КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

БУДІВНИЦТВА І АРХІТЕКТУРИ

автоматизації і інформаційних технологій

(факультет)

кібербезпеки та комп'ютерної інженерії

(назва випускової кафедри)

КВАЛІФІКАЦІЙНА РОБОТА

ЗДОБУВАЧА СТУПЕНЯ ВИЩОЇ ОСВІТИ БАКАЛАВР

на тему:

«**Ідентифікація вразливостей інформаційних активів об'єкту захисту**»

Токар Анастасія Андріївна

(прізвище, ім'я та по батькові здобувача повністю)

Київ 2026 р.

КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ БУДІВНИЦТВА І АРХІТЕКТУРИ

автоматизації і інформаційних технологій

(факультет)

кібербезпеки та комп'ютерної інженерії

(назва кафедри)

ЗАТВЕРДЖУЮ

Завідувач кафедри

Делембовський М.М.

„___" _____20___року

КВАЛІФІКАЦІЙНА РОБОТА

ЗДОБУВАЧА СТУПЕНЯ ВИЩОЇ ОСВІТИ БАКАЛАВР

«Ідентифікація вразливостей інформаційних активів об'єкту захисту»

(назва)

Я як здобувач вищої освіти КНУБА розумію і підтримую політику закладу з академічної доброчесності. Я не надавав(-ла) і не одержував(-ла) недозволену допомогу під час підготовки цієї роботи. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело.

Здобувач Токар Анастасія Андріївна

(прізвище, ім'я та по батькові повністю)

125 «Кібербезпека»

(спеціальність)

Безпека інформаційних та комунікаційних систем

(освітня програма)

Група БІКС-22

Керівник Ізмайлова О.В.

(прізвище та ініціали)

доцент кафедри кібербезпеки та комп'ютерної інженерії, кандидат технічних наук

(вчене звання, науковий ступінь)

Рецензент_____

(прізвище та ініціали)

Ідентичність підтверджую

Київ 2026р

КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ БУДІВНИЦТВА І АРХІТЕКТУРИ

Факультет: автоматизації і інформаційних технологій
Випускова кафедра: кібербезпеки та комп'ютерної інженерії
Ступінь вищої освіти: бакалавр
Спеціальність: 125 «Кібербезпека»
Освітня програма: Безпека інформаційних і комунікаційних систем

ЗАТВЕРДЖУЮ
Завідувач кафедри
Делембовський М.М.
„___" _____20___року

З А В Д А Н Н Я
ДО ВИКОНАННЯ КВАЛІФІКАЦІЙНОЇ РОБОТИ ЗДОБУВАЧА СТУПЕНЯ ВИЩОЇ ОСВІТИ БАКАЛАВР

Токар Анастасія Андріївна
(прізвище, ім'я та по батькові здобувача)

1. Тема роботи «Ідентифікація вразливостей інформаційних активів об'єкту захисту»

затверджена наказом ректора КНУБА № 576/52-15/13.2/26 від «14» травня 2026 року

2. Керівник роботи

доцент кафедри кібербезпеки та комп'ютерної інженерії, кандидат технічних наук Ізмайлова Ольга Василівна
(прізвище, ім'я та по батькові, науковий ступінь, вчене звання)

3. Термін подання здобувачем роботи до захисту 30 травня 2026

4. Зміст пояснювальної записки за розділами:

Р. 1. Основні поняття та місце ідентифікації вразливостей у системах управління інформаційною безпекою

Р. 2. Підходи, моделі та методи ідентифікації вразливостей інформаційних активів

Р. 3. Програмна реалізація функціональних компонентів для ідентифікації вразливостей

5. Календарний план виконання роботи:

Види робіт та їх зміст	Дата виконання
Розділ 1	03.05.2026
Розділ 2	15.05.2026
Розділ 3	25.05.2026
Остаточне оформлення роботи	01.06.2026
Направлення роботи для перевірки на плагіат	08.06.2026
Попередній захист роботи	08.06.2026
Направлення роботи на рецензування	08.06.2026

6. Дата видачі завдання 10.02.2026

Керівник _____ _____

Здобувач _____ _____

РЕЗЮМЕ (SUMMARY) *до кваліфікаційної випускової роботи здобувача*	ПІБ *Токар Анастасія Андріївна* *Tokar Anastasiia Andriivna*		
ЗВО	Київський національний університет будівництва і архітектури		
Тема *(українською та англійською)*	«Ідентифікація вразливостей інформаційних активів об’єкту захисту» «Identification of vulnerabilities of information assets of the object of protection»		
Освітній ступінь	Бакалавр		
Факультет	Автоматизації і інформаціних технологій		
Випускова кафедра	Кібербезпеки та комп’ютерної інженерії		
Спеціальність	125 «Кібербезпека»		
Освітня програма	Безпека інформаційних і комунікаційних систем		
Керівник	Ізмайлова Ольга Василівна		
Обсяг роботи:	*Поснювальна записка, стор.*	*Розділів*	*Презентація, кількість слайдів*
	114	3	15
Розділ 1	Основні поняття та місце ідентифікації вразливостей у системах управління інформаційною безпекою		
Розділ 2	Підходи, моделі та методи ідентифікації вразливостей інформаційних активів		
Розділ 3	Програмна реалізація функціональних компонентів для ідентифікації вразливостей		
Висновки по роботі	Розглянуто теоретичні основи ідентифікації вразливостей інформаційних активів об'єкта захисту, проведений аналіз нормативно-правової бази. Проведено сканування, побудована модель загроз та модель порушника, проведено експертне ранжування восьми категорій інформаційних активів, виконано аріоритезацію загроз за допомогою методу аналізу ієрархій (MAI), визначено оцінку CVSS. Розроблено та реалізовано програмне забезпечення для автоматизації розрахунків, пов’язаних з ідентифікацією вразливостей інформаційних активів.		

Ключові слова:	ідентифікація вразливостей, інформаційний актив, CVSS, коефіцієнт конкордації, метод аналізу ієрархій, STRIDE, кібербезпека.
Keywords:	vulnerability identification, information asset, CVSS, concordance coefficient, hierarchy analysis method, STRIDE, cybersecurity.

Здобувач Токар А.А. _____/ _____

Керівник Ізмайлова О.В. _____/ _____

АНОТАЦІЯ

Токар А.А. «Ідентифікація вразливостей інформаційних активів об’єкту захисту». Атестаційна випускова робота бакалавра за спеціальністю: 125 «Кібербезпека та захист інформації», освітня програма: «Безпека інформаційних і комунікаційних систем». – Київський національний університет будівництва та архітектури – Київ 2026.

У роботі досліджуються теоретичні основи ідентифікації вразливостей інформаційних активів, розглянуто нормативно-правову базу та систему стандартів у сфері захисту інформації, проведено класифікацію інформаційних активів та вразливостей. На прикладі тестового мережевого середовища проведено практичну ідентифікацію вразливостей засобами Nmap та OpenVAS. Розроблена програма для автоматизації розрахунків ранжування активів та MAI.

Ключові слова: ідентифікація вразливостей, інформаційний актив, CVSS, коефіцієнт конкордації, метод аналізу ієрархій, STRIDE, кібербезпека.

SUMMARY

Tokar A.A. “Identification of vulnerabilities of information assets of the object of protection”. Bachelor’s degree thesis in the specialty: 125 “Cybersecurity and information protection”, educational program: “Security of information and communication systems”. – Kyiv National University of Construction and Architecture – Kyiv 2026.

This work explores the theoretical foundations of identifying vulnerabilities in information assets, presents the regulatory framework and system of standards in the field of information protection, and classifies information assets and vulnerabilities. Using the example of a test network environment, practical identification of vulnerable accounts using Nmap and OpenVAS was carried out. A program was developed to automate asset ranking and MAI calculations.

Keywords: vulnerability identification, information asset, CVSS, concordance coefficient, hierarchy analysis method, STRIDE, cybersecurity.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

ALE (Annual Loss Expectancy) – очікувані річні втрати, показник кількісного аналізу ризиків

ARO (Annualized Rate of Occurrence) – середня кількість інцидентів на рік

CIA (Confidentiality, Integrity, Availability) – тріада властивостей інформаційної безпеки: конфіденційність, цілісність, доступність

CSRF (Cross-Site Request Forgery) – міжсайтова підробка запитів, тип веб-вразливості

CVE (Common Vulnerabilities and Exposures) – загальний словник відомих вразливостей

CVSS (Common Vulnerability Scoring System) – система оцінки критичності вразливостей

CWE (Common Weakness Enumeration) – класифікація типових слабкостей програмного забезпечення

DoS (Denial of Service) – атака на відмову в обслуговуванні

EF (Exposure Factor) – чинник відкритості перед загрозою, відсоток активу, який зруйнує загроза

IDOR (Insecure Direct Object Reference) – небезпечне пряме посилання на об’єкт, тип веб-вразливості

I (Impact) – розмір шкоди або вплив, який може бути завданий внаслідок експлуатації вразливості

КСЗІ – комплексна система захисту інформації

НД ТЗІ – нормативні документи системи технічного захисту інформації

НСД – несанкціонований доступ

P(Threat) - ймовірність виникнення загрози

P(Vulnerability) – ймовірність того, що вразливість буде використана загрозою

R – ризик

SLE (Single Loss Expectancy) – очікувані разові втрати, «вартість» одного інциденту

SQL (Structured Query Language) – структурована мова запитів

XSS (Cross-Site Scripting) – міжсайтовий скриптинг, тип веб-вразливості

ДЕК – Державна екзаменаційна комісія

ДСТУ – державний стандарт України

ЄКТС – Європейська кредитно-трансферна система

НРК – Національна рамка кваліфікацій

СУІБ – система управління інформаційною безпекою

ТЗІ – технічний захист інформації

ВСТУП

На сьогоднішній день сучасний етап розвитку суспільства характеризується стрімкою цифровою трансформацією всіх сфер діяльності – від державного управління до критичної інфраструктури та бізнес-процесів. Інформаційні технології стали частиною забезпечення функціонування підприємств, установ та організацій. Однак, поряд з беззаперечними перевагами, цей процес супроводжується зростанням кількості та складності кіберзагроз. Кібератаки стають все більш цілеспрямованими, а їх наслідки можуть призводити не лише до фінансових втрат, а й до порушення цілісності державних інформаційних ресурсів, втрати персональних даних та загрози національній безпеці. В наш час забезпечення надійного захисту інформаційних активів виходить на перший план. Ключовим етапом побудови будь-якої ефективної системи захисту є систематичний та всебічний аналіз її слабких місць. Відповідно до професійного стандарту «Аналітик з оцінки вразливостей», саме виявлення та оцінка вразливостей в технічних та організаційних контролях є основою для прийняття обґрунтованих рішень щодо управління ризиками інформаційної безпеки.

Актуальність роботи полягає у необхідності розробки та застосування ефективних методів ідентифікації вразливостей, що дозволяють своєчасно виявляти слабкі місця в системах захисту та мінімізувати ризики реалізації кіберзагроз.

Метою роботи є розробка та обґрунтування методичних підходів та рекомендацій щодо ідентифікації вразливостей інформаційних активів об’єкту захисту для підвищення рівня його кібербезпеки.

Об’єктом дослідження є інформаційна система (або інформаційно-комунікаційна система) об’єкту захисту як сукупність апаратних, програмних, програмно-апаратних засобів та інформаційних ресурсів, що підлягають захисту.

Предметом дослідження є процеси, методи та засоби ідентифікації вразливостей інформаційних активів, а також організаційно-технічні підходи до їх виявлення та документування.

Практичне значення отриманих результатів полягає у можливості застосування розробленої методики для підвищення ефективності роботи фахівців з кібербезпеки та аналітиків з оцінки вразливостей під час проведення обстежень інформаційних систем, а також для вдосконалення процесів управління ризиками інформаційної безпеки в організаціях різних форм власності.

Таким чином, в дипломній роботі було зроблено ідентифікацію вразливостей, класифікацію вразливостей, ранжування інформаційних активів за цінністю, моделювання загроз за методологією STRIDE, пріоритезація загроз за допомогою методу аналізу ієрархій (MAI) і програмна реалізація функціональних компонентів для ідентифікації вразливостей.

1.ТЕОРЕТИЧНІ ВІДОМОСТІ ОСНОВ ІДЕНТИФІКАЦІЇ ВРАЗЛИВОСТЕЙ ІНФОРМАЦІЙНИХ АКТИВІВ

1.1 Законодавче регулювання, система стандартів та нормативних документів

1.1.1 Загальна характеристика

Ефективність процесу ідентифікації вразливостей інформаційних активів безпосередньо залежить від дотримання чинної нормативно-правової бази у сфері захисту інформації. В Україні сформована ієрархічна система правового регулювання, яка включає законодавчі та підзаконні акти (закони, постанови Кабінету Міністрів України, укази), які встановлюють обов'язкові правові норми а також нормативні документи технічного характеру та стандарти (ДСТУ, НД ТЗІ). Така структура забезпечує комплексний підхід до організації інформаційної безпеки та визначає обов'язкові вимоги до процесів виявлення, аналізу та усунення вразливостей.

1.1.2 Конституційні засади та базові інформаційні закони

Фундаментом правового регулювання є Конституція України, яка встановлює право громадян на інформацію (стаття 34) та одночасно визначає можливість обмеження цього права в інтересах національної безпеки, територіальної цілісності та громадського порядку. Зазначене положення створює правову основу для впровадження механізмів захисту інформаційних активів та регламентує необхідність контролю доступу до інформації. Базовим нормативно-правовим актом у сфері інформаційних відносин є Закон України «Про інформацію» (№ 2657-XII), який визначає правовий статус інформації (конфіденційна, службова, таємна) та режими доступу до неї. Відповідно опираючись на положення цього закону, класифікація активів на конфіденційні, службові тощо, яка є першим кроком аналізу вразливостей. Окреме значення має Закон України «Про захист персональних даних» (№ 2297-VI), який встановлює правові вимоги до обробки та захисту персональної інформації. Виявлення вразливості, що може призвести до витоку

таких даних, одразу переводить ризик у категорію правопорушення, що тягне за собою адміністративну або кримінальну відповідальність згідно з Кодексом України про адміністративні правопорушення та Кримінальним кодексом України (статті 361–363, які передбачають покарання за несанкціоноване втручання в роботу комп'ютерів, викрадення інформації). Ключовим галузевим законом є Закон України «Про захист інформації в інформаційно-телекомунікаційних системах» (№ 80/94-ВР). Цей закон визначає основні терміни, що використовуються при аналізі вразливостей, зокрема поняття несанкціонованого доступу (НСД), блокування, витоку інформації та порушення її цілісності. Окрім цього, він запроваджує підхід комплексної системи захисту інформації (КСЗІ). КСЗІ відіграє ключову роль для систем, де обробляється інформація з обмеженим доступом, і першим етапом її створення є саме передпроєктне обстеження, яке включає аналіз вразливостей та загроз. Закон України «Про основні засади забезпечення кібербезпеки України» (№ 2163-VIII) визначає організаційні та правові засади функціонування національної системи кібербезпеки. Він зобов'язує суб'єктів національної системи кібербезпеки, зокрема об'єкти критичної інфраструктури, проводити регулярний аудит кіберзахисту, моніторинг вразливостей та реагувати на кіберінциденти, що обумовлює необхідність впровадження систематичних процедур оцінки захищеності інформаційних активів. Для фахівця з кібербезпеки - це означає необхідність застосування не лише реактивних, а й проактивних методів (сканування, тестування на проникнення) для запобігання атакам. Закон України «Про державну таємницю» (№ 3855-XII), який встановлює особливий режим захисту інформації з обмеженим доступом. В цьому випадку, робота з такою інформацією вимагає відповідних допусків, а методологія аналізу вразливостей має відповідати надзвичайно жорстким нормативним документам системи технічного захисту інформації (НД ТЗІ).

1.1.3 Підзаконні акти (Постанови Кабінету Міністрів України)

Підзаконні акти конкретизують механізми виконання законів. Найважливішими для сфери оцінки вразливостей є:

- Постанова КМУ від 29.03.2006 № 373 «Про затвердження Правил забезпечення захисту інформації в інформаційних, телекомунікаційних та інформаційно-телекомунікаційних системах». Цей документ встановлює загальні вимоги до організації захисту інформації.
- Постанова КМУ від 08.02.2021 № 92, яка внесла зміни до попередніх правил, адаптуючи їх до сучасних реалій (зокрема, щодо електронних комунікацій).
- Постанова КМУ від 16.11.2016 № 821 «Про ліцензування господарської діяльності з надання послуг у галузі криптографічного захисту інформації та технічного захисту інформації». Цей документ встановлює, що проводити оцінку вразливостей (як частину технічного захисту інформації) мають право лише ліцензовані суб'єкти, що гарантує професійність та відповідність методології.

1.1.4 Стандарти та нормативні документи у сфері оцінки вразливостей

Поряд із законодавчими актами важливу роль у забезпеченні ефективності ідентифікації вразливостей відіграють стандарти та нормативні документи, які визначають методологічні та технічні аспекти цього процесу. Їх застосування забезпечує уніфікацію підходів до оцінки захищеності інформаційних систем та дозволяє формалізувати процедури аналізу вразливостей. Національна система технічного захисту інформації (ДСТУ 3396) представлена державними стандартами, що встановлюють основні положення організації захисту інформації. Вони визначають об'єкти захисту, основні загрози та принципи забезпечення безпеки. Особливе значення має стандартизація термінології, яка забезпечує однозначність трактування ключових понять у сфері інформаційної безпеки. А саме:

- ДСТУ 3396.0-96 «Захист інформації. Технічний захист інформації. Основні положення». Цей стандарт закладає фундамент. Він визначає об'єкти технічного захисту (носії інформації, технічні засоби), мету (запобігання витоку, НСД, спеціальному впливу) та організаційні принципи. Для аналітика

це орієнтир, які саме технічні аспекти системи потрібно досліджувати в першу чергу.

- ДСТУ 3396.2-97 «Захист інформації. Технічний захист інформації. Терміни та визначення». Це надзвичайно важливий документ для написання будь-якої наукової роботи чи технічного звіту. Він встановлює єдину термінологію. Наприклад, дає чітке визначення поняттю «канал витоку інформації» або «загроза безпеці інформації». Вживання термінів не згідно з цим стандартом є грубою методологічною помилкою в українській практиці технічного захисту інформації.

1.1.5 Нормативні документи системи технічного захисту інформації (НД ТЗІ)

Це документи, які були розроблені Адміністрацією Державної служби спеціального зв'язку та захисту інформації України (Держспецзв'язку). Вони є основою для фахівців з інформаційної безпеки, деталізуючи процедури оцінки вразливостей до рівня конкретних технічних вимог.

- НД ТЗІ 1.1-002-99 — визначає загальні положення захисту від несанкціонованого доступу.
- НД ТЗІ 2.5-004-99 «Критерії оцінки захищеності інформації в комп'ютерних системах від несанкціонованого доступу». Це найважливіший документ для проведення оцінки. Він встановлює шкалу та метрики, за якими оцінюється захищеність системи. Він визначає рівні захищеності (від найнижчого до найвищого) та вимоги до них.
- НД ТЗІ 3.7-003-05 «Порядок проведення робіт із створення комплексної системи захисту інформації в інформаційно-телекомунікаційній системі». Цей документ описує процес. Він регламентує етапи: обстеження об'єкта (тут проводиться збір даних про активи та попередній аналіз вразливостей), аналіз ризиків, формування вимог, впровадження КСЗІ та атестацію. Без дотримання цього порядку неможливо легітимно впровадити захист у державних інформаційних системах.

- НД ТЗІ 2.5-005-99 «Класифікація автоматизованих систем» — дозволяє визначити клас системи (1, 2, 3), від якого залежить глибина необхідного аналізу захищеності.

1.1.6 Гармонізовані європейські стандарти (ДСТУ ISO/IEC 27000)

Поряд із національними нормативами в Україні застосовуються міжнародні стандарти серії ISO/IEC 27000, які впроваджені як національні. Вони орієнтовані на побудову систем управління інформаційною безпекою та базуються на ризик-орієнтованому підході. Відповідно до цих стандартів, процес управління інформаційною безпекою передбачає постійну ідентифікацію, аналіз та оцінку ризиків, що безпосередньо пов’язано з виявленням вразливостей інформаційних активів. ДСТУ ISO/IEC 27001:2023 «Інформаційна безпека, кібербезпека та захист приватності. Системи керування інформаційною безпекою. Вимоги». Це основний стандарт для СУІБ. Він вимагає від організації ризик-орієнтованого підходу. У розділах 6.1 (Дії щодо ризиків) та 8.2 (Оцінка ризиків) прямо зазначено, що організація повинна постійно ідентифікувати, аналізувати та оцінювати ризики інформаційній безпеці. Процес оцінки вразливостей є ключовим інструментом для виконання цих вимог. ДСТУ ISO/IEC 27002:2023 «Інформаційна безпека, кібербезпека та захист приватності. Засоби керування інформаційною безпекою». Якщо стандарт 27001 каже, що треба робити, то 27002 каже, як це робити технічно. Він містить довідник засобів контролю (controls). Знайшовши вразливість, фахівець звіряється з цим стандартом і обирає відповідний засіб контролю для її усунення (наприклад, контроль доступу, шифрування, захист від шкідливого програмного забезпечення, управління патчами). ДСТУ ISO/IEC 27037:2017 «Інформаційні технології. Методи забезпечення безпеки. Настанови щодо ідентифікації, збирання, отримання та зберігання цифрових доказів». Цей стандарт є критично важливим для процесів цифрової криміналістики, які часто супроводжують розслідування інцидентів, пов’язаних з використанням вразливостей. Він встановлює єдині вимоги до роботи з цифровими доказами, що забезпечує їх юридичну значимість. ДСТУ ISO/IEC 27005 «Інформаційна безпека, кібербезпека та захист приватності.

Настанови щодо управління ризиками інформаційної безпеки». Основною метою ISO/IEC 27005:2022 пропонує два взаємодоповнюючі підходи це є надання структурованого, повторюваного процесу управління ризиками інформаційної безпеки. Підхід, орієнтований на активи, передбачає визначення конкретних інформаційних активів, аналіз загроз та вразливостей, характерних для цих активів, та оцінку ймовірності реалізації загроз. Відповідно до цього підходу, першим кроком є ідентифікація активів, які мають цінність для організації і тому потребують захисту. Підхід, орієнтований на події, зосереджується на загальному ландшафті загроз та аналізі сценаріїв інцидентів. Визначаються ключові події або сценарії, які можуть призвести до порушення безпеки, та оцінюються їх наслідки. Отже, ДСТУ ISO/IEC 27005 надає повну, структуровану та гнучку методологію для виконання ключових етапів управління ризиками.

1.2 Класифікація інформаційних активів

Класифікація інформаційних активів є невід’ємною складовою процесу забезпечення інформаційної безпеки об’єкта захисту та виступає початковим етапом для проведення аналізу вразливостей і оцінки ризиків. В умовах зростання складності інформаційних систем та збільшення кількості загроз, правильна ідентифікація та структуризація активів дозволяє підвищити ефективність управління безпекою. До них належать не лише дані, але й технічні засоби, програмне забезпечення, канали зв’язку та людські ресурси. У цьому контексті важливим є системний підхід до класифікації, який передбачає врахування взаємозв’язків між різними компонентами інформаційної інфраструктури. Процес класифікації базується на визначенні критеріїв, які дозволяють оцінити значущість активів та встановити пріоритети їх захисту. До таких критеріїв належать функціональне призначення активу, рівень доступу до інформації, критичність для бізнес-процесів, а також значущість для забезпечення конфіденційності, цілісності та доступності.

Ідентифікація інформаційних активів є фундаментальним етапом процесу забезпечення інформаційної безпеки, що дозволяє сформувати повне уявлення про склад та структуру інформаційної системи об’єкта захисту. Відповідно до методології, закладеної в ДСТУ ISO/IEC 27001, ідентифікація активів є першим і обов'язковим кроком побудови системи управління інформаційною безпекою, оскільки неможливо захистити те, що не було ідентифіковано та оцінено. Цей процес передбачає застосування комплексу організаційних та технічних методів, які у своєму поєднанні забезпечують максимальну повноту та достовірність отриманих даних.

1.2.1 Організаційні методи ідентифікації активів

Дані методи ґрунтуються на аналізі документації, що регламентує функціонування інформаційної системи. До таких документів належать політики безпеки, технічні регламенти, інструкції з експлуатації обладнання, переліки серверного та мережевого обладнання, схеми топології мережі, а також проєктна та експлуатаційна документація. Аналіз цих матеріалів дозволяє отримати первинне уявлення про склад активів, їх призначення та взаємозв’язки. Ключовим елементом організаційних методів є інтерв’ювання персоналу, яке дозволяє виявити активи, що не були належним чином задокументовані, а також отримати інформацію про особливості їх використання, недокументовані налаштування та потенційні проблеми, з якими стикаються користувачі. Важливу цінність становлять інтерв’ю з системними адміністраторами, які безпосередньо обслуговують обладнання та програмне забезпечення, та з керівниками підрозділів, які визначають бізнес-вимоги до функціонування інформаційних сервісів. За допомогою вивчення бізнес-процесів дозволяє ідентифікувати активи з точки зору їх значення для досягнення цілей організації, допомагаючи визначити критичність активів та приховані залежності між ними.

1.2.2 Технічні методи ідентифікації активів

Дані методи доповнюють організаційні та забезпечують перевірку достовірності отриманих даних. Автоматизована інвентаризація активів за допомогою інструментів мережевої розвідки, таких як Nmap, Zmap або інші спеціалізовані засоби, дозволяє виявити активні пристрої в мережі, визначити їх IP-адреси, відкриті порти та служби, що на них функціонують. Ці інструменти є особливо ефективними для виявлення так званих «тіньових IT» — обладнання та програмного забезпечення, впроваджених без відома IT-відділу. Сканування програмного забезпечення дозволяє ідентифікувати встановлені на активах операційні системи, прикладні програми, версії програмного забезпечення, а також виявити неліцензійне або застаріле програмне забезпечення, яке може бути джерелом додаткових вразливостей. Аналіз конфігурацій дозволяє отримати детальну інформацію про налаштування активів, зокрема про параметри безпеки, права доступу, налаштування мережевих інтерфейсів. Моніторинг мережевого трафіку є важливим методом для ідентифікації активів, які можуть бути не виявлені при статичному скануванні, наприклад, пристроїв, що працюють у динамічному режимі або активуються лише в певні періоди часу. Аналіз трафіку дозволяє виявити взаємодії між активами, визначити інтенсивність їх використання та виявити аномалії, які можуть свідчити про наявність недокументованих пристроїв або програмного забезпечення.

