

Міністерство освіти і науки України
Державний університет «Житомирська політехніка»
Факультет інформаційно-комп'ютерних технологій
Кафедра комп'ютерної інженерії та кібербезпеки

ПОЯСНЮВАЛЬНА ЗАПИСКА

до випускної кваліфікаційної роботи магістра
на тему: «Методи та технології аналізу підозрілої активності в ІТ-інфраструктурі»

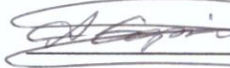
Виконав: студент 2-го курсу групи КБм-24-1
спеціальності 125 «Кібербезпека та захист інформації»

 Олександр ЗАРИЦЬКИЙ

Керівник: доцент кафедри КІ та КБ, к.пед.н., доцент

 Олена ГОЛОВНЯ

Рецензент: завідувач кафедри КІ та КБ, к.т.н., доцент

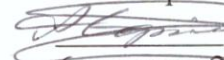
 Андрій ЄФІМЕНКО

Державний університет «Житомирська політехніка»
Факультет інформаційно-комп'ютерних технологій
Кафедра комп'ютерної інженерії та кібербезпеки
Спеціальність 125 «Кібербезпека та захист інформації»
Освітня програма «Кібербезпека»

ЗАТВЕРДЖУЮ

Завідувач

кафедри комп'ютерної
інженерії та кібербезпеки

 Андрій ЄФІМЕНКО
6 жовтня 2025 р.

ЗАВДАННЯ

на випускню кваліфікаційну роботу магістра

Студент: Зарицький Олександр Васильович

Тема роботи: «Методи та технології аналізу підозрілої активності в ІТ-інфраструктурі»

затверджена Наказом університету від 6 жовтня 2025 р. №566/с

Термін здачі студентом закінченої роботи 16 грудня 2025 р.

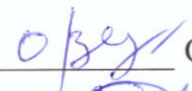
Вихідні дані роботи: об'єктом дослідження є процес виявлення та аналізу підозрілої активності в ІТ-інфраструктурі; предметом дослідження є методи та технології аналізу подій безпеки на хостовому та мережевому рівнях з використанням систем моніторингу та виявлення вторгнень.

Обладнання та програмне забезпечення:

1. VirtualBox
2. Ubuntu Desktop
3. Windows 11
4. Kali Linux
5. Wazuh

Консультанти з випускної кваліфікаційної роботи із зазначенням розділів, що їх стосуються

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв
1	Олександра ПОКОТИЛО	1.10.2025 	1.10.2025 
2	Олександра ПОКОТИЛО	22.11.2025 	22.11.2025 
3	Олександра ПОКОТИЛО	22.11.2025 	22.11.2025 

Студент  Олександр ЗАРИЦЬКИЙ

Керівник  Олена ГОЛОВНЯ

Календарний план

№ з/п	Назви етапів випускної кваліфікаційної роботи	Термін виконання етапів роботи	Примітка
1	Постановка завдання	1.10.2025	Виконано
2	Дослідження сучасних кіберзагроз	15.10.2025	Виконано
3	Аналіз методів і технологій виявлення підозрілої активності	19.10.2025 – 31.10.2025	Виконано
4	Огляд та порівняння існуючих рішень для аналізу подій безпеки	1.11.2025 – 8.11.2025	Виконано
5	Формування вимог до системи аналізу підозрілої активності	14.11.2025 – 22.11.2025	Виконано
6	Проектування архітектури та логічної моделі системи	22.11.2025 – 26.11.2025	Виконано
7	Вибір програмних компонентів і технологій реалізації	27.11.2025	Виконано
8	Розгортання та тестування системи та аналіз виявлених інцидентів	1.12.2025 – 6.12.2025	Виконано
9	Оформлення пояснювальної записки та підготовка до захисту	7.12.2025 – 15.12.2025	Виконано
10	Захист кваліфікаційної роботи		Виконано

Студент  Олександр ЗАРИЦЬКИЙ

Керівник  Олена ГОЛОВНЯ

АНОТАЦІЯ

Кваліфікаційна робота присвячена дослідженню методів виявлення та аналізу підозрілої активності в ІТ-інфраструктурі, а також обґрунтуванню вибору технологій і програмних засобів для моніторингу подій безпеки. У роботі розглянуто сучасні підходи до організації систем моніторингу та реагування на інциденти інформаційної безпеки з використанням відкритих програмних рішень.

Текстова частина роботи викладена на 69 сторінках друкованого тексту. Пояснювальна записка містить 2 сторінки додатків. Список використаних джерел налічує 29 найменувань і займає 3 сторінки. У роботі наведено 50 рисунків та 6 таблиць. Загальний обсяг кваліфікаційної роботи становить 71 сторінка.

Метою кваліфікаційної роботи є дослідження методів і технологій аналізу підозрілої активності в ІТ-інфраструктурі, а також проектування, розгортання та тестування системи моніторингу подій безпеки для виявлення та аналізу кіберінцидентів на хостовому та мережевому рівнях.

У межах виконання кваліфікаційної роботи проведено аналіз сучасних загроз інформаційній безпеці та методів їх виявлення на хостовому і мережевому рівнях. Розглянуто принципи побудови систем моніторингу подій безпеки, проаналізовано можливості таких інструментів, як Wazuh, Suricata та допоміжних засобів аналізу мережевої і файлової активності. Спроектовано логічну архітектуру системи моніторингу, реалізовано її розгортання та налаштування клієнтських компонентів.

Запропоноване рішення може бути використане для підвищення рівня захищеності інформаційних ресурсів організацій та є актуальним у контексті зростання кількості кіберзагроз і складності сучасних атак.

КЛЮЧОВІ СЛОВА: ПІДОЗРІЛА АКТИВНІСТЬ, ІНФОРМАЦІЙНА БЕЗПЕКА, МОНІТОРИНГ ПОДІЙ, SIEM, WAZUH, SURICATA, IDS, MITRE ATT&CK, ІТ-ІНФРАСТРУКТУРА.

ANNOTATION

The thesis is devoted to the study of methods for detecting and analysing suspicious activity in IT infrastructure, as well as justifying the choice of technologies and software tools for monitoring security events. The thesis considers modern approaches to the organisation of monitoring and response systems for information security incidents using open source software solutions.

The text part of the work is presented on 69 pages of printed text. The explanatory note contains 2 pages of appendices. The list of references includes 29 titles and occupies 3 pages. The work contains 50 figures and 6 tables. The total volume of the qualification work is 71 pages.

The aim of the work is to design and implement a system for detecting suspicious activity in IT infrastructure using modern security event monitoring tools to improve information security.

As part of the qualification work, an analysis of modern threats to information security and methods for detecting them at the host and network levels was carried out. The principles of building security event monitoring systems were considered, and the capabilities of tools such as Wazuh, Suricata, and auxiliary means of analysing network and file activity were analysed. The logical architecture of the monitoring system was designed, and its deployment and configuration of client components were implemented.

The proposed solution can be used to improve the security of organisations' information resources and is relevant in the context of the growing number of cyber threats and the complexity of modern attacks.

KEYWORDS: SUSPICIOUS ACTIVITY, INFORMATION SECURITY, EVENT MONITORING, SIEM, WAZUH, SURICATA, IDS, MITRE ATT&CK, IT INFRASTRUCTURE.

ЗМІСТ

ВСТУП	8
РОЗДІЛ 1 АНАЛІЗ ПІДОЗРІЛОЇ АКТИВНОСТІ В ІТ-ІНФРАСТРУКТУРІ ТА ОГЛЯД ІСНУЮЧИХ РІШЕНЬ	10
1.1 Поняття підозрілої активності та кіберзагроз	10
1.2 Методи виявлення підозрілої активності.....	12
1.3 Підходи та засоби аналізу подій безпеки в ІТ-інфраструктурі	18
1.3.1 IDS/IPS системи (Snort, Suricata).....	18
1.3.2 SIEM-системи.....	20
1.3.3 Засоби моніторингу мережі	27
1.3.4 Системи логування	29
1.4 Огляд інтегрованих та спеціалізованих рішень для аналізу підозрілої активності	31
РОЗДІЛ 2 ПРОЄКТУВАННЯ СИСТЕМИ АНАЛІЗУ ПІДОЗРІЛОЇ АКТИВНОСТІ В ІТ-ІНФРАСТРУКТУРІ.....	38
2.1 Загальна концепція та архітектура системи	38
2.2 Логічна модель мережевої інфраструктури.....	41
2.3 Обґрунтування вибору технологій та програмних компонентів.....	43
РОЗДІЛ 3 ДОСЛІДЖЕННЯ РОБОТИ ТА ТЕСТУВАННЯ СИСТЕМИ АНАЛІЗУ ПІДОЗРІЛОЇ АКТИВНОСТІ.....	47
3.1 Розгортання системи моніторингу та підключення клієнтів	47
3.2 Тестування виявлення підозрілої активності на різних рівнях	53
3.3 Аналіз виявлених інцидентів з використанням MITRE ATT&CK	63
ВИСНОВКИ.....	67
СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ	68
ДОДАТКИ.....	71

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

ІБ – інформаційна безпека

ІТ – інформаційні технології

ОС – операційна система

ПЗ – програмне забезпечення

HIDS (Host-based Intrusion Detection System) – хостова система виявлення вторгнень

IDS (Intrusion Detection System) – система виявлення вторгнень

IOA (Indicator of Attack) – індикатор атаки

IOC (Indicator of Compromise) – індикатор компрометації

IPS (Intrusion Prevention System) – система запобігання вторгненням

MITRE ATT&CK – база знань тактик і технік атак зловмисників

NIDS (Network-based Intrusion Detection System) – мережева система виявлення вторгнень

SIEM (Security Information and Event Management) – система управління подіями та інформацією безпеки

SOAR (Security Orchestration, Automation and Response) – оркестрація, автоматизація та реагування на інциденти безпеки

SOC (Security Operations Center) – центр моніторингу та реагування на інциденти безпеки

TTP (Tactics, Techniques and Procedures) – тактики, техніки та процедури зловмисників

UEBA (User and Entity Behavior Analytics) – аналіз поведінки користувачів і сутностей

WAF (Web Application Firewall) – міжмережевий екран вебзастосунків

ВСТУП

В умовах стрімкої цифровізації та зростання ролі інформаційних технологій у діяльності організацій питання забезпечення кібербезпеки ІТ-інфраструктур набуває особливої актуальності. Сучасні комп'ютерні мережі та інформаційні системи постійно зазнають впливу різноманітних кіберзагроз, зокрема несанкціонованого доступу, атак типу brute-force, мережевого сканування, шкідливого програмного забезпечення та внутрішніх порушень політик безпеки. Значна частина таких загроз проявляється у вигляді підозрілої активності, яка не завжди однозначно ідентифікується традиційними засобами захисту.

Актуальність теми зумовлена необхідністю своєчасного виявлення, аналізу та кореляції подій безпеки в ІТ-інфраструктурі з метою мінімізації ризиків інформаційної безпеки. Зростання обсягів мережевого трафіку, кількості логів та складності атак ускладнює ручний аналіз інцидентів, що обґрунтовує доцільність використання спеціалізованих систем моніторингу, таких як IDS/IPS, SIEM-платформи, засоби аналізу мережевого трафіку та централізованого логування. У межах даної роботи розглядається проблема комплексного аналізу підозрілої активності з використанням сучасних методів і технологій, що дозволяє підвищити рівень захищеності інформаційних ресурсів.

