, визначається як, K = (Width/N)×(Height/M), де «K» це кількість блоків по горизонталі (Width/N) та по вертикалі (Height/M) вихідного масиву. Формування одного блоку потребує O(N×M) операцій зчитування чи копіювання. Сумарна кількість операцій для вихідного масиву дорівнює: - O(K×N×M) = O(Width×Height). Зменшення розміру блоків призводить до збільшення їх загальної кількості «K», але водночас зменшується кількість елементів, що обробляються у кожному з них (що суттєво для етапу кодування з перетворенням [4-6]). В цілому, обидва процеси практично компенсують один одного, тому підсумкова складність формування залишається лінійною - O(W×H). Незалежно від обраної схеми розгортки вихідного зображення, алгоритмічна складність процесу формування масиву серій ОБ [5,8,15] залишається лінійною відносно загальної кількості елементів вихідного зображення. Т.ч., обчислювальні витрати алгоритму зростають лінійно зі збільшенням розміру зображення, а просторова складність визначається обсягом пам'яті, необхідним для зберігання одного блока, і становить O(N×M). Усі схеми розгортки мають лінійну часову складність O(W×H), оскільки кожен піксель обробляється рівно один раз незалежно від типу траєкторії сканування. Це означає, що збільшення ширини чи висоти зображення призводить до пропорційного збільшення обчислювальних витрат. Просторова складність залишається сталою відносно всього зображення і становить O(N×M), оскільки в пам'яті одночасно утримується лише один блок розміру N×M. Отже, навіть при роботі з великими зображеннями, обсяг необхідної пам'яті визначається лише параметрами блока, а не розмірами вхідних даних.

Додаткові процедури, котрі характерні для окремих схем розгортки, такі як зміна напрямку руху (змійка, зигзаґ, спіраль), додають лише сталу кількість операцій O(1) та не впливають на асимптотичний порядок складності. Для схем із кратним скануванням, де виконується розділення вихідного масиву на окремі підмножин вибірки (рис.1(е,з)), обчислювальні витрати збільшуються лише вдвічі порівняно з простими схемами (рис.1(а-в, д)), що все одно залишається в межах лінійної складності O(W×H). Додаткові етапи класифікації та об'єднання результатів додають практично незмінне навантаження та не змінюють асимптотичного характеру складності алгоритму. Т.ч., усі розглянуті процедури характеризуються лінійною часовою складністю O(W×H) та сталою просторовою складністю O(N×M). Вони мають детермінований характер «поведінки» при масштабуванні, ефективно використовують ресурси пам'яті й забезпечують стабільну швидкодію, що робить їх придатними для обробки зображень у режимі реального часу, та застосування в системах, які функціонують в умовах спорадичних обмежень наявних програмно-апаратних ресурсів.

Оцінка значень PSNR та часових характеристик різних схем розгортки

Розглянемо результати імітаційного моделювання, що представлені на рис.1. В даному разі, результати оцінки обчислювальної складності та візуалізація реалізованих схем розгорток, розглядалися на прикладі 3-х зразків тестових зображень після виконання процедур попередньої обробки, а саме 2-го варіанту передобробки [16-17], як найбільш збалансованого з точки зору наслідків процедури згладжування.

В межах моделювання досліджувалися якість відтворення тестових зображень [2-3], яка оцінювалась за показником PSNR (пікового значення С/Ш), та обчислювальна складність (за часом виконання) для різних схем розгортки без врахування етапу передобробки [5-6], для блоків розмірністю 8×8 елементів. Узагальнення отриманих даних підтверджує ефективність запропонованих схем у різних сценаріях їх налаштувань і атак [8-15]. На рис. 2-4 представлені результати моделювання процедур атаки обраних зразків тестових зображень в умовах реалізації хибної схеми розгортки (рис. 2-4 (а-к)).

Наведені зразки наочно демонструють характер спотворень вихідних зображень, отриманих за результатами підбору (атаки) діючих параметрів формування масиву серій ОБ. В даному випадку для атаки застосовувалася «хибна» розгортка «По рядках». Також, на рис.2-4, (зразки «м»), представлена візуалізація відмінностей між вихідним та попередньо

згладженим (обробленим) зображеннями, у вигляді штучно синтезованої «піксельної різниці» (за абсолютною величиною). Ці зразки відображають умовну «мапу різниць», де білій колір (яскравість - 255) відповідає максимальний помилці (тобто max різницю пікселів), а чорний тон (яскравість - 0) позначає min помилку, тобто повну збіжність. Візуалізація наслідків атаки (зразки (а-к)), дозволяє вербально оцінити структуру і характер спотворень тестових зразків, отриманих за результатами Етапу їх передобробки [5]: - 2-й вар. згладжування; - поріг загрублення $P_{Z1} = 3$; - розмірність матриць згладжування (N) 3×3 елемента.