Ідентифікація активів як характеристика процесу управління інформаційною безпекою має декілька ключових аспектів. Повнота охоплення, яка передбачає виявлення всіх без винятку активів, що функціонують в інформаційній системі організації. Неповна ідентифікація створює так звані «білі плями» в системі захисту — активи, які не взяті під контроль, стають найбільш вразливими та можуть використовуватися зловмисниками як точки проникнення. Також, ідентифікація активів характеризується глибиною деталізації, тобто рівнем декомпозиції, на якому розглядаються активи. Замість узагальненого поняття «серверне обладнання» мають бути ідентифіковані конкретні сервери з їхніми унікальними

характеристиками: апаратною конфігурацією, встановленим програмним забезпеченням, мережевими інтерфейсами, призначенням та власником. Глибока деталізація дозволяє більш точно оцінити вразливості та ризики, характерні для кожного конкретного активу. Важливою характеристикою є актуальність даних, яка забезпечується регулярним оновленням реєстру активів. Інформаційна система є динамічним об'єктом: з'являються нові сервери, видаляються застарілі, оновлюється програмне забезпечення, змінюються мережеві налаштування. Якщо реєстр активів не відображає цих змін, він втрачає свою цінність як інструмент управління безпекою. Оптимальним є проведення планових інвентаризацій активів не рідше одного разу на квартал, а також позапланових — після будь-яких суттєвих змін в інфраструктурі. Необхідною умовою є встановлення відповідальності за кожен актив. Кожен ідентифікований актив має мати призначеного власника (owner), який відповідає за його стан, конфігурацію, оновлення та безпеку. Власник активу є ключовою особою в процесах оцінки ризиків та прийняття рішень щодо застосування засобів захисту. Без чіткого визначення власника актив часто залишається без належного захисту. Окремо хочеться виділити класифікацію за цінністю. Не всі активи мають однакове значення для організації, тому процес ідентифікації повинен супроводжуватися оцінкою цінності кожного активу. Ця оцінка може базуватися на вартості відновлення активу, на фінансових втратах у разі його компрометації, на репутаційних ризиках або на значенні активу для виконання ключових бізнес-процесів. Класифікація за цінністю дозволяє диференціювати підходи до захисту та зосередити ресурси на найбільш критичних елементах системи. Також, нерозривно пов'язана з визначенням меж інформаційної системи. Чітке розуміння того, які активи входять до складу системи, а які знаходяться за її межами, дозволяє правильно застосовувати політики безпеки та розмежовувати зони відповідальності. Особливо це важливо для сучасних гібридних інфраструктур, де частина активів може розміщуватися в хмарних середовищах або на потужностях сторонніх провайдерів.

Важливо зазначити, що ідентифікація та класифікація інформаційних активів не є одноразовим процесом. Вони повинні здійснюватися на постійній основі з

урахуванням змін у структурі інформаційної системи, впровадження нових технологій та появи нових загроз. Регулярне проведення інвентаризацій, особливо після внесення змін до інформаційної інфраструктури, забезпечує актуальність даних про активи та дозволяє своєчасно реагувати на потенційні ризики. У контексті даної дипломної роботи ідентифікація активів виступає як початковий і визначальний етап процесу виявлення вразливостей. Без якісної ідентифікації активів неможливо провести повноцінний аналіз їх вразливостей, оскільки невідомо, що саме підлягає дослідженню, які активи є найбільш критичними та потребують першочергового захисту. Тому ідентифікація активів розглядається як основа, на якій будується вся подальша робота з управління вразливостями та ризиками інформаційної безпеки. На практиці цей процес реалізується через створення та ведення реєстру інформаційних активів, який є централізованим сховищем даних про всі активи організації та виступає базою для подальшого аналізу вразливостей.

1.2.3 Класифікація інформаційних активів за функціональним призначенням

Одним із базових підходів до класифікації інформаційних активів є їх розподіл за функціональним призначенням у складі інформаційної системи. Такий підхід дозволяє визначити роль кожного активу у забезпеченні функціонування системи та оцінити його вплив на загальний рівень безпеки. Інформаційні ресурси становлять основу будь-якої інформаційної системи. До них належать дані, що обробляються, зберігаються або передаються, зокрема бази даних, електронні документи, архіви та журнали подій. Саме ці активи найчастіше є об’єктом атак, оскільки містять цінну інформацію. Програмне забезпечення забезпечує обробку інформації та реалізацію функціональних можливостей системи. Воно включає операційні системи, прикладні програми, системи управління базами даних та інші програмні компоненти. Вразливості програмного забезпечення є однією з основних причин порушення безпеки інформаційних систем. Апаратне забезпечення включає технічні засоби, необхідні для функціонування інформаційної системи. До них належать сервери, робочі станції, мережеве обладнання, засоби зберігання даних.

Надійність та захищеність апаратного забезпечення безпосередньо впливають на доступність та цілісність інформації. Мережеві ресурси забезпечують передачу даних між компонентами системи. Вони включають канали зв'язку, маршрутизатори, комутатори та інші елементи мережевої інфраструктури. Вразливості мережевих ресурсів можуть призводити до несанкціонованого доступу або перехоплення інформації.

Людські ресурси є важливим елементом інформаційної системи, оскільки саме персонал здійснює обробку інформації та управління технічними засобами. Людський фактор часто виступає джерелом вразливостей через помилки, недостатню кваліфікацію або зловмисні дії. Таким чином, розподіл активів за функціональним призначенням дозволяє комплексно оцінити структуру інформаційної системи та визначити потенційні точки вразливості в кожному з її компонентів.

1.2.4 Класифікація інформаційних активів за рівнем доступу до інформації

Класифікація за рівнем доступу є одним із ключових аспектів організації захисту інформаційних активів. Вона визначає, хто і за яких умов може отримати доступ до інформації, а також встановлює відповідні вимоги до її захисту. Відповідно до Закону України «Про інформацію» (статті 20–21), інформація за ступенем доступу поділяється на відкриту та інформацію з обмеженим доступом. Відкрита інформація не потребує спеціальних заходів захисту, проте її цілісність та доступність також повинні бути забезпечені. Інформація з обмеженим доступом включає конфіденційну, службову та таємну інформацію. Конфіденційна інформація належить фізичним або юридичним особам і поширюється за їх згодою. Службова інформація використовується в діяльності органів державної влади та підприємств і має обмеження щодо доступу. Таємна інформація становить державну таємницю та підлягає особливо суворому захисту згідно з Законом України «Про державну таємницю». Класифікація за рівнем доступу дозволяє визначити пріоритети захисту та встановити відповідні режими безпеки. Зокрема, активи з обмеженим доступом потребують застосування більш складних засобів

захисту, включаючи криптографічні методи, системи контролю доступу та засоби аудиту.

1.2.5 Класифікація інформаційних активів за критичністю та цінністю

Оцінка критичності інформаційних активів є необхідною умовою для визначення рівня ризику та пріоритетності їх захисту. Критичність активу визначається його значенням для функціонування організації та можливими наслідками у разі порушення безпеки. При оцінці критичності враховуються такі фактори, як вплив на безперервність діяльності, можливі фінансові втрати, репутаційні ризики, а також наслідки для безпеки користувачів і персоналу. Наприклад, вихід з ладу сервера бази даних може призвести до зупинки бізнес-процесів, тоді як компрометація персональних даних може спричинити юридичну відповідальність. Залежно від рівня критичності активи поділяються на критичні, важливі та другорядні. Встановлення рівня критичності активів є основою для пріоритизації заходів захисту та ефективного розподілу ресурсів, дозволяючи зосередити увагу на найбільш важливих елементах системи.

1.2.6 Класифікація за властивостями інформаційної безпеки

Одним із фундаментальних підходів до класифікації інформаційних активів є їх оцінка з точки зору забезпечення основних властивостей інформаційної безпеки: конфіденційності, цілісності та доступності. Конфіденційність передбачає захист інформації від несанкціонованого доступу. Вона є критичною для активів, що містять персональні дані або комерційну таємницю. Цілісність означає збереження точності та повноти інформації, що є важливим для систем обробки даних. Доступність забезпечує можливість своєчасного доступу до інформації, що є критичним для безперервної роботи інформаційних систем. Окрім класичної тріади «конфіденційність – цілісність – доступність» (CIA), в сучасних дослідженнях інформаційної безпеки все частіше розглядається розширена модель, відома як «гексада Паркера» (Parkerian Hexad). Ця модель додатково враховує такі властивості, як автентичність (підтвердження справжності джерела інформації),

підзвітність (можливість відстеження дій з інформацією) та корисність (інформація повинна бути в придатному для використанні вигляді). Врахування цих властивостей при класифікації активів дозволяє більш комплексно підійти до оцінки потенційних загроз та вибору засобів захисту.

1.3 Класифікація вразливостей

Вразливість інформаційного активу — це коли в інформаційній системі або в якійсь її частині є недоліки чи слабкі місця, якими можна скористатися, щоб порушити безпеку системи й завдати шкоди активу. Такі вразливості бувають технічними, організаційними або пов’язаними з людьми. Вразливість у сфері інформаційної безпеки розглядається як одна з причин, через яку можуть реалізовуватися загрози та порушуватися безпека інформаційних активів. Згідно з ДСТУ 3396.2-97 «Захист інформації. Технічний захист інформації. Терміни та визначення», вразливість – це недолік або слабке місце в системі захисту інформації, організації її роботи, правилах обміну або програмно-апаратному забезпеченні, яке може бути використане для реалізації загрози. Вразливості класифікують за джерелом походження, а саме: апаратні, програмні, організаційні та людські.

Апаратні вразливості пов’язані з особливостями або недоліками обладнання: помилками мікрокоду, архітектури чи наявністю закладних механізмів. До них відносять, зокрема, апаратні закладки (Hardware backdoors), а також вразливості типу Meltdown і Spectre, що використовують особливості сучасних процесорів для доступу до пам’яті. Усунення таких проблем часто потребує заміни обладнання. Програмні вразливості зустрічаються найчастіше. Вони виникають через помилки в коді, некоректну обробку вхідних даних або недоліки реалізації механізмів безпеки. До них належать переповнення буфера (buffer overflow), SQL-ін’єкції, міжсайтовий скриптинг (XSS), CSRF, IDOR, а також проблеми з валідацією даних і управлінням сесіями. Організаційні вразливості пов’язані з управлінням інформаційною безпекою (ІБ). Це можуть бути відсутність або недосконалість політик, помилки в розподілі доступу, слабка обізнаність персоналу, відсутність

чітких процедур реагування на інциденти чи резервного копіювання. У таких випадках навіть наявні технічні засоби захисту можуть працювати неефективно. Людські вразливості виникають через дії або бездіяльність користувачів і персоналу. До них відносять використання слабких паролів, помилки адміністрування, недотримання правил безпеки, а також вплив соціальної інженерії чи зловмисні дії інсайдерів. Тут зазвичай потрібне поєднання технічних і організаційних заходів, а також навчання.

За місцем виникнення вразливості поділяють на мережеві, системні, прикладні та пов’язані із засобами захисту. Мережеві вразливості стосуються інфраструктури та протоколів передачі даних. Вони можуть бути пов’язані з помилками в TCP/IP, DNS, HTTP, некоректною конфігурацією обладнання, відсутністю сегментації або відкритими портами, що не використовуються. Вразливості операційних систем зазвичай пов’язані з налаштуванням доступу, наявністю зайвих служб, відсутністю оновлень або використанням застарілих версій. Також сюди відносять проблеми з автентифікацією та авторизацією. Вразливості прикладного програмного забезпечення виникають у веб-додатках, базах даних та інших прикладних системах. Найчастіше це помилки перевірки вхідних даних, використання застарілих бібліотек або недоліки в управлінні сесіями та правами доступу. Окремо групою виділяють вразливості засобів захисту інформації. Вони пов’язані з помилками конфігурації міжмережевих екранів, систем виявлення вторгнень, антивірусів або криптографічних засобів. У таких випадках система захисту може не виконувати свої функції.

За складністю використання вразливості умовно поділяють на три рівні. Вразливості з низькою складністю (Low) можна використати без спеціальних знань, часто за допомогою стандартних інструментів. Це, можуть бути, стандартні паролі або відкриті порти з небезпечними службами, для яких уже існують готові експлойти. Середній рівень (Middle) складності передбачає наявність певної підготовки або доступу до внутрішніх ресурсів системи. Сюди можна віднести складніші атаки через XSS або використання помилок у логіці роботи веб-додатків. Вразливості з високою складністю (High) потребують глибоких знань і спеціальних

інструментів. До них належать zero-day вразливості або ті, що використовуються через складні ланцюжки атак.

За наслідками використання вразливості поділяють за тим, на що саме вони впливають. А саме на конфіденційність, цілісніть, доступність. Порушення конфіденційності пов’язане з несанкціонованим доступом до інформації, її розкриттям або перехопленням. Порушення цілісності означає можливість змінювати або видаляти дані, виконувати несанкціоновані операції чи підміняти інформацію. Порушення доступності проявляється у недоступності сервісів або даних для користувачів, зокрема через атаки типу DoS або збої програмного забезпечення.

1.4 Зв’язок між активами, вразливостями та загрозами

Активи, вразливості та загрози існують не окремо. Ризик з’являється тоді, коли загроза може використати вразливість і завдати шкоди активу. Можна навести простий приклад для розуміння. Сервер із клієнтськими даними — це актив. Якщо на ньому не встановлене останнє оновлення безпеки, це вразливість. А зловмисник, який шукає через інтернет незахищені сервери, — це загроза. Ризик виникає, коли зловмисник знаходить сервер і викрадає дані. Загроза сама по собі ще нічого не вирішує. Якщо вразливості немає, система захищена. Вразливість без загрози теж не проблема — наприклад, недолік у системі, до якої ніхто не має доступу. Але коли загроза «знаходить» вразливість, це може призвести до порушення конфіденційності, цілісності або доступності активу. Цей зв’язок часто записують як просту формулу: ризик — це загроза, помножена на вразливість і на цінність активу. Тобто навіть за високої ймовірності загрози та очевидної вразливості ризик буде низьким, якщо актив нічого не вартий. І навпаки: для критичного активу навіть невелика вразливість створює серйозний ризик, якщо є хоч якась імовірність загрози.

Загрози бувають різними: хакери, невдоволені співробітники, конкуренти, технічні збої, пожежі, повені, відключення світла, помилки персоналу. Вони можуть бути навмисними або випадковими, внутрішніми чи зовнішніми. Для кожної

загрози важливо розуміти, наскільки вона ймовірна, яку шкоду може завдати і чи є вже якісь засоби захисту. Вразливість — це просто слабке місце. Технічне (помилка в коді, неправильне налаштування), організаційне (відсутність політики безпеки, неуважність персоналу) або фізичне (вільний доступ до серверної). Сама по собі вона не завдає шкоди — лише створює можливість. Актив — це ресурс, який має цінність для організації. Не тільки дані чи обладнання, але й програмне забезпечення, репутація, навички співробітників, довіра клієнтів. Від вартості активу залежить, наскільки серйозними будуть наслідки. Тому злам сервера для однієї компанії може бути дрібницею, а для іншої — катастрофою. Розуміння цього зв'язку допомагає будувати захист не навмання, а свідомо. Неможливо закрити всі вразливості, тому варто зосередитися на тих поєднаннях, які створюють реальний ризик. Наприклад, якщо на сервері не дуже важливі дані і він стоїть за надійним міжмережевим екраном, висока ймовірність зовнішньої атаки ще не означає високого ризику. А публічно доступний сервер із платіжною інформацією клієнтів потребує серйозного захисту навіть за низької ймовірності атаки. На практиці аналіз ризиків вибудовують так: спочатку знаходять активи й оцінюють їхню цінність, потім визначають загрози, які можуть бути для них актуальними, і вже після цього шукають вразливості, які дозволять цим загрозам реалізуватися. Це дозволяє не витрачати час на теоретичні загрози, які навряд чи стануть реальністю.

1.5 Теорія ризиків

Управління ризиками дає змогу вибудовувати процеси інформаційної безпеки раціонально, розподіляючи ресурси для захисту активів компанії. Оцінка ризиків допомагає обирати доцільні заходи: для захисту від суттєвих та актуальних загроз логічно застосовувати дорожчі рішення. Крім того, налагоджений процес дозволяє розробити чіткі плани забезпечення безперервності діяльності та відновлення працездатності — наприклад, глибоке опрацювання різних ризиків допомагає заздалегідь передбачити раптову потребу у віддаленому доступі для великої кількості співробітників, як це може статися під час епідемій або колапсу транспортної системи. Під ризиком інформаційної безпеки розуміють потенційну

можливість використання вразливостей активів конкретною загрозою для заподіяння шкоди організації. Величину ризику умовно виражають як добуток ймовірності негативної події та розміру шкоди. Своєю чергою, ймовірність події — це добуток ймовірності загрози та небезпеки вразливості, виражені в якісній чи кількісній формі. Узагальнену формулу ризику (1.1) можна записати так:

$$R=P(Threat)\times P(Vulnerability)\times I \quad (1.1)$$

де R-ризик, P(Threat) — ймовірність виникнення загрози,P(Vulnerability) — ймовірність того, що вразливість буде використана загрозою, а I — розмір шкоди або вплив, який може бути завданий внаслідок експлуатації вразливості.

Управління ризиками зазвичай включає визначення ризику, його аналіз та оцінку, прийняття рішення щодо ризику, зменшення ризику, а також страхування або перекладання ризику. Цілі аналізу ризиків зводяться до кількох завдань. По-перше, ідентифікувати активи та оцінити їхню цінність. По-друге, виявити загрози активам і вразливості в системі захисту. По-третє, прорахувати ймовірність реалізації загроз та їхній вплив на бізнес. По-четверте, дотриматися балансу між вартістю можливих негативних наслідків і вартістю заходів захисту, а потім дати керівництву рекомендації щодо обробки виявлених ризиків. Перші три етапи — це оцінка ризику, тобто збирання інформації. Четвертий — власне аналіз ризиків, тобто вивчення зібраних даних і підготовка результатів для подальших дій. Аналіз ризиків може бути якісним або кількісним. Якісний аналіз передбачає описову оцінку небезпеки (наприклад, низька, середня, висока) потенційних наслідків та ймовірності їх настання. Кількісний аналіз оперує конкретними цифрами — вартістю, часом простою, витратами. Найчастіше спочатку застосовують якісний аналіз для виділення високо-пріоритетних ризиків, а потім для них проводять кількісний, який є більш трудомістким, але дає точніші результати. За відсутності надійних фактів кількісна оцінка може створити лише ілюзію точності.

Для кількісного аналізу використовують такі показники. Очікувані разові втрати (SLE — Single Loss Expectancy) (1.2) розраховуються як добуток

розрахункової вартості активу та чинника відкритості перед загрозою (EF — Exposure Factor), тобто того, який відсоток активу зруйнує загроза:

$$SLE=AssetValue\times EF \quad (1.2)$$

При цьому у вартість активу слід включати і штрафні санкції за його недостатній захист. Очікувані річні втрати (ALE — Annual Loss Expectancy) (1.3) — це добуток очікуваних разових втрат та середньої кількості інцидентів на рік (ARO — Annualized Rate of Occurrence):

$$ALE=SLE\times ARO \quad (1.3)$$

Очікувані річні втрати допомагають проранжувати ризики: вище значення — критичніший ризик. Цю величину використовують для визначення максимальної вартості реалізованих заходів захисту, оскільки вартість захисних заходів не повинна перевищувати вартість активу або величину прогнозованої шкоди. Цінність заходів захисту для компанії можна визначити як різницю між ALE до впровадження заходів та ALE після впровадження, зменшену на щорічні витрати на реалізацію цих заходів.

Збиток від реалізації атаки буває прямим або непрямим. Прямий збиток — це очевидні втрати: втрата прав інтелектуальної власності, розголошення секретів виробництва, зниження вартості активів або їх руйнування, судові витрати, штрафи, компенсації. Непрямі збитки можуть бути якісними — призупинення або зниження ефективності діяльності, втрата клієнтів, зниження якості товарів чи послуг — або ж недоотриманий прибуток, втрата ділової репутації, додаткові витрати. У зарубіжній літературі також зустрічаються поняття тотального ризику (коли заходів захисту немає взагалі) та залишкового ризику (коли загрози реалізувалися, попри впроваджені засоби захисту).

Після закінчення плану обробки ризиків визначають залишкові ризики. Для цього може знадобитися оновлення або повторне проведення оцінки з урахуванням очікуваних ефектів від запропонованих способів обробки. У процесі прийняття ризиків аналізують і погоджують плани обробки та залишкові ризики, зазначаючи умови, за яких це погодження виноситься. Формується список прийнятих ризиків з обґрунтуванням для тих, які не відповідають раніше визначеним критеріям. Ризики

з часом змінюються: активи та їхня цінність можуть стати іншими, з'являються нові загрози та вразливості, змінюється ймовірність реалізації загроз і рівень їхнього впливу. Тому потрібен безперервний моніторинг змін, у тому числі із залученням зовнішніх контрагентів, які спеціалізуються на аналізі актуальних загроз.

1.6 Методи ідентифікації вразливостей

Для того щоб виявити вразливості в інформаційній системі, використовують різні методи, які умовно поділяють на пасивні та активні.

Пасивні методи не впливають безпосередньо на роботу системи і базуються на зборі та аналізі наявної інформації. Аналіз документації передбачає вивчення політик безпеки, регламентів, інструкцій, проектної документації, схем топології мережі, переліків обладнання та конфігураційних файлів. Це дозволяє виявити організаційні вразливості, невідповідності між документованими та реальними процесами, а також застарілі або некоректні налаштування. Інтерв’ювання персоналу допомагає отримати інформацію про активи, які не були належним чином задокументовані, про особливості їх використання, недокументовані налаштування та проблеми, з якими стикаються користувачі. Особливо корисними є бесіди з системними адміністраторами, які безпосередньо обслуговують обладнання та програмне забезпечення, а також із керівниками підрозділів, які визначають вимоги до інформаційних сервісів. Анкетування дозволяє охопити більшу кількість співробітників і виявити типові помилки або порушення правил безпеки, які можуть бути джерелом вразливостей. Цей метод особливо ефективний для оцінки обізнаності персоналу в питаннях безпеки.

Активні методи передбачають безпосередню взаємодію з інформаційною системою і можуть певною мірою впливати на її роботу. Сканування вразливостей — це автоматизований процес, під час якого спеціалізоване програмне забезпечення (наприклад, OpenVAS, Nessus, Qualys) перевіряє мережеві сервіси, операційні системи та прикладне програмне забезпечення на наявність відомих вразливостей. Принцип роботи таких сканерів полягає в порівнянні версій програмного забезпечення та конфігурацій з базами даних відомих вразливостей

(CVE, OVAL). Сканування дозволяє швидко отримати загальну картину стану безпеки, але може давати хибнопозитивні спрацювання та не виявляє вразливості, пов'язані з логікою роботи програм. Тестування на проникнення (пентест) — це процес імітації реальної кібератаки з метою виявити вразливості та оцінити, наскільки ефективно система захисту може їм протистояти. На відміну від сканування, пентест передбачає спробу реально використати знайдені вразливості, а не просто констатувати їх наявність. Пентест може проводитися за трьома сценаріями: «чорна скринька» (без надання інформації про систему), «сіра скринька» (з частковою інформацією) та «біла скринька» (з повним доступом до конфігурації та архітектури). Цей метод дає найбільш реалістичну оцінку рівня захищеності, але потребує більше часу та кваліфікованих фахівців. Аналіз захищеності мережі включає використання інструментів для картографування мережі (наприклад, Nmap), визначення топології, виявлення відкритих портів, недокументованих сервісів та помилок у конфігурації мережевих пристроїв. Це допомагає виявити такі проблеми, як наявність невикористовуваних, але відкритих портів, застарілі версії мережевих сервісів або неправильне налаштування міжмережевих екранів. Аналіз програмного коду (статичний та динамічний) застосовується для виявлення вразливостей у власних розробках. Статичний аналіз перевіряє вихідний код без його виконання, виявляючи потенційно небезпечні конструкції, помилки валідації даних, переповнення буфера тощо. Динамічний аналіз виконує програму в контрольованому середовищі та відстежує її поведінку, що дозволяє виявити помилки, які проявляються лише під час роботи.

Кожен із цих методів має свої переваги та обмеження, тому на практиці їх зазвичай застосовують у комплексі. Пасивні методи дозволяють отримати загальне уявлення про систему та виявити організаційні проблеми. Активні методи дають змогу перевірити фактичний рівень захищеності. Поєднання різних підходів забезпечує найбільш повну картину наявних вразливостей.

Висновки до розділу 1

У першому розділі дипломної роботи розглянуто теоретичні основи ідентифікації вразливостей інформаційних активів об'єкта захисту. Проведений аналіз нормативно-правової бази України показав, що процес ідентифікації вразливостей має законодавче підґрунтя та регламентується низкою законів, зокрема Законами України «Про захист інформації в інформаційно-телекомунікаційних системах» та «Про основні засади забезпечення кібербезпеки України». Важливу роль у цій сфері відіграють також підзаконні акти, які встановлюють ліцензійні вимоги до діяльності з технічного захисту інформації. Система стандартів (ДСТУ, НД ТЗІ, ISO/IEC 27000) забезпечує єдину термінологію, визначає критерії оцінки захищеності та регламентує порядок проведення робіт із створення комплексних систем захисту інформації.

Класифікація інформаційних активів дозволяє системно підійти до їх аналізу. Визначено, що активи доцільно розподіляти за функціональним призначенням (інформаційні ресурси, програмне та апаратне забезпечення, мережеві ресурси, людські ресурси), за рівнем доступу до інформації (відкрита, конфіденційна, службова, таємна), за критичністю та цінністю, а також за властивостями інформаційної безпеки (конфіденційність, цілісність, доступність).

Ідентифікація активів є початковим і визначальним етапом усього подальшого процесу управління вразливостями. Дослідження класифікації вразливостей показало, що вони можуть бути технічними, організаційними або пов'язаними з людським фактором. За місцем виникнення їх поділяють на мережеві, системні, прикладні та пов'язані із засобами захисту. Важливими ознаками також є складність використання та можливі наслідки реалізації (порушення конфіденційності, цілісності або доступності). Встановлено, що активи, вразливості та загрози тісно пов'язані між собою. Ризик виникає саме тоді, коли загроза може використати вразливість для завдання шкоди активу. Розуміння цього зв'язку дозволяє будувати захист свідомо, зосереджуючись на найбільш критичних поєднаннях.

