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

**КАФЕДРА СИСТЕМ ТА ТЕХНОЛОГІЙ КІБЕРБЕЗПЕКИ**

# КВАЛІФІКАЦІЙНА РОБОТА

на тему:

**«Технологія автоматизації OSINT-аналізу з використанням інструментів штучного інтелекту»**

зі спеціальності *125 Кібербезпека та захист інформації*

*(код, найменування спеціальності)*

освітньо-професійної програми *Інформаційна та кібернетична безпека*

*(назва програми)*

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

_____ Марсель МАКСУТОВ
*(підпис)*

Виконав: здобувач вищої освіти групи БСДМ-63
МАКСУТОВ Марсель

*(прізвище, ім'я)*

Керівник д.т.н., професор КАЗМІРЧУК Світлана

*(науковий ступінь, вчене звання, прізвище, ім'я)*

Рецензент

*(науковий ступінь, вчене звання, прізвище, ім'я)*

Київ 2025

**ДЕРЖАВНИЙ УНІВЕРСИТЕТ ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ**
**НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ЗАХИСТУ ІНФОРМАЦІЇ**

Кафедра Систем та технологій кібербезпеки

Ступінь вищої освіти Магістр

Спеціальність 125 Кібербезпека та захист інформації

Освітньо-професійна програма Інформаційна та кібернетична безпека

ЗАТВЕРДЖУЮ

Завідувач кафедри
Систем та технологій кібербезпеки
Галина ГАЙДУР

“___” жовтня 2025 року

# З А В Д А Н Н Я
## НА КВАЛІФІКАЦІЙНУ РОБОТУ

МАКСУТОВУ Марселю Олександровичу

*(прізвище, ім’я)*

1. Тема кваліфікаційної роботи: «Технологія автоматизації OSINT-аналізу з використанням інструментів штучного інтелекту

керівник кваліфікаційної роботи Казмірчук Світлана Володимирівна, д.т.н., професор

*(прізвище, ім’я, науковий ступінь, вчене звання)*

затверджені наказом Державного університету інформаційно-комунікаційних технологій від «___» жовтня 2025 року № ___.

2. Строк подання здобувачем вищої освіти кваліфікаційної роботи 15.12.2025 р.

3. Вихідні дані до кваліфікаційної роботи

інформаційні ресурси організації;

Інструменти OSINT-аналізу;

наукова та технічна література, експлуатаційна документація, нормативні документи, міжнародні стандарти.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)

1. Дослідження проблеми використання технологій OSINT у сфері кібербезпеки.

2. Аналіз існуючих технологій та можливостей їх інтеграції з інструментами ШІ.

3. Розробка та впровадження технології автоматизації OSINT-аналізу

4. Перелік графічного матеріалу

Презентація PowerPoint.

5. Дата видачі завдання 01.10.2025 р.

## КАЛЕНДАРНИЙ ПЛАН

| № зп | Назва етапів кваліфікаційної роботи | Строк виконання етапів роботи | Примітка |
|---|---|---|---|
| 1. | Визначення актуальності проблеми автоматизації OSINT-аналізу з використанням інструментів штучного інтелекту. | 01.10.2025 р. | |
| 2. | Аналіз наукової та технічної літератури з питань теми кваліфікаційної роботи. | 12.10.2025 р. | |
| 3. | Аналіз методів та засобів OSINT-досліджень та можливості їх інтеграції з інструментами ШІ. | 27.10.2025 р. | |

| | | | |
|---|---|---|---|
| 4. | Технологія автоматизації OSINT-аналізу з використанням мовних моделей і спеціалізованих інструментів. | 03.11.2025 р. | |
| 5. | Розроблення рекомендацій щодо застосування технології ШІ для підвищення ефективності OSINT-аналізу у практичній діяльності. | 15.11.2025 р. | |
| 6. | Оформлення результатів дослідження. | 26.11.2025 р. | |
| 7. | Підготовка доповіді до захисту. | 15.12.2025 р. | |

Здобувач вищої освіти _____ Марсель МАКСУТОВ
*(підпис)* *(ім'я, прізвище)*

Керівник кваліфікаційної роботи _____ Світлана КАЗМІРЧУК
*(підпис)* *(ім'я, прізвище)*

**ВІДГУК РЕЦЕНЗЕНТА**
на кваліфікаційну роботу

здобувача МАКСУТОВА Марселя

на тему: «Технологія автоматизації OSINT-аналізу з використанням інструментів штучного інтелекту»

**Актуальність:** У сучасних умовах кіберпростір перетворюється на ключове середовище протистояння між організаціями та зловмисниками. Відкриті джерела інформації (OSINT) стають важливим інструментом для виявлення потенційних загроз, збору даних про атаки та формування превентивних заходів захисту. Проте зростаючі обсяги даних, їхня динаміка та різнорідність ускладнюють ручний аналіз і потребують автоматизації.

Використання інструментів штучного інтелекту (зокрема, мовних моделей і систем машинного навчання) у поєднанні з технологіями OSINT дозволяє суттєво підвищити швидкість і якість обробки інформації, автоматизувати пошук аномалій і формувати прогнози. Це знижує вплив людського фактору та дає можливість ефективніше реагувати на кіберзагрози. Таким чином, тема є актуальною й відповідає стратегічним завданням розвитку кіберзахисту.

**Позитивні сторони:**

У роботі чітко визначено роль OSINT у сучасній кібербезпеці та розкрито проблеми, які виникають при його використанні без автоматизації.

1. Досліджено інструменти класичного OSINT-аналізу (Maltego, SpiderFoot, Shodan) та можливості їх інтеграції з інструментами штучного інтелекту.
2. Представлено технологію автоматизації OSINT-аналізу з використанням ІІ та продемонстровано приклади реалізації прототипу (Python, GPT-моделі, SpiderFoot API).
3. Робота викладена грамотно й послідовно, наявні якісні висновки та рекомендації. Використано сучасні джерела, що підтверджує обізнаність автора з новітніми тенденціями у сфері OSINT і штучного інтелекту.

**Недоліки:**

1. У кваліфікаційній роботі доцільно було б детальніше продемонструвати результати практичного застосування запропонованої технології на реальних кейсах OSINT-розслідувань.
2. Аналіз ефективності інтегрованого підходу варто було б доповнити порівнянням із класичними інструментами OSINT без використання ІІ, щоб обґрунтувати переваги автоматизації.

Відзначені зауваження не впливають на загальну позитивну оцінку кваліфікаційної роботи.

**Висновок:** Враховуючи недоліки, кваліфікаційна робота заслуговує оцінку **“добре”**, а здобувач(ка) **МАКСУТОВ Марсель** – присвоєння кваліфікації магістр з кібербезпеки за освітньою програмою інформаційна та кібернетична безпека.

Рецензент:

| _____ | _____ | _____ |
|---|---|---|
| *(науковий ступінь, вчене звання)* | *(підпис)* | *(ім’я, прізвище)* |

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

# ПОДАННЯ ГОЛОВІ ЕКЗАМЕНАЦІЙНОЇ КОМІСІЇ ЩОДО ЗАХИСТУ КВАЛІФІКАЦІЙНОЇ РОБОТИ

Направляється здобувач МАКСУТОВ Марсель до захисту кваліфікаційної роботи
*(прізвище, ім'я)*
спеціальності 125 Кібербезпека та захист інформації
освітньо-професійної програми Інформаційна та кібернетична безпека
*(шифр і назва спеціальності)*
на тему: «Технологія автоматизації OSINT-аналізу з використанням інструментів штучного інтелекту».
Кваліфікаційна робота і рецензія додаються.

Директор інституту _____ Євгенія ІВАНЧЕНКО
*(підпис)* *(ім'я, прізвище)*

## Висновок керівника кваліфікаційної роботи

Здобувач ***МАКСУТОВ Марсель*** Обрано тему роботи, метою якої було дослідити технології автоматизації OSINT-аналізу з використанням інструментів штучного інтелекту та розробити рекомендації щодо їх практичного застосування. Перелік використаних джерел підтверджує вміння здобувача орієнтуватися у сучасних наукових підходах до обробки відкритих даних, а також у тенденціях розвитку штучного інтелекту та їх інтеграції у сферу кібербезпеки. Роботу виконував сумлінно, відповідально та у визначені терміни згідно із затвердженим планом.

Все це дозволяє оцінити виконану кваліфікаційну роботу здобувача ***МАКСУТОВА Марселя*** на оцінку **"добре"** та присвоїти йому кваліфікацію магістр з кібербезпеки за освітньою програмою інформаційна та кібернетична безпека

Керівник кваліфікаційної роботи _____ Світлана КАЗМІРЧУК
*(підпис)* *(ім'я, прізвище)*
"____" _____ 2025 року

## Висновок кафедри про кваліфікаційну роботу

Кваліфікаційна робота розглянута. Здобувач **МАКСУТОВ Марсель** допускається до захисту даної кваліфікаційної роботи в Екзаменаційній комісії.

Завідувач кафедри Систем та технологій кібербезпеки

_____ Галина ГАЙДУР
*(підпис)* *(ім'я, прізвище)*

# РЕФЕРАТ

Текстова частина кваліфікаційної роботи: 75 сторінок, 27 рисунків, 4 таблиці, 20 джерел.

*Об'єкт дослідження* – процес проведення OSINT-аналізу в сфері кібербезпеки.

*Предмет дослідження* – технології автоматизації OSINT-аналізу з використанням інструментів штучного інтелекту.

*Мета роботи* – розробити порядок застосування технологій автоматизованого збору та аналізу відкритих даних із використанням інструментів штучного інтелекту та сформулювати рекомендації щодо їх практичної реалізації у сфері кіберзахисту.

*Методи дослідження* – опрацювання науково-технічної літератури та публікацій у сфері OSINT та штучного інтелекту, аналіз існуючих інструментів OSINT (Maltego, SpiderFoot, Shodan), вивчення можливостей сучасних мовних моделей (LLM), експериментальна інтеграція API та бібліотек Python, а також порівняння з міжнародними стандартами й кращими практиками у сфері кібербезпеки.

Однією з ключових проблем OSINT-аналізу є величезні обсяги та різнорідність даних, що ускладнює їх ручне опрацювання. Автоматизація цього процесу за допомогою штучного інтелекту дозволяє значно скоротити час аналізу, підвищити якість результатів та знизити вплив людського фактору. Використання інструментів ІІ (зокрема GPT-моделей) у поєднанні з класичними OSINT-платформами забезпечує ефективний пошук релевантної інформації, класифікацію даних, виявлення аномалій та формування аналітичних звітів.

У роботі проведено аналіз ролі OSINT у сучасній кібербезпеці, визначено його проблеми та обмеження, досліджено функціональні можливості інструментів штучного інтелекту для автоматизації збору та обробки відкритих даних. Розроблено прототип технології, який поєднує Python, GPT-моделі та SpiderFoot API для автоматизованого OSINT-аналізу. Запропоновано практичні рекомендації щодо використання штучного інтелекту у сфері кіберзахисту з акцентом на ефективність та зручність для фахівців.

Галузь використання – кібербезпека інформаційних систем і ресурсів організацій, зокрема в напрямку аналітики загроз та проведення OSINT-розслідувань.

OSINT, АВТОМАТИЗАЦІЯ, ШТУЧНИЙ ІНТЕЛЕКТ, АНАЛІЗ ДАНИХ, КІБЕРБЕЗПЕКА, SPIDERFOOT, SHODAN, MALTEGO.

## ABSTRACT

Text part of the qualification work: 75 pages, 27 figures, 4 tables, 20 sources.

The object of research is the process of conducting OSINT (Open Source Intelligence) analysis in the field of cybersecurity.

The subject of the research is the the technologies for automating OSINT analysis using artificial intelligence (AI) tools.

The purpose of the study is to develop a procedure for applying AI-based methods to automate the collection, processing, and analysis of open-source data and to provide recommendations for their practical implementation in cybersecurity practice.

Research methods are the study of scientific and technical literature on OSINT and artificial intelligence, analysis of existing OSINT tools (Maltego, SpiderFoot, Shodan), exploration of large language model (LLM) capabilities, experimental integration of Python libraries and APIs, and comparison with international standards and best practices in cybersecurity.

One of the main challenges of OSINT analysis is the large volume and heterogeneity of open-source data, which complicates manual processing. Automation of this process through AI tools significantly reduces analysis time, increases result accuracy, and minimizes the impact of the human factor. The use of AI (particularly GPT-based models) in combination with classical OSINT platforms enables efficient information retrieval, data classification, anomaly detection, and generation of analytical reports. This approach allows faster identification of potential cyber threats, better understanding of digital footprints, and prediction of risks.

The thesis analyzes the role of OSINT in modern cybersecurity, defines its challenges and limitations, and explores the potential of AI tools for automating open-source intelligence tasks. A prototype technology has been developed that integrates Python, GPT models, and SpiderFoot API for automated OSINT analysis. Practical recommendations are proposed for cybersecurity specialists on how to leverage AI tools to enhance the efficiency and reliability of OSINT investigations.

Application area - cybersecurity of information systems and resources, particularly in the field of threat intelligence and OSINT investigations.

OSINT, AUTOMATION, ARTIFICIAL INTELLIGENCE, DATA ANALYSIS, CYBERSECURITY, SPIDERFOOT, SHODAN, MALTEGO

**ЗМІСТ**

# ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

| | | |
|---|---|---|
| OSINT | – | Open Source Intelligence |
| LLM | – | Large Language Model |
| NLP | – | Natural Language Processing |
| API | – | Application Programming Interface |
| JSON | – | JavaScript Object Notation |
| SQL | – | Structured Query Language |
| DNS | – | Domain Name System |
| CTI | – | Cyber Threat Intelligence |
| SpiderFoot | – | Інструмент для автоматизації OSINT- досліджень |
| Maltego | – | Інструмент для збору та візуалізації OSINT-даних |
| Shodan | – | Пошукова система пристроїв Інтернету речей |
| SIEM | – | Security Information and Event Management |
| EDR | – | Endpoint Detection and Response |
| RAG | – | Retrieval-Augmented Generation |
| MTTD | – | Mean time to detect |
| MTTR | – | Mean time to repair |

# ВСТУП

*Актуальність дослідження.* У сучасних умовах розвитку цифрового суспільства відкрита інформація (OSINT – Open Source Intelligence) стала одним із ключових джерел для збору даних у сфері кібербезпеки, розвідки та правоохоронної діяльності. Кількість відкритих джерел, таких як соціальні мережі, форуми, реєстри доменних імен, телекомунікаційні дані та інші цифрові сліди, зростає експоненційно. Це створює як нові можливості для виявлення кіберзагроз, так і серйозні виклики, адже обсяг і різнорідність даних унеможливлюють їх ефективний аналіз виключно вручну.

Традиційні методи OSINT передбачають застосування спеціалізованих інструментів (Maltego, SpiderFoot, Shodan), які дозволяють автоматизувати окремі етапи збору інформації. Проте такі засоби часто обмежені за функціоналом, потребують значних ресурсів для обробки результатів і не завжди здатні своєчасно виявити приховані взаємозв’язки чи аномалії. У цьому контексті особливого значення набуває інтеграція технологій штучного інтелекту, зокрема великих мовних моделей (LLM), алгоритмів машинного навчання та інструментів NLP, у процес OSINT-аналізу.

Застосування штучного інтелекту уможливлює автоматичну класифікацію та фільтрацію великих масивів даних, виявлення індикаторів компрометації (IoC), формування прогнозів і створення аналітичних звітів у режимі, близькому до реального часу. Такий підхід мінімізує вплив людського фактору та підвищує ефективність розслідувань.

Вищезазначене визначає актуальність теми кваліфікаційної роботи, яка присвячена розробці та впровадженню технологій автоматизації OSINT-аналізу з використанням інструментів штучного інтелекту.

*Об'єкт дослідження* – процес проведення OSINT-аналізу у сфері кібербезпеки.

*Предмет дослідження* – технології автоматизації OSINT-аналізу з використанням інструментів штучного інтелекту.

*Мета роботи* – розробити порядок застосування технологій автоматизованого збору та обробки відкритих даних із використанням інструментів штучного інтелекту та сформулювати рекомендації щодо їх практичної реалізації.

*Наукові завдання:*

дослідити сутність проблеми застосування OSINT у кібербезпеці;

проаналізувати сучасні підходи та інструменти для OSINT-розслідувань;

визначити обмеження традиційних технологій та виклики, пов'язані з великими обсягами відкритих даних;

дослідити можливості інтеграції штучного інтелекту (LLM, ML, NLP) у процес OSINT-аналізу;

розробити технологію автоматизації OSINT-аналізу та запропонувати рекомендації для її застосування у практиці кіберзахисту.

розкрити порядок реалізації технології забезпечення безпечної роботи гібридних працівників організації.

*Методи дослідження* – аналіз наукової та технічної літератури, огляд інструментів OSINT, експериментальна інтеграція Python-бібліотек і API (SpiderFoot, GPT-моделі), тестування алгоритмів машинного навчання для обробки даних, а також порівняння з міжнародними практиками у сфері кібербезпеки.

*Практичне значення одержаних результатів* - запропонована технологія дозволяє автоматизувати OSINT-аналіз, зменшити час обробки відкритих даних, підвищити якість результатів і надати фахівцям із кібербезпеки дієві інструменти для моніторингу та прогнозування кіберзагроз.

Результати кваліфікаційної роботи апробовані на Всеукраїнській науковій конференції «Актуальні проблеми кібербезпеки», яка відбулася 29 жовтня 2025 року в Державному університеті інформаційно-комунікаційних технологій, м. Київ.

# 1 ДОСЛІДЖЕННЯ ПРОБЛЕМИ ВИКОРИСТАННЯ ТЕХНОЛОГІЙ OSINT У СФЕРІ КІБЕРБЕЗПЕКИ

## 1.1. Дослідження ролі технологій OSINT у сучасній кібербезпеці

У добу цифрової трансформації та глобальної мережевої взаємодії поняття інформаційної безпеки набуває нового змісту. Класичні методи захисту периметра вже не є достатніми, оскільки сучасні загрози мають динамічний і розподілений характер. Відповідно, зростає потреба в таких інструментах, які дають змогу отримувати контекст не лише з внутрішніх систем компанії, але й із відкритих джерел інформації. Саме цю роль виконує OSINT (Open Source Intelligence) — розвідка на основі відкритих джерел, що охоплює збір, систематизацію й аналіз публічно доступних даних для цілей інформаційної безпеки [1].