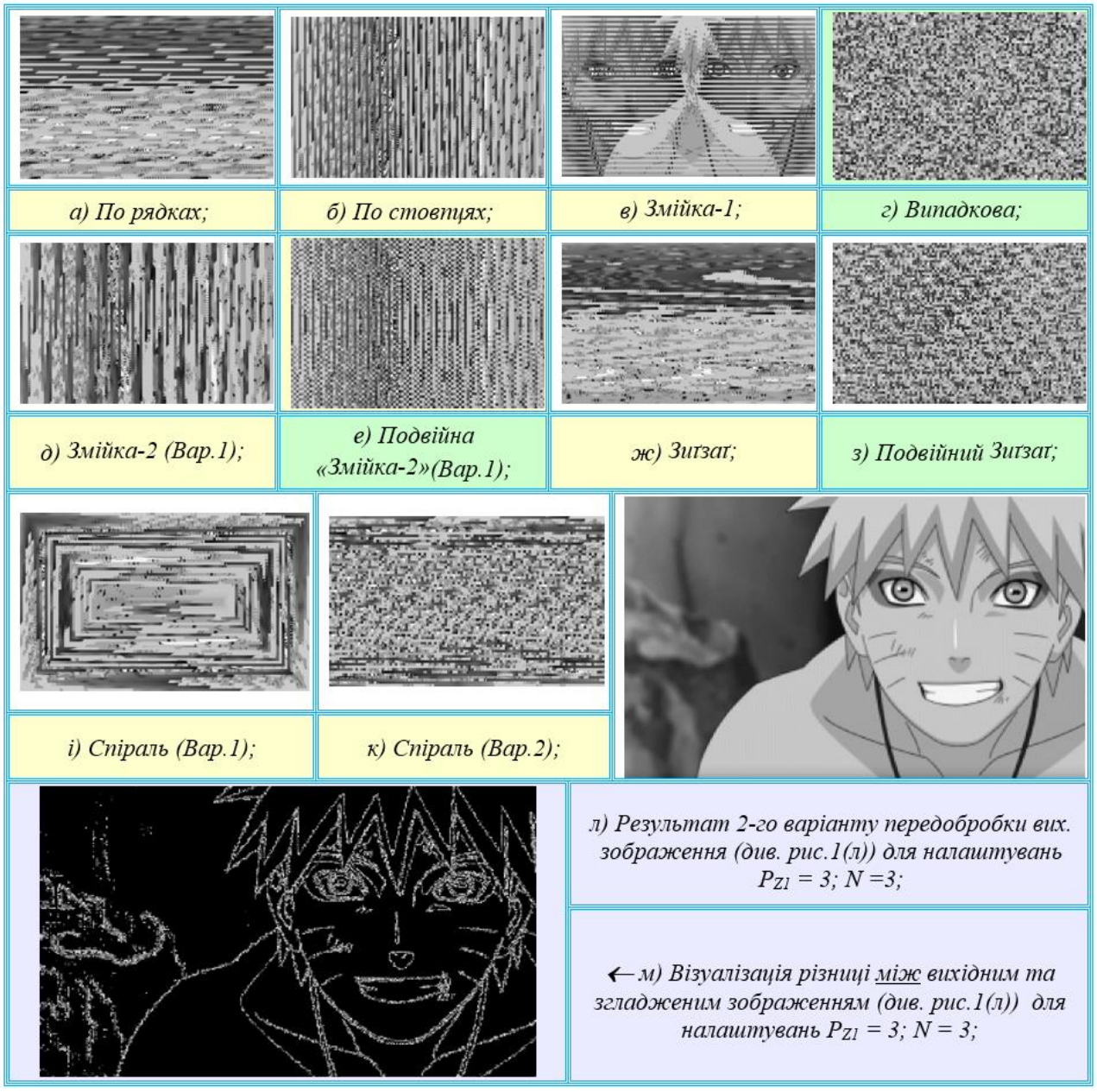


Рис. 2. Візуалізація атаки різних схем розгортки масиву серій ОБ (*а-к*) та результати згладжування вихідного контенту *(л-м*) для зображення *«Портрет» (CG)»*

Гістограми на рис.5(а) демонструють залежність значення PSNR від застосованих схем розгорток для обраних тестових зображень [4]. Ці гістограми характеризують кількісну міру відмінності між вихідним зображенням і тим, що було відновлено при застосуванні відповідних схем розгортки (рис.1(а-к)). Вищі значення PSNR свідчать про менші розбіжності між оригінальним і відновленим зображеннями. З прикладу зразків (в) на рис.2-4 слід, що

незважаючи на суттєві спотворення у випадку деяких схем атак (в даному разі, близької за принципом руху, хибної схеми «По рядках» при діючої «Змійка-1»), можна розпізнати основні риси оригінального зображення. Зрозуміло, що це є неприпустимим, однак при цьому важливо розуміти що такий результат отримано без врахування наслідків реалізації інших - передбачених процедур [5, 9, 12-13] для протидії спробам неавторизованого вилучення контенту (атакам). Така парадигма моделювання обрана навмисно для демонстрації впливу й наслідків реалізації тільки одного - першого рівню захисту [5, 12].

