МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

**ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ**

**НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ КІБЕРБЕЗПЕКИ ТА ЗАХИСТУ ІНФОРМАЦІЇ
КАФЕДРА УПРАВЛІННЯ КІБЕРБЕЗПЕКОЮ ТА ЗАХИСТОМ ІНФОРМАЦІЇ**

КВАЛІФІКАЦІЙНА РОБОТА

на тему: “СИСТЕМА ОЦІНЮВАННЯ РИЗИКІВ ІНФОРМАЦІЙНОЇ БЕЗПЕКИ ОРГАНІЗАЦІЇ НА ОСНОВІ OSINT-ДАНИХ”

на здобуття освітнього ступеня бакалавра
зі спеціальності 125 Кібербезпека
освітньої програми Управління інформаційною та кібернетичною безпекою

Кваліфікаційна робота містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело

_____ Максим КЛЄЩОВ
(підпис) *Ім’я, ПРІЗВИЩЕ здобувача*

Виконав(ла): здобувач(ка) вищої освіти гр. УБД-42

Максим КЛЄЩОВ
Ім’я, ПРІЗВИЩЕ

Керівник: Ірина ЛОЗОВА
к.т.н. Ім’я, ПРІЗВИЩЕ

Рецензент: Сергій ГАХОВ
к.т.н., доцент Ім’я, ПРІЗВИЩЕ

Київ 2026

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ

Навчально-науковий інститут кібербезпеки та захисту інформації

Кафедра управління кібербезпекою та захистом інформації

Ступінь вищої освіти бакалавр

Спеціальність 125 Кібербезпека

Освітня програма Управління інформаційною та кібернетичною безпекою

ЗАТВЕРДЖУЮ
Завідувач кафедри УКБЗІ
_____ Світлана ЛЕГОМІНОВА
“____”_____2026 р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

Клєщову Максиму Олександровичу
(прізвище, ім’я, по батькові здобувача)

1. Тема кваліфікаційної роботи “Система оцінювання ризиків інформаційної безпеки організації на основі OSINT-даних”,
керівник кваліфікаційної роботи ЛОЗОВА Ірина, к.т.н.
(ПРІЗВИЩЕ, Ім’я,, науковий ступінь, вчене звання)

затверджені наказом Державного університету інформаційно-комунікаційних технологій від “20“ лютого 2026 р. №51

2. Строк подання кваліфікаційної роботи “12” травня 2026р.

3. Вихідні дані до кваліфікаційної роботи: *методи оцінювання ризиків інформаційної безпеки, джерела та інструменти OSINT, моделі аналізу загроз і вразливостей, міжнародні стандарти, наукова та технічна література*

4. Перелік питань, які мають бути розроблені:

4.1. Проаналізувати існуючі методи та моделі оцінювання ризиків.

4.2. Дослідити підходи до збору та обробки OSINT-даних і розробити власну систему оцінювання ризиків інформаційної безпеки організації на їх основі.

4.3. Провести експериментальне дослідження ефективності запропонованої системи оцінювання ризиків на основі OSINT-даних.

5. Перелік ілюстративного матеріалу: презентація PowerPoint

6. Дата видачі завдання “15“ квітня 2026 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Етапи кваліфікаційної роботи	Термін виконання етапів роботи	Примітка
1.	Визначення об’єкту, предмету, мети та завдань дослідження.	19.04.2026	
2.	Збір та аналіз літератури.	25.04.2026	
3.	Аналіз сучасних методів оцінювання ризиків ІБ та ролі OSINT-даних у виявленні зовнішніх загроз.	7.05.2026	
4.	Розробка моделі інтеграції OSINT-даних у систему оцінювання ризиків організації.	10.05.2026	
5.	Проведення експериментального дослідження та оцінка ефективності запропонованого підходу	17.05.2026	
6.	Формулювання висновків за результатами проведеного дослідження.	22.05.2026	
7.	Оформлення роботи.	24.05.2026	
8.	Оформлення презентації.	26.05.2026	
9.	Отримання рецензії на роботу.	28.05.2026	
10.	Захист в ЕК.	_.06.2026	

Здобувач вищої освіти _____ Максим КЛЄЩОВ
(підпис) *(Ім’я, ПРІЗВИЩЕ)*

Керівник
кваліфікаційної роботи _____ Ірина ЛОЗОВА
(підпис) *(Ім’я, ПРІЗВИЩЕ)*

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ КІБЕРБЕЗПЕКИ ТА ЗАХИСТУ ІНФОРМАЦІЇ

ПОДАННЯ ГОЛОВІ ЕКЗАМЕНАЦІЙНОЇ КОМІСІЇ ЩОДО ЗАХИСТУ КВАЛІФІКАЦІЙНОЇ РОБОТИ
на здобуття освітнього ступеня бакалавра

Направляється здобувач Клєщов М. О. до захисту кваліфікаційної роботи
(прізвище та ініціали)
за спеціальністю 125 Кібербезпека
(код, найменування спеціальності)
освітньої програми Управління інформаційною та кібернетичною безпекою
(назва)
на тему: “Система оцінювання ризиків інформаційної безпеки організації на основі OSINT-даних”

Кваліфікаційна робота і рецензія додаються.

Директор ННІЗІ _____ Євгенія ІВАНЧЕНКО
(підпис) *(Ім’я, ПРІЗВИЩЕ)*

Висновок керівника кваліфікаційної роботи

Здобувач КЛЄЩОВ Максим у кваліфікаційній роботі проаналізував сучасні методи та засоби оцінювання ризиків інформаційної безпеки, дослідив можливості використання відкритих джерел інформації у процесі виявлення загроз і вразливостей, розробив систему автоматизованого аналізу та кореляції OSINT-даних із потенційними загрозами, а також практично реалізував і протестував програмну систему оцінювання ризиків інформаційної безпеки організації.

КЛЄЩОВ Максим продемонстрував розуміння досліджуваної проблеми та чітке бачення теоретичних і практичних шляхів її вирішення, підтвердив володіння методами наукового дослідження та навичками розробки програмного забезпечення, проявив себе як відповідальний та ініціативний виконавець. Результати дослідження апробовані на Всеукраїнській науково-практичній конференції "Стратегії кіберстійкості: управління ризиками та безперервність бізнесу" у 2026 році.

Все це дозволяє оцінити кваліфікаційну роботу здобувача КЛЄЩОВА Максима на оцінку "відмінно" та присвоїти йому кваліфікацію бакалавра з кібербезпеки за освітньою програмою Управління інформаційною та кібернетичною безпекою.

Керівник кваліфікаційної роботи _____ Ірина ЛОЗОВА
(підпис) *(Ім’я, ПРІЗВИЩЕ)*

“____“_____2026 року

Висновок кафедри про кваліфікаційну роботу

Кваліфікаційна робота розглянута. Здобувач Клєщов М. О. допускається до захисту даної роботи в Екзаменаційній комісії.

Завідувач кафедри управління кібербезпекою та захистом інформації _____ Світлана ЛЕГОМІНОВА
(підпис) *(Ім’я, ПРІЗВИЩЕ)*

ВІДГУК РЕЦЕНЗЕНТА
на кваліфікаційну бакалаврську роботу

здобувача вищої освіти КЛЄЩОВА Максима
на тему “Система оцінювання ризиків інформаційної безпеки організації на основі OSINT-даних”

Актуальність. В умовах стрімкої цифровізації діяльності організацій та постійного зростання кількості кіберзагроз особливої актуальності набувають питання своєчасного виявлення ризиків інформаційної безпеки. Значна частина інформації про організацію є доступною у відкритому доступі та може використовуватися зловмисниками під час підготовки атак. Тому застосування технологій OSINT для автоматизованого збору та аналізу відкритих даних дозволяє своєчасно виявляти потенційні загрози та оцінювати рівень ризику для інформаційної інфраструктури.

З огляду на зазначене дослідження можливість інтеграції OSINT-даних у процес оцінювання ризиків інформаційної безпеки є актуальним науковим завданням.

Позитивні сторони.

1. У роботі проведено порівняльний аналіз сучасних методів, моделей та засобів оцінювання ризиків інформаційної безпеки, що дозволило обґрунтувати доцільність інтеграції OSINT-даних у традиційні підходи.
2. Кваліфікаційна робота оформлена відповідно до встановлених вимог. Матеріал викладено послідовно та логічно, ключові положення проілюстровано рисунками та таблицями, зроблено обґрунтовані висновки.
3. Автор опрацював значну джерельну базу з 41 публікацій, серед яких наукові статті, міжнародні стандарти та технічна документація.
4. Розроблена програмна система практично реалізована та протестована на контрольованому середовищі, що підтверджує її працездатність.

Недоліки.

До недоліків роботи можна віднести те, що розроблена система орієнтована переважно на аналіз зовнішньої поверхні атаки організації та не враховує внутрішні ризики інформаційної безпеки, пов'язані з бізнес-процесами, інсайдерськими загрозами та фізичною безпекою.

Однак, вищезгадані зауваження не впливають на загальну позитивну оцінку кваліфікаційної роботи.

Висновок: Кваліфікаційна робота виконана на належному науково-методичному рівні і заслуговує оцінки "відмінно", а здобувач КЛЄЩОВ Максим заслуговує присвоєння кваліфікації бакалавра з кібербезпеки за освітньою програмою Управління інформаційною та кібернетичною безпекою.

Рецензент: _____ Сергій ГАХОВ
к.в.н., доцент *підпис* Ім’я, ПРІЗВИЩЕ

РЕФЕРАТ

Кваліфікаційна робота присвячена дослідженню системи оцінювання ризиків інформаційної безпеки організації на основі OSINT-даних. Робота складається зі вступу, трьох розділів, що містять 26 рисунків, висновків, списку використаних джерел із 41 найменувань та двох додатків. Загальний обсяг роботи становить 100 аркушів, з яких 6 аркуші займають перелік умовних скорочень і список використаних джерел.

Метою роботи є розробка та дослідження системи оцінювання ризиків інформаційної безпеки організації на основі використання даних відкритих джерел.

Об'єктом дослідження є процес оцінювання ризиків інформаційної безпеки організації.

Предмет дослідження - методи, моделі та засоби оцінювання ризиків інформаційної безпеки.

Методи дослідження. Для досягнення поставленої мети використано методи системного аналізу, теорію ризиків, методи статистичного спостереження, класифікацію, а також техніки OSINT-розвідки.

У результаті виконання роботи проведено аналіз сучасних методів оцінювання ризиків інформаційної безпеки, визначено роль і можливості використання OSINT-даних у процесі виявлення загроз, запропоновано підхід до інтеграції таких даних у процес оцінювання ризиків та виконано перевірку його придатності.

Галузь застосування. Розроблені підходи можуть бути використані в системах управління інформаційною безпекою організацій для підвищення ефективності процесів оцінювання ризиків, а також у практиці кібербезпекового аналізу та моніторингу загроз.

Ключові слова: ІНФОРМАЦІЙНА БЕЗПЕКА, OSINT, ОЦІНЮВАННЯ РИЗИКІВ, АНАЛІЗ ЗАГРОЗ, ВІДКРИТІ ДЖЕРЕЛА, УПРАВЛІННЯ РИЗИКАМИ.

ABSTRACT

The qualification paper is devoted to the study of an information security risk assessment system for organizations based on OSINT data. The paper consists of an introduction, three chapters containing 26 figures, conclusions, a list of references including 40 sources, and two appendices. The total volume of the paper is 100-pages, of which 6 pages are occupied by the list of abbreviations and the references section.

The purpose of the study is to develop and research an information security risk assessment system for an organization based on the use of open-source intelligence data.

The object of the study is the process of assessing an organization's information security risks.

The subject of the study is the analysis of methods, models and tools for information security risk assessment.

Research methods. To achieve the stated objective, methods of system analysis, risk theory, statistical observation methods, classification, and OSINT techniques were utilized.

As a result of the work, an analysis of modern information security risk assessment methods was performed, the role and possibilities of using OSINT data in the threat detection process were determined, an approach to integrating such data into the risk assessment process was proposed, and its suitability was verified.

Field of application. The developed approaches can be utilized in information security management systems of organizations to increase the efficiency of risk assessment processes, as well as in cybersecurity analysis and threat monitoring practices.

Keywords: INFORMATION SECURITY, OSINT, RISK ASSESSMENT, THREAT ANALYSIS, OPEN SOURCES, RISK MANAGEMENT.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ І СКОРОЧЕНЬ

API	Application Programming Interface. Програмний інтерфейс застосунку, набір правил для взаємодії між програмними компонентами.
CMS	Content Management System. Система керування вмістом вебсайту
CVE	Common Vulnerabilities and Exposures. База відомих вразливостей інформаційних систем
DNS	Domain Name System. Система доменних імен, що забезпечує перетворення доменних імен у IP-адреси.
FTP	File Transfer Protocol. Протокол передачі файлів
HTML	HyperText Markup Language. Мова розмітки гіпертексту для створення вебсторінок.
HTTP	HyperText Transfer Protocol. Протокол передачі даних у мережі Інтернет.
HTTPS	HyperText Transfer Protocol Secure. Захищена версія протоколу HTTP із шифруванням даних.
IoC	Indicators of Compromise. Індикатори компрометації.
IP	Internet Protocol. Протокол адресації та маршрутизації пакетів даних у мережі
NIST	National Institute of Standards and Technology. Національний інститут стандартів і технологій США.
OSINT	Open Source Intelligence. Розвідка на основі відкритих джерел інформації.
SSH	Secure Shell. Протокол захищеного віддаленого доступу
URL	Uniform Resource Locator. Уніфікований локатор ресурсу.
WHOIS	Сервіс отримання реєстраційної інформації про доменні імена та IP-адреси.
X	Соціальна мережа для публікації коротких повідомлень.

ВСТУП

Актуальність теми. В умовах постійного зростання кількості кіберзагроз та ускладнення методів їх реалізації питання оцінювання ризиків інформаційної безпеки набуває критичного значення для діяльності сучасних організацій. Ефективне управління ризиками передбачає не лише врахування внутрішніх характеристик інформаційних систем, але й аналіз зовнішнього середовища, у якому функціонує організація. Традиційні моделі оцінювання ризиків, що зосереджені виключно на внутрішньому аудиті, поступово втрачають ефективність, адже вони не враховують ту інформацію про компанію, яка вже доступна в мережі та може бути використана для підготовки нападу.

Більшість сучасних цілеспрямованих атак починається з етапу розвідки, де хакери використовують методи OSINT (Open Source Intelligence) для пошуку вразливих місць або даних про персонал у публічному просторі.

Впровадження інструментів OSINT у систему оцінювання ризиків дозволяє трансформувати захист із реактивного на превентивний, забезпечуючи можливість виявляти потенційні загрози ще на стадії їх підготовки.

Мета роботи полягає у розробці та дослідженні системи оцінювання ризиків інформаційної безпеки організації на основі використання даних відкритих джерел.

Об'єкт дослідження - процес оцінювання та прогнозування ризиків інформаційної безпеки організації.

Предмет дослідження - методи, моделі та засоби оцінювання ризиків інформаційної безпеки.

Новизна одержаних результатів полягає в удосконаленому підході до оцінювання ризиків інформаційної безпеки за рахунок автоматизованого аналізу, обробки та кореляції OSINT-даних із використанням методів нечіткої логіки, що дозволило виявляти складені загрози, скоротити час проведення безпекового аналізу, мінімізувати вплив людського фактора та підвищити точність оцінювання ризиків в умовах неповноти або фрагментарності вхідних даних

Для досягнення цієї мети необхідно виконати наступні ***завдання:***

1. Проаналізувати сучасні методи, моделі та засоби оцінювання ризиків інформаційної безпеки та визначити обмеження традиційних підходів.
2. Дослідити роль OSINT у моделюванні загроз та розробити систему оцінювання ризиків на основі кореляції відкритих даних.
3. Провести експериментальну перевірку ефективності запропонованої системи та порівняти її результати з існуючими рішеннями.

Методи дослідження. Для досягнення поставленої мети у роботі використано методи системного аналізу, узагальнення та порівняння для дослідження існуючих підходів до оцінювання ризиків, методи класифікації для структурування загроз і джерел даних, а також елементи моделювання для розробки системи оцінювання ризиків з урахуванням OSINT-даних.

Практичне значення одержаних результатів. Запропонований підхід до оцінювання ризиків інформаційної безпеки з використанням OSINT-даних може бути застосований у діяльності організацій для підвищення ефективності виявлення загроз, покращення якості аналізу ризиків та підтримки прийняття управлінських рішень у сфері інформаційної безпеки.

Апробація результатів кваліфікаційної роботи відбулася на Всеукраїнській науково-практичній конференції “Стратегії кіберстійкості: управління ризиками та безперервність бізнесу” 25 лютого 2026 року,де було представлено тези доповіді "Система оцінювання ризиків інформаційної безпеки організації на основі OSINT-даних" [41].

РОЗДІЛ 1 АНАЛІЗ МЕТОДІВ ТА ЗАСОБІВ ОЦІНЮВАННЯ РИЗИКІВ ІНФОРМАЦІЙНОЇ БЕЗПЕКИ

1.1 Процес оцінювання ризиків інформаційної безпеки та його основні етапи

Стрімке зростання кількості кібератак у сучасному цифровому середовищі ставить перед організаціями вимогу безперервного розвитку засобів і методів захисту. Саме управління ризиками інформаційної безпеки забезпечує здатність організації своєчасно реагувати на зміни в ландшафті загроз і відповідно вдосконалювати механізми протидії їм [1].

ISO/IEC 27005 – це міжнародний стандарт з інформаційної безпеки, який надає настанови щодо управління ризиками інформаційної безпеки, прийнятий в Україні як національний стандарт ДСТУ ISO/IEC 27005 [1]. Стандарт базується на загальних принципах управління ризиками, визначених ISO 31000 [5], проте є більш деталізованим та орієнтованим саме на сферу інформаційної безпеки.

У сфері інформаційної безпеки під ризиком розуміють потенційну можливість виникнення подій, що можуть призвести до порушення конфіденційності, цілісності або доступності інформації, а також спричинити збитки для організації.

Згідно з ISO/IEC 27005, управління ризиками інформаційної безпеки складається з наступних етапів [1,4]:

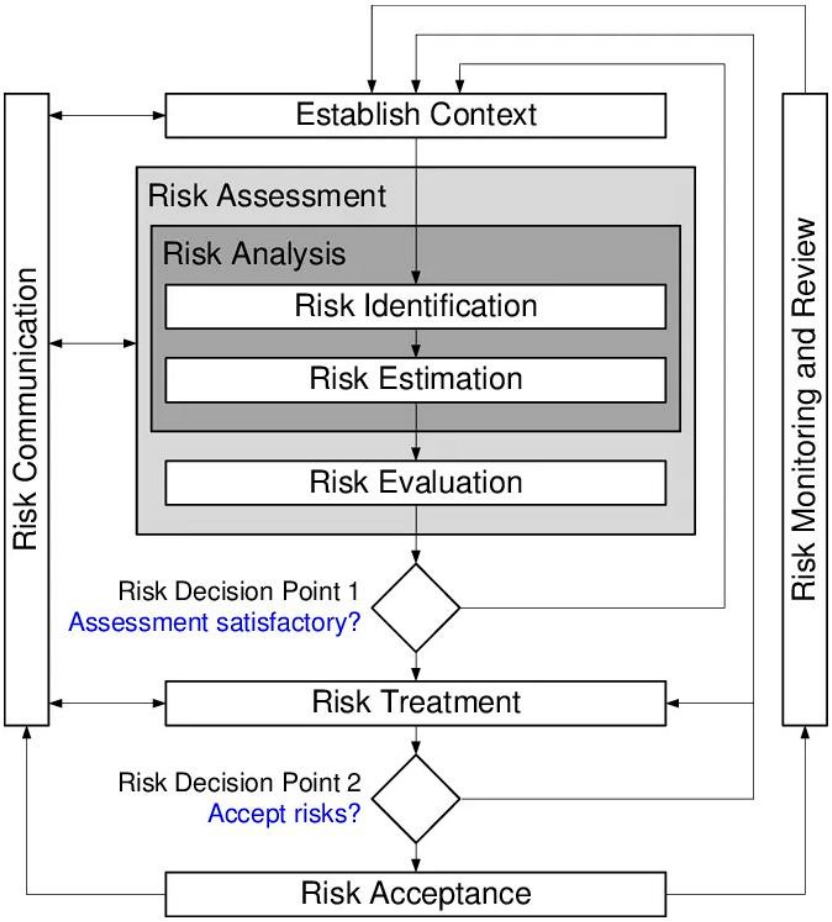


Рис. 1.1. Схема управління ризиками відповідно до ISO/IEC 27005 [3]

1. Визначення контексту

Цей етап охоплює встановлення загального операційного середовища організації, її стратегічних цілей, ключових процесів та чинників, що формують рівень захищеності інформації.

Спочатку проводиться комплексний аналіз внутрішніх і зовнішніх умов діяльності організації, який охоплює вивчення особливостей бізнес-процесів, оцінку операційного середовища та ідентифікацію потенційних загроз. Водночас визначаються нормативні та регуляторні вимоги, а також очікування всіх зацікавлених сторін - замовників, ділових партнерів, власників та органів державного нагляду.

Невід'ємним елементом цього етапу є проведення інвентаризації інформаційних ресурсів організації, що передбачає складання їх переліку,

встановлення відповідальних осіб та визначення ступеня їх критичності для забезпечення безперервності діяльності. Отримані результати слугують підґрунтям для розстановки пріоритетів у подальшому процесі аналізу загроз.

На завершення формуються підходи та критерії управління ризиками: організація розробляє відповідну політику, обирає методи оцінювання та визначає механізми впровадження захисних заходів. У цьому процесі беруться до уваги стратегічні цілі бізнесу, значущість інформаційних ресурсів і систем їх опрацювання, а також чинні правові та нормативні вимоги.

2. Ідентифікація ризиків

На цьому кроці відбувається виявлення та документування ризиків, здатних негативно вплинути на її інформаційні ресурси. Головною метою є отримання повного уявлення про природу ризиків, їх першопричини та можливі наслідки для функціонування організації.

Підсумком ідентифікації є створення реєстру ризиків, що містить впорядковані відомості про інформаційні активи, актуальні загрози, наявні вразливості, задіяні механізми захисту та очікувані негативні наслідки. Цей реєстр виступає відправною точкою для проведення подальшого аналізу, кількісного та якісного оцінювання ризиків і вибору заходів їх нейтралізації в рамках системи управління інформаційною безпекою.

3. Аналіз та оцінка ризиків.

Після завершення ідентифікації ризиків організація переходить до їх аналізу та оцінювання. На цьому етапі узагальнюється інформація про активи, загрози, вразливості та наявні засоби захисту з метою подальшого прийняття обґрунтованих управлінських рішень.

Першим кроком є визначення підходів і методології аналізу ризиків. Організація обирає, чи буде застосовуватися якісний, кількісний або комбінований підхід, а також встановлює відповідні критерії оцінювання, показники ефективності та метрики для аналізу ризиків.

У процесі аналізу здійснюється оцінка ймовірності реалізації ризику, яка відображає можливість виникнення небажаної події з урахуванням

особливостей інформаційної системи, наявних вразливостей, технологічного середовища, а також внутрішніх і зовнішніх факторів. Паралельно визначається потенційний вплив ризику, тобто можливі наслідки для організації у разі реалізації загрози. До таких наслідків можуть належати фінансові втрати, порушення безперервності бізнес-процесів, зниження репутації або інші негативні ефекти.

На основі отриманих оцінок виконується визначення рівня ризику. Це дозволяє порівнювати ризики між собою та встановлювати їх пріоритетність.

4. Оброблення ризиків.

Далі організація переходить до етапу їх оброблення, метою якого є визначення оптимальних заходів реагування для зниження рівня ризику до прийнятного значення. На цьому етапі приймаються управлінські рішення щодо подальших дій стосовно кожного ідентифікованого ризику з урахуванням його рівня, можливих наслідків та пріоритетності.

Оброблення ризиків передбачає вибір відповідної стратегії реагування, які наведені в таблиці 1.1 [3].

Таблиця 1.1

Стратегії реагування на ризики інформаційної безпеки

Варіант обробки	Заходи
Прийняття	Усвідомлене визнання можливості настання несприятливої події та готовність покрити пов'язані з нею втрати власними фінансовими ресурсами
Передача	Делегування відповідальності за наслідки ризику стороннім суб'єктам шляхом укладення відповідних договірних угод або придбання страхового покриття
Ухилення	Повна нейтралізація загрози або її джерела через виключення умов, за яких вона може виникнути
Зниження	Обмеження ймовірності реалізації загрози або мінімізація обсягу потенційних збитків від її настання без безпосереднього усунення джерела небезпеки

Вибір конкретної стратегії залежить від бізнес-цілей організації, доступних ресурсів, вимог безпеки та рівня допустимого ризику. У деяких

випадках організація може комбінувати декілька підходів для досягнення оптимального результату.

Результатом даного етапу є формування плану оброблення ризиків, який містить перелік заходів безпеки, відповідальних осіб, строки реалізації та критерії оцінювання ефективності впроваджених рішень. Такий план забезпечує системний підхід до управління ризиками та слугує основою для подальшого контролю, моніторингу та вдосконалення системи інформаційної безпеки.