Метою кваліфікаційної роботи є дослідження методів і технологій аналізу підозрілої активності в ІТ-інфраструктурі, а також проєктування, розгортання та тестування системи моніторингу подій безпеки для виявлення та аналізу кіберінцидентів на хостовому та мережевому рівнях.

Об'єктом дослідження є процес забезпечення інформаційної безпеки ІТ-інфраструктури шляхом моніторингу та аналізу подій безпеки.

Предметом дослідження є методи, засоби та технології виявлення і аналізу підозрілої активності, зокрема IDS/IPS-системи, SIEM-платформи, засоби мережевого моніторингу, системи логування та підходи до кореляції подій безпеки.

Визначена мета обумовлює необхідність розв'язання таких завдань:

- проаналізувати поняття підозрілої активності та основні типи кіберзагроз в ІТ-інфраструктурі;
- дослідити методи виявлення підозрілої активності на хостовому та мережевому рівнях;
- здійснити огляд та порівняльний аналіз сучасних інструментів і платформ аналізу подій безпеки;
- спроектувати логічну модель системи аналізу підозрілої активності в ІТ-інфраструктурі;
- реалізувати тестове середовище з використанням клієнтських та серверних компонентів системи моніторингу;
- провести тестування роботи системи шляхом імітації підозрілої активності та проаналізувати виявлені інциденти з використанням моделі MITRE ATT&CK.

У процесі виконання кваліфікаційної роботи було **розроблено та досліджено проєкт системи аналізу підозрілої активності в ІТ-інфраструктурі**, що включає сервер моніторингу та клієнтські хости. Реалізована система дозволяє здійснювати збір та аналіз подій безпеки, а також оцінювати ефективність виявлення різних типів підозрілої активності в умовах змодельованої мережевої взаємодії.

РОЗДІЛ 1 АНАЛІЗ ПІДОЗРІЛОЇ АКТИВНОСТІ В ІТ-ІНФРАСТРУКТУРІ ТА ОГЛЯД ІСНУЮЧИХ РІШЕНЬ

1.1 Поняття підозрілої активності та кіберзагроз

Сучасні ІТ-інфраструктури складаються з великої кількості взаємопов'язаних компонентів, серед яких серверні системи, робочі станції, мережеве обладнання, хмарні сервіси, мобільні пристрої та спеціалізовані IoT-рішення. Зростання масштабів та розгалуженості таких інфраструктур призводить до збільшення їхньої складності, а отже і до підвищення рівня кіберризиків. Кожен новий сервіс або вузол може стати потенційним вектором атаки, а швидка еволюція загроз робить традиційні підходи до забезпечення безпеки недостатніми.

За останні роки статистика інцидентів лише підтверджує зростання рівня ризиків. Наприклад, за даними державних органів України, у 2023 році було зафіксовано тисячі кіберінцидентів, причому їхня кількість зросла порівняно з 2022 роком; Служба безпеки України повідомляє про понад 4500 кіберзагроз, тоді як інші органи кіберзахисту фіксують загальне збільшення атак на державний та приватний сектор. CERT-UA, урядова команда реагування на комп'ютерні надзвичайні події України, лише за 2024 рік опрацювала 4315 кіберінциденти, серед яких домінували розповсюдження шкідливого ПЗ, фішинг, компрометація облікових записів і систем.[1]

У цьому контексті особливого значення набуває поняття *підозрілої активності*, яке є проміжною ланкою між нормальним функціонуванням системи та підтвердженим інцидентом інформаційної безпеки. Виявлення таких подій дозволяє зупинити атаку на ранньому етапі, ще до того, як вона перейде у фазу реальної компрометації ресурсів. Підозріла активність не завжди означає безпосередній злам, але завжди вказує на відхилення від звичного профілю роботи системи, які не можуть залишатися без уваги.

Для розуміння цього явища доречно розмежувати три базові поняття: *нормальна діяльність, підозріла активність та інцидент інформаційної безпеки*. Нормальна діяльність визначається характерною, типовою поведінкою користувачів, процесів, сервісів і мережевого трафіку. Інцидент, навпаки, є фактом реалізації загрози – тобто подією, яка призвела або могла призвести до порушення конфіденційності, цілісності чи доступності інформації. Підозріла активність займає проміжний стан: вона відхиляється від «норми», але не має достатніх ознак для однозначної класифікації як інцидент.

Підозріла активність може проявлятися у різних формах. Найчастіше вона включає спроби несанкціонованого доступу, нетипові операції автентифікації, розвідувальну активність у мережі, аномальні обсяги трафіку, запуск невідомих процесів, модифікацію конфігураційних файлів, звернення до сторонніх ресурсів або інші дії, які не відповідають нормальній поведінці системи. Ці ознаки не обов'язково означають інцидент, проте саме вони є першими сигналами, на основі яких формуються механізми виявлення атак.

На рівні мережевої взаємодії підозрілою активністю можуть бути масові спроби встановлення з'єднань, сканування портів, використання протоколів нетипових для конкретної мережі або різке збільшення вихідного чи вхідного трафіку. Такі події часто передують складнішим загрозам, включно з розповсюдженням шкідливого ПЗ, експлуатацією вразливостей або витоком даних.

На рівні користувачів прояви підозрілої активності можуть включати спроби входу з нових або географічно нетипових локацій, часті невдалі логіни, звернення до систем, які не входять до зони відповідальності користувача, або виконання масових дій, що не характерні їхній ролі. На цьому рівні важливим є поведінковий аналіз – відстеження відхилення від звичного робочого ритму конкретної особи або групи користувачів.

Окрему категорію підозрілої активності становлять аномалії на рівні операційної системи та прикладних сервісів, зокрема створення нових служб, встанов-

лення програм без відома адміністратора, зміна системних бібліотек, запуск невідомих процесів або модифікація критичних гілок реєстру, що часто свідчить про зараження шкідливим ПЗ чи закріплення зловмисника в системі. Аналогічно, нетипова поведінка прикладних сервісів — незвичні запити до вебдодатків, зростання кількості помилок авторизації, аномальне навантаження на API або використання параметрів, характерних для експлуатації вразливостей, — може бути раннім індикатором підготовки до атак, таких як SQL-ін'єкції або виконання довільного коду.

Ключова складність у роботі з підозрілою активністю полягає в тому, що вона завжди є багаторівневим і контекстно залежним явищем. Для її коректного аналізу необхідно одночасно враховувати події з різних джерел: журналів операційних систем, мережевого трафіку, поведінки користувачів, логів застосунків, а також зовнішніх даних про загрози. Саме тому сучасні системи моніторингу безпеки – зокрема SIEM та HIDS – створюються як комплексні рішення, що дозволяють поєднувати інформацію з багатьох каналів у єдину картину.

Практика розслідування інцидентів підтверджує, що ігнорування ранніх ознак підозрілої активності часто стає причиною масштабних атак. У більшості випадків перед серйозним порушенням відбуваються дрібні аномалії – одиничні незвичні запити, окремі невдалі спроби входу, незначні зміни у поведінці процесів. Саме тому надійні системи безпеки повинні бути здатні не лише фіксувати критичні події, а й аналізувати малопомітні відхилення, які можуть бути частиною більшої загрози.

1.2 Методи виявлення підозрілої активності

При аналізі підозрілої активності в IT-інфраструктурі застосовується кілька підходів, які доповнюють один одного. Загалом виділяють два основні типи методів: сигнатурні (на основі відомих шаблонів атак) та поведінкові (аналітика аномалій). Сигнатурний аналіз виявляє загрози шляхом пошуку у подіях характерних ознак відомих атак, тоді як поведінкові методи зосереджені на відхиленнях у нормальній діяльності систем і користувачів. Надалі буде розглянуто ключові підходи до

виявлення підозрілої активності: сигнатурний аналіз (на прикладі IDS типу Snort/Suricata), евристичні методи на основі поведінкового аналізу (аналіз активності користувачів, процесів, трафіку), статистичні методи та машинне навчання (включно з технологією UEBA), а також кореляція подій у системах класу SIEM (наприклад, ELK, Splunk, Wazuh).

Сигнатурний аналіз

Сигнатурний метод виявлення атак полягає у пошуку в досліджуваних подіях ознак, що точно відповідають записаним у базі даних сигнатурам відомих атак або шкідливих дій. Система виявлення вторгнень (IDS, Intrusion Detection System) типу Snort або Suricata, аналізуючи мережевий трафік, співставляє його з набором відомих сигнатур і при знаходженні збігу формує сповіщення про потенційну загрозу. Ефективність сигнатурного методу залежить від актуальності бази даних: для розпізнавання нових або модифікованих атак необхідно оперативно додавати їхні «відбитки» (сигнатури) до бази, інакше такі загрози можуть лишитися непоміченими. Постійне оновлення бази є водночас сильним боком методу (забезпечує актуальність) і його слабким місцем, адже до появи відповідної сигнатури новий напад не буде виявлено[2].

Основна перевага сигнатурного аналізу – висока точність і швидкодія при виявленні вже відомих типів атак. Система майже миттєво розпізнає шаблон атаки, що міститься у базі, що дозволяє оперативно реагувати на загрозу. При сигнатурному моніторингу також спостерігається низький рівень хибнопозитивних спрацювань, оскільки детектор орієнтується на чітко визначені шаблони і не помилково трактує легітимну активність як атаку. Водночас істотний недолік цього підходу – нездатність виявити нові, ще невідомі атаки (так звані атаки нульового дня) або модифіковані загрози, що не мають записаної сигнатури. Зловмисник може мінімально змінити характер шкідливої активності, і система її не розпізнає[2]. Таким чином, для підтримання ефективності сигнатурні IDS потребують регулярного оновлення баз правил та оперативного доповнення їх новими даними про уразливості й атаки.

На практиці сигнатурний підхід є основою багатьох систем кіберзахисту. Зокрема, Snort – одна з найпоширеніших мережових IDS з відкритим кодом – вже понад два десятиліття залишається стандартом де-факто для сигнатурного аналізу мережового трафіку. Новіша система Suricata розвинула ідеї Snort, зберігши сумісність з його сигнатурами та додавши багатопоточну архітектуру для вищої продуктивності аналізу трафіку. Сигнатурні методи також застосовуються у системах запобігання вторгнень (IPS) та антивірусних сканерах, які ідентифікують шкідливе ПЗ за унікальними «цифровими відбитками» відомих вірусів (сигнатурами).

Евристичні методи поведінкового аналізу

На відміну від сигнатурного підходу, який орієнтується на відомі шаблони атак, поведінковий аналіз фокусується на виявленні аномалій у типових процесах функціонування системи. У цьому випадку визначальним є нетиповий характер активності, навіть за відсутності конкретної сигнатури загрози. Система формує профілі нормальної поведінки користувачів, хостів і мережі та відстежує відхилення від них. Наприклад, вхід користувача з нетипового географічного регіону або звернення до незвичних ресурсів може бути визначене як потенційно підозріла активність.