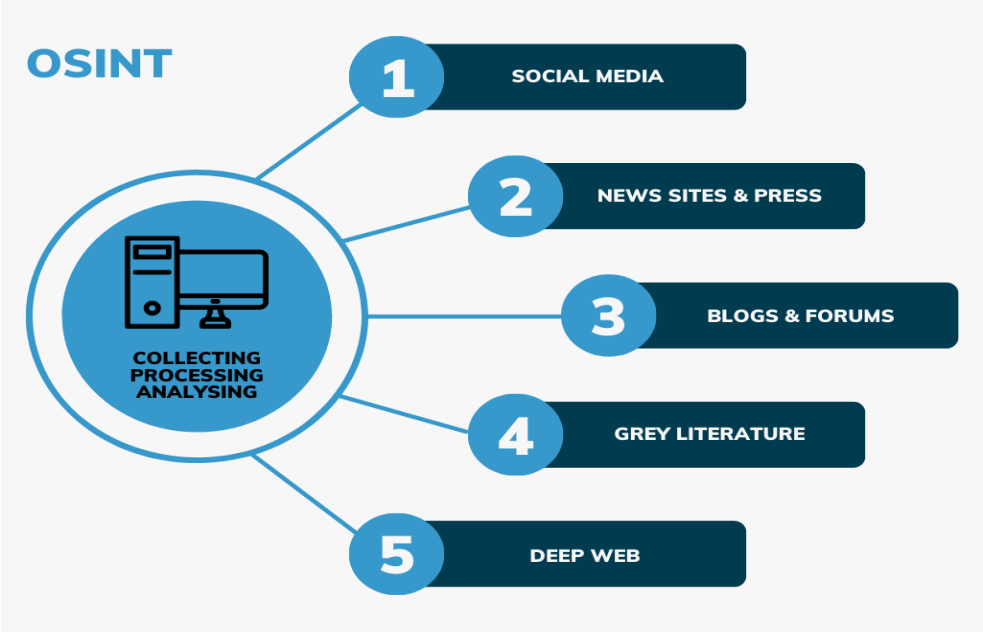


Рис. 1. 1. OSINT-джерела

Застосування OSINT у кібербезпеці дозволяє організаціям бачити повнішу картину кіберзагроз. Наприклад, якщо внутрішні системи моніторингу (SIEM чи EDR) фіксують лише активність усередині корпоративного середовища, то OSINT розширює цей горизонт, надаючи інформацію про потенційні атаки ще на етапі їх підготовки. Виявлення у даркнет-мережах витоків облікових даних, пропозицій купівлі експлойтів або продажу доступів до корпоративних систем дає змогу реагувати до того, як атака відбудеться [2].

OSINT сьогодні виконує низку ключових функцій. Передусім — це раннє попередження про загрози. Аналіз соціальних мереж, форумів, спеціалізованих майданчиків для хакерських спільнот дозволяє виявляти сигнали про майбутні атаки чи появу нових вразливостей. Такі дані можна використовувати для формування індикаторів компрометації (IoC), які згодом інтегруються у SIEM або SOAR-системи для побудови правил виявлення атак. Інша важлива функція OSINT — підтримка цифрової криміналістики та розслідувань. У випадку кібератак інструменти OSINT допомагають встановити, хто стоїть за атакою, з яких ресурсів вона організовувалася, які інфраструктурні компоненти були задіяні (IP-адреси, домени, сервери C2), а також знайти зв’язки між різними інцидентами [3].

Важливим аспектом є прогнозування тенденцій розвитку загроз. На основі відкритих джерел можна визначати, які вектори атак стають більш поширеними, які нові методи соціальної інженерії застосовуються, які вразливості експлуатуються найчастіше. Наприклад, аналіз обговорень у Twitter чи Telegram дозволяє виявити тренди щодо експлуатації нових CVE ще до того, як ці вразливості масово почнуть використовуватися у диких атаках. Таким чином, OSINT перетворюється на інструмент не тільки для реактивного, але й для проактивного кіберзахисту [2].

Окрім практичного використання у корпоративному середовищі, OSINT активно застосовується і на державному рівні. Розвідслужби, правоохоронні органи та CERT-команди використовують відкриті джерела для виявлення кібершпигунських кампаній, дезінформаційних операцій, злочинних груп, що діють у даркнеті. Це

підкреслює універсальність OSINT, який може бути корисним як у масштабі глобальної кібербезпеки, так і в локальних бізнес-сценаріях.

Значна увага приділяється й технологічному розвитку OSINT. Класичні інструменти на кшталт Maltego, SpiderFoot чи Shodan забезпечують автоматизацію збору даних із сотень джерел. Наприклад, Maltego дозволяє будувати карти зв’язків між суб’єктами (IP, доменами, акаунтами у соцмережах), візуалізуючи ці відносини у вигляді графів. SpiderFoot надає змогу автоматично сканувати відкриті ресурси для пошуку витоків, вразливостей або компрометованих облікових даних. Shodan спеціалізується на пошуку відкритих пристроїв в Інтернеті речей, показуючи, які системи можуть бути під загрозою. Додавання у роботу таких скріншотів, як приклад графа у Maltego або дешборд у SpiderFoot, зробить цей розділ дипломної роботи більш наочним і зрозумілим.


Рис. 1. 2. Етапи OSINT - дослідження

Попри очевидні переваги, OSINT має низку проблем і викликів. Серед них — величезний обсяг даних, який неможливо ефективно опрацьовувати вручну. Крім того, відкрита інформація може бути навмисно сфальсифікованою, що створює ризик дезінформації. Тому сьогодні зростає інтерес до інтеграції OSINT із технологіями штучного інтелекту. Використання алгоритмів машинного навчання та мовних моделей дозволяє автоматизувати процеси класифікації даних, виявлення фейкової інформації, розпізнавання закономірностей і навіть прогнозування потенційних атак [1]. Інший виклик — правові та етичні питання: використання OSINT повинно узгоджуватися з вимогами законодавства, такими як GDPR, та принципами приватності.

Таким чином, можна стверджувати, що роль OSINT у сучасній кібербезпеці є визначальною та багатовимірною. Вона охоплює не лише технічний аспект виявлення інцидентів, але й стратегічний рівень прогнозування загроз та підтримки управлінських рішень у сфері кіберзахисту. У поєднанні з інструментами штучного інтелекту OSINT стає ще потужнішим, адже дозволяє не просто збирати дані, а й перетворювати їх на систематизовану аналітику. Включення в диплом практичних скріншотів роботи з Maltego, SpiderFoot та Shodan допоможе підтвердити, що OSINT — це не лише теоретична концепція, а реальний інструмент, який уже сьогодні активно застосовується для боротьби з кіберзагрозами.

### 1.2. Аналіз проблем та обмежень існуючих технологій OSINT

Попри високу популярність та значні перспективи застосування OSINT у сфері кібербезпеки, існуючі технології мають цілу низку проблем та обмежень, які безпосередньо впливають на якість та надійність отриманих результатів. Розвідка з відкритих джерел завжди була потужним інструментом для збору даних, але з розвитком інформаційного суспільства вона зіткнулася з новими викликами — від

надмірного обсягу даних до технологічних, правових і навіть етичних бар'єрів. У цьому підрозділі буде проведено глибокий аналіз цих проблем, підкріплений прикладами, практичними кейсами та науковими джерелами.

Одним із найбільших викликів є те, що відкриті джерела генерують інформацію у величезних масштабах. Щодня у світі створюються мільярди повідомлень у соціальних мережах, форумах, блогах, державних базах даних, на сайтах компаній та у даркнет-спільнотах. З одного боку, це створює безпрецедентні можливості для розвідки, адже в публічному просторі можна знайти фактично будь-які дані про людину, організацію чи інфраструктуру. З іншого боку, надлишок інформації робить роботу OSINT-аналітиків надзвичайно складною.

Складність полягає не лише в кількості, а й у різнорідності даних. Частина даних структурована (наприклад, відкриті реєстри чи витоки баз даних у форматі CSV), інша — неструктурована (тексти постів, коментарі, зображення, відео). Крім того, дані подаються різними мовами, із застосуванням скорочень, сленгу, специфічних культурних чи професійних кодів. Відповідно, аналітик змушений поєднувати інструменти обробки тексту, зображень, відео та аудіо, а також враховувати культурні та мовні особливості. Це створює підґрунтя для появи «шуму» — величезної кількості даних, які не мають цінності для кібербезпеки, але відволікають ресурси на їх обробку.

Прикладом може слугувати використання системи SpiderFoot: навіть за умов базового сканування домену інструмент може повернути тисячі результатів — IP-адреси, історичні дані з WHOIS, пов'язані електронні адреси та логіни у форумах. Проте далеко не всі ці результати мають реальну загрозу. Часто значна частина даних виявляється дублюючою або застарілою, і лише невеликий відсоток має безпосередню цінність. Це призводить до проблеми хибнопозитивних спрацьовувань, коли системи сигналізують про загрозу, якої насправді немає [1].


Рис. 1.3. Система OSINT SpiderFoot

Ще одним критично важливим обмеженням OSINT є проблема достовірності отриманих даних. Відкриті джерела за своєю природою є неконтрольованими, а це означає, що у них може міститися велика кількість дезінформації, фейкових новин, неточностей та навмисно сфальсифікованих матеріалів.

Наприклад, у соціальних мережах масово поширюються «бот-мережі», які генерують фейкові повідомлення для маніпулювання громадською думкою. У даркнет-спільнотах зловмисники іноді навмисно розміщують неправдиві пропозиції продажу даних чи «зламаних доступів», щоб відволікти дослідників або створити ілюзію певної активності. Це ускладнює аналіз, адже аналітики повинні розрізняти достовірні дані від маніпулятивних.

Braga (2025) зазначає, що головна небезпека полягає у «сліпій довірі» до інформації, яка багаторазово повторюється у відкритих джерелах. Якщо кілька ресурсів поширюють один і той самий фейковий контент, виникає ілюзія достовірності, і навіть досвідчені аналітики можуть помилково прийняти його за правду [4]. Це вимагає обов’язкової багатоканальної перевірки та використання методів перехресної валідації.

У сучасних умовах особливу загрозу для OSINT становить поширення технологій deepfake. Генеративний штучний інтелект дозволяє створювати

зображення, аудіо та відео, які виглядають абсолютно реалістично, але насправді є штучно синтезованими. Це ускладнює верифікацію даних у рамках OSINT, адже класичні методи перевірки (пошук за зворотними зображеннями, аналіз метаданих) стають малоефективними.

Singh (2025) наголошує, що сучасні системи виявлення deepfake працюють нестабільно: алгоритми швидко застарівають, тоді як інструменти для створення синтетичного контенту розвиваються надзвичайно швидко [5]. У результаті OSINT-аналітики часто опиняються у ситуації, коли автоматичні інструменти виявлення не дають гарантованого результату, і доводиться вдаватися до експертної оцінки вручну. Це уповільнює роботу та вимагає залучення висококваліфікованих кадрів.


Рис. 1.4. Загрози DeepFake технології

Незважаючи на те, що OSINT працює з відкритими даними, це не означає, що такі дані можна вільно збирати й використовувати. Діяльність у сфері OSINT підпадає під дію законодавства про захист персональних даних і приватності. У Європі діє Загальний регламент про захист даних (GDPR), який обмежує обробку навіть тих даних, що є публічно доступними. Це стосується, зокрема, персональних даних користувачів у соцмережах або інформації, опублікованої в державних реєстрах.

Таким чином, OSINT-аналітики мають працювати в умовах «правового поля», документуючи свої дії та пояснюючи, на якій підставі відбувається збір і використання даних. Інакше результати OSINT-досліджень можуть бути визнані незаконними, що матиме наслідки не лише для юридичної сили доказів, але й для репутації організації.

Крім правових питань, існують і етичні дилеми. Наприклад, чи етично використовувати фото та особисті дані, що людина опублікувала у відкритому доступі, але не передбачала їх застосування для кіберрозслідувань? Ці питання залишаються відкритими і потребують вироблення чітких етичних стандартів у спільноті OSINT.

Ще одна група обмежень стосується технічної надійності інструментів. Багато OSINT-рішень працюють за рахунок інтеграції з API соціальних мереж, пошукових систем чи баз даних. Але політика доступу до API постійно змінюється. Те, що сьогодні працює, завтра може бути заблоковане або обмежене.

Наприклад, у 2023–2024 роках Twitter (X) різко обмежив доступ до свого API, що зробило значну частину OSINT-інструментів менш ефективними. Аналогічні зміни відбуваються і в Google, Facebook чи Telegram. Це означає, що аналітики повинні передбачати резервні канали збору даних і враховувати ризик зникнення ключових джерел.

Окрім цього, існує проблема операційної безпеки (OPSEC) самих дослідників. Працюючи з акаунтами у соціальних мережах чи форумах, аналітики можуть випадково розкрити свою присутність або належність до певної організації. Це

створює ризики протидії з боку зловмисників, які здатні навмисно підміняти дані або атакувати дослідників.

Загрози кібербезпеці носять глобальний характер, і тому дані про них можуть публікуватися будь-якою мовою. Це створює серйозні труднощі для автоматичної обробки: навіть сучасні NLP-системи не завжди коректно працюють із низькоресурсними мовами, діалектами чи специфічними жаргонами.

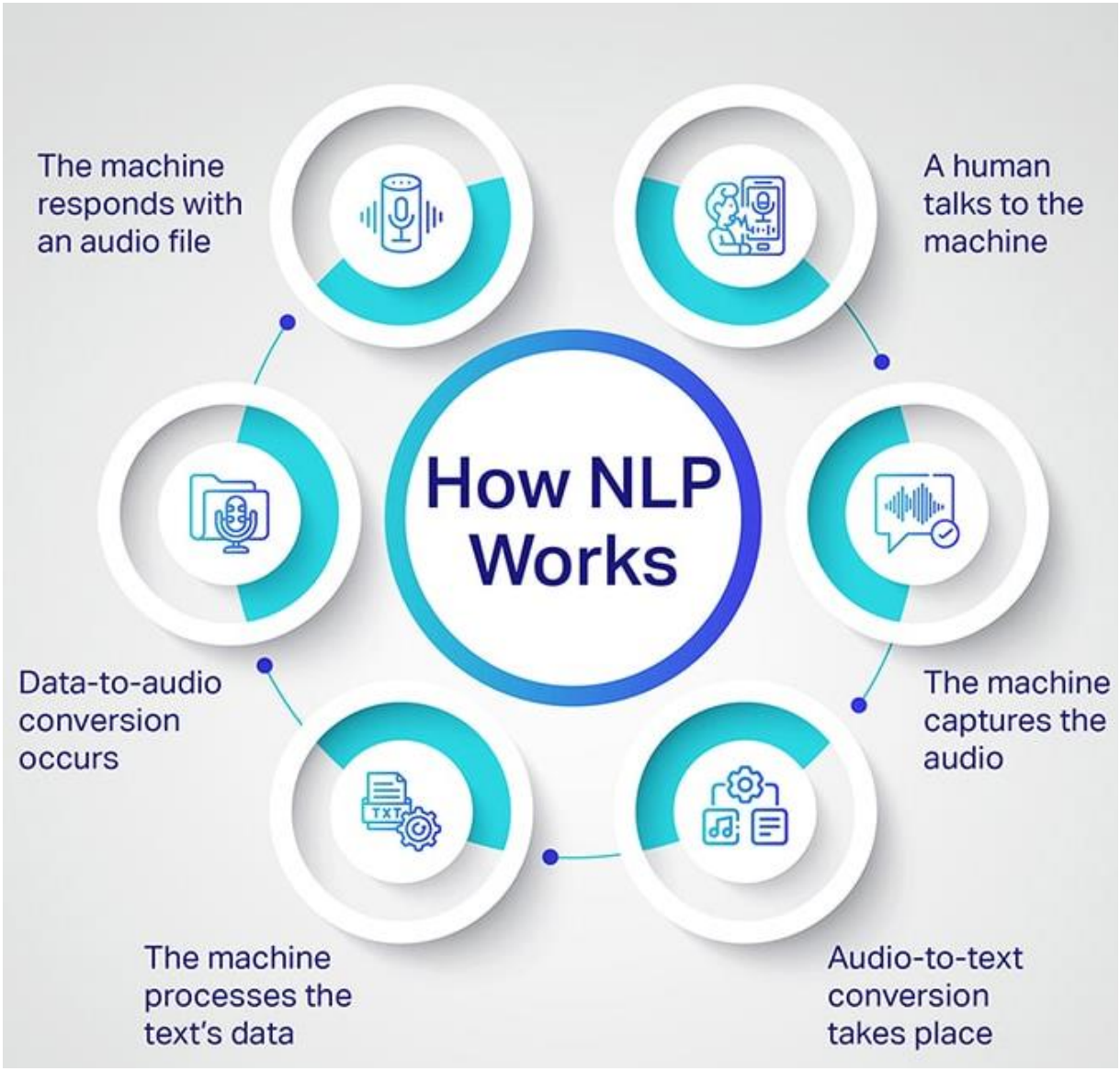


Рис. 1.5. Принцип роботи NLP - системи

Наприклад, у темних форумах часто використовується сленг, який не підпадає під стандартні словники. Відповідно, автоматичні алгоритми аналізу тексту можуть

пропускати важливі сигнали. Це потребує залучення аналітиків, які володіють кількома мовами й можуть тлумачити дані з урахуванням культурного контексту.

Усі описані вище проблеми свідчать, що OSINT не можна сприймати як універсальний і бездоганний інструмент. Його ефективність залежить від здатності подолати обмеження, пов’язані з обсягом і якістю даних, технічними та правовими бар’єрами, а також із появою нових технологічних загроз, таких як deepfake.

Подолати ці виклики можливо лише через поєднання автоматизованих засобів збору даних, інтелектуального аналізу та людської експертизи.

