КІБЕРБЕЗПЕКА: освіта, наука, техніка
CYBERSECURITY: EDUCATION, SCIENCE, TECHNIQUE
№ 1 (33), 2026
ISSN 2663 – 4023
DOI 10.28925/2663-4023.2026.33.1219
УДК 004.056.5:004.415.2:004.7

Полотай Орест Іванович
кандидат технічних наук, доцент, доцент кафедри управління інформаційною безпекою
Львівський державний університет безпеки життєдіяльності, Львів, Україна
ORCID: 0000-0003-4593-8601
orest.polotaj@gmail.com

Кухарська Наталія Павлівна
кандидат фізико-математичних наук, доцент, доцент кафедри безпеки інформаційних технологій
Національний університет Львівська Політехніка, Львів, Україна
ORCID: 0000-0002-0896-8361
kukharska.n@gmail.com

Ткаченко Артур Мар'янович
викладач кафедри управління інформаційною безпекою
Львівський державний університет безпеки життєдіяльності, Львів, Україна
ORCID: 0009-0009-6830-4741
tkachenko.am14@gmail.com

Сєдін Євген Олександрович
старший викладач кафедри управління інформаційною безпекою
Львівський державний університет безпеки життєдіяльності, Львів, Україна
ORCID: 0009-0004-9654-0577
ye.sedin@ldubgd.edu.ua

Николайчук Максим Ігорович
викладач кафедри управління інформаційною безпекою
Львівський державний університет безпеки життєдіяльності, Львів, Україна
ORCID: 0009-0004-6313-5196
nykolaychuk.maksym.i@ldubgd.edu.ua

МЕТОДИ АВТОМАТИЗАЦІЇ РОЗСЛІДУВАННЯ ІНЦИДЕНТІВ КІБЕРБЕЗПЕКИ НА ОСНОВІ ЛОГІВ ОПЕРАЦІЙНОЇ СИСТЕМИ WINDOWS З ВИКОРИСТАННЯМ PYTHON ДЛЯ ПІДТРИМКИ УПРАВЛІННЯ ІНФОРМАЦІЙНОЮ БЕЗПЕКОЮ

Анотація. У статті представлено підхід до аналізу та візуалізації ризиків інформаційної безпеки на основі обробки системних та мережевих логів із застосуванням програмних засобів автоматизації. Розроблено алгоритм на мові Python, який забезпечує збір, структуризацію та аналіз подій, що відбуваються в операційній системі та мережевих пристроях, з метою виявлення потенційно підозрілої активності. Для обробки даних використано бібліотеки pandas та datetime, що дозволяють ефективно працювати з великими обсягами інформації та часовими мітками, а для візуалізації результатів застосовано matplotlib, що забезпечує наочне подання закономірностей та аномалій. Алгоритм здійснює класифікацію подій за визначеними критеріями підозрілої активності, враховуючи їх тип, частоту та часові характеристики. Отримані графічні моделі дозволяють оцінити рівень ризику у різних сегментах системи та приймати обґрунтовані управлінські рішення щодо забезпечення інформаційної безпеки. Проведено експериментальну перевірку алгоритму на прикладі реальних логів, що підтвердило його ефективність у ранньому виявленні аномальної поведінки та оптимізації процесів моніторингу. Результати дослідження підкреслюють важливість інтеграції методів аналізу логів та візуалізації даних у сучасні системи управління інформаційною безпекою. Використання програмних засобів автоматизації сприяє мінімізації людського фактора, підвищенню точності оцінки ризиків та оперативності реагування на загрози. Стаття має практичне та наукове значення, оскільки пропонує методологію побудови ефективної системи моніторингу та раннього попередження інцидентів кібербезпеки.

Ключові слова: операційні системи, інформаційна безпека, кіберзагрози, інциденти кібербезпеки, програмування, менеджмент інформаційної безпеки, логування, комп'ютерна криміналістика, ризики інформаційної безпеки.

ВСТУП

У сучасних умовах стрімкого розвитку цифрових технологій та зростання залежності організацій від інформаційних систем питання забезпечення інформаційної безпеки набуває особливої актуальності. Кількість кібератак та інцидентів кібербезпеки щороку зростає, а їх наслідки можуть призводити до витоку конфіденційних даних, порушення працездатності критичних сервісів та значних фінансових втрат. У зв'язку з цим важливою складовою кіберзахисту є не лише попередження атак, але й оперативне та якісне розслідування інцидентів інформаційної безпеки.

Одним із ключових джерел інформації для проведення розслідувань є журнали подій операційних систем. Операційна система Windows, яка широко використовується у корпоративному середовищі, формує значну кількість подій у системних журналах, зокрема у розділах Security, System та Application. Дані журнали містять інформацію про входи користувачів, зміни прав доступу, запуск процесів, встановлення програмного забезпечення, помилки системи та інші дії, що можуть бути пов'язані з інцидентами інформаційної безпеки. Проте ручний аналіз великого обсягу логів є трудомістким і неефективним, що створює потребу у впровадженні автоматизованих методів обробки та інтерпретації подій.

Автоматизація процесів розслідування інцидентів дозволяє скоротити час реагування, знизити навантаження на спеціалістів з інформаційної безпеки та підвищити точність виявлення підозрілої активності. Одним із ефективних інструментів для створення таких рішень є мова програмування Python, яка завдяки своїм бібліотекам та простоті інтеграції дозволяє реалізувати аналіз логів, фільтрацію подій та формування звітів для подальшого прийняття управлінських рішень.

Постановка проблеми. Зростання кількості кібератак та інцидентів кібербезпеки вимагає від організацій оперативного реагування та ефективного розслідування подій, пов'язаних із порушенням конфіденційності, цілісності або доступності інформаційних ресурсів. Проте на практиці процес розслідування інцидентів часто ускладнюється значним обсягом інформації, яку необхідно обробити, а також недостатнім рівнем автоматизації аналітичних процедур. Особливо це стосується аналізу журналів подій операційних систем, які можуть містити десятки тисяч записів за короткий період часу.