5. Прийняття ризиків.

На цьому етапі організація визначає подальшу долю ризиків, що залишаються після реалізації заходів опрацювання. Підставою для прийняття є відповідність таких ризиків заздалегідь встановленим критеріям та допустимому порогу.

Якщо досягти прийнятного рівня ризику неможливо або економічно невиправдано, організація може обрати один із таких шляхів:

- запровадити додаткові компенсуючі заходи (посилений моніторинг, резервування ресурсів, розширення контролів);
- обґрунтувати виправданість прийняття ризику з огляду на можливі наслідки.

Рішення про прийняття ризику в обов'язковому порядку фіксується в документації та затверджується керівництвом, що забезпечує чіткий розподіл відповідальності за встановлений рівень ризику.

6. Моніторинг і перегляд

Завершальна стадія циклу управління ризиками інформаційної безпеки. Її призначення - підтримувати безперервний контроль за динамікою ризиків, актуалізувати їх оцінку та своєчасно коригувати захисні заходи у відповідь на зміни операційного середовища.

У разі зниження ефективності наявних механізмів захисту або виявлення нових загроз здійснюється їх перегляд, що може передбачати впровадження додаткових контролів, посилення існуючих заходів або вдосконалення процесів управління ризиками.

За результатами формується звітність, що містить реєстр виявлених ризиків, їхні рівні, відомості про застосовані заходи опрацювання та прийняті управлінські рішення.

Принципово важливо, що моніторинг і перегляд ризиків є не разовою процедурою, а безперервним процесом, зумовленим постійною мінливістю інформаційного середовища. З огляду на це організаціям рекомендується проводити регулярний перегляд не рідше одного разу на рік.

Отже, система управління ризиками має бути гнучкою та здатною адаптуватися до технологічних змін, еволюції загроз і трансформацій бізнес-середовища, що є запорукою її тривалої ефективності.

1.2 Сучасні методи, моделі та засоби оцінювання ризиків інформаційної безпеки

1.2.1 Методи оцінювання ризиків інформаційної безпеки

Актуальні методи аналізу ризиків інформаційної безпеки передбачають інтеграцію якісного та кількісного оцінювання [6]. Кількісний підхід орієнтований на отримання числового вираження ризику, а саме - визначення рівня окремих загроз і сукупного ризику в межах проєкту. Якісний підхід натомість застосовується для виявлення ключових чинників ризику, їхніх характеристик, діапазону прояву та систематизації потенційних загроз [7].

Завданням *якісного аналізу ризику* є ідентифікація можливих ризиків, визначення їх причин і факторів виникнення, а також оцінка потенційних наслідків для діяльності організації. Це дозволяє встановити зони підвищеного ризику та своєчасно врахувати можливі негативні впливи.

У якісній оцінці виділяють експертний метод, метод аналізу доцільності витрат і метод аналогій [6].

В основі *експертного методу* лежить узагальнення суджень спеціалістів, які мають відповідну теоретичну підготовку та практичний досвід у досліджуваній галузі. Кожен із залучених фахівців отримує індивідуальний перелік можливих ризиків і самостійно оцінює ймовірність їх виникнення.

Отримані дані перевіряються на узгодженість між собою. Допустимим вважається розходження між оцінками двох спеціалістів щодо одного ризику не більше ніж на 50 %, що забезпечує зниження суб'єктивного впливу та запобігає принципово протилежним висновкам щодо ймовірності настання ризикових подій [6]. У результаті застосування методу формуються експертні оцінки рівня допустимого або критичного ризику, а також прогнозуються найбільш імовірні втрати. Найчастіше експертний метод використовується на початкових етапах аналізу ризиків для їх виявлення та попереднього оцінювання [6].

Метод аналізу доречності витрат спрямований на виявлення областей підвищеного ризику та застосовується з метою захисту капіталу організації від надмірних фінансових втрат [6]. Серед основних чинників, що зумовлюють перевищення запланованих витрат, виділяють чотири групи [8]:

1) занижена початкова оцінка вартості проєкту;

2) корективи, внесені у процесі його виконання;

3) невідповідність фактичних технічних характеристик запланованим;

4) несприятливий вплив макроекономічного середовища.

Послідовне фінансування за етапами дає змогу оперативно виявляти ознаки зростання ризиків та своєчасно ухвалювати рішення про коригування або припинення реалізації проєкту.

Метод аналогії - ґрунтується на вивченні досвіду виконання організацією схожих проєктів у попередні періоди з метою оцінювання імовірності виникнення фінансових втрат [8]. Цей підхід може застосовуватися як на окремих стадіях реалізації проєкту, так і впродовж усього його життєвого циклу.

Кількісна оцінка ризику дає змогу отримати найбільш точні рішення порівняно з якісною оцінкою. Цей аналіз передбачає чисельне визначення величин окремих ризиків і ризику проєкту загалом. Він базується на теорії ймовірностей, математичній статистиці, теорії досліджень операцій. У практиці оцінювання ризиків найбільш поширеними є такі методи кількісного аналізу [6] :

1) коригування норми дисконту;

2) аналіз чутливості показників ефективності;

3) сценарний аналіз;

4) побудова дерева рішень;

5) імітаційне моделювання (метод Монте-Карло).

Метод коригування норми дисконту передбачає перерахунок майбутніх грошових потоків у поточну вартість із застосуванням підвищеної ставки. Суть підходу полягає у збільшенні вихідної безризикової або мінімально прийнятної ставки дисконтування на величину ризикової надбавки. Така надбавка додається до базового показника і формує скориговану норму дисконту. Розмір ризикової надбавки зазвичай встановлюється на основі експертних оцінок і визначається специфікою реалізації проєкту, кон'юнктурою ринку та рядом інших чинників. Метод вирізняється простотою застосування і є поширеним інструментом на практиці, однак не забезпечує глибокого аналізу ризику, а лише інтегрально відображає його вплив [6].

Аналіз чутливості спрямований на встановлення того, як зміна окремих вхідних параметрів позначається на ключових показниках ефективності проєкту - чистій приведеній вартості (NPV), внутрішній нормі рентабельності (IRR) та строку окупності. В основі методу лежить послідовна варіація одного з чинників (наприклад, темпу інфляції або обсягу витрат) за незмінності решти, що дає змогу оцінити чутливість кінцевого результату до кожного з них. Таким чином вдається виокремити найбільш значущі параметри, що визначають успішність реалізації проєкту [6].

Сценарний метод ґрунтується на побудові прогнозів розвитку зовнішнього середовища та розрахунку показників інвестиційної ефективності для кожного з них. Як правило, формуються три варіанти: несприятливий, базовий і сприятливий, що дозволяє орієнтовно оцінити діапазон можливих результатів проєкту та його фінансові наслідки за різних економічних умов [6].

Метод дерева рішень передбачає побудову графічної схеми, що відображає можливі варіанти перебігу подій та відповідні управлінські рішення

на кожному етапі. На відміну від сценарного підходу, він явно моделює активну роль організації у виборі дій у процесі реалізації проєкту [6].

Метод Монте-Карло належить до інструментів імітаційного моделювання і передбачає багаторазове відтворення випадкових сценаріїв для оцінювання сукупного впливу невизначених чинників на результати проєкту. Перевагою методу є можливість одночасного варіювання кількох параметрів, а обчислення здійснюються на основі математичних моделей ефективності з використанням спеціалізованого програмного забезпечення [6, 7].

1.2.2 Моделі та методології управління ризиками інформаційної безпеки

У процесі управління ризиками інформаційної безпеки використовуються різні методології, які дозволяють систематизувати підхід до ідентифікації, аналізу та оцінювання ризиків. До найбільш поширених належать ISO/IEC 27005, NIST Risk Management Framework, OCTAVE, COBIT. Кожна з цих моделей має свої особливості застосування та рівень деталізації процесів оцінювання ризиків.

Одним з базових і найбільш поширених є стандарт *ISO/IEC 27005* [2], який визначає загальні принципи та рекомендації щодо управління ризиками інформаційної безпеки. Він передбачає поетапний підхід до процесу оцінювання ризиків, що включає встановлення контексту, ідентифікацію, аналіз, оцінювання та оброблення ризиків, а також подальший моніторинг і перегляд .

Основні етапи вже були розглянуті у підрозділі 1.1, де детально описано його логіку процесу управління ризиками відповідно.

Стандарт NIST 800-30 є одним із найбільш визнаних інструментів оцінювання ризиків інформаційної безпеки, створеним Національним інститутом стандартів і технологій США. В основі методики лежить аналіз двох ключових складових: імовірності настання загрози та масштабу можливих наслідків. Процедура оцінювання охоплює низку послідовних кроків - визначення активів, загроз і вразливостей, дослідження наявних захисних

механізмів, встановлення рівня ймовірності та впливу, обчислення показника ризику і формування відповідних заходів реагування [9, 16].

Відмінною рисою стандарту є чітка структурованість і широкі можливості для пристосування до потреб організацій різного профілю та розміру. NIST 800-30 підтримує як якісний, так і кількісний підходи до аналізу, що надає аналітикам гнучкість у виборі інструментарію залежно від наявних даних і специфіки організації. Водночас методика передбачає не разове, а безперервне управління ризиками з обов'язковим переглядом оцінок у разі трансформації операційного середовища, появи нових векторів атак або впровадження сучасних інформаційних технологій [16].

Рис. 1.2.1. Алгоритм NIST 800-30 [9]

Метод OCTAVE розроблений в Університеті Карнегі-Меллон (США) та призначений для оцінювання ризиків через аналіз критичних активів, загроз і вразливостей. Особливістю методики є те, що вона виконується переважно внутрішніми силами організації без залучення зовнішніх експертів. Процес оцінки ризиків за OCTAVE складається з трьох основних етапів [11]:

1) формування профілю загроз і оцінка активів;
2) аналіз інфраструктурних вразливостей;
3) розробка стратегії безпеки та планів оброблення ризиків.

На першому етапі визначаються критичні активи та пов’язані з ними загрози. На другому - аналізуються технічні та організаційні вразливості інфраструктури. На третьому етапі виконується оцінка ризиків і формуються заходи щодо їх зниження або прийняття.

Оцінювання ризику в OCTAVE переважно якісне і базується на можливих наслідках реалізації загроз, включаючи фінансові втрати, шкоду репутації та вплив на діяльність організації [11].

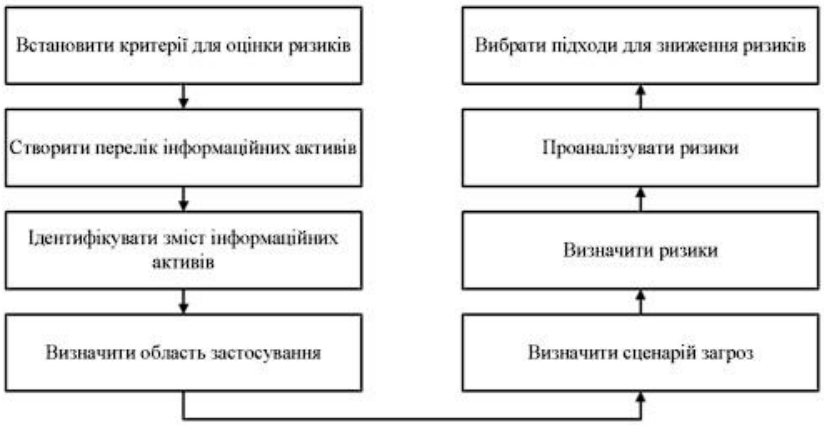


Рис. 1.2.2. Алгоритм OCTAVE [9]

Широко відомим методом управління ризиками є *COBIT 5 for Risk*, створений асоціацією ISACA на основі міжнародних стандартів у цій галузі. Методологія розглядає ризики інформаційної безпеки не відокремлено, а як невід'ємну складову загальних бізнес-ризиків організації, та пропонує підходи до їх дослідження й опрацювання [12, 17].

Ключовим елементом методології є апарат ризикових сценаріїв - описів подій, реалізація яких може перешкодити досягненню стратегічних цілей організації. COBIT містить розгалужену бібліотеку типових сценаріїв (більше ста), що охоплюють широкий спектр напрямів: IT-інфраструктуру, кадрові питання, програмне забезпечення, кіберзагрози, а також природні та техногенні чинники [12, 17]. Структура методології наведена на рис. 1.2.3.

Рис. 1.2.3. Структура методології COBIT 5 for Risk [12]

Кожен сценарій містить відомості про джерело загрози, характер події, задіяні активи та можливі наслідки для організації. На підставі сценарного аналізу обирається стратегія опрацювання ризику: його уникнення, зниження, передача третій стороні або свідоме прийняття. Подальша робота з ризиком передбачає оцінку залишкового рівня та, за потреби, впровадження додаткових захисних заходів. Відповідні рекомендації проілюстровано на рис. 1.2.4.

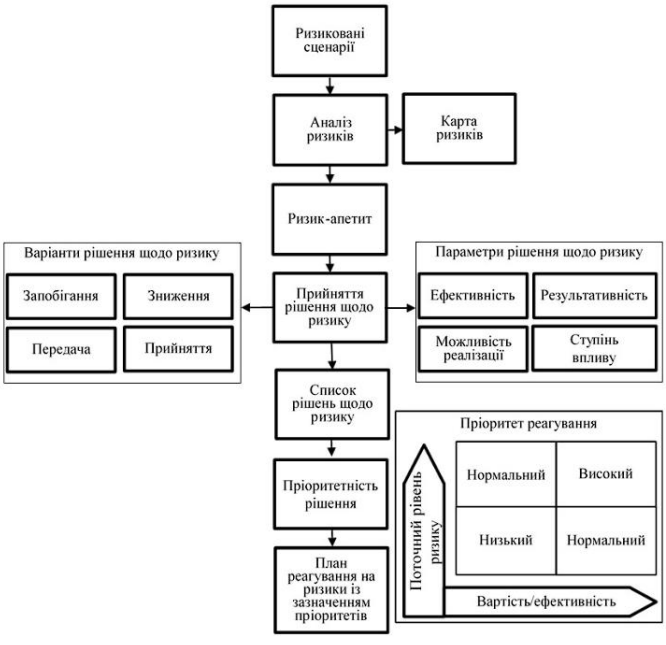


Рис. 1.2.4. Структура методології COBIT 5 for Risk [12]

Методологія COBIT є найбільш ефективною для великих організацій із високим рівнем залежності від інформаційних технологій, оскільки дозволяє інтегрувати управління ризиками інформаційної безпеки з загальною системою управління IT та підвищити ефективність захисту.

1.2.3 Методи з програмними комплексами для оцінювання ризиків

CRAMM є однією з перших формалізованих методик аналізу ризиків інформаційної безпеки, розробленою Службою безпеки Великобританії на основі стандартів BS7799 (нині ISO 27000). Вона поєднує якісні та кількісні підходи.[9].

Процес оцінки ризиків за CRAMM є структурованим і включає три основні етапи [10]:

1) ідентифікація та оцінка IT-активів;
2) аналіз загроз і вразливостей;
3) розрахунок ризику та вибір заходів захисту.

На відміну від попередніх методологій, CRAMM має програмну реалізацію. Спеціалізовані програмні засоби дозволяють автоматизувати процес оцінювання ризиків, включаючи збір даних, розрахунок показників та формування рекомендацій. Дане рішення є універсальним і придатним як для великих, так і для малих організацій, як державного, так і комерційного сектору [9].

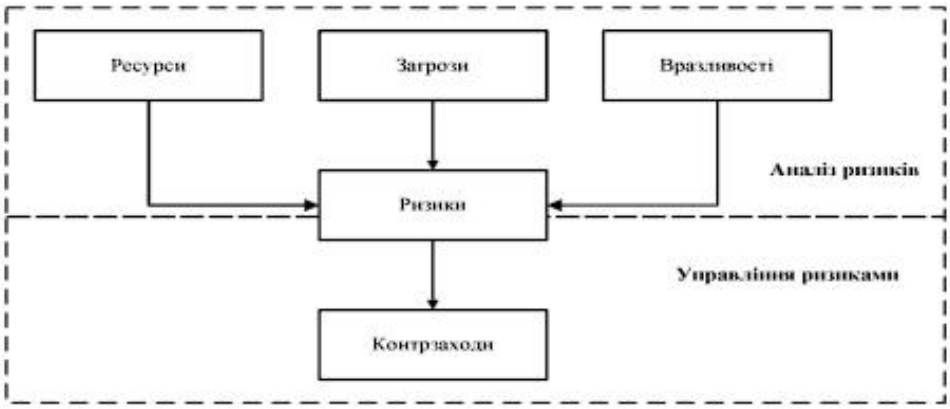


Рис. 1.2.5. Алгоритм CRAMM [9]

Серед методик із програмною підтримкою процесу оцінювання ризиків інформаційної безпеки варто також виділити *метод MEHAR*I, створений асоціацією CLUSIF. Його концепція побудована на забезпеченні всебічного управління ризиками та передбачає упорядкований підхід до їх виявлення, дослідження та опрацювання. Методика включає широкий спектр інструментів: структуровані бази даних, матриці відповідності, градуйовані шкали оцінювання, аналітичні моделі та анкети для збору інформації. Здебільшого ці інструменти реалізуються у формі автоматизованих або частково автоматизованих рішень — зокрема, у вигляді електронних таблиць - що забезпечує певний рівень автоматизації аналітичної роботи [18].

У межах методу MEHARI процедура оцінювання ризиків охоплює встановлення поточних загроз, визначення переліку активів і ступеня їх важливості, а також формування комплексу захисних заходів, спрямованих на недопущення інцидентів або обмеження їх негативних наслідків. Відмінною рисою методу є застосування сценарного моделювання, яке уможливлює відтворення різних варіантів розвитку загроз і поглиблений аналіз їх потенційного впливу на операційну діяльність організації [13, 18].

Метод спирається на засади системного підходу та реалізований за модульним принципом, що забезпечує логічну послідовність, відтворюваність та гнучкість процесу оцінювання ризиків. На рисунку 1.2.6 наведено структуру модулів MEHARI та основні підходи до проведення оцінювання ризиків інформаційної безпеки.

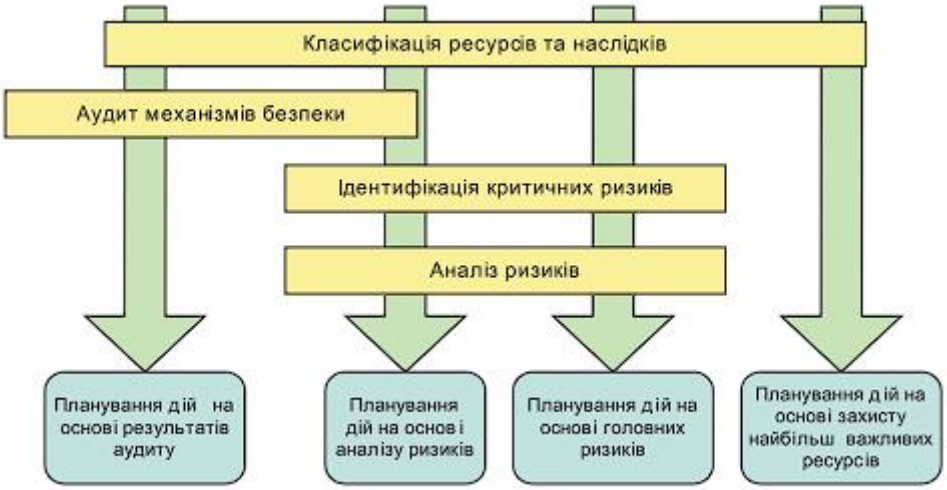


Рис. 1.2.6. Модулі MEHARI та підходи до проведення оцінювання ризиків [13]

Методологія RiskWatch розроблена однойменною компанією та реалізується у вигляді сімейства програмних засобів для оцінювання ризиків інформаційної безпеки [15]. Орієнтована на кількісну оцінку ризиків і базується на аналізі очікуваних річних втрат (Annual Loss Expectancy, ALE) та оцінці економічної ефективності заходів захисту (Return on Investment, ROI). Основною метою є визначення співвідношення між потенційними втратами від реалізації загроз та витратами на впровадження засобів захисту [14].

Процес оцінювання ризиків у RiskWatch складається з чотирьох основних етапів [14]:

Етап 1. Визначається предмет дослідження: тип організації, склад інформаційної системи, категорії активів, загроз і можливих втрат.

Етап 2. Здійснюється збір детальних даних про систему, включаючи ресурси, інциденти та вразливості, що можуть визначатися за допомогою великої бази питань і довідкових даних.

Етап 3. Передбачає кількісну оцінку ризиків, яка базується на розрахунку очікуваних втрат.

Етап 4. Включає формування звітності, яка охоплює оцінку ризиків, вартість активів, перелік загроз, а також розрахунок економічної ефективності впроваджених засобів захисту.

Додатково методологія RiskWatch передбачає використання сценарного аналізу типу, який дозволяє оцінити вплив впровадження різних заходів захисту на рівень ризику. На основі порівняння очікуваних втрат до і після впровадження контрзаходів визначається їх ефективність та доцільність застосування. Такий підхід забезпечує можливість не лише оцінити поточний рівень ризиків, але й обґрунтувати управлінські рішення щодо інвестування у систему інформаційної безпеки.

1.3 Порівняльний аналіз існуючих методів, моделей та засобів оцінювання ризиків інформаційної безпеки

Розглянуті у попередніх підрозділах міжнародні стандарти, методи та програмні комплекси, які відрізняються підходами до аналізу, рівнем формалізації та можливостями практичного застосування. Кожен із них має свої переваги, обмеження та орієнтований на різні умови використання - від якісного експертного оцінювання до кількісного аналізу із застосуванням спеціалізованого програмного забезпечення.

У цьому підрозділі буде проведено порівняльний аналіз розглянутих підходів з метою визначення їх ефективності, універсальності та доцільності використання в різних умовах. Для цього буде сформовано систему критеріїв, за якими здійснюватиметься оцінювання методів, моделей і засобів, що дозволить виявити їх сильні та слабкі сторони, а також обґрунтувати вибір найбільш придатного рішення для подальшого застосування.

Для проведення порівняльного аналізу методів і засобів оцінювання ризиків інформаційної безпеки використано систему критеріїв, для кожного з яких введено умовні позначення у вигляді числової шкали. Такий підхід дозволяє зменшити обсяг текстової інформації в узагальнюючій таблиці та забезпечити зручність порівняння результатів. Кожному значенню шкали відповідає певний рівень прояву критерію, що визначається на основі характеристик відповідного методу або засобу. Для всіх критеріїв застосовано трирівневу шкалу оцінювання.

Таблиця 1.3.1

Тип оцінювання ризику

Значення	*Характеристика*
1	Якісне оцінювання
2	Кількісне оцінювання
3	Змішане оцінювання

Таблиця 1.3.2

Джерела вхідних даних

Значення	*Характеристика*
1	Внутрішні дані (внутрішні звіти, результати перевірок та аудитів безпеки)
2	Гібридні дані (внутрішня інформація доповнюється зовнішніми статичними джерелами)
3	OSINT-орієнтовані дані (використання відкритих джерел та актуальної інформації про загрози з зовнішнього середовища)

Таблиця 1.3.3

Можливість автоматизації процесу

Значення	*Характеристика*
1	Відсутня (процес виконується вручну)
2	Часткова автоматизація (окремі етапи підтримуються інструментами)
3	Повна автоматизація (використання спеціалізованого ПЗ)

Таблиця 1.3.4

Адаптивність до різних типів організацій

Значення	*Характеристика*
1	Обмежена (орієнтована на вузьку сферу застосування)
2	Середня (може використовуватись у різних організаціях після налаштування та адаптації)
3	Висока (універсальний підхід, придатний для організацій різних масштабів і сфер діяльності)

Таблиця 1.3.5

Складність впровадження та використання

Значення	*Характеристика*
1	Низька (простий у впровадженні, мінімальні ресурси)
2	Середня (потребує підготовки та певних ресурсів)
3	Висока (складний, потребує значних ресурсів і кваліфікованих фахівців)

Для зручності представлення результатів порівняльного аналізу та зменшення обсягу текстової інформації в узагальнюючій таблиці, для кожного з обраних критеріїв введено умовні позначення. Це дозволяє компактно відобразити характеристики методів і засобів оцінювання ризиків та забезпечити наочність їх порівняння.

Використані умовні позначення критеріїв:

1) Тип оцінювання ризику – ТО

2) Джерела вхідних даних – ВД

3) Можливість автоматизації процесу – АП

4) Адаптивність до різних типів організацій – АТ

5) Складність впровадження та використання – СВ

На основі визначених критеріїв та відповідних шкал оцінювання далі формується узагальнююча таблиця порівняльного аналізу методів, моделей і засобів оцінювання ризиків інформаційної безпеки.