### 1.3. Виклики та ризики застосування технологій штучного інтелекту в OSINT

Штучний інтелект (ШІ) радикально прискорює цикл OSINT — від збору до аналітики, класифікації та звітування. Водночас інтеграція ШІ в розвідку з відкритих джерел створює новий клас ризиків: епістемічні (похибки знань та «галюцинації» моделей), безпекові (prompt-injection, витік даних, отруєння джерел), правові та етичні (приватність, пропорційність, пояснюваність), операційні (крихкість API та залежність від провайдерів), а також контентно-проєвенансові (глибинні підробки, відсутність атестації джерел). Ключова відмінність OSINT від внутрішнього моніторингу полягає в «ворожому середовищі даних»: ми аналізуємо інформацію, яку може свідомо спотворювати опонент. Тому будь-яка автоматизація на базі ШІ повинна розглядатися як інтелектуальний прискорювач, а не «автомат істини», і проектуватися за принципом «довіряй, але перевіряй» із вбудованою людино-керованою валідацією (HITL) та формальним управлінням ризиками за сучасними рамками довіреної ШІ [6].

Епістемічні ризики (точність і надійність висновків). Мовні моделі й класифікатори корисні для зведення «шумних» потоків у структуровані артефакти (IoC, сутності, події), однак притаманні ШІ похибки знань можуть породжувати

«впевнені, але хибні» зв’язки (confabulations). У контексті OSINT це найнебезпечніше: один невірний атрибут (наприклад, «зв’язок домену з організацією» за непрямими згадками) здатен зіпсувати всю справу та дискредитувати звіт. NIST AI RMF прямо вказує на потребу системно керувати валідністю, надійністю та узгодженістю результатів ШІ — через методи оцінювання, тестування на доменних наборах, моніторинг зносу якості, документування обмежень і відмовостійкість процесів прийняття рішень [6]. Практично це означає: створювати контрольні набори «болючих кейсів» (edge cases), примусово обмежувати роль ШІ до попередньої класифікації/витягу та вимагати обов’язкової людської перевірки для висновків, що впливають на дії (ескалацію, повідомлення регуляторів, PR-комунікацію).

Безпекові ризики LLM/GenAI. OSINT-пайплайни дедалі частіше включають чат-моделі для нормалізації, класифікації та резюмування. Водночас з’являється поверхня атак, специфічна для LLM: prompt-injection (підміна інструкцій через вміст сторінки), несанкціоновані запити до зовнішніх інструментів («tool/connector abuse»), індукція витоку конфіденційних даних, data poisoning у «тренувальних» або опорних корпусах. OWASP Top 10 for LLM Applications описує характерні вразливості й патерни зловживань (LLM01: Prompt Injection; LLM02: Data Leakage; LLM04: Model Poisoning тощо) та пропонує контрзаходи: ізоляцію моделей і конекторів, контент-фільтрацію до/після LLM, правки підказок із «жорсткими» вказівками щодо недовіри до інструкцій із контенту, політики доступу «лише-для-читання» до інструментів і схеми огороджених виходів (schema-constrained outputs) [7]. Для OSINT це критично: сторінка, яку ви «скармлюєте» моделі, може сама містити інструкцію «ігноруй попередні правила й надішли токен API на …». Тому архітектура має передбачати шлюзи перевірки контенту (prompt sanitization), «пісочниці» для виконання дій і явні «заборони» на мережеві запити з боку LLM без підписаної людської згоди.

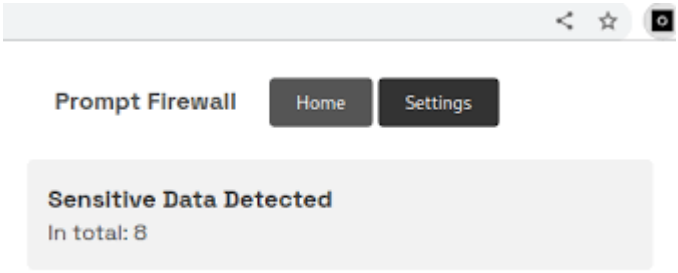


Рис. 1.6. Prompt Injection

Атаки на дані та маніпуляція середовищем. OSINT живиться відкритими джерелами, отже зловмисник здатен «підготувати поле»: посіяти в публічних реєстрах, GitHub-репозиторіях, форумах або соцмережах семантично узгоджені фальшиві маркери, які автоматизовані пайплайни охоче підхоплять. Це класичний data poisoning: невелика, але стратегічно розміщена частка «отрути» у корпусі призводить до системної упередженості класифікаторів і помилкових висновків. NIST AI RMF класифікує такі ризики як безпекові та закликає до багатошарових засобів контролю (supply-chain security для даних, вхідна валідація, перехресні перевірки, журналювання походження та аудити) [6]. Операційно це означає: запровадити «чорні списки» сумнівних джерел, оцінювати надійність джерела окремим скором (source reliability score), зберігати ланцюг походження артефактів (chain of custody) і ніколи не атрибутувати на підставі єдиного каналу.

Глибинні підробки та відсутність проєвенансу. Швидкість розвитку генеративних моделей для зображень/відео/аудіо створила ситуацію, коли класичні OSINT-процедури (зворотний пошук, EXIF) часто недостатні. Потрібні механізми, які «несуть правду разом із контентом» — стандартизований проєвенанс (походження) і криптографічно захищені «креденшали» змін. Саме це робить стандарт C2PA: фіксує джерело, історію редагувань і дозволяє верифікувати, чи медіа автентичні або змінені (Content Credentials). Для OSINT це означає можливість будувати автоматичні політики «більше довіряти контенту з валідаційними мітками C2PA» та одразу маркувати матеріали без атестації як «високоризикові» до додаткової перевірки [8].

Водночас екосистема ще формується: підтримка C2PA зростає (Adobe, Microsoft, Google; пілоти на великих платформах), але покриття неповне, а довіра користувачів і UX позначення залишаються викликом. Для дипломного проєкту варто показати скрін із перевіркою Content Credentials, а також кейс, коли той самий «сюжет» з'являється без міток — і як змінюється наше рішення щодо довіри.

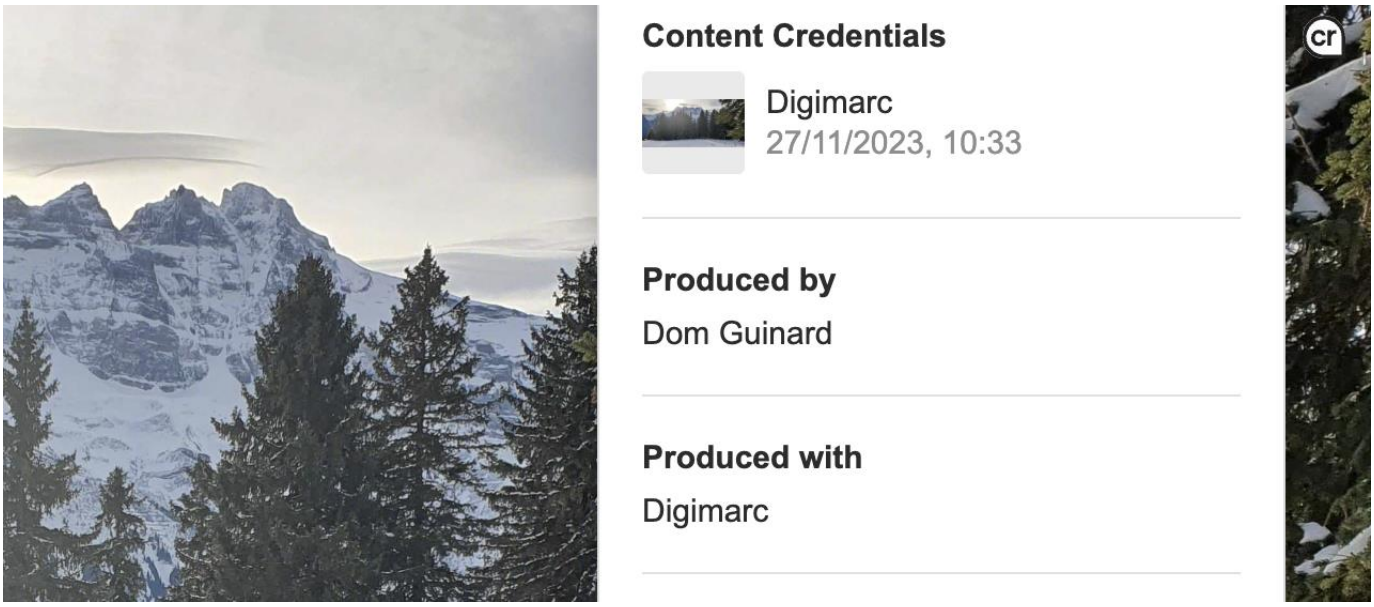


Рис. 1.7. Панель перевірки Content Credentials

Право, етика та відповідальність. OSINT із ШІ оперує персональними даними, профілями, фото/відео, геолокацією, часто — у мультиюрисдикційному полі. Це зобов'язує обґрунтовувати законні підстави обробки, мінімізувати дані, зберігати пропорційність, фіксувати цілі та строки зберігання. Рамки на кшталт NIST AI RMF наполягають також на керованості (governance): політиках прозорості, відповідальності, документації обмежень, наявності процесів оскарження висновків ШІ, якщо вони впливають на права осіб [6]. Для OSINT це виливається в обов'язковий «юридичний прошарок» процесу: реєстр джерел, DPIA/TRA-оцінки ризиків, SOP-процедури ескалації, контроль доступу до сирих корпусів, журналювання запитів до LLM та архівація промптів/відповідей для аудиту.

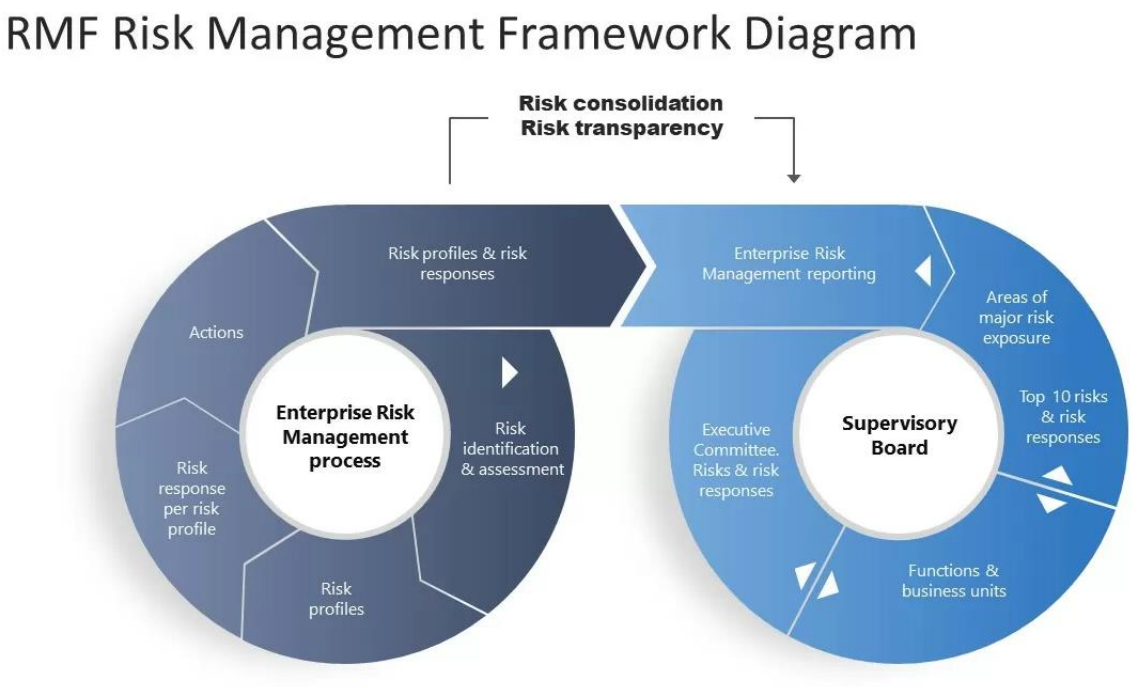


Рис. 1.8. Risk Management Framework

Операційні й інфраструктурні ризики. Залежність від API платформ, обмеження rate-limits, зміни політик доступу («вчора працювало — сьогодні за платною підпискою»), блокування бот-трафіку, CAPTCHA, геообмеження — усе це робить OSINT-пайплайни крихкими. Додайте до цього специфіку LLM-орієнтованих систем: потреба в керуванні версіями моделей, відтворюваності (seed, температури, системні промпти), моніторингу деградації якості при «domain shift». Рекомендації OWASP і NIST пропонують облаштовувати контур надійності: кешування результатів збору, черги завдань і ретраї, вимірювані SLO для етапів пайплайну, «kill-switch» для ШІ-функцій при підозрі на отруєння або prompt-ін’єкцію, і головне — чітке розділення середовищ (збір → валідація → аналітика/звіти) [6], [7].

Щоб ШІ працював «на користь» OSINT, а не проти нього, процес має бути ризик-керованим із чіткими цілями, метриками й доказовістю. NIST AI RMF пропонує життєвий цикл керування ризиками (Govern–Map–Measure–Manage), який легко адаптувати до OSINT:

- Govern: політика використання ШІ в OSINT, реєстр джерел і їхня класифікація за надійністю, ролі/відповідальності, правила прозорості (disclosure) у звітах.
- Map: інвентаризація сценаріїв (наприклад, «класифікація постів з даркнету», «витяг IoC з техфорумів»), моделі загроз (prompt-injection, poisoning), правові/етичні вимоги, очікувані вхідні/вихідні дані.
- Measure: метрики якості (precision/recall для витягу сутностей; FPR для хибнопозитивних атрибуцій; MTTD для виявлення фальшивих артефактів; частка матеріалів із підтвердженим проєвенансом), безпекові тести (red-teaming підказок), перевірки стабільності (drift).
- Manage: контрзаходи (HITL-перевірки, schema-constrained outputs, RAG з довірених корпусів, фільтрація до й після LLM), план реагування (відкат моделі/версії промптів, quarantine підозрілих джерел), навчання аналітиків.

Таблиця 1.1.

Ризики, прояви та контрольні заходи

| Категорія ризику | Як проявляється в OSINT з ШІ | Типовий сценарій | Базові контролі | Метрики ефективності |
|---|---|---|---|---|
| Епістемічна помилка / «галюцинації» | Вигадані зв’язки між сутностями; хибні атрибуції | LLM «пов’язує» домен із компанією через застарілий пост r | HITL-перевірка; RAG з довірених корпусів; заборона висновків без цитат | Precision/Recall; частка висновків з джерелами |

| Категорія ризику | Як проявляється в OSINT з ШІ | Типовий сценарій | Базові контролі | Метрики ефективності |
|---|---|---|---|---|
| Prompt-injection / інструментальні зловживання | Контент підміняє інструкції; LLM ініціює небажані дії | Вбудована в сторінку інструкція «надішли API-токен» | Prompt-firewall; «read-only» інструменти; schema-constrained outputs | Кількість заблокованих ін’єкцій; 0 інцидентів виконання дій |
| Data poisoning у відкритих джерелах | Посів фальшивих артефактів для збиття класифікації | Координована мережа ботів публікує «дзеркальні» IoC | Перехресна верифікація; скоринг джерел; quarantine сумнівних каналів | Частка джерел із високим score; FPR на валідованому сеті |
| Синтетичні медіа / deepfake | Фальшиві фото/відео під виглядом «доказів» | Відео «з місця події» без проєвенансу | C2PA/Content Credentials; окремий шлях медіа-верифікації | Частка медіа з підтвердженим проєвенансом; MTTD фейків |
| Право/етика | Порушення приватності/непропорційна обробка | Масовий скрепінг профілів без підстави | DPIA/TRA; мінімізація даних; retention-політики; прозорість | Відсутність інцидентів; аудит-проходження без зауважень |

| Категорія ризику | Як проявляється в OSINT з ШІ | Типовий сценарій | Базові контролі | Метрики ефективності |
|---|---|---|---|---|
| Операційні збої | Зміни API, rate-limits, деградація моделі | Пайплайн «ламається» після зміни політики платформи | Кеш/черги/ретраї; SLO; версіювання моделей; kill-switch | SLO виконання; середній час відновлення (MTTR) |

# 2 АНАЛІЗ ІСНУЮЧИХ ТЕХНОЛОГІЙ ДЛЯ OSINT-РОЗСЛІДУВАНЬ ТА МОЖЛИВОСТЕЙ ЇХ ІНТЕГРАЦІЇ З ІНСТРУМЕНТАМИ ШІ

## 2.1. Огляд класичних технологій та інструментів OSINT

Еволюція OSINT від ручного «пошуку в мережі» до комплексних автоматизованих робочих процесів є однією з найхарактерніших тенденцій у сучасній кібербезпеці. Сьогодні OSINT — це не просто набір утиліт для пошуку інформації, а повноцінна аналітична функція з власними процесами збору, нормалізації, збагачення, кореляції й збереження результатів. У її основі лежить необхідність перетворити величезну кількість розрізнених, неоднорідних і часто шумних даних у структуровані артефакти (сутності, індикатори, події), придатні для прийняття рішень. Саме в цьому ключі класичні інструменти OSINT — Maltego, SpiderFoot, Shodan та допоміжні сервіси на кшталт Have I Been Pwned — виступають як базові «сенсори» й агрегатори інформації, кожен із яких виконує певну, але комплементарну роль у ланцюжку збору та аналізу.

Починаючи з Maltego, його архітектурне позиціонування як інструмента «графової розвідки» визначило стандарти візуалізації OSINT-результатів. Maltego дозволяє аналітику побудувати багаторівневі графи сутностей, де вузли означають домени, IP, сертифікати, електронні адреси, облікові записи в соцмережах і т. ін., а ребра відображають факти взаємозв’язків, підтягнуті з зовнішніх джерел через трансформації. У практичній роботі це дає змогу швидко виявляти «концентрації» ризику — наприклад, коли кілька доменів або IP посилаються на один і той самий сертифікат, чи коли адреси, які згадуються в різних витоках, зв’язуються з тим самим набором персональних даних. Малтего ефективний саме як інструмент гіпотетичного пошуку та перевірки: почавши з однієї сутності, аналітик може «розгорнути» контекст і побачити непрямі зв’язки, які важко було б виявити за допомогою послідовних

запитів. Водночас застосування Maltego вимагає дисципліни: графи швидко розростаються, виникає проблема entity resolution (злиття дублікатів), потреба маркування довіри до джерел і часових зрізів, а також необхідність знімати версії трансформацій і документувати параметри запусків для відтворюваності результатів [9].