Операційна система Windows формує різноманітні журнали подій, які фіксують активність користувачів, зміни у системі, мережеві підключення, запуск процесів та інші важливі дії. Водночас значна частина подій має технічний або службовий характер, що ускладнює виявлення справді критичних записів, пов'язаних із потенційними інцидентами. Ручний перегляд логів є трудомістким, потребує високої кваліфікації спеціалістів та може призводити до помилок або пропуску важливих ознак атаки.

Таким чином, виникає проблема необхідності створення ефективних методів автоматизованого аналізу логів Windows, які дозволяють швидко виділяти підозрілі події, здійснювати первинну кореляцію даних та формувати результати у вигляді зрозумілих звітів. Додатково важливою є інтеграція таких методів у систему управління інформаційною безпекою організації, оскільки результати розслідування мають використовуватися для оцінки ризиків, прийняття управлінських рішень та вдосконалення політик безпеки.

Отже, актуальною задачею є розробка підходу до автоматизації розслідування інцидентів інформаційної безпеки на основі журналів подій Windows із використанням Python як універсального інструменту програмної реалізації, що дозволяє забезпечити гнучкість аналізу та можливість подальшого розвитку системи.

Аналіз останніх досліджень і публікацій. Одним із ключових напрямів сучасної науки в інформаційній безпеці є систематизація та формалізація процесів розслідування інцидентів. У роботі Цирканюка досліджено механіку розслідування інцидентів, що включає етапи моніторингу, збору даних, усунення та закриття подій, а також запропоновано моделі автоматизації процесу реагування на основі формалізованого плану обробки інцидентів у вигляді стейт-машини. Це дозволяє підвищити систематизацію та можливість автоматичного реагування на загрози у великих інформаційних системах [11].

У практичних дослідженнях впровадження автоматизації розслідування також розглядаються приклади розробки інструментів цифрової форензики для підтримки incident response (IR). Розробка автоматизованого інструментарію для цифрової форензики дозволяє ефективніше збирати, аналізувати і

зберігати цифрові докази із застосуванням розширюваних модулів та аналітичних механізмів, що сприяє підвищенню ефективності розслідування та відповідає сучасним вимогам кібербезпеки [6].

У контексті обробки логів важливим є розвиток методів лог-аналізу як фундаментального елементу розслідування інцидентів. Аналіз системних та подій журналів є основою для виявлення підозрілих дій та відстеження причин інцидентів, що підкреслюється в загальних оглядах технік обробки логів у сучасних інформаційних системах [4].

Крім того, останні дослідження з фокусом на лог-аналіз включають підходи із застосуванням машинного навчання та глибоких нейронних мереж для класифікації та семантичної обробки подій журналів. Наприклад, у статті Алзу'бі та співавторів розроблено семантичну векторизацію лог-повідомлень з використанням глибокого навчання для класифікації інцидентів, що може допомогти автоматизувати аналіз великих обсягів даних логів [1].

Існують також приклади наукових праць, що підкреслюють важливість інструментів для автоматичного розслідування, наприклад через Python-орієнтовані рішення, які інтегруються в операційні центри безпеки (SOC) та забезпечують форензичний аналіз та реагування на кіберзагрози. Такі підходи демонструють, як використання Python-фреймворків може сприяти створенню гнучких систем для автоматичної обробки сигналів безпеки, аналізу інцидентів та інтеграції в загальну безпекову архітектуру організації [3].

Незважаючи на значний прогрес, більшість наукових джерел фокусуються на узагальнених методах або симуляціях розслідувань, і лише небагато з них детально розглядають обробку логів саме операційної системи Windows як джерела даних для автоматичного розслідування. Це створює дослідницький пробіл, який виправдовує необхідність розробки спеціалізованих методів автоматизації аналізу логів Windows із використанням Python – підходу, що об'єднує операційні системи, автоматизацію та менеджмент інформаційної безпеки.

Мета статті. Метою статті є дослідження методів автоматизації розслідування інцидентів інформаційної безпеки на основі логів операційної системи Windows із застосуванням Python. У роботі розглядаються можливості використання системних журналів подій для аналізу інцидентів, описується алгоритм автоматизованої обробки подій та пропонується прототип програмного рішення, що може бути використаний як елемент підтримки управління інформаційною безпекою в організації.

ТЕОРЕТИЧНІ ОСНОВИ ДОСЛІДЖЕННЯ

Розслідування інцидентів інформаційної безпеки є важливим елементом системи кіберзахисту організації та передбачає комплекс заходів, спрямованих на виявлення причин інциденту, визначення його наслідків, встановлення джерела загрози та формування рекомендацій для запобігання подібним випадкам у майбутньому. Incident Response Technologies включають методи збору, аналізу та інтерпретації цифрових даних, що формуються інформаційними системами під час їх функціонування. Основною метою таких технологій є скорочення часу реагування та підвищення точності прийняття рішень у процесі реагування на кіберінциденти [9].

Одним із базових методів розслідування інцидентів є аналіз логів (log analysis), який полягає у дослідженні журналів подій, що містять інформацію про системну активність, мережеві з'єднання, автентифікацію користувачів, запуск процесів та зміни у конфігурації системи. Логи операційних систем є ключовим джерелом цифрових доказів, оскільки дозволяють відновити послідовність подій у часі та визначити, які саме дії могли призвести до порушення безпеки. Проте ефективність аналізу логів значною мірою залежить від налаштувань журналювання, повноти даних та здатності аналітика швидко відфільтрувати важливі події серед великого потоку інформації.