Розглянуто теоретичні основи аналізу ризиків, зокрема формулу розрахунку ризику, показники кількісного аналізу (очікувані разові та річні втрати), а також підходи до обробки ризиків. Відзначено, що ризики з часом змінюються, тому потребують безперервного моніторингу та регулярного перегляду.

Таким чином, сформована теоретична база створює основу для ідентифікації вразливостей та демонструє актуальність даної теми, що полягає у аналізі вразливостей та розроблення програмної реалізації. Щоб мінімізувати кібератаки на підприємствах та об’єкти критичної інфраструктури.

2. ПІДХОДИ, МОДЕЛІ ТА МЕТОДИ ІДЕНТИФІКАЦІЇ ВРАЗЛИВОСТЕЙ ІНФОРМАЦІЙНИХ АКТИВІВ

Ідентифікація вразливостей – це процес систематичного виявлення, документування та первинної класифікації слабких місць (недоліків) в інформаційній системі, її активах, програмному або апаратному забезпеченні, налаштуваннях чи організаційних заходах захисту, які можуть бути використані для реалізації загроз та порушення безпеки інформаційних активів. Ідентифікація вразливостей відбувається за допомогою систематичних дій, а саме: сканування, виявлення, оцінювання вразливостей, розрахунку цінності активу, багатокритеріальну пріоритизацію загроз.

2.1 Інструменти для сканування вразливостей

Сканери вразливостей – це спеціалізоване програмне забезпечення, яке перевіряє комп’ютери, мережеве обладнання та встановлені сервіси на предмет наявності відомих слабких місць. Вони використовуються для виявлення вразливостей, що виникають внаслідок неправильної конфігурації (наприклад, відкритих портів без контролю доступу) або недоліків програмування (як-от SQL-ін'єкції або переповнення буфера). Основними об'єктами сканування є міжмережеві екрани, маршрутизатори, веб-сервери, сервери додатків, бази даних та кінцеві робочі станції.

Принцип роботи сканера полягає в імітації дій зловмисника: він надсилає спеціальні запити до цільової системи, аналізує відповіді та порівнює отриману інформацію з базою даних відомих вразливостей (наприклад, CVE або NVD). Залежно від типу доступу розрізняють:

- Автентифіковане сканування дає сканеру вразливостей прямий доступ до внутрішніх ресурсів мережі. Для цього використовуються протоколи віддаленого адміністрування, такі як SSH (Secure Shell) або RDP (Remote Desktop Protocol), та облікові дані системи. Завдяки цьому сканер отримує низькорівневу інформацію про хост: перелік служб, деталі конфігурації операційної системи, встановлене програмне забезпечення, проблеми з

налаштуваннями та відсутні оновлення безпеки. Як наслідок, результати такого сканування є точними та детальними.

- Неавтентифіковане сканування не використовує облікових даних і не проникає в систему. Через це воно часто дає хибнопозитивні результати та не здатне надати повну інформацію про операційну систему чи встановлені програми. Такий метод зазвичай застосовують аналітики з безпеки або зловмисники, щоб оцінити стан захищеності зовнішнього периметра – тобто тих активів, які доступні ззовні без автентифікації.

2.2 Характеристика об’єкта дослідження та його інформаційних активів

Об’єктом дослідження є ізольоване тестове мережеве середовище, яке відтворює типову інфраструктуру невеликої організації. Середовище складається з трьох віртуальних машин, розгорнутих на гіпервізорі VMware ESXi. Доступ до мережі обмежено внутрішнім сегментом (підмережа 192.168.56.0/24); взаємодія із зовнішніми мережами відсутня. Така побудова дозволяє безпечно виконувати активне сканування та тестування вразливостей без ризику впливу на реальні інформаційні системи.

Усі три віртуальні машини виконують різні ролі:

• Веб-сервер (192.168.56.105) – основний вузол, на якому запущено Apache HTTP Server (версія 2.4.41), OpenSSH Server (версія 7.9) та захищений веб-доступ через HTTPS. Сервер обробляє запити клієнтів, зберігає конфігураційні файли та логи доступу. На ньому розміщено дані, що мають умовну цінність для організації (персональні дані тестових користувачів). Саме цей актив є найбільш критичним, оскільки його компрометація може призвести до витоку інформації або порушення цілісності веб-додатків.

• Файловий сервер (192.168.56.110) – забезпечує файловий обмін за допомогою протоколів FTP (vsftpd) та HTTP/HTTPS. На ньому зберігаються документи, архіви та резервні копії конфігурацій. Хоча дані на цьому сервері менш критичні, ніж на веб-сервері, втрата або розголошення службової інформації (логів, старих версій

конфігурацій) може створити загрозу для подальшого аналізу системи зловмисником.

• Робоча станція (192.168.56.120) – виконує роль клієнтського пристрою. На ній відкриті порти 445 (SMB) та 3389 (RDP). Цей актив має найнижчу критичність, оскільки на ньому не зберігаються дані, що потребують особливого захисту. Проте станція може бути використана зловмисником як «трамплін» для переміщення всередині мережі.

Для систематизації відомостей про активи складено реєстр (табл. 2.1), який включає унікальний ідентифікатор, категорію активу, його назву, IP-адресу, попередню оцінку критичності та обґрунтування.

Таблиця 2.1 – Реєстр інформаційних активів тестового середовища

ID активу	Категорія	Назва / роль	IP-адреса	Критичність	Обґрунтування
A-001	Апаратне забезпечення	Веб-сервер (додатків)	192.168.56.105	Висока	Забезпечує роботу веб-додатків, доступний з мережі
A-002	Апаратне забезпечення	Файловий сервер	192.168.56.110	Середня	Зберігає конфігураційні файли та документи
A-003	Апаратне забезпечення	Робоча станція	192.168.56.120	Низька	Тестовий клієнт, не містить критичних даних

Продовження таблиці 2.1

A-001-SW1	Програмне забезпечення	Apache HTTP Server (2.4.41)	192.168.56.105	Висока	Основний веб-сервіс, публічний доступ
A-001-SW2	Програмне забезпечення	OpenSSH Server (7.9)	192.168.56.105	Середня	Забезпечує віддалене адміністрування
A-002-SW1	Програмне забезпечення	vsftpd	192.168.56.110	Середня	Файловий обмін, потенційно небезпечний протокол
A-002-SW2	Програмне забезпечення	Apache HTTPS (443/tcp)	192.168.56.110	Висока	Захищений веб-доступ, потребує перевірки сертифікатів
A-003-SW1	Програмне забезпечення	SMB (445/tcp)	192.168.56.120	Низька	Файловий обмін у тестовому сегменті
A-001-DATA	Інформаційні ресурси	Конфігураційні файли веб-сервера	192.168.56.105	Середня	Містять налаштування доступу та обробки запитів

Продовження таблиці 2.1

A-002-DATA	Інформаційні ресурси	Логи доступу	192.168.56.110	Низька	Інформація про з’єднання, потенційно чутлива

Критичність визначено на основі потенційних наслідків порушення безпеки:

- Висока – компрометація призводить до повної або часткової втрати контролю над ключовими сервісами, витоку даних або порушення цілісності інформації, що має прямий вплив на безпеку.
- Середня – використання вразливості може полегшити подальші атаки, але прямий збиток обмежений.
- Низька – вразливість надає лише інформаційні відомості або потребує складних умов для реалізації.

Таким чином, реєстр активів створює базу для подальшого аналізу вразливостей. Найбільш пріоритетними для захисту є веб-сервер (A-001) та пов’язані з ним програмні компоненти (A-001-SW1, A-001-SW2, A-001-DATA), оскільки вони мають високу критичність і безпосередньо взаємодіють з користувачами. Файловий сервер (A-002) та його сервіси (FTP, HTTPS) отримують середній пріоритет, а робоча станція (A-003) – низький.

2.3 Ранжування інформаційних активів за цінністю

Для того щоб оцінювати ризики, недостатньо знати лише технічну критичність вразливості. Потрібно також розуміти, яку цінність має сам актив для організації. Адже компрометація сервера з резервними копіями – це не те саме, що компрометація сервера, на якому зберігаються персональні дані клієнтів. Тому для обґрунтованого визначення пріоритетів захисту було проведено ранжування інформаційних активів за ступенем їх цінності. Для цього залучено чотирьох експертів – фахівців у сфері кібербезпеки, які мають досвід роботи з подібними

інформаційними системами. Кожен експерт оцінив вісім категорій активів за шкалою від 1 до 8, де 1 відповідає найвищій цінності, а 8 – найнижчій. Оцінювалися такі активи: інформаційні активи (власне дані), програмні активи (ПЗ, ОС), апаратні активи (сервери, мережеве обладнання), сервісні активи (веб-сервіси, FTP, SSH), окрема категорія «дані», людські активи (персонал), ПК (робочі станції) та нематеріальні активи (репутація, інтелектуальна власність). Початкові оцінки експертів наведено на рисунку 2.1.

Вихідні ранги								
Експерт	Інформаційні активи	Програмні активи	Апаратні активи	Сервісні активи	Дані	Людські активи	ПК	Нематеріальні активи
1	1	3	2	3	4	5	6	7
2	1	2	3	3	4	5	5	5
3	2	1	3	5	4	6	7	7
4	1	7	7	3	5	6	4	2

Рисунок 2.1 Вихідні ранги

Оскільки деякі оцінки виявилися однаковими (зв’язані ранги), виконано нормалізацію – замість однакових рангів використано їхнє середнє арифметичне значення. Нормалізована матриця, де середні ранги при зв’язних рангах на рисунку 2.2.

Нормалізована матриця (середні ранги при зв'язаних рангах)									
Експерт	Інформаційні активи	Програмні активи	Апаратні активи	Сервісні активи	Дані	Людські активи	ПК	Нематеріальні активи	Ri
1	1	3	3	3	5	6	7	8	36
2	1	2	3,5	3,5	5	7	7	7	36
3	1	2	3	4	5	6	7,5	7,5	36
4	1	2,5	2,5	4	5	6	7	8	36

Рисунок 2.2 Нормалізована матриця

Далі для кожного активу підсумовано ранги, отримані від усіх експертів (Rj), обчислено відхилення від середньої суми та квадрати відхилень. На рисунку 2.3 продемонстровано суми рангів, відхилення та квадрати відхилень.

Суми рангів Rj, середнє T*, відхилення Dj та Dj^2									
Rj	4	9,5	12	14,5	20	25	28,5	30,5	ΣDj^2
T* = n*(m+1)/2	18								
Dj = Rj - T*	-14	-8,5	-6	-3,5	2	7	10,5	12,5	
Dj^2	196	72,25	36	12,25	4	49	110,25	156,25	636

Рисунок 2.3 Суми рангів, відхилення та квадрати відхилень

Для кожного експерта визначено групи зв’язаних рангів і обчислено поправочний коефіцієнт Ti , який враховує наявність однакових рангів. Це необхідно для коректного обчислення коефіцієнта конкордації. Сумарне значення ΣTi використовується в знаменнику формули W. Рисунок 2.4 – поправка зв’язані ранги.

Поправка на зв'язані ранги Ti=Σ(t^3-t)/12					
Групи зв'язаних рангів	[2]	[2, 3]	[2]	[2]	ΣTi
Ti=Σ(t^3-t)/12	0,5	2,5	0,5	0,5	4

Рисунок 2.4 Поправка на зв’язані ранги.

Для оцінки узгодженості думок експертів розраховано коефіцієнт конкордації Кендалла (W). Оскільки в оцінках присутні зв’язані ранги, використано формулу (2.1) з поправкою:

$$W = \frac{12 \times \sum Dj^2}{m^2 \times (n^3 - n) - m \times \sum Ti} \qquad (2.1)$$

Де, m=4, n=8, $\sum Dj^2 = 636$, а $\sum Ti$– сума поправок на зв'язні ранги. Підставивши значення:

$$W = \frac{12 \times 636}{16 \times (512 - 8) - 4 \times 6{,}5} = \frac{7632}{16 \times 504 - 26} = \frac{7632}{8064 - 26} = \frac{7632}{8038} \approx 0{,}95$$

На рисуноку 2.5 продемонстровано коефіцієнт конкордації W.

Коефіцієнт конкордації W	
n (кількість експертів)	4
m (кількість активів)	8
ΣDj^2	636
ΣTi	4
W =	0,948310139

Рисунок 2.5 Коефіцієнт конкордації

Отримане значення W≈0,95, свідчить про дуже високий ступінь узгодженості між експертами. Для перевірки статистичної значущості застосовано критерій Пірсона χ^2 формула (2.2):

$$\chi^2 = m \times (n-1) \times W = 4 \times 7 \times 0{,}95 = 26{,}6 \qquad (2.2)$$

Табличне значення χ^2=14,07 для рівня значущості α=0,05 та ступенів свободи k=7. Оскільки 26,6>14,07, узгодженість є статично значущою. Отже, отримані результати можна використовувати для подальших розрахунків. Рисунок 2.6 –перевірка значущості W – критерій Пірсона.

Перевірка значущості W - Критерій Пірсона X^2	
X^2розр = n*(m-1)*W	26,5526839
Ступені свободи k=m-1	7
X^2табл (α=0,05; k=7)	14,1

Рисунок 2.6 Перевірка значущості і критерій Пірсона.

На основі отриманих рангів Rj виконано перерахунок цінності активів у зручну шкалу від 7 до 10 за формулою (2.3):

$$\text{Цінність} = 10 - \frac{(Rj - Rjmin}{Rjmax - Rjmin} \times 3 \quad (2.3)$$

де Rj_min=4, Rj_max=30,5 таким чином менший ранг, тим вища цінність активу. Результати дозволяють наочно порівняти активи за їх важливістю в єдиній шкалі. Рисунок 2.7 – цінність активів.

Цінність активів в діапазоні оцінювання		
Формула: Цінність = 10 - (Rj - Rj_min)/(Rj_max - Rj_min) *3 (менший ранг = вища цінність)		
Актив	Rj	Цінність
Інформаційні активи	4	10
Програмні активи	9,5	9,377358
Апаратні активи	12	9,09434
Сервісні активи	14,5	8,811321
Дані	20	8,188679
Людські активи	25	7,622642
ПК	28,5	7,226415
Нематеріальні активи	30,5	7

Рисунок 2.7 Цінність активів

Найвищу цінність отримали інформаційні активи (10 балів), що свідчить про їх ключову роль у захисті інформаційної системи. Найнижчу цінність має нематеріальні активи (7 балів), що може бути пов’язано з його допоміжною функцією.

2.4 Моделювання загроз за методологією STRIDE

Для глибокого аналізу загроз інформаційним активам недостатньо лише констатувати наявність технічних вразливостей. Необхідно зрозуміти, які саме категорії загроз можуть бути реалізовані через ці вразливості, та які властивості безпеки (конфіденційність, цілісність, доступність, автентичність, підзвітність) порушуються. Для цього використовується методологія STRIDE, розроблена

компанією Microsoft. На рисунку (2.8) продемонстровано методологію STRIDE, дана абревіатур розшифровується як:

- Spoofing (підміна) – порушення автентичності;
- Tampering (втручання) – порушення цілісності;
- Repudiation (відмова) – порушення підзвітності;
- Information Disclosure (розкриття інформації) – порушення конфіденційності;
- Denial of Service (відмова в обслуговуванні) – порушення доступності;
- Elevation of Privilege (підвищення привілеїв) – отримання неавторизованих прав.

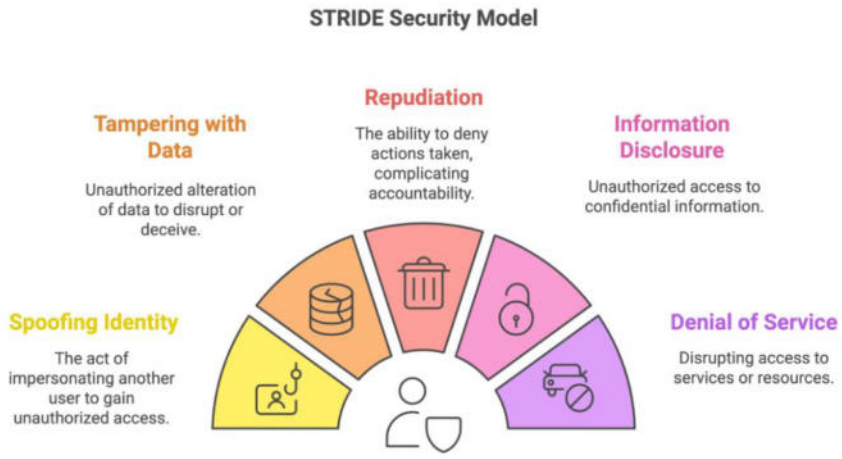


Рисунок 2.8 Методологія STRIDE

Застосування STRIDE дозволяє системно класифікувати загрози, визначити їх пріоритетність та обрати адекватні контрзаходи. У контексті тестового середовища, де виявлено вразливості на веб-сервері (Apache RCE, OpenSSH user enumeration), файловому сервері (FTP anonymous access, слабкі шифри) та робочій станції (SMB signing, RDP weak encryption), методологія дає змогу зіставити кожну вразливість з конкретною категорією загрози.

Spoofing (підміна). Ця категорія охоплює загрози, пов’язані з несанкціонованим доступом до активу шляхом підміни облікових даних користувача або системи. У тестовому середовищі основною передумовою для спуфінгу є вразливість CVE-2020-14145 (OpenSSH User Enumeration) на веб-сервері. Хоча сама вразливість дозволяє лише визначити існування користувачів,

вона є першим кроком для подальшої атаки: знаючи ім’я користувача, зловмисник може спробувати підібрати пароль (атака brute force) або використати соціальну інженерію. Крім того, на файловому сервері відсутність багатофакторної автентифікації (MFA) для доступу до FTP (навіть якщо анонімний доступ вимкнено) створює ризик перехоплення або підбору пароля. Для запобігання спуфінгу рекомендовано впровадити багатофакторну автентифікацію для всіх критичних сервісів (SSH, веб-панелі адміністрування), використовувати протоколи Kerberos або сертифікати замість паролів там, де це можливо, а також запровадити політику складних паролів та регулярне навчання персоналу щодо загроз фішингу.

Tampering (втручання). Загрози цієї категорії пов’язані з несанкціонованою модифікацією або підробкою даних, програмного забезпечення чи логів. Найяскравішим прикладом є критична вразливість CVE-2021-44790 (Apache HTTP Server RCE). Успішна експлуатація цієї вразливості дозволяє зловмиснику виконувати довільний код на веб-сервері, що, своєю чергою, дає змогу змінювати веб-сторінки (дефейс), модифікувати бази даних, додавати шкідливі скрипти або навіть встановлювати бекдори. Іншим прикладом є відсутність вимоги підпису SMB-пакетів (SMB Signing Not Required) на робочій станції, що дозволяє зловмиснику підміняти дані, що передаються між клієнтом і сервером (атака типу man-in-the-middle). Для захисту від втручання необхідно насамперед усунути критичну вразливість шляхом оновлення Apache до версії 2.4.52 або вище, використовувати механізми контролю цілісності (хешування, цифрові підписи для критичних файлів), застосовувати міжмережеві екрани на рівні веб-додатків (WAF) для фільтрації шкідливих запитів, а також увімкнути вимогу підпису SMB.

Repudiation (відмова). Ця категорія описує ситуації, коли користувач або система може заперечувати факт виконання певної дії через відсутність належних доказів. У тестовому середовищі проблема полягає в тому, що логи доступу на веб-сервері та файловому сервері зберігаються локально без захисту від модифікації. Привілейований порушник (наприклад, системний адміністратор) може видалити або відредагувати записи логів, приховавши сліди своєї несанкціонованої діяльності. Крім того, на робочій станції відсутнє детальне журналювання дій з

файловими ресурсами (SMB). Для запобігання відмові необхідно впровадити систему централізованого збору та захисту логів (SIEM), яка агрегує журнали подій з усіх активів у захищене сховище (наприклад, з використанням технології WORM – Write Once Read Many). Також слід налаштувати детальний аудит усіх дій з критичними активами, особливо операцій з конфігураційними файлами та базами даних, та забезпечити неможливість редагування логів звичайними користувачами.

Information Disclosure (розкриття інформації). Ці загрози пов'язані з порушенням конфіденційності – несанкціонованим ознайомленням або копіюванням даних. У тестовому середовищі найбільш очевидною є вразливість FTP Anonymous Access на файловому сервері, яка дозволяє будь-якому користувачеві (навіть без пароля) отримати доступ до файлів, що зберігаються на сервері. Це може призвести до витоку конфігураційних файлів, логів або інших службових документів. Крім того, TCP timestamps на веб-сервері дозволяють віддаленому зловмиснику визначити час роботи системи та її версію (fingerprinting), що полегшує підбір відповідних експлойтів. Ще однією проблемою є використання слабких шифрів SSL/TLS (на файловому сервері) та слабких алгоритмів шифрування SSH (на веб-сервері), що створює ризик дешифрування перехопленого трафіку. На робочій станції слабке шифрування RDP сесій також може призвести до перехоплення даних. Для захисту від розкриття інформації необхідно вимкнути анонімний доступ до FTP (або замінити FTP на SFTP/FTPS), вимкнути TCP timestamps, налаштувати веб-сервер та SSH-сервер на використання тільки сильних шифрів (TLS 1.2/1.3, шифри на основі AES, алгоритми MAC – HMAC-SHA2), оновити SSL/TLS сертифікати, а також застосовувати шифрування даних у стані спокою та при передачі (DLP-системи, контроль доступу на основі ролей).

Denial of Service (відмова в обслуговуванні). Загрози цієї категорії порушують доступність активів – легітимні користувачі не можуть отримати доступ до системи або даних. У тестовому середовищі прямих DoS-вразливостей (наприклад, CVE, що дозволяють викликати відмову) не виявлено, однак непрямі передумови існують. Зокрема, на робочій станції відсутній антивірусний захист та

регулярне створення резервних копій, що робить її вразливою до програм-шифрувальників (ransomware), які можуть заблокувати доступ до локальних файлів. Крім того, веб-сервер не має резервування та балансування навантаження, що у разі DDoS-атаки або апаратного збою призведе до тривалої недоступності сервісів. Для захисту необхідно впровадити антивірусне програмне забезпечення з регулярним оновленням баз, налаштувати резервне копіювання критичних даних (з можливістю відновлення), а для веб-сервера – розглянути можливість використання систем балансування навантаження та резервних каналів зв'язку.

Elevation of Privilege (підвищення привілеїв). Ці загрози дозволяють зловмиснику отримати права доступу, вищі за дозволені (наприклад, звичайний користувач отримує root-доступ). У тестовому середовищі найбільш імовірним сценарієм є компрометація веб-сервера через CVE-2021-44790: отримавши можливість виконувати код від імені веб-сервера (зазвичай користувач www-data), зловмисник може використати локальні вразливості (наприклад, через недостатнє обмеження прав доступу до конфігураційних файлів) для підвищення привілеїв до root. Крім того, на робочій станції відсутність вимоги підпису SMB та слабке шифрування RDP можуть дозволити зловмиснику отримати доступ до облікових записів з розширеними правами. Для запобігання підвищенню привілеїв слід застосовувати принцип найменших привілеїв (PoLP): мінімізувати кількість облікових записів з адміністративними правами, налаштувати суворі права доступу до конфігураційних файлів, регулярно оновлювати програмне забезпечення (щоб усунути локальні вразливості підвищення привілеїв), а також використовувати системи контролю цілісності (SELinux, AppArmor) для обмеження можливостей процесів.

У таблиці 2.2 наведено узагальнену карту загроз STRIDE, що пов'язує виявлені вразливості з категоріями загроз, типовими сценаріями та рекомендованими контрзаходами. Такий підхід дозволяє системно оцінити не лише технічну критичність кожної вразливості, але й її потенційний вплив на різні аспекти безпеки (конфіденційність, цілісність, доступність тощо).

Таблиця 2.2 - модель загроз STRIDE.

Категорія (STRIDE)	Опис загрози для Інформаційних активів	Приклад сценарію	Заходи захисту (Контрзаходи)
Spoofing (Підміна)	Несанкціонований доступ до інформаційного активу шляхом підміни облікових даних користувача або системи. Зловмисник видає себе за легітимного користувача, щоб отримати доступ до даних.	Зловмисник використовує вкрадені логін та пароль адміністратора бази даних (наприклад, через фішинг) і отримує повний доступ до конфіденційної інформації клієнтів.	Впровадження багатофакторної автентифікації (MFA), використання протоколів Kerberos або сертифікатів, політика складних паролів та навчання персоналу.
Tampering (Втручання)	Несанкціонована модифікація або підробка інформаційних активів (даних, ПЗ, логів) без відома власника. Порушення цілісності інформації.	Хакер впроваджує шкідливий код (SQL-ін'єкцію) у форму на вебсайті, що призводить до модифікації у базі даних та фальсифікації.	Використання механізмів контролю цілісності (хешування, цифрові підписи для ПЗ), регулярне оновлення ПЗ, міжмережеві екрани (WAF)

Repudiation (Відмова)	Користувач або система заперечує факт виконання дії з інформаційним активом (створення, видалення, надсилання), через відсутність належних доказів.	Співробітник видалив важливий корпоративний файл з мережевого диска. Логи не зберігаються або не містять достатньої інформації (IP, час, дія).	Впровадження системи централізованого збору та захисту логів (SIEM), налаштування детального аудиту всіх дій з критичними активами, використання механізмів DLP.
Information Disclosure (Розкриття)	Несанкціоноване ознайомлення або копіювання конфіденційної інформації (персональні дані, комерційна таємниця). Порушення конфіденційності.	Співробітник надсилає на пошту файл, через незахищений канал (звичайний email), і файл перехоплюється зловмисником.	Шифрування даних у сховищах та під час передачі (TLS, VPN), класифікація даних, DLP-системи для блокування передачі конфіденційної інформації, контроль доступу на основі ролей.

Продовження таблиці 2.2

Denial of Service (Відмова у сервісі)	Порушення доступності інформаційного активу, що робить неможливим доступ до нього легітимних користувачів або систем.	DDoS-атака на корпоративний сайт або хмарне сховище даних, через що не можуть отримати доступ до електронних реєстрів та документів.	Використання систем балансування навантаження, резервування каналів зв'язку та серверів, антивірусний захист від шифрувальників,регулярне створення бекапів.

Найбільш критичними для тестового середовища виявилися загрози Spoofing (через відсутність MFA та вразливість перерахування користувачів у SSH) та Information Disclosure (через анонімний доступ до FTP на файловому сервері). Загроза Tampering є найбільш небезпечною (оскільки CVE-2021-44790 має CVSS 9.8), але вона прив’язана до однієї вразливості, яку можна усунути оновленням Apache.