Таблиця 1.3.6

Порівняльна таблиця

Метод / модель / засіб	ТО	ВД	АП	АТ	СВ
ISO/IEC 27005	3	2	2	3	2
NIST 800-30	3	2	2	3	2
OCTAVE	1	1	1	2	2
COBIT 5 for Risk	3	2	2	3	3
CRAMM	3	2	3	3	3
MEHARI	3	2	2	3	2
RiskWatch	2	2	3	2	3

Проведений порівняльний аналіз методів, моделей та засобів оцінювання ризиків інформаційної безпеки показав, що сучасні підходи суттєво відрізняються за типом оцінювання, джерелами вхідних даних, рівнем автоматизації та складністю впровадження. Найбільш універсальними є методології ISO/IEC 27005, NIST 800-30, COBIT 5 for Risk та MEHARI, які поєднують якісні та кількісні підходи, мають достатній рівень адаптивності до різних типів організацій та забезпечують системний підхід до управління ризиками.

Метод OCTAVE характеризується простотою реалізації та орієнтацією на внутрішні ресурси організації, однак має обмеження щодо автоматизації та використання зовнішніх джерел даних. У свою чергу CRAMM та RiskWatch, забезпечують більш високий рівень автоматизації процесів оцінювання ризиків та дозволяють отримувати формалізовані результати, проте потребують значних ресурсів і характеризуються підвищеною складністю впровадження.

Аналіз джерел вхідної інформації показує, що переважна більшість розглянутих методів ґрунтується на внутрішніх даних організації або їх поєднанні з обмеженими зовнішніми джерелами. При цьому використання відкритих зовнішніх джерел розвідки (OSINT) не реалізоване. Такий підхід ускладнює врахування динамічних змін у сучасному середовищі та знижує оперативність реагування на нові ризики.

Отже, кожен із розглянутих методів має власну сферу застосування та повинен обиратися з урахуванням специфіки організації, доступних ресурсів і цілей оцінювання ризиків. Водночас проведений аналіз дозволяє зробити важливий висновок про доцільність інтеграції традиційних методів оцінювання ризиків із сучасними підходами, зокрема використанням OSINT, що відкриває перспективи для підвищення актуальності та ефективності процесу управління ризиками інформаційної безпеки у подальших дослідженнях.

Висновки до розділу 1

У першому розділі було розглянуто основні етапи процесу оцінювання ризиків інформаційної безпеки, міжнародні стандарти та методології управління ризиками, а також проведено їх порівняльний аналіз. Детально проаналізовано підходи, що використовуються на різних етапах оцінювання ризиків, включаючи ідентифікацію, аналіз, оцінку та оброблення ризиків, а також розглянуто їх реалізацію в межах провідних міжнародних стандартів.

У ході дослідження встановлено, що сучасні підходи до оцінювання ризиків інформаційної безпеки відрізняються за типом оцінювання, рівнем формалізації, можливістю автоматизації та складністю впровадження.

Міжнародні стандарти та методології забезпечують системний підхід до управління ризиками, проте мають різні сфери застосування та орієнтовані на різні організаційні умови.

Проведений порівняльний аналіз показав, що найбільш універсальними є підходи, які поєднують якісне та кількісне оцінювання ризиків, підтримують часткову або повну автоматизацію процесів та можуть адаптуватися до організацій різних типів і масштабів. Водночас методики з високим рівнем автоматизації потребують значних ресурсів, спеціалізованого програмного забезпечення та відповідної кваліфікації персоналу.

Встановлено, що більшість розглянутих методів базуються переважно на внутрішніх даних організації або їх поєднанні з обмеженими зовнішніми джерелами інформації. При цьому використання відкритих джерел розвідки (OSINT) у традиційних підходах практично не реалізоване.

Таким чином, результати проведеного аналізу свідчать про необхідність удосконалення існуючих підходів до оцінювання ризиків інформаційної безпеки шляхом інтеграції сучасних зовнішніх джерел даних та засобів автоматизованого збору інформації. Це дозволить підвищити актуальність, адаптивність та ефективність процесів управління ризиками інформаційної безпеки, що обумовлює доцільність подальшого дослідження можливостей використання OSINT-технологій у системах оцінювання ризиків.

РОЗДІЛ 2 РОЗРОБКА МОДЕЛІ СИСТЕМИ ОЦІНЮВАННЯ РИЗИКІВ НА ОСНОВІ ІНТЕГРАЦІЇ ВІДКРИТИХ ДАНИХ

2.1 Класифікація OSINT-даних та джерел отримання інформації

Суттєва частка сучасних кібератак реалізується саме на основі даних, здобутих із загальнодоступних ресурсів. Зловмисники активно використовують методи OSINT для отримання відомостей про цифрову інфраструктуру цілі, її персонал, застосовувані технології та наявні сервіси [20]. Разом із тим самі організації можуть використовувати ті самі інструменти для самостійного виявлення вразливих місць, контролю власного цифрового профілю та підвищення якості процесу управління ризиками інформаційної безпеки.

OSINT - це процес збору, аналізу та інтерпретації інформації з відкритих джерел, доступних без порушення законодавства або спеціальних механізмів доступу. До таких джерел належать офіційні вебсайти організацій, соціальні мережі, пошукові системи, доменні та DNS-реєстри, публічні бази витоків даних, форуми, сервіси моніторингу мережевої інфраструктури та інші відкриті інформаційні ресурси [19].

Перш ніж класифікувати OSINT-дані, доцільно розглянути основні джерела отримання відкритої інформації [21, 22, 23]:

1. Соціальні мережі.

LinkedIn, Facebook, X (Twitter), Instagram є джерелом інформації про персонал організації та структуру компанії. Профілі співробітників можуть містити відомості про програмне забезпечення, мережеву інфраструктуру, професійні навички та поточні проєкти, що може використовуватися для проведення фішингових атак або соціальної інженерії.

2. Офіційні вебсайти організації.

Містять інформацію про структуру організації, контактні дані, партнерів, сервіси, технології та програмні продукти. Аналіз HTML-коду, документів або структури вебсайту дозволяє отримати додаткові технічні відомості про серверне програмне забезпечення, CMS-системи та використовувані технології.

3. DNS та WHOIS-сервіси.

Ці сервіси дозволяють отримати інформацію про домени організації, IP-адреси, субдомени, поштові сервери та мережеву інфраструктуру. Такі дані можуть використовуватися для побудови карти зовнішньої інфраструктури організації та виявлення потенційних точок атаки.

4. Пошукові системи.

Google, Bing та спеціалізовані пошукові оператори дають можливість знаходити документи, резервні копії, відкриті каталоги, сторінки авторизації та іншу службову інформацію, яка випадково потрапила у відкритий доступ

5. Платформи спільної розробки.

GitHub, GitLab, Bitbucket: можуть містити відкриті репозиторії програмного коду, конфігураційні файли, API-ключі, токени доступу та службову документацію.

6. Бази витоків даних.

Have I Been Pwned, DeHashed та інші сервіси містять інформацію про витоки облікових записів, електронних адрес та паролів. Такі дані можуть свідчити про компрометацію корпоративних облікових записів та підвищення ризику несанкціонованого доступу.

7. Сервіси моніторингу мережевої інфраструктури.

Shodan, Censys, ZoomEye: дозволяють виявляти доступні з Інтернету сервери, відкриті порти, вебсервіси, сертифікати та мережеве обладнання організації. За допомогою таких сервісів можна отримати інформацію про використовувані технології та потенційно вразливі сервіси.

8. Новинні ресурси, форуми та блоги.

Новинні сайти, тематичні форуми, блоги та професійні спільноти можуть містити інформацію про інциденти безпеки, витоки даних, репутаційні проблеми або обговорення діяльності організації. Аналіз таких ресурсів дозволяє виявляти потенційні репутаційні ризики та ознаки інформаційних атак.

9. Платформи Threat Intelligence та бази вразливостей.

CVE Details, MITRE CVE, CERT-UA, AlienVault OTX та інші ресурси містять інформацію про актуальні вразливості, індикатори компрометації (IOC), шкідливі IP-адреси та сучасні кіберзагрози. Використання таких джерел дозволяє оцінювати актуальність загроз для організації та своєчасно виявляти потенційно небезпечні сервіси або програмне забезпечення.

10. Картографічні сервіси.

Google Maps, Google Street View, OpenStreetMap: можуть містити інформацію про фізичне розташування організації, структуру будівель, розміщення офісів, серверних приміщень, систем контролю доступу та інших об’єктів інфраструктури.

Розглянуті джерела формують значний обсяг різнорідної інформації про організацію, її інфраструктуру, персонал та цифрові ресурси. Проте отримані дані відрізняються за змістом, рівнем деталізації, способом отримання та потенційним впливом на інформаційну безпеку. У зв’язку з цим виникає необхідність їх систематизації та класифікації, що дозволяє структурувати OSINT-дані відповідно до їх призначення та ролі в процесі оцінювання ризиків.

OSINT-дані, що використовуються в процесі оцінювання ризиків інформаційної безпеки, можуть бути класифіковані за типом інформації, яку вони містять. Такий підхід дозволяє систематизувати відкриті дані відповідно до їх призначення та спростити подальший аналіз потенційних загроз і вразливостей організації. Основні категорії OSINT-даних наведені нижче [22, 23]:

1. Технічні дані.

До цієї категорії належать відомості про мережеву та інформаційну інфраструктуру організації: IP-адреси, домени, субдомени, DNS-записи, відкриті порти, вебсервери, SSL-сертифікати, використовувані CMS, операційні системи та мережеві сервіси. Такі дані дозволяють визначити зовнішню поверхню атаки організації та виявити потенційно вразливі компоненти інфраструктури.

2. Дані про персонал організації.

Містять інформацію про співробітників, керівництво, адміністраторів систем та IT-фахівців. Джерелами можуть бути соціальні мережі, професійні платформи та публічні документи. Дані можуть включати посади, електронні адреси, професійні навички, використовувані технології та службові контакти. Аналіз такої інформації дозволяє оцінити ризики соціальної інженерії, фішингових атак та компрометації облікових записів.

3. Дані про програмне забезпечення та технології.

До цієї категорії належать відомості про використовувані організацією програмні продукти, вебтехнології, фреймворки, серверне програмне забезпечення, системи керування базами даних та хмарні сервіси. Подібна інформація може бути отримана шляхом аналізу вебресурсів, метаданих або відкритих репозиторіїв програмного коду. Її використання дозволяє визначати потенційні вразливості та застарілі компоненти системи.

4. Облікові та ідентифікаційні дані.

Включають електронні адреси, логіни, облікові записи, номери телефонів та інші ідентифікатори, що можуть бути пов’язані з організацією. Частина таких даних може бути виявлена у відкритих витоках інформації або публічних сервісах. Їх аналіз дозволяє оцінити ризик компрометації облікових записів та повторного використання паролів.

5. Документація.

До них належать документи, опубліковані у відкритому доступі: PDF-файли, презентації, звіти, технічна документація, тендерна документація та службові файли. Вони можуть містити імена користувачів, структуру каталогів, версії програмного забезпечення та іншу службову інформацію, яка може бути використана під час підготовки атаки.

6. Дані про репутацію та інформаційний фон.

Ця категорія охоплює згадки про організацію у новинах, соціальних мережах, форумах, блогах та тематичних спільнотах. Аналіз інформаційного фону дозволяє виявляти витоки інформації, конфлікти, шахрайські кампанії або репутаційні ризики, що можуть впливати на діяльність організації.

7. Геопросторові та картографічні дані.

Містять інформацію про фізичне розташування об’єктів організації, офісів, дата-центрів та технічної інфраструктури. Джерелами можуть бути картографічні сервіси, супутникові знімки, фотографії та геотеги у соціальних мережах. Такі дані можуть використовуватися для аналізу фізичної безпеки об’єктів та виявлення додаткових ризиків.

8. Дані про кіберзагрози та інциденти.

Включають інформацію про відомі вразливості, індикатори компрометації (IOC), витоки даних, шкідливі IP-адреси, домени та активність кіберзлочинних угруповань. Джерелами таких даних є спеціалізовані OSINT-платформи, бази CVE, CERT-звіти та сервіси threat intelligence. Використання цих даних дозволяє підвищити актуальність оцінювання ризиків та враховувати сучасний ландшафт кіберзагроз.

Для узагальнення розглянутих джерел відкритої інформації та відповідних категорій OSINT-даних доцільно сформувати таблицю, яка відображає взаємозв’язок між джерелами інформації, типами отримуваних даних та потенційними ризиками інформаційної безпеки. Такий підхід дозволяє наочно продемонструвати, які саме відкриті ресурси можуть бути використані для збору критично важливої інформації про організацію та яким чином це може впливати на рівень її захищеності.

Таблиця 2.1.1

Джерела OSINT-даних, типи інформації та можливі ризики

Джерело інформації	Тип OSINT-даних	Можливі загрози
Соціальні мережі	Дані про персонал організації	Соціальна інженерія, фішинг, компрометація облікових записів
Офіційні вебсайти організації	Дані про програмне забезпечення та технології	Виявлення вразливостей вебсервісів, підготовка цільових атак
DNS та WHOIS-сервіси	Технічні дані	Атаки на мережевий периметр, перехоплення DNS-запитів, сканування прихованих сервісів.

Продовження таблиці 2.1.1

Джерело інформації	Тип OSINT-даних	Можливі загрози
Пошукові системи	Документи	Витік службової інформації, розкриття внутрішньої структури
Платформи спільної розробки	Дані про програмне забезпечення та технології	Компрометація сервісів, витік вихідного коду та ключів доступу
Бази витоків даних	Облікові та ідентифікаційні дані	Атаки методом перебору паролів, несанкціонований доступ до корпоративних акаунтів.
Сервіси моніторингу мережевої інфраструктури	Технічні дані	Експлуатація відкритих портів, атаки на незахищені пристрої IoT.
Новинні ресурси, форуми та блоги	Дані про репутацію та інформаційний фон	Репутаційні втрати, розголошення інформації про інциденти
Threat Intelligence платформи, CVE та CERT-бази	Дані про кіберзагрози та інциденти	Використання актуальних вразливостей та підвищення рівня загроз
Картографічні сервіси	Геопросторові та картографічні дані	Аналіз фізичної безпеки та ризиків фізичного доступу

2.2 Інтеграція OSINT-даних у процес оцінювання ризиків інформаційної безпеки на основі методології NIST 800-30

2.2.1 Аналіз методології оцінювання ризиків NIST 800-30

Методологія NIST SP 800-30 [16] є складовою загальної системи управління ризиками інформаційної безпеки, визначеної стандартом NIST SP 800-39 [25]. На відміну від NIST SP 800-39, який описує загальні принципи управління ризиками, стандарт NIST SP 800-30 зосереджується безпосередньо на процесі оцінювання ризиків інформаційної безпеки. Документ містить 3 розділи, однак у межах даної роботи будуть розглянуті другий розділ «Основи» та третій розділ «Процес», оскільки саме вони описують базові принципи оцінювання ризиків і послідовність виконання цього процесу.

У другому розділі стандарту розглядаються фундаментальні поняття такі як основні компоненти ризику, підходи до аналізу та місце оцінювання у

загальному процесі управління ризиками організації. Процес управління ризиками включає такі етапи [16]:

Формування контексту ризику - визначення середовища, у межах якого організація приймає рішення щодо інформаційної безпеки. На цьому етапі формується стратегія управління ризиками, визначаються критерії оцінювання, рівень допустимого ризику та загальні підходи до реагування на загрози.

Оцінювання ризику - етап спрямований на виявлення загроз, вразливостей, можливих наслідків їх реалізації та визначення ймовірності виникнення негативних подій. Результатом цього процесу є визначення рівня ризику, який зазвичай розглядається як комбінація потенційного впливу та ймовірності реалізації загрози.

Реагування на ризик - організація визначає можливі варіанти реагування, аналізує їх ефективність, обирає оптимальні заходи відповідно до допустимого рівня ризику та реалізує обрані механізми захисту.

Моніторинг ризику - передбачає постійний моніторинг ризиків і контроль ефективності впроваджених заходів безпеки.

Схему процесу управління ризиками відповідно до NIST наведено на рис. 2.2.1.

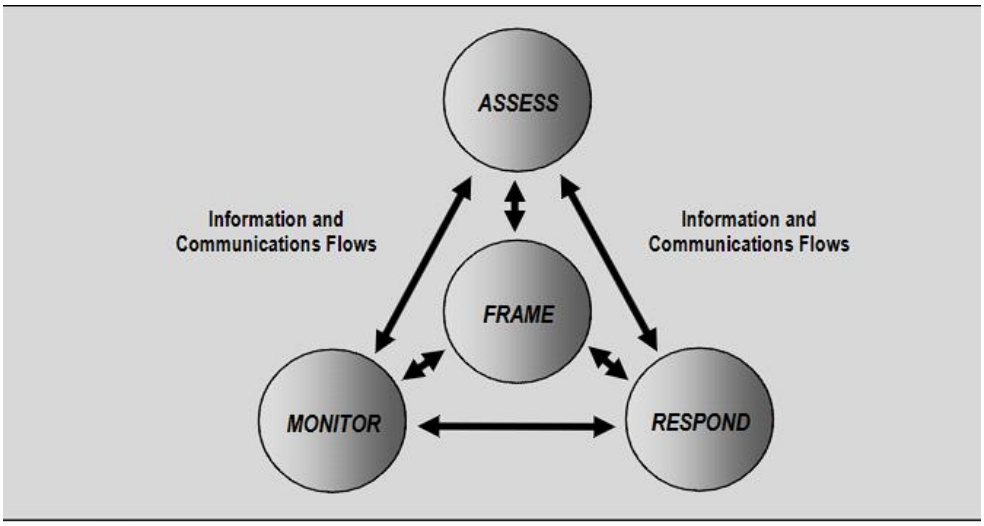


Рис. 2.2.1. Процес управління ризиками

У методології оцінювання ризиків стандарт NIST SP 800-30 виділяє декілька ключових компонентів: модель ризику, підходи до оцінювання, методи аналізу ризику та фактори, що впливають на результати оцінювання. Також організації можуть адаптувати методики оцінювання залежно від власних бізнес-процесів, критичності інформаційних ресурсів та організаційної культури.

Модель ризику визначає основні фактори, що враховуються під час оцінювання ризику, а також взаємозв'язки між ними. До них належать загрози, вразливості, наслідки, ймовірність реалізації загроз та передумови, які можуть впливати на стан безпеки організації.

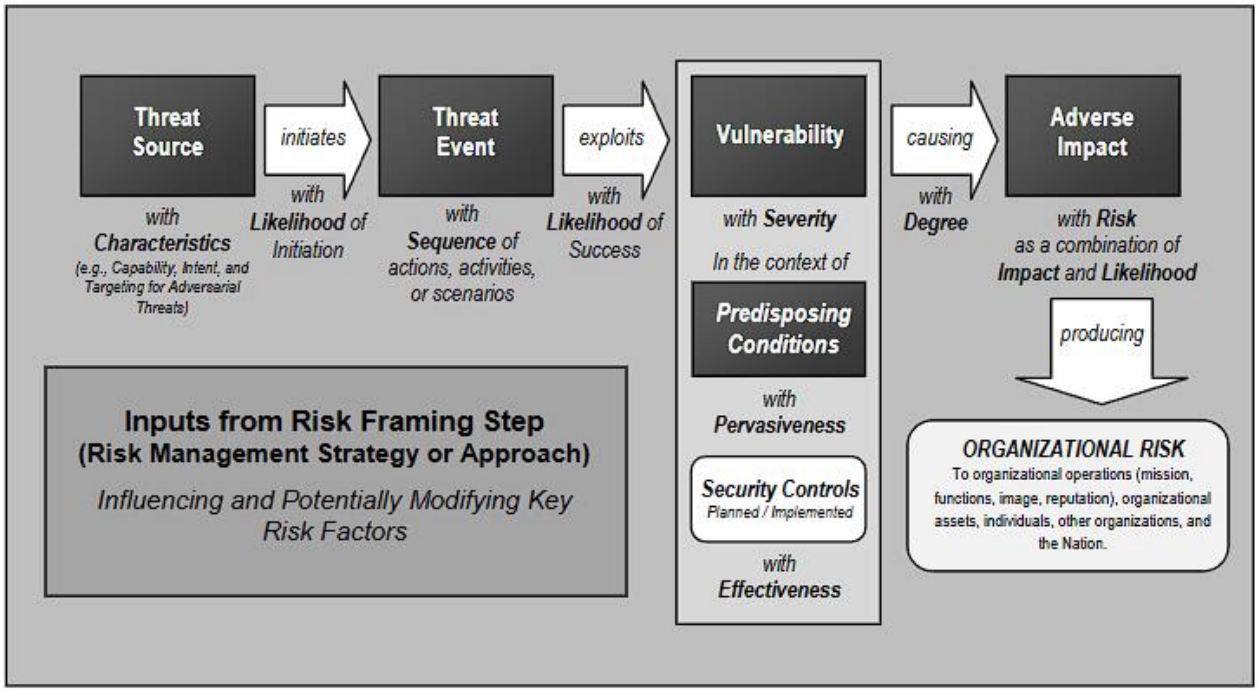


Рис. 2.2.2. Узагальнена модель ризику та взаємозв'язок її компонентів.

NIST SP 800-30 [16] виділяє три основні підходи до оцінювання ризиків: кількісний, якісний та напівкількісний. Їх особливості та класифікація були детально розглянуті у першому розділі роботи.

Крім підходів до оцінювання ризиків, стандарт також визначає три рівні управління ризиками, що дозволяють здійснювати аналіз ризиків на різних рівнях діяльності організації.

Стратегічний рівень організації (Tier 1) використовується для формування загальної стратегії управління ризиками, політик безпеки та процедур захисту інформації. На цьому рівні аналізуються ризики, що можуть впливати на діяльність організації загалом, її ресурси, персонал, репутацію та критично важливі функції. Результати оцінювання ризиків Tier 1 застосовуються для визначення загальноорганізаційних заходів безпеки та передаються на нижчі рівні управління ризиками.

Рівень місій та бізнес-процесів (Tier 2) спрямований на забезпечення стійкості та безперервності виконання ключових функцій організації. Аналіз проводиться щодо окремих бізнес-процесів, сегментів інфраструктури та взаємодії між інформаційними системами. Результати оцінювання ризиків Tier 2 використовуються як для підтримки стратегічних рішень на рівні організації, так і для налаштування захисту конкретних інформаційних систем.

Рівень інформаційних систем (Tier 3) орієнтований на оцінювання ризиків для окремих інформаційних систем, технічних компонентів та середовищ їх функціонування. На цьому рівні здійснюється аналіз конкретних вразливостей, загроз і механізмів захисту, що безпосередньо впливають на безпеку системи. Результати використовуються для вибору, налаштування та вдосконалення засобів захисту, впровадження додаткових контролів безпеки та підтримки процесу авторизації інформаційних систем.

Таким чином, ефективне оцінювання ризиків потребує постійного обміну інформацією між усіма рівнями управління ризиками, оскільки результати аналізу на нижчих рівнях використовуються для формування рішень на вищих рівнях і навпаки.

NIST SP 800-30 також передбачає інтеграцію оцінювання ризиків із Risk Management Framework (RMF), визначеним у NIST SP 800-37 [26]. У межах RMF результати оцінювання ризиків використовуються на всіх етапах життєвого циклу системи.

Основні етапи інтеграції [26]:

- Категоризація

- Вибір засобів захисту
- Впровадження
- Оцінювання
- Авторизація
- Моніторинг

Такий підхід забезпечує безперервний процес управління ризиками та дозволяє оперативно реагувати на зміни у середовищі функціонування інформаційних систем.

У третьому розділі стандарту NIST SP 800-30 розглядається безпосередній процес проведення оцінювання ризиків інформаційної безпеки в організації який складається з чотирьох основних етапів [16,24]:

Етап 1. Підготовка

Організація визначає мету проведення аналізу ризиків, тобто окреслює коло відомостей, які мають бути отримані за підсумками роботи. Також встановлюється сфера охоплення — інформаційні системи, бізнес-процеси, структурні підрозділи або технологічні рішення, що підлягають дослідженню.

Додатково фіксуються припущення та обмеження, що можуть позначитися на результатах: нестача ресурсів, неповнота вихідних даних або особливості побудови інформаційної інфраструктури. Для забезпечення аналізу визначаються внутрішні та зовнішні інформаційні джерела. На цьому ж кроці обирається модель ризику та методологія оцінювання, на основі яких надалі встановлюватимуться рівень ризику, імовірність реалізації загроз і масштаб потенційних негативних наслідків.

Етап 2. Оцінка ризиків

Формується перелік ризиків інформаційної безпеки, які в подальшому ранжуються за ступенем критичності з метою обґрунтованого вибору заходів їх нейтралізації. Аналіз проводиться в межах контексту, сформованого на підготовчому етапі.

Організація виявляє джерела загроз, що можуть становити небезпеку для інформаційних систем, бізнес-процесів або діяльності установи загалом.

Далі ідентифікуються події загроз - дії, процеси або обставини, здатні порушити конфіденційність, цілісність чи доступність інформації. Для кожної такої події визначається ступінь її актуальності та встановлюються можливі джерела виникнення. При цьому одна подія може бути пов'язана з кількома джерелами, а одне джерело — породжувати різні типи загрозових подій.

На наступному кроці виявляються вразливості та чинники, що створюють передумови для реалізації загроз: недоліки програмного забезпечення, хибні конфігурації, людський фактор або прогалини в організаційному захисті.