Евристичні методи зазвичай реалізуються у вигляді набору правил і порогових значень, які описують потенційно небезпечну поведінку. Замість пошуку точного збігу з відомим патерном система аналізує сукупність ознак, таких як незвично велика кількість операцій за короткий проміжок часу або нетипова послідовність дій. Поведінковий аналіз може застосовуватися на різних рівнях: користувачькому (виявлення компрометації облікових записів), хостовому (моніторинг процесів і системних подій) та мережевому. Окремо виділяють мережовий поведінковий аналіз (Network Behavior Analysis), що дозволяє виявляти аномальні патерни трафіку, зокрема сканування портів, сплески активності або ознаки DDoS-атак.

Основною перевагою поведінкових методів є здатність виявляти нові та нестандартні атаки, для яких ще не існує сигнатур. Водночас такі системи мають і суттєві обмеження: вони є ресурсомісткими та схильними до хибнопозитивних

спрацювань, особливо за неточного налаштування профілів «норми». Легітимна, але нетипова активність може бути помилково інтерпретована як загроза, що знижує ефективність моніторингу. Тому евристичні методи потребують регулярного тюнінгу й зазвичай застосовуються у поєднанні з сигнатурним аналізом, доповнюючи його та підвищуючи загальний рівень виявлення підозрілої активності.

Статистичні методи та машинне навчання (UEBA)

Сучасний розвиток поведінкового підходу до виявлення кіберзагроз пов'язаний із технологією UEBA (User and Entity Behavior Analytics – аналітика поведінки користувачів та об'єктів). Ця система використовує великі обсяги даних і алгоритми штучного інтелекту для побудови профілів нормальної поведінки та виявлення відхилень.

UEBA збирає дані з різноманітних джерел: журнали дій користувачів, системні логи, мережевий трафік, телеметрію. На етапі навчання вона формує нормальний шаблон поведінки кожного користувача чи пристрою – типові години активності, географія доступу, обсяги запитів тощо. Далі за допомогою статистичних моделей і алгоритмів машинного навчання UEBA ідентифікує аномалії, наприклад, раптове зростання активності або нетипові дії, які автоматично позначаються як потенційні загрози.[4]

Ці системи застосовують широкий спектр AI-алгоритмів – від простого профілювання до кластеризації та класифікації поведінкових патернів. На основі десятків показників – часу та локації активності, команд, що виконуються, взаємодії з іншими сервісами – формується оцінка ризику для кожного об'єкта. UEBA здатна виявляти тихі атаки, зловживання інсайдерів, латеральний рух атакувальника або прихований перебір паролів ще до того, як інцидент стане критичним.

Сильна сторона UEBA – автоматизований аналіз великих обсягів логів. Це знижує навантаження на аналітиків і дозволяє зосередитись на справді важливих сигналах. Проте впровадження UEBA – складний процес. Воно потребує значних обчислювальних ресурсів, часу на початкове навчання, кваліфікованого персоналу. На старті можливі помилкові тривоги. Крім того, система зазвичай лише виявляє

аномалії, але не зупиняє атаки, тож потрібна інтеграція з SOAR для автоматичної реакції. [4]

Кореляція подій у SIEM-системах

SIEM (Security Information and Event Management) – це клас систем, що здійснюють централізований збір і зберігання журналів подій безпеки з різних джерел та їх кореляційний аналіз у режимі реального часу. Однією з ключових функцій SIEM є саме кореляція подій – автоматизований аналіз взаємозв’язків між подіями з різних систем з метою виявлення ознак комплексних атак або порушень політик безпеки. SIEM-платформа отримує логи від мережевого обладнання, серверів, застосунків, баз даних, засобів безпеки (файрволів, IDS/IPS тощо) і застосовує набір правил та шаблонів, щоб знайти серед цього потоку даних індикатори відомих атак або аномальні комбінації подій. Приміром, система SIEM може зіставити серію розрізнених сигналів – спроб сканування портів на периферійному маршрутизаторі, появу нетипових записів у системному журналі сервера та незвичну активність адміністративного облікового запису – які в сукупності вказують на цілеспрямовану атаку, хоча окремо кожна з цих подій могла видатися незначною. [3]

Завдяки кореляції, SIEM надає аналітикам широкий контекстний огляд ситуації. Об’єднуючи розрізнені сигнали з багатьох джерел, система може виявити складні багатоступеневі атаки, які неможливо розпізнати, аналізуючи лише поодинокі журнали.

Водночас ефективність SIEM значною мірою залежить від коректного налаштування правил кореляції та якості обробки даних. Надмірно загальні правила можуть призводити до великої кількості хибних сповіщень, тоді як недостатнє покриття сценаріїв — до пропуску реальних атак. Крім того, SIEM-системи є ресурсомісткими та потребують потужної інфраструктури й кваліфікованого персоналу для впровадження та підтримки. Комерційні рішення (Splunk, IBM QRadar, ArcSight, Microsoft Sentinel) пропонують розширені аналітичні можливості, масштабованість і інтеграцію з розвідкою загроз, але мають високу вартість володіння. Відкриті рішення, зокрема Elastic Stack у поєднанні з Wazuh, забезпечують базові

SIEM-функції та гнучкість, проте потребують ручної інтеграції компонентів і глибокої експертизи. Попри ці складнощі, SIEM залишається центральним елементом інфраструктури безпеки підприємств [3].

Порівняння підходів та комбіновані системи

Розглянуті методи помітно різняться за принципами виявлення загроз та мають свої сильні і слабкі сторони. Сигнатурний аналіз забезпечує швидке й чітке спрацювання на відомі шаблони атак з мінімумом хибних тривог, проте не охоплює нові, ще не задокументовані загрози. Натомість поведінкові (аномалійні) методи здатні виявляти невідомі раніше атаки за відхиленнями від норми, але вони генерують більше помилкових сповіщень і висувають вищі вимоги до ресурсів. Простіші евристичні правила можна впровадити швидко, проте вони покривають лише типові сценарії; складніші статистичні моделі та ML-алгоритми виявляють «тонкі» аномалії та приховані поведінкові патерни, але потребують більше даних, обчислювальних ресурсів і експертизи. Кореляційний підхід (SIEM) вирізняється тим, що об'єднує інформацію з багатьох джерел: він добре виявляє комплексні мультиетапні атаки, які проявляються через серію дрібних подій у різних системах, і надає широкий контекст, проте сильно залежить від наперед заданих правил та якості зібраних даних.

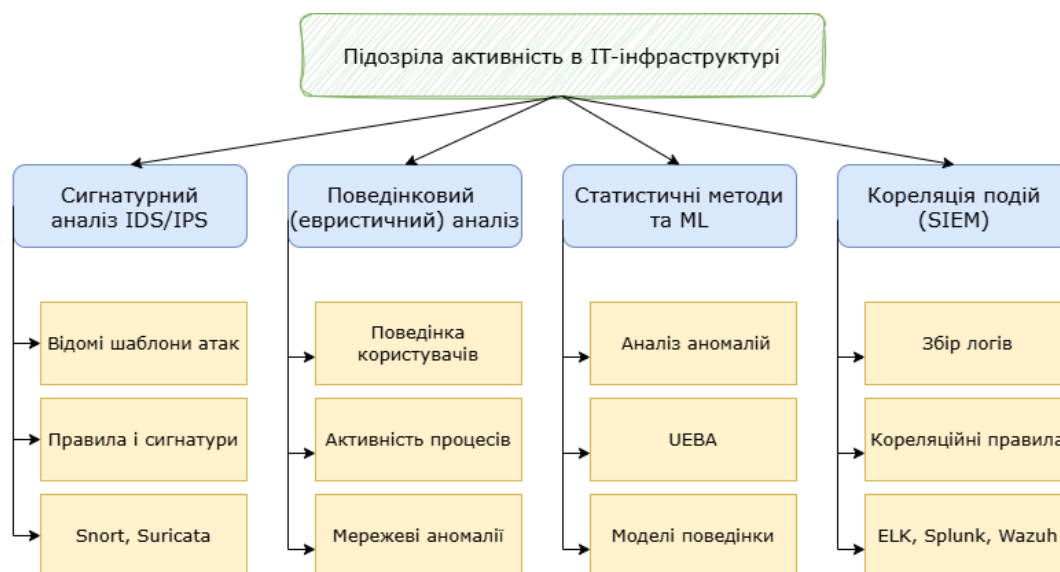


Рисунок 1.1 – Основні методи виявлення підозрілої активності

Зважаючи на ці відмінності, на практиці часто застосовується комбінований підхід. Гібридні системи виявлення вторгнень інтегрують сигнатурні та аномалійні механізми, щоб компенсувати недоліки кожного: у таких системах відомі загрози фіксуються сигнатурним модулем, тоді як нові й непередбачені атаки вилучаються поведінковим аналізом. Комбінування методів підвищує загальну ефективність: зокрема, гібридна IDS здатна виявляти ширший спектр атак порівняно з використанням лише одного підходу. Так само і в межах SOC рекомендовано інтегрувати UEBA-модуль із традиційним SIEM: поведінкова аналітика додає можливість спіймати приховані аномалії, тоді як SIEM надає цілісне бачення подій та централізоване керування інцидентами. У результаті поєднання інструментів (наприклад, зв'язка IDS + SIEM з UEBA) дозволяє досягти принципу «глибокого захисту»: сигнатурні правила дають чіткі індикатори відомих атак, поведінковий аналіз покриває «сліпі зони» нових загроз, а кореляція подій збирає розрізнені сигнали в єдиний сценарій атаки. В додатку А наведена таблиця узагальнено відмінностей розглянутих підходів.

1.3 Підходи та засоби аналізу подій безпеки в IT-інфраструктурі

У цьому розділі розглянуто ключові технології та інструменти моніторингу, що використовуються для виявлення підозрілої активності в IT-інфраструктурі. До них належать системи виявлення та запобігання вторгнень (IDS/IPS), платформи SIEM (Security Information and Event Management), засоби моніторингу мережевого трафіку, а також системи логування. Для кожного інструмента проаналізовано принципи роботи, переваги й недоліки, приклади використання у виявленні загроз, а також архітектурні чи функціональні особливості.

1.3.1 IDS/IPS системи (Snort, Suricata)

Системи виявлення та запобігання вторгнень (IDS/IPS) призначені для моніторингу мережевого трафіку з метою виявлення зловмисної активності або по-

рушень політик безпеки. Вони працюють шляхом аналізу мережових пакетів та застосування наборів правил або моделей поведінки для ідентифікації загроз. IDS-системи зазвичай працюють у режимі *пасивного моніторингу* (сповіщають про загрозу), тоді як IPS може *активно блокувати* або запобігати виявленим атакам. Розгортають IDS/IPS у ключових точках мережі (на периметрі та в сегментах локальної мережі) для перехоплення всього вхідного й вихідного трафіку, що дозволяє виявляти атаки як ззовні, так і можливе горизонтальне поширення загроз всередині мережі.