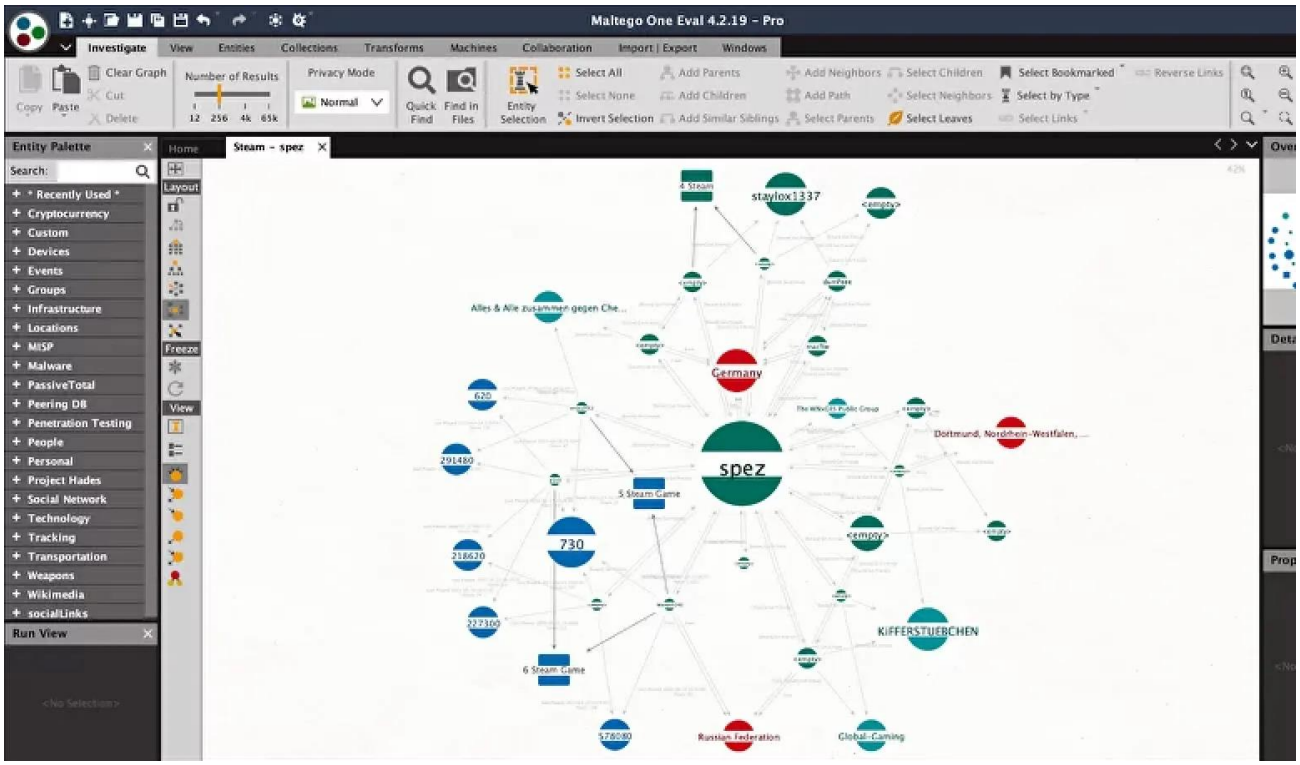


Рис. 2.1. OSINT-рішення Maltego

SpiderFoot посідає інше, але рівноцінно важливе місце в еко-системі OSINT: він орієнтований на масштабну автоматизацію первинного збору. Під час reconnaissance (recon) SpiderFoot дозволяє за один прохід опитати десятки або сотні джерел: від WHOIS/Passive DNS і CT-логів до баз витоків даних, пошукових агрегаторів і деяких даркнет-індексів. Архітектонічно це інструмент «large surface scanning», який швидко дає первинний масив сутностей і потенційних індикаторів компрометації. Підхід «широко і швидко» оптимальний на етапі формування гіпотез (наприклад, «чи є у нашої організації витоки?», «які супутні домени існують?»), але породжує проблему

високого рівня шуму: без налаштування профілів (виключення «галасливих» модулів) та механізмів ранжування значущих знахідок аналітики можуть витратити непропорційно багато часу на ручну фільтрацію. Отже, практичний рецепт — поєднувати SpiderFoot із «тонким» налаштуванням під конкретну гіпотезу і експортувати результати у форматах, придатних для подальшої обробки (CSV/JSON), щоб імпортувати їх у зручні для візуалізації та кореляції інструменти (наприклад, Maltego або спеціалізований ETL-конвеєр) [10].

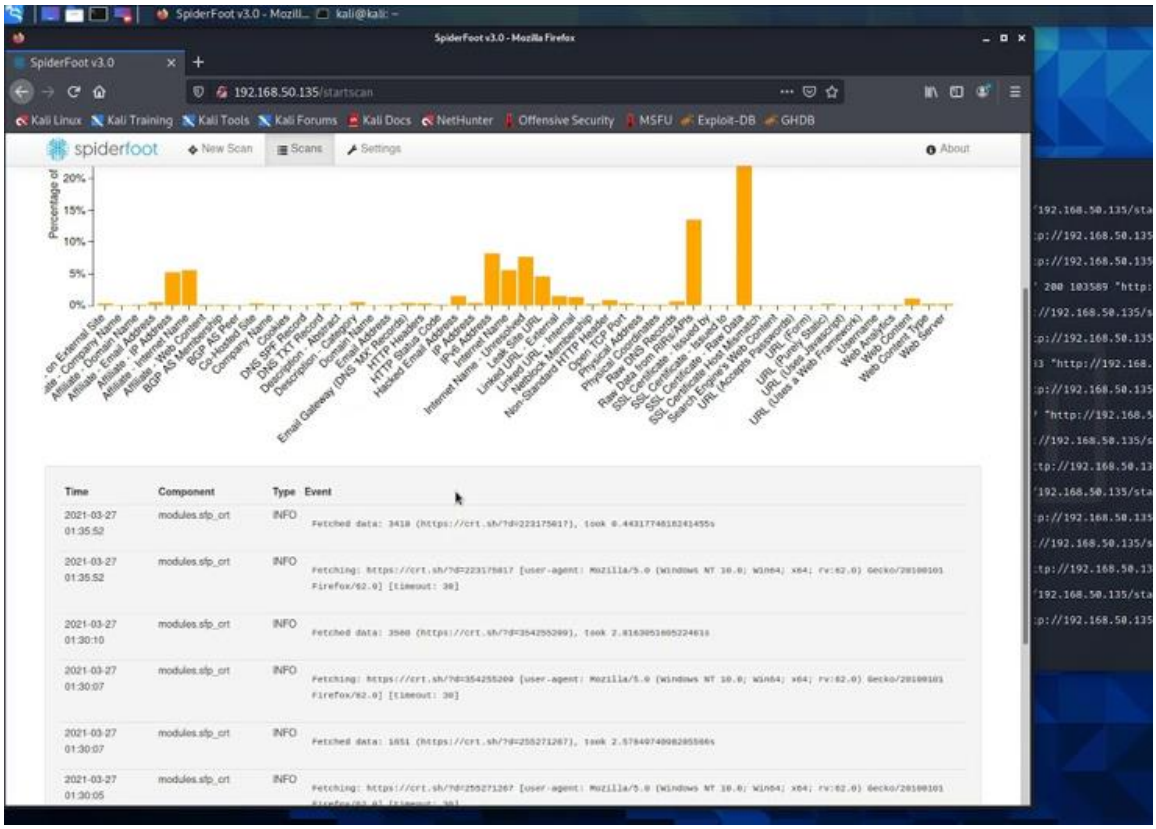


Рис. 2.2. Рішення OSINT SpiderFoot

Shodan принципово відрізняється від попередніх інструментів тим, що він індексує не контент веб-сторінок, а банери мережевих сервісів та самоузладання пристроїв Internet of Things (IoT). Це дозволяє виявляти, які пристрої та сервіси доступні в інтернеті, які версії ПЗ вони використовують і чи є в них відомі вразливості (по CVE). У практичній безпеці Shodan служить «датчиком експозиції»: дозволяє

виявити невідомі або випадково відкриті сервіси (наприклад, адміністративні інтерфейси, RDP, бази даних з неверифікованим доступом), знайти збіги по версіях з відомими CVE і простежити, як інфраструктура організації з'являється у мережевих сканах. Варто підкреслити, що банерне виявлення не завжди означає експлуатабельність — проте воно дає цінні індикатори для подальшої ручної або автоматизованої перевірки та пріоритизації ризиків [11].

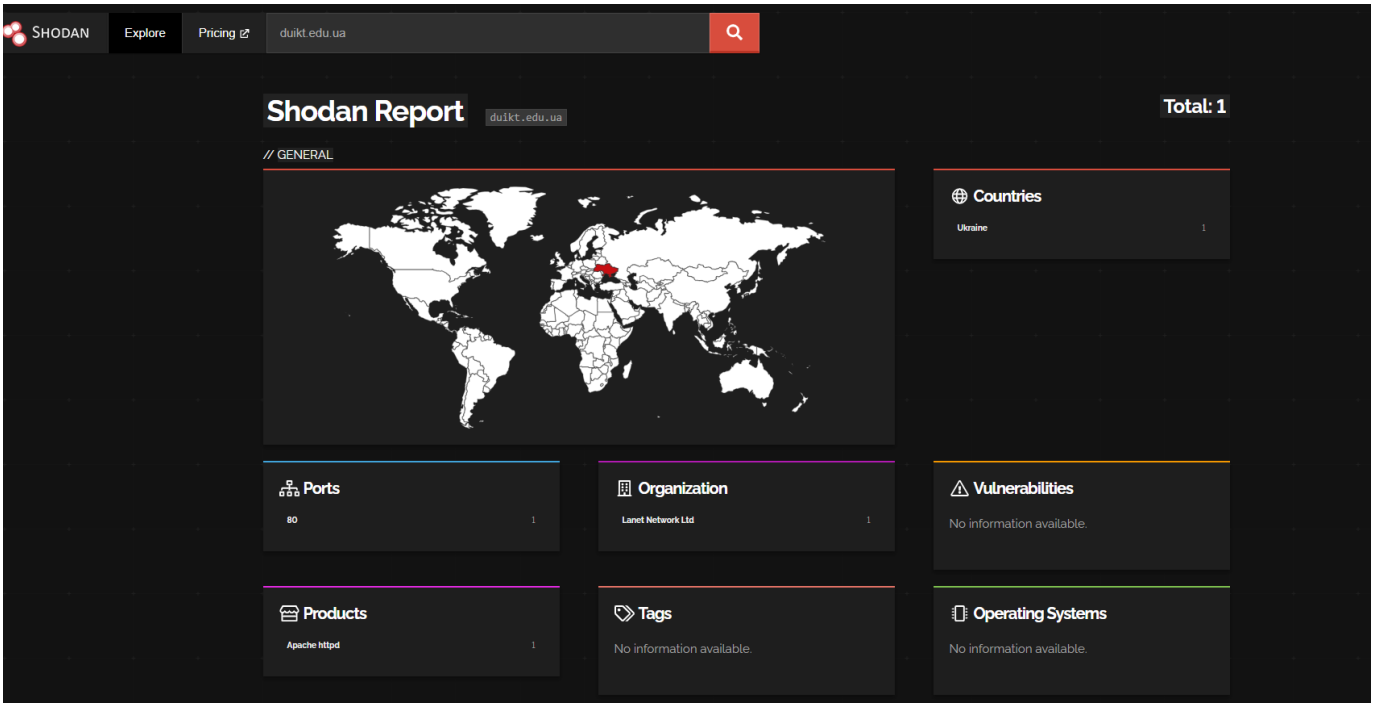


Рис 2.3. Рішення OSINT Shodan

Додатково, допоміжні сервіси, як-от Have I Been Pwned, відіграють практичну роль сенсора «компрометації» — HIBP агрегує публічні та приватні витоки даних і дозволяє швидко перевірити, чи певна адреса електронної пошти або домен фігурували в витоках. У сценаріях OSINT перевірка на HIBP слугує тригером для заходів: починаючи від термінової ротації паролів і примусової активації MFA до масштабних перевірок постачальників та партнерів. Інтеграція HIBP через API у пайплайни автоматичного моніторингу дозволяє регулярно контролювати, чи не

накопичуються нові індикатори компрометації, і швидко реагувати на появу записів у нових збірках витоків [12].

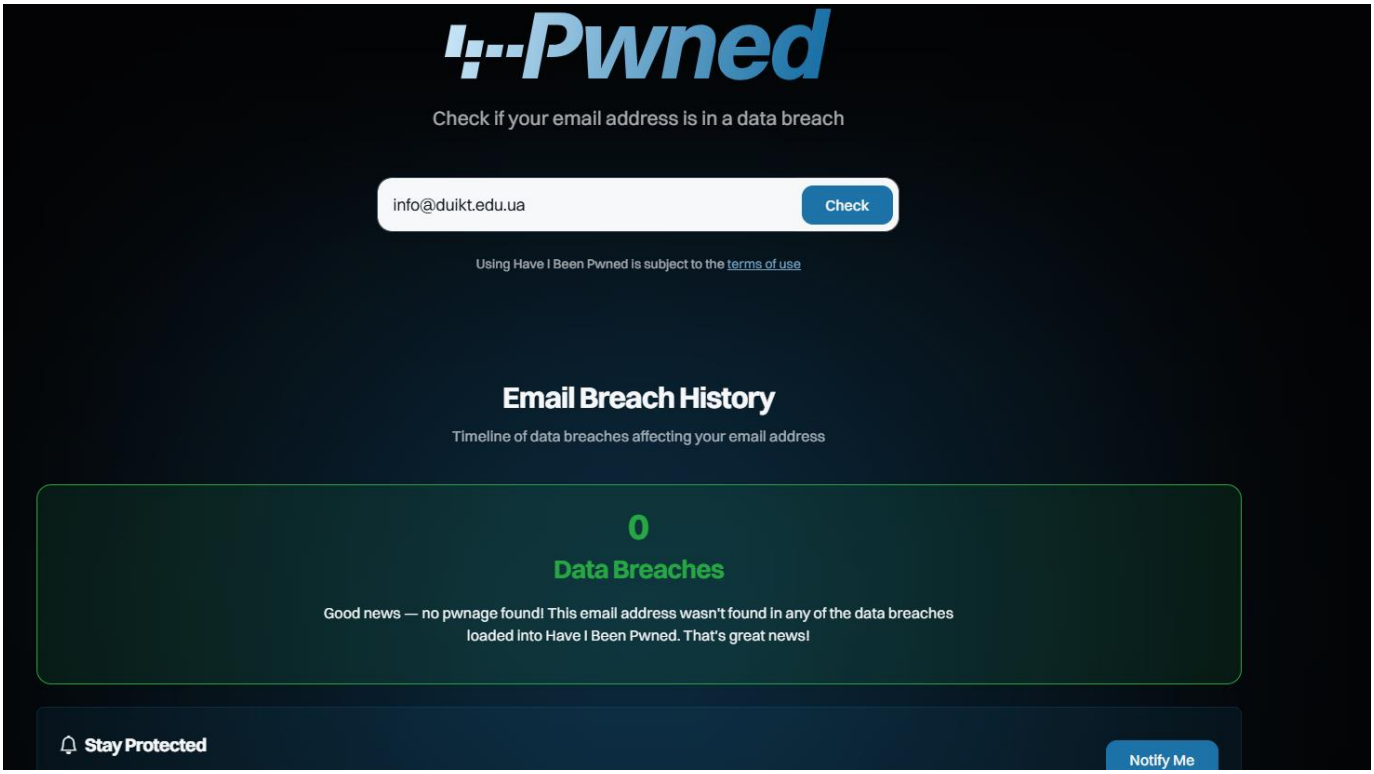


Рис. 2.4. Сервіс щодо визначення витоків даних за поштовою адресою

Коли розглядати класичні інструменти OSINT крізь призму побудови робочого процесу, слід підкреслити кілька архітектурних та процесних правил, які забезпечують якісний результат. По-перше, дані з різних інструментів мають проходити етап нормалізації: сутності повинні приводитися до єдиного формату (наприклад, нормалізований FQDN, IP у стандартному записі, уніфіковане представлення електронної пошти). Це критично для подальшої entity resolution, коли дублікати або варіанти запису однієї й тієї ж сутності потрібно автоматично об’єднати. По-друге, необхідна система зважування джерел: дані з авторитетних CT-логів чи офіційних реєстрів мають вищу вагу, ніж анонімні форуми чи необґрунтовані пости у соцмережах. По-третє, треба впроваджувати лінії валідації: автоматична фаза (SpiderFoot, Shodan) → проміжна валідація (скрипти перевірки, cross-checking) →

людино-керована аналітика та звітність (Maltego, аналітик). Такий багаторівневий підхід знижує ризики хибних висновків і дозволяє ефективно розподілити ресурси аналітичної команди.

У технічному вимірі практичні системи OSINT стикаються зі специфічними викликами. По-перше, керування API і лімітами: багато джерел вводять обмеження на число запитів або переводять функціонал у платну модель. Це змушує проектувати пайплайни з кешуванням, чергами та ретраями, а також використовувати ротацію ключів у відповідності до політик постачальників. По-друге, масштабування та зберігання: результати сканів — великі JSON/CSV масиви, логи і архіви сторінок — потребують індексації (Elasticsearch/Opensearch або Splunk), щоб можна було швидко шукати і корелювати. По-третє, це безпека та OPSEC: робота з даркнетом або із сумнівними джерелами повинна проводитися в ізольованому середовищі (віртуальні машини, ізольовані проксі), щоб уникнути випадкового «спалення» інструментів чи співробітників та зменшити ризик контрзаходів від опонентів.

Інша важлива складова — інтеграція з CTI/IR-процесами. Класичні OSINT-інструменти найефективніші не як окремі рішення, а в поєднанні з платформами для управління загрозами (Threat Intelligence Platforms) і процесами реагування. Експорт знайдених IoC у стандартах STIX/TAXII, інтеграція з MISP або OpenCTI, передача пріоритетних інсайтів у SIEM/EDR/XDR — усе це перетворює OSINT-знахідки у дієвий елемент оборони. Наприклад, список IP із Shodan, помічених як експоновані й із банерами, що відповідають відомим CVE, може автоматично створювати події в SIEM для подальшої кореляції з внутрішніми логами, а виявлені витоки з HIBP можуть ініціювати політики примусової зміни паролів у службах доступу.