Після цього оцінюється імовірність настання загроз і аналізуються можливі наслідки для організації з урахуванням впливу на інформаційні активи, бізнес-процеси, фінансовий стан, репутацію та безперервність діяльності. На підставі співвідношення імовірності та масштабу наслідків визначається рівень ризику для кожної події, а отримані результати документуються і використовуються в подальшому управлінні ризиками.

Етап 3. Обмін результатами оцінювання.

Результати оцінювання доводяться до керівництва, власників інформаційних систем, фахівців із безпеки та інших відповідальних осіб. Передача може здійснюватись у формі звітів, аналітичних довідок, реєстрів ризиків або інших документів, передбачених внутрішніми регламентами та вимогами системи управління інформаційною безпекою.

Етап 4. Підтримка актуальності оцінювання.

Метою цього кроку є безперервний моніторинг змін в операційному середовищі організації та своєчасне оновлення результатів оцінювання ризиків інформаційної безпеки.

Результати підлягають регулярному перегляду та коригуванню з урахуванням даних моніторингу, аналізу подій безпеки і змін у діяльності організації. Повторне оцінювання може проводитися як планово через встановлені інтервали, так і позапланово — після суттєвих змін в інфраструктурі, бізнес-процесах або внаслідок інцидентів інформаційної безпеки.

Оновлені результати передаються керівництву та відповідальним особам для прийняття управлінських рішень, що забезпечує безперервність процесу управління ризиками, своєчасне реагування на нові загрози та підтримання належного рівня захисту інформаційних ресурсів.

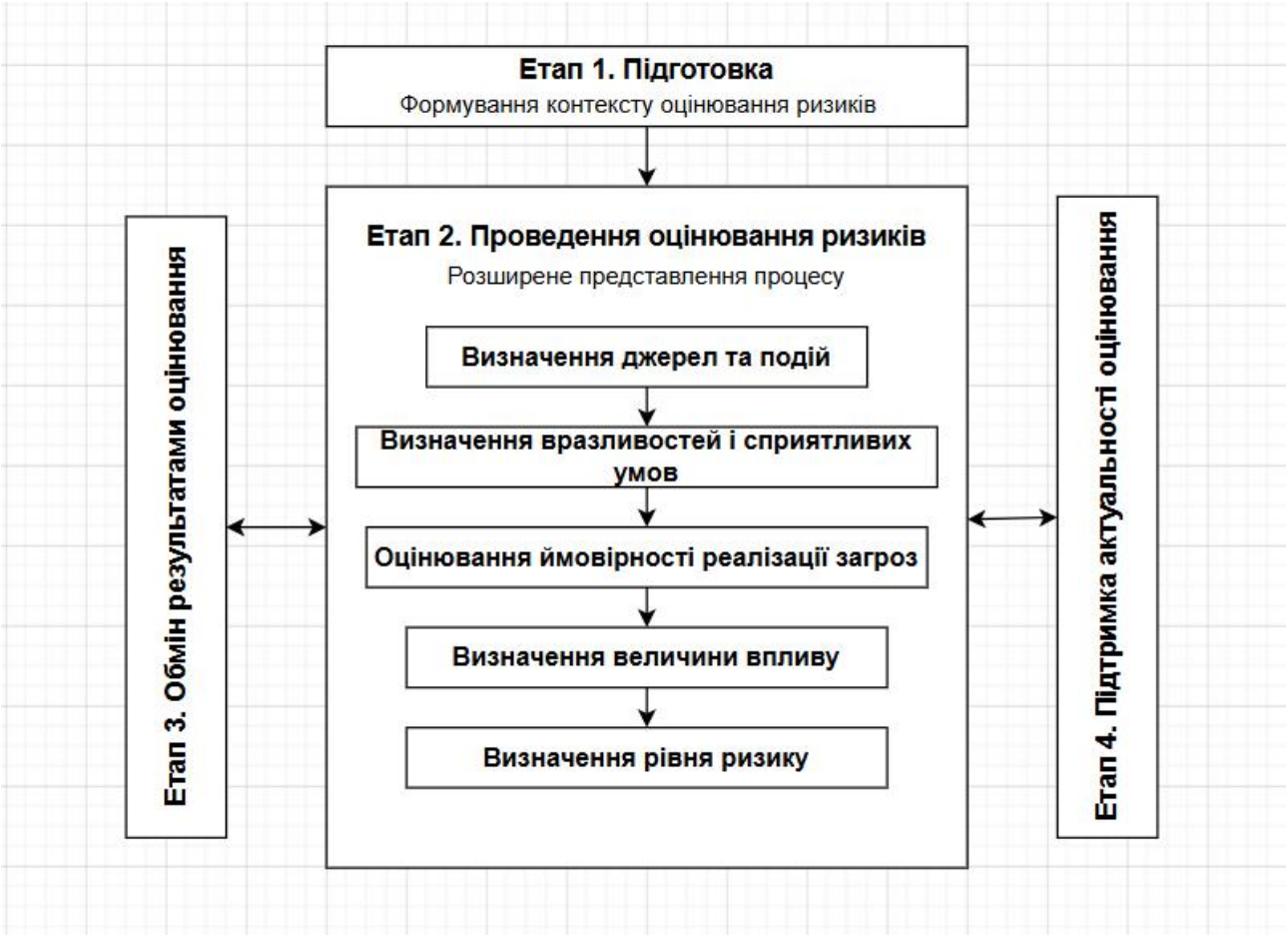


Рис. 2.2.3. Процес оцінювання ризиків

2.2.2 Інтеграція OSINT-даних у цикл оцінювання ризиків

Методологія NIST SP 800-30 забезпечує комплексний підхід до оцінювання ризиків інформаційної безпеки та передбачає використання внутрішніх і зовнішніх джерел інформації для аналізу загроз, вразливостей і потенційних наслідків інцидентів безпеки. Проте у стандарті відсутні згадки про використання OSINT-даних у процесі оцінювання ризиків, хоча відкриті джерела можуть містити значний обсяг відомостей про зовнішню поверхню

атаки організації, її цифрову інфраструктуру, персонал, використовувані технології та актуальні кіберзагрози.

Інтеграція OSINT у методологію NIST SP 800-30 допоможе підвищити актуальність оцінювання ризиків, автоматизувати збір зовнішніх даних та забезпечити більш повне виявлення потенційних загроз і вразливостей. Розглянемо детальніше інтеграцію з відкритими джерелами інформації у ключові етапи оцінювання стандарту:

Етап 1. Підготовка до оцінювання.

На етапі підготовки в стандарті NIST SP 800-30 визначаються цілі оцінювання ризиків, область дослідження, джерела інформації, обмеження та підхід до подальшого аналізу. Саме на тут доцільно інтегрувати OSINT-дані як додаткове джерело зовнішньої інформації про організацію та її інфраструктуру.

Це дозволить отримати більш повне уявлення про цифровий профіль організації, доступні сервіси, публічно оприлюднені технології та потенційні точки компрометації. На відміну від внутрішніх джерел інформації, відкриті дані дозволяють оцінити саме ту інформацію, яка доступна потенційному зловмиснику під час підготовки атаки.

У процесі підготовки можуть використовуватися різні категорії OSINT-даних, розглянуті у підрозділі 2.1. Наприклад на рівні інформаційних систем збір технічних OSINT-даних дозволяє точно визначити реальну зовнішню поверхню атаки. Отримані результати аналізу таких даних можуть використовуватися на стратегічному рівні для прийняття рішень щодо політик безпеки, управління ризиками, пріоритезації ресурсів і вдосконалення механізмів захисту. Таким чином забезпечується взаємозв’язок між технічним аналізом зовнішньої поверхні атаки та стратегічним управлінням інформаційною безпекою організації.

Етап 2. Проведення оцінювання.

На етапі оцінювання ризиків інтеграція OSINT-даних застосовуються безпосередньо для аналізу ризиків інформаційної безпеки. Для систематизації отриманої інформації під час оцінювання ризиків доцільно використовувати

класифікацію наведену у підрозділі 2.1. Такий підхід дозволяє структурувати процес аналізу відповідно до типів інформації та пов’язаних із ними ризиків. Зокрема, технічні дані можуть використовуватися для виявлення мережевих вразливостей та аналізу зовнішньої поверхні атаки, дані про персонал - для оцінювання ризиків соціальної інженерії, а дані про програмне забезпечення та технології - для визначення потенційно вразливих або застарілих компонентів інформаційної інфраструктури.

Відкриті дані також можуть використовуватися для оцінювання ймовірності реалізації загроз. Якщо інформація про організацію активно представлена у відкритому доступі, а її інфраструктура містить доступні з Інтернету сервіси або відомі вразливості, це підвищує ймовірність проведення цільових атак.

Результатом інтеграції OSINT-даних на етапі оцінювання ризиків є формування більш повного переліку загроз, вразливостей та потенційних сценаріїв атак. Це дозволяє підвищити об’єктивність оцінювання ризиків та забезпечити прийняття рішень на основі актуальної інформації про зовнішнє середовище.

Етап 4. Підтримка оцінювання ризиків в актуальному стані.

Використання відкритих джерел інформації дозволяє здійснювати постійне оновлення даних про стан інформаційної безпеки та вчасно реагувати на зміни.

Крім технічного моніторингу, OSINT дозволяє здійснювати контроль інформаційного фону навколо організації. Аналіз згадок у новинах, форумах, соціальних мережах та спеціалізованих спільнотах дає можливість виявляти ознаки витоків даних, репутаційних загроз, шахрайських кампаній або підготовки цільових атак. Подібна інформація може бути використана для своєчасного реагування та коригування заходів захисту. Схему інтеграції OSINT-даних у процес оцінювання ризиків відповідно до методології NIST SP 800-30 наведено на рис. 2.2.3.

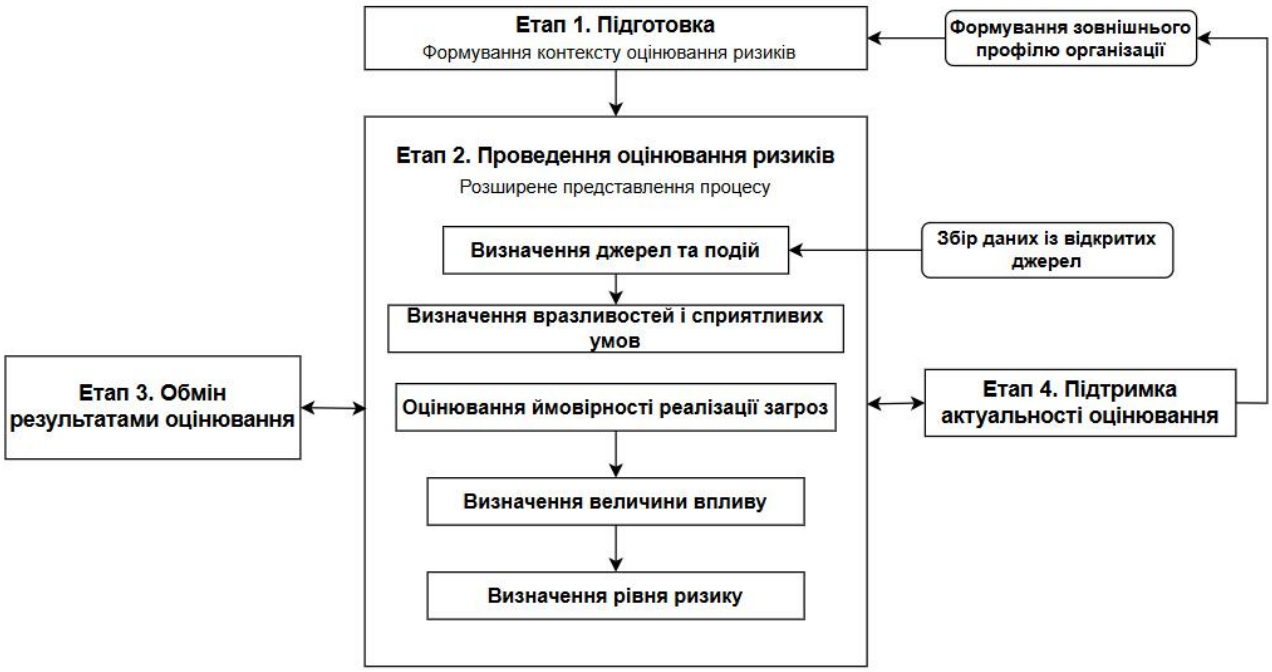


Рис. 2.2.3. Інтеграція OSINT-даних у процес оцінювання ризиків

Таким чином, інтеграція OSINT-даних у методологію NIST SP 800-30 дозволяє розширити процес оцінювання ризиків за рахунок використання актуальної інформації з відкритих джерел. Використання даних з відкритих джерел забезпечує більш повне виявлення загроз, вразливостей і потенційних сценаріїв атак, а також підвищує ефективність моніторингу зовнішньої поверхні атаки організації. Це створює основу для побудови моделі автоматизованого аналізу та кореляції OSINT-даних із потенційними загрозами інформаційної безпеки.

2.3 Побудова системи автоматизованого аналізу та кореляції OSINT-даних із потенційними загрозами

Однією з основних проблем використання OSINT у сфері інформаційної безпеки є необхідність встановлення взаємозв’язків між окремими даними, джерелами інформації, потенційними загрозами та вразливостями. Дані, отримані з відкритих джерел, часто є фрагментованими, мають різний формат та рівень достовірності, що ускладнює їх використання у процесі оцінювання

ризиків. У зв’язку з цим виникає необхідність побудови моделі, яка забезпечуватиме автоматизований аналіз, кореляцію та інтерпретацію OSINT-даних у контексті інформаційної безпеки організації.

2.3.1 Загальна структура системи

Запропонована модель автоматизованого аналізу та кореляції OSINT-даних складається з декількох взаємопов’язаних функціональних модулів, кожен із яких виконує окремі завдання у процесі оцінювання ризиків інформаційної безпеки. Загальна структура моделі наведена на рис. 2.3.1 та алгоритм роботи на рис. 2.3.2.

Основними компонентами моделі є:

1. Модуль збору OSINT-даних.
2. Модуль обробки та нормалізації даних.
3. Модуль аналізу та кореляції.
4. Модуль оцінювання ризиків.
5. Модуль моніторингу та візуалізації результатів.

Запропонована структура моделі дозволяє забезпечити безперервний процес збору та аналізу OSINT-даних, автоматизувати виявлення потенційних загроз і підвищити ефективність оцінювання ризиків інформаційної безпеки організації.

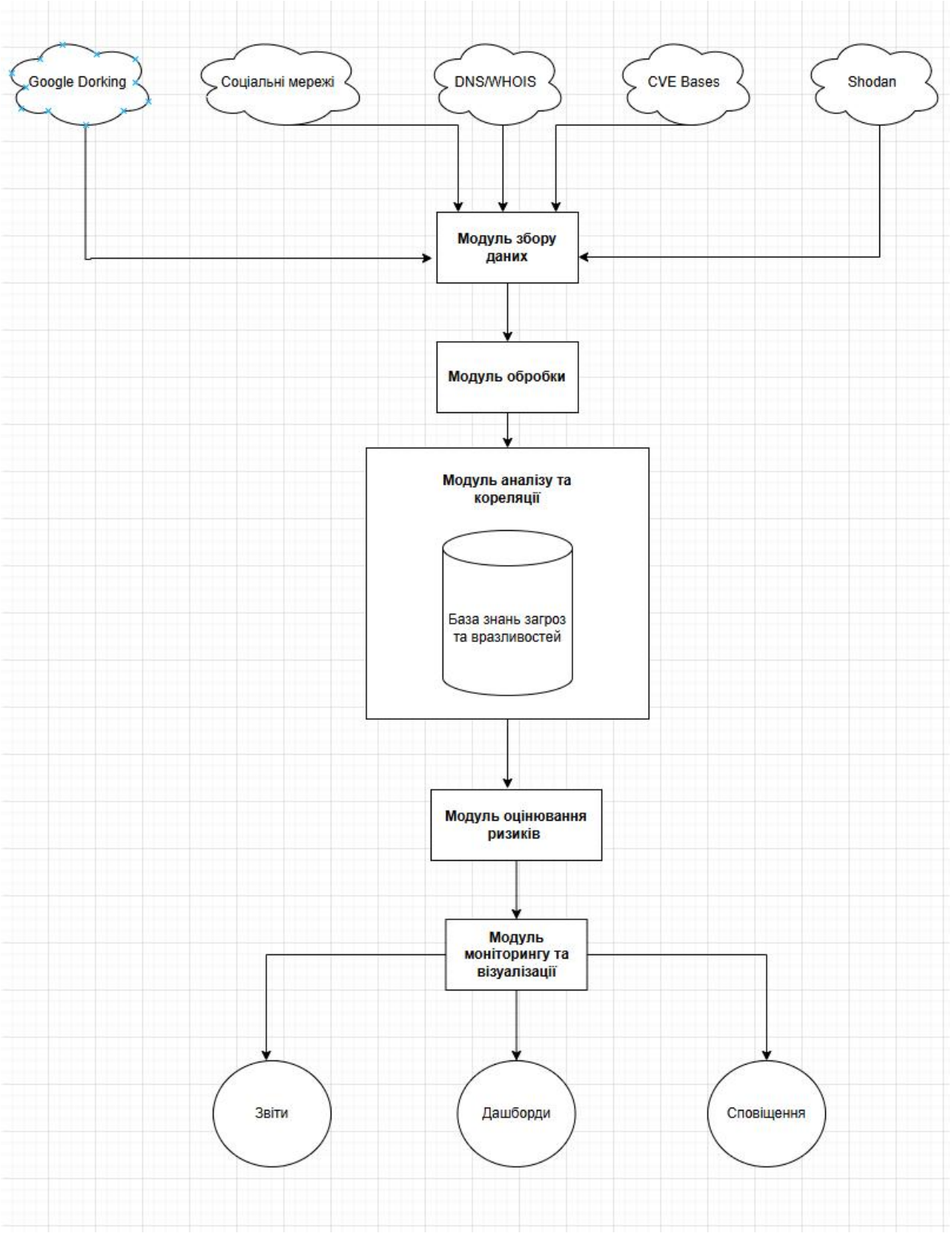


Рис. 2.3.1. Загальна структура системи

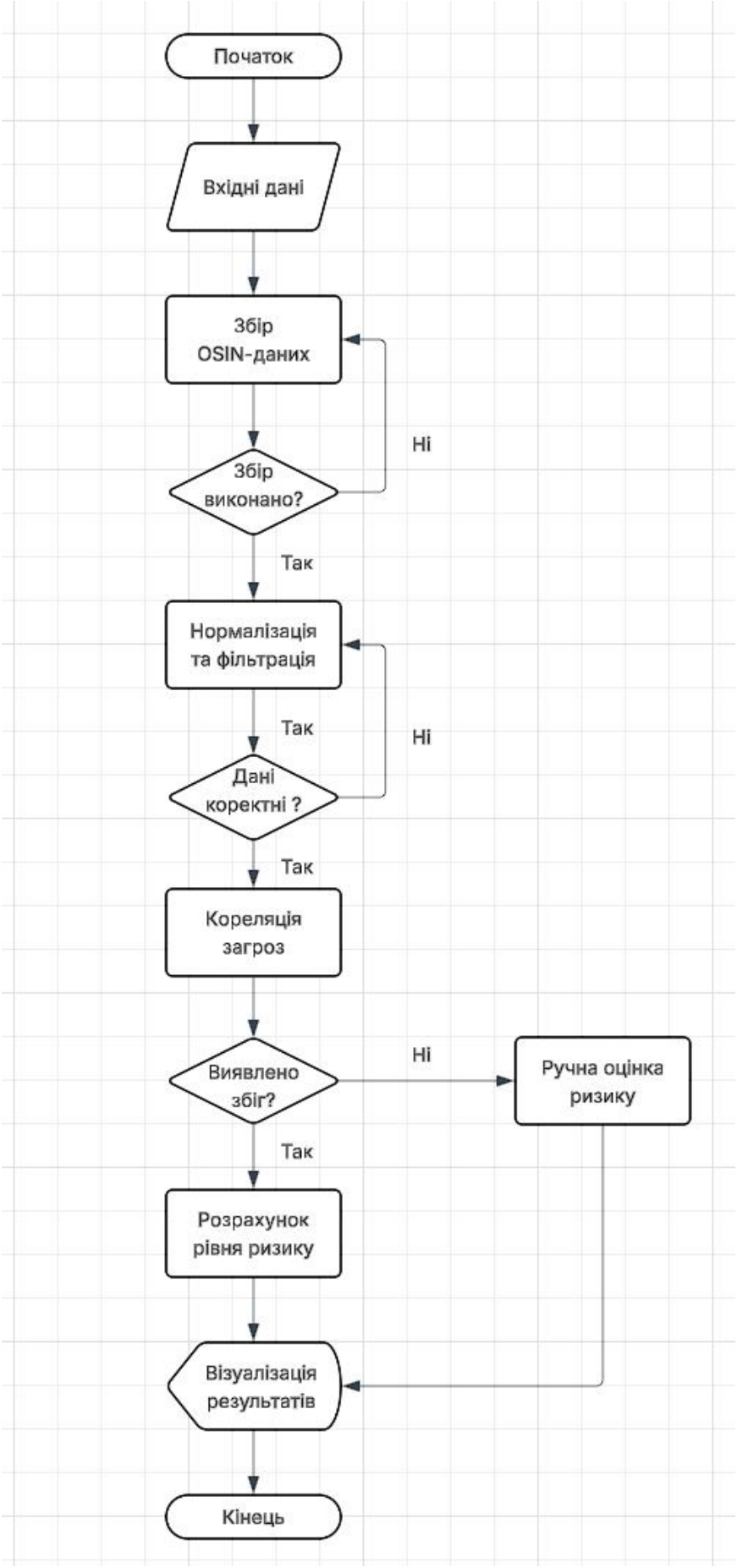


Рис. 2.3.2. Алгоритм роботи

2.3.2 Модуль збору OSINT-даних

Основним призначенням модуля є автоматизований збір, оновлення та початкова обробка відомостей про організацію, її цифрову інфраструктуру, персонал, інформаційні активи та потенційні кіберзагрози.

Для отримання інформації можуть використовуватися різні технології обробки даних залежно від типу джерела та формату матеріалів. Зокрема, взаємодія із зовнішніми сервісами може здійснюватися через API-запити, web scraping, парсинг HTML-сторінок і аналіз метаданих. У разі виявлення документів у форматах PDF, DOCX, XLSX або графічних файлів система здатна застосовувати технології OCR [28], для автоматичного розпізнавання тексту та подальшого аналізу вмісту.

Модуль підтримує інтеграцію з різними типами OSINT-сервісів і платформ відкритої розвідки. Для забезпечення актуальності відомостей він працює у циклічному режимі - через визначені проміжки часу система повторно виконує отримання даних із підключених джерел, оновлює наявну інформацію та фіксує виявлені зміни. Такий підхід дозволяє швидко виявляти нові фактори, що можуть впливати на рівень ризику інформаційної безпеки організаці.

2.3.3 Модуль обробки та нормалізації OSINT-даних

Основною задачею модуля є приведення різної інформації до єдиного структурованого вигляду, придатного для подальшого аналізу, кореляції та оцінювання ризиків інформаційної безпеки.

На етапі обробки система виконує:

1. очищення даних від дублікатів, помилкових або неповних записів;
2. стандартизацію форматів даних;
3. виділення ключових атрибутів;
4. класифікацію даних відповідно до категорій;
5. формування структурованих записів для подальшого аналізу.

Для різних типів інформації можуть використовуватися окремі механізми обробки. Наприклад: HTML-сторінки очищуються від службових елементів та перетворюються у текстовий формат; JSON та XML-відповіді API приводяться до уніфікованої структури; PDF-документи, зображення та скановані файли можуть додатково оброблятися за допомогою OCR-технологій для виділення текстової інформації;

У випадку виявлення некоректних, пошкоджених або неповних даних система виконує їх додаткову перевірку та фільтрацію. Якщо запис не відповідає встановленим критеріям якості або не може бути коректно нормалізований, він позначається як недостовірний та виключається з подальшого автоматизованого аналізу. У разі можливості система може повторно виконувати запит до джерела інформації або здійснювати спробу доповнення даних шляхом кореляції з іншими OSINT-джерелами.

2.3.4 Модуль аналізу та кореляції OSINT-даних

На цьому етапі система обробляє вже нормалізовані дані та виконує їх порівняння з відомими загрозами, вразливостями та індикаторами компрометації за допомогою правил кореляції та бази знань.

Головним компонентом тут виступає *база знань*. Вона містить систематизовану інформацію про відомі методи атак (на основі MITRE ATT&CK [29]), актуальні вразливості (CVE), індикатори компрометації (IoC) та сигнатури потенційних загроз, характерні для різних галузей діяльності.

Під час аналізу система виконує запити до бази знань та порівнює отримані OSINT-дані з наявними записами. Наприклад, якщо було виявлено відкритий порт або конкретну версію вебсервера, система перевіряє, чи існують для нього відомі вразливості або небезпечні конфігурації.