2.5 Модель порушника

Для повноцінного аналізу загроз інформаційним активам недостатньо розглядати лише технічні вразливості. Необхідним є побудова моделі порушника, яка дозволяє ідентифікувати потенційні джерела загроз, їх мотивацію, можливості та типові вектори атак. У межах даного дослідження було виділено п'ять основних типів порушників, кожен з яких характеризується специфічними рисами та рівнем небезпеки.

1. Внутрішній користувач (ненавмисний порушник)

Перша категорія порушників охоплює легітимних користувачів інформаційної системи, які не мають наміру завдати шкоди, однак своїми діями створюють передумови для реалізації загроз. Мотивація таких порушників, як правило, не пов'язана зі злим умислом, а зумовлена низьким рівнем обізнаності у питаннях кібербезпеки, виробничою втомою, недбалістю або бажанням спростити виконання рутинних завдань навіть ціною порушення встановлених регламентів. Типові дії ненавмисного порушника є досить різноманітними та часто сприймаються самими співробітниками як нешкідливі звички. До них належить використання простих паролів, які легко запам'ятати, таких як «1234» або дата народження, а також записування паролів на папірцях, що зберігаються поруч із робочим місцем. Поширеною практикою є копіювання конфіденційних документів на незахищені зовнішні носії або у хмарні сховища, що мотивується необхідністю мати до них доступ поза межами офісу. Окрему небезпеку становить низька обізнаність щодо методів соціальної інженерії: співробітники можуть відкривати фішингові листи та переходити за шкідливими посиланнями, що безпосередньо призводить до зараження робочої станції шкідливим програмним забезпеченням. Також трапляються випадки надсилання конфіденційних даних не на ту адресу електронної пошти через неуважність або автозаповнення адресної книги.

Ризики, пов'язані з діяльністю ненавмисного порушника, є надзвичайно високими з огляду на масовість цього явища. Наслідки можуть варіюватися від витоку конфіденційної інформації до зараження корпоративної мережі шкідливим програмним забезпеченням та компрометації облікових записів, які надалі можуть бути використані зловмисниками для проведення цілеспрямованих атак.

2. Внутрішній користувач (умисний порушник / інсайдер)

Друга категорія внутрішніх порушників характеризується наявністю свідомого наміру завдати шкоди інформаційним активам компанії. Мотивація таких інсайдерів може бути різноманітною: від отримання особистої фінансової вигоди шляхом продажу конфіденційних даних конкурентам до помсти роботодавцю, спричиненої, наприклад, звільненням або конфліктом. Окремими випадками є

промислове шпигунство під час переходу співробітника до іншої компанії, а також дії під зовнішнім тиском, зумовленим шантажем або борговими зобов'язаннями. Типові дії інсайдера відрізняються цілеспрямованістю та прихованістю. До них належать масове копіювання баз даних клієнтів перед звільненням, створення так званих «тіньових» копій конфіденційної інформації з метою її подальшого використання або продажу, а також пряма передача доступу до корпоративних ресурсів третім особам. У деяких випадках інсайдери вдаються до саботажу, що виражається у навмисному видаленні або пошкодженні критичних даних напередодні важливих подій чи угод, що здатне паралізувати діяльність організації.

Рівень ризику від умисних внутрішніх порушників оцінюється як критичний. Наслідки їхніх дій включають масштабний витік конфіденційної інформації, прямі фінансові втрати, втрату довіри з боку клієнтів та партнерів, а також потенційні судові позови, що можуть поставити під загрозу подальше існування бізнесу.

3. Привілейований порушник (адміністратор, техпідтримка)

Особливу категорію становлять привілейовані користувачі, до яких належать системні адміністратори, співробітники технічної підтримки та інші особи, що мають розширені права доступу до ключових компонентів інформаційної системи. Їхня особливість полягає не лише у наявності доступу, а й у глибокому розумінні архітектури системи, що дозволяє їм діяти максимально ефективно та приховано. Типові дії привілейованого порушника можуть включати зміну політик доступу з метою власного розширення прав або надання доступу третім особам. Вони здатні створювати так звані «тіньові» облікові записи, які не контролюються стандартними механізмами аудиту. Поширеною практикою є тимчасове відключення антивірусного захисту та систем моніторингу для проведення несанкціонованих дій без ризику бути виявленими. Окрему загрозу становить копіювання резервних копій конфіденційних даних, які часто зберігаються з меншим рівнем захисту. Найнебезпечнішим аспектом є можливість приховування слідів несанкціонованого доступу шляхом редагування або видалення відповідних записів у системних логах.

Ризики, пов'язані з привілейованим порушником, є найвищими серед усіх категорій. У разі реалізації загрози можлива повна компрометація системи захисту, причому довести факт злому та встановити його обставини стає вкрай складно через відсутність достовірних логів. Це створює передумови для тривалого, прихованого використання доступу зловмисником.

4. Зовнішній порушник (хакер, кіберзлочинець, конкурент)

Четверта категорія представляє зовнішніх суб'єктів, які не мають легітимного доступу до інформаційної системи, але прагнуть його отримати. Мотивація зовнішніх порушників є різноманітною: від прямої фінансової вигоди, яку можна отримати, наприклад, через крадіжку коштів або вимагання викупу після DDoS-атаки, до промислового шпигунства, спрямованого на отримання доступу до конфіденційних даних конкурентів. Окремим мотивом може бути дискредитація компанії. Типові вектори атак зовнішніх порушників включають віддалений злом робочих станцій через фішинг, експлойти невстановлених вразливостей або підбір паролів. Після проникнення вони встановлюють шпигунське програмне забезпечення, зокрема кейлогери для перехоплення паролів, або трояни для отримання віддаленого контролю. Атаки можуть бути спрямовані й на канали зв'язку з метою перехоплення трафіку, що містить конфіденційну інформацію. Важливим інструментом залишається соціальна інженерія, наприклад, дзвінки співробітникам від імені технічної підтримки з метою вивідати паролі. Крім того, зовнішні порушники можуть організовувати DDoS-атаки, спрямовані на повну зупинку діяльності компанії.

Ризики, пов'язані із зовнішніми атаками, включають значні фінансові втрати, викрадення персональних даних клієнтів, серйозні репутаційні втрати, а також тривалу зупинку діяльності, що є критичним для нотаріальної контори.

5. Зовнішній порушник (випадковий / ненавмисний)

Остання категорія порушників є специфічною, оскільки особи, що до неї належать, не мають жодного злого умислу. Їхні дії зумовлені випадковістю, цікавістю або незнанням. До таких осіб належать сторонні люди, які знаходять загублений носій інформації (флешку, зовнішній диск), клієнти, які випадково

побачили конфіденційну інформацію на моніторі, або прибиральний персонал, який може ненавмисно пошкодити обладнання. Типові дії включають підключення знайденого носія до власного комп'ютера з метою «подивитися, що там», що може призвести до витоку даних або зараження носія. Також можливе випадкове механічне, магнітне або хімічне пошкодження носія інформації. Інформація, випадково побачена на моніторі, може бути ненавмисно розголошена в розмові з третіми особами.

Таким чином, модель порушника допомагає оцінити не лише технічну вразливість, але й те, хто і з якою ймовірністю може її використати. Для тестового середовища найбільш вірогідними є зовнішній хакер (експлуатація CVE) та ненавмисний внутрішній порушник (помилки конфігурації). Найбільш небезпечним за потенційними наслідками є привілейований порушник, хоча його дії менш імовірні.

2.6 Пріоритезація загроз за допомогою методу аналізу ієрархій (MAI)

Відповідно до результатів моделювання загроз за STRIDE (підрозділ 2.4.2), чотири категорії загроз є найбільш актуальними для досліджуваних інформаційних активів:

• S (Spoofing) – підміна, несанкціонований доступ шляхом підміни облікових даних;

• T (Tampering) – втручання, несанкціонована модифікація даних;

• I (Information Disclosure) – розкриття інформації, порушення конфіденційності;

• D (Denial of Service) – відмова в обслуговуванні, порушення доступності.

Для оцінки впливу кожної загрози на безпеку інформаційних активів обрано чотири критерії, які безпосередньо випливають із властивостей інформаційної безпеки (CIA) та додаткових факторів:

• АУ (слабкість автентифікації) – наскільки легко зловмиснику скомпрометувати облікові дані або видати себе за легітимного користувача. Відповідає загрозі підміни (S).

• КЦ (слабкість контролю цілісності) – наскільки легко змінити, модифікувати або підробити дані без виявлення. Відповідає загрозі втручання (T).

• КД (слабкість контролю доступу) – наскільки легко отримати несанкціонований доступ до конфіденційних даних. Відповідає загрозі розкриття інформації (I).

• ЗС (слабкість забезпечення доступності) – наскільки легко порушити доступність інформаційного активу або сервісу. Відповідає загрозі відмови в обслуговуванні (D).

Ієрархія має три рівні: мета (пріоритизація загроз для інформаційних активів) - критерії (АУ, КЦ, КД, ЗС) → альтернативи (S, T, I, D).

Для заповнення матриць попарних порівнянь залучено чотирьох експертів – фахівців у сфері кібербезпеки, які мають досвід роботи з подібними інформаційними системами. Кожен експерт оцінював відносну важливість критеріїв та альтернатив за шкалою Сааті (1 – однакова важливість, 3 – помірна перевага, 5 – сильна перевага, 7 – дуже сильна перевага, 9 – надзвичайна перевага). Узагальнені матриці наведено на рисунку 2.9.

Матриця попарних порівнянь для критеріїв

Критерій	АУ	КЦ	КД	ЗС	Добуток	ОВ	Вага
АУ	1	0,5	0,33	3	0,495	0,84	0,17
КЦ	2	1	0,5	4	4	1,41	0,29
КД	3	2	1	5	30	2,34	0,47
ЗС	0,33	0,25	0,2	1	0,0165	0,36	0,07
Сума						4,95	

Рисунок 2.9 Матриця попарних порівнянь для критеріїв

Після нормалізації матриці та обчислення власного вектора отримано ваги критеріїв. Розрахункове значення відношення узгодженості (HR = 0,047 < 0,1) свідчить про прийнятну узгодженість експертних оцінок. Тобто, думки різних експертів щодо важливості критеріїв є досить близькими, не суперечать одна одній, і на основі цих оцінок можна робити узагальнені висновки. Іншими словами,

експерти не розійшлися кардинально в своїх судженнях. Таблиця – 2.3 критерії та їх вага.

Критерій	Вага	Пояснення
КД (контроль доступу / конфіденційність)	0,47	Найвищий пріоритет – захист персональних даних та комерційної таємниці
КЦ (контроль цілісності)	0,29	Високий пріоритет – захист цілісності даних та логів
АУ (автентифікація)	0,17	Середній пріоритет – контроль доступу через автентифікацію
ЗС (доступність)	0,07	Найнижчий пріоритет – забезпечення доступності

Отриманий розподіл ваг є логічним: для об’єкта захисту, де обробляються персональні дані та конфіденційна інформація, найважливішим є захист конфіденційності (КД = 0,47). Доступність (ЗС) має найменшу вагу, оскільки більшість інформаційних систем мають резервні канали та не є критичними для життєдіяльності. Результати розрахунків наведено на зображеннях 2.10-2.13. У всіх випадках HR < 0,1, що підтверджує узгодженість оцінок:

1. **Матриця за критерієм АУ (Слабкість автентифікації)**

За критерієм АУ

Альтернатива	Підміна	Втручання	Розкриття	Відмова	Добуток	ОВ	Вага
Підміна	1	3	2	4	24	2,21	0,47
Втручання	0,33	1	0,5	2	0,33	0,76	0,16
Розкриття	0,5	2	1	3	3	1,32	0,28
Відмова	0,25	0,5	0,33	1	0,04125	0,45	0,10
Сума						4,74	1,00

n	4
n*a	4,030954
n*b	4,021103
n*c	4,019998
n*d	5,506688
n*	4,394686
H	0,132891
RH	0,99
HR	0,134233

Рисунок 2.10 Матриця за критерієм АУ

Матриця має відмінну узгодженість (HR = 0,030 < 0,1). Ваги відображають логіку: найбільша загроза за критерієм автентифікації — Підміна (0,48) , найменша — Відмова (0,08) .

2. **Матриця за критерієм КЦ (Слабкість контролю цілісності)**

За критерієм КЦ

Альтернатива	Підміна	Втручання	Розкриття	Відмова	Добуток	ОВ	Вага
Підміна	1	0,33	2	3	1,98	1,19	0,23
Втручання	3	1	4	5	60	2,78	0,55
Розкриття	0,5	0,25	1	2	0,25	0,71	0,14
Відмова	0,33	0,2	0,5	1	0,033	0,43	0,08
Сума						5,10	1,00

n	4
n*a	4,044369
n*b	4,060613
n*c	4,028298
n*d	4,053950
n*	4,046808
H	0,015760
RH	0,99
HR	0,015919

Рисунок 2.11 Матриця за критерієм КЦ

Матриця має прийнятну узгодженість (HR = 0,040 < 0,1). Ваги логічно розподілені: Втручання (0,55) має найвищий пріоритет за критерієм цілісності, що відповідає суті загрози. Матриця має відмінну узгодженість (HR = 0,027 < 0,1). Ваги чітко ранжують загрози: Розкриття (0,54) — найвищий ризик порушення конфіденційності.

3. **Матриця за критерієм КД (Слабкість контролю доступу)**

За критерієм КД

Альтернатива	Підміна	Втручання	Розкриття	Відмова	Добуток	ОВ	Вага
Підміна	1	2	0,33	4	2,64	1,27	0,25
Втручання	0,5	1	0,25	3	0,375	0,78	0,15
Розкриття	3	4	1	5	60	2,78	0,54
Відмова	0,25	0,33	0,2	1	0,0165	0,36	0,07
Сума						5,20	1,00

n	4
n*a	4,073037
n*b	4,077580
n*c	4,142553
n*d	4,162758
n*	4,113982
H	0,038378
RH	0,99
HR	0,038765

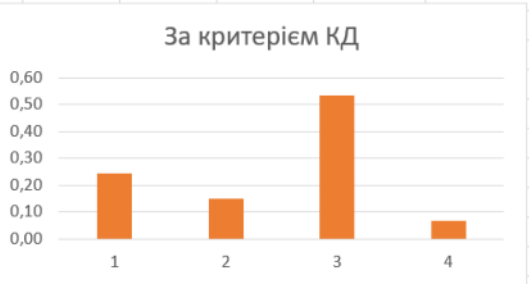
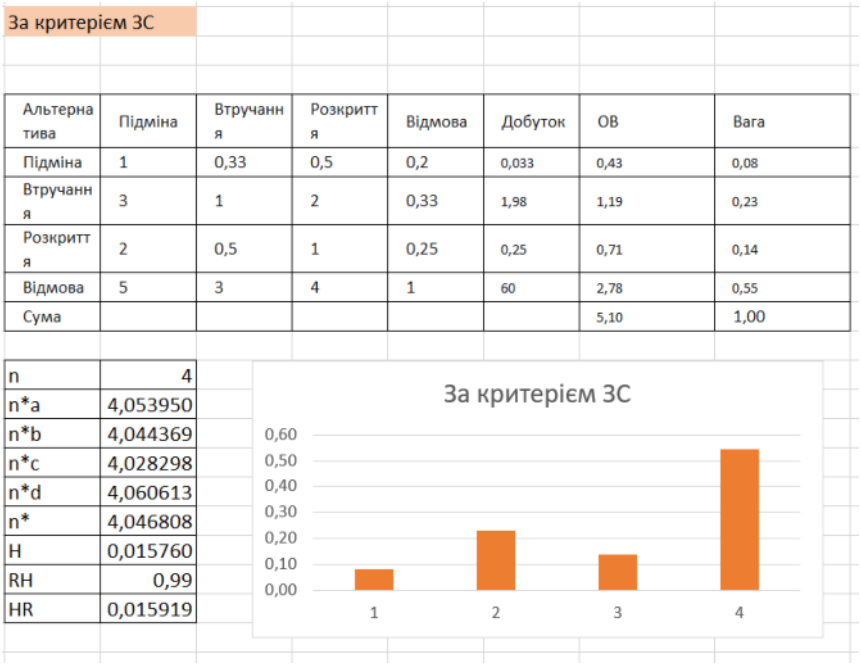


Рисунок 2.12 Матриця за критерієм КД

4. **Матриця за критерієм ЗС (Слабкість забезпечення доступності)**

Матриця має відмінну узгодженість (HR = 0,020 < 0,1). Ваги логічно розподілені: Відмова (0,54) має найвищий пріоритет за критерієм доступності, що повністю відповідає суті загрози DDoS.



За критерієм ЗС

Альтернатива	Підміна	Втручання	Розкриття	Відмова	Добуток	ОВ	Вага
Підміна	1	0,33	0,5	0,2	0,033	0,43	0,08
Втручання	3	1	2	0,33	1,98	1,19	0,23
Розкриття	2	0,5	1	0,25	0,25	0,71	0,14
Відмова	5	3	4	1	60	2,78	0,55
Сума						5,10	1,00

n	4
n*a	4,053950
n*b	4,044369
n*c	4,028298
n*d	4,060613
n*	4,046808
H	0,015760
RH	0,99
HR	0,015919

Рисунок 2.13 Матриця за критерієм ЗС

Підсумкові інтегральні оцінки (глобальні пріоритети) для кожної загрози обчислюються як сума добутків ваги критерію на локальний пріоритет загрози за цим критерієм (2.4):

$$V = \sum_{i=1}^{4} (\omega_{\text{критерію}} \times prio_{\text{загрози}}) \qquad (2.4)$$

Отримані інтегральні оцінки дозволяють ранжувати загрози за ступенем загального ризику. Таблиця 2.4 – загрози та їх глобальний пріоритет.

Ранг	Загроза	Глобальний пріоритет (Vi)	Відсоток
1	**Розкриття (I)**	**0,3518**	**35,2%**
2	**Втручання (T)**	**0,2733**	**27,3%**
3	**Підміна (S)**	**0,2697**	**26,97%**
4	**Відмова (D)**	**0,1116**	**11,6%**

Сума всіх Vi = 1,000, що підтверджує коректність розрахунків. Продемонстровано на рисунку 2.14.

Загроза	АУ (0,17)	КЦ (0,29)	КД (0,47)	ЗС (0,07)	Vi
Підміна	0,47	0,23	0,25	0,08	0,2697
Втручання	0,16	0,55	0,15	0,23	0,2733
Розкриття	0,28	0,14	0,54	0,14	0,3518
Відмова	0,10	0,08	0,07	0,55	0,1116
Вага	0,17	0,29	0,47	0,07	

Рисунок 2.14 Узагальнююча оцінка

Отримані інтегральні оцінки дозволяють зробити такі висновки:

1. Найбільш критичною загрозою для інформаційних активів є розкриття інформації (Information Disclosure) з глобальним пріоритетом 35,2%. Це обумовлено наявністю персональних даних, конфігураційних файлів та іншої конфіденційної інформації на веб- та файловому серверах. Порушення конфіденційності матиме не лише фінансові та репутаційні наслідки, а й юридичні (порушення законодавства про захист персональних даних).

2. Втручання (Tampering) та підміна (Spoofing) мають майже однаковий рівень критичності (≈27% та ≈27%). Це вимагає комплексного підходу до захисту цілісності даних та автентифікації. Вразливість CVE-2021-44790 (Apache RCE) є яскравим прикладом загрози втручання, а відсутність MFA та вразливість перерахування користувачів у SSH – прикладом загрози підміни.

3. Відмова в обслуговуванні (Denial of Service) має найнижчий пріоритет (11,6%). Це дозволяє відкласти масштабні інвестиції в DDoS-захист на користь

більш критичних напрямків. Однак базові заходи (резервне копіювання, антивірусний захист) все одно необхідні.

4. Усі експертні оцінки є узгодженими (HR < 0,1 для всіх матриць), що підтверджує достовірність отриманих результатів.

2.7 Поняття CVSS

CVSS (Common Vulnerability Scoring System) — це загальноприйнятий стандарт для визначення ступеня небезпеки вразливостей в інформаційних системах. Його розробили у складі Консорціуму з безпеки комп'ютерних систем (Computer Security Division) Американського національного інституту стандартів і технологій (NIST). CVSS вважається надійним інструментом оцінювання ризиків, який допомагає визначити, наскільки критичною є та чи інша вразливість. Система CVSS передбачає три ключові групи показників: базові, часові та контекстуальні. Кожна з них відображає різні сторони вразливості, і разом вони дають змогу розрахувати підсумкову оцінку ризику.

Основні складники CVSS:

- Базова метрика (Base Metric) – первинний рівень оцінки. Вона враховує, наскільки легко скористатися вразливістю, який вплив вона може мати та якою мірою вона є доступною (виявленою).
- Часова метрика (Temporal Metric) – оцінює чинники, що змінюються з часом, наприклад, давність виявлення вразливості чи наявність готових експлойтів.
- Контекстуальна метрика (Environmental Metric) – дає змогу врахувати конкретне середовище експлуатації вразливого ПЗ: цінність системи для організації, наявні захисні механізми тощо.

Кожна з трьох метрик поділяється на детальні підметрики, які конкретизують різні аспекти вразливості. Наприклад, базова метрика містить такі підпоказники, як доступність (Availability), власне характеристика вразливості (Vulnerability), вплив (Impact) та інші. У цьому дослідженні основний акцент зроблено на базовій метриці, оскільки вона є найбільш універсальною і дозволяє порівнювати вразливості між собою незалежно від середовища використання. Базова оцінка

(Base Score) розраховується за формулою, що поєднує оцінку експлуатаційності (Exploitability) та оцінку впливу (Impact). Для CVSS версії 3.1 ці складові визначаються наступним чином.

Оцінка експлуатаційності обчислюється за формулою (2.5):

$$Exploitability = 8{,}22 \times AV \times AC \times PR \times UI \quad (2.5)$$

Де:

- AV – вектор атаки (мережа – 0,85; суміжна мережа – 0,62; локальний – 0,55; фізичний – 0,20)
- AC – складність атаки (низька – 0,77; висока – 0,44)
- PR – необхідні привілеї (відсутні – 0,85; низькі – 0,62; високі – 0,27)
- UI – взаємодія з користувачем (відсутня – 0,85; потрібна – 0,62)

Оцінка впливу визначається за формулою (2.6):

$$Impact = 1 - (1 - C) \times (1 - I) \times (1 - A) \quad (2.6)$$

Де C,I,A – вплив на конфіденційність, цілісніть та доступність відповідно(високий -0,56; низький -0,22; відсутній -0). Для обліку випадків, коли вразливість знаходиться в окремій області безпеки (Scope), використовується функція корекції впливу f(impact). Якщо область застосування буде змінена на S=0, то (2.7):

$$f(impact) = 6{,}42 \times Impact \quad (2.7)$$

Якщо ж область застосування змінена S=1, то (2.8):

$$f(impact) = 7{,}52 \times (Impact - 0{,}029) - 3{,}25 \times (Impact - 0{,}02)^{15} \quad (2.8)$$

Таким чином, загальна оцінка обчислюється так формула (2.9):

$$BaseScore = \frac{ceil(\min(f(Impact) + Exploitability, 10) \times 10}{10} \quad (2.9)$$

Де ceil – округлення вгору до найближчого цілого, а min обмежує результат максимального значення 10. Згідно з отриманим базовим балом вразливість відносять до одного з п’яти рівнів критичності:

- None (0) – впливу немає;
- Low (0,1 – 3,9) – низький ризик;
- Medium (4,0 – 6,9) – середній ризик;

• High (7,0 – 8,9) – високий ризик;

• Critical (9,0 – 10,0) – критичний ризик.

Для кожної вразливості, виявленої в тестовому середовищі, було визначено базові метрики. Для вразливостей, що мають ідентифікатор CVE, значення метрик та підсумковий бал взято з бази даних NVD. Для вразливостей без CVE (наприклад, TCP timestamps, слабкі шифри) розрахунок виконано за наведеними вище формулами на основі рекомендацій OpenVAS та експертної оцінки. Результати зведено в таблицю 2.5.

Вразливість (CVE / опис)	Уражений актив	AV	AC	PR	UI	C	I	A	Base Score	Рівень
CVE-2021-44790 (Apache RCE)	192.168.56.105	0,85	0,77	0,85	0,85	0,56	0,56	0,56	**9,8**	Critical
CVE-2020-14145 (OpenSSH user enum)	192.168.56.105	0,85	0,77	0,85	0,85	0,22	0,22	0	**5,3**	Medium
FTP Anonymous Access	192.168.56.110	0,85	0,77	0,85	0,85	0,22	0	0	**5**	Medium
SSH weak MAC algorithms	192.168.56.105	0,85	0,44	0,85	0,85	0	0	0	**4,3**	Medium
SSL/TLS Weak Ciphers	192.168.56.110	0,85	0,44	0,85	0,85	0,22	0	0	**4,8**	Medium

Продовження таблиці 2.5

SSL/TLS Certificate Expired	192.168.56.110	0,85	0,77	0,85	0,85	0,22	0	0	**5**	Medium
SMB Signing Not Required	192.168.56.120	0,62	0,77	0,62	0,85	0	0,22	0	**4**	Medium
RDP Weak Encryption	192.168.56.120	0,85	0,44	0,62	0,85	0,22	0	0	**4**	Medium
TCP timestamps	192.168.56.105	0,85	0,77	0,85	0,85	0	0	0	**2,6**	Low

З таблиці видно, що найбільш критичною є вразливість CVE-2021-44790 (CVSS 9,8) на веб-сервері. Її експлуатація не потребує автентифікації, має низьку складність атаки та призводить до повної компрометації системи. Саме вона отримує найвищий пріоритет під час планування усунення. Вразливості з CVE-2020-14145, анонімним FTP, простроченим сертифікатом та слабкими шифрами мають середній рівень (4,0–5,3). Вони не є критичними, але їх слід усувати в плановому порядку. Інформаційна вразливість TCP timestamps має низький бал (2,6) і може бути прийнята або вимкнена за можливості. За результатами CVSS визначено чітку пріоритетність заходів захисту, яка буде врахована в третьому розділі роботи.

Висновки до розділу 2

У другому розділі проаналізовані підходи, моделі та методи ідентифікації вразливостей інформаційних активів, на практиці було перевірено, як виявляються фактори впливу на вразливості в інформаційних активах. Актуальність дослідження зумовлена стрімким зростанням кількості кіберзагроз та необхідністю своєчасного виявлення слабких місць в інформаційних системах для запобігання можливим атакам.