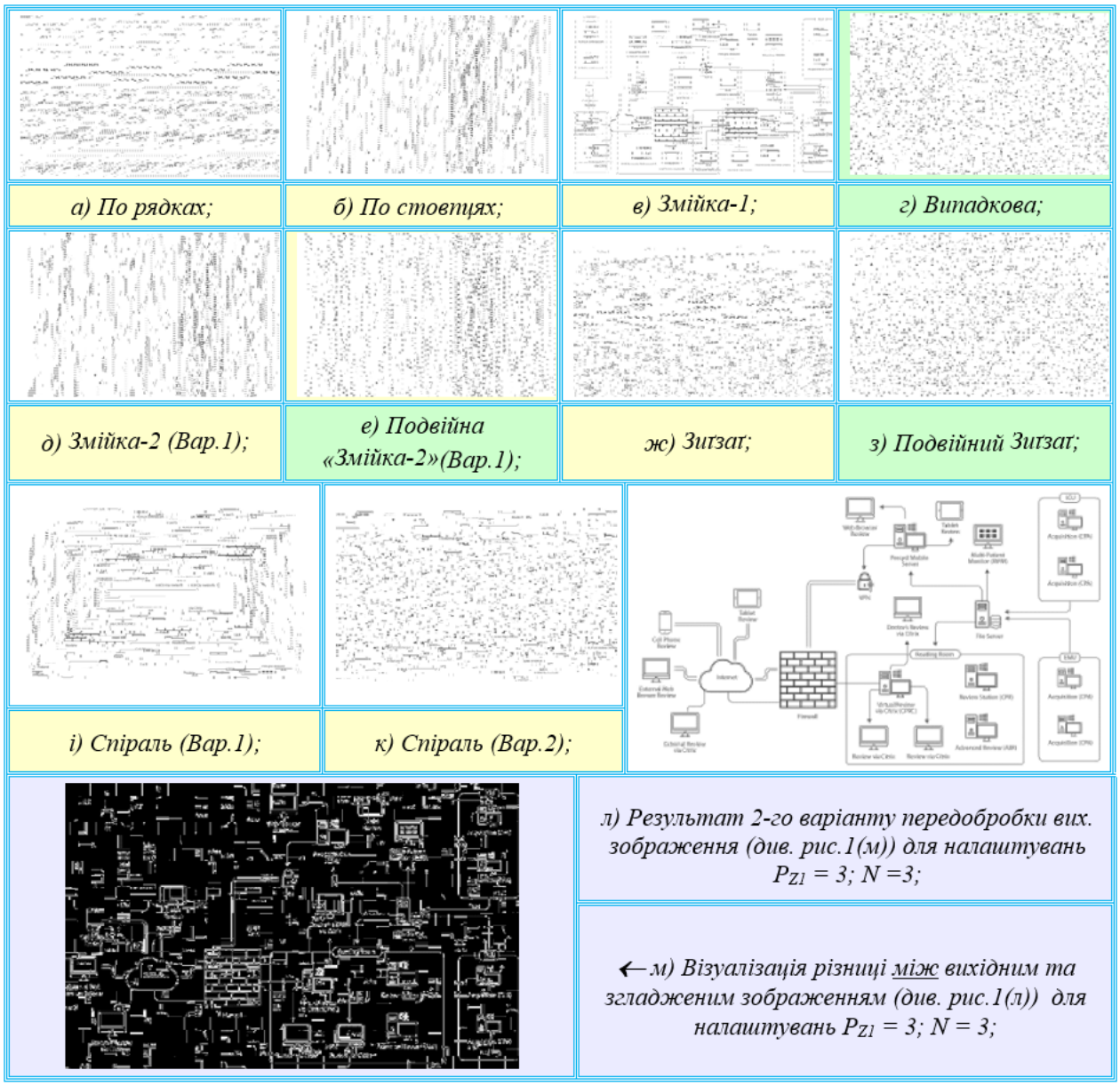


Рис. 3. Візуалізація атаки різних схем розгортки масиву серій ОБ (*а-к*) та результати згладжування вихідного контенту (л-м) для зображення типу *«Мнемосхема»*

Т.ч., реалізований сценарій експерименту виключає вплив інших передбачених шарів захисту, фокусуючи увагу на особливостях та наслідках використання механізму «розгорток». Важливо підкреслити, що суміщення принципу кодування довжин серій [4] із практикою використання різних схем розгорток, одночасно дозволяє вирішити два важливих завдання: 1 – забезпечити один з рівнів захисту [8, 12]; 2 – створити умови для зниження обчислювальної складності всього алгоритму на наступних етапах обробки (особливо на етапі

© Гончаров М. О., Малахов С. В., Жілєнков Д. В. (2025) Дослідження складності та властивостей різних схем розгортки вихідних даних стеганоконтента, в умовах зміни сценаріїв атак. Сучасний захист інформації, 4(64), 44–58. https://doi.org/10.31673/2409-7292.2025.041205

DCT) [11, 14-15]. Враховуючи загальну операційну простоту й невибагливість до вільних ресурсів, поєднання зазначених вище підходів надає можливість синтезу малоресурсного процедурного ланцюга, що забезпечує багатошарову структуру захисту даних [5,7]. Інакше кажучи, етап розгортки є лише першим кроком, ефект від використання якого тільки посилюється, по мірі реалізації інших процедур (наприклад, зміни просторової орієнтації ОБ, одержуваного масиву довжин серій [9]).

Коментуючи результати, що відображає зразок (в) на рис.2-4, слід підкреслити, що така ситуація характерна для випадку, коли атакуюча сторона застосувала схему розгортки, яка при всіх інших припущеннях (розмір блоків; точка «старту» руху; кратність скану тощо), частково враховує локальні кореляційні зв'язки формованих ОБ, що спричиняє відповідне візуальне спотворення (двоїння із зсувом) вилученого вихідного зображення. Як слід із інших зразків при застосуванні інших схем розгортки, ефект може бути суттєво більш руйнівний (див. зразки (а-б, г-к) на рис. 2-4).