Наступним важливим підходом є кореляція подій (event correlation), яка передбачає зіставлення подій з різних джерел для побудови повної картини інциденту. Кореляція дозволяє об'єднати записи з логів операційної системи, мережевого обладнання, антивірусних систем або систем моніторингу безпеки. Такий підхід дає можливість виявляти складні багатокрокові атаки, коли окремі події не виглядають небезпечними, однак у сукупності формують ознаки цілеспрямованої загрози. Саме кореляція є основою функціонування багатьох SIEM-систем (Security Information and Event Management), які використовуються для централізованого збору та аналізу подій безпеки [10].

Важливу роль у технологіях розслідування відіграє сигнатурний аналіз (signature-based analysis), що базується на пошуку відомих шаблонів або ознак атак [5]. У межах цього методу події системи порівнюються із заздалегідь сформованими правилами або сигнатурами, що відповідають конкретним типам загроз. Наприклад, багаторазові невдалі спроби входу в систему, запуск підозрілих процесів чи виконання команд, характерних для шкідливого ПЗ, можуть бути виявлені через сигнатурні правила.

Основною перевагою цього методу є висока точність при виявленні відомих атак, однак недоліком є обмежена здатність до розпізнавання нових або модифікованих загроз [7].

Окрему групу методів становить поведінковий аналіз (behavior-based analysis), який орієнтований на виявлення аномалій у роботі системи та користувачів. На відміну від сигнатурного аналізу, поведінковий підхід базується на визначенні нормальної поведінки системи та пошуку відхилень, що можуть свідчити про компрометацію. Наприклад, незвичний час входу користувача, запуск невідомих процесів або різке зростання кількості запитів до системних ресурсів можуть розглядатися як потенційні індикатори атаки. Поведінковий аналіз є перспективним напрямом, оскільки дозволяє виявляти невідомі загрози, однак потребує належного збору статистики та більш складних алгоритмів обробки даних.

Таким чином, сучасні технології розслідування інцидентів інформаційної безпеки включають аналіз логів, кореляцію подій, сигнатурний та поведінковий аналіз. Поєднання цих методів дозволяє підвищити ефективність розслідувань, забезпечити швидке визначення причин інциденту та сформувати обґрунтовані управлінські рішення у сфері інформаційної безпеки. У контексті даного дослідження особливу увагу буде приділено аналізу логів операційної системи Windows та застосуванню інструментів автоматизації на основі Python для підтримки процесів розслідування інцидентів.

Операційні системи відіграють ключову роль у процесі розслідування інцидентів інформаційної безпеки, оскільки саме вони є центральною ланкою взаємодії користувача, прикладного програмного забезпечення та апаратних ресурсів. У процесі функціонування операційна система фіксує значну кількість подій, які можуть бути використані як цифрові докази під час розслідування. Таким чином, системні журнали подій є важливим джерелом інформації для аналізу активності користувачів, оцінки стану системи та виявлення ознак компрометації.

Операційна система Windows має розвинений механізм логування, що реалізується через Windows Event Logs (рис. 1). Події реєструються у вигляді структурованих записів, які містять інформацію про час події, ідентифікатор події (Event ID), джерело (Source), рівень критичності, опис дії та додаткові параметри. Такий формат дозволяє проводити детальний аналіз як на рівні адміністратора, так і на рівні спеціаліста з кібербезпеки. Особливу цінність у розслідуваннях мають журнали, що зберігаються у розділах Security, System та Application, оскільки вони охоплюють широкий спектр подій – від автентифікації користувачів до змін конфігурації системи.

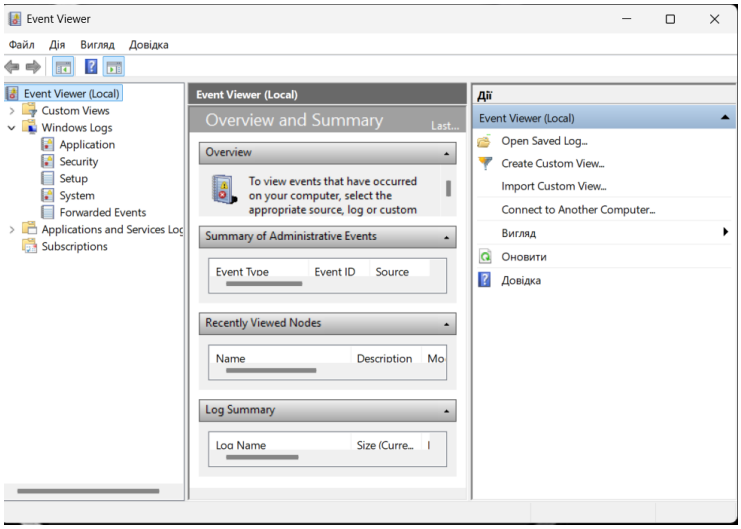

Рис. 1. Windows Event Logs

Журнал Security є основним джерелом даних для аналізу подій, пов'язаних із доступом до системи. Він містить інформацію про успішні та невдалі спроби входу, використання привілеїв, доступ до об'єктів системи, створення або видалення облікових записів, а також зміну прав доступу. Ці дані дозволяють встановити потенційні спроби несанкціонованого доступу, атаки методом підбору пароля або використання викрадених облікових даних. У свою чергу, журнал System фіксує події, пов'язані зі станом операційної системи, драйверів та служб, що може бути корисним для визначення змін у системному середовищі, перезавантажень, помилок компонентів або спроб модифікації критичних служб. Журнал Application містить інформацію про роботу програм, збої, помилки та взаємодію застосунків із системними ресурсами, що може вказувати на використання вразливостей або шкідливу активність у програмному середовищі.

Використання Windows Event Logs у розслідуваннях дозволяє відновити хронологію дій у системі, визначити початковий момент атаки, ідентифікувати потенційного порушника та встановити наслідки інциденту. Однак значним викликом є великий обсяг журналів, що швидко накопичуються у корпоративних середовищах. Це створює потребу в автоматизації процесів фільтрації, сортування та кореляції подій. Саме тому застосування програмних засобів для аналізу Windows Event Logs є актуальним напрямом, що дозволяє підвищити ефективність розслідувань та підтримати процеси управління інформаційною безпекою в організації.