У випадку виявлення збігів система автоматично формує запис про потенційну загрозу та формує попередню оцінку її критичності. Наприклад, якщо знайдено сервіс із критичною CVE-вразливістю, система автоматично

встановлює підвищений рівень ризику та передає інформацію до модуля оцінювання ризиків.

Якщо ж система не знаходить збігів із відомими загрозами або правилами кореляції, тоді оцінювання виконується аналітиком інформаційної безпеки вручну. Спеціаліст самостійно аналізує отримані дані та визначає можливий рівень ризику. За необхідності нова інформація може бути додана до бази знань для подальшого автоматичного використання системою.

Також модуль може аналізувати декілька подій одночасно та встановлювати між ними взаємозв'язки. Наприклад, окремо відкритий порт або витік електронної адреси можуть не створювати критичного ризику, однак поєднання декількох факторів може свідчити про підвищену ймовірність компрометації системи.

2.3.5 Модуль оцінювання ризиків

Тут система оцінює наскільки виявлені загрози та вразливості можуть впливати на інформаційні ресурси та діяльність організації.

Для реалізації оцінювання ризиків у запропонованій моделі доцільно використовувати підхід на основі нечіткої логіки. У сучасних дослідженнях зазначається, що методи нечіткої логіки є ефективним інструментом для моделювання надзвичайних ситуацій, ідентифікації загроз та оцінювання ризиків [27, 30]. Основною перевагою такого підходу є можливість працювати з невизначеними, неповними або неточними даними.

У запропонованій моделі оцінювання ризику виконується на основі трьох основних параметрів [27]:

P_z - ймовірність виникнення загрози;

V_0 - рівень впливу загрози;

E_z - ефективність засобів захисту.

Загальний рівень ризику визначається як функція зазначених параметрів [27]:

$$R=f(P_z,V_O,E_z) \qquad (2.3.1)$$

Для кожного параметра вводяться три рівні оцінювання: «низький», «середній» та «високий». Формалізація нечітких множин здійснюється за допомогою функцій належності. Вони використовуються в нечіткій логіці для визначення ступеня відповідності певного значення до заданої нечіткої множини. На відміну від класичної логіки, де елемент або належить множині, або ні, нечітка логіка дозволяє задавати проміжний ступінь належності в діапазоні від 0 до 1.

Функції належності для P_z визначаються таким чином [27]:

$$\begin{cases} \mu P_z(\text{низька}) = max\ (0,1\text{-}3\times P_z) \\ \mu P_z(\text{середня}) = max\ (0,1\text{-}|2\times P_z\text{-}1|) \\ \mu P_z(\text{висока}) = max\ (0,3\times P_z\text{-}2) \end{cases} \quad (2.3.2)$$

де μP_z - функція належності для параметра P_z.

Функції належності для V_0 визначаються таким чином:

$$\begin{cases} \mu V_O(\text{низька}) = max\ (0,1\text{-}3\times V_O) \\ \mu V_O(\text{середня}) = max\ (0,1\text{-}|2\times V_O\text{-}1|) \\ \mu V_O(\text{висока}) = max\ (0,3\times V_O\text{-}2) \end{cases} \quad (2.3.3)$$

Функції належності для E_z визначаються таким чином:

$$\begin{cases} \mu E_z(\text{низька}) = max\ (0,1\text{-}3\times E_z) \\ \mu E_z(\text{середня}) = max\ (0,1\text{-}|2\times E_z\text{-}1|) \\ \mu E_z(\text{висока}) = max\ (0,3\times E_z\text{-}2) \end{cases} \quad (2.3.4)$$

Після формалізації нечітких множин система переходить до використання нечіткої бази правил [27, 30]. База правил містить набір логічних конструкцій типу «якщо - то», які визначають взаємозв’язок між параметрами загрози та рівнем ризику.

Таблиця 2.3

Нечітка база правил

Правило	P_z	V_0	E_z	R
1	Низька	Низький	Високий	Низький
2	Низька	Високий	Низький	Середній
3	Середня	Середній	Середній	Середній
4	Висока	Високий	Низький	Високий
5	Середня	Високий	Високий	Низький
6	Висока	Низький	Середній	Високий

Наприклад, система може використовувати такі правила:

- якщо ймовірність загрози висока, вплив високий, а ефективність захисту низька - ризик оцінюється як високий;
- якщо ймовірність загрози низька, а засоби захисту мають високу ефективність - ризик визначається як низький;

На основі сформованої нечіткої бази правил система виконує автоматизоване оцінювання рівня ризику для кожного виявленого об’єкта, сервісу або події. Під час аналізу враховуються результати кореляції OSINT-даних, наявність збігів із базою знань загроз та вразливостей, а також контекст функціонування інформаційної системи.

Після завершення оцінювання система виконує пріоритезацію ризиків відповідно до отриманого рівня небезпеки та передає результати до модуля моніторингу й візуалізації.

2.3.6 Модуль моніторингу та візуалізації результатів

Призначений для постійного контролю стану інформаційної безпеки організації, відстеження змін у зовнішній поверхні атаки та зрозумілого представлення результатів оцінювання ризиків.

Для візуалізації можуть використовуватися таблиці, графіки, діаграми, heatmap-карти ризиків, часові шкали подій та інші засоби графічного представлення інформації. Такий підхід значно спрощує аналіз великих обсягів OSINT-даних та дозволяє швидко визначати найбільш критичні загрози.

Модуль моніторингу працює у режимі постійного оновлення даних. Після кожного циклу збору та аналізу OSINT-інформації система автоматично перевіряє наявність нових вразливостей, змін у зовнішній інфраструктурі, появи нових сервісів або витоків даних. У разі виявлення змін система повторно запускає процес оцінювання ризиків, оновлює візуалізацію та реалізовує механізми автоматичного сповіщення відповідальних осіб про критичні події інформаційної безпеки.

Отримані результати можуть використовуватися як на технічному рівні (Tier 3) для моніторингу інформаційних систем, так і на стратегічному рівні (Tier 1) для підтримки прийняття управлінських рішень щодо інформаційної безпеки організації.

Висновки до розділу 2

У другому розділі було розглянуто особливості використання відкритих джерел інформації (OSINT) у процесі оцінювання ризиків інформаційної безпеки та розроблено модель автоматизованого аналізу й кореляції OSINT-даних із потенційними загрозами.

Проведений аналіз показав, що відкриті джерела інформації можуть містити значний обсяг даних про цифрову інфраструктуру організації, персонал, програмне забезпечення, мережеві сервіси, витоки облікових даних та інші дані. У результаті було сформовано класифікацію OSINT-даних та визначено основні джерела їх отримання, а також встановлено взаємозв’язок між типами відкритої інформації та можливими ризиками інформаційної безпеки.

У роботі також було проаналізовано методологію оцінювання ризиків NIST SP 800-30 та визначено можливості інтеграції OSINT-даних у ключові етапи процесу оцінювання ризиків.

Результатом даного розділу стала побудова системи автоматизованого аналізу та кореляції OSINT-даних із потенційними загрозами. У якій реалізовано використання бази знань загроз і вразливостей, механізмів автоматичного зіставлення даних та методів нечіткої логіки для роботи з неповною або невизначеною інформацією.

Розроблений підхід забезпечує автоматизацію аналізу відкритих джерел інформації, підвищує швидкість виявлення потенційних загроз та сприяє більш ефективному прийняттю рішень у процесі управління ризиками інформаційної безпеки. Отримані результати можуть бути використані як основа для подальшої практичної реалізації системи та оцінювання ефективності її застосування в реальних умовах.

РОЗДІЛ 3 ТЕСТУВАННЯ СИСТЕМИ ОЦІНЮВАННЯ РИЗИКІВ НА ОСНОВІ OSINT-ДАНИХ

3.1 Опис середовища дослідження

Для проведення тестування системи оцінювання ризиків на основі OSINT-даних було створено окреме контрольоване середовище, яке імітує інфраструктуру невеликої організації. Основною метою його створення є моделювання реальних умов збору відкритої інформації про організацію та демонстрація можливостей автоматизованого аналізу зовнішньої поверхні атаки.

Спочатку ми розгорнули віртуальний сервер на базі операційної системи Ubuntu Server у програмному забезпеченні VMware Workstation [31, 32]. Використання віртуальної машини дозволяє ізолювати процес тестування від основної операційної системи, безпечно змінювати конфігурацію сервера та швидко відновлювати систему у разі помилок або пошкодження налаштувань.

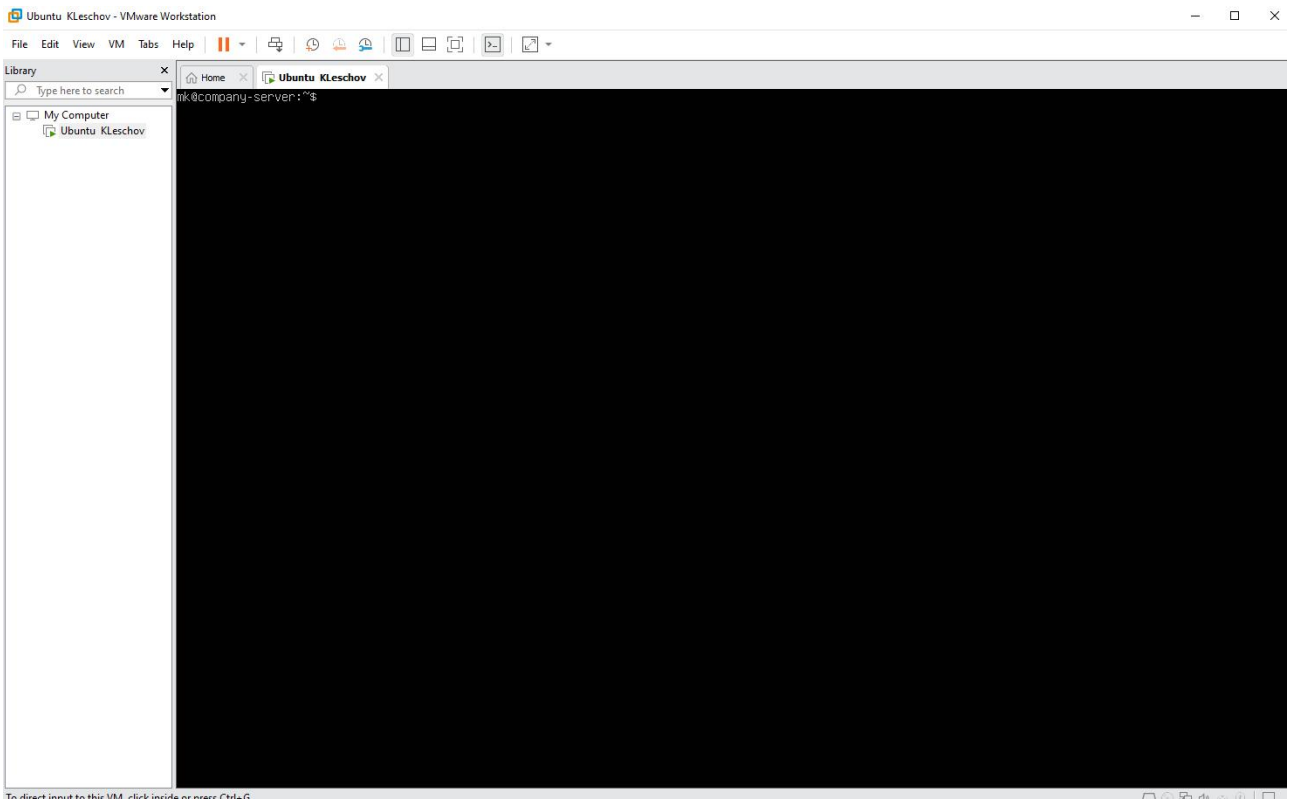


Рис. 3.1.1. Створення віртуального сервера Ubuntu Server

Після розгортання сервера було створено вебресурс умовної організації - «Organization A». Яка працює у сфері 3D-дизайну та цифрової візуалізації. Основним напрямом діяльності компанії є створення тривимірних моделей, архітектурних візуалізацій, концептів інтер'єрів, рекламної графіки та цифрового дизайну для клієнтів малого бізнесу і приватних замовників. У процесі виконання проєктів компанія використовує сучасні засоби 3D-моделювання, рендерингу та графічного проєктування.

Штат організації складається з невеликої кількості працівників, кожен із яких виконує окремі функції у процесі створення та супроводу проєктів. Для моделювання OSINT-витоків було створено умовні профілі співробітників та інформаційні ресурси компанії.

Усі імена, посади, назви проєктів та інші дані, наведені в дослідженні, є вигаданими. Будь-які збіги з реальними особами або організаціями є випадковими. Організаційна структура компанії наведена у таблиці 3.1.1

Таблиця 3.1.1

Організаційна структура "Organization A"

№	ПІБ	Посада	Основні обов'язки
1	Авраменко Олег Сергійович	Директор	Управління компанією, робота з клієнтами, контроль проєктів
2	Коваленко Ірина Андріївна	3D Designer	Створення 3D-моделей та візуалізацій
3	Ткаченко Максим Олександрович	Senior 3D Artist	Розробка складних сцен та рендеринг
4	Мельник Андрій Вікторович	Web Administrator	Підтримка вебсайту та сервера
5	Бондар Софія Романівна	Graphic Designer	Створення графічних матеріалів і UI-елементів
6	Шевченко Владислав Ігорович	Project Manager	Координація виконання проєктів
7	Литвин Катерина Олегівна	Content Manager	Наповнення сайту та ведення соціальних мереж
8	Марченко Денис Павлович	Junior 3D Designer	Допомога у створенні моделей та обробці сцен

Вебсайт організації функціонує на базі Ubuntu Server під керуванням вебсервера Nginx [33]. У конфігурації сервера навмисно залишено окремі потенційно небезпечні елементи, що дозволяють змоделювати зовнішню поверхню атаки та виконати подальший аналіз ризиків.

Після розгортання серверного середовища було виконано підключення до сервера з операційної системи Windows за допомогою SSH-клієнта. Це дозволило здійснювати віддалене адміністрування сервера та редагування файлів вебресурсу.

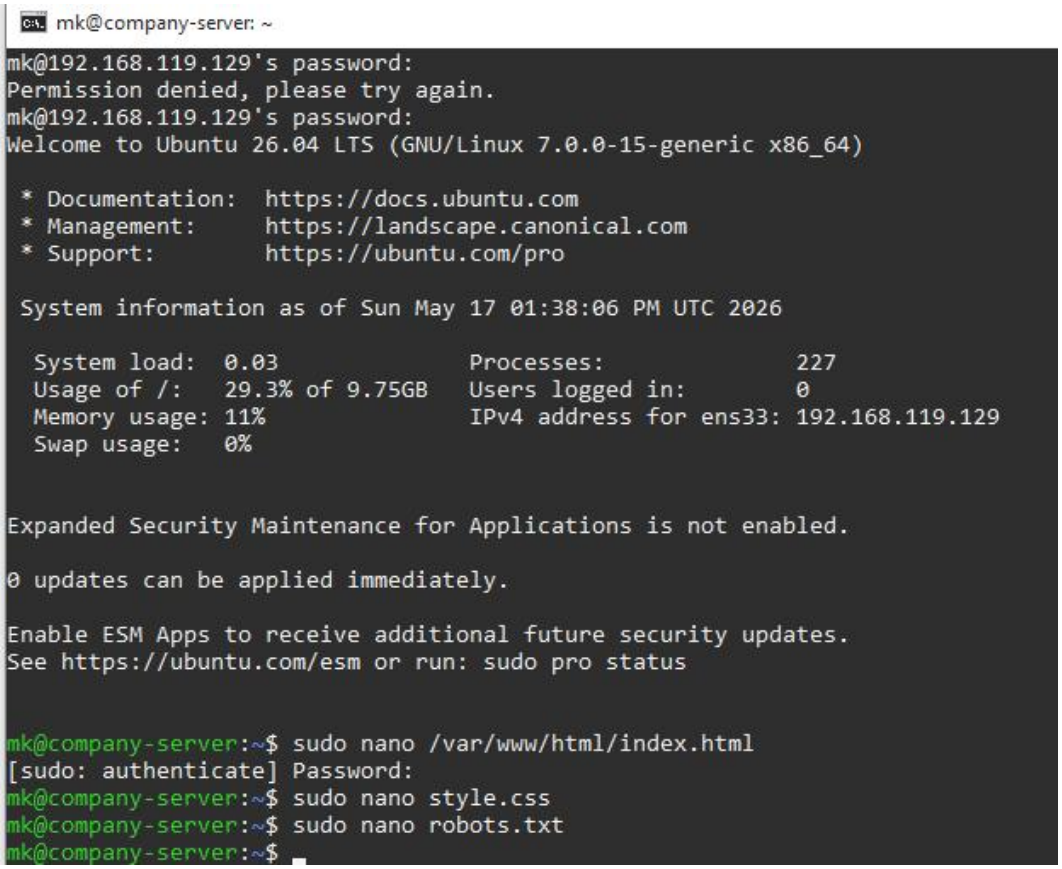


Рис. 3.1.2. Підключення до Ubuntu Server із Windows-хоста

Далі було створено основні файли вебсайту:

- index.html - головна HTML-сторінка сайту;
- style.css - файл стилів вебсайту;
- robots.txt - службовий файл із навмисно доданими прихованими директоріями.

Код зазначених файлів наведено у додатку А.

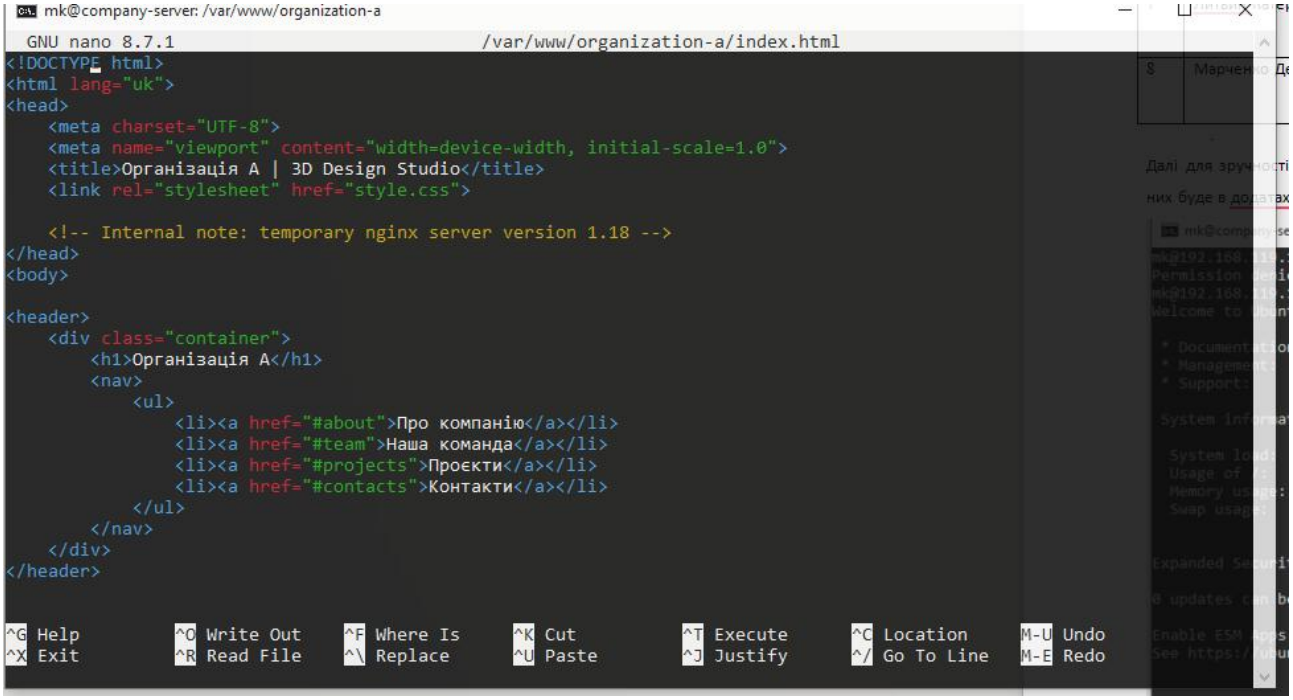


Рис. 3.1.3. Код для головної сторінки сайту

Після створення вебсайту було виконано налаштування Nginx та активацію конфігурації сайту. Основні команди, що використовувались під час налаштування, наведено у таблиці 3.1.2 [33].

Таблиця 3.1.2

Команди налаштування Nginx

Команда	Призначення
sudo rm /etc/nginx/sites-enabled/default	Видалення стандартної конфігурації Nginx
sudo nginx -t	Перевірка правильності конфігураційних файлів
sudo systemctl restart nginx	Перезапуск вебсервера для застосування змін

Далі ми підключилися до вебсайту через браузер Windows-хоста для перевірки працездатності створеного ресурсу.

На вебсайті було розміщено інформацію про компанію, перелік співробітників, напрями діяльності та контактні дані. Частина інформації була

спеціально додана для подальшої демонстрації можливостей OSINT-аналізу та виявлення потенційних витоків інформації.

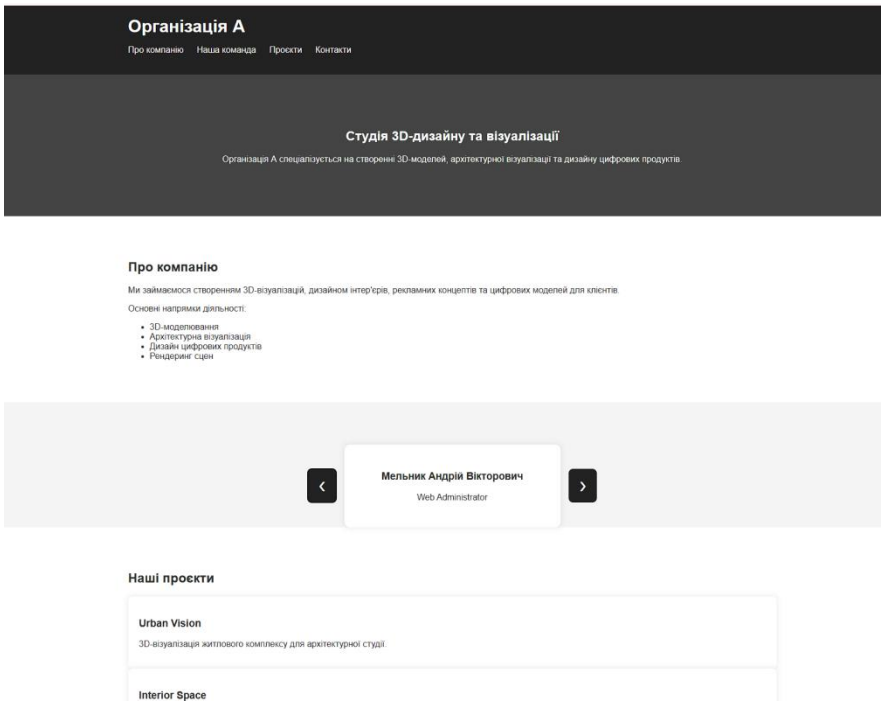


Рис. 3.1.4. Головна сторінка сайту

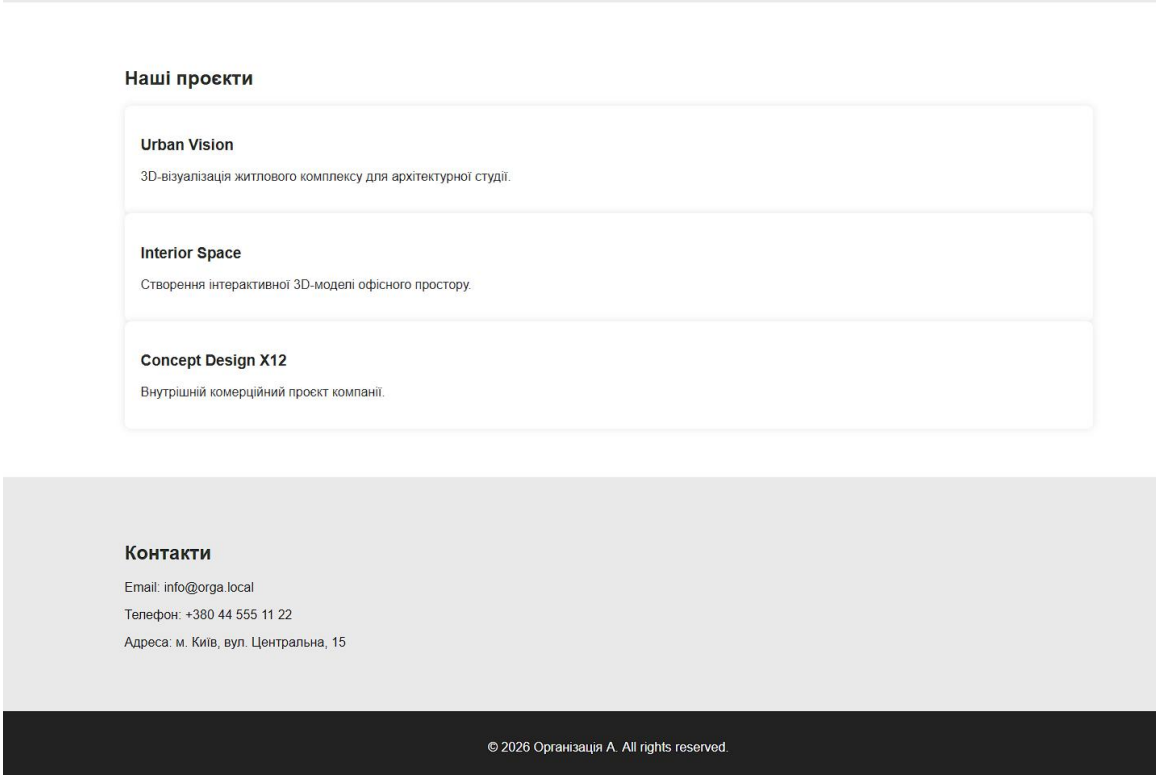


Рис.3.1.5. Інформація на сайті

З метою моделювання зовнішньої поверхні атаки та демонстрації можливостей OSINT-аналізу в тестовому середовищі були навмисно створені окремі потенційно небезпечні конфігурації та інформаційні витоки.