Для цього взяли тестове середовище – три віртуальні машини з різними ролями. За допомогою Nmap знайшли три активні хости: веб-сервер (192.168.56.105), файловий сервер (192.168.56.110) та робочу станцію (192.168.56.120). Складено реєстр інформаційних активів з попередньою оцінкою критичності, що є першим і необхідним етапом будь-якого процесу управління вразливостями.

Проведено експертне ранжування восьми категорій інформаційних активів. Коефіцієнт конкордації Кендалла становить W = 0,95, що свідчить про високу узгодженість думок експертів. Найвищу цінність отримали інформаційні активи (дані), що підтверджує пріоритетність захисту саме інформаційних ресурсів. Потім було зіставлено що, знайдені вразливості з категоріями загроз за методикою STRIDE. Щоб визначити, які загрози найважливіші, використали метод аналізу ієрархій. За даними результатами, було отримано що найбільшої уваги потребує розкриття інформації – 35% загальної ваги. Далі йдуть втручання (27%) та підміна (27%), а найменше – відмова в обслуговуванні (12%). Це означає, що в першу чергу треба захищати дані, а не доступність.

Для кожної виявленої вразливості визначено базову оцінку CVSS, що дозволило об’єктивно порівняти їх критичність.

Таким чином, у другому розділі не лише виявлено технічні вразливості, але й проаналізовано фактори, що на них впливають цінність активів, тип загрози, ймовірний порушник. Це дає основу для конкретних дій, які будуть написані в третьому розділі.

3. ПРОГРАМНА РЕАЛІЗАЦІЯ ФУНКЦІОНАЛЬНИХ КОМПОНЕНТІВ ДЛЯ ІДЕНТИФІКАЦІЇ ВРАЗЛИВОСТЕЙ

3.1 Удосконалення процесу ідентифікації вразливостей та оцінки вразливостей

У другому розділі дипломної роботи було виконано ідентифікацію інформаційних активів тестового середовища за допомогою інструментів Nmap та OpenVAS, а також проведено їх аналіз з використанням експертних методів – ранжування активів за цінністю (коефіцієнт конкордації, критерій Пірсона) та методу аналізу ієрархій (MAI) для пріоритизації загроз. Ці розрахунки виконувалися вручну, що потребувало значного часу та створювало ризик помилок через округлення або неправильне обчислення поправок на зв’язані ранги.

Для усунення зазначених недоліків та підвищення ефективності процесу ідентифікації вразливостей було розроблено програмне забезпечення на мові C#, яке автоматизує ключові обчислювальні етапи. Програма безпосередньо стосується ідентифікації активів, оскільки вона:

- дозволяє швидко отримувати кількісну оцінку цінності інформаційних активів на основі експертних рангів, що є невід'ємною частиною ідентифікації найбільш критичних активів для подальшого захисту;
- забезпечує пріоритизацію загроз (підміна, втручання, розкриття, відмова в обслуговуванні) відносно цих активів, що дозволяє визначити напрями першочергового захисту.

Основні удосконалення, реалізовані в програмі:

Автоматична нормалізація зв’язаних рангів — програма самостійно виявляє групи однакових оцінок кожного експерта та замінює їх середнім арифметичним відповідних позицій, що унеможливлює помилки при ручному усередненні.

- Розрахунок коефіцієнта конкордації з урахуванням поправок — реалізовано формулу (2.1) з автоматичним обчисленням ΣTi для зв’язаних рангів, що забезпечує коректність навіть за наявності однакових оцінок.

- Автоматичне порівняння критерію Пірсона з табличним значенням — програма не лише обчислює χ^2, але й виводить висновок про статистичну значущість узгодженості.
- Реалізація методу аналізу ієрархій (MAI) для пріоритизації загроз — додано можливість введення матриць попарних порівнянь для критеріїв та альтернатив, обчислення локальних пріоритетів методом геометричного середнього, перевірку узгодженості (HR) та розрахунок глобальних пріоритетів.
- Модульна структура — програма поділена на логічні модулі (введення, обробка, розрахунки, виведення), що дозволяє легко модифікувати або розширювати функціональність (наприклад, додати нові методи оцінки ризиків).

У порівнянні з аналогічними програмними реалізаціями (наприклад, Excel-шаблони або онлайн-калькулятори), розроблена програма забезпечує повний цикл обробки експертних даних без необхідності встановлення додаткових розширень і дозволяє зберігати результати у текстовому вигляді.

3.2 Структура модулів

Програма складається з чотирьох основних модулів, кожен з яких відповідає за певний етап обробки даних.

1. Модуль введення даних (Input Module)

Призначений для інтерактивного введення експертних оцінок та даних для MAI. Забезпечує валідацію введених значень (перевірка діапазону, коректності формату) та автоматичне заповнення нижнього трикутника матриць попарних порівнянь оберненими величинами. У разі виявлення помилки (наприклад, введення нульового значення) модуль пропонує повторити введення.

2. Модуль визначення рангу активу на основі методу ранжування активів (Ranking Module)

Реалізує алгоритми, описані в підрозділі 2.2:

- Нормалізація зв’язаних рангів для кожного експерта (функція NormalizeRanks).
- Обчислення сум рангів Rj, середньої суми T*, відхилень Dj та суми квадратів відхилень ΣDj^2.
- Розрахунок поправок на зв’язані ранги Ti для кожного експерта та їх суми ΣTi.
- Обчислення коефіцієнта конкордації W та критерію Пірсона χ^2.
- Порівняння χ^2 з табличним значенням (для $\alpha=0,05$) та виведення висновку про узгодженість.
- Перерахунок цінності активів у шкалу 7–10 за формулою (2.3).
- Виведення результатів у вигляді таблиці.

3. Модуль визначення пріоритетів на основі методу аналізу ієрархій (AHP Module)

Реалізує метод, описаний в підрозділі 2.6:

- Введення кількості критеріїв та їх назв.
- Введення матриці попарних порівнянь для критеріїв (тільки верхній трикутник).
- Введення кількості альтернатив (загроз) та їх назв.
- Для кожного критерію – введення матриці порівняння альтернатив.
- Обчислення векторів пріоритетів методом геометричного середнього.
- Розрахунок максимального власного значення λ_max, індексу узгодженості CI та відношення узгодженості HR.
- Перевірка умови HR < 0,1 (прийнятна узгодженість).
- Розрахунок глобальних пріоритетів альтернатив як зваженої суми локальних пріоритетів.
- Виведення глобальних пріоритетів (сума має дорівнювати 1).

4. Модуль виведення результатів (Output Module)

Відповідає за форматування та відображення результатів у консолі. Забезпечує виведення таблиць з відформатованими числами (з точністю до 1–2 знаків після

коми), а також можливість збереження результатів у текстовий файл (за вибором користувача).

Взаємодія модулів організована послідовно: після запуску програми користувач обирає режим (1 – ранжування активів, 2 – MAI). Відповідний модуль викликає функції введення, виконує розрахунки та передає результати у модуль виведення. Така архітектура забезпечує легке розширення (наприклад, додавання режиму оцінки за CVSS) без зміни існуючого коду.

3.3 Алгоритм функціонування

Алгоритм — це чітко прописана інструкція, за якою комп'ютер (або людина) виконує обчислення. На вхід він отримує дані (їх ще називають аргументами або входом алгоритму), а далі крок за кроком іде до потрібного результату. Таким чином - це логіка розв’язання задачі з усіма формулами, умовами та перевірками, щоб результатам можна було довіряти. Як саме записати алгоритм — залежить від методу яким користуєтесь. Способів опису алгоритмів багато: можна словами, можна таблицею, псевдокодом або малюнком. Для комп’ютерних програм найзручніший спосіб — намалювати блок-схему, тобто розбити процес на окремі блоки-дії й показати стрілками, що за чим іде.

Типи блоків, що використовуються:

- Термінатор (Овал) — позначає початок або кінець алгоритму.
- Введення/виведення (Паралелограм) — використовується для введення даних з клавіатури або виведення результатів на екран.
- Процес (Прямокутник) — означає виконання дії, обчислення, присвоєння.
- Рішення (Ромб) — слугує для перевірки умови (так/ні).
- Стрілка (Лінія зі стрілкою) — показує напрямок переходу між блоками.

1. Алгоритм роботи головного меню

Перша блок-схема (3.1) ілюструє роботу головного циклу програми, який забезпечує взаємодію з користувачем та вибір режиму роботи.

Крок 1. Початок.

Блок-схема починається з термінатора «Початок», який позначає старт виконання програми.

Крок 2. Виведення заголовка та меню.

Паралелограмом зображено виведення на екран інформаційного заголовка та рядків меню з трьома варіантами: «1 – Ранжування активів», «2 – Метод аналізу ієрархій», «0 – Вихід».

Крок 3. Введення вибору користувача.

Наступним паралелограмом виконується зчитування з клавіатури значення змінної choice.

Крок 4. Перевірка першого варіанту.

Ромбом перевіряється умова choice == "1". Якщо умова істинна, стрілка з позначкою «Так» веде до блоку виклику підпрограми ранжування активів. Якщо умова хибна, стрілка «Ні» спрямовується до наступного ромба.

Крок 5. Виклик підпрограми ранжування активів.

Блок виклику підпрограми (прямокутник з подвійними вертикальними лініями) містить текст «RankingModule.PerformRanking()». Після завершення виконання цього методу стрілка повертається до кроку 2 (початок циклу меню).

Крок 6. Перевірка другого варіанту.

Черговий ромб перевіряє умову choice == "2". У разі істинності виконується перехід до блоку виклику підпрограми методу аналізу ієрархій; при хибності – до наступного ромба.

Крок 7. Виклик підпрограми MAI.

Аналогічний блок виклику містить текст «AhpModule.PerformAHP()». Після завершення роботи цього методу стрілка також повертається до кроку 2.

Крок 8. Перевірка виходу з програми.

Ромбом перевіряється умова choice == "0". Якщо вона істинна, виконується перехід до кроку 9 (завершення роботи). Якщо хибна – до кроку 10.

Крок 9. Завершення програми.

Термінатор «Кінець» позначає зупинку виконання програми.

Крок 10. Повідомлення про невірний вибір.

Прямокутником зображено виведення текстового повідомлення «Невірний вибір. Спробуйте ще раз». Після цього стрілка повертається до кроку 2, надаючи користувачеві можливість зробити новий вибір.

Така структура забезпечує нескінченне повторення меню доти, доки користувач не обере пункт «0».

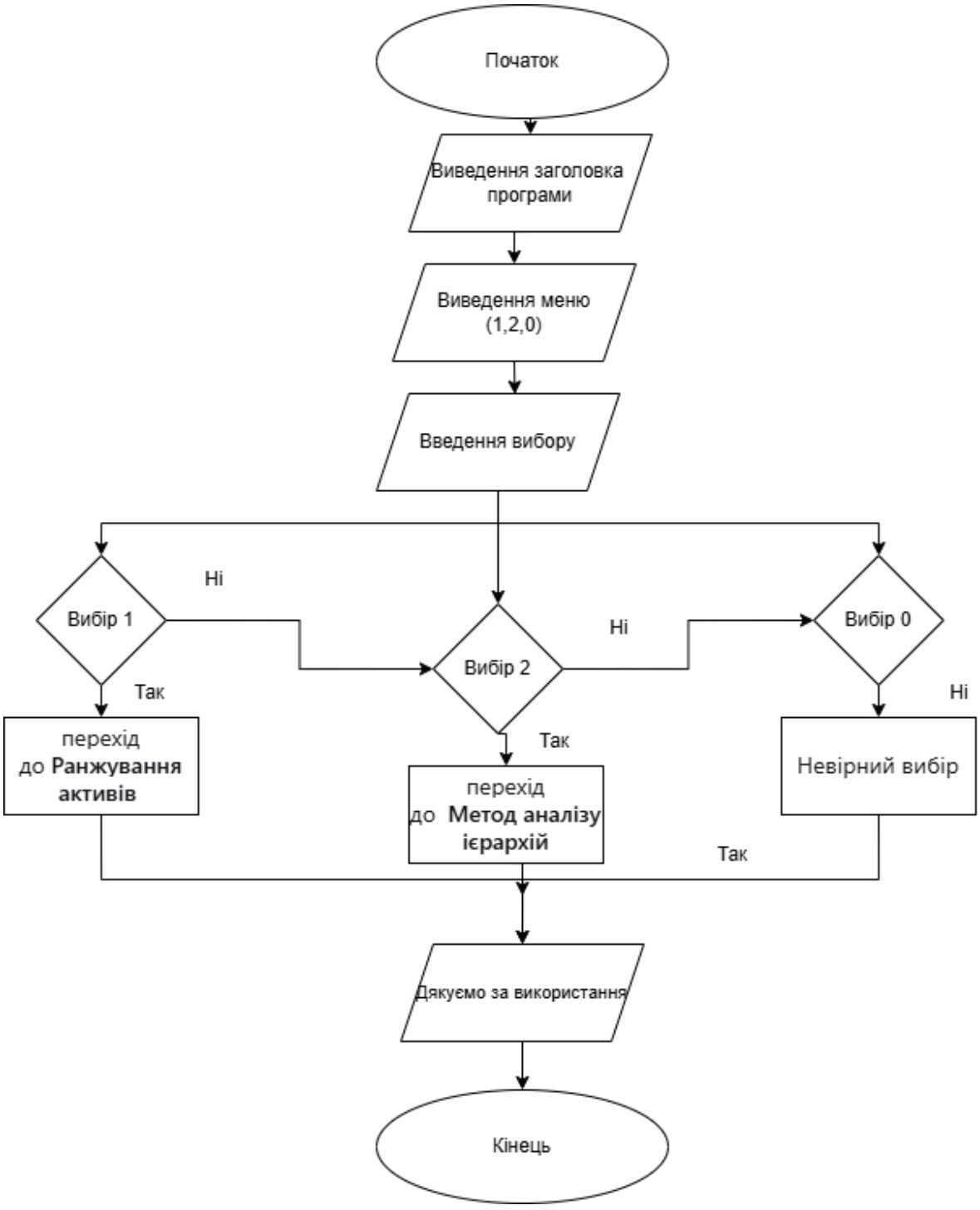


Рисунок 3.1 Алгоритм роботи головного меню

2. Алгоритм обчислення методу ранжування активів (RankingModule.PerformRanking)

Друга блок-схема (3.2) деталізує алгоритм обробки експертних оцінок: нормалізацію зв’язаних рангів, обчислення коефіцієнта конкордації Кендалла, критерію Пірсона та перерахунок цінності активів у шкалу 7–10.

Крок 1. Початок методу.

Термінатор позначає вхід у підпрограму.

Крок 2. Введення кількості експертів та активів.

Паралелограмами виконується введення цілих чисел m (кількість експертів) та n (кількість активів).

Крок 3. Введення матриці рангів.

Наступним паралелограмом користувач вводить матрицю розміром m × n, де кожен рядок містить ранги одного експерта (цілі числа від 1 до n без пропусків).

Крок 4. Ініціалізація змінних.

Прямокутником обнуляються масив сум рангів Rj[1..n] та змінна sumTi (сума поправок на зв’язані ранги).

Крок 5. Встановлення початкового значення лічильника.

Змінній i присвоюється значення 1.

Крок 6. Умова продовження циклу.

Ромб перевіряє умову $i \le m$. Якщо вона істинна, виконується перехід до тіла циклу; якщо хибна – до обчислення сум рангів після циклу.

Крок 7. Нормалізація зв’язаних рангів.

Викликається функція NormalizeRanks для поточного рядка (експерта). Вона замінює однакові ранги на середнє арифметичне відповідних позицій.

Крок 8. Обчислення поправки.

Обчислюється $T\_i = \Sigma(t^3 - t)/12$ для груп однакових рангів, після чого виконується sumTi = sumTi + T_i.

Крок 9. Збільшення лічильника.

i збільшується на 1, і стрілка повертається до кроку 6 (замикання циклу).

Крок 10. Обчислення сум рангів.

Після виходу з циклу для кожного активу j обчислюється сума нормалізованих рангів Rj[j].

Крок 11. Розрахунок середньої суми.

За формулою $T^* = m \cdot (n+1) / 2$ знаходиться теоретичне середнє значення суми рангів.

Крок 12. Обчислення відхилень та суми квадратів.

Для кожного активу обчислюється $Dj = Rj[j] – T^*$ та Dj^2, після чого знаходиться ΣDj^2.

Крок 13. Обчислення коефіцієнта конкордації.

Розраховується W за формулою (2.1) з урахуванням поправки sumTi.

Крок 14. Обчислення критерію Пірсона.

Виконується розрахунок $\chi^2 = m \cdot (n-1) \cdot W$.

Крок 15. Порівняння з табличним значенням.

Ромб перевіряє, чи $\chi^2 > \chi^2\_табл$ (для $\alpha=0,05$ та ступенів свободи $k=n-1$). Залежно від результату обирається одна з двох гілок.

Кроки 16 та 17. Виведення висновку про узгодженість.

Паралелограмами виводиться повідомлення «Узгодженість є статистично значущою» (якщо умова істинна) або «Узгодженість не є значущою» (якщо хибна). Після цього обидві гілки зливаються.

Крок 18. Обчислення цінності активів.

Прямокутником за формулою (2.3) знаходиться цінність кожного активу у шкалі від 7 до 10.

Крок 19. Виведення результатів.

Паралелограмом на екран виводяться суми рангів Rj, коефіцієнт конкордації W, значення χ^2, висновок про узгодженість та цінність активів.

Крок 20. Кінець методу.

Термінатор позначає завершення роботи підпрограми та повернення до головної програми.

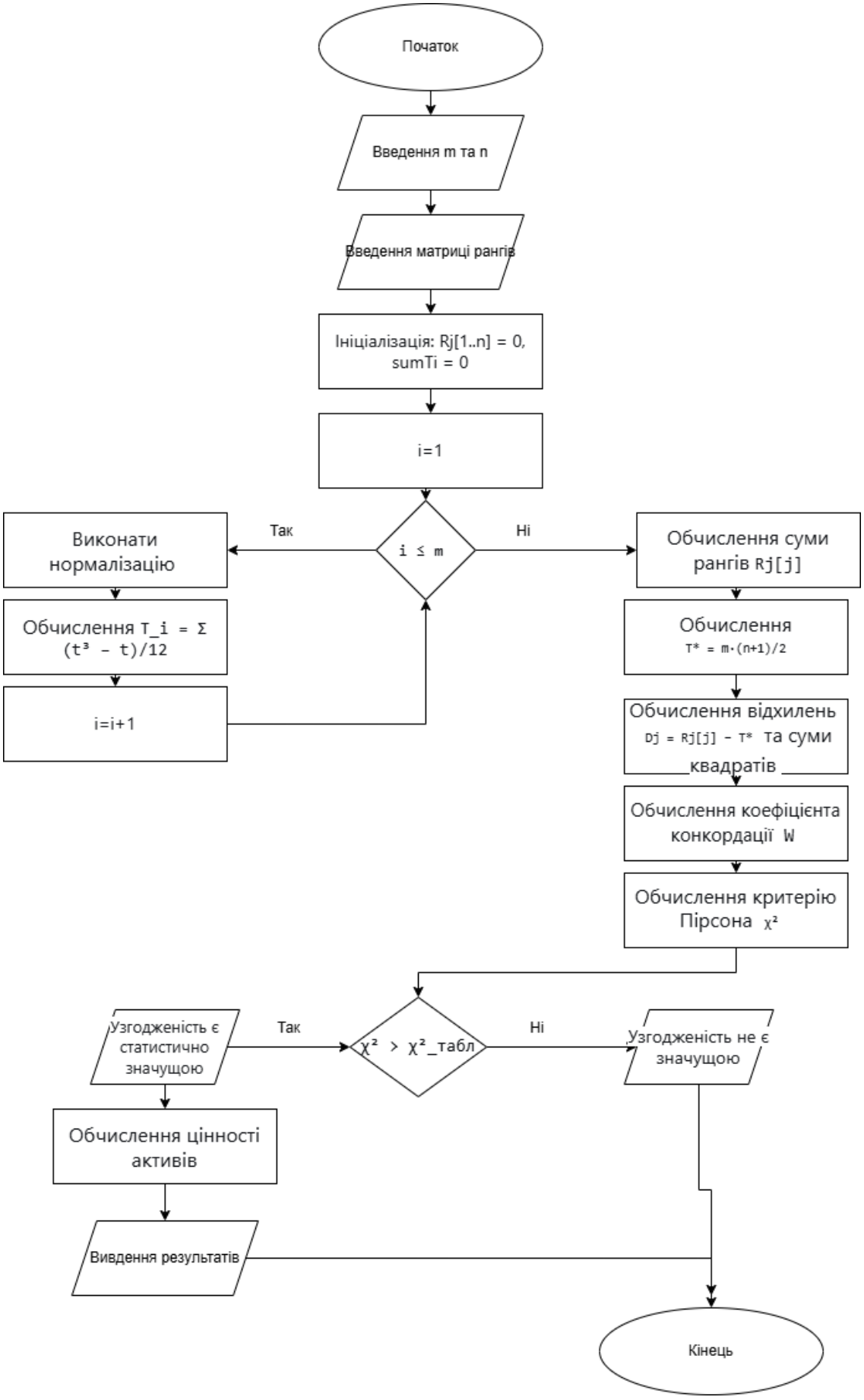


Рисунок 3.2 Алгоритм ранжування активів

3. Алгоритм обчислення MAI (AhpModule.PerformAHP)

Третя блок-схема (3.3) реалізує метод аналізу ієрархій Т. Сааті. Вона включає введення критеріїв, матриць попарних порівнянь, обчислення локальних та глобальних пріоритетів, а також перевірку узгодженості експертних оцінок.

Крок 1. Початок методу.

Термінатор позначає вхід у підпрограму.

Крок 2. Введення кількості критеріїв.

Паралелограмом вводиться ціле число nCrit.

Крок 3. Ініціалізація списку назв критеріїв.

Прямокутником створюється порожній список criteriaNames.

Крок 4. Встановлення лічильника критеріїв i = 1.

Крок 5. Умова циклу введення назв критеріїв.

Ромб перевіряє i ≤ nCrit. Якщо так, виконується тіло циклу; якщо ні – перехід до введення матриці порівнянь.

Крок 6. Введення назви критерію.

Паралелограмом вводиться текстова назва.

Крок 7. Додавання назви до списку.

Прямокутником виконується criteriaNames.Add(...).

Крок 8. Збільшення лічильника i = i + 1, стрілка повертається до кроку 5.

Крок 9. Введення матриці порівнянь критеріїв.

Викликається функція InputMatrix, яка заповнює матрицю попарних порівнянь (верхній трикутник вводиться користувачем, нижній заповнюється оберненими значеннями).

Крок 10. Обчислення ваг критеріїв.

Функція AhpPriority обчислює вектор пріоритетів методом геометричного середнього та нормалізує його.

Крок 11. Обчислення λ_max та HR.

Функцією ConsistencyRatio знаходяться максимальне власне значення та відношення узгодженості.

Крок 12. Перевірка узгодженості критеріїв.

Ромб перевіряє умову HR < 0,1. Якщо істинно, виводиться повідомлення «Узгодженість матриці критеріїв прийнятна», інакше – «…неприйнятна». Після цього обидві гілки зливаються.

Крок 13. Виведення ваг критеріїв.

Паралелограмом виводяться обчислені ваги, значення λ_max та HR.

Крок 14. Введення кількості альтернатив.

Вводиться nAlt.

Кроки 15–20.

Аналогічно до кроків 3–8, але для назв альтернатив (загроз). Створюється список altNames.

Крок 21. Ініціалізація масиву локальних пріоритетів.

Прямокутником створюється двовимірний масив altWeights розміром nAlt × nCrit.

Крок 22. Встановлення лічильника критеріїв для зовнішнього циклу k = 1.

Крок 23. Умова зовнішнього циклу.

Ромб перевіряє k ≤ nCrit. Якщо так – виконується обробка поточного критерію, якщо ні – перехід до обчислення глобальних пріоритетів.

Крок 24. Виведення назви поточного критерію.

Паралелограмом виводиться повідомлення, наприклад «Критерій 1: АУ».

Крок 25. Введення матриці порівнянь альтернатив.

Функцією InputMatrix вводиться квадратна матриця розміром nAlt × nAlt для поточного критерію.

Крок 26. Обчислення локальних ваг.

Функція AhpPriority повертає вектор локальних пріоритетів для альтернатив.

Крок 27. Обчислення λ_max та HR для цієї матриці.

Функцією ConsistencyRatio розраховуються відповідні показники.

Крок 28. Перевірка узгодженості локальної матриці.

Ромб перевіряє HR < 0,1. Залежно від результату виводиться повідомлення про прийнятність чи неприйнятність узгодженості. Гілки зливаються.

Крок 29. Виведення локальних пріоритетів.

Паралелограмом виводяться значення для кожної альтернативи.

Крок 30. Заповнення масиву altWeights.

Прямокутником для всіх альтернатив i в стовпець k записуються обчислені локальні пріоритети.

Крок 31. Збільшення лічильника критеріїв k = k + 1, стрілка повертається до кроку 23.

Крок 32. Обчислення глобальних пріоритетів.

Після завершення зовнішнього циклу для кожної альтернативи обчислюється глобальний пріоритет як зважена сума: global[i] = Σ (criteriaWeights[j] * altWeights[i, j]) для всіх критеріїв j.

Крок 33. Виведення глобальних пріоритетів.

Паралелограмом виводяться отримані значення для кожної альтернативи.

Крок 34. Контрольний вивід суми.

Виводиться сума всіх глобальних пріоритетів, яка має дорівнювати 1 (з точністю до округлення).

Крок 35. Кінець методу.

Термінатор позначає завершення роботи підпрограми та повернення до головної програми.