Windows Event Logs здатні фіксувати широкий спектр подій, які можуть бути індикаторами інцидентів інформаційної безпеки. До найбільш поширених типів інцидентів, що відображаються у логах, належать масові невдалі спроби входу, що можуть вказувати на brute-force атаку або підбір пароля. Також важливими є успішні входи в систему у нетиповий час, входи під обліковими записами з підвищеними правами або зміни параметрів безпеки. Крім того, журнали можуть містити події, що сигналізують про зміни в системі, наприклад створення або видалення облікових записів, зміну групових політик, встановлення нового програмного забезпечення, модифікацію служб або зміну налаштувань системи. Окремо слід виділити події, пов’язані із запуском підозрілих процесів, використанням командних оболонок, сценаріїв PowerShell або інших інструментів, які часто застосовуються зловмисниками для закріплення у системі та подальшого розвитку атаки.

Попри значну інформативність журналів Windows, ефективність їх використання у розслідуванні інцидентів часто обмежується складністю ручного аналізу. У великих інформаційних системах кількість подій може досягати десятків або сотень тисяч записів за короткий період часу, що робить ручний перегляд практично неможливим. Додатковою проблемою є наявність великої кількості службових записів, які не мають прямого відношення до безпеки, однак перевантажують журнал та ускладнюють пошук критичних подій. Також складність аналізу підвищується через необхідність врахування часових залежностей, кореляції подій між різними журналами та інтерпретації технічних параметрів, що потребує досвіду і високої кваліфікації аналітика.

У зв’язку з цим автоматизація аналізу логів Windows є актуальним напрямом розвитку технологій розслідування інцидентів. Використання автоматизованих засобів дозволяє швидко фільтрувати події за визначеними критеріями, виявляти підозрілі шаблони, формувати часові ланцюги подій та створювати структуровані звіти для подальшого аналізу. Автоматизація сприяє скороченню часу реагування на інциденти, зменшенню людського фактору та підвищенню точності оцінки ситуації. Таким чином, застосування програмних інструментів для обробки Windows Event Logs, зокрема із використанням Python, дозволяє суттєво підвищити ефективність розслідування інцидентів та забезпечити підтримку управління інформаційною безпекою в організації.

Мова програмування Python широко застосовується у задачах кібербезпеки, цифрової форензики та аналізу логів. Її популярність пояснюється простотою синтаксису, наявністю великої кількості бібліотек, а також можливістю швидкого створення прототипів програмних рішень.

Python дозволяє реалізувати автоматизований збір та аналіз подій із журналів операційної системи Windows. Завдяки бібліотекам на кшталт pywin32 або вбудованим механізмам роботи з файлами та системними командами, можливе отримання даних із Windows Event Logs, їх фільтрація за часовими проміжками, ідентифікаторами подій (Event ID), рівнем критичності або іншими параметрами. Це дає змогу значно зменшити час, необхідний для первинного аналізу інциденту, та підвищити точність пошуку важливих подій серед великого обсягу інформації.

Крім збору інформації, Python також є зручним інструментом для подальшої обробки та структурування даних. Наприклад, використання бібліотеки pandas дозволяє представити журнальні записи у вигляді таблиць, здійснювати сортування, групування та статистичний аналіз. Це є важливим етапом при розслідуванні інцидентів, оскільки дає можливість визначити закономірності, повторювані події та потенційні індикатори компрометації (Indicators of Compromise, IoC). Окрім цього, Python підтримує генерацію звітів у форматах CSV, JSON або TXT, що дозволяє легко передавати результати аналізу іншим учасникам процесу реагування на інциденти.

Важливою перевагою Python є можливість реалізації алгоритмів сигнатурного та поведінкового аналізу на базовому рівні без використання складних платформ. Наприклад, можна створити набір правил, що визначають підозрілі події (масові невдалі входи, запуск підозрілих процесів, часті зміни системних налаштувань), та автоматично формувати список потенційних інцидентів для подальшого розслідування. Таким чином, Python може виступати як інструмент первинного аналізу, що забезпечує швидке виявлення аномалій та підготовку даних для більш глибокої форензичної експертизи.

Таким чином, використання Python у розслідуванні інцидентів інформаційної безпеки дозволяє автоматизувати процеси збору, фільтрації, аналізу та документування подій із Windows Event Logs (рис. 2).

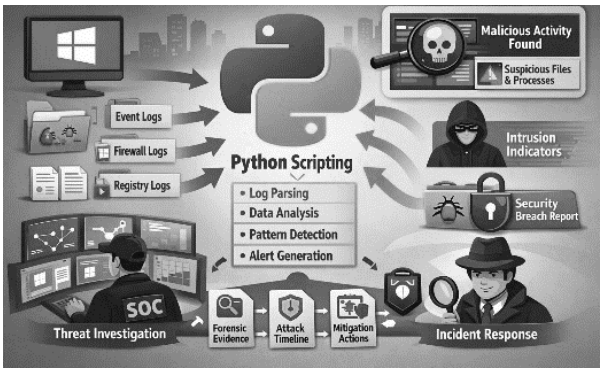

Рис. 2. Роль Python в розслідуванні інцидентів кібербезпеки

Це сприяє підвищенню ефективності роботи спеціалістів з кібербезпеки та забезпечує підтримку управлінських процесів у сфері інформаційної безпеки, зокрема при оцінці ризиків, контролі виконання політик та підготовці аналітичної звітності.

Для підвищення ефективності розслідування інцидентів кібербезпеки доцільним є застосування автоматизованого підходу, який дозволяє скоротити час аналізу подій та зменшити вплив людського фактору. Оскільки журнали подій операційної системи Windows містять значні обсяги інформації, їх ручна обробка є складною та малоефективною, особливо у випадках, коли інцидент потребує швидкого реагування та оперативного прийняття управлінських рішень.