Зокрема, було активовано directory listing, що дозволяє переглядати вміст директорій вебсервера через браузер у випадку відсутності index-файлів.

Для цього виконувалося редагування конфігурації Nginx за допомогою команди:

sudo nano /etc/nginx/sites-available/organization-a

Також на сервері було встановлено FTP-сервіс vsftpd [40], який використовується як приклад потенційно небезпечного сервісу віддаленого доступу.

Для встановлення та запуску використовувалися команди які наведені в таблиці 3.1.3.

Таблиця 3.1.3

Команди встановлення FTP-сервісу

Команда	Призначення
sudo apt install vsftpd -y	Встановлення FTP
sudo systemctl enable vsftpd	Додавання FTP-сервісу до автозавантаження
sudo systemctl start vsftpd	Запуск FTP
sudo systemctl status vsftpd	Перевірка поточного стану

Після встановлення FTP-сервісу відповідний порт став доступним у мережі, що в подальшому дозволить системі виявляти його під час OSINT-аналізу та сканування зовнішньої поверхні атаки.

Щоб змоделювати витік інформації через соціальні мережі було створено умовний профіль системного адміністратора у соціальній мережі X (Twitter). У профілі були опубліковані повідомлення технічного характеру, які містять конфіденційну інформацію щодо внутрішньої інфраструктури організації.

Приклади опублікованих повідомлень:

“Finally finished configuring the new ngnix-server for Organization A. The website is now running on Ubuntu Server 22.04. Enable remote SSH access for maintenance tasks. Need to review firewall rules later.”

“Temporary FTP backup service is still active while we migrate old project archives.”

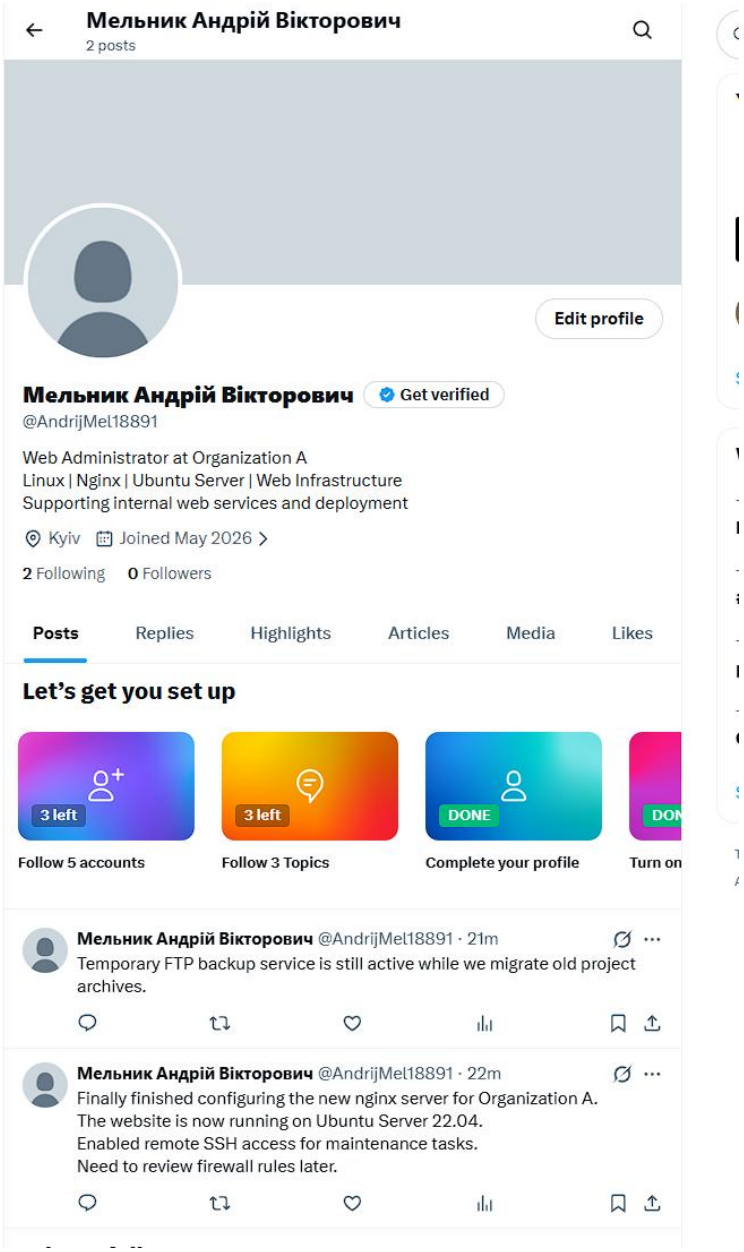


Рис. 3.1.6. Профіль веб адміністратора у соціальній мережі X

Подібна інформація може використовуватися зловмисниками під час проведення OSINT-розвідки, соціальної інженерії або підготовки до подальших атак на інформаційну інфраструктуру організації.

3.2 Застосування системи оцінювання ризиків на основі OSINT-даних

Для практичної перевірки запропонованої моделі було реалізовано програмну систему автоматизованого оцінювання ризиків інформаційної безпеки на основі OSINT-даних. Реалізація виконувалась мовою програмування Python у середовищі розробки Visual Studio Code [34]. Тестування проводилось у контрольованому середовищі, описаному у підрозділі 3.1. Під час розробки програмної системи використовувалось віртуальне середовище Python .venv, яке забезпечує ізольоване встановлення бібліотек, спрощує керування програмними залежностями проєкту та дозволило уникнути конфліктів між різними версіями програмних компонентів.

Основною метою практичної реалізації стало підтвердження можливості автоматизованого збору інформації з відкритих джерел, її аналізу, кореляції із відомими загрозами та подальшого оцінювання рівня ризику для інформаційної системи організації.

Спочатку ми підготували середовище розробки, встановивши необхідні програмні компоненти. Для реалізації окремих функціональних модулів системи використовувався набір спеціалізованих бібліотек Python. Зокрема, для сканування мережевих портів застосовувалась бібліотека python-nmap [36], яка забезпечує взаємодію з утилітою Nmap та дозволяє автоматизувати процес аналізу мережевих сервісів.

Для збору та аналізу вебконтенту використовувались бібліотеки requests та BeautifulSoup4 [35], що забезпечують отримання HTML-вмісту вебсторінок і подальший парсинг структури сайту.

Реалізаія механізму оцінювання ризиків на основі нечіткої логіки була здійснена за допомогою бібліотеки scikit-fuzzy [37]. Вона дозволила реалізувати функції належності, нечіткі множини та систему правил, описаних у попередньому розділі роботи.

Для формування графіків та візуалізації результатів аналізу - matplotlib і plotly [38]. Збір даних із соціальної мережі X реалізовувався за рахунок ntscraper [39], а для створення вебінтерфейсу та генерації HTML-звіту - Flask.

Рис. 3.2.1. Встановлення необхідних бібліотек

Програмна система була реалізована із використанням модульної структури, що дозволило логічно розділити окремі етапи обробки даних та забезпечити гнучкість подальшого розвитку програмного забезпечення. Кожен модуль виконує окрему функцію у процесі збору, аналізу та оцінювання ризиків, що спрощує супровід системи, оновлення окремих компонентів та тестування програмного коду.

Центральним елементом системи є файл *main.py,* який забезпечує запуск усіх функціональних компонентів та координує послідовність виконання основних етапів аналізу. Збір OSINT-даних реалізовано у компоненті *collector.py*, який здійснює автоматизоване отримання інформації з вебресурсів, мережевих сервісів та соціальних мереж. Обробка та стандартизація отриманих даних виконується у файлі normalizer.py, який приводить інформацію до єдиного структурованого вигляду для подальшого аналізу.

Аналіз зібраної інформації та пошук взаємозв’язків між виявленими подіями реалізовано у модулі analyzer.py. Даний модуль виконує порівняння отриманих результатів із базою знань загроз та визначає потенційно небезпечні комбінації подій і конфігурацій. Оцінювання рівня ризику виконується програмним компонентом risk_engine.py, у якому реалізовано механізм нечіткої

логіки для визначення ступеня критичності виявлених загроз. Формування підсумкового HTML-звіту та візуалізація результатів аналізу здійснюється модулем report.py.

Для зберігання бази знань системи було створено окремий файл knowledge_base.json, у якому містяться відомості про потенційно небезпечні мережеві сервіси, відкриті порти, характерні ознаки вразливостей, ключові слова та індикатори компрометації, що використовуються під час кореляції даних.

Повний програмний код реалізованих модулів наведено у додатку Б.

Після завершення реалізації програмної системи було виконано її практичний запуск та тестування роботи у створеному середовищі. На цьому етапі модуль *collector.py* автоматично виконав збір OSINT-даних щодо тестового сервера організації.Під час мережевого сканування системою було виявлено три відкритих мережевих порти:

- порт 21 (FTP);
- порт 22 (SSH);
- порт 80 (HTTP).

Окрім мережевого сканування, система автоматично виконала аналіз вебсайту організації. Було отримано HTML-вміст головної сторінки сайту, проаналізовано файл robots.txt та виявлено приховані директорії, які були навмисно створені у попередньому підрозділі.

Додатково система встановила, що для директорії /backup увімкнено directory listing, що дозволяє переглядати вміст каталогу без проходження автентифікації. Подібна конфігурація може призвести до витоку службових або резервних файлів організації.

Також модуль збору OSINT-даних виконав аналіз публікацій із профілю вебадміністратора організації у соціальній мережі X. У процесі аналізу було виявлено публікації, що містили згадки про активний FTP-сервіс та віддалений SSH-доступ, що підтверджується знайденими відкритими портами.

Наведена інформація може використовуватися потенційним зловмисником під час проведення соціальної інженерії або підготовки цілеспрямованих атак на інфраструктуру організації.

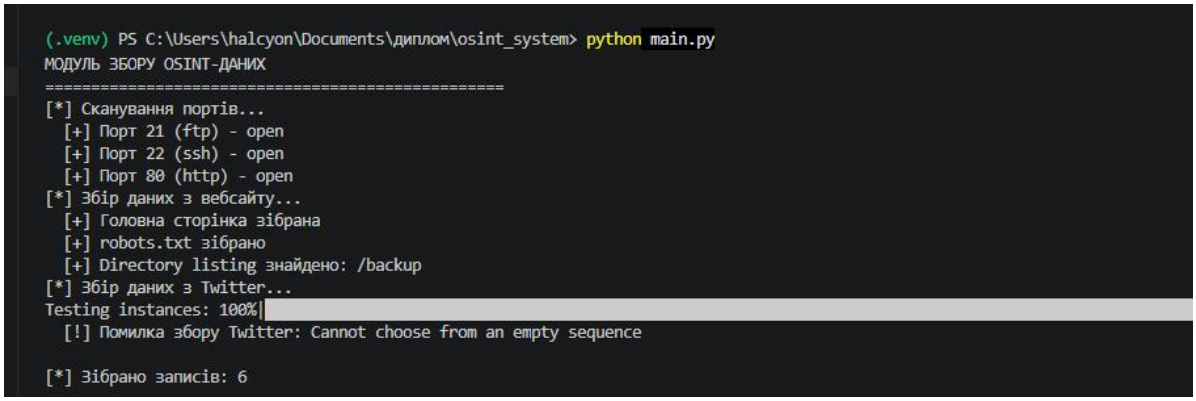


Рис. 3.2.2. Результат роботи модулю збору даних

Після завершення збору інформації компонент *normalizer.py* виконує обробку та нормалізацію отриманих даних. На цьому етапі система видаляла дублікати записів, стандартизувала формати IP-адрес, URL-адрес та мережевих сервісів, а також формувала структуровані записи для подальшого аналізу.

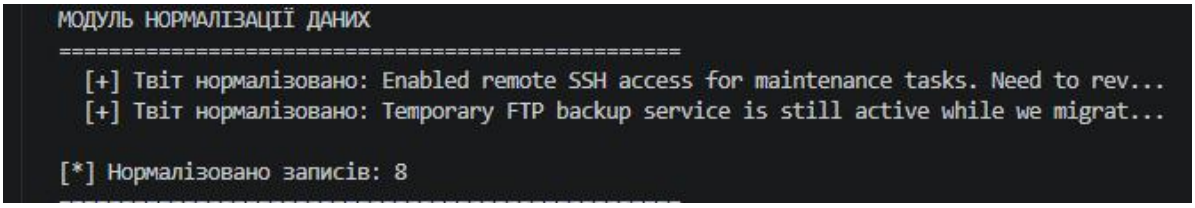


Рис. 3.2.3. Результат роботи модулю нормалізації даних

Далі модуль *analyzer.py* здійснив кореляцію отриманих даних із базою знань загроз та вразливостей. Зокрема, система автоматично встановила зв’язок між відкритим FTP-портом та згадками про FTP у соціальних мережах; відкритим SSH-доступом та публікаціями про віддалене адміністрування та наявністю прихованих директорій та активним directory listing.

На основі знайдених збігів система сформувала перелік потенційних загроз та передала їх до модуля оцінювання ризиків.

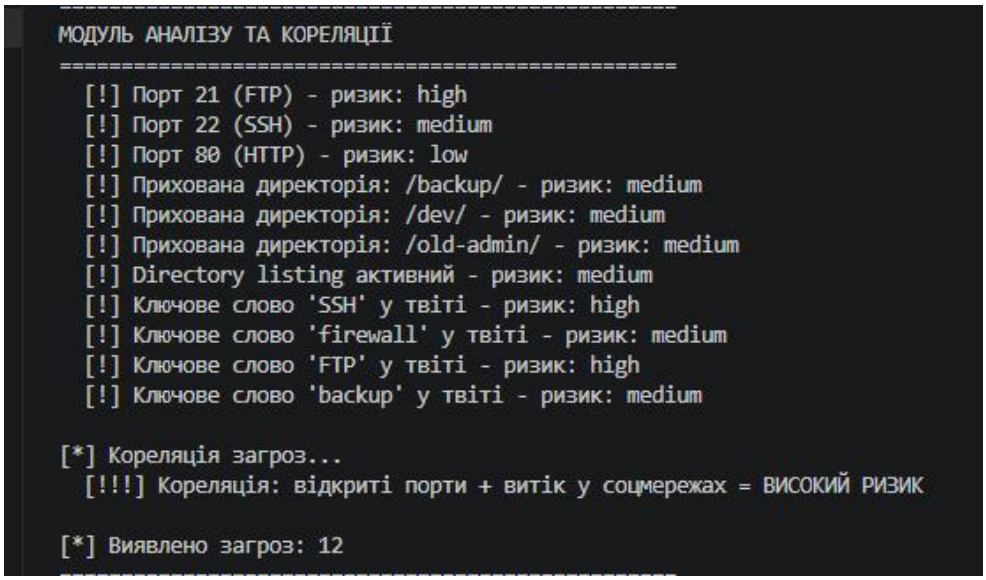


Рис. 3.2.4. Результат роботи модулю аналізу та кореляції даних

На наступному етапі програмний компонент *risk_engine.py* виконав оцінювання ризиків із використанням механізмів нечіткої логіки. Для кожної виявленої загрози система автоматично визначила значення трьох основних параметрів:

- Pz - ймовірність реалізації загрози;
- V0 - рівень потенційного впливу;
- Ez - ефективність наявних засобів захисту.

Після цього на основі функцій належності та нечіткої бази правил було обчислено інтегральний рівень ризику для кожної події. Значення інтегрального показника ризику визначалось у діапазоні від 0 до 1, де значення від 0 до 0.3 відповідали низькому рівню ризику (LOW), значення від 0.31 до 0.7 - середньому рівню ризику (MEDIUM), а значення понад 0.7 - високому рівню ризику (HIGH).

Найвищий рівень ризику отримала комбінована загроза, яка поєднувала відкриті мережеві сервіси, витік технічної інформації у соціальних мережах та наявність небезпечних конфігурацій вебсервера. Для цієї загрози система визначила рівень ризику HIGH із числовим показником score = 0.859. Інші

виявлені події отримали рівень MEDIUM, оскільки окремо вони не створювали критичної загрози, однак могли бути використані у складі комплексної атаки.

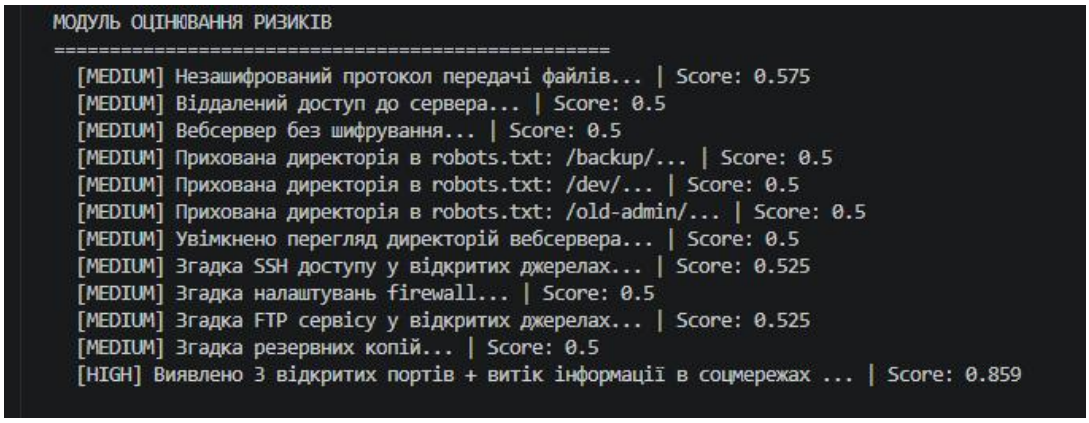


Рис. 3.2.5. Результат роботи модулю оцінки ризиків

На завершальному етапі *report.py* автоматично сформував загальну саттистику та HTML-звіт із результатами аналізу та оцінювання ризиків. У звіті відображалась загальна статистика виявлених загроз, розподіл ризиків за рівнями критичності, а також детальна таблиця з описом кожної події.

Загалом у процесі тестування система виявила 12 потенційних загроз, серед яких:

- 1 загроза отримала рівень HIGH;
- 11 загроз отримали рівень MEDIUM.

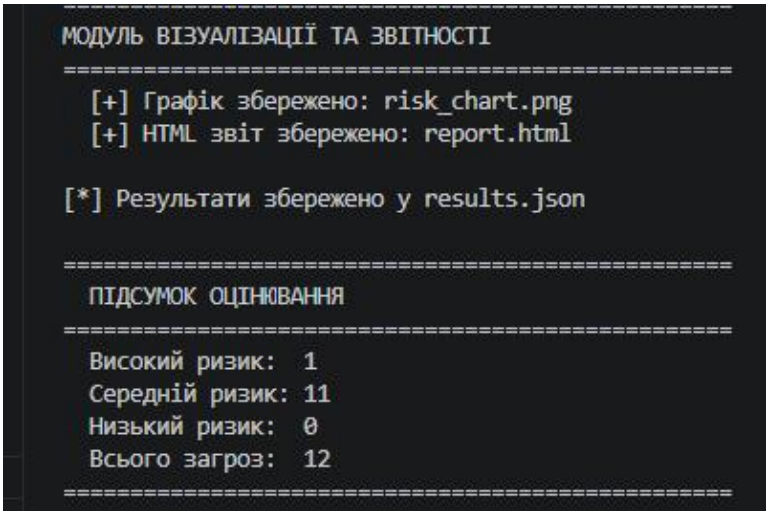
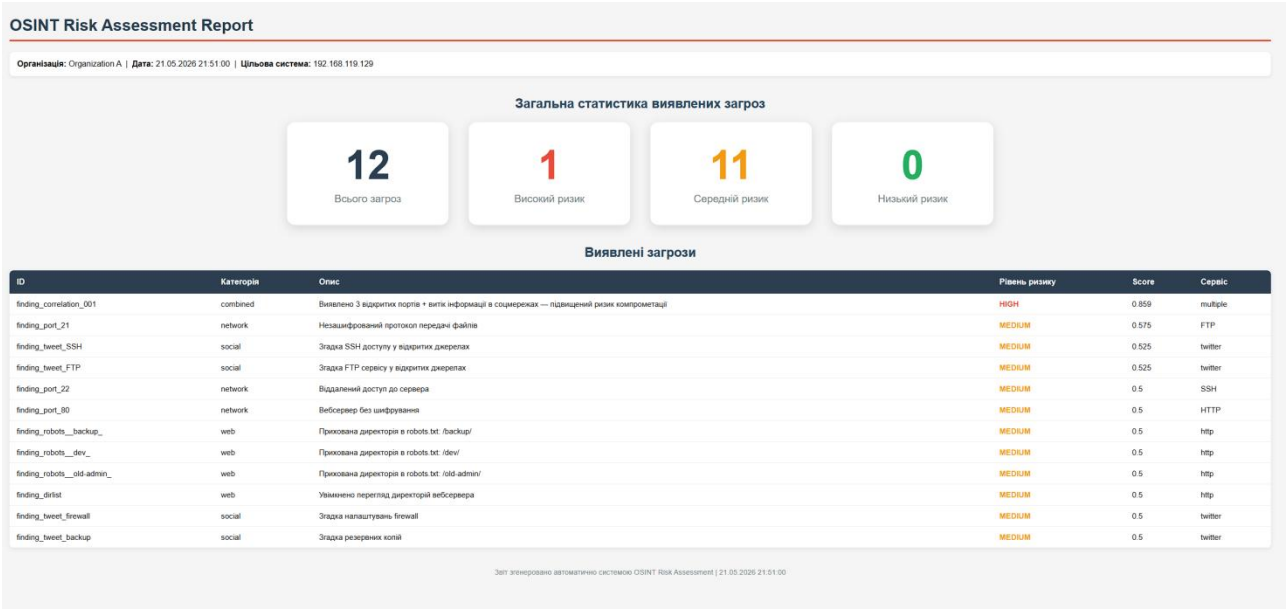


Рис. 3.2.6. Результат роботи модулю звітності

Приклад сформованого звіту наведено на рис. 3.2.7.



OSINT Risk Assessment Report

Організація: Organization A | **Дата:** 21.05.2026 21:51:00 | **Цільова система:** 192.168.119.129

Загальна статистика виявлених загроз

12	1	11	0
Всього загроз	Високий ризик	Середній ризик	Низький ризик

Виявлені загрози

ID	Категорія	Опис	Рівень ризику	Score	Сервіс
finding_correlation_001	combined	Виявлено 3 відкритих портів + витік інформації в соцмережах — підвищений ризик компрометації	HIGH	0.859	multiple
finding_port_21	network	Незашифрований протокол передачі файлів	MEDIUM	0.575	FTP
finding_tweet_SSH	social	Згадка SSH доступу у відкритих джерелах	MEDIUM	0.525	twitter
finding_tweet_FTP	social	Згадка FTP сервісу у відкритих джерелах	MEDIUM	0.525	twitter
finding_port_22	network	Віддалений доступ до сервера	MEDIUM	0.5	SSH
finding_port_80	network	Вебсервер без шифрування	MEDIUM	0.5	HTTP
finding_robots__backup_	web	Прихована директорія в robots.txt: /backup/	MEDIUM	0.5	http
finding_robots__dev_	web	Прихована директорія в robots.txt: /dev/	MEDIUM	0.5	http
finding_robots__old-admin_	web	Прихована директорія в robots.txt: /old-admin/	MEDIUM	0.5	http
finding_dirlist	web	Увімкнено перегляд директорій вебсервера	MEDIUM	0.5	http
finding_tweet_firewall	social	Згадка налаштувань firewall	MEDIUM	0.5	twitter
finding_tweet_backup	social	Згадка резервних копій	MEDIUM	0.5	twitter

Звіт згенеровано автоматично системою OSINT Risk Assessment | 21.05.2026 21:51:00

Рис. 3.2.7. Візуальне представлення виявлених загроз у звіті

Згенерований HTML-звіт дозволяє у зручному вигляді аналізувати результати роботи системи та може використовуватись як для технічного моніторингу, так і для підтримки прийняття рішень у сфері інформаційної безпеки.