Snort та **Suricata** – два найпопулярніших інструменти відкритого коду для мережевого виявлення вторгнень. Обидва є *сигнатурними NIDS*, тобто використовують наперед визначені правила (сигнатури) для пошуку відомих атак та аномалій у мережевому трафіку[5].

Snort відомий великою бібліотекою правил і гнучкістю налаштування. У режимі прослуховування він аналізує копії мережевого трафіку та надсилає сповіщення при виявленні сигнатур загроз. До переваг належать перевірена надійність, активна спільнота, широка інтеграція зі сторонніми інструментами й наявність веб-інтерфейсів для аналізу подій. Недоліки – обмежена продуктивність через однопотокową архітектуру, відсутність власного GUI та залежність від сигнатур, що знижує ефективність проти атак нульового дня.

Suricata – сучасна IDS/IPS від OISF, розроблена як продуктивніша альтернатива Snort. Працює в режимах IDS або IPS та аналізує трафік на основі правил. Ключова перевага – багатопотокова архітектура з підтримкою апаратного прискорення, що забезпечує високу пропускну здатність. Suricata підтримує розширений аналіз трафіку (HTTP, DNS, TLS), збір додаткових даних і сумісна з правилами Snort. Недоліки – підвищені вимоги до ресурсів і складніше налаштування. [6]

Приклади використання IDS/IPS. Snort широко застосовується у корпоративних мережах і навіть в домашніх лабораторіях для виявлення сканувань портів, відомих експлойтів, атак на веб-сервери тощо. Завдяки відкритим наборам правил Snort здатний розпізнавати сигнатури троянів, DDoS-атак та інших загроз. Suricata

часто розгортається в сучасних мережах з високою пропускнуою здатністю або як частина комплексних рішень (наприклад, у складі платформи Security Onion чи разом з ELK-стеком для аналізу логів). Suricata ефективна у виявленні складних багатопакетних атак, виявленні аномалій (наприклад, нестандартне використання протоколів чи портів) та може слугувати інструментом глибокого аналізу мережових інцидентів, оскільки надає багаті дані про сесію (HTTP запити, DNS запити тощо). Обидва інструменти інтегруються з SIEM-системами: наприклад, сповіщення Snort або Suricata можна надсилати до Elastic Stack чи Splunk для подальшої кореляції з іншими подіями.[5]

Порівняння Snort vs Suricata. Вибір між Snort і Suricata залежить від потреб: Snort має довгу історію та широку сумісність із різними платформами, а також величезну спільноту, що забезпечує своєчасне оновлення правил. Suricata надає кращу продуктивність у сучасних багатоядерних середовищах і генерує більш детальні лог-дані (зручні для *Network Security Monitoring*). У багатьох випадках Suricata випереджає Snort за швидкістю обробки трафіку та глибиною аналізу, що підтверджують незалежні тести продуктивності. Водночас Snort залишається «золотим стандартом» IDS і продовжує розвиватися (версія Snort 3 отримала підтримку багатопотоковості та модульну архітектуру. Обидва рішення є відкритими й безкоштовними, що робить їх привабливими для організацій з обмеженим бюджетом.[6]

1.3.2 SIEM-системи

SIEM (Security Information and Event Management) – це платформи, призначені для централізованого збору журналів та подій з різних джерел, їх збереження, кореляції та аналізу з метою виявлення інцидентів безпеки. SIEM поєднує можливості *моніторингу в режимі реального часу, довгострокового зберігання логів, кореляції подій і генерації сповіщень/звітів* про підозрілі дії. Традиційно SIEM-системи отримують дані від IDS/IPS, брандмауерів, антивірусів, серверів, додатків тощо, аби дати цілісний огляд безпеки всієї інфраструктури.

ELK Stack (Elastic Stack) – відкрита платформа для збору та аналізу логів, часто використовується як основа для SIEM-рішень. Назва ELK походить від компонентів: Elasticsearch (пошуково-аналітичний рушій), Logstash (агрегатор та процесор логів), Kibana (візуалізація та дашборди) та додатково Beats (агенти для збору даних з хостів). Логи з різних джерел надходять до Logstash або Beats, де вони нормалізуються, далі зберігаються в Elasticsearch для швидкого пошуку, а Kibana дозволяє створювати графіки та дашборди для аналізу інцидентів [28].

Переваги ELK – гнучкість, масштабованість та розвинені засоби аналітики. Стек підтримує обробку великих обсягів даних, розподілене розгортання та, за допомогою Elastic Security, виявлення аномалій і базове машинне навчання для оцінки ризиків користувачів. ELK доступний безкоштовно у базовій версії, що робить його привабливим для організацій будь-якого розміру.

Недоліки: ELK не є готовим SIEM «з коробки» – впровадження повноцінних функцій вимагає конфігурації правил кореляції, індексації та оповіщень. Крім того, деякі компоненти Elastic стали платними, тому повністю відкритим аналогом став OpenSearch від AWS [7].

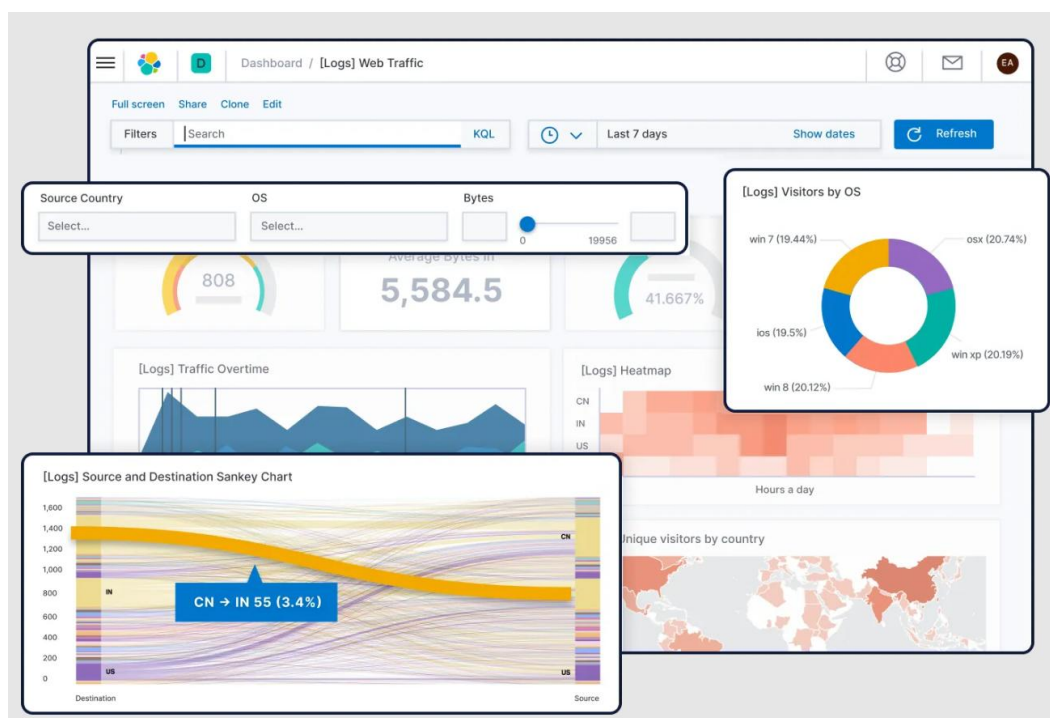


Рисунок 1.2 – ELK Stack[28]

Splunk – комерційна платформа класу SIEM для збору та аналізу логів з різних джерел. Логи збираються через агенти (форвардери), індексуються та доступні для пошуку мовою SPL. Модуль Enterprise Security надає готові кореляційні правила, аналітику поведінки користувачів (UEBA) та дашборди для SOC-аналітиків.

Переваги Splunk: інтегрований підхід – готові механізми оцінки ризиків, кастомні дашборди, довгострокове зберігання даних (гарячі, теплі, холодні індекси) та елементи автоматизації реагування (SOAR). Платформа добре масштабується, підтримує кластеризацію та реплікацію для великих підприємств.

Недоліки: висока вартість ліцензії, великі вимоги до апаратних ресурсів і складність розгортання. Незважаючи на це, Splunk залишається одним із найефективніших SIEM-рішень для великих організацій завдяки поєднанню збору, кореляції та аналітики в одному рішенні.[8]

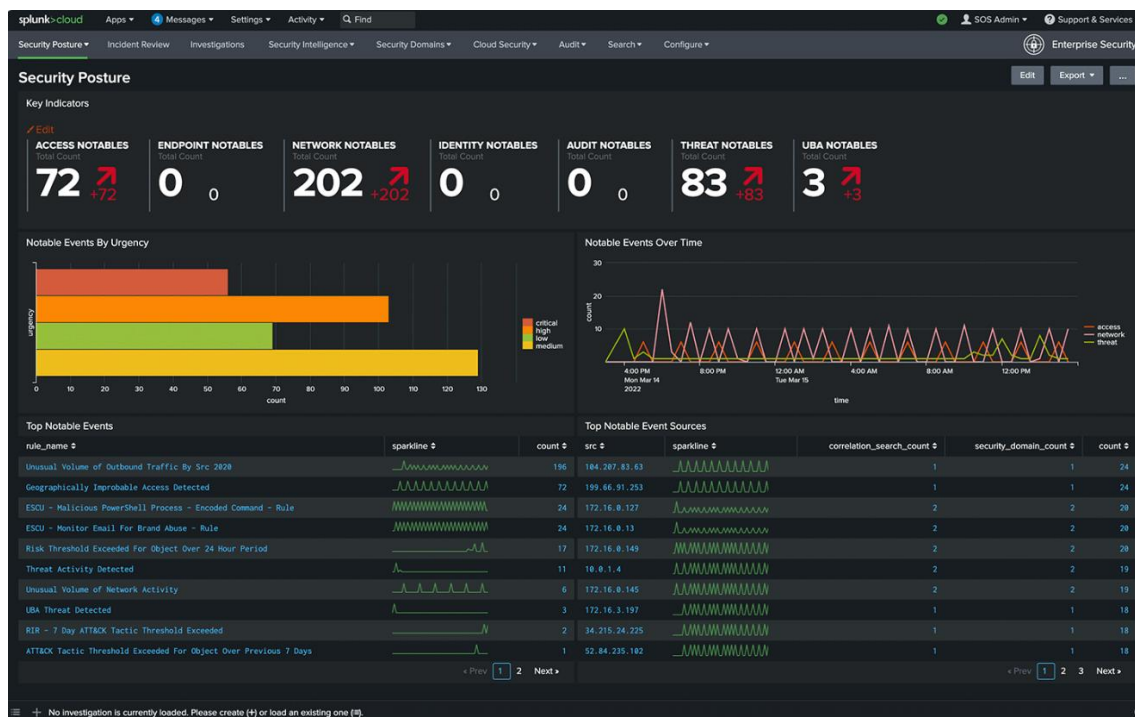


Рисунок 1.3 – Splunk[29]

Wazuh – відкрита SIEM/HIDS-платформа, форк OSSEC, що поєднує моніторинг безпеки, виявлення загроз та реагування на інциденти. Легковагі агенти встановлюються на хости (Windows, Linux, macOS) і збирають дані про систему: жур-

нали, процеси, цілісність файлів, конфігурації тощо. Дані надсилаються на центральний сервер, де аналізуються за правилами та зіставляються з відомими IoC. Оброблені події зберігаються в Elasticsearch і візуалізуються через Kibana. [7]

Переваги Wazuh: повністю відкритий код, поєднання HIDS та SIEM, контроль цілісності файлів у реальному часі, моніторинг системних журналів на підозрілі події, інтегровані правила атак та політик безпеки, підтримка стандартів (PCI DSS, HIPAA) і активні реакції на загрози (наприклад, блокування IP).