З метою оптимізації процесу розслідування було запропоновано алгоритм автоматизації аналізу логів Windows, який забезпечує послідовний перехід від збору даних до формування підсумкових результатів. Запропонований підхід базується на поетапному виконанні ключових дій: отриманні подій із системних журналів, відборі потенційно небезпечних записів за визначеними критеріями та підготовці структурованого звіту для адміністратора або менеджера інформаційної безпеки.

Схематично даний процес представлений у вигляді алгоритму (рис. 3), що складається з трьох основних етапів: збір подій з логів Windows, фільтрація підозрілої активності та формування звітної інформації. Така структура дозволяє систематизувати процес розслідування інцидентів і забезпечити можливість подальшого масштабування рішення шляхом додавання нових правил аналізу або розширення джерел даних.

Реалізація даного алгоритму буде здійснена з використанням мови програмування Python, оскільки вона є зручним інструментом для автоматизованої обробки даних, підтримує роботу з системними компонентами Windows та дозволяє швидко формувати звіти у різних форматах. Використання бібліотек pywin32, pandas та datetime забезпечить можливість зчитування подій із Windows Event Logs, їх структурованої обробки та аналізу у часовому контексті. Таким чином, запропонований алгоритм стане основою практичної реалізації прототипу системи автоматизації розслідування інцидентів інформаційної безпеки на базі логів Windows.

Рис. 3. Алгоритм автоматизації розслідування інцидентів кібербезпеки

На рисунку представлено алгоритм автоматизації розслідування інцидентів кібербезпеки, який складається з трьох послідовних етапів. На першому етапі здійснюється збір подій із журналів операційної системи Windows, зокрема з основних категорій Security, System та Application. Цей етап забезпечує отримання первинних даних, які містять інформацію про активність користувачів, системні процеси та роботу програмного забезпечення.

Другий етап алгоритму передбачає фільтрацію підозрілої активності. На цьому кроці виконується відбір подій, які можуть свідчити про можливий інцидент інформаційної безпеки. До таких подій відносяться невдалі спроби входу, запуск підозрілих процесів, а також зміни в системі, які можуть бути наслідком несанкціонованих дій або втручання зловмисника.

Третій етап полягає у формуванні підсумкового звіту, який містить структуровану інформацію для адміністратора системи або менеджера інформаційної безпеки. Звіт дозволяє швидко оцінити ситуацію, визначити характер інциденту та прийняти управлінські рішення щодо реагування, документування або вдосконалення політик безпеки.

Таким чином, запропонований алгоритм демонструє просту, але ефективну модель автоматизованого підходу до розслідування інцидентів на основі логів Windows.

Для реалізації автоматизації аналізу логів Windows доцільно застосувати набір інструментів, що забезпечують зручний доступ до журналів подій, обробку даних та формування звітів. У межах даного дослідження обрано бібліотеки pywin32, pandas та модуль datetime, оскільки вони дозволяють реалізувати практичну частину швидко та з мінімальними затратами.

Бібліотека pywin32 є одним із основних засобів взаємодії Python із компонентами операційної системи Windows. Вона надає можливість працювати з Windows API, зокрема отримувати доступ до Windows Event Logs. За допомогою pywin32 можна зчитувати журнали Security, System та Application, отримувати дані про події, такі як час виникнення, Event ID, джерело та опис. Таким чином, pywin32 забезпечує базовий функціонал збору інформації, необхідної для подальшого розслідування інцидентів.

Бібліотека pandas використовується для структурованої обробки та аналізу отриманих подій. Вона дозволяє представляти журнальні записи у вигляді таблиць (DataFrame), здійснювати фільтрацію, сортування, групування та статистичний аналіз. Застосування pandas значно спрощує процес пошуку закономірностей у логах, а також дає змогу формувати результати у зручному форматі для подальшого документування або звітування.

Модуль datetime, який є стандартним компонентом Python, використовується для коректної роботи з часовими параметрами. Він дозволяє здійснювати фільтрацію подій за датою та часом, визначати часові інтервали між подіями, будувати хронологію інциденту та аналізувати активність користувачів у певні періоди. Часові характеристики є критично важливими для розслідування інцидентів, оскільки дозволяють відновити послідовність дій та встановити момент компрометації.

Отже, поєднання pywin32 (збір логів), pandas (аналітика та структуризація) та datetime (часова кореляція подій) формує ефективний набір інструментів для реалізації прототипу системи автоматизації розслідування інцидентів інформаційної безпеки на основі журналів Windows.

РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

Для ефективного виявлення підозрілої активності в комп’ютерній системі було розроблено алгоритм, який ґрунтується на аналізі логів операційної системи та поведінкових характеристиках користувача. Основна ідея полягає у виділенні аномалій, які відрізняються від стандартного профілю активності, що дозволяє своєчасно ідентифікувати потенційні загрози.

Для реалізації алгоритму обрано мову програмування Python через її потужний набір бібліотек для обробки даних та автоматизації аналізу логів:

- pywin32 – дозволяє взаємодіяти з системними журналами подій Windows, отримувати інформацію про запуски програм, помилки системи, входи та виходи користувачів.
- pandas – зручний інструмент для обробки та структуризації даних, включаючи фільтрацію, групування та обчислення статистичних показників.
- datetime – використовується для роботи з часовими мітками логів, визначення часових інтервалів та аналізу активності користувача у часі.

Цей набір інструментів дозволяє автоматизувати процес збору даних, їх обробку та подальший аналіз.

Для виявлення потенційно небезпечних подій встановлюються наступні критерії:

- Незвичний час активності – запуски програм або підключення до системи у незвичні години (наприклад, уночі).

- Відхилення від звичних патернів – значні відхилення у частоті або типі дій користувача у порівнянні зі стандартним профілем.
- Спроби доступу до критичних ресурсів – незвичні спроби відкриття або модифікації системних файлів, реєстру або облікових записів.
- Серійні помилки або аварійні події – повторювані системні помилки чи відмови, що можуть свідчити про спробу експлуатації вразливостей.