3.3 Аналіз результатів та порівняння з традиційними методами

Після завершення тестування розробленої системи було проведено аналіз отриманих результатів та виконано порівняння запропонованого підходу з традиційними методами оцінювання ризиків інформаційної безпеки, розглянутими у розділі 1. Розшифрування скорочень критеріїв та відповідні шкали їх оцінювання наведено у таблицях 1.3.1–1.3.5 підрозділу 1.3.

Таблиця 3.3.1

Порівняльна таблиця

Метод / модель / засіб	ТО	ВД	АП	АТ	СВ
ISO/IEC 27005	3	2	2	3	2
NIST 800-30	3	2	2	3	2
OCTAVE	1	1	1	2	2
COBIT 5 for Risk	3	2	2	3	3
CRAMM	3	2	3	3	3
MEHARI	3	2	2	3	2
RiskWatch	2	2	3	2	3
Розроблена система	3	3	3	2	2

Аналіз наведеної таблиці дозволяє зробити низку важливих висновків щодо характеристик розробленої системи у порівнянні з традиційними методами.

Насамперед, щодо типу оцінювання ризиків, розроблена система реалізує змішаний підхід, що поєднує якісний та кількісний аналіз. Це досягається завдяки використанню методів нечіткої логіки у модулі оцінювання ризиків, де формалізовані функції належності дозволяють отримувати числові показники ризику навіть за умов неповноти вхідних даних. Такий підхід відповідає рівню найбільш зрілих методологій – ISO/IEC 27005, NIST 800-30, COBIT 5 for Risk та MEHARI.

Ключовою відмінністю розробленої системи є використання відкритих джерел інформації як основного джерела даних для оцінювання ризиків. На відміну від традиційних методів, які переважно базуються на внутрішній документації, результатах аудитів або експертних оцінках, запропонований підхід використовує OSINT-дані, що відображають інформацію про організацію, доступну у зовнішньому інформаційному просторі. Це дозволяє виявляти потенційні загрози та вразливості, які можуть залишатися поза увагою внутрішніх механізмів контролю, проте можуть бути використані потенційним зловмисником під час підготовки атаки.

Переваги використання відкритих джерел інформації для оцінювання ризиків інформаційної безпеки та можливості переходу від реактивного до

проактивного управління ризиками були попередньо обґрунтовані у тезах доповіді [41].

Результати проведеного тестування підтвердили практичну доцільність такого підходу та його ефективність для аналізу зовнішньої поверхні атаки організації.Щодо можливості автоматизації, розроблена система має можливість повної автоматизації всіх процесів, від збору даних через відкриті джерела до формування підсумкового HTML-звіту, все реалізується без ручного втручання. Аналогічний рівень автоматизації серед традиційних підходів характерний лише для CRAMM та RiskWatch, однак вони не підтримують OSINT-інтеграцію та потребують значних ресурсів для впровадження.

Адаптивність до різних типів організацій розробленої системи визначена як середня. Це пов'язано з тим, що поточна реалізація системи орієнтована на організації з публічно доступними вебресурсами та мережевою інфраструктурою. Для малих і середніх організацій система є цілком придатною без суттєвої адаптації — її модульна архітектура та відкритий програмний код дозволяють швидко розгорнути систему та налаштувати під конкретні потреби з мінімальними витратами ресурсів, що підтверджено у ході тестування. Натомість для великих організацій із складними гетерогенними середовищами повноцінне впровадження потребує значного розширення бази знань, доопрацювання модулів збору даних та залучення кваліфікованих фахівців, що суттєво збільшує витрати часу та ресурсів.

Складність впровадження та використання (СВ) оцінена як середня. Система реалізована мовою Python, використовує стандартні бібліотеки та не потребує ліцензованого програмного забезпечення. Розгортання відбувається у віртуальному середовищі, що спрощує процес встановлення. Порівняно з CRAMM, RiskWatch та COBIT 5 for Risk, які характеризуються високою складністю впровадження, розроблена система є менш ресурсомісткою.

Принципово новими характеристиками, що відсутні у традиційних методах, є автоматична кореляція даних та оновлення в реальному часі.

Механізм кореляції, реалізований у модулі analyzer.py, дозволяє автоматично встановлювати взаємозв'язки між окремими OSINT-подіями, що в сукупності можуть свідчити про підвищений рівень ризику. Саме завдяки цьому механізму системі вдалося виявити загрозу рівня HIGH, її особливістю стало те, що вона була результатом не одного окремого фактору, а кореляції кількох незалежних джерел інформації: відкритих мережевих портів, технічних публікацій у соціальних мережах та небезпечних конфігурацій вебсервера. Жоден із традиційних методів, розглянутих у першому розділі, не передбачає автоматизованої кореляції даних такого типу в режимі реального часу.

Переваги розробленої системи порівняно з традиційними підходами:

- використання виключно відкритих джерел даних – система не потребує доступу до внутрішньої документації організації та дозволяє здійснювати оцінювання ризиків з позиції зовнішнього спостерігача або потенційного зловмисника;
- повна автоматизація процесу – від збору OSINT-даних до формування звіту з оцінками ризиків, що скорочує час проведення аналізу та мінімізує вплив людського фактору;
- автоматична кореляція різнорідних подій – система здатна виявляти складені загрози, що виникають унаслідок взаємодії кількох незалежних факторів;
- безперервний моніторинг у режимі реального часу – модуль моніторингу забезпечує своєчасне виявлення змін у зовнішній поверхні атаки та нових потенційних загроз;
- відсутність ліцензійних обмежень та низька вартість впровадження – система реалізована на базі відкритих бібліотек Python і не потребує ліцензованого програмного забезпечення;
- застосування методів нечіткої логіки – дозволяє отримувати числові оцінки ризику навіть за умов неповних або неточних вхідних даних.

Обмеження розробленої системи:

- система орієнтована виключно на аналіз зовнішньої поверхні атаки та не охоплює внутрішні ризики організації, пов'язані з бізнес-процесами, фізичною безпекою або інсайдерськими загрозами;
- якість результатів безпосередньо залежить від повноти та актуальності бази знань загроз і вразливостей – обмежена або застаріла база знань може призводити до неповного виявлення загроз;
- середній рівень адаптивності до великих організацій із складними гетерогенними середовищами потребує додаткового налаштування та розширення функціональних можливостей системи;
- певні категорії OSINT-даних, зокрема інформація з соціальних мереж, можуть бути недоступні у разі введення платформами обмежень на автоматизований доступ або зміни умов використання API.

Отже, проведений порівняльний аналіз підтверджує, що розроблена система оцінювання ризиків на основі OSINT-даних не є прямою заміною традиційних методологій, а виступає їхнім функціональним доповненням. Традиційні підходи, такі як ISO/IEC 27005 або NIST 800-30, залишаються актуальними для комплексного управління ризиками всередині організації, тоді як запропонована система забезпечує принципово новий рівень аналізу – автоматизоване виявлення загроз на основі відкритих даних із зовнішнього середовища.

Результати тестування підтверджують практичну ефективність системи: за один цикл аналізу було автоматично виявлено 12 потенційних загроз, серед яких одну критичну, що стала наслідком кореляції кількох незалежних OSINT-подій. Подібний результат неможливо отримати засобами традиційних методів без значних ручних зусиль та витрат часу. Водночас застосування нечіткої логіки для оцінювання ризиків забезпечило числову інтерпретацію загроз, що відповідає вимогам кількісного підходу й дозволяє пріоритизувати заходи реагування.

Таким чином, практичне впровадження розробленої системи може суттєво підвищити оперативність та повноту процесу оцінювання ризиків

інформаційної безпеки, особливо в частині аналізу зовнішньої поверхні атаки та виявлення загроз, що базуються на відкрито доступних даних про організацію.

Висновки до розділу 3

У третьому розділі було практично реалізовано та протестовано систему автоматизованого оцінювання ризиків інформаційної безпеки на основі OSINT-даних, а також проведено порівняльний аналіз отриманих результатів із традиційними методами.

Спочатку було створено контрольоване тестове середовище, що імітує інфраструктуру реальної малої організації. Середовище включало вебсервер на базі Ubuntu Server під керуванням Nginx, навмисно налаштовані потенційно небезпечні конфігурації - активний directory listing, відкриті мережеві порти FTP та SSH, а також змодельовані витоки інформації через соціальну мережу X.

У ході дослідження було підтверджено працездатність усіх функціональних компонентів. Реалізована програмна система забезпечила автоматизований збір інформації з відкритих джерел, включаючи мережеві сервіси, вебресурси та соціальні мережі, після чого виконала їх подальший аналіз і оцінювання рівня ризику із використанням механізмів нечіткої логіки.

Результати тестування продемонстрували здатність системи виявляти не лише окремі потенційно небезпечні події, а й складені загрози, що формуються внаслідок взаємозв’язку кількох незалежних факторів.

Проведене порівняння з традиційними методами оцінювання ризиків показало, що розроблена система характеризується високим рівнем автоматизації, можливістю роботи в режимі реального часу та використанням відкритих джерел даних для аналізу зовнішньої поверхні атаки організації. При цьому система не потребує ліцензованого програмного забезпечення та може бути адаптована для використання у малих і середніх організаціях із мінімальними витратами ресурсів.

ВИСНОВКИ

У цій роботі було досліджено можливості використання відкритих джерел інформації (OSINT) у процесі оцінювання ризиків інформаційної безпеки та розроблено систему автоматизованого аналізу й кореляції OSINT-даних із потенційними загрозами.

У першому розділі проведено аналіз сучасних методів та засобів оцінювання ризиків інформаційної безпеки. Розглянуто основні міжнародні стандарти та методології - ISO/IEC 27005, NIST SP 800-30, OCTAVE, COBIT 5 for Risk, CRAMM, MEHARI та RiskWatch. Проведений порівняльний аналіз показав, що більшість існуючих підходів базується переважно на внутрішніх або гібридних джерелах інформації та не передбачає систематичного використання відкритих зовнішніх джерел розвідки.

У другому розділі розроблено систему автоматизованого аналізу та кореляції OSINT-даних із потенційними загрозами інформаційної безпеки. Сформовано класифікацію OSINT-даних за типами інформації та визначено основні джерела їх отримання. Проаналізовано методологію NIST SP 800-30 та обґрунтовано можливості інтеграції відкритих джерел на ключових етапах процесу оцінювання ризиків. Запропонована система складається з п'яти взаємопов'язаних модулів: збору даних, їх нормалізації, аналізу та кореляції, оцінювання ризиків і моніторингу та візуалізації результатів. Для оцінювання ризиків застосовано метод нечіткої логіки, який забезпечує можливість отримання числових оцінок навіть за умов неповноти або неточності вхідних даних.

У третьому розділі виконано практичну реалізацію та тестування розробленої системи. Було створено контрольоване лабораторне середовище, що імітує інфраструктуру реальної організації із навмисно введеними вразливостями: активним directory listing, відкритими портами FTP та SSH, а також змодельованими витоками інформації через соціальну мережу X. Програмна система реалізована мовою Python у середовищі Visual Studio Code з

використанням модульної архітектури. У результаті тестування система автоматично виявила 12 потенційних загроз, серед яких одна отримала високий рівень, що стало наслідком кореляції кількох незалежних OSINT-подій — відкритих мережевих портів, технічних публікацій у соціальних мережах та небезпечних конфігурацій вебсервера.

Порівняльний аналіз із традиційними методами підтвердив, що розроблена система забезпечує кращу автоматизацію та унікальну можливість роботи з відкритими джерелами даних, яка відсутня у всіх розглянутих традиційних підходах. При цьому система не потребує ліцензованого програмного забезпечення та може бути розгорнута у малих і середніх організаціях із мінімальними витратами ресурсів.

Таким чином, результати дипломної роботи підтверджують доцільність та ефективність інтеграції OSINT-технологій у процес оцінювання ризиків інформаційної безпеки. Розроблена система може слугувати функціональним доповненням до традиційних методологій управління ризиками, забезпечуючи принципово новий рівень автоматизованого аналізу зовнішньої поверхні атаки організації на основі відкритих джерел інформації.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Ризики інформаційної безпеки. Чому це важливо і як з ними працювати відповідно до ISO/IEC 27005?. *IT Specialist.* URL: https://my-itspecialist.com/iso-iec-27005-risk-management (дата звернення: 10.05.2026).

2. ДСТУ ISO/IEC 27005:2019 Інформаційні технології. Методи захисту. Управління ризиками інформаційної безпеки.

3. Garkusha V. METHODICAL APPROACH TO THE RISK ASSESSMENT OF AN ENTERPRISE INFORMATION SECURITY. *Pryazovskyi Economic Herald.* 2020. № 2(19). URL: https://doi.org/10.32840/2522-4263/2020-2-15 (дата звернення: 10.05.2026).

4. Що таке процес управління ризиками інформаційної безпеки згідно з ISO 27005?. *Lazarus Alliance,* Inc. URL: https://lazarusalliance.com/uk/Що-таке-процес-управління-ризиками-інформаційної-безпеки-за-стандартом-ISO-27005/ (дата звернення: 10.04.2026).

5. Роголь Г. Управління ризиками відповідно до стандарту ISO 31000:2018. *Онлайн-консультант фахівця з якості.* URL: https://qualityexpert.com.ua/articles/657421-upravlinnya-ryzykamy-vidpovidno-do-standartu-iso-310002018 (дата звернення: 10.05.2026).

6. Shurda K. Methods of qualitative and quantitative risk analysis. *Balanced nature using*. 2020. № 4. С. 64–72. URL: https://doi.org/10.33730/2310-4678.4.2020.226622 (дата звернення: 12.04.2026).

7. Основні методи аналізу ризиків. *BukLib.net.* URL: https://buklib.net/books/26596/ (дата звернення: 12.04.2026).

8. Методи якісної оцінки ризиків. *Stud.* URL: https://stud.com.ua/47509/investuvannya/metodi_yakisnoyi_otsinki_rizikiv (дата звернення: 12.04.2026).

9. Analysis of methods for assessing and managing cyber risks and information security / O. Potii та ін. *Radiotekhnika.* 2021. № 206. С. 5–24. URL: https://doi.org/10.30837/rt.2021.3.206.01 (дата звернення: 12.05.2026).

10. Методики і програмні продукти для оцінки ризиків. *Stud.* URL: https://stud.com.ua/179799/informatika/metodiki_programni_produkti_otsinki_riziki v (дата звернення: 12.05.2026).

11. Методика OCTAVE. *Stud.* URL: https://stud.com.ua/179801/informatika/metodika_octave (дата звернення: 12.05.2026).

12. Методи управління ризиками інформаційної безпеки CRAMM та COBIT 5 for Risk / П. Сидоркін та ін. *Сучасні інформаційні технології у сфері безпеки та оборони.* 2023. Т. 47, № 2. С. 41–47. URL: https://doi.org/10.33099/2311-7249/2023-47-2-41-47 (дата звернення: 15.05.2026).

13. Леншин А., Хоменко А. В. Результати системного аналізу методів оцінки ризиків безпеці інформації MAGERIT та MEHARI. *Прикладна радіоелектроніка*. 2009. Т. 8, № 3.

14. Методика RiskWatch. *Stud.* URL: https://stud.com.ua/179802/informatika/metodika_riskwatch (дата звернення: 15.05.2026).

15. Best Risk and Compliance Assessment Software Solutions. *RiskWatch*. URL: https://www.riskwatch.com/ (date of access: 15.05.2026).

16. NIST Special Publication 800-30. Guide for Conducting Risk Assessments. URL: https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-30r1.pdf.

17. Industry News 2017 COBIT 5 for RiskA Powerful Tool for Risk Management. *ISACA.* URL: https://www.isaca.org/resources/news-and-trends/industry-news/2017/cobit-5-for-riska-powerful-tool-for-risk-management (date of access: 15.05.2026).

18. КОНЦЕПТУАЛЬНІ ПІДХОДИ ІНТЕГРАЦІЇ ЕТИЧНИХ НОРМ У ПОЛІТИКУ ІНФОРМАЦІЙНОЇ БЕЗПЕКИ / С. Легомінова et al. *Сучасний захист інформації*. 2025. Vol. 63, no. 3. P. 90–98. URL: https://doi.org/10.31673/2409-7292.2025.031152 (date of access: 18.05.2026).

19. Дрижакова Д. Ю., Волинець Р. А. ВИКОРИСТАННЯ ВІДКРИТИХ ДЖЕРЕЛ ІНФОРМАЦІЇ (OSINT) У СФЕРІ БЕЗПЕКИ ДЕРЖАВИ:

ТЕХНОЛОГІЇ ТА ПЕРСПЕКТИВИ. *Цифрове наукове суспільство: соціально-економічні, правові та міжнародні аспекти.* 2025. URL: https://www.researchgate.net/publication/390316593_VIKORISTANNA_VIDKRITIH_DZEREL_INFORMACII_OSINT_U_SFERI_BEZPEKI_DERZAVI_TEHNOLOGII_TA_PERSPEKTIVI.

20. OSINT як інструмент для атаки на бізнес. *IAM*. URL: https://www.intelligence.org.ua/instrument-dlia-ataky-na-biznes/ (дата звернення: 18.05.2026).

21. Maxym Z. Розвідка з відкритих джерел. *Maxym Zosym.* URL: https://www.maxzosim.com/rozvidka-z-vidkritikh-dzherel-osint/ (дата звернення: 18.05.2026).

22. OSINT: технологія збору та аналізу даних з відкритих джерел. Softlist. URL: https://softlist.com.ua/ua/news/osint-tekhnologiya-sbora-i-analiza-dannyh-iz-otkrytyh-istochnikov (дата звернення: 18.05.2026).

23. OSINT Framework. *OSINT Framework.* URL: https://osintframework.com/ (date of access: 18.05.2026).

24. How to Implement NIST SP 800-30, Complete Guide for 2025. *Isora GRC.* URL: https://www.saltycloud.com/blog/how-to-implement-nist-800-30/ (date of access: 18.05.2026).

25. NIST Special Publication 800-39. Managing Information Security Risk Organization, Mission, and Information System View. Official edition. URL: https://nvlpubs.nist.gov/nistpubs/legacy/sp/nistspecialpublication800-39.pdf.

26. NIST Special Publication 800-37. Risk Management Framework for Information Systems and Organizations A System Life Cycle Approach for Security and Privacy. URL: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-37r2.pdf.

27. Murasov R., Nikitin A., Meshcheriakov I. Математична модель оцінювання ризиків функціонування об’єктів критичної інфраструктури на основі теорії нечіткої логіки. *Journal of Scientific Papers "Social Development and*

Security". 2024. Т. 14, № 5. С. 166–174. URL: https://doi.org/10.33445/sds.2024.14.5.17 (дата звернення: 20.05.2026).

28. Технологія OCR: як автоматизація розпізнавання тексту може революціонізувати ваш бізнес. *SalesBox.* URL: https://salesbox.ua/blog/tekhnologiya-ocr-yak-avtomatizatsiya-rozpiznavannya-tekstu-mozhe-revolyutsionizuvati-vash-biznes/ (дата звернення: 20.05.2026).

29. MITRE ATT&CK. *MITRE ATT&CK.* URL: https://attack.mitre.org/ (дата звернення: 20.05.2026).

30. Методи нечіткої логіки в задачах кібербезпеки. *Житомирська політехніка.* URL: https://learn.ztu.edu.ua/mod/resource/view.php?id=175750 (дата звернення: 20.05.2026).

31. Setting up an Ubuntu Linux VM in VMware. *GitHub*. URL: https://w4118.github.io/guides/vm-setup.html (date of access: 22.05.2026).

32. Sreenath. How to Install Ubuntu Linux in VMWare. *It's FOSS*. URL: https://itsfoss.com/install-ubuntu-in-vmware/ (date of access: 22.05.2026).

33. Glass E. How to Install and Configure Nginx on Ubuntu. *DigitalOcean*. URL: https://www.digitalocean.com/community/tutorials/how-to-install-nginx-on-ubuntu-20-04 (date of access: 22.05.2026).

34. Microsoft. Visual Studio Code - The open source AI code editor. *Visual Studio Code*. URL: https://code.visualstudio.com/ (date of access: 24.05.2026).

35. Requests / Beautiful Soup / Python. *Stack Overflow*. URL: https://stackoverflow.com/questions/71991873/requests-beautiful-soup-python (date of access: 22.05.2026).

36. Parakkaden A. T. Automating Network Scanning with Python and Nmap. Medium. URL: https://medium.com/@amaltomparakkaden/automating-network-scanning-with-python-and-nmap-948948f0b161/

37. GitHub - scikit-fuzzy/scikit-fuzzy: Fuzzy Logic SciKit (Toolkit for SciPy). *GitHub.* URL: https://github.com/scikit-fuzzy/scikit-fuzzy (date of access: 22.05.2026).

38. Compatibility between matplotlib and plotly: Is there a solution?. *Plotly Community Forum.* URL: https://community.plotly.com/t/compatibility-between-matplotlib-and-plotly-is-there-a-solution/82624/2 (date of access: 24.05.2026).

39. OrdinaryDry3358. Twitter Tweets web scraping help!. Reddit. URL: https://www.reddit.com/r/learnpython/comments/1m387i5/twitter_tweets_web_scraping_help/ (date of access: 24.05.2026).

40. Як встановити і налаштувати FTP-сервер vsftpd на Linux: Ubuntu, Centos, Debian. *Вікіпедія серверів и хостингу.* URL: https://vps.ua/wiki/ukr/install-vsftpd/ (дата звернення: 24.05.2026).