Недоліки: самостійне розгортання та підтримка Elastic/Kibana може бути складним для невеликих команд, серверні компоненти працюють тільки на Linux, а велика кількість агентів створює додаткове навантаження на мережу та сервер. Wazuh підходить для організацій, що хочуть SIEM/моніторинг безкоштовно або для побудови власного SOC.

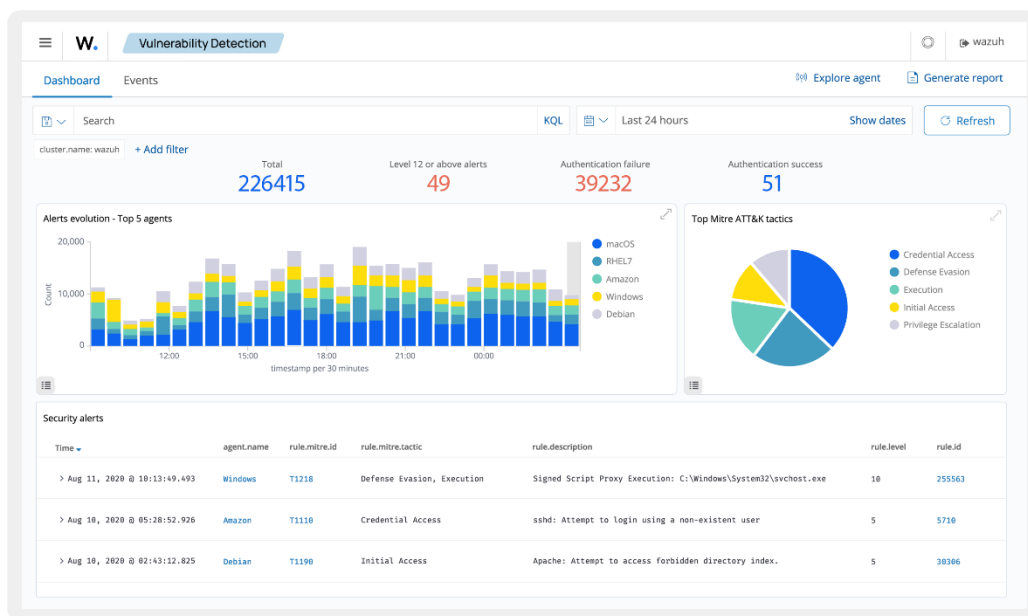


Рисунок 1.4 – Wazuh[7]

Graylog – ще одна популярна платформа для управління логами з відкритим кодом, яку можна використовувати як основу SIEM. Graylog складається з сервера, що приймає і обробляє логи (підтримує Syslog, API, та багато форматів), сховища на основі Elasticsearch/OpenSearch та веб-інтерфейсу. Ключова ідея Graylog – простота розгортання і використання: він надає єдиний інтуїтивний інтерфейс для пошуку по логах, налаштування оповіщень та побудови дашбордів.

Переваги Graylog: централізований збір логів у одному місці, швидкий пошук та фільтрація, гнучкі пайплайни обробки для парсингу та збагачення даних, налаштування оповіщень за умовами (наприклад, кількість невдалих входів) та підтримка ролей для різних користувачів.

Недоліки: базова відкрита версія обмежена функціями SIEM – історичне архівування, тонке налаштування прав і аномалійне виявлення доступні лише у комерційній версії. Проте Graylog підходить малим і середнім компаніям для централізованого збору логів і базового моніторингу безпеки, швидкого розгортання та початкового аналізу подій.[8]

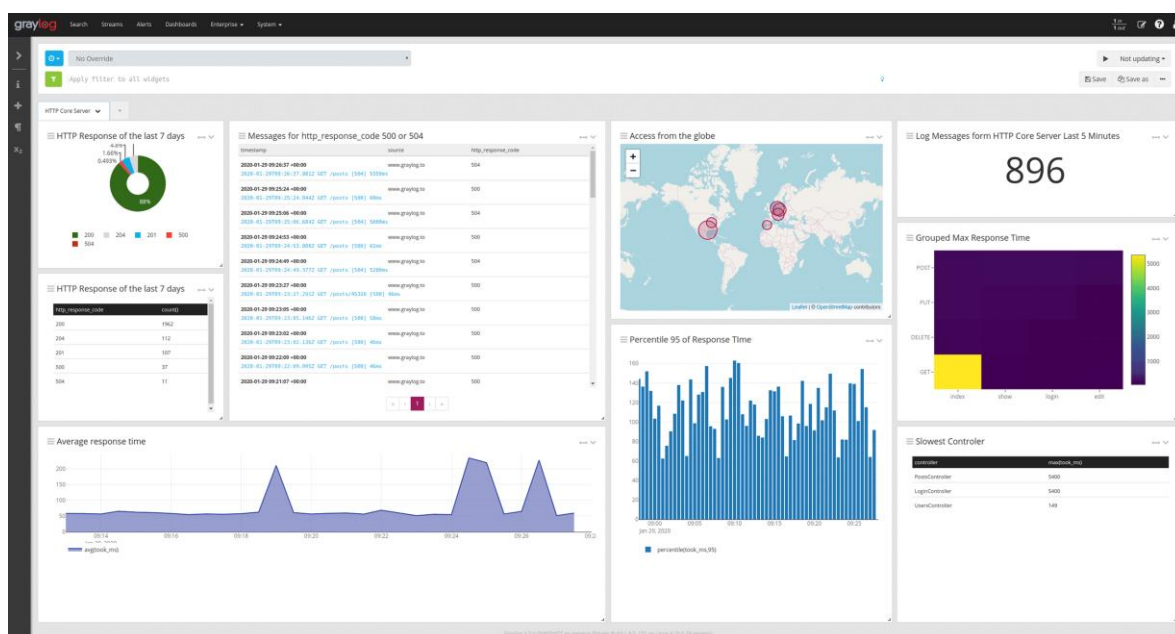


Рисунок 1.5 – Graylog[8]

OSSEC (Open Source Security) – це відкрита хостова система виявлення вторгнень (HIDS), призначена для моніторингу стану кінцевих систем. Вона використовує агент-серверну архітектуру, де агенти встановлюються на хости під управлінням Windows, Linux, Unix або macOS і передають дані на центральний менеджер для аналізу та формування сповіщень.

OSSEC забезпечує контроль цілісності файлів, аналіз системних журналів, виявлення руткітів, моніторинг реєстру Windows та підтримує механізми активної відповіді. Завдяки мультиплатформеності, безкоштовності та зрілій базі правил

OSSEC широко застосовується для базового захисту серверів і робочих станцій у невеликих та середніх інфраструктурах.

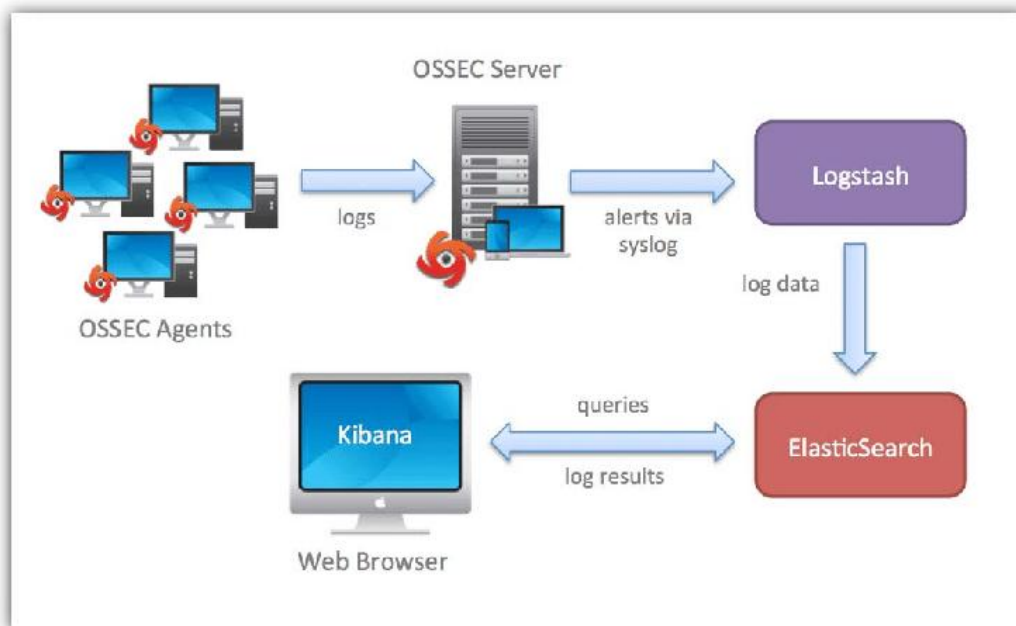


Рисунок 1.6 – Приклад роботи системи OSSEC [9]

До переваг OSSEC можна віднести *простоту розгортання та невибагливість*. У порівнянні з багатьма більш комплексними платформами, OSSEC відносно легко встановити та налаштувати для базових потреб моніторингу хостів. Агент споживає мінімум ресурсів, що дозволяє використовувати його навіть на застарілих або слабких системах. OSSEC підтримує велику різноманітність платформ і може агрегувати логи з різних систем в єдиному центрі для аналізу. Існують базові веб-інтерфейси для перегляду подій, хоча за замовчуванням моніторинг здійснюється через консоль.

Недоліки проявляються при масштабуванні та високих вимогах: розвиток оригінального проекту сповільнився, масштабованість обмежена архітектурою «один сервер + агенти», а інтеграція з іншими системами потребує сторонніх рішень (SIEM, Elastic Stack тощо) [17].