Кожен критерій оцінюється на основі логів, а події, які відповідають більшості критеріїв, отримують підвищений рівень ризику.

Алгоритм передбачає такі етапи (таблиця 1):

Таблиця 1

Основні етапи алгоритму аналізу логів операційної системи та поведінкових характеристиках користувача

Етап	Опис етапу
Збір даних	Автоматизоване отримання логів системи за допомогою pywin32
Обробка та нормалізація	Структуризація даних у таблиці за допомогою pandas, конвертація часових міток через datetime
Аналіз активності	Порівняння поточних дій користувача з історичними патернами та визначення відхилень
Оцінка ризику	Формування бальної системи, де кожне відхилення від стандартного профілю додає ризик до події.
Візуалізація та звітність	Підготовка таблиць та графіків для швидкого визначення підозрілих дій

На рис. 4 показано схематичний опис цього алгоритму.

Рис. 4. Опис реалізації алгоритму та формування критеріїв підозрілої активності

У підсумку, алгоритм забезпечує автоматичне відстеження підозрілої активності та формує структуровані дані, які можна використовувати для подальшого реагування на загрози.

Для реалізації алгоритму необхідно встановити потрібні бібліотеки. Для цього в Windows Power Shell використовуємо команду python -m pip install pandas matplotlib. Запуститься процес встановлення бібліотек (рис. 5).

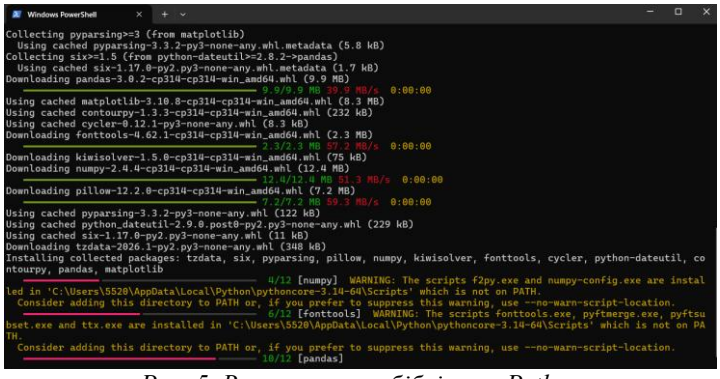

Рис. 5. Встановлення бібліотек Python

ISSN 2663 – 4023

Для дослідження ризиків у кібербезпеці ми розробили програму на Python, яка має назву suspicious_events.py та автоматично аналізує події в логах системи та оцінює їхню потенційну небезпеку (рис. 6). Основна мета цього коду – спростити процес виявлення підозрілих дій і представити результати у наочному вигляді.

Код виконує такі основні функції:

- Збір та обробка логів: програма приймає дані про події у системі (логіни, доступ до файлів, зміни привілеїв тощо) і структурує їх у вигляді таблиці для подальшого аналізу.
- Формування критеріїв підозрілості: кожна подія оцінюється за рівнем ризику на основі встановлених правил. Наприклад, багаторазові невдалі спроби входу або доступ до критичних файлів отримують високі оцінки ризику.
- Агрегація подій: однакові або схожі події групуються, щоб уникнути дублювання, і обчислюється максимальний ризик для кожного типу події.
- Візуалізація результатів: за допомогою бібліотеки matplotlib програма будує графік ризиків підозрілих подій. Це дозволяє швидко визначити найбільш критичні події, що потребують уваги, і наочно представити їх для аналізу або доповіді.

Використання Python для цього аналізу забезпечує автоматизацію процесу, економить час аналітиків і робить оцінку ризиків інформаційної безпеки більш прозорою та наочною. Крім того, код легко модифікувати під конкретні потреби системи або додавати нові критерії оцінки ризику.

Рис. 6. Фрагмент коду suspicious_events.py

Для запуску розробленого додатку для аналізу, скористаємося командою python suspicious_events.py.

Результатом виконання даного додатку є графік результатів проведеного аналізу (рис. 7).

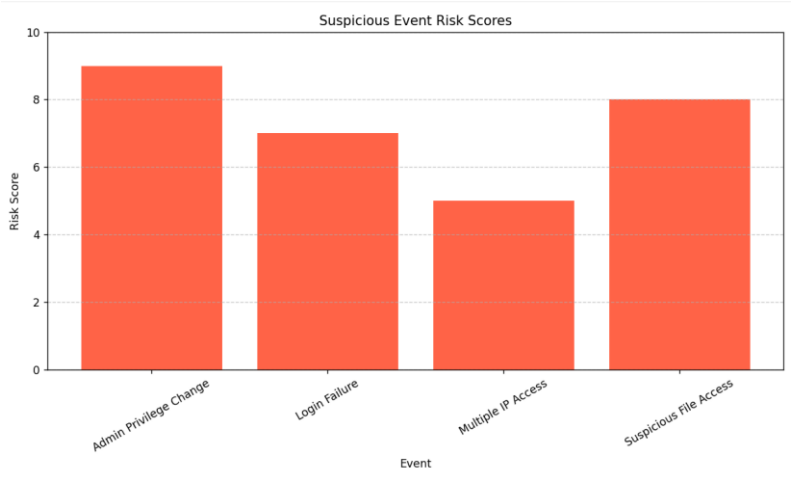

Рис. 7. Фрагмент коду suspicious_events.py

На графіку зображені підозрілі події в системі та їх оцінка ризику:
Вісь X (Event) – типи подій, які аналізувала програма:
Admin Privilege Change – зміни адміністративних прав

Login Failure – невдалі спроби входу

Multiple IP Access – доступ до системи з кількох IP

Suspicious File Access – підозрілі доступи до файлів

Ось Y (Risk Score) – оцінка ризику події за шкалою від 1 до 10. Вона показує, наскільки потенційно небезпечна або підозріла кожна подія.