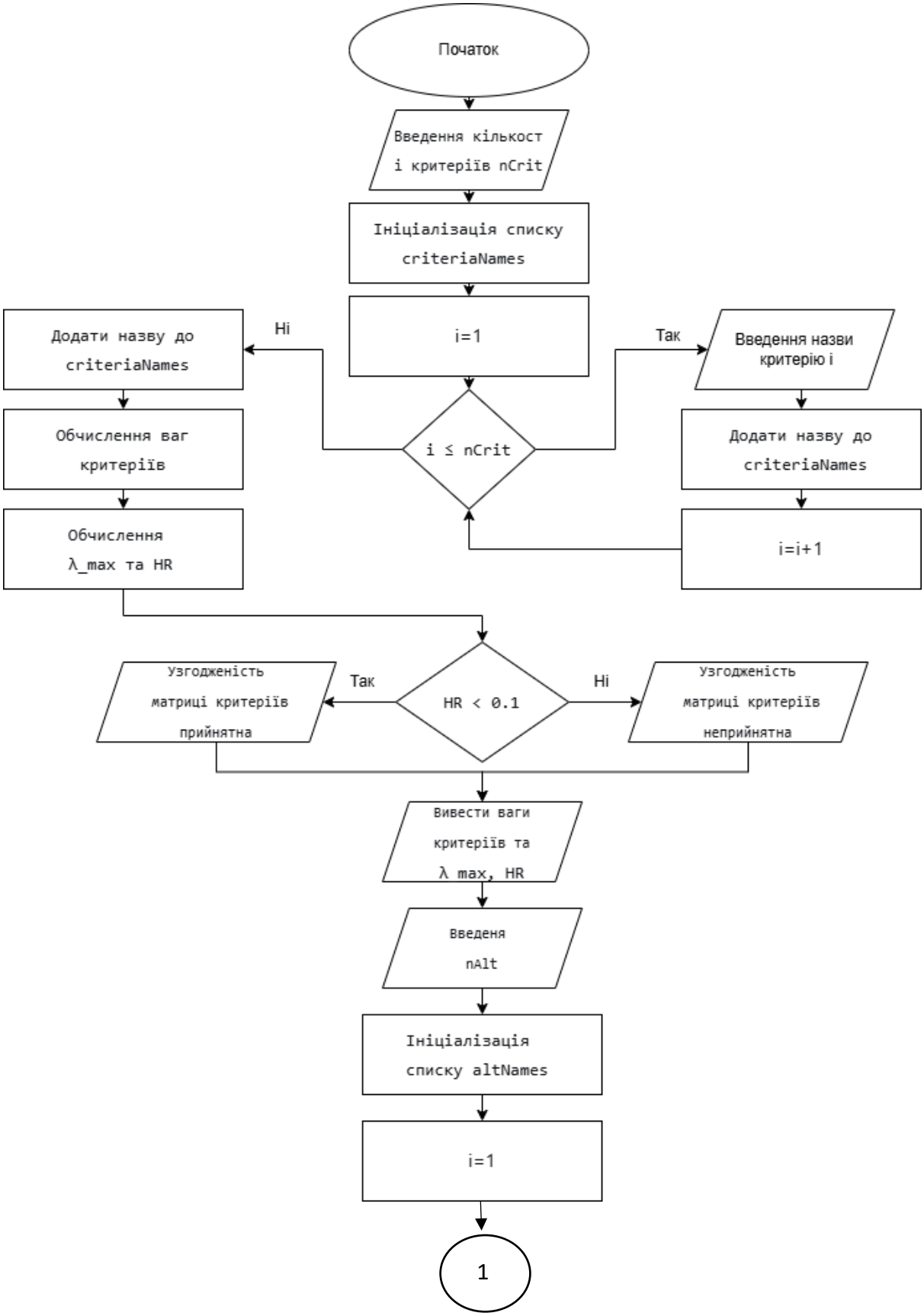
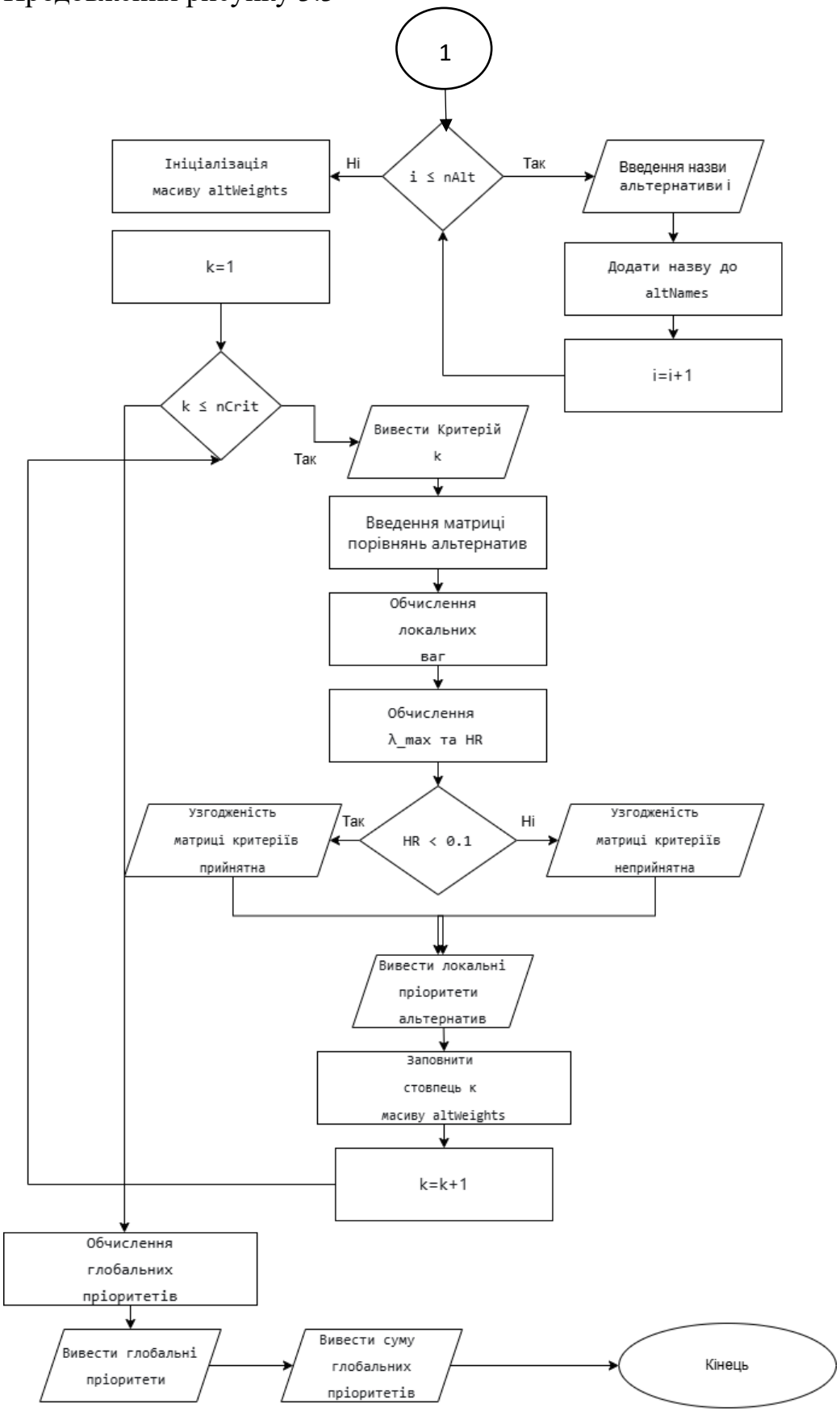


Рисунок 3.3 Алгоритм обчислення локальних та глобальних пріоритетів, перевірки узгодженості експертних оцінок за допомогою MAI

Продовження рисунку 3.3



Завершення рисунку 3.3

3.4 Вибір мови програмування

Програма виконана мовою програмування C# (версія .NET 6.0) в інтегрованому середовищі розробки Microsoft Visual Studio 2022. Microsoft Visual Studio — це лінійка продуктів компанії Microsoft, яка включає інтегроване середовище розробки програмного забезпечення та інші інструментальні засоби. Ці продукти дозволяють розробляти як консольні додатки, так і додатки з графічним інтерфейсом, зокрема з підтримкою технології Windows Forms, а також веб-сайти, веб-додатки, веб-служби як в рідному, так і в керованому кодах для всіх платформ, підтримуваних Windows, Windows Mobile, Windows CE, .NET Framework, Xbox, Windows Phone .NET Compact Framework і Silverlight. Вибір саме цієї мови зумовлений сукупністю технічних та практичних факторів.

Основні причини вибору C#:

1. Зручність роботи з масивами та матрицями.

Алгоритми ранжування активів та методу аналізу ієрархій оперують двовимірними масивами даних (матриці рангів, матриці попарних порівнянь). C# надає прості та зрозумілі засоби для створення, індексації та обробки масивів (double[,], List<List<double>>), а також потужну бібліотеку LINQ, яка дозволяє виконувати складні операції вибору, сортування та групування даних без зайвого коду.

2. Висока точність обчислень.

Для розрахунку коефіцієнта конкордації, власних векторів матриць та перевірки узгодженості необхідна робота з дробовими числами з подвійною точністю. Тип double у C# забезпечує точність до 15-16 значущих цифр, що цілком достатньо для таких обчислень і гарантує мінімальну похибку округлення.

3. Крос-платформеність.

Завдяки .NET Core/.NET 6.0 програма може виконуватися не лише в операційній системі Windows, але й у середовищах Linux та macOS без зміни вихідного коду. Це розширює потенційне коло користувачів та робить розробку більш універсальною.

4. Простота створення консольного інтерфейсу.

Для вирішення поставленого завдання достатньо консольного додатку, оскільки програма не потребує складної графіки або взаємодії з користувачем у реальному часі. C# надає зручні засоби для виведення форматованих таблиць, введення даних з клавіатури та керування потоком виконання. За необхідності в майбутньому функціонал можна легко доповнити графічним інтерфейсом (Windows Forms, WPF, Avalonia) без переписування логіки.

5. Наявність засобів валідації та обробки винятків.

Консольний додаток на C# дозволяє легко реалізувати перевірку коректності введених даних (наприклад, перевірку діапазону рангів, недопущення введення нульових значень) та обробку помилок за допомогою конструкцій try-catch, що підвищує надійність роботи програми.

6. Документація та підт.римка спільноти.

C# є однією з найбільш поширених мов програмування в академічному середовищі та промисловості. Велика кількість прикладів, готових рішень та детальна документація Microsoft спрощують процес розробки та налагодження.

3.5 Тестування роботи програми

Для підтвердження коректності роботи розробленого програмного забезпечення було проведено перевірку розрахунків на різних вхідних даних. Перевірка охоплювала як режим ранжування активів (коефіцієнт конкордації, критерій Пірсона, цінність активів), так і режим методу аналізу ієрархій (MAI). Результати роботи програми порівнювалися з очікуваними значеннями.

Перевірка ранжування активів

На першому етапі перевірки було використано матрицю експертних рангів, яка відповідає (4 експерти, 8 активів). Результати роботи програми показали:

- Суми рангів Rj: 4,0; 9,5; 12,0; 14,5; 20,0; 25,0; 28,5; 30,5;
- Коефіцієнт конкордації W = 0,9490 (висока узгодженість);
- Критерій Пірсона $\chi^2 = 26,57$;
- Табличне значення;
- Програма визначила узгодженість як статично значущу;

- Цінність активів розподілилася від 10,00 до 7,00 відповідно до зменшення сум рангів.

Отримані результати відповідають очікуванням: чим вища узгодженість експертів, тим ближчий коефіцієнт конкордації до одиниці. Програма коректно обчислила всі показники. Продемонстрована рисунках 3.4-3.5.

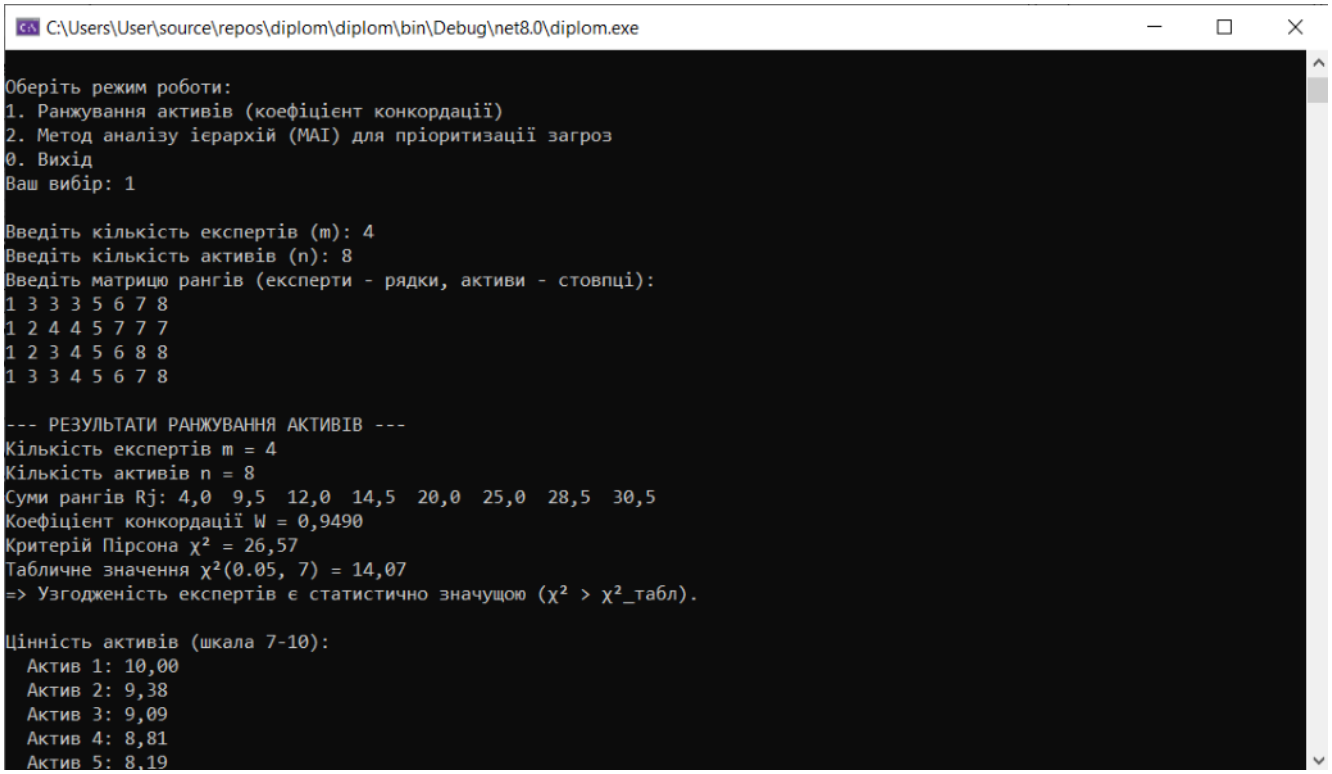


Рисунок 3.4 Ранжування активів початок

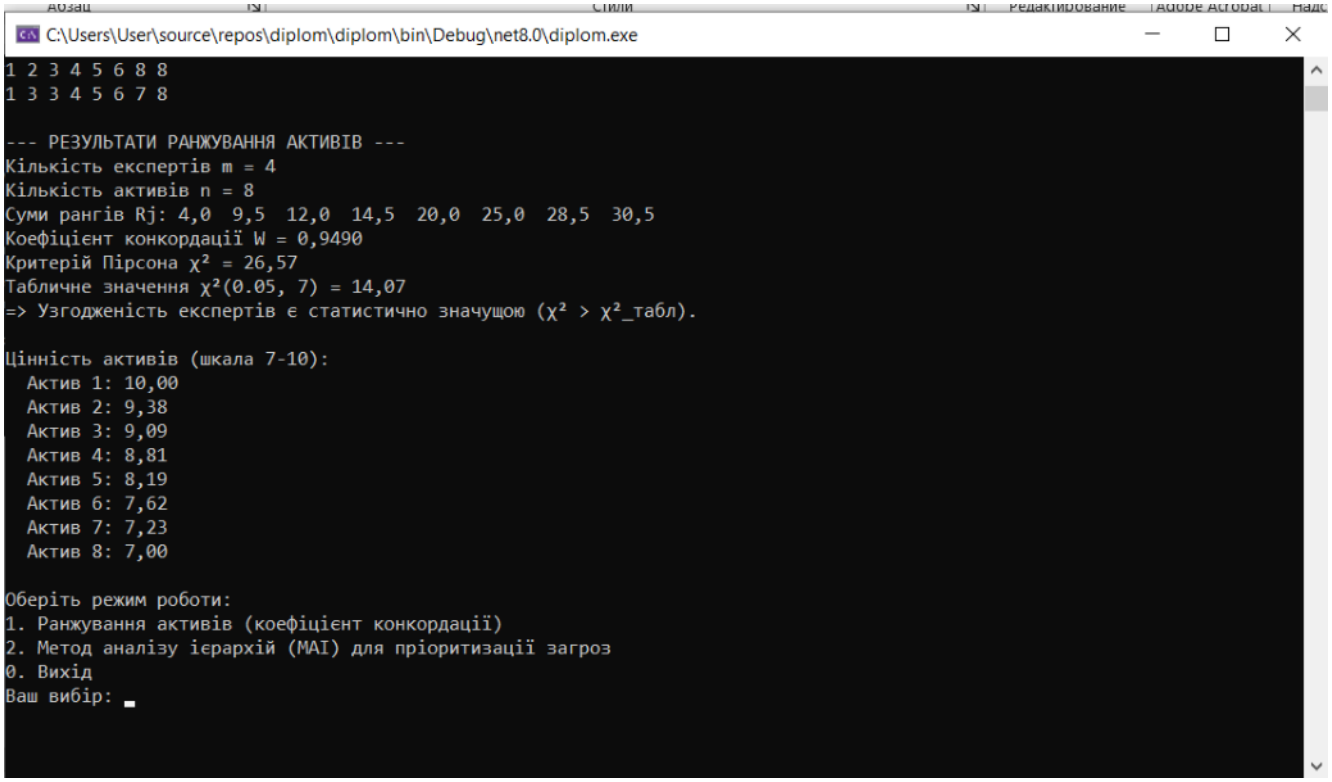


Рисунок 3.5 Ранжування активів кінець

Перевірка режиму методу аналізу ієрархій (MAI)

Для перевірки цього режиму було введено матриці попарних порівнянь згідно з даними, наведеними в підрозділі 2.6 дипломної роботи (рис. 2.9–2.14). Результати роботи програми (рис. 3.6 – рис. 3.10) показали:

- ваги критеріїв: АУ = 0,1692, КЦ = 0,2852, КД = 0,4732, ЗС = 0,0725;
- λ_max = 4,0515, HR = 0,0191 (прийнятна узгодженість, оскільки HR < 0,1);
- для кожного критерію отримано локальні пріоритети альтернатив (загроз S, T, I, D) з відповідними значеннями HR (всі менше 0,1);
- глобальні пріоритети загроз: S = 0,2676; T = 0,2706; I = 0,3496; D = 0,1122;
- сума глобальних пріоритетів = 1,0000.

Отримані значення практично ідентичні тим, що були отримані вручну в розділі 2. Незначні відхилення (на рівні сотих часток відсотка) пояснюються округленням при введенні даних та машинними обчисленнями.

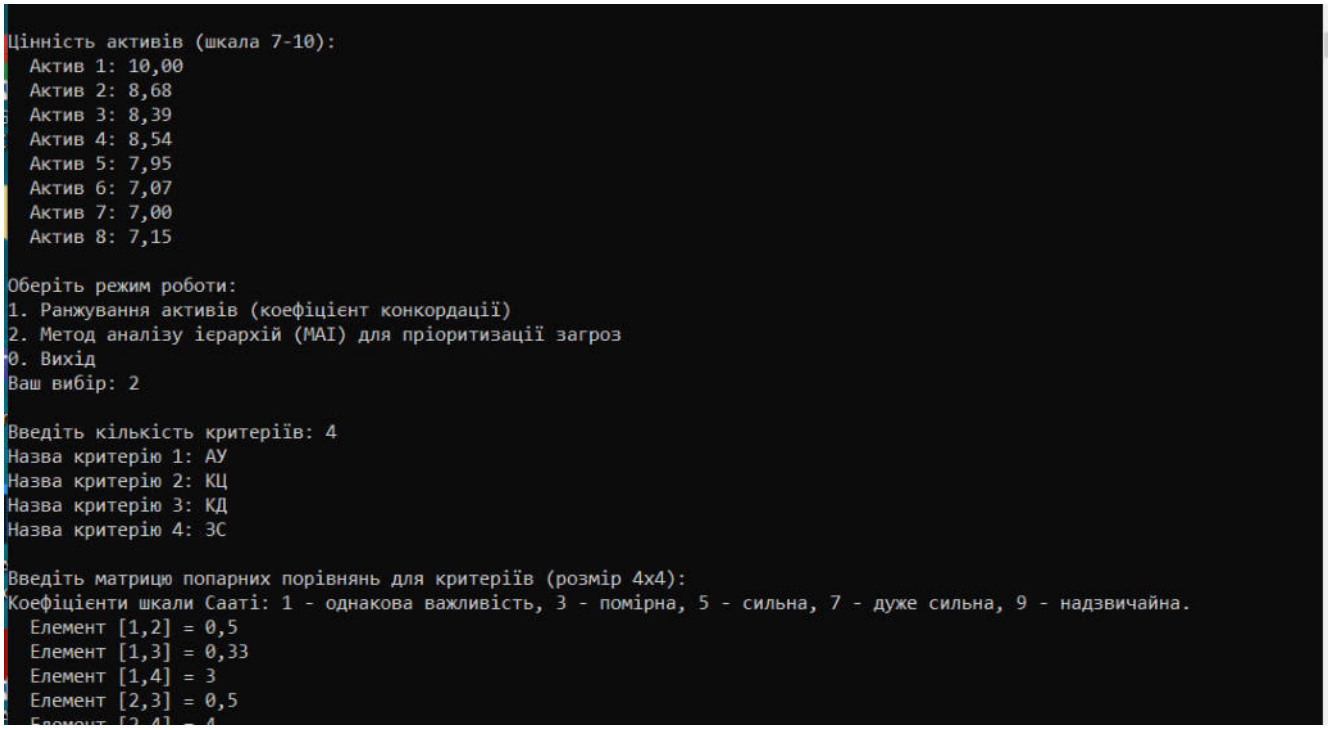


Рисунок 3.6 Користувач вводить критерії, та заповнює матрицю

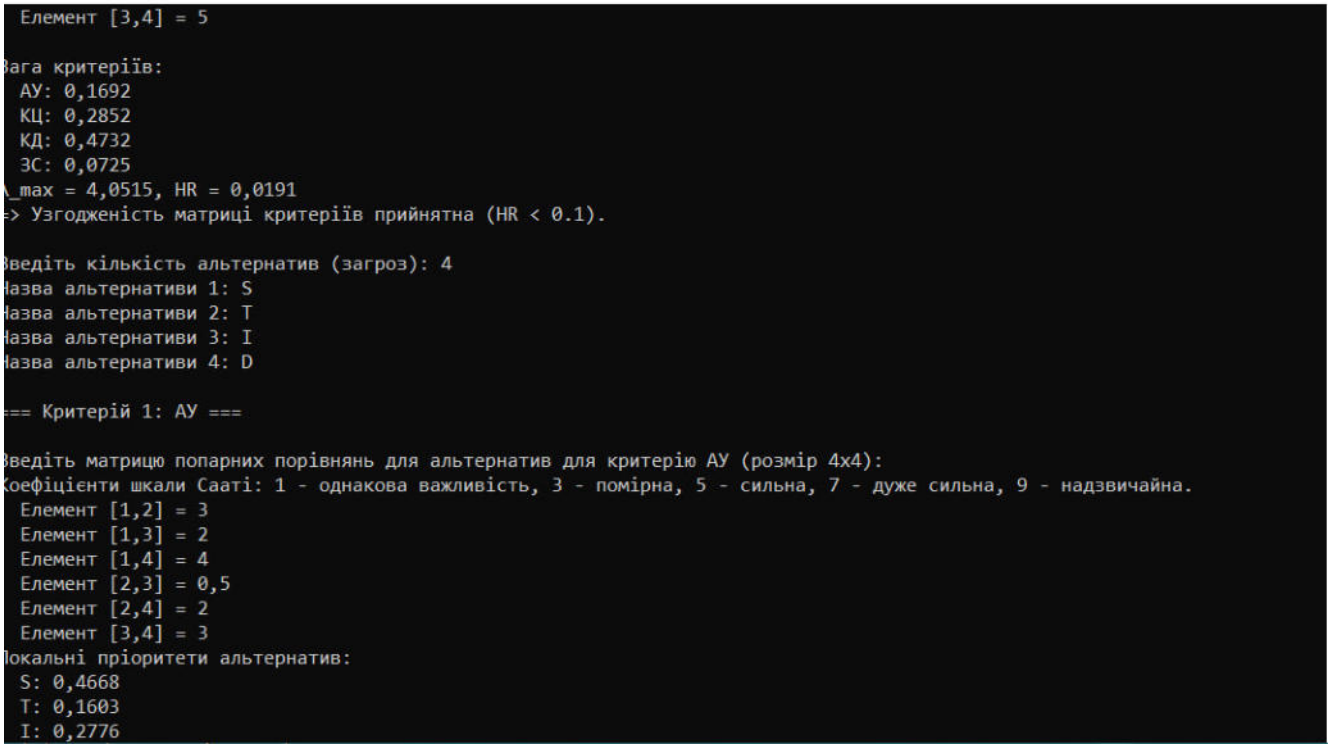


Рисунок 3.7 Оцінка критерію АУ

```
  S: 0,4668
  T: 0,1603
  I: 0,2776
  D: 0,0953
λ_max = 4,0310, HR = 0,0115 (прийнятна)

=== Критерій 2: КЦ ===

Введіть матрицю попарних порівнянь для альтернатив для критерію КЦ (розмір 4x4):
Коефіцієнти шкали Сааті: 1 - однакова важливість, 3 - помірна, 5 - сильна, 7 - дуже сильна, 9 - надзвичайна.
  Елемент [1,2] = 0,333
  Елемент [1,3] = 2
  Елемент [1,4] = 3
  Елемент [2,3] = 4
  Елемент [2,4] = 5
  Елемент [3,4] = 2
Локальні пріоритети альтернатив:
  S: 0,2328
  T: 0,5451
  I: 0,1385
  D: 0,0837
λ_max = 4,0512, HR = 0,0190 (прийнятна)

=== Критерій 3: КД ===

Введіть матрицю попарних порівнянь для альтернатив для критерію КД (розмір 4x4):
Коефіцієнти шкали Сааті: 1 - однакова важливість, 3 - помірна, 5 - сильна, 7 - дуже сильна, 9 - надзвичайна.
  Елемент [1,2] = 2
  Елемент [1,3] = 0,333
  Елемент [1,4] = 4
```