Приклади використання SIEM. Якщо IDS зафіксувала підозрілу активність (наприклад, сканування портів), SIEM може відкорелювати це з логами брандмауера, системними журналами та іншими джерелами, щоб скласти повну картину

атаки. Splunk або Elastic Security можуть побудувати часову лінію подій – від сканування до спроб експлойту – на основі правил кореляції та поведінкових моделей. Хостові рішення, зокрема OSSEC або Wazuh, здатні виявляти наслідки інциденту на рівні системи, такі як зміни критичних файлів чи конфігурацій. Graylog, у свою чергу, може сигналізувати про аномальні дії в службах каталогів, наприклад зміну привілейованих облікових записів поза робочим часом. Таким чином, SIEM забезпечує узагальнений аналіз взаємопов’язаних індикаторів безпеки по всій інфраструктурі.[9]

Нижче наведено порівняння згаданих інструментів за ключовими критеріями:

Таблиця 1.1

Порівняння SIEM систем

Інструмент	Тип	Джерела даних	Гнучкість	Типи загроз, що виявляються
ELK Stack (Elastic Stack)	Відкритий конструктор SIEM	Логи серверів, мережі, додатків, Beats	Висока (парсинг, нормалізація, побудова дашбордів, аналітика через Elastic Security)	Аномалії поведінки користувачів, інциденти безпеки, підозріла активність, витоки даних[7]
Splunk	Комерційний SIEM	Логи серверів, мережі, додатків, форвардери	Висока (SPL, кореляція, UEBA, SOAR)	Сканування портів, експлойти, підозрілі процеси, атаки на облікові записи, інсайдерські загрози[7]
Wazuh	Відкритий HIDS + SIEM	Логи систем, файли, процеси, конфігурації	Середня-висока (правила OSSEC, інтеграція з Elastic/Kibana)	Злом облікових записів, зміни файлів (FIM), руткіти, підбор паролів, порушення політик безпеки[9]
Graylog	Відкритий лог-менеджер / SIEM	Логи серверів, мережі, Syslog, API	Середня (пайплайни обробки, оповіщення, фільтрація)	Підозріла активність користувачів, незвичні події в Active Directory, мережеві аномалії[9]
OSSEC	HIDS	Системні логи, файли, конфігурації, реєстр Windows	Середня (правила, активна відповідь)	Зміни файлів, руткіти, несанкціоновані дії на хості[9]

1.3.3 Засоби моніторингу мережі

Моніторинг мережевого трафіку є важливою складовою виявлення аномальної або зловмисної активності. На відміну від IDS, які зосереджені на спрацьовуванні сигнатур чи правил, мережеві моніторингові інструменти забезпечують детальний огляд того, що відбувається у мережі, і дають змогу фахівцям вручну чи автоматично аналізувати трафік для пошуку ознак проблем. Залежно від підходу, інструменти моніторингу можна поділити на кілька основних класів.

Wireshark – найпоширеніший аналізатор мережевих пакетів, що дозволяє захоплювати та детально переглядати трафік на різних рівнях протоколів. Він підтримує фільтрацію пакетів, декодування сотень протоколів і наочну візуалізацію заголовків, що робить його важливим інструментом для адміністраторів і фахівців з безпеки.

Wireshark працює в режимі пасивного прослуховування мережевого інтерфейсу або аналізу дамів трафіку, забезпечуючи глибокий аналіз переданих даних (за відсутності шифрування). Серед його переваг – універсальність, підтримка різних мережевих середовищ, зручний графічний інтерфейс і відкритість.

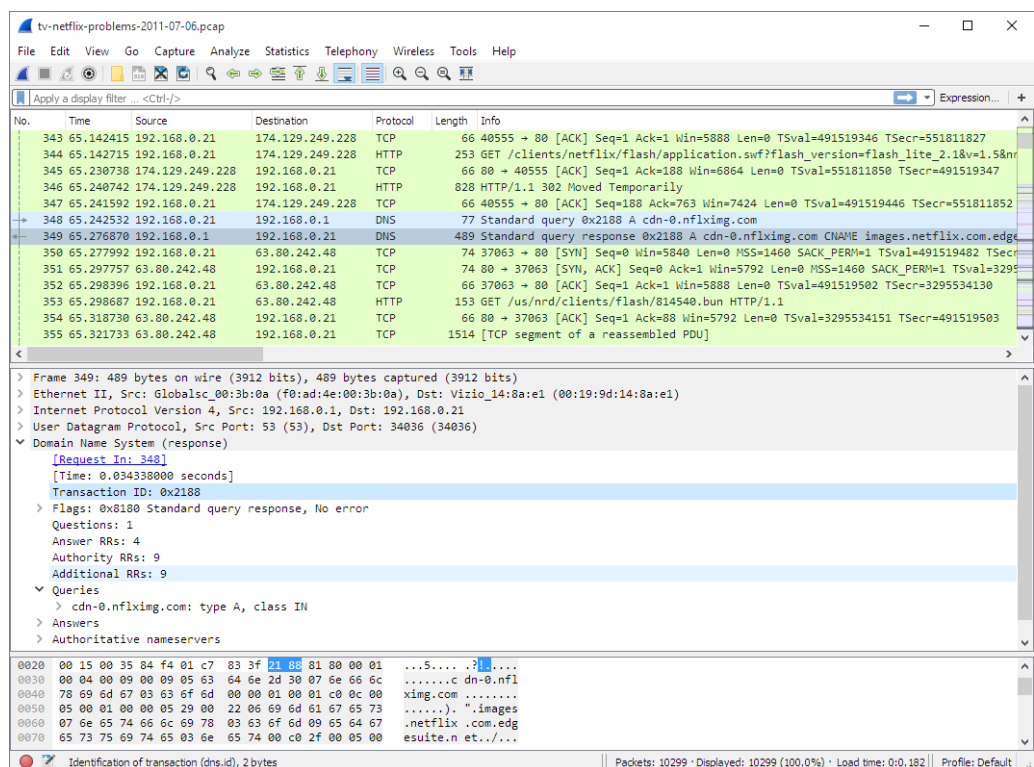


Рисунок 1.7 – Wireshark [10]

Водночас Wireshark не є системою виявлення вторгнень і не призначений для автоматичного моніторингу великих обсягів трафіку в реальному часі. Його зазвичай застосовують для форензичного аналізу інцидентів і перевірки спрацювань IDS, що дозволяє детально з'ясувати, що саме відбулося на мережевому рівні [10].

Zeek (раніше Bro) – мережевий аналізатор безпеки, який істотно відрізняється від сигнатурних IDS на кшталт Snort/Suricata. Zeek працює як *Network Security Monitor*: він не просто шукає сигнатури, а *реєструє детальні логи активності* на мережі і на основі потоку подій виконує скрипти для виявлення аномалій або певних шаблонів поведінки. Він дозволяє збирати багаті дані (HTTP, DNS, SSL, файли, з'єднання) та створювати складні правила аналізу без перекомпіляції програми.

Переваги Zeek – висока гнучкість, розширюваність за допомогою скриптів та можливість масштабування на кластері для великих мереж. Недоліки – крута крива навчання, потреба в знанні мережевих протоколів і програмуванні Bro/Zeek-скриптів, а також відсутність власного GUI. Zeek часто використовується разом зі Snort/Suricata, доповнюючи сигнатурний аналіз поведінковим і дозволяючи виявляти патерни, невидимі для класичних IDS [16].

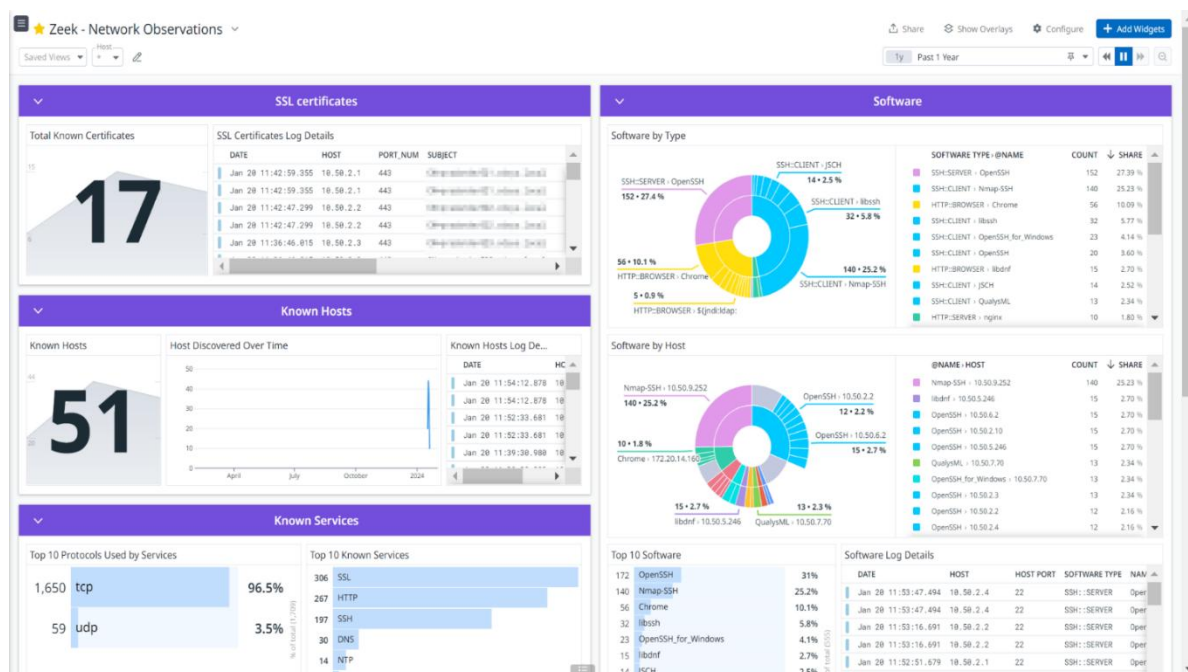


Рисунок 1.8 – Zeek [16]

NetFlow – технологія збору телеметрії про мережеві потоки, спочатку розроблена Cisco. На відміну від повного захоплення пакетів, NetFlow записує *агреговану інформацію про з'єднання*: IP-адреси джерела й призначення, порти, протокол, об'єм переданих байтів, час початку та завершення сесії тощо[11]. Переваги NetFlow – невеликий обсяг даних при збереженні ключової інформації про мережеві потоки, можливість використання для ретроспективного розслідування інцидентів і інтеграції з колекторами та SIEM-системами. Недоліки – це лише метадані, без перегляду вмісту пакетів, тому для детального аналізу потрібні додаткові інструменти.

NetFlow особливо ефективний для *виявлення аномалій*: різке зростання трафіку може свідчити про DDoS або масове викачування даних; з'явлення незвичних пар «джерело-призначення» чи великої кількості сесій.

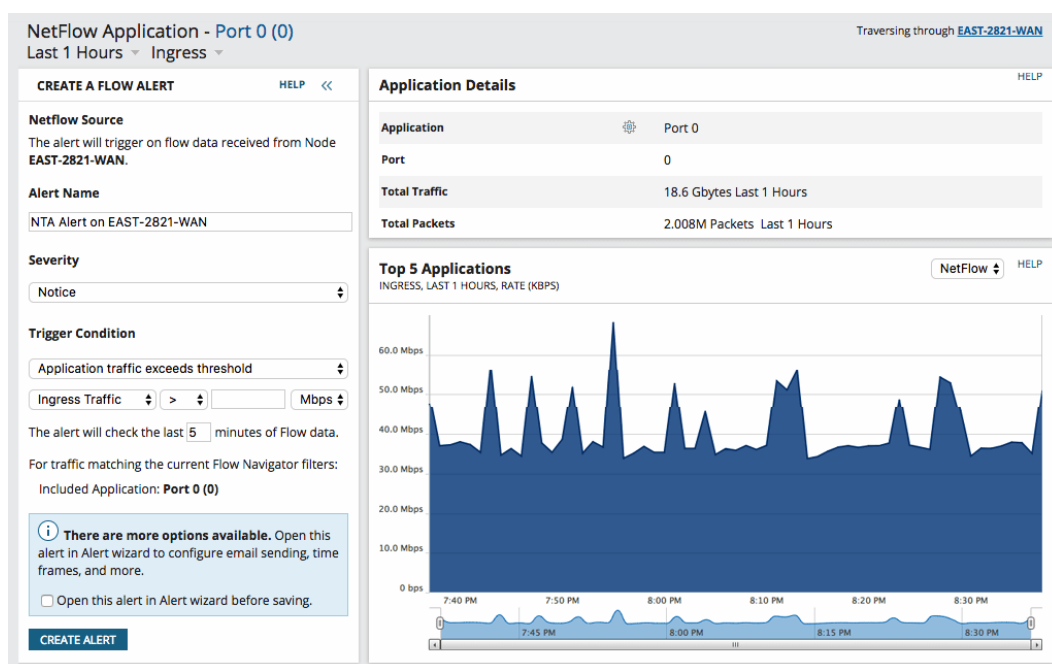


Рисунок 1.9 – NetFlow [17]

1.3.4 Системи логування

Збір і аналіз логів – фундаментальний метод виявлення підозрілої активності, адже більшість атак чи аномалій залишають слід у системних журналах. Логи дозволяють виявляти аномалії, підвищення привілеїв, запуск підозрілих процесів чи зміни критичних файлів і служб.