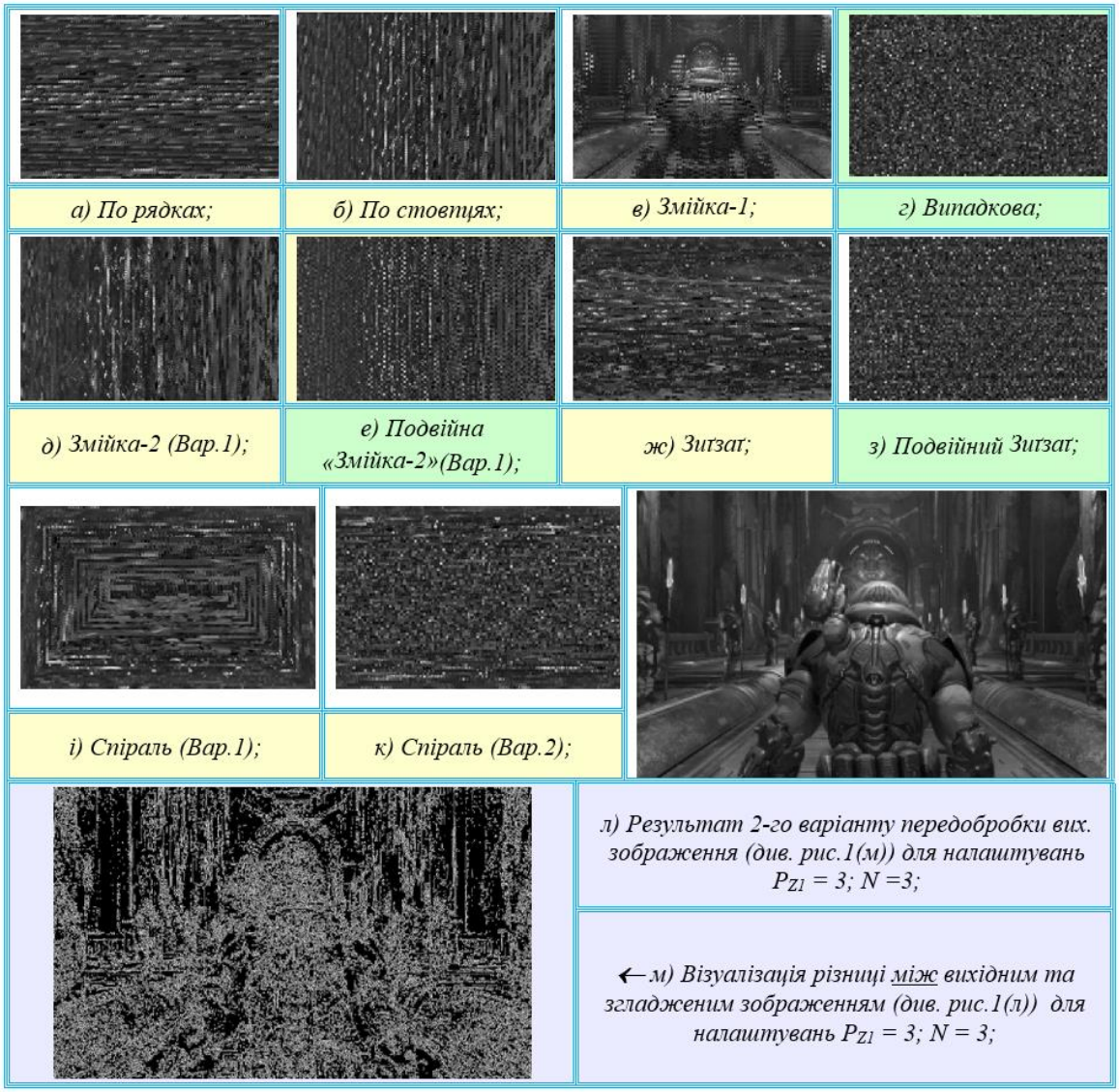
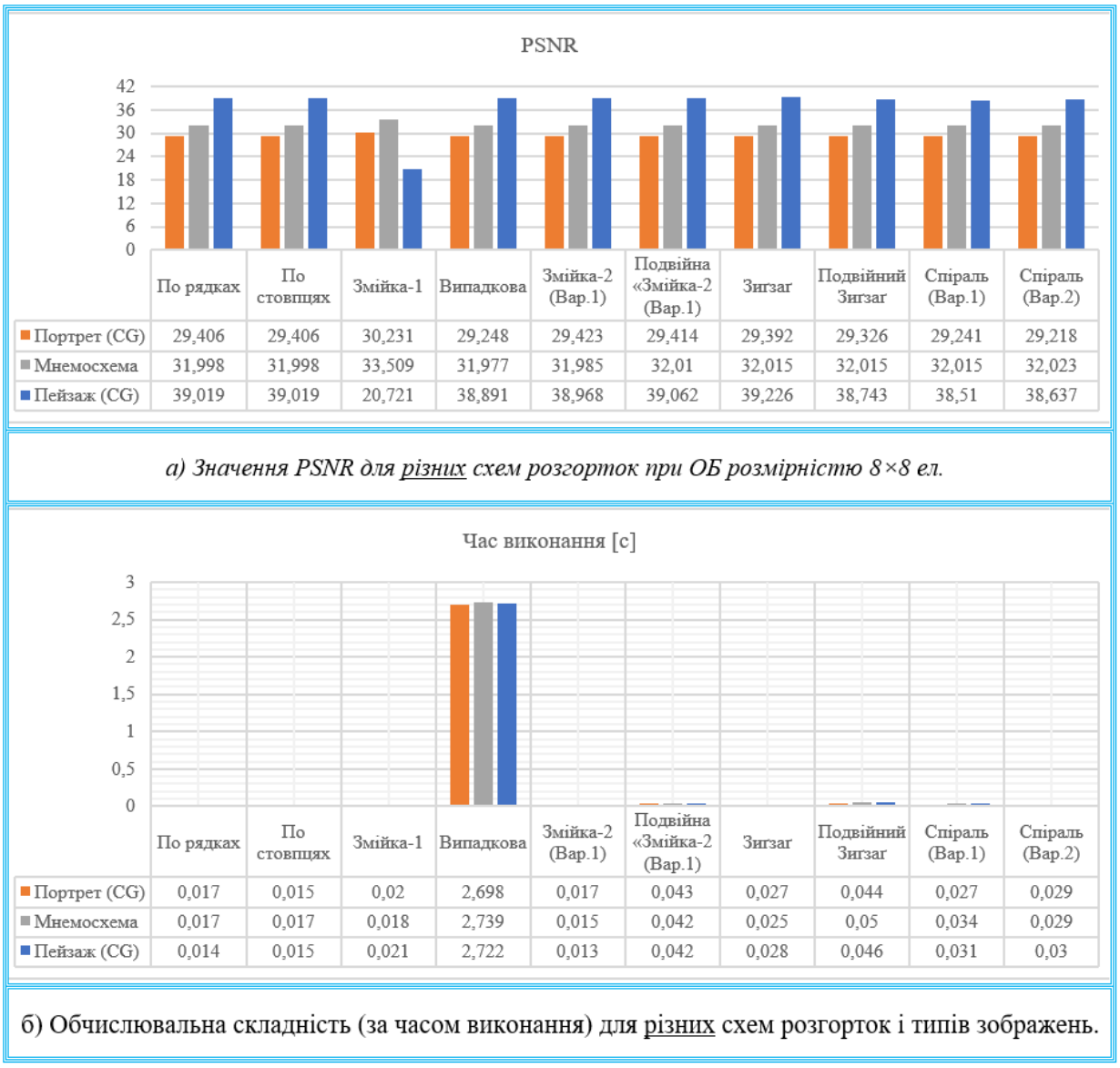


Рис. 4. Візуалізація атаки різних схем розгортки масиву серій ОБ (*а-к*) та результати згладжування вихідного контенту (л-м) для зображення типу *«Пейзаж» (CG)»*

Більш низькі значення PSNR на рис. 5(а), характеризують більшій ступінь спотворень під час спроб нелегітимного вилучення даних для різних схем розгортки, за умови збігу використаної розмірності блоків (що також є варіативної складовою процесу атаки).