Висота стовпчиків показує рівень ризику:

Найвищий ризик (9) має Admin Privilege Change – зміни адміністративних прав можуть сильно впливати на безпеку системи.

Suspicious File Access оцінено у 8, тобто теж критично.

Login Failure – 7, що показує потенційні спроби несанкціонованого входу.

Multiple IP Access – 5, нижчий ризик, але все одно підозрілий.

Колір стовпчиків (червоний) виділяє потенційно небезпечні події, роблячи графік наочним і легким для швидкого аналізу.

Отже, рафік дозволяє швидко побачити, які події в системі є найбільш ризиковими і потребують першочергового реагування, що робить аналіз логів більш ефективним та зручним для кібербезпеки.

ВИСНОВКИ ТА ПЕРСПЕКТИВИ ПОДАЛЬШИХ ДОСЛІДЖЕНЬ

Отриманий графік ризиків підозрілих подій демонструє розподіл потенційної небезпеки для різних типів подій у системі. З точки зору менеджменту інформаційної безпеки, результати дозволяють оцінити пріоритети реагування та побудувати ефективну стратегію контролю.

Події з високим ризиком:

Admin Privilege Change (ризик 9) та Suspicious File Access (ризик 8) демонструють критичний рівень загрози. Вони можуть призвести до несанкціонованого доступу до системи, зміни важливих налаштувань або витоку конфіденційної інформації.

У практичному менеджменті ці події потребують негайного реагування: моніторингу в реальному часі, обмеження прав доступу, аудит дій користувачів з адміністративними привілеями.

Події середнього ризику:

Login Failure (ризик 7) сигналізує про спроби несанкціонованого входу.

Для менеджерів інформаційної безпеки це означає необхідність налаштування політик блокування облікових записів після кількох невдалих спроб, використання багатофакторної аутентифікації та регулярного аналізу логів доступу.

Події з низьким ризиком:

Multiple IP Access (ризик 5) відображає доступ до системи з різних IP-адрес. Хоча ризик відносно нижчий, це може бути ознакою потенційного компрометаційного сценарію або використання VPN/проксі.

У менеджменті інформаційної безпеки такі події слід контролювати періодично, але вони не потребують негайних заходів.

Загальну інтерпретацію отриманих результатів для менеджменту інформаційної безпеки можна представити у вигляді таблиці 2.

Таблиця 2

Вплив результатів аналізу на менеджмент інформаційної безпеки

Етап	Опис етапу
Пріоритетність реагування	Графік дозволяє швидко визначити критичні події, які потребують першочергової уваги. Це економить ресурси та час аналітиків.
Візуальна підтримка рішень	Наявність графіка робить ризики наочними для керівництва та дозволяє аргументовано планувати заходи безпеки.
Профілактика та аудит	Регулярний аналіз логів і автоматизоване формування оцінки ризику дозволяє менеджерам інформаційної безпеки швидко виявляти тенденції загроз і коригувати політики доступу.
Інтеграція у SIEM	Подібні оцінки можуть стати частиною автоматизованої системи управління безпекою (SIEM), що підвищує ефективність моніторингу та реагування на інциденти.

Результати коду та графічна візуалізація дозволяють менеджерам інформаційної безпеки системно оцінювати ризики, визначати пріоритети реагування та приймати обґрунтовані управлінські рішення для захисту інформаційних систем.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Alzu'bi, A., Darwish, O., Albashayreh, A., & Tashtoush, Y. (2025). Cyberattack event logs classification using deep learning with semantic feature analysis. Computers & Security, 150, 104222. https://doi.org/10.1016/j.cose.2024.104222
2. Popov, O., et al. (2018). Conceptual approaches for development of informational and analytical expert system for assessing the NPP impact on the environment. Nuclear and Radiation Safety, 3(79), 56-65. https://doi.org/10.32918/nrs.2018.3(79).09
3. Muthusamy, P., Shanmugam, V., Kapilsurya, R., & Saran Kumar, R. (2024). Python-based security operations center (SOC) and forensics analysis for incident cyber threats. International Journal for Research in Applied Science and Engineering Technology. https://doi.org/10.22214/ijraset.2024.60403
4. Wikipedia contributors. (n.d.). Log analysis. Wikipedia. https://en.wikipedia.org/wiki/Log_analysis
5. VPN Unlimited. (n.d.). Malware: Definition and types of malicious software. https://www.vpnunlimited.com/ua/help/cybersecurity/malware
6. Holt, W., Dawson, R., & Agoro, H. (2021). Development of an automated digital forensics toolkit for incidentresponse.https://www.researchgate.net/publication/389815869_Development_of_an_Automated_Digital_Forensics_Toolkit_for_Incident_Response
7. Havrysh, B. M., Tymchenko, O. V., Borzov, Y. O., & Kobevko, A. T. (2022). Classification of malicious software and main protection methods. Computer Technologies of Printing, 2(48), 142-154. https://sci.ldubgd.edu.ua/jspui/handle/123456789/12981
8. Kytsiuk, V. M., & Pupynin, O. S. (2024). Enterprise information security: Theoretical aspect. Modern Information Protection, (2), 103-108
9. Polotai, O. I. (2023). Use of computer forensics to ensure effective investigation of information and cybersecurity incidents. Bulletin of Lviv State University of Life Safety, 28, 73-80. https://doi.org/10.32447/20784643.28.2023.07
10. Tkachuk, R. L., Polotai, O. I., Balatska, V. S., Brych, T. B., & Kukharska, N. P. (2025). Modeling protection of operating systems against cyberattacks using Pearson's criterion. Bulletin of Lviv State University of Life Safety, 31, 117-125. https://doi.org/10.32447/20784643.31.2025.12
11. Tsyrkaniuk, D., & Sokolov, V. (2024). Methodology for investigating information security incidents. Cybersecurity: Education, Science, Technique, 2(26), 140-154. https://doi.org/10.28925/2663-4023.2024.26.675