Syslog. Syslog – стандартна система журналювання для Unix/Linux, що збирає повідомлення від ядра, сервісів і додатків. Логи можуть надсилатися локально або на центральний сервер, що дозволяє централізовано відстежувати події всієї інфраструктури. Кожне повідомлення містить джерело (facility) та рівень важливості (severity), що допомагає структурувати і фільтрувати дані.

Системи на кшталт rsyslog або syslog-ng підтримують фільтрацію, буферизацію, шифрування і відправку логів у бази даних чи хмарні сервіси. Це забезпечує масштабоване та безпечне зберігання даних для подальшого аналізу [12].

У безпековому контексті Syslog дозволяє відстежувати підозрілу активність: багаторазові невдалі входи, звернення до критичних служб, перезапуски сервісів або зміни конфігурації. Крім того, логи з Syslog можуть інтегруватися з IDS/IPS, WAF та іншими інструментами, надаючи SOC аналітикам оперативний огляд ключових подій.

Windows Event Log. Windows Event Log збирає події ОС і додатків, які переглядаються через Event Viewer. Основні журнали: Application (події додатків), System (події системи) та Security (події безпеки). Кожна подія має унікальний Event ID, що полегшує автоматичний аналіз і виявлення підозрілої активності.

Журнали Windows дозволяють відстежувати спроби входу, зміни користувачів, запуск процесів або доступ до критичних ресурсів. За допомогою Windows Event Forwarding можна централізовано збирати події з декількох хостів.[13]

У безпеці Event Log використовується для аналізу атак на облікові записи, виявлення підозрілих процесів або потенційних витоків даних. SOC-аналітики часто створюють фільтри за Event ID для швидкого виявлення інцидентів.

systemd-journald (journald, Linux). Journald – система логування у сучасних Linux-дистрибутивах на базі systemd, яка зберігає всі повідомлення ядра, служб і додатків у централізованому журналі. Логи структуровані і містять інформацію про процеси, сервіси та користувачів, що полегшує пошук та аналіз подій. Journald підтримує фільтрування, ротацію і контроль доступу, що дозволяє безпечно зберігати

важливі дані. Логи можна експортувати в syslog або колектори для централізованого аналізу.

У безпеці journald дозволяє відстежувати системні події, ознаки експлойтів, дії користувачів з підвищеними правами та інші аномалії. Аналітики можуть швидко отримати повну хронологію подій, що спрощує розслідування інцидентів. [14]

Загальна роль систем логування. Журнали – це «чорні ящики» системи, що фіксують усі ключові події. При ознаках компрометації або аномалії саме логи дають перші підказки: незвичні спроби підвищення привілеїв, нові служби, зміни конфігурації. Безпечна інфраструктура передбачає централізований збір та зберігання логів і їх регулярний аналіз – ручний або автоматичний. Навіть без SIEM чи машинного навчання, грамотне налаштування Syslog, Event Viewer та journald разом з увагою адміністратора дозволяють виявляти більшість інцидентів на ранніх етапах [12].

1.4 Огляд інтегрованих та спеціалізованих рішень для аналізу підозрілої активності

Security Onion як комплексна платформа аналізу подій безпеки. Security Onion – це повнофункціональна відкрита платформа (Linux-дистрибутив) для побудови SOC, спроектована «захисниками для захисників». Вона забезпечує мережевий та хостовий моніторинг, створює honeypot-сегменти, збирає та корелює логи, а також надає засоби кейс-менеджменту[26]. Інакше кажучи, Security Onion «об'єднує під одним дахом» найкращі відкриті інструменти: Elastic Stack (Elasticsearch, Logstash, Kibana) для зберігання й візуалізації даних, Suricata і Zeek (Bro) як високопродуктивні NIDS для аналізу трафіку, OSSEC (або Wazuh) для моніторингу хостів, а також додаткові утиліти (osquery, CyberChef тощо). Такий «all-in-one» підхід дозволяє швидко розгорнути лабораторію безпеки чи SOC: є готові графічні консолі для перегляду сповіщень і панелі розвідки (наприклад, вбудована Security Onion Console із dashboard Kibana) та засоби заглибленого threat hunting.

Зокрема, платформа орієнтована на пошук загроз і корпоративний моніторинг безпеки, як зазначено в офіційних матеріалах[27].

Архітектура Security Onion є гнучкою: її можна запускати на одному сервері (режим standalone для невеликих мереж) або розгортати у розподіленій конфігурації з декількома сенсорами та кластером пошуку для великих підприємств. Усе зібране з датчиків (мережевих та хостових) надходить у сховище на основі Elasticsearch (або OpenSearch) і стає доступним через веб-інтерфейси та спеціалізовані консолі (alerts, dashboards, threats hunt тощо). Завдяки широкому використанню Elastic Stack у проектах на зразок Wazuh та Security Onion, ці компоненти стали дефакто-стандартом для відкритих SIEM і SOC[27]. Водночас Security Onion активно підтримується спільнотою та комерційною організацією Security Onion Solutions LLC, і вже отримала визнання в освітніх закладах і великих організаціях по всьому світу[26], що підкреслює її роль у екосистемі відкритої безпеки.

Count	rule.name	event.module	event.severity.label
29	GPL NETBIOS SMB-DS IPC\$ unicode share access	suricata	low
21	System Audit event.	ossec	low
11	ET P2P BitTorrent peer sync	suricata	high
10	GPL NETBIOS SMB IPC\$ unicode share access	suricata	low
8	ET MALWARE Generic - POST To .php w/Extended ASCII Characters (Likely Zeus Derivative)	suricata	high
8	ET MALWARE Zbot POST Request to C2	suricata	high
8	GPL SNMP public access udp	suricata	medium
6	ET POLICY PE EXE or DLL Windows file download HTTP	suricata	high
6	Listened ports status (netstat) changed (new port opened or closed).	ossec	low
5	ET HUNTING Observed Interesting Content-Type inbound (application/x-sh)	suricata	high
5	ET INFO Hiloti Style GET to PHP with invalid terse MSIE headers	suricata	high
5	ET USER_AGENTS Suspicious User-Agent - Possible Trojan Downloader (ver18/ver19 etc)	suricata	high
4	ET MALWARE Tibs/Harnig Downloader Activity	suricata	high
4	GPL P2P BitTorrent transfer	suricata	high
4	Ossec server started.	ossec	low
3	ET MALWARE GENERIC Likely Malicious Fake IE Downloading .exe	suricata	high
3	Ossec agent started.	ossec	low
2	ET MALWARE Possible Windows executable sent when remote host claims to send html content	suricata	high
2	ET MALWARE Terse alphanumeric executable downloader high likelihood of being hostile	suricata	medium
2	ET MALWARE Zbot Generic URI/Header Struct .bin	suricata	high

Рисунок 1.10 – Виявлення подій в Security Onion[26]

Обмеження інтегрованих платформ у дослідницьких сценаріях. Попри численні переваги, інтегровані рішення типу Security Onion мають свої обмеження

для наукових досліджень. Згідно з практичними спостереженнями, «безкоштовне» відкриття коду часто означає високі часові витрати на розгортання, тонке налаштування і супровід платформи. Запуск та адаптація SIEM/SOC-системи на базі Security Onion вимагатиме значної технічної експертизи (налаштування правил Suricata/Zeek, налаштування Elastic, узгодження версій компонентів тощо). Крім того, не існує жодного «ідеального» універсального рішення: досягнення повного набору функціоналу часто вимагає додаткового поєднання інших інструментів і ручної інтеграції[27]. У контексті цього дослідження Security Onion не була обрана як єдиний продукт з кількох причин. Зокрема, ускладнена структура та взаємозв'язок компонентів можуть знизити прозорість процесів: багато операцій виконуються «під капотом» платформи, що ускладнює детальний аналіз поведінки окремих модулів. Також жорстка прив'язка до певної версії стеку (Elastic, сигнатур, інтерфейсів) обмежує гнучкість: складніше експериментувати з альтернативними бібліотеками, алгоритмами чи сценаріями виявлення без перепакування всього дистрибутиву. У результаті для наукової роботи обрана комбінація окремих спеціалізованих інструментів, оскільки це дає ширші можливості тонкого налаштування, контролю кожного рівня аналізу та прозорості при тестуванні детальних гіпотез.

Спеціалізовані рішення для різних рівнів аналізу. У якості альтернативи комплексним платформам використовуються окремі інструменти, що сфокусовані на певних завданнях. Наприклад, Wazuh – це сучасний відкритий SIEM / HIDS, який є форком OSSEC і «вдосконалює» його можливості. Згідно з документацією, Wazuh поєднує збір та кореляцію логів із функціями аналізу кінцевих точок (EDR). Він моніторить цілісність файлів, проводить аналіз логів на основі сигнатур та аномалій, виконує вразливостний скан і може автоматично реагувати на інциденти[27]. Архітектура Wazuh – клієнт-серверна: легкі агенти збирають дані з хостів, центральний сервер обробляє їх, а Elasticsearch (OpenSearch) індексує та зберігає повідомлення. Така система призначена для великомасштабних розгортань (підтримує

кластеризацію, хмарні інтеграції)[9], роблячи Wazuh придатним для корпоративних середовищ. У дослідженнях Wazuh корисний для моделювання SIEM-сценаріїв, аналізу поведінки процесів на хості та порівняння з мережевими даними.

- Zeek (колишній Bro) – потужний мережевий аналізатор трафіку. Він пасивно моніторить пакети, виконує глибокий розбір протоколів і генерує дуже детальні логи транзакцій та файл-трафіку, які можна потім аналізувати для виявлення атак або аномалій. Як зазначено на офіційному сайті, Zeek «не діє як активний щит», а натомість «записує логи високої точності трафіку», що ідеально підходить для ручного аналізу або інтеграції у SIEM. У дослідженнях Zeek широко використовується для реконструкції сценаріїв атак та аналізу поведінки мережі, оскільки його скриптова модель дозволяє налаштовувати нові правила й збирати розширену телеметрію (понад 70 типів логів за замовчуванням)[16].