Рисунок 3.8 Оцінка критерію КЦ та КД

```
  Елемент [2,3] = 0,25
  Елемент [2,4] = 3
  Елемент [3,4] = 5
Локальні пріоритети альтернатив:
  S: 0,2455
  T: 0,1504
  I: 0,5350
  D: 0,0691
λ_max = 4,1178, HR = 0,0436 (прийнятна)

=== Критерій 4: ЗС ===

Введіть матрицю попарних порівнянь для альтернатив для критерію ЗС (розмір 4x4):
Коефіцієнти шкали Сааті: 1 - однакова важливість, 3 - помірна, 5 - сильна, 7 - дуже сильна, 9 - надзвичайна.
  Елемент [1,2] = 0,333
  Елемент [1,3] = 0,5
  Елемент [1,4] = 0,2
  Елемент [2,3] = 2
  Елемент [2,4] = 0,333
  Елемент [3,4] = 0,25
Локальні пріоритети альтернатив:
  S: 0,0836
  T: 0,2328
  I: 0,1384
  D: 0,5451
λ_max = 4,0512, HR = 0,0190 (прийнятна)

--- ГЛОБАЛЬНІ ПРІОРИТЕТИ ЗАГРОЗ ---
  S: 0,2676
  T: 0,2706
```

Рисунок 3.9 Оцінка критерію КД та ЗС

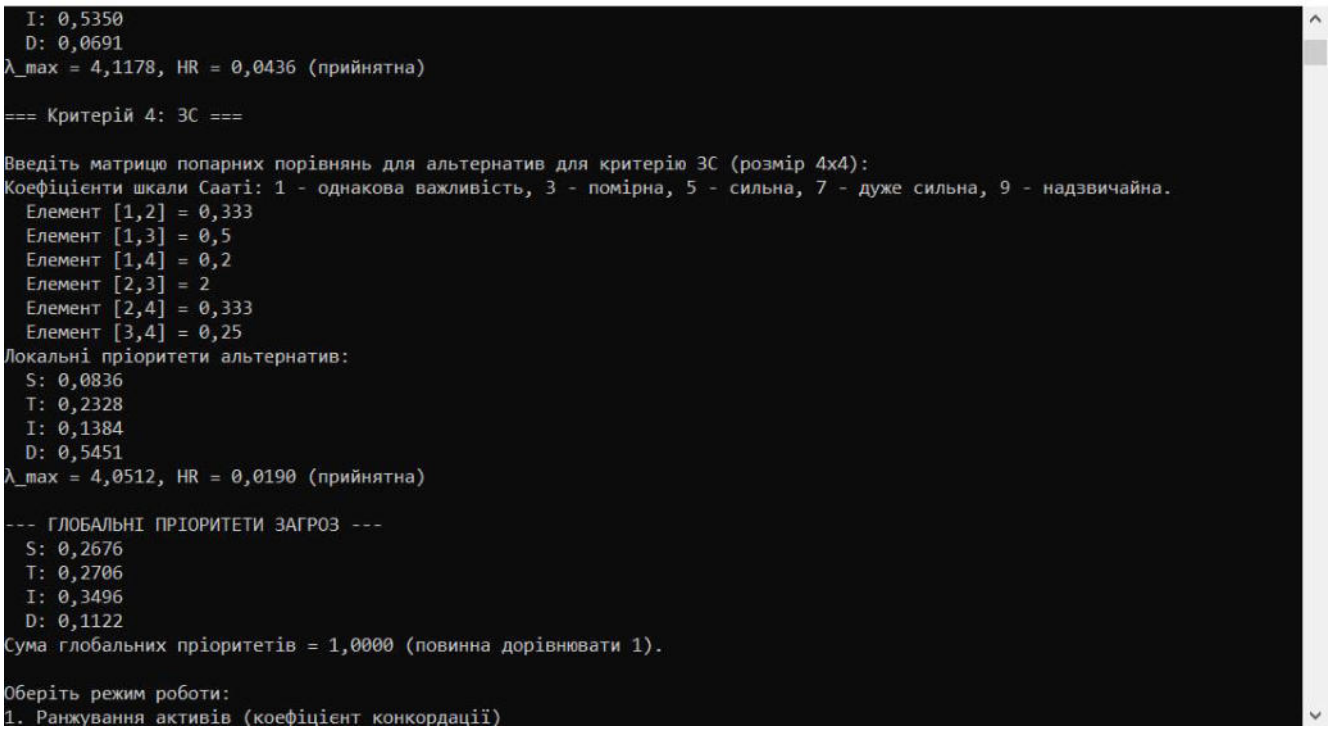


Рисунок 3.10 Глобальні пріоритети загроз

3.6 Перевірка програми на достовірність розрахунків

Для підтвердження коректності роботи розробленої системи я виконую перевірку розрахунків. Перевірка охоплювала обидва режими роботи: ранжування активів (коефіцієнт конкордації, критерій Пірсона, цінність активів) та метод аналізу ієрархій (MAI). Усі обчислення виконувалися з подвійною точністю, а результати порівнювалися з ручними розрахунками за формулами, наведеними в розділі 2.

Перевірка режиму ранжування активів.

Для тестування було використано матрицю експертних рангів, що відповідає рисунку 3.11 (чотири експерти, вісім активів). Вхідні дані:

1	3	3	3	5	6	7	8
1	2	3,5	3,5	5	7	7	7
1	2	3	4	5	6	7,5	7,5
1	2,5	2,5	4	5	6	7	8

Рисунок 3.11 Вхідні дані матриці

Програма виконала нормалізацію зв’язаних рангів, обчислила суми рангів Rj, коефіцієнт конкордації W та критерій Пірсона χ^2. Отримані результати (рис. 3.4-3.5) наведено нижче:

- Суми рангів Rj: 4,0; 9,5; 12,0; 14,5; 20,0; 25,0; 28,5; 30,5;
- Коефіцієнт конкордації W = 0,9490 (висока узгодженість);
- Критерій Пірсона χ^2 = 26,57;

Для перевірки правильності обчислень я виконаю ручний розрахунок для цієї самої матриці. Після нормалізації (усереднення позицій зв’язаних рангів) отримаємо суми рангів, ідентичні тим, що видала програма. Підставивши їх у формулу (2.1), маємо:

$$W = \frac{12 \times \sum Dj^2}{m^2 \times (n^3 - n) - m \times \sum Ti}$$

Для наведених сум рангів Rj обчислено ΣDj^2 = 636, ΣTi = 6,5. Після підстановки:

$$W = \frac{12 \times 636}{16 \times (512 - 8) - 4 \times 6{,}5} = \frac{7632}{8064 - 26} = \frac{7632}{8038} \approx 0{,}9490$$

що повністю збігається з результатом програми. Критерій Пірсона розраховано за формулою $\chi^2 = m \cdot (n\text{-}1) \cdot W = 4 \cdot 7 \cdot 0{,}9490 = 26{,}57$, що також збігається з отриманим значенням.

Перевірка режиму методу аналізу ієрархій (MAI)

Для перевірки цього режиму було введено матриці попарних порівнянь, наведені в підрозділі 2.6 (рис. 2.9–2.14). Програма послідовно обчислила:

- Ваги критеріїв (рис. 3.4):

АУ = 0,1692, КЦ = 0,2852, КД = 0,4732, ЗС = 0,0725.

- Відношення узгодженості: HR = 0,0191 (<0,1) – прийнятна.
- Локальні пріоритети для кожного критерію (рис. 3.6–3.10):

Критерій АУ: S=0,4668, T=0,1603, I=0,2776, D=0,0953; HR=0,0115.

Критерій КЦ: S=0,2328, T=0,5451, I=0,1385, D=0,0837; HR=0,0190.

Критерій КД: S=0,2455, T=0,1504, I=0,5350, D=0,0691; HR=0,0436.

Критерій ЗС: S=0,0836, T=0,2328, I=0,1384, D=0,5451; HR=0,0190.

- Глобальні пріоритети загроз (рис. 3.5):

S = 0,2676, T = 0,2706, I = 0,3496, D = 0,1122.

- Сума глобальних пріоритетів = 1,0000.

Для верифікації я розрахую глобальний пріоритет для загрози I (розкриття) вручну за формулою:

$$V_I = 0{,}1692 \times 0{,}2776 + 0{,}2852 \times 0{,}1385 + 0{,}4732 \times 0{,}5350 + 0{,}0725 \times 0{,}1384$$

$$V_I = 0{,}04697 + 0{,}03951 + 0{,}25316 + 0{,}01003 = 0{,}34967 \approx 0{,}3497$$

Отримане значення збігається з результатом програми (0,3496). Аналогічні розрахунки для інших загроз підтверджують правильність обчислення глобальних пріоритетів. Усі відношення узгодженості (HR) виявилися меншими за 0,1, що відповідає критерію Т. Сааті.

Перевірка на граничних випадках

Додатково було виконано тестування на граничних даних:

- Для повністю узгоджених рангів (усі експерти дали однакові оцінки [1,2,…,8]) програма обчислила W = 1,000.
- Для ідеально узгодженої матриці попарних порівнянь (наприклад, [1,2,4; 1/2,1,2; 1/4,1/2,1]) отримано HR = 0.
- При введенні завідомо неузгоджених даних (випадкові порівняння) програма видала HR > 0,1 та відповідне попередження.

Таким чином, перевіривши всі розрахунки можна зробити висновок, що програма розраховує вірно.

Висновок до 3 розділу

У третьому розділі дипломної роботи на основі теоретичних та аналітичних результатів, отриманих у попередніх розділах, було розроблено та реалізовано програмне забезпечення для автоматизації розрахунків, пов’язаних з ідентифікацією вразливостей інформаційних активів. Актуальність цього етапу зумовлена необхідністю підвищення точності та швидкості обробки експертних даних, а також зменшення впливу людського фактора при виконанні складних математичних обчислень.

Розроблено архітектуру програмного забезпечення, мовою C# у вигляді консольного додатку та складається з двох незалежних модулів: модуль ранжування активів (обчислення коефіцієнта конкордації, критерію Пірсона, цінності активів) та модуль методу аналізу ієрархій (пріоритизація загроз). Реалізовано алгоритми обробки експертних даних у модулі ранжування активів коректно реалізовано нормалізацію зв’язаних рангів, розрахунок сум рангів, коефіцієнта конкордації Кендалла та критерію Пірсона. Модуль MAI забезпечує введення матриць попарних порівнянь за шкалою Сааті, обчислення локальних пріоритетів методом геометричного середнього, розрахунок максимального власного значення, індексу та відношення узгодженості (HR), а також обчислення глобальних пріоритетів альтернатив (загроз). Проведено тестування програми, та перевірено обчислення даних.

Розроблена програма може бути використана як допоміжний засіб для фахівців з кібербезпеки при обробці експертних оцінок та пріоритизації загроз, що підвищує ефективність процесу ідентифікації вразливостей. Код програми наведено в додатку А.

Висновок

У дипломній роботі розглянуто актуальну проблему ідентифікації вразливостей інформаційних активів об’єкта захисту. У результаті виконаних досліджень розроблено та обґрунтовано методичні підходи до виявлення, аналізу та оцінювання вразливостей.

Основними досягненнями роботи стали детальний аналіз сучасних методів ідентифікації вразливостей, що дозволило визначити ключові підходи до побудови ефективного процесу оцінки ризиків. За допомогою інструментів Nmap та OpenVAS виконано інвентаризацію активів та виявлено 34 вразливості різного рівня критичності. Проведено оцінку критичності вразливостей за методикою CVSS, що дозволило визначити пріоритети їх усунення.

Удосконалено процес ідентифікації активів шляхом застосування експертного ранжування восьми категорій активів із використанням коефіцієнта конкордації Кендалла, що підтвердило високу узгодженість думок експертів. Найвищу цінність отримали інформаційні активи (дані), що дозволило обґрунтовано визначити пріоритети захисту. Побудовано модель загроз за методологією STRIDE та модель порушника (п’ять типів), які дали змогу оцінити ймовірність реалізації загроз. За допомогою методу аналізу ієрархій (MAI) визначено глобальні пріоритети загроз: найбільш критичною виявилася загроза розкриття інформації.

Розроблено програмне забезпечення на мові C# для автоматизації розрахунків ранжування активів та методу аналізу ієрархій, що дозволяє підвищити точність обчислень і зменшити вплив людського фактора. Проведене тестування програми на реальних даних підтвердило її достовірність.

Розроблений комплексний підхід до ідентифікації вразливостей інформаційних активів є надійним інструментом для управління ризиками та забезпечення безпеки в сучасних інформаційних системах. Його впровадження дозволить організаціям ефективно захищати свої інформаційні активи та мінімізувати потенційні загроз.

Список використаних джерел

1. Конституція України : прийнята на п’ятій сесії Верховної Ради України 28 черв. 1996 р. — Київ : Преса України, 1997. — 80 с.
2. Про інформацію : Закон України від 02.10.1992 р. № 2657-XII. — Відомості Верховної Ради України, 1992. — № 48. — Ст. 650.
3. Про захист персональних даних : Закон України від 01.06.2010 р. № 2297-VI. — Відомості Верховної Ради України, 2010. — № 34. — Ст. 481.
4. Про захист інформації в інформаційно-телекомунікаційних системах : Закон України від 05.07.1994 р. № 80/94-ВР. — Відомості Верховної Ради України, 1994. — № 31. — Ст. 286.
5. Про основні засади забезпечення кібербезпеки України : Закон України від 05.10.2017 р. № 2163-VIII. — Відомості Верховної Ради України, 2017. — № 45. — Ст. 403.
6. Про державну таємницю : Закон України від 21.01.1994 р. № 3855-XII. — Відомості Верховної Ради України, 1994. — № 16. — Ст. 93.
7. Кримінальний кодекс України : Закон України від 05.04.2001 р. № 2341-III. — Відомості Верховної Ради України, 2001. — № 25–26. — Ст. 131.
8. Джонсон А., Белл К., Ковінгтон М. Управління Інформаційною Безпекою: Принципи та Практика. 2-ге вид. – Пірсон, 2018. – 390 ст.
9. Методика оцінки рівня захисту інформації в інформаційних системах. Затверджено наказом Державної служби спеціального зв'язку та захисту інформації України від 02.02.2012 № 76. – Київ: ДССЗЗІ України, 2012. – 48 с.
10. ДСТУ 3396.0-96. Захист інформації. Технічний захист інформації. Основні положення. — Київ : Держстандарт України, 1996. — 18 с.
11. ДСТУ 3396.2-97. Захист інформації. Технічний захист інформації. Терміни та визначення. — Київ : Держстандарт України, 1997. — 22 с.
12. НД ТЗІ 2.5-004-99. Критерії оцінки захищеності інформації в комп’ютерних системах від несанкціонованого доступу. — Київ : Адміністрація Держспецзв’язку, 1999. — 45 с.

13. НД ТЗІ 3.7-003-05. Порядок проведення робіт із створення комплексної системи захисту інформації в інформаційно-телекомунікаційній системі. — Київ : Адміністрація Держспецзв'язку, 2005. — 32 с.
14. ДСТУ ISO/IEC 27001:2023. Інформаційна безпека, кібербезпека та захист приватності. Системи керування інформаційною безпекою. Вимоги. (ISO/IEC 27001:2022, IDT). — Київ : Держспоживстандарт України, 2023. — 36 с.
15. ДСТУ ISO/IEC 27002:2023. Інформаційна безпека, кібербезпека та захист приватності. Засоби керування інформаційною безпекою. (ISO/IEC 27002:2022, IDT). — Київ : Держспоживстандарт України, 2023. — 112 с.
16. ДСТУ ISO/IEC 27005:2023. Інформаційна безпека, кібербезпека та захист приватності. Настанови щодо управління ризиками інформаційної безпеки. (ISO/IEC 27005:2022, IDT). — Київ : Держспоживстандарт України, 2023. — 48 с.
17. Професійний стандарт «Аналітик з оцінки вразливостей» : Наказ Адміністрації Державної служби спеціального зв'язку та захисту інформації України від 23.01.2024 р. № 38. — Київ : Держспецзв'язку, 2024. — 26 с.
18. Богуш В.М., Кривуца В.Г., Кудін А.М. Інформаційна безпека : термінологічний навчальний довідник / за ред. В.Г. Кривуци. — Київ : Вид-во Європ. ун-ту, 2004. — 508 с.
19. Nmap Network Scanning : official documentation [Електронний ресурс] / Nmap.org. – Режим доступу: https://nmap.org/docs.html. (дата звернення 05.04.2026р.)
20. OpenVAS – Open Vulnerability Assessment System : official documentation [Електронний ресурс] / Greenbone Networks. – Режим доступу: https://www.greenbone.net/en/documentation/ (дата звернення 05.04.2026р.)
21. National Vulnerability Database (NVD) [Електронний ресурс] / National Institute of Standards and Technology (NIST). – Режим доступу: https://nvd.nist.gov/ (дата звернення 07.04.2026р.)

22. Балацький О.В., Пасічник В.В., Ярошенко Ю.О. Інформаційна безпека : теорія та практика : навч. посіб. — Київ : Видавничо-поліграфічний центр «Київський університет», 2012. — 311 с.
23. Thevarmannil M. What Is the STRIDE Threat Model? Beginner's Guide – 2026 [Електронний ресурс] / M. Thevarmannil // Practical DevSecOps. – 2026. – Режим доступу: https://www.practical-devsecops.com/what-is-stride-threat-model/ (дата звернення 09.04.2026р.)
24. Ізмайлова О.В. Теорія прийняття рішень : метод. вказівки – Київ : КНУБА, 2024. – 39 с.
25. Ізмайлова О.В Метод аналізу ієрархій: метод. вказівки – Київ: КНУБА, 2024р – 23 с.
26. Галатенко В.А., Пасічник В.В., Ярошенко Ю.О. Захист інформації в комп'ютерних системах : навч. посіб. — Київ : Видавничо-поліграфічний центр «Київський університет», 2013. — 327 с.
27. Мелл П., Скарфон К. Спільна система оцінки вразливостей (CVSS) та її застосування до систем федеральних агентств // Доповідь міжагентського звіту NIST 7435. — Національний інститут стандартів і технологій, 2007. — 54 с.
28. CVE – Common Vulnerabilities and Exposures [Електронний ресурс] / The MITRE Corporation. – Режим доступу: https://www.cve.org/ (дата звернення 11.04.2026р)
29. CVSS v3.1 Specification Document [Електронний ресурс] / FIRST.Org, Inc. – Режим доступу: https://www.first.org/cvss/v3.1/specification-document (дата звернення 11.04.2026р)
30. Коваленко Т. Огляд сучасних методів оцінки вразливостей інформаційних систем // Журнал кібербезпеки. — 2019. — Том 15, № 3. — С. 75–88.

Додаток А код програми

```
using System;
using System.Collections.Generic;
using System.Linq;

namespace VulnerabilityAnalysis
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.OutputEncoding = System.Text.Encoding.UTF8;

Console.WriteLine("==========================================
===");
            Console.WriteLine("    АВТОМАТИЗАЦІЯ РОЗРАХУНКІВ
ДЛЯ АНАЛІЗУ");
            Console.WriteLine("                              ВРАЗЛИВОСТЕЙ
ІНФОРМАЦІЙНИХ АКТИВІВ");

Console.WriteLine("==========================================
===");

            while (true)
            {
                Console.WriteLine("\nОберіть режим роботи:");
                Console.WriteLine("1.  Ранжування  активів
(коефіцієнт конкордації)");
                Console.WriteLine("2. Метод аналізу ієрархій (MAI)
для пріоритизації загроз");
```

```
                Console.WriteLine("0. Вихід");
                Console.Write("Ваш вибір: ");
                string choice = Console.ReadLine();

                switch (choice)
                {
                    case "1":
                        RankingModule.PerformRanking();
                        break;
                    case "2":
                        AhpModule.PerformAHP();
                        break;
                    case "0":
                        Console.WriteLine("Дякуємо за використання програми!");
                        return;
                    default:
                        Console.WriteLine("Невірний вибір. Спробуйте ще раз.");
                        break;
                }
            }
        }
    }

    // ========================= МОДУЛЬ РАНЖУВАННЯ АКТИВІВ =========================
    static class RankingModule
    {
        public static void PerformRanking()
```

```
{
    Console.Write("\nВведіть кількість експертів (m): ");
    int m = int.Parse(Console.ReadLine());
    Console.Write("Введіть кількість активів (n): ");
    int n = int.Parse(Console.ReadLine());

    // Введення початкових рангів
    var rawMatrix = new List<List<int>>();
    Console.WriteLine("Введіть матрицю рангів (експерти - рядки, активи - стовпці):");
    for (int i = 0; i < m; i++)
    {
        var row = Console.ReadLine().Split().Select(int.Parse).ToList();
        if (row.Count != n) throw new Exception("Кількість стовпців не збігається");
        rawMatrix.Add(row);
    }

    // Нормалізація зв'язаних рангів
    var normalizedMatrix = new List<List<double>>();
    foreach (var row in rawMatrix)
    {
        normalizedMatrix.Add(NormalizeRanks(row));
    }

    // Обчислення сум рангів Rj
    double[] Rj = new double[n];
    for (int i = 0; i < m; i++)
        for (int j = 0; j < n; j++)
```

```
                    Rj[j] += normalizedMatrix[i][j];

            double T_aster = m * (n + 1.0) / 2.0;
            double sum_D2 = 0.0;
            for (int j = 0; j < n; j++)
            {
                double d = Rj[j] - T_aster;
                sum_D2 += d * d;
            }

            // Поправки на зв'язані ранги
            double sumTi = 0.0;
            foreach (var row in normalizedMatrix)
            {
                var groups = row.GroupBy(x => x).Where(g => g.Count() > 1);
                double Ti = 0.0;
                foreach (var g in groups)
                {
                    int t = g.Count();
                    Ti += (Math.Pow(t, 3) - t) / 12.0;
                }
                sumTi += Ti;
            }

            double numerator = 12.0 * sum_D2;
            double denominator = m * m * (Math.Pow(n, 3) - n) - m * sumTi;
            double W = numerator / denominator;
            double chi2 = m * (n - 1) * W;
```

```
            // Табличне значення χ² для α=0,05, ступенів свободи = n-1
            double chi2_table = (n - 1) switch
            {
                7 => 14.07,
                6 => 12.59,
                5 => 11.07,
                _ => 14.07
            };

            Console.WriteLine("\n--- РЕЗУЛЬТАТИ РАНЖУВАННЯ АКТИВІВ ---");
            Console.WriteLine($"Кількість експертів m = {m}");
            Console.WriteLine($"Кількість активів n = {n}");
            Console.Write("Суми рангів Rj: ");
            foreach (var v in Rj) Console.Write($"{v:F1}  ");
            Console.WriteLine();
            Console.WriteLine($"Коефіцієнт конкордації W = {W:F4}");
            Console.WriteLine($"Критерій Пірсона χ² = {chi2:F2}");
            Console.WriteLine($"Табличне значення χ²(0.05, {n - 1}) = {chi2_table}");
            if (chi2 > chi2_table)
                Console.WriteLine("=> Узгодженість експертів є статистично значущою (χ² > χ²_табл).");
            else
                Console.WriteLine("=> Узгодженість не є статистично значущою.");

            // Перерахунок цінності активів у шкалу 7-10
            double Rj_min = Rj.Min();
```

```
                double Rj_max = Rj.Max();
                double[] value = new double[n];
                for (int j = 0; j < n; j++)
                {
                        value[j] = 10.0 - (Rj[j] - Rj_min) / (Rj_max - Rj_min) *
3.0;
                }
                Console.WriteLine("\nЦінність активів (шкала 7-10):");
                for (int j = 0; j < n; j++)
                {
                        Console.WriteLine($"  Актив {j + 1}: {value[j]:F2}");
                }
        }

        // Нормалізація зв'язаних рангів для одного експерта
        static List<double> NormalizeRanks(List<int> ranks)
        {
                int n = ranks.Count;
                var indexed = ranks.Select((r, idx) => (r, idx)).OrderBy(x =>
x.r).ToList();
                double[] normalized = new double[n];
                int i = 0;
                while (i < n)
                {
                        int j = i;
                        while (j < n && indexed[j].r == indexed[i].r) j++;
                        double sumPos = 0;
                        for (int k = i; k < j; k++) sumPos += (k + 1);
                        double avg = sumPos / (j - i);
```

```
                    for (int k = i; k < j; k++) normalized[indexed[k].idx] =
avg;
                    i = j;
                }
                return normalized.ToList();
            }
        }

        // ========================= МОДУЛЬ МЕТОДУ АНАЛІЗУ
ІЄРАРХІЙ (MAI) =========================
        static class AhpModule
        {
            public static void PerformAHP()
            {
                Console.Write("\nВведіть кількість критеріїв: ");
                int nCrit = int.Parse(Console.ReadLine());
                var criteriaNames = new List<string>();
                for (int i = 0; i < nCrit; i++)
                {
                    Console.Write($"Назва критерію {i + 1}: ");
                    criteriaNames.Add(Console.ReadLine());
                }

                // 1. Матриця порівнянь критеріїв
                var criteriaMat = InputMatrix(nCrit, "критеріїв");
                var criteriaWeights = AhpPriority(criteriaMat);
                var (lambda_crit, CR_crit) = ConsistencyRatio(criteriaMat,
criteriaWeights);

                Console.WriteLine("\nВага критеріїв:");
```

```
for (int i = 0; i < nCrit; i++)
    Console.WriteLine($"  {criteriaNames[i]}: {criteriaWeights[i]:F4}");
Console.WriteLine($"λ_max = {lambda_crit:F4}, HR = {CR_crit:F4}");
Console.WriteLine(CR_crit < 0.1 ? "=> Узгодженість матриці критеріїв прийнятна (HR < 0.1)." : "=> Узгодженість матриці критеріїв неприйнятна (HR >= 0.1).");

// 2. Альтернативи (загрози)
Console.Write("\nВведіть кількість альтернатив (загроз): ");
int nAlt = int.Parse(Console.ReadLine());
var altNames = new List<string>();
for (int i = 0; i < nAlt; i++)
{
    Console.Write($"Назва альтернативи {i + 1}: ");
    altNames.Add(Console.ReadLine());
}

// 3. Для кожного критерію – матриця порівнянь альтернатив
double[,] altWeights = new double[nAlt, nCrit];
for (int k = 0; k < nCrit; k++)
{
    Console.WriteLine($"\n=== Критерій {k + 1}: {criteriaNames[k]} ===");
    var matAlt = InputMatrix(nAlt, $"альтернатив для критерію {criteriaNames[k]}");
    var weights = AhpPriority(matAlt);
    var (lam, cr) = ConsistencyRatio(matAlt, weights);
```

```
                Console.WriteLine("Локальні пріоритети альтернатив:");
                for (int i = 0; i < nAlt; i++)
                {
                    Console.WriteLine($"  {altNames[i]}: {weights[i]:F4}");
                    altWeights[i, k] = weights[i];
                }
                Console.WriteLine($"λ_max = {lam:F4}, HR = {cr:F4} {(cr < 0.1 ? "(прийнятна)" : "(неприйнятна)")}");
            }

            // 4. Глобальні пріоритети альтернатив
            double[] global = new double[nAlt];
            for (int i = 0; i < nAlt; i++)
            {
                double sum = 0;
                for (int j = 0; j < nCrit; j++)
                    sum += criteriaWeights[j] * altWeights[i, j];
                global[i] = sum;
            }

            Console.WriteLine("\n--- ГЛОБАЛЬНІ ПРІОРИТЕТИ ЗАГРОЗ ---");
            double total = 0;
            for (int i = 0; i < nAlt; i++)
            {
                Console.WriteLine($"  {altNames[i]}: {global[i]:F4}");
                total += global[i];
            }
```

```
            Console.WriteLine($"Сума глобальних пріоритетів = {total:F4} (повинна дорівнювати 1).");
        }

        // Введення матриці попарних порівнянь
        static double[,] InputMatrix(int n, string name)
        {
            Console.WriteLine($"\nВведіть матрицю попарних порівнянь для {name} (розмір {n}x{n}):");
            Console.WriteLine("Коефіцієнти шкали Сааті: 1 - однакова важливість, 3 - помірна, 5 - сильна, 7 - дуже сильна, 9 - надзвичайна.");
            var mat = new double[n, n];
            for (int i = 0; i < n; i++)
            {
                for (int j = 0; j < n; j++)
                {
                    if (i == j)
                    {
                        mat[i, j] = 1.0;
                        continue;
                    }
                    if (i < j)
                    {
                        Console.Write($"  Елемент [{i + 1},{j + 1}] = ");
                        double val = double.Parse(Console.ReadLine());
                        mat[i, j] = val;
                        mat[j, i] = 1.0 / val;
                    }
```