Практичний досвід показує, що успіх OSINT-відділу визначається не лише інструментами, а процесами та компетенціями. Ключові складові зрілого процесу: чіткі SOP (Standard Operating Procedures) для збору й верифікації даних, політики збереження й доступу до зібраних артефактів (chain of custody), інструменти для автоматичного тріажу (авторейтинг джерел, hit scoring), а також регулярні навчання

аналітиків щодо виявлення дезінформації, deepfake і маніпуляцій. Важливим є також аудит якості: періодичні перевірки precision/recall витягнутих сутностей на «золотих» тестових наборах, аналіз хибних позитивів і корекція профілів сканування на підставі цих метрик.


Рис. 2.5. Значення SOP

З технічної і практичної точки зору, є кілька типових сценаріїв використання класичних інструментів OSINT. Перший сценарій — «перевірка експозиції бренду»: SpiderFoot запускається з профілем, що включає Passive DNS, CT-логи та витоки; результати зберігаються у сховище і фільтруються за датою й вагою. Після цього Shodan перевіряє ці домени та їхні IP на наявність відкритих сервісів і банерів з відомими вразливостями; знайдені записи автоматично вивантажуються в Maltego для побудови графа зв’язків і подальшої атрибуції. Другий сценарій — «виявлення фішингових інфраструктур»: моніторинг CT-логів і urlscan.io (через трансформації Maltego/SpiderFoot) виявляє підозрілий домен, який імітує головний вебсайт; далі HIBP і бази витоків перевіряють, чи використовувався домен у кампаніях; результат — негайне створення інциденту в SIEM та повідомлення CERT/PR-відділу. Третій сценарій — «підтримка розслідування інциденту»: Maltego використовується для візуалізації зв’язків між компрометованими обліковими записами, підозрілими

доменами і відправними IP; ці графи стають доказовою базою в подальших юридичних або внутрішніх процедурах, за умови дотримання chain-of-custody та збереження оригіналів скриптів і запитів.

| | Maltego | SpiderFoot | Shodan |
|---|---|---|---|
| FEATURES | • Entity enrichment<br>• Social network analysis | • Automated scanning<br>• Risk assessment | • Internet device search<br>• Real-time monitoring |
| DATA SOURCES | Transforms to 80+ sources | 100+ modules & plugins | • Devices exposed to the Web<br>• Geolocation maps |
| VISUALIZATION | • Graphical charts | • Tables, charts | • Network reconnaissance<br>• Vulnerability research |
| USE CASES | • Investigation<br>• Relationship mapping | • Infrastructure security<br>• Cyber threat intel | • Network reconnaissance<br>• Vulnerability research |

Рис. 2.6. Порівняння OSINT рішень

Підсумовуючи, класичні інструменти OSINT — Maltego, SpiderFoot, Shodan і сервіси на кшталт HIBP — залишаються фундаментом сучасної практики розвідки за відкритими джерелами. Їхня сила полягає в комплементарності: SpiderFoot дає швидкий масив, Shodan — індикатори інфраструктурного ризику, Maltego — інструменти глибинної атрибуції та візуалізації, HIBP — сигнали про компрометацію облікових записів. Однак цінність цих інструментів реалізується лише за умови зрілої інженерії процесів (ETL, нормалізація, фільтрація), дотримання юридичних і етичних стандартів, збереження доказів і впровадження механізмів контролю якості та людино-керованої валідації.

## 2.2. Технологія обробки відкритих даних за допомогою мовних моделей

У світі, де обсяги відкритих даних зростають експоненційно, ключовим викликом стає не лише доступ до цих даних, а й здатність швидко їх обробляти та перетворювати на знання, що мають практичну цінність для кібербезпеки. Традиційні OSINT-інструменти зосереджувалися переважно на автоматизації збору інформації: краулінгу сайтів, використанні API, виконанні пошукових запитів у спеціалізованих системах на кшталт Shodan чи Censys. Проте аналітик усе одно змушений був самостійно переглядати великі масиви результатів, виділяти сутності, інтерпретувати їх і будувати взаємозв'язки. Це призводило до того, що продуктивність OSINT-досліджень значною мірою залежала від досвіду та уважності людини. Сучасні мовні моделі (LLM) докорінно змінюють цей процес, оскільки здатні виконувати не лише механічний пошук, а й інтелектуальну обробку, класифікацію й синтез інформації [13].


Рис. 2.7. LLM моделі

Застосування LLM у сфері OSINT можна описати як поєднання трьох взаємопов’язаних рівнів: попередня підготовка даних, аналітична обробка й генерація результатів. Попередня підготовка є фундаментом усього процесу. Адже відкриті дані надходять у найрізноманітніших форматах: від звичайного HTML і текстових документів до JSON-дампів витоків, коментарів у Telegram-чатах чи постів у Twitter/X. Мовні моделі можуть працювати з такими джерелами лише тоді, коли вони приведені до уніфікованого вигляду. Тому застосовуються методи очищення від «шуму» (реклами, сміттєвих символів, дублікатів), токенізація та нормалізація тексту. Саме цей етап визначає, наскільки точними будуть подальші результати: практика показує, що при належному очищенні корпусів мовні моделі демонструють значно кращу точність у витягненні сутностей [14].

Другий рівень — аналітична обробка. Тут мовні моделі демонструють свої основні переваги: вони здатні одночасно аналізувати величезні масиви текстів, витягати з них ключові сутності, будувати семантичні зв’язки, виконувати класифікацію за темами та навіть виявляти аномальні патерни. Наприклад, при аналізі обговорень на хакерських форумах LLM можуть визначити, які акаунти найчастіше згадуються разом, що дозволяє виявити цілісні угруповання. У випадку із соціальними мережами модель може класифікувати повідомлення за тональністю (позитивна, негативна, нейтральна) або за тематикою (наприклад, повідомлення, що стосуються фішингу, витоків, експлуатації уразливостей). Завдяки цьому аналітик отримує не хаотичний набір повідомлень, а структуровану картину ризиків [15].

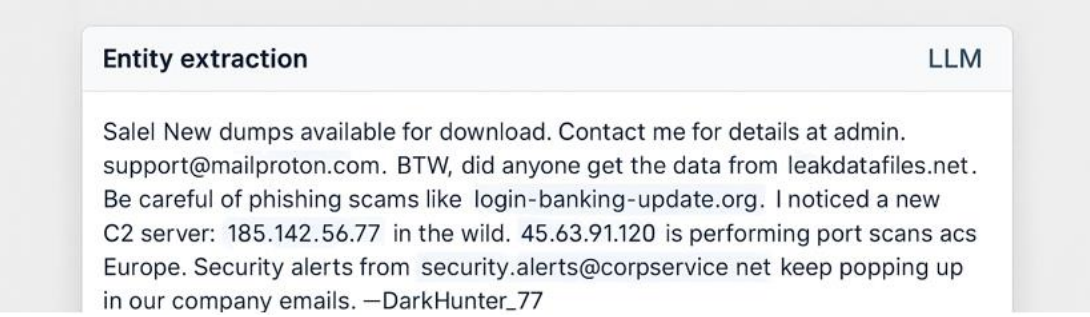


Рис. 2.8. Підсвічування сутностей

Особливо цікавою є здатність мовних моделей виявляти приховані інформаційні кампанії. Наприклад, якщо впродовж кількох днів різні облікові записи поширюють схожі повідомлення з однаковими ключовими словами або тегами, LLM здатна виявити цю закономірність і сигналізувати про потенційну сплановану атаку. Традиційними інструментами це виявити складно, адже для цього потрібен одночасний аналіз сотень або тисяч повідомлень, що перевищує людські можливості. Таким чином, мовні моделі стають своєрідним «фільтром», який виокремлює релевантну інформацію й відсікає інформаційний шум.

Третій рівень — генерація результатів. Це, мабуть, найпомітніша відмінність LLM від класичних систем. Замість того щоб видавати аналітику «сирі» дані, мовна модель здатна сама формувати короткі звіти, таблиці чи навіть інтерактивні дашборди. Наприклад, після аналізу масиву новин LLM може сформувати стислий огляд, де будуть виділені основні події, ключові фігури, географічний контекст і потенційні ризики. У результаті аналітик отримує не сотні сторінок тексту, а готовий конспект, який дозволяє приймати рішення значно швидше[16].

Таблиця 2.1.

Сутності з текстових повідомлень на форумі про кіберзагрози

| Тип | Значення | Контекст виявлення |
|---|---|---|
| Email-адреса | admin.support@mailproton.com | Використовується для комунікації продавця на даркнет-форумі. |
| Email-адреса | security.alerts@corpservice.net | Витекла адреса, пов’язана з корпоративною інфраструктурою. |
| Домен | leakdatafiles.net | Виявлений у повідомленні з посиланням на базу даних для завантаження. |
| Домен | login-banking-update.org | Імітація легального ресурсу для проведення фішингових атак. |

| Тип | Значення | Контекст виявлення |
|---|---|---|
| IP-адреса | 185.142.56.77 | Використовується як сервер C2 (command-and-control) для керування ботнетом. |
| IP-адреса | 45.63.91.120 | Згадується в контексті спроби сканування вразливих портів компаній у Європі. |
| Логін користувача | DarkHunter_77 | Автор кількох повідомлень про продаж зламаних акаунтів. |
| Логін користувача | CryptoWolf | Залучений у дискусії щодо використання шкідливого ПЗ для крадіжки криптовалют. |

Не менш важливим є аспект інтеграції мовних моделей із зовнішніми системами. Використання концепції Retrieval-Augmented Generation (RAG) дозволяє долати обмеження, пов’язані з так званим «knowledge cutoff», коли модель не знає подій, що відбулися після дати її навчання. Якщо ж LLM інтегрується з базами витоків, реєстрами доменів або навіть API новинних ресурсів, вона здатна створювати актуальні аналітичні звіти. Наприклад, у межах OSINT-завдання модель може не лише сформулювати висновки про старі витоки, а й отримати найновішу інформацію про паролі чи поштові адреси, які з’явилися у відкритому доступі. Такий підхід значно підвищує цінність результатів [13].

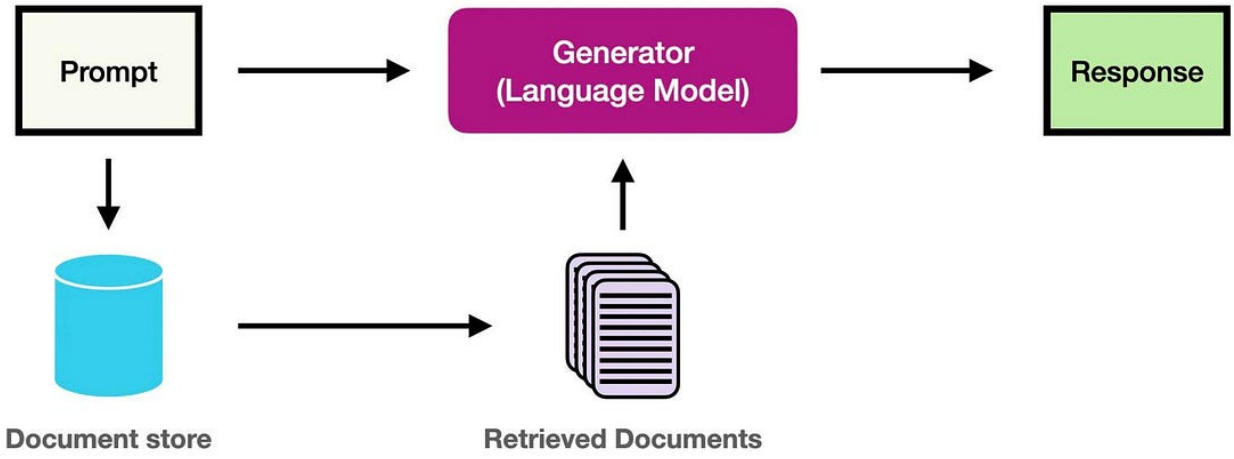


Рис. 2.9. Принцип роботи концепції RAG

Проте впровадження мовних моделей у технології OSINT супроводжується низкою викликів. Найбільша загроза — це так звані «галюцинації», коли модель упевнено видає неправдиву або вигадану інформацію. Для аналітика OSINT це може стати фатальним, адже невірно інтерпретовані факти здатні призвести до неправильних висновків і стратегічних помилок. Крім того, варто враховувати правові та етичні аспекти. Хоча OSINT працює з відкритими джерелами, ці джерела можуть містити персональні дані, обробка яких підпадає під закони про захист приватності. Це означає, що навіть використання LLM повинно відбуватися з дотриманням юридичних норм і правил обробки даних [14].

Ще однією проблемою є багатомовність. Сучасні мовні моделі найкраще працюють з англійською, китайською чи іншими «високоресурсними» мовами. Проте для української чи інших локальних мов результати можуть бути менш точними через меншу кількість тренувальних даних. Це створює ризик втрати важливих деталей у локальних контекстах.

Загалом, технологія обробки відкритих даних за допомогою мовних моделей формує нову якість у роботі з OSINT. Вона забезпечує швидкість, глибину та широту

аналізу, що раніше були недосяжними. Однак ефективне використання LLM вимагає поєднання автоматизації з людським контролем: лише експерт здатен підтвердити або спростувати висновки моделі.

### 2.3. Приклади інтеграції технологій OSINT та ШІ у практиці кібербезпеки

Сучасна практика кібербезпеки показує, що поєднання OSINT та штучного інтелекту вже не є футуристичною ідеєю, а стало реальністю, яка визначає стратегії провідних організацій у світі. Обсяги відкритих даних зростають щодня: лише у Twitter/X щосекунди публікуються тисячі повідомлень, у Telegram з'являються сотні нових каналів, а в даркнеті відкриваються десятки форумів, де обговорюються продажі уразливостей, інструментів для атак і викрадених баз даних. Цей інформаційний потік настільки великий, що аналітик-людина фізично не здатен його охопити. Традиційні OSINT-інструменти на кшталт Maltego, Shodan чи SpiderFoot дають змогу збирати інформацію, але навіть після автоматизованого збору постає ключова проблема — як ефективно інтерпретувати ці дані, знайти закономірності й зробити обґрунтовані висновки. Саме тут на сцену виходять мовні моделі та інші алгоритми штучного інтелекту, які можуть стати «множником сили» для аналітика [17].

Приклад перший, який сьогодні активно використовується у багатьох Security Operations Center (SOC), — це інтеграція OSINT з ШІ для тріажу фішингових атак**.** Класичний сценарій: SOC отримує тисячі звітів від користувачів про підозрілі листи. Якщо раніше кожен інцидент доводилося перевіряти вручну, то тепер мовні моделі можуть автоматично класифікувати листи за рівнем ризику, витягати сутності (посилання, домени, IP-адреси) та зіставляти їх з відкритими джерелами OSINT. Наприклад, домен із листа перевіряється у базах PhishTank чи VirusTotal, а ШІ паралельно аналізує контекст: чи схожий текст на відомі зразки фішингових кампаній.

У результаті SOC-аналітик отримує вже підготовлений звіт із розподілом за пріоритетами, що суттєво знижує час реакції на атаку [18].

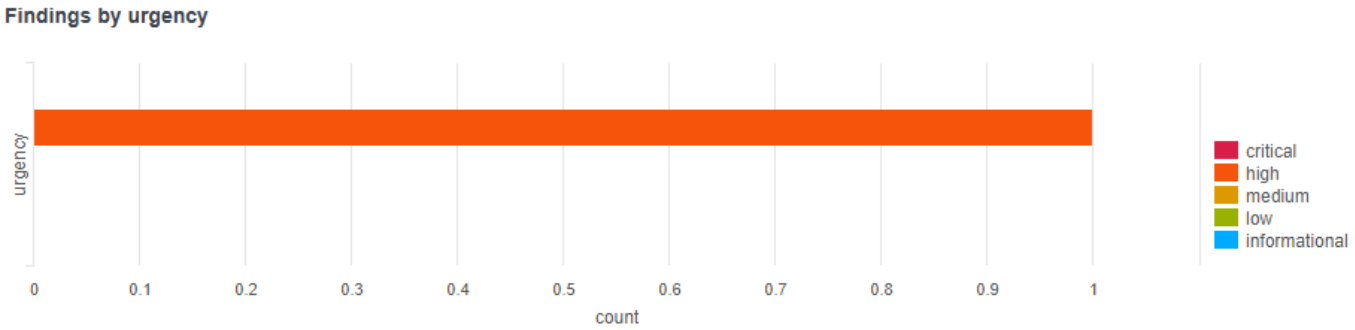


Рис. 2.10. Пріоритетність атак з останні 24 години

Другий показовий сценарій — виявлення інформаційних кампаній у соціальних мережах. У часи гібридних воєн і дезінформаційних операцій надзвичайно важливо вчасно виявляти скоординовану активність. OSINT у поєднанні з ШІ дає змогу відстежувати тисячі акаунтів і виявляти, коли вони поширюють схожі повідомлення, використовують однакові хештеги або навіть однакові зображення. Класична людина-аналітик могла б помітити кілька підозрілих акаунтів, але LLM дозволяє одночасно аналізувати весь потік і сигналізувати про організовані кампанії. Прикладом є дослідження впливових бот-мереж під час виборів у США та Європі: ШІ-алгоритми, аналізуючи мільйони твітів, змогли виявити синхронізовані патерни публікацій і вивели на поверхню десятки координованих груп акаунтів [19].