Часові характеристики для різних схем формування блоків наведено на рис. 5(б). Представлено середній час виконання процедур формування ОБ зображень для матриць 8×8 ел., що дозволяє опосередковано оцінити складність виконання представлених схем і зіставити їх ефективність для різних типів контенту. Час реалізації «простих» схем розгортки суттєво менший, ніж для більш складних, проте останні значно ускладнюють (зразки (а-б, г-к) на рис.2-4)) для атакуючої сторони локалізацію вектора потенційних пошуків, щодо застосованої послідовності обробки вихідних даних [8-9].



	По рядках	По стовпцях	Змійка-1	Випадкова	Змійка-2 (Вар.1)	Подвійна «Змійка-2 (Вар.1)	Зигзаг	Подвійний Зигзаг	Спіраль (Вар.1)	Спіраль (Вар.2)
Портрет (CG)	29,406	29,406	30,231	29,248	29,423	29,414	29,392	29,326	29,241	29,218
Мнемосхема	31,998	31,998	33,509	31,977	31,985	32,01	32,015	32,015	32,015	32,023
Пейзаж (CG)	39,019	39,019	20,721	38,891	38,968	39,062	39,226	38,743	38,51	38,637

а) Значення PSNR для різних схем розгорток при ОБ розмірністю 8×8 ел.

	По рядках	По стовпцях	Змійка-1	Випадкова	Змійка-2 (Вар.1)	Подвійна «Змійка-2 (Вар.1)	Зигзаг	Подвійний Зигзаг	Спіраль (Вар.1)	Спіраль (Вар.2)
Портрет (CG)	0,017	0,015	0,02	2,698	0,017	0,043	0,027	0,044	0,027	0,029
Мнемосхема	0,017	0,017	0,018	2,739	0,015	0,042	0,025	0,05	0,034	0,029
Пейзаж (CG)	0,014	0,015	0,021	2,722	0,013	0,042	0,028	0,046	0,031	0,03

б) Обчислювальна складність (за часом виконання) для різних схем розгорток і типів зображень.

Рис. 5. Значення PSNR та обчислювальна складність для різних схем розгорток БЕЗ врахування етапу передобробки (ОБ 8×8 ел.), для 3-х типів тестових зображень (Рис.1(л-н))

З отриманих результатів слід, що вибір схеми розгортки безпосередньо впливає на кількість операцій та, відповідно, на загальний час виконання. Варіанти з «простою» послідовністю забезпечують найменшу обчислювальну складність і доцільні к застосуванню в умовах існування обмежень на використання наявних ресурсів чи при необхідності мінімізації затримки обробки. Натомість комбіновані (двопрохідні схеми) й рандомізовані підходи, дають помітний приріст часу виконання через додаткові логічні перевірки і операції перестановки, що призводить до зростання загального часу обробки. Порівняно з іншими способами розгорток, цей приріст найвиразніший для схеми «Випадкова», що робить її

© Гончаров М. О., Малахов С. В., Жілєнков Д. В. (2025) Дослідження складності та властивостей різних схем розгортки вихідних даних стеганоконтента, в умовах зміни сценаріїв атак. Сучасний захист інформації, 4(64), 44–58. https://doi.org/10.31673/2409-7292.2025.041205

«найдорожчою» з точки зору необхідних обчислень і ресурсів. Слід зауважити, що час виконання розгортки, також, залежить від розмірності блоків, типу використаного зображення та, відповідно, кількості серій ОБ, що формуються [8-9, 11]. З точки зору головної мети – здійснення стегановставки даних, такий взаємозв'язок виглядає позитивно, зумовлюючи посилення комбінаторики мультиплексування на подальших етапах дослідного алгоритму. Т.ч., зростання чисельності серій ОБ, підвищує стійкість до локалізації діючих параметрів кодування й ускладнює завдання атакуючої сторони, але одночасно воно має очевидний мінус – значне зростання обчислювальної складності процедур на 2-му рівні захисту (в частині DCT, [14]).

Отже, виникає певне протиріччя з загальним трендом на зниження обчислювальних витрат всього алгоритму, що вирішується за рахунок впровадження концептуальної ідеї (ідеологеми): - від єдиного «складного», до ланцюжка «простого». Здебільшого це стосується реалізації процедур кодування з перетворенням [4], безпосередньо перед етапом мультиплексуванням параметра середньої яскравості наявних ОБ, на 2-му рівні захисту (див. крок №6 на рис.1 в [12]).

В цілому, основною метою проведеного моделювання була демонстрація важливості використання різних розгорток для формування потрібного компромісу між: 1 – процедурною складністю реалізації того чи іншого способу розгортки і, відповідно, її можливостями, щодо протидії спробам ідентифікації та неавторизованої екстракції контенту; 2 - впливом обраної схеми розгортки на результат формування відповідного масиву серій ОБ (кількість серій та їх середня довжина) [11]. Також, відносно рис.2-4, важливо зазначити, що ці результати отримані за умови, що усі експерименти проводилися без зміни поточних значень (тобто без їх загрублення [5]) елементів вихідних зображень, на етапі формування масиву серій ОБ при фіксованому розмірі опорних блоків (8×8 ел). Таким чином, оцінювався лише вплив обраної схеми розгортки, без будь-яких втручань безпосередньо в процес формування серій ОБ. Тобто, в частині процесу кодування довжин серій [4], все залишалось «як воно є».