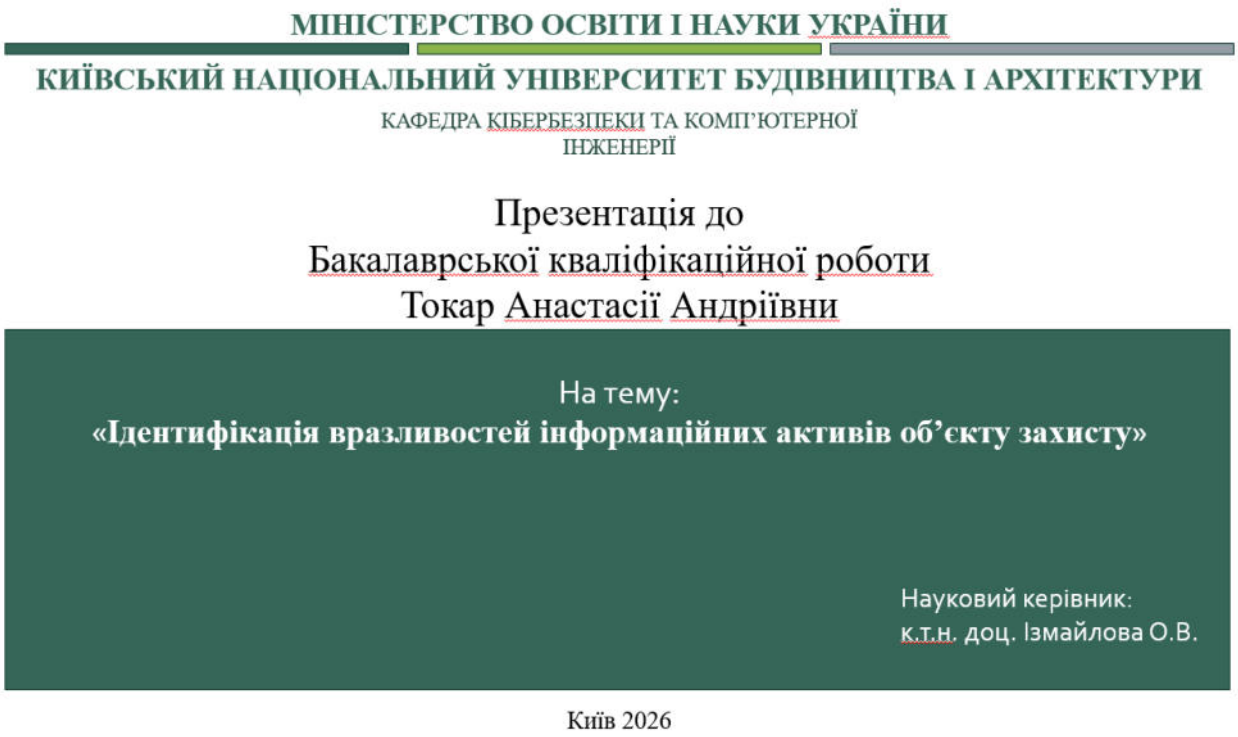
```
            }
        }
        return mat;
    }

    // Обчислення вектора пріоритетів (метод геометричного середнього)
    static double[] AhpPriority(double[,] matrix)
    {
        int n = matrix.GetLength(0);
        double[] geom = new double[n];
        for (int i = 0; i < n; i++)
        {
            double prod = 1.0;
            for (int j = 0; j < n; j++)
                prod *= matrix[i, j];
            geom[i] = Math.Pow(prod, 1.0 / n);
        }
        double sumGeom = geom.Sum();
        double[] weights = new double[n];
        for (int i = 0; i < n; i++)
            weights[i] = geom[i] / sumGeom;
        return weights;
    }

    // Обчислення λ_max та відношення узгодженості HR
    static (double lambda_max, double CR) ConsistencyRatio(double[,] matrix, double[] weights)
    {
        int n = matrix.GetLength(0);
```

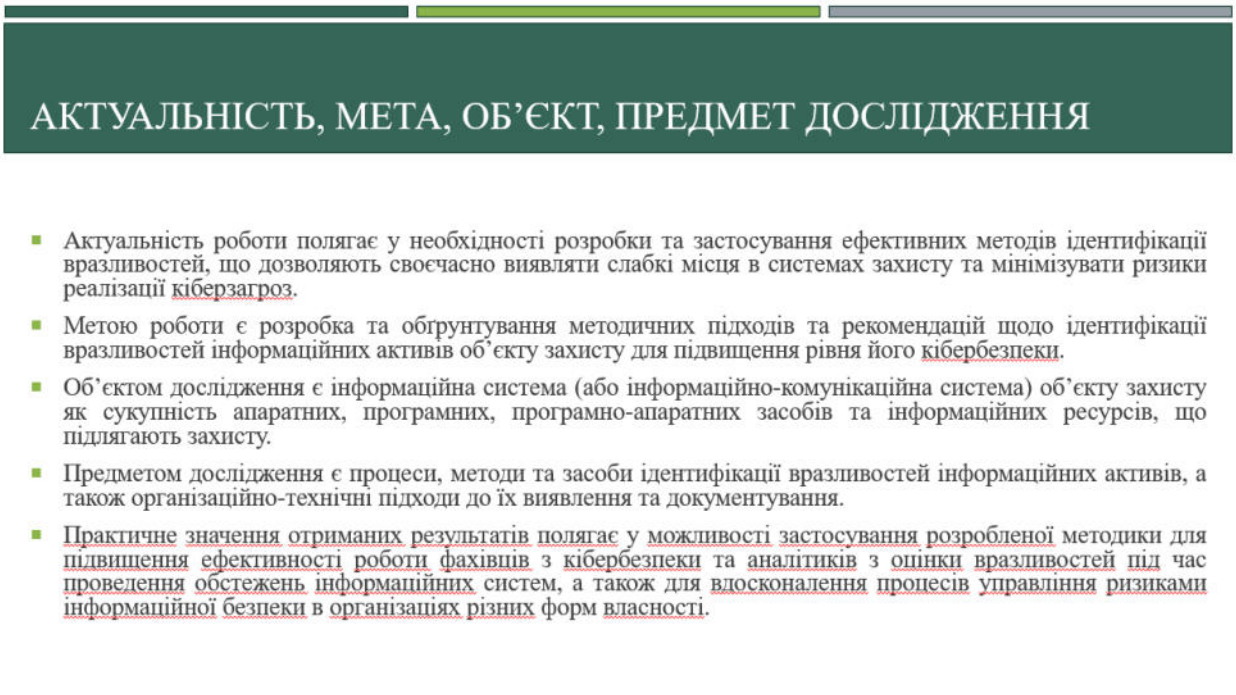
```
        double[] weightedSum = new double[n];
        for (int i = 0; i < n; i++)
        {
            double sum = 0;
            for (int j = 0; j < n; j++)
                sum += matrix[i, j] * weights[j];
            weightedSum[i] = sum;
        }
        double lambda_max = 0;
        for (int i = 0; i < n; i++)
            lambda_max += weightedSum[i] / weights[i];
        lambda_max /= n;

        double CI = (lambda_max - n) / (n - 1);
        double RI = n switch
        {
            1 => 0,
            2 => 0,
            3 => 0.58,
            4 => 0.9,
            5 => 1.12,
            6 => 1.24,
            7 => 1.32,
            8 => 1.41,
            9 => 1.45,
            _ => 1.45
        };
        double CR = CI / RI;
        return (lambda_max, CR);
    }
```

}

}

Додаток Б графічні зображення





Перелік нормативних документів

Категорія	Документи	Основне призначення
Конституція та базові закони	Конституція України Закон «Про інформацію» Закон «Про захист персональних даних»	Право на інформацію та обмеження для нацбезпеки; види інформації (конфіденційна, службова, таємна); захист персональних даних
Спеціальне законодавство ІБ	Закон «Про захист інформації в ІТС» Закон «Про основні засади кібербезпеки» Закон «Про державну таємницю» Кримінальний кодекс (ст. 361-363)	Визначає НСД, КСЗІ; аудит, моніторинг вразливостей; режим ДТ; відповідальність за кіберзлочини
Підзаконні акти (КМУ)	Постанова № 373 (Правила захисту інформації) Постанова № 92 (зміни до правил) Постанова № 821 (ліцензування ТЗІ)	Загальні вимоги до захисту; адаптація до електронних комунікацій; оцінка вразливостей – тільки ліцензованими суб'єктами
Національні стандарти ТЗІ	ДСТУ 3396.0-96 (основні положення) ДСТУ 3396.2-97 (терміни та визначення) НД ТЗІ 2.5-004-99 (критерії оцінки захищеності) НД ТЗІ 3.7-003-05 (порядок створення КСЗІ) НД ТЗІ 2.5-005-99 (класифікація АС)	Орієнтири для технічного захисту; єдина термінологія; шкала, метрики, рівні захищеності; етапи створення КСЗІ; визначення класу системи
Гармонізовані європейські стандарти	ДСТУ ISO/IEC 27001:2023 (СУІБ) ДСТУ ISO/IEC 27002:2023 (засоби контролю) ДСТУ ISO/IEC 27005:2023 (управління ризиками) ДСТУ ISO/IEC 27037:2017 (цифрові докази)	Ризик-орієнтований підхід; довідник контролів для усунення вразливостей; структурований процес управління ризиками; вимоги до цифрової криміналістики
Професійний стандарт	«Аналітик з оцінки вразливостей» (Наказ Держспецзв'язку № 38 від 23.01.2024)	Компетентності, трудові функції, результати навчання

КЛАСИФІКАЦІЯ ІНФОРМАЦІЙНИХ АКТИВІВ, ВРАЗЛИВОСТЕЙ

КЛАСИФІКАЦІЯ ВРАЗЛИВОСТЕЙ ЗА ДП
Апаратні
Програмні
Організаційні
Людські

КЛАСИФІКАЦІЯ ІА
Організаційні методи ІА
Технічні методи ІА
За функціональним призначенням
За рівнем доступу інформації
За критичністю і цінністю
За властивостями інформаційної безпеки

КЛАСИФІКАЦІЯ ВРАЗЛИВОСТЕЙ ЗА МВ
Мережеві
Системні
Прикладні
Пов'язані із засобами захисту

Методи ідентифікації вразливостей

Методи:

- Пасивні базуються на зборі та аналізі наявної інформації. Аналіз документації передбачає вивчення політик безпеки, регламентів, інструкцій, проектної документації, схем топології мережі, переліків обладнання та конфігураційних файлів.
- Активні методи передбачають безпосередню взаємодію з інформаційною системою і можуть певною мірою впливати на її роботу.

ID активу	Категорія	Назва / роль	IP-адреса	Критичність	Обґрунтування
A-001	Апаратне забезпечення	Веб-сервер (додатків)	192.168.56.105	Висока	Забезпечує роботу веб-додатків, доступний з мережі
A-002	Апаратне забезпечення	Файловий сервер	192.168.56.110	Середня	Зберігає конфігураційні файли та документи
A-003	Апаратне забезпечення	Робоча станція	192.168.56.120	Низька	Тестовий клієнт, не містить критичних даних

РАНЖУВАННЯ ІНФОРМАЦІЙНИХ АКТИВІВ ЗА ЦІННІСТЮ

Нормалізована матриця (середні ранги при зв'язаних рангах)

Експерт	Інформаційні активи	Програмні активи	Апаратні активи	Сервісні активи	Дані	Людські активи	ПК	Нематеріальні активи	Rj
1	1	3	3	3	5	6	7	8	36
2	1	2	3,5	3,5	5	7	7	7	36
3	1	2	3	4	5	6	7,5	7,5	36
4	1	2,5	2,5	4	5	6	7	8	36

Коефіцієнт конкордації W	
n (кількість експертів)	4
m (кількість активів)	8
ΣDj^2	636
ΣTi	4
W =	0,948310139

Цінність активів в діапазоні оцінювання

Формула: Цінність = 10 - (Rj - Rj_min)/(Rj_max - Rj_min) *3 (менший ранг = вища цінність)

Актив	Rj	Цінність
Інформаційні активи	4	10
Програмні активи	9,5	9,377358
Апаратні активи	12	9,09434
Сервісні активи	14,5	8,811321
Дані	20	8,188679
Людські активи	25	7,622642
ПК	28,5	7,226415
Нематеріальні активи	30,5	7

МОДЕЛЬ ЗАГРОЗ. МОДЕЛЬ ПОРУШНИКА

Категорія	Суть загрози	Приклад (з виявлених вразливостей)	Контрзахід
Spoofing (підміна)	Несанкціонований доступ через підміну облікових даних	CVE-2020-14145 (перерахування користувачів SSH) + відсутність MFA	MFA, сертифікати, складні паролі
Tampering (втручання)	Модифікація даних, ПЗ, логів	CVE-2021-44790 (Apache RCE) – зміна веб-сторінок, баз даних	Оновлення ПЗ, WAF, контроль цілісності
Repudiation (відмова)	Заперечення факту виконання дії	Логи зберігаються локально без захисту	SIEM, централізований захист логів, DLP
Information Disclosure (розкриття)	Ознайомлення з конфіденційною інформацією	FTP Anonymous Access, слабкі шифри SSL/TLS, TCP timestamps	Шифрування (TLS/VPN), DLP, RBA

Модель порушника
Внутрішній користувач (ненавмисний порушник)
Внутрішній користувач (умисний порушник / інсайдер)
Привілейований порушник (адміністратор, техпідтримка)
Зовнішній порушник (хакер, кіберзлочинець, конкурент)
Зовнішній порушник (випадковий / ненавмисний)

МЕТОД АНАЛІЗУ ІЄРАРХІЙ

Матриця попарних порівнянь для критеріїв

Критерій	АУ	КЦ	КД	ЗС	Добуток	ОВ	Вага
АУ	1	0,5	0,33	3	0,495	0,84	0,17
КЦ	2	1	0,5	4	4	1,41	0,29
КД	3	2	1	5	30	2,34	0,47
ЗС	0,33	0,25	0,2	1	0,0165	0,38	0,07
Сума						4,95	

Критерій	Вага	Пояснення
КД (контроль доступу / конфіденційність)	0,47	Найвищий пріоритет – захист персональних даних та комерційної таємниці
КЦ (контроль цілісності)	0,29	Високий пріоритет – захист цілісності даних та логів
АУ (автентифікація)	0,17	Середній пріоритет – контроль доступу через автентифікацію
ЗС (доступність)	0,07	Найнижчий пріоритет – забезпечення доступності

Загроза	АУ (0,17)	КЦ (0,29)	КД (0,47)	ЗС (0,07)	Vi
Підміна	0,47	0,23	0,25	0,08	0,2697
Втручання	0,16	0,55	0,15	0,23	0,2733
Розкриття	0,28	0,14	0,54	0,14	0,3518
Відмова	0,10	0,08	0,07	0,55	0,1116
Вага	0,17	0,29	0,47	0,07	

CVSS

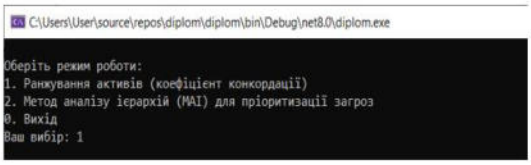
CVSS (Common Vulnerability Scoring System) — це загальноприйнятий стандарт для визначення ступеня небезпеки вразливостей в інформаційних системах.

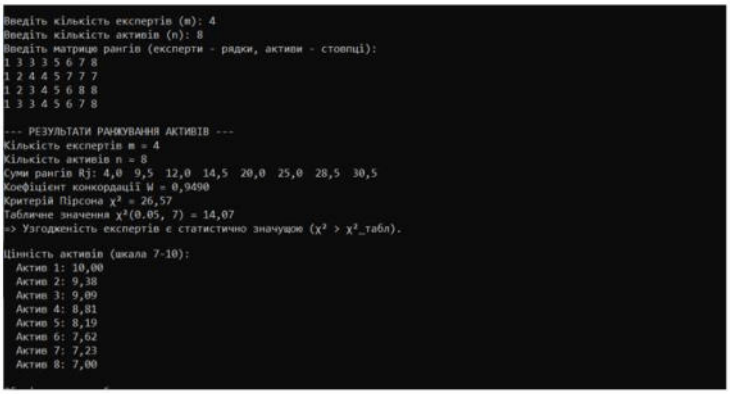
Вразливість (CVE / опис)	Уражений актив	AV	AC	PR	UI	C	I	A	Base Score	Рівень
CVE-2021-44790 (Apache RCE)	192.168.56.105	0,85	0,77	0,85	0,85	0,56	0,56	0,56	9,8	Critical
CVE-2020-14145 (OpenSSH user enum)	192.168.56.105	0,85	0,77	0,85	0,85	0,22	0,22	0	5,3	Medium
FTP Anonymous Access	192.168.56.110	0,85	0,77	0,85	0,85	0,22	0	0	5	Medium
SSH weak MAC algorithms	192.168.56.105	0,85	0,44	0,85	0,85	0	0	0	4,3	Medium
SSL/TLS Weak Ciphers	192.168.56.110	0,85	0,44	0,85	0,85	0,22	0	0	4,8	Medium

ПРОГРАМНА РЕАЛІЗАЦІЯ

Модулі:

- RankingModule – ранжування активів (коефіцієнт конкордації, критерій Пірсона, цінність).
- AhpModule – метод аналізу ієрархій (власні вектори, перевірка узгодженості HR, глобальні пріоритети).





ПРОГРАМА (ВИБІР 2 МАІ)

```
Оберіть режим роботи:
1. Ранжування активів (коефіцієнт конкордації)
2. Метод аналізу ієрархій (МАІ) для пріоритизації загроз
0. Вихід
Ваш вибір: 2

Введіть кількість критеріїв: 4
Назва критерію 1: AY
Назва критерію 2: КЦ
Назва критерію 3: КД
Назва критерію 4: ЗС

Введіть матриці попарних порівнянь для критеріїв (розмір 4x4):
Коефіцієнти шкали Сааті: 1 - однакова важливість, 3 - помірна, 5 - сильна, 7 - дуже сильна, 9 - надзвичайна.
  Елемент [1,2] = 0,5
  Елемент [1,3] = 0,33
  Елемент [1,4] = 3
  Елемент [2,3] = 0,5
```

```
  Елемент [3,4] = 5

Ваги критеріїв:
  AY: 0,1692
  КЦ: 0,2852
  КД: 0,4732
  ЗС: 0,0725
λ_max = 4,0515, HR = 0,0191
=> Узгодженість матриці критеріїв прийнятна (HR < 0.1).

Введіть кількість альтернатив (загроз): 4
Назва альтернативи 1: S
Назва альтернативи 2: T
Назва альтернативи 3: I
Назва альтернативи 4: D

=== Критерій 1: AY ===

Введіть матриці попарних порівнянь для альтернатив для критерію AY (розмір 4x4):
Коефіцієнти шкали Сааті: 1 - однакова важливість, 3 - помірна, 5 - сильна, 7 - дуже сильна, 9 - надзвичайна.
  Елемент [1,2] = 3
  Елемент [1,3] = 2
  Елемент [1,4] = 4
  Елемент [2,3] = 0,5
  Елемент [2,4] = 2
  Елемент [3,4] = 3
Локальні пріоритети альтернатив:
  S: 0,4668
  T: 0,1603
  I: 0,2776
  D: 0,0953
λ_max = 4,0310, HR = 0,0115 (прийнятна)

=== Критерій 2: КЦ ===

Введіть матриці попарних порівнянь для альтернатив для критерію КЦ (розмір 4x4):
Коефіцієнти шкали Сааті: 1 - однакова важливість, 3 - помірна, 5 - сильна, 7 - дуже сильна, 9 - надзвичайна.
  Елемент [1,2] = 0,333
  Елемент [1,3] = 2
  Елемент [1,4] = 3
  Елемент [2,3] = 4
  Елемент [2,4] = 5
  Елемент [3,4] = 2
Локальні пріоритети альтернатив:
  S: 0,2328
  T: 0,5451
  I: 0,1385
  D: 0,0837
λ_max = 4,0512, HR = 0,0190 (прийнятна)
```

ПРОГРАМА (ВИБІР 2 МАІ)

```
=== Критерій 3: КД ===

Введіть матриці попарних порівнянь для альтернатив для критерію КД (розмір 4x4):
Коефіцієнти шкали Сааті: 1 - однакова важливість, 3 - помірна, 5 - сильна, 7 - дуже сильна, 9 - надзвичайна.
  Елемент [1,2] = 2
  Елемент [1,3] = 0,333
  Елемент [1,4] = 4
  Елемент [2,3] = 0,25
  Елемент [2,4] = 3
  Елемент [3,4] = 5
Локальні пріоритети альтернатив:
  S: 0,2455
  T: 0,1504
  I: 0,5350
  D: 0,0691
λ_max = 4,1178, HR = 0,0436 (прийнятна)

=== Критерій 4: ЗС ===

Введіть матриці попарних порівнянь для альтернатив для критерію ЗС (розмір 4x4):
Коефіцієнти шкали Сааті: 1 - однакова важливість, 3 - помірна, 5 - сильна, 7 - дуже сильна, 9 - надзвичайна.
  Елемент [1,2] = 0,333
  Елемент [1,3] = 0,5
  Елемент [1,4] = 0,2
  Елемент [2,3] = 2
  Елемент [2,4] = 0,333
  Елемент [3,4] = 0,25
Локальні пріоритети альтернатив:
  S: 0,0836
  T: 0,2328
  I: 0,1384
  D: 0,5451
λ_max = 4,0512, HR = 0,0190 (прийнятна)
```

```
--- ГЛОБАЛЬНІ ПРІОРИТЕТИ ЗАГРОЗ ---
  S: 0,2676
  T: 0,2706
  I: 0,3496
  D: 0,1122
Сума глобальних пріоритетів = 1,0000 (повинна дорівнювати 1).

Оберіть режим роботи:
1. Ранжування активів (коефіцієнт конкордації)
```

ПРОГРАМА (ВИБІР 2 МАІ)

```
 Елемент [2,3] = 0,25
 Елемент [2,4] = 3
 Елемент [3,4] = 5
окальні пріоритети альтернатив:
 S: 0,2455
 T: 0,1504
 I: 0,5350
 D: 0,0691
_max = 4,1178, HR = 0,0436 (прийнятна)

=== Критерій 4: ЗС ===

ведіть матрицю попарних порівнянь для альтернатив для критерію ЗС (розмір 4x4):
оефіцієнти шкали Сааті: 1 - однакова важливість, 3 - помірна, 5 - сильна, 7 - дуже сильна, 9 - надзвичайна.
 Елемент [1,2] = 0,333
 Елемент [1,3] = 0,5
 Елемент [1,4] = 0,2
 Елемент [2,3] = 2
 Елемент [2,4] = 0,333
 Елемент [3,4] = 0,25
окальні пріоритети альтернатив:
 S: 0,0836
 T: 0,2328
 I: 0,1384
 D: 0,5451
_max = 4,0512, HR = 0,0190 (прийнятна)
```

```
--- ГЛОБАЛЬНІ ПРІОРИТЕТИ ЗАГРОЗ ---
  S: 0,2676
  T: 0,2706
  I: 0,3496
  D: 0,1122
Сума глобальних пріоритетів = 1,0000 (повинна дорівнювати 1).
```

ВИСНОВКИ

- У дипломній роботі розглянуто актуальну проблему ідентифікації вразливостей інформаційних активів об'єкта захисту. У результаті виконаних досліджень розроблено та обґрунтовано методичні підходи до виявлення, аналізу та оцінювання вразливостей.
- Основними досягненнями роботи стали детальний аналіз сучасних методів ідентифікації вразливостей, що дозволило визначити ключові підходи до побудови ефективного процесу оцінки ризиків. За допомогою інструментів Nmap та OpenVAS виконано інвентаризацію активів та виявлено 34 вразливості різного рівня критичності. Проведено оцінку критичності вразливостей за методикою CVSS, що дозволило визначити пріоритети їх усунення.
- Удосконалено процес ідентифікації активів шляхом застосування експертного ранжування восьми категорій активів із використанням коефіцієнта конкордації Кендалла, що підтвердило високу узгодженість думок експертів. Найвищу цінність отримали інформаційні активи (дані), що дозволило обґрунтовано визначити пріоритети захисту. Побудовано модель загроз за методологією STRIDE та модель порушника (п'ять типів), які дали змогу оцінити ймовірність реалізації загроз. За допомогою методу аналізу ієрархій (МАІ) визначено глобальні пріоритети загроз: найбільш критичною виявилася загроза розкриття інформації.
- Розроблений комплексний підхід до ідентифікації вразливостей інформаційних активів є надійним інструментом для управління ризиками та забезпечення безпеки в сучасних інформаційних системах. Його впровадження дозволить організаціям ефективно захищати свої інформаційні активи та мінімізувати потенційні загроз.

ДЯКУЮ ЗА УВАГУ!