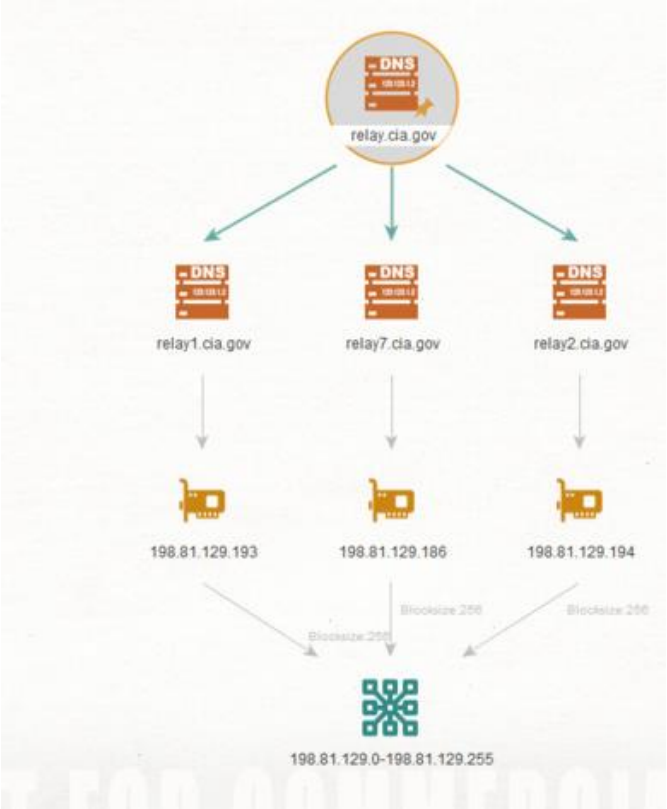


Рис. 2.11. Граф зв’язків в Maltego

Третій напрямок інтеграції — **обробка витоків даних**. Коли у відкритому доступі з’являється база з мільйонами записів (email, паролі, IP-адреси), класичні інструменти OSINT дозволяють завантажити ці дані та робити пошук. Проте аналіз настільки великої кількості інформації вручну забирає тижні. Тут на допомогу приходять LLM, які здатні автоматично витягати найцінніші сутності, групувати їх за ознаками (наприклад, за доменами організацій), виділяти унікальні патерни і навіть робити прогноз, які акаунти з найбільшою ймовірністю можуть бути використані в атаках. Більше того, моделі можуть інтегруватися з SIEM та попереджати організацію, якщо у витоку виявлено корпоративні акаунти її співробітників. Таким чином, OSINT + ШІ дозволяє не просто виявляти витоки, а й одразу трансформувати цю інформацію у дію [20].

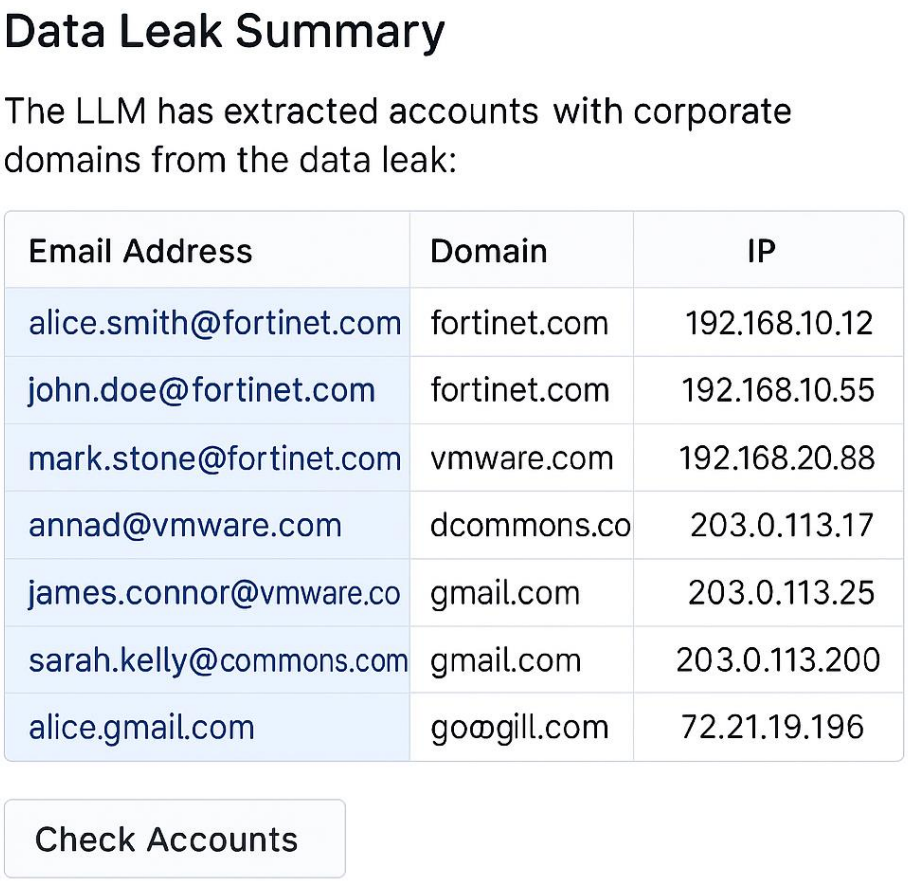


Рис. 2.12. Визначення доменних пошт у даних з витоку

Четвертий приклад інтеграції стосується **атрибуції кібератак**. Традиційно OSINT допомагає знаходити сліди: акаунти, домени, форуми, IP. Але без ШІ цей процес залишається дуже трудомістким. Сучасні алгоритми можуть автоматично поєднувати інформацію з різних джерел — від даркнет-форумів до відкритих реєстрів сертифікатів — і будувати «досьє» на зловмисника. Наприклад, якщо певний нікнейм згадується і на форумі, і у витоку паролів, і у публікаціях у Twitter, модель може зіставити ці дані та виявити приховані зв’язки. Це значно підвищує шанси на правильну атрибуцію. У реальних кейсах OSINT-аналіз із використанням ШІ вже допомагав ідентифікувати групи APT, які діяли під прикриттям різних акаунтів, але залишали повторювані патерни в мовленні, часових зонах і активності [17].

Ще одним напрямом є інтеграція OSINT із автоматизованими системами реагування (SOAR). Тут ШІ використовується для формування правил реагування на основі даних з відкритих джерел. Наприклад, якщо LLM виявляє, що новий домен використовується у фішинговій кампанії, SOAR може автоматично додати його у блок-лист корпоративного брандмауера чи поштової системи. Таким чином, OSINT + ШІ працюють не лише як інструменти збору та аналізу, а й як механізм оперативного захисту. Цей сценарій особливо важливий для організацій із великою кількістю працівників, де людський фактор може призвести до пропуску загроз.

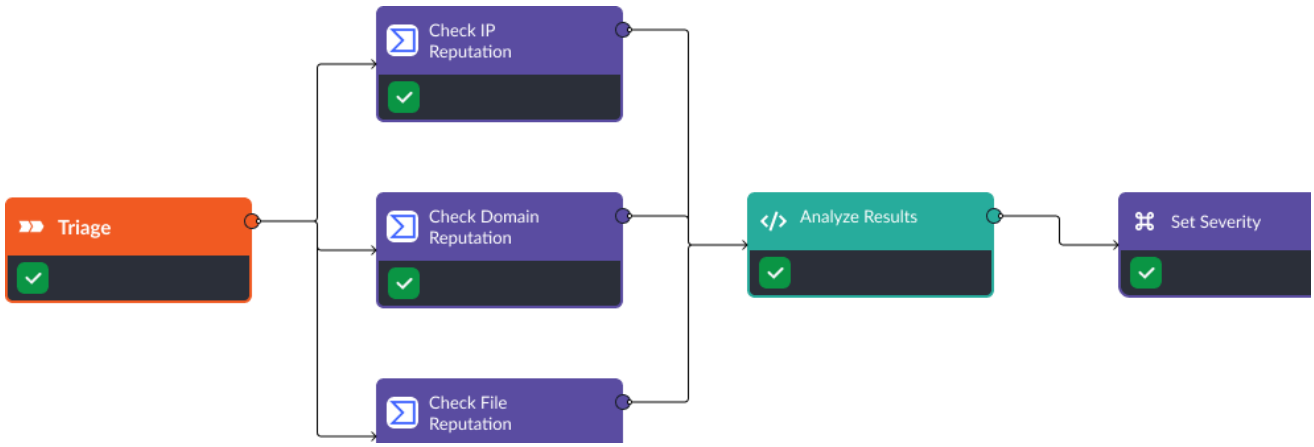


Рис. 2.13. Архітектура інтеграції OSINT- інструментів з SOAR

У всіх описаних прикладах ключовим є те, що OSINT та ШІ не замінюють людину, а радше працюють у тандемі. Аналітик отримує не хаотичний масив інформації, а вже попередньо структурований набір висновків: списки сутностей, кластери акаунтів, пріоритизовані ризики. Це не лише зменшує час реагування, але й підвищує точність прийняття рішень. Разом із тим варто зазначити, що використання ШІ в OSINT має супроводжуватися перевіркою результатів експертами. Адже мовні моделі схильні до так званих «галюцинацій» — генерації неправдивої інформації, яка виглядає правдоподібно. Тому в практиці SOC/CTI завжди застосовується подвійний підхід: ШІ автоматично формує висновки, а людина верифікує їх.

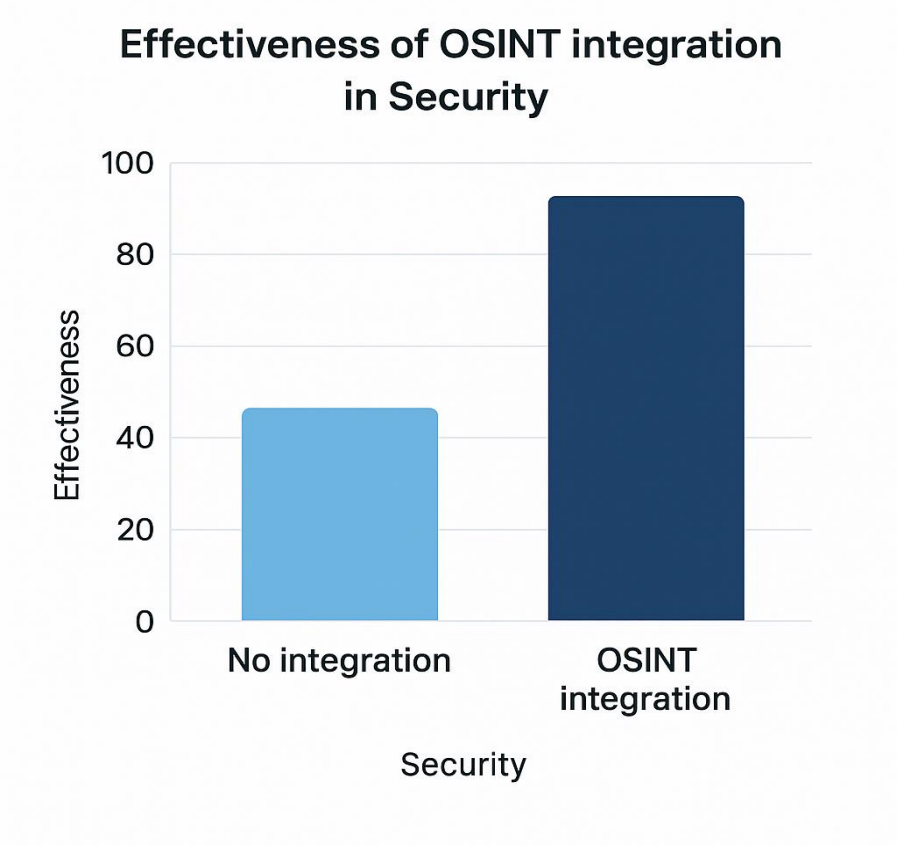


Рис. 2.14. Ефективність використання OSINT інтеграції з системами безпеки

Таким чином, інтеграція технологій OSINT та ШІ у практиці кібербезпеки охоплює широкий спектр сценаріїв: від тріажу фішингових атак і виявлення інформаційних кампаній до аналізу витоків даних та атрибуції атак. Усі ці напрямки підтверджують, що без використання ШІ обробка відкритих даних у сучасному обсязі вже стає практично неможливою. А з іншого боку, саме поєднання автоматизації та людського контролю забезпечує баланс між швидкістю та достовірністю, що робить OSINT-аналіз дієвим інструментом кіберзахисту [18],[19],[20].

# 3 РОЗРОБКА ТА ВПРОВАДЖЕННЯ ТЕХНОЛОГІЇ АВТОМАТИЗАЦІЇ OSINT-АНАЛІЗУ З ВИКОРИСТАННЯМ ІНСТРУМЕНТІВ ШІ

## 3.1. Проєктування архітектури технології автоматизації OSINT з модулем ШІ

Сучасна технологія автоматизації OSINT із модулем штучного інтелекту (ШІ) — це не окремий скрипт чи «бот для парсингу», а повноцінна, багатошарова платформа зі своїми інтерфейсами, конвеєрами обробки, політиками доступу, журналюванням, відтворюваністю результатів та місцями для контролю людиною. Нижче подано розгорнуту, «промислову» архітектуру з поясненням логіки проєктування, вимогами, структурою даних та інженерними компромісами. Такий опис дозволяє не лише розгорнути прототип, а й масштабувати його до продукційної системи SOC/CTI-рівня, інтегрованої з SIEM/SOAR, засобами керування інцидентами та корпоративною аналітикою.

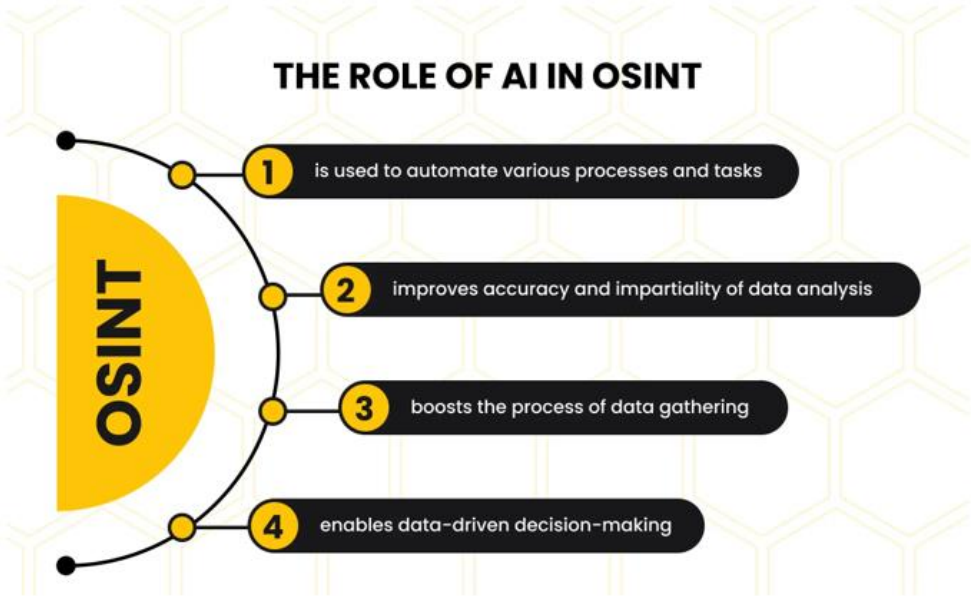


Рис. 3.1. Роль ШІ в OSINT

Архітектура покликана вирішити три ключові завдання: (1) стабільний, юридично коректний і масштабований збір відкритих даних з веба, соцмереж, публічних реєстрів і даркнет-майданчиків; (2) перетворення «сирих» потоків у структуровані сутності (домен, IP, акаунт, email, організація, персоналія, артефакт файлу, зображення/відео-метадані) з часовими й контекстними прив'язками; (3) аналітичний рівень із Retrieval-Augmented Generation (RAG) та LLM-агентами, які здійснюють класифікацію, витяг фактів, побудову зв'язків, резюмування та створення керованих виходів (звіт, індикатори, задачі у SOAR). Над усім цим — керування якістю, безпекою, відтворюваністю й аудиторськими ланцюжками. Почнемо з функціональних та нефункціональних вимог.

**Функціональні:** багато-джерельний збір (HTTP/API/уебхуки/черги), нормалізація, дедуплікація, витяг сутностей (NER/NED), збагачення (реєстри, CT-логи, пасивний DNS, витоки), побудова знаннєвого графа, RAG-пошук, генерація звітів/алертів, інтеграції з SIEM/SOAR/TIP, рольовий доступ і журналювання.

**Нефункціональні:** масштабованість (горизонтальна), стійкість до відмов (відновлення задач), низька латентність для «гарячих» потоків (соцмережі, фішингові кампанії), відтворюваність (версії даних, версії промптів і моделей), безпека (OPSEC, мінімізація PII), спостережуваність (трасування пайплайнів, метрики), керовані витрати. Наступним кроком є визначення базової референс архітектури.

**Джерела даних (Sources)**

Відкритий веб (новини, блоги, форуми, paste-сайти), соцмережі (X/Twitter, Telegram, Reddit, GitHub/Issues), даркнет (форумні дзеркала, маркетплейси), реєстри (WHOIS, реєстри компаній, судові/державні портали), технічні індексатори (CT-логи, пасивний DNS, Shodan/Censys), бази витоків (відкриті/партнерські), мультимедіа (зображення/відео з публічними метаданими та C2PA-креденшалами).

**Шар інжесту (Ingestion)**

Планувальники/краулери (розклад або подієві тригери), конектори API (з повагою до rate-limit), черги подій (Kafka/RabbitMQ) для вирівнювання навантаження,

«проксі-ферми»/ізольовані середовища (OPSEC), початкові фільтри (дзеркала сторінок, антидублікат, антиспам, базова мовна ідентифікація).

**Попередня обробка і нормалізація (Preprocessing)**

Очистка HTML/скриптів/реклами, уніфікація кодувань, токенізація, сегментація документів, детект мови, видалення дублів, зведення до уніфікованих схем: Document (text+meta), Event (подія/згадка), Entity (типізована сутність), Indicator (IoC), Media (з метаданими EXIF/C2PA).

**Сховище та індексація**

Data Lake (об'єктне сховище для «сирих» і нормалізованих даних), пошуковий індекс (Elasticsearch/OpenSearch) для текстового/фасетного пошуку, графова БД (Neo4j/JanusGraph) для знаннєвого графа, timeseries для телеметрії, окремі колекції для артефактів (хеші, зразки, попередні класифікації).

**Збагачення і кореляція (Enrichment)**

Підключення зовнішніх сервісів (реєстри, CT-логи, пасивний DNS, HIBP, репутаційні списки), внутрішніх довідників (активи, «whitelist»/«greylist»/«blacklist»), прив'язка часових шкал, геоприв'язка, «entity resolution» (злиття сутностей, що дублюються), обчислення ознак для ML/LLM.

**Знаннєвий граф (Knowledge Graph)**

Вузли: Actor/Person, Org, Account, Email, Domain, IP, Cert, Repo, Media, Location, Narrative. Ребра: owns, uses, mentions, registered_to, resolves_to, co_occurs_with, part_of_campaign, linked_by_certificate, seen_in_breach, disseminates. Граф слугує контекстом для RAG та основою пояснюваності.