Результати моделювання атак тестового контенту

На рис.6 представлені результати спроб несанкціонованого вилучення тестових зображень для модифікацій розгортки «Подвійна Змійка-2», які відповідають наступним параметрам передобробки вихідного контенту [6]: - маска згладжування 3×3 ел.; - другий варіант передобробки [16-17]; - порогове значення рівня відмінності яскравості елементів сусідніх блоків [5] вихідного зображення, P_{Z1}=3.

Наведені на рис.6 зразки відповідають ситуації при якій нападник вірно підібрав розмірність ОБ, але припустив помилку в діючий схемі розгортки зображення-контенту, застосувавши для цього розгортку типу «Змійка-2» (рис.1(д)). З аналізу наведених на рис.6 зразків слід, що при діючої комбінації використаних розгорток (вихідна ↔ атакуюча), структура артефактів «зламаних» зображень, в цілому, дозволяє ідентифікувати атакований контент, щонайменше на рівні класифікації типу вихідних зображень. Аналогічно випадку із зразком (в) на рис.2-4, це зумовлено збігом певних ділянок траєкторії руху використаних схем розгортки, й відповідно, близькістю кореляційних характеристик формованого масиву серій ОБ (щонайменше на таких відрізках траєкторії руху розгортки).

Незважаючи на наявність артефактів (зразки (а-в) на рис.6), результати атаки дозволяють скласти загальне уявлення про характерні риси вихідного зображення (контури обличчя, структурну схему, основну композицію сцени тощо). При цьому, безумовно, ступінь спотворення зображення залишається достатньо високою щоб ускладнити точну візуальну ідентифікацію всіх об'єктів (елементів) сцени, що відображається. Інакше кажучи, часткова впізнанність оригіналу спостерігається, проте вона не надає зловмиснику простого і надійного способу визначити весь вміст прихованого контенту та/чи однозначно відновити його дрібні деталі. А цього, як було зазначено вище, вочевидь мало, особливо з урахуванням роботи інших рівнів захисту (в даному випадку, деактивованих). Аналіз застосування різних схем розгортки,

в більшості випадків, виявив присутність схожих наслідків у структурі й інтенсивності викривлень, в разі спроб їх компрометації (підбору діючих параметрів). Це, опосередковано, знижує ймовірність успішності атаки, не надаючи атакуючому чіткого уявлення про «вірний» вектор зусиль.

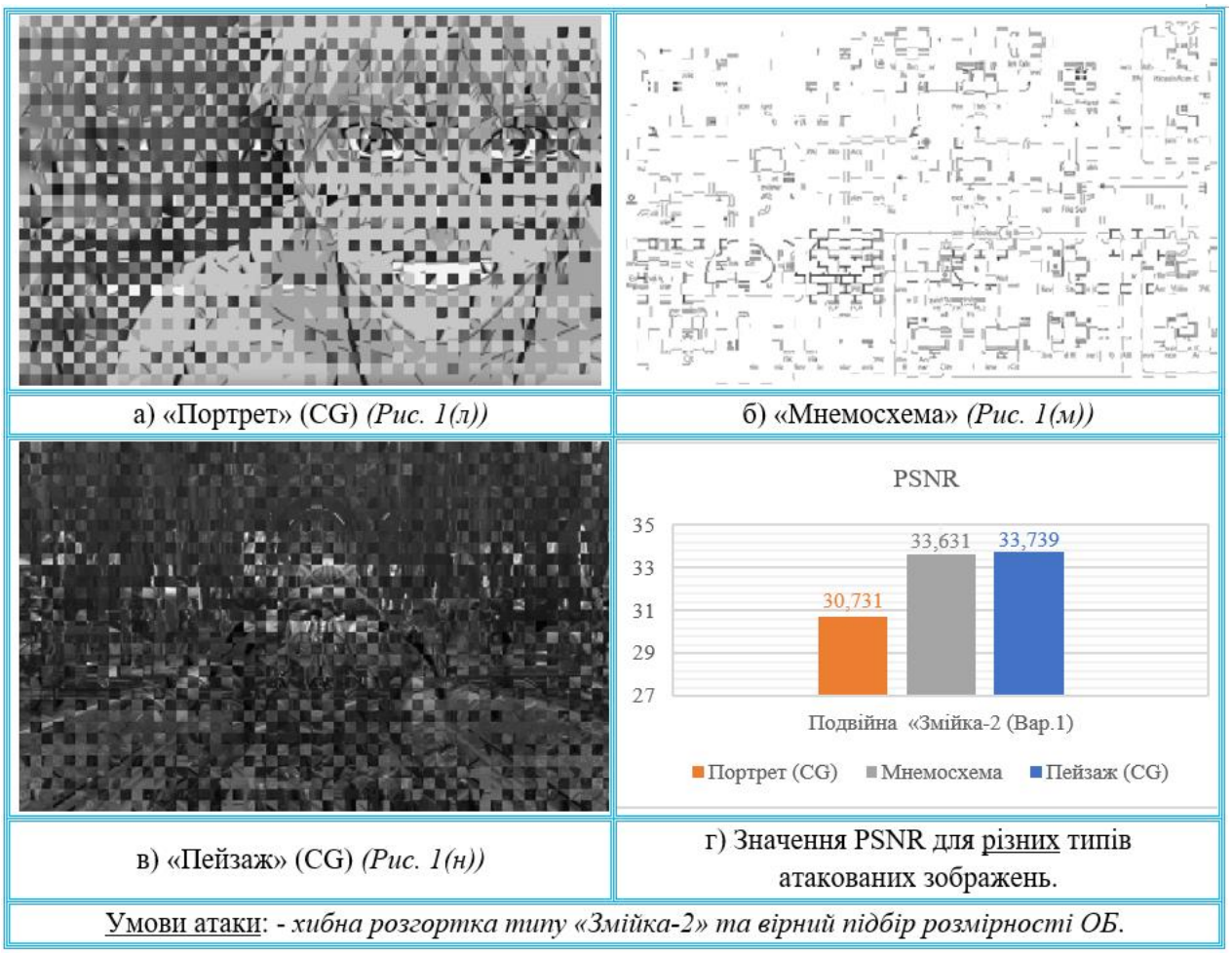


Рис. 6. Результати атаки тестового контенту для розгортки «*Подвійна Змійка-2*» (ОБ 16×16 ел.), в умовах вилучення контенту за хибною схемою «*Змійка-2*»