КІБЕРБЕЗПЕКА: освіта, наука, техніка
CYBERSECURITY: EDUCATION, SCIENCE, TECHNIQUE
№ 1 (33), 2026
ISSN 2663 – 4023

Orest Polotai
Candidate of Technical Sciences, Associate Professor, Associate Professor of the Department of Information Security Management
Lviv State University of Life Safety, Lviv, Ukraine
ORCID: 0000-0003-4593-8601
orest.polotaj@gmail.com

Natalia Kukharska
Candidate of Physical and Mathematical Sciences, Associate Professor, Associate Professor of the Department of Information Technology Security
Lviv Polytechnic National University, Lviv, Ukraine
ORCID: 0000-0002-0896-8361
kukharska.n@gmail.com

Artur Tkachenko
Lecturer of the Department of Information Security Management
Lviv State University of Life Safety, Lviv, Ukraine
ORCID: 0009-0009-6830-4741
tkachenko.am14@gmail.com

Ievgeny Siedin
Senior lecturer of the Department of Information Security Management
Lviv State University of Life Safety, Lviv, Ukraine
ORCID: 0009-0004-9654-0577
ye.sedin@ldubgd.edu.ua

Maxim Nykolaichuk
Lecturer of the Department of Information Security Management
Lviv State University of Life Safety, Lviv, Ukraine
ORCID: 0009-0004-6313-5196
nykolaychuk.maksym.i@ldubgd.edu.ua

METHODS FOR AUTOMATING CYBERSECURITY INCIDENT INVESTIGATION BASED ON WINDOWS OPERATING SYSTEM LOGS USING PYTHON TO SUPPORT INFORMATION SECURITY MANAGEMENT

Abstract. The article presents an approach to the analysis and visualization of information security risks based on the processing of system and network logs using automation software. An algorithm in Python has been developed that provides collection, structuring and analysis of events occurring in the operating system and network devices in order to detect potentially suspicious activity. Pandas and datetime libraries were used for data processing, which allow for efficient work with large amounts of information and time stamps, and matplotlib was used to visualize the results, which provides a visual representation of patterns and anomalies. The algorithm classifies events according to certain criteria of suspicious activity, taking into account their type, frequency and time characteristics. The resulting graphical models allow assessing the level of risk in different segments of the system and making informed management decisions regarding information security. An experimental verification of the algorithm was carried out using real logs, which confirmed its effectiveness in early detection of anomalous behavior and optimization of monitoring processes. The results of the study emphasize the importance of integrating log analysis and data visualization methods into modern information security management systems. The use of automation software helps minimize the human factor, increase the accuracy of risk assessment and the efficiency of responding to threats. The article has practical and scientific significance, as it offers a methodology for building an effective monitoring system and early warning of cybersecurity incidents.

Keywords: operating systems, information security, cyber threats, cybersecurity incidents, programming, information security management, logging, computer forensics, information security risks.

REFERENCES (TRANSLATED AND TRANSLITERATED)

1. Alzu'bi, A., Darwish, O., Albashayreh, A., & Tashtoush, Y. (2025). Cyberattack event logs classification using deep learning with semantic feature analysis. Computers & Security, 150, 104222. https://doi.org/10.1016/j.cose.2024.104222
2. Popov, O., et al. (2018). Conceptual approaches for development of informational and analytical expert system for assessing the NPP impact on the environment. Nuclear and Radiation Safety, 3(79), 56-65. https://doi.org/10.32918/nrs.2018.3(79).09
3. Muthusamy, P., Shanmugam, V., Kapilsurya, R., & Saran Kumar, R. (2024). Python-based security operations center (SOC) and forensics analysis for incident cyber threats. International Journal for Research in Applied Science and Engineering Technology. https://doi.org/10.22214/ijraset.2024.60403
4. Wikipedia contributors. (n.d.). Log analysis. Wikipedia. https://en.wikipedia.org/wiki/Log_analysis
5. VPN Unlimited. (n.d.). Malware: Definition and types of malicious software. https://www.vpnunlimited.com/ua/help/cybersecurity/malware
6. Holt, W., Dawson, R., & Agoro, H. (2021). Development of an automated digital forensics toolkit for incidentresponse.https://www.researchgate.net/publication/389815869_Development_of_an_Automated_Digital_Forensics_Toolkit_for_Incident_Response
7. Havrysh, B. M., Tymchenko, O. V., Borzov, Y. O., & Kobevko, A. T. (2022). Classification of malicious software and main protection methods. Computer Technologies of Printing, 2(48), 142-154. https://sci.ldubgd.edu.ua/jspui/handle/123456789/12981
8. Kytsiuk, V. M., & Pupynin, O. S. (2024). Enterprise information security: Theoretical aspect. Modern Information Protection, (2), 103-108
9. Polotai, O. I. (2023). Use of computer forensics to ensure effective investigation of information and cybersecurity incidents. Bulletin of Lviv State University of Life Safety, 28, 73-80. https://doi.org/10.32447/20784643.28.2023.07
10. Tkachuk, R. L., Polotai, O. I., Balatska, V. S., Brych, T. B., & Kukharska, N. P. (2025). Modeling protection of operating systems against cyberattacks using Pearson's criterion. Bulletin of Lviv State University of Life Safety, 31, 117-125. https://doi.org/10.32447/20784643.31.2025.12
11. Tsyrkaniuk, D., & Sokolov, V. (2024). Methodology for investigating information security incidents. Cybersecurity: Education, Science, Technique, 2(26), 140-154. https://doi.org/10.28925/2663-4023.2024.26.675

Отримано редакцією журналу / Received: 16.02.26
Прорецензовано / Revised: 28.02.26
Схвалено до друку / Accepted: 25.06.26