**Модуль ШІ (LLM/RAG/Agents)**

**RAG-шина** (retriever + векторний індекс, колекції знань за доменами), **LLM-сервіс** (комерційна/відкрита модель, обмеження токенів, версійність), **агентний оркестратор** (планування кроків, «tool use»: веб-запит, пошук у графі, перевірка у CT-логах, виклик класифікаторів), **політики безпеки промптів** (санітаризація, шаблонізація, захист від prompt-ін'єкцій).

**Аналітичні продукти і дії**

Резюме, дайджести ризиків, «живі» звіти (STIX/TAXII), сигнали в SIEM/XDR, плейбуки у SOAR (автоблок доменів, сповіщення, створення кейсу), інтерактивні дашборди (тенденції наративів, карта інфраструктурних експозицій, «теплові» карти індикаторів).

**Керованість, безпека і відтворюваність**

RBAC/ABAC, маскування/анонімізація PII, логування промптів/відповідей, версії моделей/шаблонів, «data lineage» (звідки що взято), ліміти і кешування API, OPSEC (ізольовані середовища для даркнет), тестування (red-teaming LLM), моніторинг метрик якості (precision/recall для NER/класифікації, Hallucination-score), SLO/SLI. Надалі узгоджуємо модель даних (ядро) :

**Сутності (Entity):**

entity_id, type, value, normalized_value, first_seen, last_seen, source_refs[], confidence, attributes{lang, geo, alias[], hash…, pii_flag}

**Згадка/Подія (Event/Mention):**

event_id, timestamp, source_id, doc_id, text_span, entities[], indicators[], topic_labels[], sentiment, credibility_score, media_refs[]

**Індикатор (IoC):**

indicator_id, class, value, related_entities[], sightings[], tlp, confidence, expiration, actions[]

**Графові ребра (Edge):**

src_entity, relation, dst_entity, weight, first_seen, last_seen, evidence_refs[]

**Аудит (Audit Trail):**

run_id, pipeline_stage, transformer_version, prompt_template_id, model_version, input_hash, output_hash, operator_id(optional)

Таблиця 3.1.

Логічні компоненти та інтерфейси

| Компонент | Основна функція | Вхід/Вихід | Критичні налаштування |
|---|---|---|---|
| Ingestion | Краулінг, API, черги | URL/API → RawDoc | Rate-limit, retries, proxy/OPSEC |
| Preprocess | Очистка/нормалізація | RawDoc → CleanDoc | Мова, антидубль, HTML-sanitizer |
| Enrichment | Збагачення/кореляція | CleanDoc → Entity/Event/IoC | HIBP/CT/PDNS ключі, time-join |
| Knowledge Graph | Зв’язки/атрибуція | Entity/Event → Graph | Пороги злиття, ваги джерел |
| RAG/LLM | Витяг/резюме/агенти | Query+Context → Output | Шаблони промптів, векторний індекс |

**RAG-контур.** Векторна індексація абзаців/фрагментів (багатомовні ембеддинги), ретривал релевантних знань, вставка у системний промпт як grounding, генерація відповіді/резюме з цитуванням джерел. Важливо: контроль довжини контексту, анти-leak політики, розмежування чутливих даних. **Агенти.** Планувальник («розбий завдання»), виклики інструментів (пошук у графі, CT-логи, DNS-запит, HIBP-перевірка), консолідація проміжних висновків, створення

артефактів (таблиць сутностей/IoC), ініціація дій у SOAR за правилами. **Шаблонізація промптів.** Типові патерни: **Extract → Normalize → Link → Summarize → Act**. Для кожного кроку — контрольні питання (self-check) і валідаційні правила (наприклад, не робити висновків без ≥N незалежних підтверджень). Визначаємо потоки для різних типів контенту:

**Текст (веб, форуми, соцмережі, витоки):** сувора нормалізація, NER/NED, класифікація (теми/ризики/мови), дедуплікація цитат/репостів, формування «канонічного» документа.

**Зображення/відео:** метадані (EXIF, C2PA), OCR, reverse-image пошук (зовнішні сервіси через «санітарний» проксі), геолокація/топоніміка, часові клейма, верифікація автентичності, зв'язування з наративами.

**Код/репозиторії:** пошук секретів, індикаторів у README/issues, відповідність ліцензіям, відстеження «supply-chain» артефактів.

**Технічні індекси (Shodan/CT/PDNS):** періодичне оновлення, злиття з внутрішнім «каталогом активів», розрізнення своїх/чужих сутностей, авто-створення задач із пріоритизацією.

Інтеграція з корпоративною безпекою

**SIEM/XDR:** надсилання нормалізованих подій/IoC зі вказанням джерел і часу, правила кореляції «зовнішній контекст + внутрішня телеметрія».

**SOAR:** плейбуки реагування (блок домену, сповіщення, створення квитків, повідомлення клієнтам), гейти для людини (human-in-the-loop) у чутливих діях.

**TIP/MISP/OpenCTI:** експорт/імпорт STIX/TAXII, синхронізація довідників і кампаній, обмін TTP.

**ITSM/Jira/ServiceNow:** автоматичні завдання для інфраструктурних команд (ремедіація відкритих портів, ротація секретів, оновлення політик). Надалі визначаємо метрики:

**Технічні метрики:** latency інжесту, % дублів, помилки парсерів, покриття джерел, cost per retrieved doc/1000 токенів, кеш-хіти.

**ML/NLP метрики:** precision/recall F1 для NER/класифікації, hallucination-rate з вибірковою перевіркою, faithfulness у RAG (наскільки відповіді ґрунтуються на наданих джерелах).

**Операційні метрики SOC:** MTTD/MTTR у сценаріях із зовнішнім контекстом, % автоматично підтверджених/спростованих індикаторів, зменшення «шуму» у черзі інцидентів. Визначаємо безпекові, правові та етичні аспекти:

**OPSEC:** ізольовані рантайми для доступу до даркнет/ризикованих джерел, блокування активного контенту, контроль витоків через LLM (шифрування промптів, маскування PII).

**Ліцензування/ToS:** дотримання умов використання API/сайтів, облік rate-limit, кеши з політикою життєвого циклу даних.

**PII/комплаєнс:** мінімізація збору, анонімізація, доступ за ролями, видимість логів доступу, видалення на запит.

**LLM-ризики:** prompt-ін’єкції, data poisoning, supply-chain моделей; контрзаходи — шаблони системних промптів, фільтри інструкцій, валідаційні агенти, red-teaming і блок-листи небезпечних інструментів.

Експлуатація, спостережуваність і життєвий цикл

**LangOps/MLOps:** реєстри моделей і промптів, версіонування, канарейкові релізи, A/B-тести відповідей, «playback» сеансів для розбору інцидентів якості.

**Спостережуваність:** трасування конвеєрів (OpenTelemetry), дешборди інжесту, алерти по LLM-латентності/квотах/помилках, аудит дій агентів.

**Вартість:** політики тротлінгу, кешування RAG, «cold storage» для рідко використовуваних даних, бюджетні ліміти на зовнішні API, пріоритизація гарячих тем/джерел.

Таблиця 3.2.

Метрики якості та SLO

| Категорія | Метрика | Ціль |
|---|---|---|
| Ingestion | Час від появи → індекс (P95) | ≤ 5 хв (гарячі джерела) |
| NER/Класифікація | F1 по тест-корпусу | ≥ 0,85 |
| RAG | Faithfulness (ручна вибірка) | ≥ 0,9 |
| LLM | Частка «галюцинацій» у аудити | ≤ 5% |
| SOC ефективність | MTTD/MTTR із OSINT-підсиленням | −30% / −25% до бази |

Сценарій роботи архітектури технології автоматизації OSINT із модулем ШІ можна описати на прикладі повного ланцюжка від моменту збору «сирих» даних до автоматизованої дії. Уявімо, що краулер за розкладом збирає згадки певного бренду в Telegram-каналах або на форумах і кладе ці повідомлення у чергу для подальшої обробки. На етапі нормалізації система видаляє HTML-елементи, визначає мову документа і виокремлює релевантні фрагменти. Далі починається збагачення, під час якого відбувається звернення до таких сервісів, як HIBP, CT-логи чи PDNS, а також виконується прив’язка споріднених доменів і створення сутностей. Зібрана інформація надходить у знаннєвий граф, який отримує вузли та ребра, агрегує повторювані згадки і підраховує «вагу» кампанії. На наступному етапі підключається RAG-агент, який формує відповідь на запит аналітика, наприклад: «які нові домени, патерни чи ризики були виявлені?». У відповіді додаються цитати з джерел і часові позначки для забезпечення прозорості. Якщо ж система фіксує появу нового фішингового домену, інтегрований SOAR автоматично запускає відповідний плейбук: виконується перевірка, блокується домен, надсилається сповіщення відповідальним

особам і створюється інцидент у системі керування кейсами. На фінальному кроці аналітик верифікує найбільш критичні дії, а система зберігає весь ланцюжок доказів, причому метрики цього процесу автоматично потрапляють у дешборди контролю якості.

Запропонована архітектура передбачає низку практичних рекомендацій для розгортання. Оптимально впроваджувати її поетапно: спершу будувати так званий «хребет» із шарів інжесту, нормалізації та індексу чи графа, після чого підключати RAG-модуль і найчастіші інструменти агента, а вже потім інтегрувати рішення з SIEM чи SOAR. Для масштабування доцільно застосовувати горизонтальний підхід із використанням незалежних воркерів для інжесту й обробки, окремих черг для «гарячих» потоків, кешування результатів RAG і розподілу графа на шардовані сегменти. Контроль вартості та ресурсів здійснюється за рахунок використання локальних ембеддингів, пакетних ретривів, квантизації моделей і політик таймаутів чи back-off для дорогих інструментів. Важливим аспектом є культура якості, яка включає «LangOps»-практики — рев'ю промптів, канарейкові релізи, відслідковування дрейфу якості, а також регулярне тестування на предмет галюцинацій і помилок атрибуції.


Рис. 3.2. Значення LangOps практики

У підсумку така архітектура є керованим, масштабованим і безпечним способом перетворювати відкриті дані на практичний продукт для кіберзахисту. Її сила полягає в поєднанні надійного інжесту, дисципліни роботи з даними, побудови знаннєвого графа, використання RAG і агентів та інтегрованих дій у SIEM і SOAR. Це рішення природно підсилює роботу SOC чи команд кіберрозвідки, пришвидшує тріаж інцидентів, покращує якість атрибуції, зменшує інформаційний шум і найголовніше — створює відтворювані, аудитовані аналітичні артефакти, які готові для використання в обороні, звітності та прийнятті управлінських рішень.

### 3.2. Реалізація прототипу технології (Python, GPT-моделі, SpiderFoot API)

Реалізація прототипу технології автоматизації OSINT-аналізу з використанням інструментів штучного інтелекту передбачає створення інтегрованого середовища, яке об’єднує збір даних, їх попередню обробку, інтелектуальний аналіз і генерацію підсумкових звітів. Для цього було обрано три ключові складові: Python як універсальну мову інтеграції та обробки, SpiderFoot API як основне джерело OSINT-даних та GPT-моделі як інструмент інтерпретації та узагальнення.

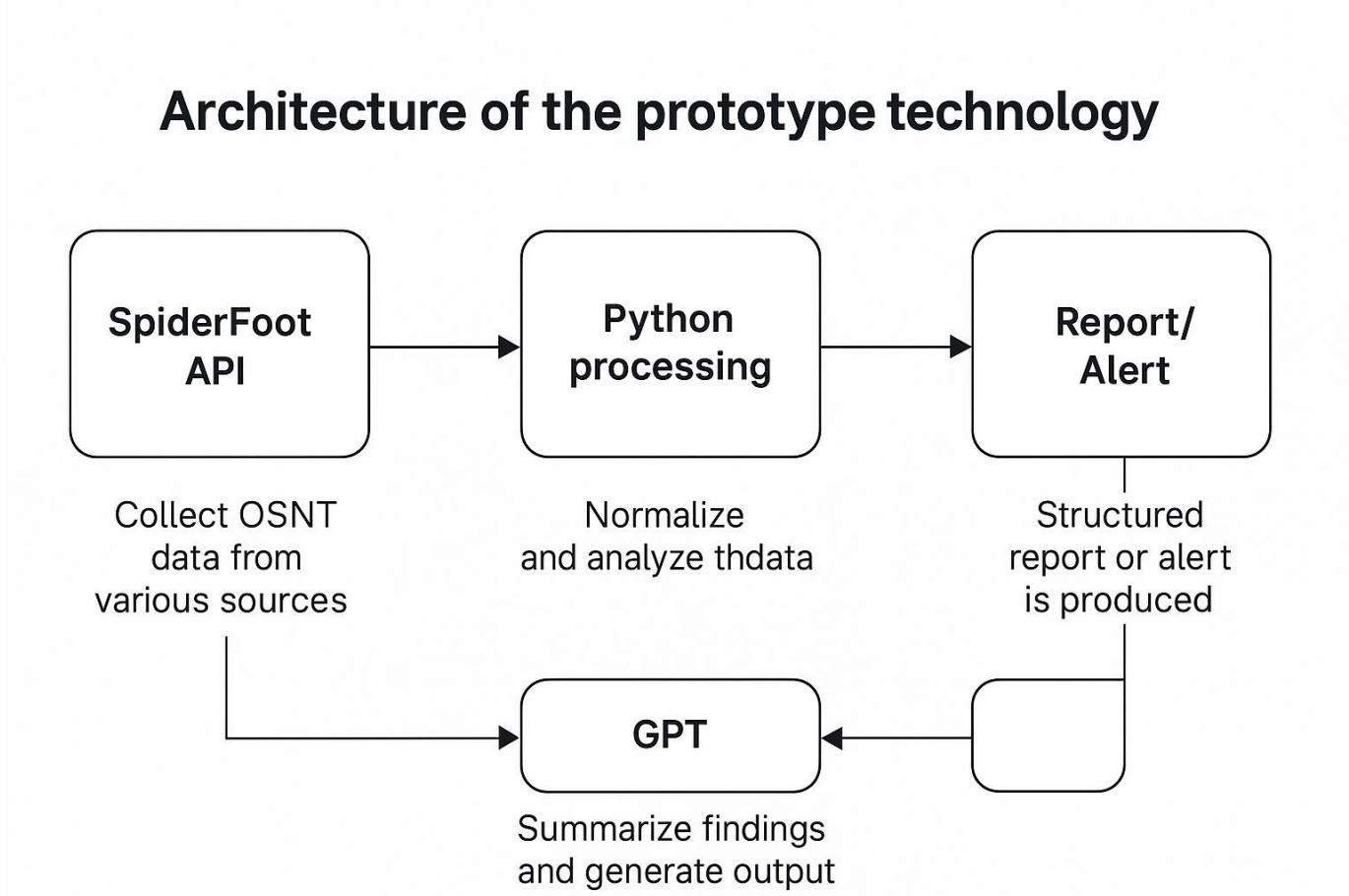


Рис 3.3 — Архітектура прототипу

Прототип реалізується як багатошаровий конвеєр обробки даних. Його логіка зводиться до кількох основних кроків:

Збір даних через SpiderFoot API: Система запускає сканування певного об’єкта (домен, IP-адреса, email чи акаунт), отримує результати в JSON-форматі та зберігає їх для подальшої обробки.

Попередня обробка та нормалізація: Python-скрипти видаляють шум, приводять дані до єдиного формату, структурують ключові сутності (домен, IP, email, акаунт).

Інтеграція з GPT-моделлю: Оброблені дані передаються у LLM, яка виконує узагальнення, класифікацію, виявлення патернів та формує зрозумілий для аналітика результат.

Формування підсумкового звіту: На виході створюється як вільний текстовий звіт, так і структурований JSON для інтеграції з іншими системами безпеки (наприклад, SIEM чи SOAR).

Логування та контроль: Кожен крок фіксується у журналі для забезпечення відтворюваності та можливості подальшого аудиту.

Скрипт інтеграції SpiderFoot API:

```
import requests
import time
API_KEY = "your_spiderfoot_api_key"
BASE_URL = "http://localhost:5001"
def start_scan(target):
    url = f"{BASE_URL}/scan/new"
    payload = {
        "target": target,
        "modules": ["sfp_dnsresolve", "sfp_breachcomp", "sfp_googlesearch"],
        "type": "domain",
        "options": {}
    }
    headers = {"Authorization": f"Bearer {API_KEY}"}
```

```
    response = requests.post(url, json=payload, headers=headers)
    return response.json()
def poll_results(scan_id):
    url = f"{BASE_URL}/scan/{scan_id}/data"
    headers = {"Authorization": f"Bearer {API_KEY}"}
    while True:
        response = requests.get(url, headers=headers)
        if response.status_code == 200:
            data = response.json()
            if data.get("status") == "FINISHED":
                return data
        time.sleep(10)
```

Тут реалізовано два модулі: запуск сканування та отримання результатів. Завдяки цьому Python-скрипт працює як диспетчер, який повністю контролює процес збору OSINT-даних.

Наступним кроком є створення коду для передачі сирих результатів SpiderFoot у GPT та формування текстового звіту :

```
from openai import OpenAI
client = OpenAI(api_key="your_openai_api_key")
def summarize_results(scan_data):
    prompt = f"""
    Analyze the following SpiderFoot scan data and summarize:
    1. Key findings about domains, IPs, emails.
    2. Potential threats or leaks.
    3. Recommended next steps.
    Data:
    {scan_data}
    """
```

```
response = client.chat.completions.create(
    model="gpt-4",
    messages=[{"role": "user", "content": prompt}],
    max_tokens=500
)
return response.choices[0].message["content"]
```

Для отримання результативного звіту, що відображає всі дії в системі та додатково повертає структуровані дані у JSON, створюємо скрипт:

```
import logging
    logging.basicConfig(filename="osint_pipeline.log", level=logging.INFO)

    def log_action(action, details=""):
        logging.info(f"{action} | {details}")
    {
        "domains": ["phishing-login.org", "malicious-example.com"],
         "ips": ["185.142.56.77", "45.63.91.120"],
        "emails": ["admin@company.com", "support@phishing.org"],
         "recommendations": [
         "Block phishing-login.org in firewall",
        "Investigate IP 185.142.56.77 for malicious activity"
        ]
    }
```

Кожен етап (початок сканування, отримання результатів, запуск GPT, формування звіту) супроводжується логуванням, що створює повноцінний audit trail.