Узагальнюючи все вищесказане, можна констатувати, що розгортки, які реалізують зигзагоподібні та/або кратні схеми, найкращим чином враховують структурні особливості, що притаманні зображенням різного типу, забезпечуючи при цьому max відстороненість від наявних кореляційних зв'язків між різними елементами (окремими пікселями та цілими блоками/фрагментами) вихідного зображення. Ці схеми, за замовчуванням, створюють найсприятливіші умови для max ускладнення процедур зворотної компіляції вихідного контенту [8-14]. Крім того, варіативність реалізації різних схем (наприклад, «старт» у різних точках чи зміна руху на першому кроці), додатково нарощує комбінаторику відповідного елементу в структурі ключа екстрактора даних (див. рис. 1 в роботі [7]). Навіть у разі компрометації основних захисних механізмів відразу на двох основних рівнях мультиплексування [12,14], використання різних варіацій схем розгортки дозволяє успішно протидіяти спробам нелегітимної екстракції контенту.

Висновки

1. Проведене моделювання має демонстраційний характер та дозволяє візуалізувати наслідки використання різних схем розгортки (принципу формування масивів серій ОБ) в умовах компрометації інших, основних рівнів захисту, дослідного стеганоалгоритму.

2. Дослідження різних розгорток свідчить, що впровадження механізму зміни використовуваної схеми розгортки, є ефективним та обчислювально простим інструментом для протидії спробам неавторизованого доступу до контенту. В окремих випадках (піковий ресурсний дефіцит за умов збереження необхідності підтримки обміну інформацією), за умови використання відповідних режимів й налаштувань (*зигзагоподібні або кратні схеми*), етап реалізації розгортки може застосовуватися навіть, як самостійний інструмент захисту.

3. Лінійні розгортки (*рис.1(а-в,д*) легше реалізуються, проте більш «слабкіші» до атак, і навпаки - більш складніші схеми підвищують захищеність, однак ускладнюють процес обробки та в меншій мірі сприяють підтримці ресурсного консенсусу в умовах дефіциту наявних обчислювальних потужностей, використовуваних апаратних платформ.

4. Варіативність налаштувань використовуваних схем розгортки (зсув точки, «старту» або зміна напряму руху на першому кроці тощо), забезпечує комбінаторність відповідного елементу в структурі ключа екстрактора даних (рис. 1 в [7]), що виступає ефективним інструментом з протидії спробам неавторизованого вилучення контенту.

5. Представлені результати свідчать, що «вдалий» підбір діючих параметрів обробки даних, навіть відразу на двох основних рівнях захисту [5,12], не гарантує успішної зворотної компіляції вихідного контенту, що добре підтверджують відповідними зразками «атакованих» тестових зображень (див. зразки (ж) та (з) на рис.2-4).

6. Розгортки, що реалізують зигзагоподібні та будь-які «кратні» схеми (рис.1(е,з)), за замовчуванням, створюють найсприятливіші умови для max ускладнення процедур неавторизованої вилучення вихідного контенту.

Перелік посилань

1. Fridrich, J. (2009). Steganography in Digital Media: Principles, Algorithms, and Applications. Cambridge: Cambridge University Press.

2. Конахович, Г., Прогонов, Д., & Пузиренко, О. (2018). Комп'ютерна стеганографічна обробка й аналіз мультимедійних даних : підручник. Київ: Центр навчальної літератури.

3. Кузнецов, О. О., Євсеєв, С. П., & Король, О. Г. (2011). Стеганографія: навч. посіб. Видавництво ХНЕУ.

4. Pratt, W. K. (1978). Digital Image Processing. John Wiley & Sons.

5. Лесная, Ю., Гончаров, Н., & Малахов, С. (2021). Отработка концепта многоуровневого мультиплекса данных гибридного стеганоалгоритма. Збірник наукових праць SCIENTIA. URL: https://ojs.ukrlogos.in.ua/index.php/scientia/article/view/17666.

6. Гончаров, М. О., & Малахов, С. В. (2021, 21–23 квітня). Моделювання процедур підготовки даних стеганоалгоритма з багаторівневим мультиплексуванням контенту. Комп'ютерне моделювання в наукоємних технологіях (КНМТ-2021): матеріали 7-ї міжнар. наук.-техн. конф. Харків: ХНУ ім. В. Н. Каразіна, 118–122. URL: http://surl.li/axsna.

7. Honcharov, M., & Malakhov, S. (2024). Modeling attempts of unauthorized extraction of steganocontent under different combinations of data key-extractor. Collection of Scientific Papers «ΛΌГОΣ», (March 1, 2024; Paris, France), 234–245. DOI: 10.36074/logos-01.03.2024.053.

8. Honcharov, M., & Malakhov, S. (2023). Дослідження способів розгортки вихідних блоків зображення-стеганоконтенту як механізму протидії від несанкціонованої екстракції даних. Наука і техніка сьогодні, 4(18), 293–308. DOI: 10.52058/2786-6025-2023-4(18)-293-308.

9. Гончаров, М., Малахов, С., & Колованова, Є. (2024). Результати моделювання різних схем просторової орієнтації та розгортки серій опорних блоків зображень для протидії несанкціонованої екстракції стеганографічних даних. Комп'ютерні науки та кібербезпека, (2), 58–70. DOI: 10.26565/2519-2310-2023-2-06.