41. Лозова І. Л., Клєщов М. О. Система оцінювання ризиків інформаційної безпеки організації на основі OSINT-даних. *Стратегії кіберстійкості: управління ризиками та безперервність бізнесу*: матеріали всеукр. наук.-практ. конф., м. Київ, 25 лютого 2026 р. С. 33-36. Україна

ДОДАТКИ

Додаток А

Програмний код вебсайту «Organization A»

index.html

```
<!DOCTYPE html>
<html lang="uk">
<head>
    <meta charset="UTF-8">
    <meta  name="viewport"  content="width=device-width,  initial-
scale=1.0">
    <title>Організація A | 3D Design Studio</title>
    <link rel="stylesheet" href="style.css">
    <!-- Internal note: temporary nginx server version 1.18 -->
</head>
<body>
<header>
    <div class="container">
        <h1>Організація A</h1>
        <nav>
            <ul>
                <li><a href="#about">Про компанію</a></li>
                <li><a href="#team">Наша команда</a></li>
                <li><a href="#projects">Проєкти</a></li>
                <li><a href="#contacts">Контакти</a></li>
            </ul>
        </nav>
    </div>
</header>
<section class="hero">
    <div class="container">
        <h2>Студія 3D-дизайну та візуалізації</h2>
        <p>
            Організація A спеціалізується на створенні 3D-моделей,
            архітектурної   візуалізації   та   дизайну   цифрових
продуктів.
        </p>
    </div>
</section>
<section id="about" class="content-section">
    <div class="container">
        <h2>Про компанію</h2>
        <p>
            Ми  займаємося  створенням  3D-візуалізацій,  дизайном
інтер'єрів,
            рекламних концептів та цифрових моделей для клієнтів.
        </p>
        <p>
            Основні напрямки діяльності:
        </p>
```

```
        <ul>
            <li>3D-моделювання</li>
            <li>Архітектурна візуалізація</li>
            <li>Дизайн цифрових продуктів</li>
            <li>Рендеринг сцен</li>
        </ul>
    </div>
</section>
<section id="team" class="content-section gray">
    <div class="container">
        <h2>Наша команда</h2>
        <div class="team-grid">
            <div class="member-card">
                <h3>Авраменко Олег Сергійович</h3>
                <p>Директор</p>
                <p>Email: director@orga.local</p>
            </div>
            <div class="member-card">
                <h3>Коваленко Ірина Андріївна</h3>
                <p>3D Designer</p>
                <p>Email: iryna.k@orga.local</p>
            </div>
            <div class="member-card">
                <h3>Ткаченко Максим Олександрович</h3>
                <p>Senior 3D Artist</p>
                <p>Email: max.t@orga.local</p>
            </div>
            <div class="member-card">
                <h3>Мельник Андрій Вікторович</h3>
                <p>Web Administrator</p>
                <p>Email: admin@orga.local</p>
            </div>
        </div>
    </div>
</section>
<section id="projects" class="content-section">
    <div class="container">
</html>
```

style.css

```
body {
    margin: 0;
    font-family: Arial, sans-serif;
    background-color: #f4f4f4;
    color: #333;
}
.container {
    width: 90%;
    max-width: 1200px;
    margin: auto;
}
header {
    background-color: #222;
```

```
    color: white;
    padding: 20px 0;
}
header h1 {
    margin: 0;
}
nav ul {
    list-style: none;
    padding: 0;
}
nav ul li {
    display: inline-block;
    margin-right: 20px;
}
nav ul li a {
    color: white;
    text-decoration: none;
}
.hero {
    background: #444;
    color: white;
    padding: 80px 0;
    text-align: center;
}
.content-section {
    padding: 60px 0;
    background-color: white;
}
.gray {
    background-color: #e9e9e9;
}
.team-grid {
    display: grid;
    grid-template-columns: repeat(auto-fit, minmax(250px, 1fr));
    gap: 20px;
}
.member-card,
.project-card {
    background: white;
    padding: 20px;
    border-radius: 8px;
    box-shadow: 0 0 10px rgba(0,0,0,0.1);
}
footer {
    background-color: #222;
    color: white;
    text-align: center;
    padding: 20px 0;
                                    }
```

robots.txt

```
User-agent: *
Disallow: /backup/
Disallow: /dev/
Disallow: /old-admin/
```

Конфігурація Nginx

```
server {
    listen 80;
    server_name _;
    root /var/www/organization-a;
    index index.html;
    location / {
        try_files $uri $uri/ =404;
    }
    location /backup/ {
        autoindex on;
    }
}
```

Додаток Б

Програмний код модулів системи оцінювання ризиків

knowledge_base.json

```
{
  "ports": {
    "21": {"service": "FTP", "risk": "high", "description":
"Незашифрований протокол передачі файлів"},
    "22": {"service": "SSH", "risk": "medium", "description":
"Віддалений доступ до сервера"},
    "80": {"service": "HTTP", "risk": "low", "description":
"Вебсервер без шифрування"},
    "443": {"service": "HTTPS", "risk": "low", "description":
"Захищений вебсервер"}
  },
  "keywords": {
    "SSH": {"risk": "high", "description": "Згадка SSH доступу у
відкритих джерелах"},
    "FTP": {"risk": "high", "description": "Згадка FTP сервісу у
відкритих джерелах"},
    "firewall": {"risk": "medium", "description": "Згадка
налаштувань firewall"},
    "password": {"risk": "high", "description": "Згадка паролів у
відкритих джерелах"},
    "backup": {"risk": "medium", "description": "Згадка резервних
копій"}
  },
  "directory_listing": {
    "risk": "medium",
    "description": "Увімкнено перегляд директорій вебсервера"
```

collector.py

```
import nmap
import requests
from bs4 import BeautifulSoup
from ntscraper import Nitter
TARGET_IP = "192.168.119.129"
TARGET_URL = f"http://{TARGET_IP}"
TWITTER_USERNAME = "AndrijMel18891"

def scan_ports():
    print("[*] Сканування портів...")
    results = []
    try:
        nm = nmap.PortScanner()
        nm.scan(TARGET_IP, '21-22,80,443,8080')
        for host in nm.all_hosts():
            for proto in nm[host].all_protocols():
                for port in nm[host][proto].keys():
                    state = nm[host][proto][port]['state']
                    service = nm[host][proto][port]['name']
```

```
                    if state == 'open':
                        results.append({
                            "source": "port_scan",
                            "type": "open_port",
                            "port": str(port),
                            "service": service,
                            "state": state
                        })
                        print(f"   [+] Порт {port} ({service}) -
{state}")
    except Exception as e:
        print(f"  [!] Помилка сканування: {e}")
    return results
def scrape_website():
    print("[*] Збір даних з вебсайту...")
    results = []
    try:
        # Головна сторінка
        response = requests.get(TARGET_URL, timeout=5)
        soup = BeautifulSoup(response.text, 'html.parser')
        text = soup.get_text()
        results.append({
            "source": "website",
            "type": "webpage",
            "url": TARGET_URL,
            "content": text[:2000]
        })
        print(f"  [+] Головна сторінка зібрана")
        # robots.txt
        robots      =      requests.get(f"{TARGET_URL}/robots.txt",
timeout=5)
        if robots.status_code == 200:
            results.append({
                "source": "website",
                "type": "robots_txt",
                "url": f"{TARGET_URL}/robots.txt",
                "content": robots.text
            })
            print(f"  [+] robots.txt зібрано")
        # Перевірка directory listing
        test_dirs = ["/uploads", "/backup", "/admin", "/files"]
        for d in test_dirs:
            try:
                r = requests.get(f"{TARGET_URL}{d}", timeout=3)
                if r.status_code == 200 and "Index of" in r.text:
                    results.append({
                        "source": "website",
                        "type": "directory_listing",
                        "url": f"{TARGET_URL}{d}",
                        "content":  f"Directory  listing  активний:
{d}"
                    })
                    print(f"  [+] Directory listing знайдено: {d}")
```

```
            except:
                pass
    except Exception as e:
        print(f"  [!] Помилка збору сайту: {e}")
    return results
def scrape_twitter():
    print("[*] Збір даних з Twitter...")
    results = []
    try:
        scraper = Nitter()
        tweets = scraper.get_tweets(TWITTER_USERNAME, mode='user',
number=20)
        if tweets and 'tweets' in tweets:
            for tweet in tweets['tweets']:
                text = tweet.get('text', '')
                results.append({
                    "source": "twitter",
                    "type": "tweet",
                    "username": TWITTER_USERNAME,
                    "content": text
                })
                print(f"  [+] Твіт: {text[:80]}...")
    except Exception as e:
        print(f"  [!] Помилка збору Twitter: {e}")
    return results
def collect_all():
    print("=" * 50)
    print("МОДУЛЬ ЗБОРУ OSINT-ДАНИХ")
    print("=" * 50)
    data = []
    data.extend(scan_ports())
    data.extend(scrape_website())
    data.extend(scrape_twitter())
    print(f"\n[*] Зібрано записів: {len(data)}")
    return data
if __name__ == "__main__":
    results = collect_all()
    for r in results:
        print(r)
```

normalizer.py

```
import json
from datetime import datetime
def normalize_port(item):
    return {
        "id": f"port_{item['port']}",
        "timestamp": datetime.now().isoformat(),
        "source": "port_scan",
        "category": "network",
        "type": "open_port",
        "value": item['port'],
        "service": item['service'],
```

```
        "description":     f"Відкритий     порт     {item['port']}
({item['service']})",
        "raw": item
    }
def normalize_webpage(item):
    return {
        "id": "webpage_main",
        "timestamp": datetime.now().isoformat(),
        "source": "website",
        "category": "web",
        "type": "webpage",
        "value": item['url'],
        "service": "http",
        "description": f"Вебсторінка: {item['url']}",
        "raw":      {"url":      item['url'],      "content_length":
len(item['content'])}
    }
def normalize_robots(item):
    disallowed = []
    for line in item['content'].splitlines():
        if line.startswith("Disallow:"):
            path = line.replace("Disallow:", "").strip()
            if path:
                disallowed.append(path)
    return {
        "id": "robots_txt",
        "timestamp": datetime.now().isoformat(),
        "source": "website",
        "category": "web",
        "type": "robots_txt",
        "value": item['url'],
        "service": "http",
        "description":  f"Приховані  директорії  в  robots.txt:  {',
'.join(disallowed)}",
        "extra": {"disallowed_paths": disallowed},
        "raw": item
    }
def normalize_directory_listing(item):
    return {
        "id": f"dirlist_{item['url'].split('/')[-1]}",
        "timestamp": datetime.now().isoformat(),
        "source": "website",
        "category": "web",
        "type": "directory_listing",
        "value": item['url'],
        "service": "http",
        "description":      f"Directory      listing      активний:
{item['url']}",
        "raw": item
    }
def normalize_tweet(item):
    return {
        "id": f"tweet_{hash(item['content']) % 10000}",
```

```
        "timestamp": datetime.now().isoformat(),
        "source": "twitter",
        "category": "social",
        "type": "tweet",
        "value": item['content'],
        "service": "twitter",
        "description": f"Твіт від @{item['username']}",
        "raw": item
    }
def normalize_all(raw_data):
    print("=" * 50)
    print("МОДУЛЬ НОРМАЛІЗАЦІЇ ДАНИХ")
    print("=" * 50)
    normalized = []
    for item in raw_data:
        try:
            if item['type'] == 'open_port':
                normalized.append(normalize_port(item))
            elif item['type'] == 'webpage':
                normalized.append(normalize_webpage(item))
            elif item['type'] == 'robots_txt':
                normalized.append(normalize_robots(item))
            elif item['type'] == 'directory_listing':

normalized.append(normalize_directory_listing(item))
            elif item['type'] == 'tweet':
                normalized.append(normalize_tweet(item))
        except Exception as e:
            print(f"  [!] Помилка нормалізації: {e}")
    # Додаємо твіти з JSON файлу
    try:
        with open('twitter_data.json', 'r', encoding='utf-8') as f:
            twitter = json.load(f)
            for tweet in twitter['tweets']:
                normalized.append(normalize_tweet({
                    'content': tweet['text'],
                    'username': tweet['username']
                }))
                print(f"           [+]     Твіт     нормалізовано:
{tweet['text'][:60]}...")
    except Exception as e:
        print(f"  [!] Помилка читання twitter_data.json: {e}")
    print(f"\n[*] Нормалізовано записів: {len(normalized)}")
    return normalized
if __name__ == "__main__":
    # Тест нормалізатора
    from collector import collect_all
    raw = collect_all()
    result = normalize_all(raw)
    for r in result:
        print(f"  [{r['category']}] {r['description']}")
```

analyzer.py

```
import json
def load_knowledge_base():
    with open('knowledge_base.json', 'r', encoding='utf-8') as f:
        return json.load(f)
def analyze_port(item, kb):
    findings = []
    port = item['value']
    if port in kb['ports']:
        info = kb['ports'][port]
        findings.append({
            "id": f"finding_port_{port}",
            "source": item['source'],
            "category": "network",
            "type": "port_risk",
            "value": port,
            "service": info['service'],
            "risk_level": info['risk'],
            "description": info['description'],
            "pz": 0.8 if info['risk'] == 'high' else 0.5 if
info['risk'] == 'medium' else 0.2,
            "vo": 0.8 if info['risk'] == 'high' else 0.5 if
info['risk'] == 'medium' else 0.2,
            "ez": 0.3
        })
        print(f"   [!] Порт {port} ({info['service']}) - ризик:
{info['risk']}")
    return findings
def analyze_robots(item, kb):
    findings = []
    paths = item.get('extra', {}).get('disallowed_paths', [])
    for path in paths:
        findings.append({
            "id": f"finding_robots_{path.replace('/', '_')}",
            "source": item['source'],
            "category": "web",
            "type": "hidden_path",
            "value": path,
            "service": "http",
            "risk_level": "medium",
            "description": f"Прихована директорія в robots.txt:
{path}",
            "pz": 0.5,
            "vo": 0.5,
            "ez": 0.4
        })
        print(f"    [!] Прихована директорія: {path} - ризик:
medium")
    return findings
def analyze_directory_listing(item, kb):
    info = kb['directory_listing']
    finding = {
```

```
        "id": f"finding_dirlist",
        "source": item['source'],
        "category": "web",
        "type": "directory_listing",
        "value": item['value'],
        "service": "http",
        "risk_level": info['risk'],
        "description": info['description'],
        "pz": 0.6,
        "vo": 0.5,
        "ez": 0.3
    }
    print(f"      [!]   Directory   listing   активний   -   ризик:
{info['risk']}")
    return [finding]
def analyze_tweet(item, kb):
    findings = []
    content = item['value'].lower()
    for keyword, info in kb['keywords'].items():
        if keyword.lower() in content:
            findings.append({
                "id": f"finding_tweet_{keyword}",
                "source": item['source'],
                "category": "social",
                "type": "social_leak",
                "value": keyword,
                "service": "twitter",
                "risk_level": info['risk'],
                "description": info['description'],
                "pz": 0.7 if info['risk'] == 'high' else 0.4,
                "vo": 0.7 if info['risk'] == 'high' else 0.4,
                "ez": 0.3
            })
            print(f"    [!]  Ключове  слово  '{keyword}'  у  твіті  -
ризик: {info['risk']}")
    return findings
def correlate(findings):
    print("\n[*] Кореляція загроз...")
    ports = [f for f in findings if f['type'] == 'port_risk']
    leaks = [f for f in findings if f['type'] == 'social_leak']
    if len(ports) >= 2 and len(leaks) >= 1:
        findings.append({
            "id": "finding_correlation_001",
            "source": "correlation",
            "category": "combined",
            "type": "combined_risk",
            "value": "multiple",
            "service": "multiple",
            "risk_level": "high",
            "description":   f"Виявлено   {len(ports)}   відкритих
портів  +  витік  інформації  в  соцмережах  —  підвищений  ризик
компрометації",
            "pz": 0.9,
```

```
            "vo": 0.8,
            "ez": 0.2
        })
        print("    [!!!] Кореляція: відкриті порти + витік у
соцмережах = ВИСОКИЙ РИЗИК")
    return findings
def analyze_all(normalized_data):
    print("=" * 50)
    print("МОДУЛЬ АНАЛІЗУ ТА КОРЕЛЯЦІЇ")
    print("=" * 50)
    kb = load_knowledge_base()
    findings = []
    for item in normalized_data:
        if item['type'] == 'open_port':
            findings.extend(analyze_port(item, kb))
        elif item['type'] == 'robots_txt':
            findings.extend(analyze_robots(item, kb))
        elif item['type'] == 'directory_listing':
            findings.extend(analyze_directory_listing(item, kb))
        elif item['type'] == 'tweet':
            findings.extend(analyze_tweet(item, kb))
    findings = correlate(findings)
    print(f"\n[*] Виявлено загроз: {len(findings)}")
    return findings
if __name__ == "__main__":
    from collector import collect_all
    from normalizer import normalize_all
    raw = collect_all()
    normalized = normalize_all(raw)
    findings = analyze_all(normalized)
    for f in findings:
        print(f"  [{f['risk_level'].upper()}] {f['description']}")
```

risk_engine.py

```
import numpy as np
import skfuzzy as fuzz
from skfuzzy import control as ctrl
def create_fuzzy_system():
    # Вхідні змінні
    pz = ctrl.Antecedent(np.arange(0, 1.1, 0.1), 'pz')
    vo = ctrl.Antecedent(np.arange(0, 1.1, 0.1), 'vo')
    ez = ctrl.Antecedent(np.arange(0, 1.1, 0.1), 'ez')
    # Вихідна змінна
    risk = ctrl.Consequent(np.arange(0, 1.1, 0.1), 'risk')
    # Функції належності для pz
    pz['low'] = fuzz.trimf(pz.universe, [0, 0, 0.33])
    pz['medium'] = fuzz.trimf(pz.universe, [0.17, 0.5, 0.83])
    pz['high'] = fuzz.trimf(pz.universe, [0.67, 1, 1])
    # Функції належності для vo
    vo['low'] = fuzz.trimf(vo.universe, [0, 0, 0.33])
    vo['medium'] = fuzz.trimf(vo.universe, [0.17, 0.5, 0.83])
```

```
    vo['high'] = fuzz.trimf(vo.universe, [0.67, 1, 1])
    # Функції належності для ez
    ez['low'] = fuzz.trimf(ez.universe, [0, 0, 0.33])
    ez['medium'] = fuzz.trimf(ez.universe, [0.17, 0.5, 0.83])
    ez['high'] = fuzz.trimf(ez.universe, [0.67, 1, 1])
    # Функції належності для risk
    risk['low'] = fuzz.trimf(risk.universe, [0, 0, 0.33])
    risk['medium'] = fuzz.trimf(risk.universe, [0.17, 0.5, 0.83])
    risk['high'] = fuzz.trimf(risk.universe, [0.67, 1, 1])
    # База правил з таблиці 2.3
    rules = [
        ctrl.Rule(pz['low'] & vo['low'] & ez['high'], risk['low']),
        ctrl.Rule(pz['low']    &    vo['high']    &    ez['low'],
risk['medium']),
        ctrl.Rule(pz['medium']   &   vo['medium']   &   ez['medium'],
risk['medium']),
        ctrl.Rule(pz['high']    &    vo['high']    &    ez['low'],
risk['high']),
        ctrl.Rule(pz['medium']    &    vo['high']    &    ez['high'],
risk['low']),
        ctrl.Rule(pz['high']    &    vo['low']    &    ez['medium'],
risk['high']),
    ]
    system = ctrl.ControlSystem(rules)
    return ctrl.ControlSystemSimulation(system)
def calculate_risk(pz_val, vo_val, ez_val, fuzzy_sim):
    try:
        fuzzy_sim.input['pz'] = pz_val
        fuzzy_sim.input['vo'] = vo_val
        fuzzy_sim.input['ez'] = ez_val
        fuzzy_sim.compute()
        score = fuzzy_sim.output['risk']
        if score >= 0.67:
            level = "HIGH"
        elif score >= 0.33:
            level = "MEDIUM"
        else:
            level = "LOW"
        return round(score, 3), level
    except Exception as e:
        return 0.5, "MEDIUM"
def assess_risks(findings):
    print("=" * 50)
    print("МОДУЛЬ ОЦІНЮВАННЯ РИЗИКІВ")
    print("=" * 50)
    fuzzy_sim = create_fuzzy_system()
    assessed = []
    for finding in findings:
        pz = finding.get('pz', 0.5)
        vo = finding.get('vo', 0.5)
        ez = finding.get('ez', 0.5)
        score, level = calculate_risk(pz, vo, ez, fuzzy_sim)
        finding['risk_score'] = score
```

```
        finding['risk_level_fuzzy'] = level

        assessed.append(finding)
        print(f"    [{level}]  {finding['description'][:60]}...  |
Score: {score}")
    # Сортування за рівнем ризику
    assessed.sort(key=lambda x: x['risk_score'], reverse=True)
    print(f"\n[*] Оцінено загроз: {len(assessed)}")
    return assessed
if __name__ == "__main__":
    from collector import collect_all
    from normalizer import normalize_all
    from analyzer import analyze_all
    raw = collect_all()
    normalized = normalize_all(raw)
    findings = analyze_all(normalized)
    results = assess_risks(findings)
    print("\n--- ПРІОРИТИЗОВАНІ РИЗИКИ ---")
    for r in results:
        print(f"  [{r['risk_level_fuzzy']}] score={r['risk_score']}
| {r['description']}")
```

report.py

```
import json
from datetime import datetime
import matplotlib
matplotlib.use('Agg')
def                          generate_html_report(assessed_findings,
filename="report.html"):
    print("=" * 50)
    print("МОДУЛЬ ВІЗУАЛІЗАЦІЇ ТА ЗВІТНОСТІ")
    print("=" * 50)
    high = [f for f in assessed_findings if f['risk_level_fuzzy']
== 'HIGH']
    medium = [f for f in assessed_findings if f['risk_level_fuzzy']
== 'MEDIUM']
    low = [f for f in assessed_findings if f['risk_level_fuzzy']
== 'LOW']
    timestamp = datetime.now().strftime("%d.%m.%Y %H:%M:%S")
    rows = ""
    for f in assessed_findings:
        color = "#e74c3c" if f['risk_level_fuzzy'] == 'HIGH' else
"#f39c12" if f['risk_level_fuzzy'] == 'MEDIUM' else "#27ae60"
        rows += f"""
        <tr>
            <td>{f['id']}</td>
            <td>{f['category']}</td>
            <td>{f['description']}</td>
            <td             style="color:{color};             font-
weight:bold;">{f['risk_level_fuzzy']}</td>
            <td>{f['risk_score']}</td>
```

```
            <td>{f.get('service', '-')}</td>
        </tr>"""
    html = f"""<!DOCTYPE html>
<html lang="uk">
<head>
    <meta charset="UTF-8">
    <title>OSINT Risk Assessment Report</title>
    <style>
        body {{
            font-family: Arial, sans-serif;
            margin: 30px;
            background: #f5f5f5;
        }}
        h1 {{
            color: #2c3e50;
            border-bottom: 3px solid #e74c3c;
            padding-bottom: 10px;
        }}
        h2 {{
            color: #34495e;
            margin-top: 30px;
            text-align: center;
            font-size: 1.6em;
        }}
        .info {{
            background: white;
            padding: 15px;
            border-radius: 8px;
            margin: 10px 0;
            box-shadow: 0 2px 6px rgba(0,0,0,0.08);
        }}
        .stats-center {{
            text-align: center;
            margin: 40px 0;
        }}
        .stats-row {{
            display: flex;
            justify-content: center;
            gap: 40px;
            margin-top: 20px;
        }}
        .stat-box {{
            padding: 40px 60px;
            border-radius: 16px;
            background: white;
            box-shadow: 0 6px 20px rgba(0,0,0,0.12);
            min-width: 200px;
        }}
        .stat-number {{
            font-size: 5em;
            font-weight: bold;
            color: #2c3e50;
        }}
```

```
        .stat-label {{
            font-size: 1.3em;
            color: #7f8c8d;
            margin-top: 8px;
        }}
        .stat-high .stat-number {{ color: #e74c3c; }}
        .stat-medium .stat-number {{ color: #f39c12; }}
        .stat-low .stat-number {{ color: #27ae60; }}
        .stat-total .stat-number {{ color: #2c3e50; }}
        table {{
            width: 100%;
            border-collapse: collapse;
            background: white;
            border-radius: 8px;
            overflow: hidden;
            box-shadow: 0 2px 8px rgba(0,0,0,0.08);
        }}
        th {{
            background: #2c3e50;
            color: white;
            padding: 14px;
            text-align: left;
            font-size: 1em;
        }}
        td {{
            padding: 12px 14px;
            border-bottom: 1px solid #eee;
            font-size: 0.95em;
        }}
        tr:hover {{ background: #f9f9f9; }}
        .footer {{
            margin-top: 40px;
            color: #7f8c8d;
            font-size: 0.9em;
            text-align: center;
        }}
    </style>
</head>
<body>
    <h1>OSINT Risk Assessment Report</h1>
    <div class="info">
        <strong>Організація:</strong> Organization A  | 
        <strong>Дата:</strong> {timestamp}  | 
        <strong>Цільова система:</strong> 192.168.119.129
    </div>
    <div class="stats-center">
        <h2>Загальна статистика виявлених загроз</h2>
        <div class="stats-row">
            <div class="stat-box stat-total">
                <div                                    class="stat-
number">{len(assessed_findings)}</div>
                <div class="stat-label">Всього загроз</div>
            </div>
```

```
            <div class="stat-box stat-high">
                <div class="stat-number">{len(high)}</div>
                <div class="stat-label">Високий ризик</div>
            </div>
            <div class="stat-box stat-medium">
                <div class="stat-number">{len(medium)}</div>
                <div class="stat-label">Середній ризик</div>
            </div>
            <div class="stat-box stat-low">
                <div class="stat-number">{len(low)}</div>
                <div class="stat-label">Низький ризик</div>
            </div>
        </div>
    </div>
    <h2>Виявлені загрози</h2>
    <table>
        <thead>
            <tr>
                <th>ID</th>
                <th>Категорія</th>
                <th>Опис</th>
                <th>Рівень ризику</th>
                <th>Score</th>
                <th>Сервіс</th>
            </tr>
        </thead>
        <tbody>
            {rows}
        </tbody>
    </table>
    <div class="footer">
        <p>Звіт  згенеровано  автоматично  системою  OSINT  Risk
Assessment | {timestamp}</p>
    </div>
</body>
</html>"""
    with open(filename, 'w', encoding='utf-8') as f:
        f.write(html)
    print(f"  [+] HTML звіт збережено: {filename}")
    return filename
if __name__ == "__main__":
    from collector import collect_all
    from normalizer import normalize_all
    from analyzer import analyze_all
    from risk_engine import assess_risks
    raw = collect_all()
    normalized = normalize_all(raw)
    findings = analyze_all(normalized)
    assessed = assess_risks(findings)
    generate_html_report(assessed)
```

main.py

```
import json
from collector import collect_all
from normalizer import normalize_all
from analyzer import analyze_all
from risk_engine import assess_risks
from report import generate_html_report
import webbrowser
import os
def main():
    print("\n" + "=" * 50)
    print("  СИСТЕМА ОЦІНЮВАННЯ РИЗИКІВ НА ОСНОВІ OSINT")
    print("  Організація: Organization A")
    print("=" * 50 + "\n")
    # Крок 1 - Збір даних
    raw_data = collect_all()
    # Крок 2 - Нормалізація
    normalized_data = normalize_all(raw_data)
    # Крок 3 - Аналіз та кореляція
    findings = analyze_all(normalized_data)
    # Крок 4 - Оцінювання ризиків
    assessed = assess_risks(findings)
    # Крок 5 - Звіт
    report_file = generate_html_report(assessed)
    # Збереження результатів у JSON
    with open('results.json', 'w', encoding='utf-8') as f:
        json.dump(assessed, f, ensure_ascii=False, indent=2)
    print(f"\n[*] Результати збережено у results.json")
    # Підсумок
    high = [f for f in assessed if f['risk_level_fuzzy'] == 'HIGH']
    medium = [f for f in assessed if f['risk_level_fuzzy'] ==
'MEDIUM']
    low = [f for f in assessed if f['risk_level_fuzzy'] == 'LOW']
    print("\n" + "=" * 50)
    print("  ПІДСУМОК ОЦІНЮВАННЯ")
    print("=" * 50)
    print(f"  Високий ризик:  {len(high)}")
    print(f"  Середній ризик: {len(medium)}")
    print(f"  Низький ризик:  {len(low)}")
    print(f"  Всього загроз:  {len(assessed)}")
    print("=" * 50)
    # Відкрити звіт у браузері
    report_path = os.path.abspath(report_file)
    print(f"\n[*] Відкриваємо звіт у браузері...")
    webbrowser.open(f"file://{report_path}")
if __name__ == "__main__":
    main()
```