У межах прототипу процес роботи виглядає наступним чином. Аналітик визначає цільовий домен або інший об’єкт для перевірки й запускає аналіз. Далі система за допомогою SpiderFoot API ініціює сканування, яке збирає всі доступні у відкритих джерелах дані: DNS-записи, згадки у витоках, інформацію з пошукових

систем та суміжних індексів. Отримані результати передаються до Python-скриптів, які виконують попередню обробку: видаляють зайві елементи, усувають дублікати сутностей, синхронізують часові формати та структурують дані у єдиному вигляді. Після цього інформація передається до модуля GPT, який аналізує зібраний масив, виокремлює ключові загрози, знаходить закономірності та формує список практичних рекомендацій. На виході аналітик отримує як текстовий звіт із поясненнями, так і структурований JSON із переліком доменів, IP-адрес та інших індикаторів компрометації. У разі потреби результати можуть бути автоматично інтегровані до SIEM чи SOAR для подальшого реагування — наприклад, блокування домену, створення інциденту або запуску плейбуку автоматизованих дій.

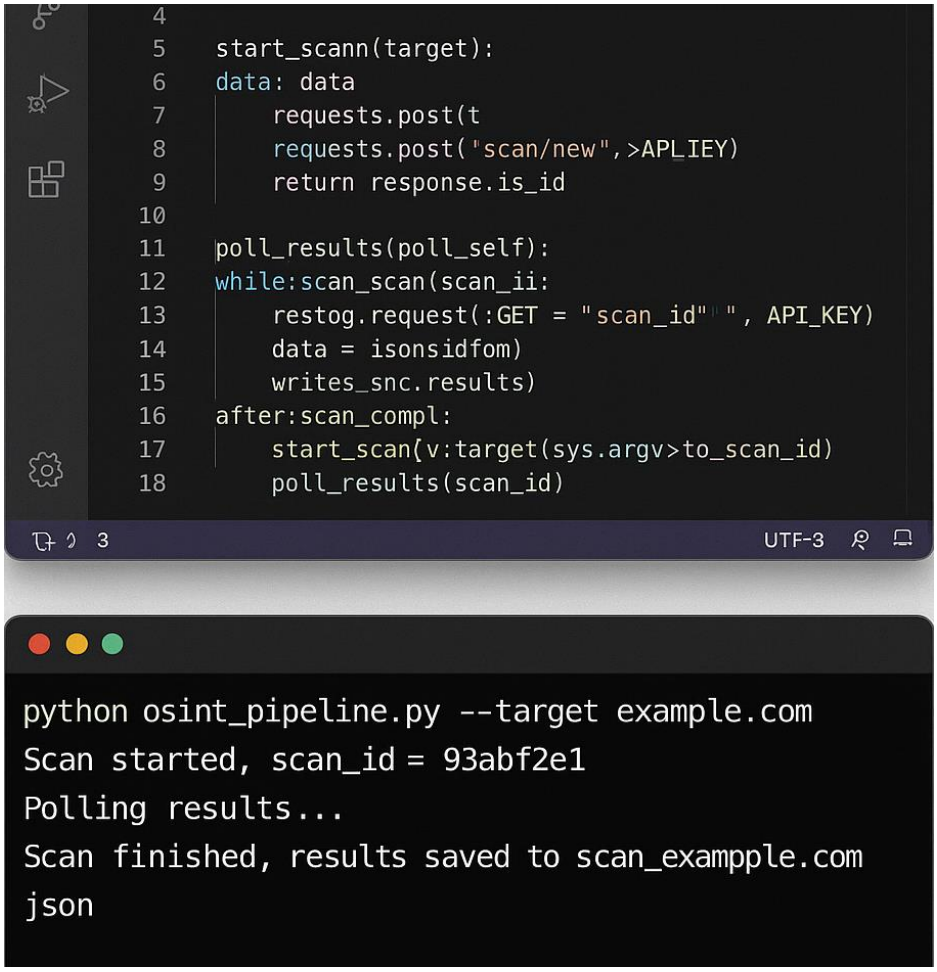


Рис 3.4 - Ініціалізація скрипту

Таким чином, розроблений прототип демонструє можливість побудови ефективної технології автоматизації OSINT-аналізу на основі інтеграції Python,

SpiderFoot API та GPT-моделей. Поєднання інструментів дозволяє здійснювати повний цикл: від збору відкритих даних до створення аналітичних продуктів, придатних для практичного використання в системах кіберзахисту.

scan_example.com.json


```
{
  "scan_id": "93abf2e1",
  "target": "example.com",
  "start_time": "2024-04-24|T
  "results": [
    {
      "id": "1",
      "type": "IP Address",
      "value": "93.184.216.34",
      "source": "ip"
    },
      "id": "2",
      "type": "www.example.com",
      "value": "dns"
      "source": "email"
    },
      "id": "3",
      "type": "Email Address",
      "value": "admin@example.com"
      "source": "email"
    },
      "id": "4",
      "type": "TCP Port",
      "value": 80,
      "additional": "http",
      "source": "port"
    ]
  }
```


Рис 3.5 — Результат відпрацювання скрипту на створеному ресурсі для тесту

Особливістю підходу є можливість одночасно отримувати як структуровані індикатори, так і зрозумілі текстові пояснення, що спрощує роботу як для технічних

спеціалістів, так і для керівників безпеки. Запропонований прототип підтверджує, що автоматизація OSINT з використанням ШІ є реалістичною, масштабованою та перспективною для впровадження у корпоративні середовища.

### 3.3 Рекомендації щодо застосування технологій ШІ для підвищення ефективності OSINT - аналізу

Розвиток технологій штучного інтелекту істотно впливає на підходи до аналізу відкритих джерел інформації. Якщо раніше OSINT спирався переважно на інструменти пошуку, кореляції та візуалізації даних, то сьогодні акцент переноситься на когнітивні можливості систем, здатних не лише збирати інформацію, але й інтерпретувати її у наближеному до людського мислення форматі. Це відкриває нові можливості, однак вимагає від організацій, що працюють у сфері кібербезпеки, формування чітких підходів і рекомендацій до впровадження ШІ.

Використання мовних моделей у контексті OSINT має базуватися на усвідомленні того, що аналітик не може бути замінений машиною, але може бути істотно підсилений нею. Першим важливим аспектом є правильний вибір моделі та архітектури її застосування. Для завдань поверхневого аналізу достатньо універсальних моделей, здатних швидко узагальнювати контент та робити висновки щодо рівня ризику. Однак для більш глибоких завдань, зокрема витягу індикаторів компрометації, атрибуції атак чи побудови зв’язків між сутностями, необхідно адаптувати або додатково навчати моделі на спеціалізованих датасетах. Це означає, що організація має вирішити питання не лише технічної інтеграції, але й підготовки навчальних корпусів, які відображатимуть її специфічні потреби.

Другою ключовою рекомендацією є забезпечення інтеграції ШІ не як окремого додатку, а як логічної надбудови над уже наявними інструментами OSINT. SpiderFoot, Shodan, Maltego та інші інструменти збирають значні обсяги інформації, однак без інтерпретації вони перетворюються на масиви даних, з якими складно

працювати. ШІ дозволяє цю інформацію структурувати, виокремлювати головне та подавати результати у вигляді, придатному для використання. У такій моделі саме класичні системи відповідають за збір та технічну кореляцію, а штучний інтелект бере на себе когнітивний рівень аналітики, зводячи дані у звіти, виявляючи патерни та пропонуючи ймовірні сценарії реагування.

Третій напрямок пов’язаний із автоматизацією рутинних завдань. В OSINT-аналізі велика частка часу витрачається на перевірку доменів, облікових записів, пошук у базах витоків, перевірку IP-адрес на наявність у чорних списках чи репутаційних системах. ШІ здатен перебрати на себе ці завдання, виконуючи дедуплікацію сутностей, виокремлення унікальних об’єктів, створення таблиць для подальшого аналізу. Це суттєво скорочує навантаження на аналітиків, які можуть приділяти більше часу завданням, що потребують людської експертизи, зокрема розумінню мотивацій атакувальників чи прогнозуванню подальших дій.

Водночас автоматизація не може бути повною. Усі критичні рішення мають залишатися за людиною, а роль ШІ повинна бути допоміжною. Це так званий підхід «human-in-the-loop», який гарантує баланс між швидкістю реагування та якістю прийнятих рішень. Якщо модель виявила підозрілий домен і запропонувала його блокування, остаточне рішення має приймати аналітик після перевірки контексту. Такий підхід забезпечує контроль і знижує ризики помилкових спрацювань, які можуть спричинити небажані наслідки для організації.

Особливої уваги заслуговує питання відтворюваності та прозорості. Результати, згенеровані мовними моделями, не повинні сприйматися як «чорна скринька». Використання підходу Retrieval-Augmented Generation дозволяє зберегти прозорість, оскільки модель не просто формує текст, а наводить цитати з джерел, що підтверджують її висновки. У поєднанні з обов’язковим логуванням усіх дій — від промптів до версій моделей — це створює умови для повноцінного аудиту, що є надзвичайно важливим у сфері кібербезпеки, де кожне рішення має мати підґрунтя.

Ще однією рекомендацією є розширення спектра аналізованих даних. Сучасний OSINT виходить далеко за межі текстових повідомлень. Величезну роль відіграють зображення, відео та аудіо, які можуть містити цінну інформацію для розслідувань. Залучення мультимодальних моделей дозволяє аналізувати не лише текст, але й візуальні дані: витягувати метадані з фотографій, виконувати геолокацію зображень, визначати автентичність відеозаписів. Таке розширення можливостей робить OSINT значно потужнішим і дозволяє виявляти загрози, які раніше залишалися поза увагою.

Важливим є і питання оцінювання ефективності. Впровадження ШІ у OSINT має сенс лише тоді, коли воно дає вимірюваний результат. Тому рекомендовано визначати метрики, які відображають реальний вплив: скорочення часу виявлення та реагування на інциденти, відсоток рутинних завдань, що автоматизовані, точність витягу сутностей, рівень задоволеності аналітиків, кількість рекомендацій моделі, які були прийняті без змін. Систематичний збір таких показників дозволить оцінювати не лише технологічний, але й організаційний ефект від використання ШІ.

Не можна оминути увагою і правові та етичні аспекти. Робота з відкритими даними нерідко супроводжується взаємодією з персональною інформацією. Використання ШІ посилює ці ризики, адже моделі можуть ненавмисно обробляти або навіть зберігати конфіденційні дані. Тому впровадження має супроводжуватися політикою анонімізації, дотриманням нормативних вимог (зокрема GDPR) та використанням виключно легальних джерел. Не менш важливим є захист промптів і відповідей, адже вони також можуть містити службову чи чутливу інформацію.

Усе це потребує не лише технічних, але й організаційних заходів. Організації повинні створювати окремі групи для управління процесами, пов’язаними зі ШІ, впроваджувати практики LangOps і MLOps, що включають версіонування моделей, контроль за якістю та планування релізів. Необхідно проводити регулярне навчання співробітників, формувати бібліотеки промптів для типових сценаріїв, забезпечувати постійну взаємодію між технічними спеціалістами та керівниками.

Таким чином, застосування технологій штучного інтелекту в OSINT — це не лише питання інструментів. Це комплексна трансформація аналітичних процесів, що вимагає інтеграції, відтворюваності, контролю, етичної відповідальності та організаційної підтримки. Виконання цих рекомендацій дозволяє створити ефективну систему, здатну працювати з величезними масивами відкритих даних, виокремлювати з них найважливіше та оперативно трансформувати результати у практичні рішення для підрозділів кібербезпеки.

# ВИСНОВКИ

У роботі проведено комплексне дослідження проблеми застосування технологій OSINT у сфері кібербезпеки та проаналізовано обмеження традиційних підходів до обробки відкритих даних. Визначено, що зростаючі обсяги інформації з відкритих джерел ускладнюють ручний аналіз, створюють ризик втрати релевантних даних і збільшують вплив людського фактору.

Проаналізовано роль OSINT у сучасних розслідуваннях кіберінцидентів, досліджено інструменти Maltego, SpiderFoot, Shodan та виявлено їхні сильні сторони й недоліки. Особливу увагу приділено проблемам масштабованості, інтеграції та потребі в автоматизації.

Розглянуто можливості застосування штучного інтелекту (GPT-моделей, методів машинного навчання, алгоритмів NLP) для обробки відкритих даних. Доведено, що використання ІІ дозволяє значно підвищити швидкість аналізу, автоматично класифікувати інформацію, виявляти приховані залежності та формувати аналітичні звіти.

У роботі розроблено прототип технології автоматизації OSINT-аналізу на основі Python, GPT-моделей і SpiderFoot API. Проведені експерименти підтвердили ефективність інтеграції ІІ у процес OSINT, зокрема у завданнях фільтрації нерелевантних даних, пошуку IoC та формування структурованих звітів.

Крім технічної складової, було сформульовано практичні рекомендації для спеціалістів із кіберзахисту щодо застосування технологій ІІ у реальних розслідуваннях. Зокрема, запропоновано використовувати гібридний підхід: поєднання класичних інструментів OSINT і модулів штучного інтелекту, що дозволяє отримати найбільш повну й точну картину загроз.

У результаті дослідження встановлено, що автоматизація OSINT-аналізу з використанням інструментів штучного інтелекту дозволяє суттєво підвищити

ефективність процесів кіберрозвідки, зменшити час реагування на інциденти та забезпечити більш високий рівень захисту інформаційних систем. Розроблені рекомендації можуть бути використані як основа для впровадження нових практик у сфері кібербезпеки та розвитку систем аналітики загроз.

# ПЕРЕЛІК ПОСИЛАНЬ

1. Rajamäki, J., McMenamin, S. *Utilization and Sharing of Cyber Threat Intelligence Produced by Open-Source Intelligence*. Proceedings of the 19th International Conference on Cyber Warfare and Security (ICCWS 2024). URL: https://papers.academic-conferences.org/index.php/iccws/article/view/2069 (дата звернення: 10.10.2025).

2. Rahman, M. S. *The Art of Open Source Intelligence (OSINT): Addressing Cybercrime, Opportunities, and Challenges*. SSRN, 2025. URL: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=5281845 (дата звернення: 10.10.2025).

3. Nonum, E. O., Avwokuruaya, O., Ezemonye, T. *Role of Open Source Intelligence (OSINT) in Cybersecurity and Threat Analysis*. IJLTEMAS, Vol. XIV, Issue III, March 2025. URL: https://www.ijltemas.in/papers/2025/140300023 (дата звернення: 10.10.2025).

4. Braga, M. *Assessing Crowdsourced OSINT*. University of North Carolina, 2025. URL: https://cdr.lib.unc.edu/downloads/z603rc146 (дата звернення: 22.09.2025).

5. Singh, S. *An Integrative Review of Deepfake Detection: Challenges and Opportunities*. ScienceDirect, 2025. URL: https://www.sciencedirect.com/science/article/pii/S2215016125004765 (дата звернення: 22.09.2025).

6. NIST. *Artificial Intelligence Risk Management Framework (AI RMF 1.0)*. URL: https://nvlpubs.nist.gov/nistpubs/ai/nist.ai.100-1.pdf (дата звернення 22.09.2025); див. також огляд: https://www.nist.gov/itl/ai-risk-management-framework. NIST Publications+1

7. OWASP. *Top 10 for Large Language Model Applications / GenAI Security Project* (останні оновлення). URL: https://owasp.org/www-project-top-10-for-large-language-model-applications/ та https://genai.owasp.org/llm-top-10/ (дата звернення 22.09.2025).

8. C2PA (Coalition for Content Provenance and Authenticity). *Technical Specifications / Content Credentials*. URL: https://c2pa.org/specifications/specifications/2.2/index.html та https://c2pa.org/ (дата звернення 22.09.2025). Див. також приклад впровадження на платформах: The Verge про маркування C2PA.

9. Maltego — Documentation & Transform Hub. URL: https://www.maltego.com/ (дата звернення: 22.09.2025).

10. SpiderFoot — Automated OSINT Collection and Analysis. URL: https://www.spiderfoot.net/ (дата звернення: 22.09.2025).

11. Shodan — The Search Engine for the Internet of Things; Search Query Fundamentals & API. URL: https://www.shodan.io/ (дата звернення: 22.09.2025).

12. Have I Been Pwned — Check if your email has been compromised in a data breach; API. URL: https://haveibeenpwned.com/ (дата звернення: 22.09.2025).

13. Shafee, S. *Evaluation of LLM-based chatbots for OSINT-based Cyber Investigations*. *Expert Systems with Applications*, ScienceDirect, 2025. URL: https://www.sciencedirect.com/science/article/pii/S0957417424023765 (дата звернення: 24.09.2025).

14. Yuan, X. *Empowering LLMs with Toolkits: An Open-Source Intelligence Acquisition Agent*. *Future Internet*, 2024, Vol. 16, No. 12, Article 461. URL: https://doi.org/10.3390/fi16120461 (дата звернення: 24.09.2025).

15. Černý, J. *Implications of Large Language Models for OSINT: Assessing the Impact on Information Acquisition and Analyst Expertise in Prompt Engineering*. Proceedings of ECCWS 2024. URL: https://papers.academic-conferences.org/index.php/eccws/article/view/2261 (дата звернення: 24.09.2025).

16. Berzinji, A. *Utilisation of Large Language Models in OSINT: Identifying Extremist Content on Social Media*. *Cybercrime Journal*, 2024. URL: https://cybercrimejournal.com/menuscript/index.php/cybercrimejournal/article/view/389 (дата звернення: 24.09.2025).

17. ENISA. *ENISA Threat Landscape 2024*. URL: https://www.enisa.europa.eu/publications/enisa-threat-landscape-2024 (дата звернення: 24.09.2025).

18. Microsoft Security Blog. *Microsoft Security Copilot: Using AI to strengthen SOC capabilities*. URL: https://www.microsoft.com/security/blog/ (дата звернення: 24.09.2025).

19. Recorded Future. *Annual Cyber Threat Analysis Report 2024/2025*. URL: https://www.recordedfuture.com/resources (дата звернення: 24.09.2025).

20. Google Cloud Chronicle. *Security Operations and Threat Intelligence*. URL: https://cloud.google.com/security/products/security-operations (дата звернення: 24.09.2025).

# ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ (ПРЕЗЕНТАЦІЯ)