10. Лєсная, Ю., Гончаров, М., Азаров, С., & Малахов, С. (2023). Візуалізація спроб несанкціонованої екстракції стеганоконтенту при помилковому визначенні діючих способів розгортки серій. Grail of Science, (24), 335–340. DOI: 10.36074/grail-of-science.17.02.2023.061.

11. Лєсная, Ю., Гончаров, М., Семенов, А., & Малахов, С. (2023). Моделювання розгортки серій опорних блоків зображення, як інструменту з протидії спробам несанкціонованої екстракції стеганоконтенту. Collection of Scientific Papers «ΛΌГОΣ», (March 31, 2023; Zurich, Switzerland), 109–115. DOI: 10.36074/logos-31.03.2023.33.

12. Лєсная, Ю., Гончаров, М., & Малахов, С. (2023). Результати моделювання спроб несанкціонованого вилучення стеганоконтенту для різних комбінацій атаки дослідного стегоалгоритму. Scientific Collection «InterConf», (141), 338–345. URL: https://archive.interconf.center/index.php/conference-proceeding/article/view/2319.

13. Лєсная, Ю., Гончаров, М., Малахов, С., & Мелкозьорова, О. (2023). Результати несанкціонованої екстракції стеганоконтенту при реалізації двохпрохідної розгортки серій вихідних блоків. Collection of Scientific Papers «ΛΌГОΣ», (March 3, 2023; Bologna, Italy), 65–67. DOI: 10.36074/logos-03.03.2023.19.

14. Лесная, Ю., Гончаров, Н., & Малахов, С. (2023). Способы развертки параметров серий опорных блоков изображений, как элемент составного ключа экстрактора данных стегоалгоритма. Grail of Science, (23), 254–258. DOI: 10.36074/grail-of-science.23.12.2022.37.

15. Гончаров, М., & Малахов, С. (2025). Моделювання і оцінка обчислювальної складності основних процедур, початкових етапів гібридного стеганоалгоритму. Комп'ютерні науки та кібербезпека, (1), 41-59. DOI: 10.26565/2519-2310-2025-1-04.

16. Гончаров, М., Нарєжній, О., & Малахов, С. (2025). Аналіз передумов для забезпечення консенсусу ресурсів при виконанні процедур вставлення стеганографічних даних. Сучасний захист інформації, 63(3), 37–47. DOI: 10.31673/2409-7292.2025.030518.

17. Honcharov, M., & Malakhov, S. (2025). Assessment of the complexity of input data preprocessing procedures for implementing steganographic transformations. Scientific Trends and Trends in The Context of Globalization. Proceedings of the 9th International Scientific and Practical Conference. Umea, Sweden. 2025. Pp. 275-283. URL: https://archive.interconf.center/index.php/2709-4685/issue/view/19-20.06.2025/262.

Надійшла 26.09.2025

УДК 004.05:004.942
DOI: 10.31673/2409-7292.2025.041206

Добришин Ю.Є.

МЕТОДИКА ОПИСУ ВЗАЄМОЗВ'ЯЗКІВ МІЖ МНОЖИНАМИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ, ДЕФЕКТІВ, СПОСОБІВ ЇХ ВИЯВЛЕННЯ У ПРОЦЕСІ ДІАГНОСТУВАННЯ ПОШКОДЖЕНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

У статті розглянуто актуальну науково-прикладну проблему діагностування пошкодженого програмного забезпечення, що зазнало впливу кібератак, у контексті підвищення кіберстійкості сучасних інформаційних систем. Метою дослідження є формалізація процесу діагностування пошкодженого програмного забезпечення внаслідок дії кібератак, що забезпечує системне представлення взаємозв'язків між множинами програмних модулів, дефектів, способів їх виявлення та методів відновлення. Для досягнення поставленої мети застосовано методи системного аналізу, математичного моделювання, теорії множин і формальної логіки, які дозволили описати технологічний процес діагностування у вигляді сукупності взаємопов'язаних формальних відношень. Розроблена формалізована методика дає можливість перейти до автоматизованого діагностування пошкодженого програмного забезпечення. Вона забезпечує систематизацію відомостей про дефекти, підвищує точність локалізації порушень, знижує трудомісткість аналізу та скорочує час відновлення після інцидентів. Математичний апарат, що запропонований у роботі, дозволяє відображати не лише факт появи дефекту, але й його властивості, взаємозв'язки з іншими дефектами, умови виникнення та можливі способи усунення. Результати дослідження мають теоретичне і практичне значення. Запропонована методика сприяє підвищенню ефективності відновлення працездатності програмного забезпечення після кібератак, мінімізації фінансових і репутаційних втрат, а також покращенню якості та надійності функціонування інформаційних систем. Вона створює передумови для подальших досліджень у напрямі розробки інтелектуальних діагностичних систем, заснованих на машинному навчанні та логічному моделюванні взаємозв'язків між дефектами.

Ключові слова: автоматизація діагностики, відновлення працездатності систем, діагностування програмного забезпечення, кібербезпека, кібератаки, математичне моделювання, формалізована модель.

Вступ і формулювання проблеми

На теперішній час значно зростає кількість та складність кібератак, які становлять критичну загрозу для функціонування інформаційних систем. Еволюція технологій кіберзлочинності, зокрема застосування штучного інтелекту для генерації поліморфного шкідливого коду, здатного створювати тисячі варіантів і уникати виявлення традиційними засобами захисту, вимагає переосмислення існуючих підходів до забезпечення кібербезпеки.

У цьому контексті особливого значення набуває діагностування пошкодженого програмного забезпечення, яке дозволяє оцінити стан компонентів системи після впливу кібератак і визначити оптимальні шляхи їх відновлення. Критичним аспектом діагностики є систематизація виявлених пошкоджень через класифікацію та кодування дефектів програмного забезпечення, що дозволяє стандартизувати процес документування інцидентів та забезпечити ефективний обмін інформацією між фахівцями. Вона є необхідною