- Suricata – високопродуктивна система IDS/IPS з відкритим кодом. Вона використовує сигнатурно-правилкове виявлення (сумісна з правилами Snort/ET), виконує багатопотоковий аналіз трафіку і глибоке DPI («глибокий аналіз пакету»). Suricata може працювати як пасивне IDS або активний IPS, а також формувати мережеву телеметрію (NSM), фіксуючи параметри потоків і характерні метадані трафіку[27]. Через сучасну архітектуру з підтримкою багатоядерності Suricata добре масштабовується на потужних системах. В академічних дослідженнях її часто використовують для перевірки ефективності сигнатур, побудови правил виявлення та вивчення впливу навантаження на IDS.

- OSSEC – класичний HIDS з відкритим кодом, предок Wazuh. Це багатифункціональний монітор змін файлів і логів на хості. Як зазначено у довідникових матеріалах, OSSEC пропонує потужне управління логами та реальний час моніторингу подій, але зараз він поступово витісняється Wazuh[17]. Наразі у дослідницьких проєктах OSSEC згадують переважно з історичної точки зору – це основа, на базі якої створено Wazuh. OSSEC сам по собі менш активний в розробці, проте він залишається показовим прикладом архітектури агент-менеджер.

Поєднання цих спеціалізованих інструментів дає гнучкість: дослідник може встановити та налаштувати потрібні засоби окремо, аналізувати «чисті» вихідні дані та оцінювати внесок кожного компонента. Наприклад, Wazuh схиляє до єдиного SIEM-ритму, а використання окремих Suricata/Zeek дозволяє варіювати стратегії мережевого моніторингу [27]. Таким чином, комбінування спеціалізованих інструментів із відкритим кодом дозволяє балансувати між контролем кожного компонента та зручністю централізованого аналізу.

Узагальнення підходів до вибору засобів аналізу. Інтегровані системи (наприклад, Security Onion) надають готовий набір інструментів, що полегшує старт і забезпечує єдину панель управління. Вони швидко запускаються як повноцінний SOC, але часто є досить «важкими» та менш гнучкими для модифікацій. Спеціалізовані рішення (Zeek, Suricata, Wazuh, OSSEC) вимагають більше початкової роботи з інтеграції, проте вони надають прямий контроль над кожним елементом аналітики. У навчальних та дослідницьких цілях оптимальним зазвичай є комбінований підхід: наприклад, реалізувати власний SOC, встановивши окремі сенсори (Zeek, Suricata) і централізовані сервери (Wazuh/ELK), а не готову збірку. Це дозволяє експериментувати з різними комбінаціями інструментів, варіювати параметри ідео та аналізувати кожен рівень по черзі.

Для наочності порівняння розглянутих рішень, наведено зведену таблицю їх основних характеристик, сильних та слабких сторін (табл. 1.2).

Таблиця 1.2

Порівняння Wazuh, Suricata, Zeek, OSSEC та Security Onion

Рішення	Класифікація та сфера	Ключові функції	Сильні сторони	Обмеження
Wazuh	HIDS + SIEM	HIDS, моніторинг логів, контроль цілісності файлів, виявлення rootkit, оцінка вразливостей, активна відповідь[19].	Мультиплатформовість, гнучке налаштування, безкоштовне, підтримка хмарних середовищ та кластеризації, REST API, інтеграція з іншими системами[9].	Високі вимоги до ресурсу (Elasticsearch/Kibana), складність розгортання і налаштування, крута крива навчання, можливі false positives[9].

Рішення	Класифікація та сфера	Ключові функції	Сильні сторони	Обмеження
Suricata	NIDS/IPS	Аналіз мережевого трафіку в реальному часі, виявлення атак за сигнатурами та аномаліями, підтримка багатьох протоколів[5].	Швидка детекція атак, інтеграція з SIEM, високопродуктивна, підтримка IPS режиму.	Може генерувати велику кількість хибних спрацювань, потребує тонкого налаштування правил
Zeek	Мережевий моніторинг (NSM)	Глибокий аналіз мережевого трафіку, логування протоколів, створення користувацьких скриптів для виявлення аномалій[16].	Гнучкі скрипти, детальний аналіз мережевих сесій, підтримка високонавантажених мереж, інтеграція з SIEM/SOC.	Не блокує трафік, відсутність UI, високий поріг входження, великий обсяг логів, потребує зовнішніх систем для аналізу та кореляції.
OSSEC	Хостовий моніторинг (HIDS)	Моніторинг цілісності файлів, аналіз логів, виявлення rootkit, активна відповідь, моніторинг реєстру Windows[17].	Простота розгортання, мінімальні ресурси агентів, мультиплатформовість, надійний базовий HIDS.	Обмежена масштабованість, старіше ядро проекту, менша частота оновлень, обмежена інтеграція та аналітика без зовнішніх SIEM[9].
Security Onion	Інтегрована SOC-платформа	Мережевий IDS/IPS (Suricata, Zeek), хостовий моніторинг (Wazuh), збір і кореляція логів (Elastic Stack), packet capture, threat hunting.	Комплексність, швидке розгортання, інтегровані інструменти, готові дашборди, масштабованість[26].	Високі апаратні вимоги, складність адміністрування всіх компонентів, обмежена гнучкість, потребує експертності для тюнінгу та зменшення шуму[27].

Висновки до першого розділу

Різні інструменти моніторингу та аналізу підозрілої активності доповнюють один одного та виконують різні функції в системі забезпечення кібербезпеки. IDS/IPS-системи забезпечують перший рівень захисту, автоматично виявляючи відомі мережеві атаки та аномальну активність у режимі реального часу. SIEM-платформи здійснюють централізований збір і кореляцію подій безпеки з різних джерел, що дозволяє виявляти складні інциденти та ланцюжки подій, які не фіксуються окремими засобами моніторингу.

Хостові системи виявлення вторгнень забезпечують детальний контроль стану кінцевих систем, дозволяючи аналізувати зміни конфігурацій, файлів, процесів і системних журналів. Мережеві засоби аналізу, зокрема NIDS-рішення, формують уявлення про поточну мережеву активність та дозволяють своєчасно виявляти спроби атак на рівні мережі. Системи логування виступають фундаментом усієї аналітики безпеки, оскільки саме журнальні дані є основою для подальшої кореляції та розслідування інцидентів.

Таким чином, ефективна стратегія кібербезпеки в сучасній IT-інфраструктурі передбачає *комплексне використання SIEM-рішень, хостових та мережевих систем виявлення вторгнень*, де події з різних рівнів інфраструктури агрегуються, корелюються та аналізуються в єдиному центрі[27]. Мережеві IDS/IPS-системи формують події безпеки та можуть здійснювати первинне реагування, тоді як SIEM забезпечує узагальнений контекст інцидентів і підтримує подальший поглиблений аналіз. Такий підхід дозволяє ефективно виявляти як масові відомі атаки, так і більш складні цілеспрямовані загрози, забезпечуючи цілісне бачення стану безпеки IT-інфраструктури.

РОЗДІЛ 2 ПРОЄКТУВАННЯ СИСТЕМИ АНАЛІЗУ ПІДОЗРІЛОЇ АКТИВНОСТІ В ІТ-ІНФРАСТРУКТУРІ

2.1 Загальна концепція та архітектура системи

В умовах сучасної ІТ-інфраструктури, що охоплює численні сервери, кінцеві точки, мережеве обладнання та хмарні сервіси, постійно зростає складність кіберзагроз. Тож для аналізу потенційно підозрілої активності використовують централизовані системи, які збирають та об'єднують події з усіх джерел. Системи типу SIEM (Security Information and Event Management) отримують журнали подій від різних пристроїв (ПК, серверів, комутаторів, маршрутизаторів, фаєрволів, IDS/IPS та ін.) і дають цілісне уявлення про безпековий стан мережі. Логи –це записи подій в системах, що містять відомості про дії користувачів, процесів і додатків. Вони є критично важливими для виявлення злочинних чи аномальних дій у мережі. Наприклад, NIST визначає концепцію безперервного моніторингу (ISCM) як підтримку постійної обізнаності про стан безпеки, уразливості та загрози з метою прийняття рішень в управлінні ризиками. Таким чином, архітектура таких систем орієнтована на безперервний збір, узагальнення та аналіз телеметрії з мережі для швидкого виявлення інцидентів та реакції на них[9].

Типові компоненти системи аналізу підозрілої активності є наступними:

- *Збір та логування даних.* Цей компонент відповідає за отримання подій з різних джерел (мережеві пристрої, сервери, кінцеві точки, застосунки тощо) та їхню передачу до централізованого сховища. Логи можуть збиратися за допомогою агентів на кінцевих системах або мережевих колекторів, а також через API та стріми з хмарних сервісів. Сучасні рішення підтримують як реальний час передачі подій (streaming), так і пакетне завантаження (batch). Особливо важливо охоплювати всі ключові точки видимості безпеки, оскільки пропуски в логуванні можуть призвести до втрати інформації про злочинні дії[3].

– *Нормалізація та сховище даних.* Після збору сирі журнальні записи консоліднуються та нормалізуються у єдиний формат. Це дозволяє уніфікувати часові мітки, формати адрес та коди подій із різних систем, спрощуючи їх порівняння та кореляцію. Нормалізовані дані зберігаються в централізованому сховищі (базі даних, файловому кластері чи Big Data платформі) з урахуванням політик зберігання й ротації. Цей етап критично важливий для цілісності даних: без уніфікації форматів обробка і пошук по журналах значно ускладнюються, особливо якщо застосовуються автоматизовані методи пошуку аномалій.

– *Кореляція та аналіз подій.* Центральним компонентом є кореляційний рушій, який аналізує нормалізовані події та виявляє підозрілі шаблони. Застосовуються різні підходи: сигнатурні (за заздалегідь визначеними правилами) і аналітичні з використанням статистичних та машинно-навчальних моделей. Наприклад, відомі методи кореляції включають кінцеві автомати, баєсові мережі, нейронні мережі та інші алгоритми. Також впроваджують поведінковий аналіз (UEBA) – моделі нормальної поведінки користувачів та об'єктів, які допомагають виявляти відхилення. Процес кореляції зазвичай містить кілька етапів: фільтрацію нерелевантних подій, агрегування споріднених записів і видалення дублікатації, а також аналіз «root cause» – визначення причинно-наслідкових зв'язків між подіями. Наприклад, у деяких підходах спочатку відсіюють «шумні» записи, потім методом агрегування і дедуплікації формують узагальнені події, а наостанок аналізують причинні зв'язки[4].

– *Візуалізація та звітування.* Результати аналізу подаються у вигляді інтерактивних панелей, дашбордів і звітів, що відображають важливу інформацію про стан безпеки. Наприклад, кореляційний механізм може перетворювати великий потік сирих даних у компактні сповіщення і виводити їх на графічному інтерфейсі, показуючи поточний стан безпеки організації. Такі панелі в реальному часі дозволяють аналітикам відстежувати аномалії трафіку, підозрілі активності користувачів, останні інциденти тощо, що допомагає оперативно визначати пріоритетні загрози і вживати відповідних заходів[3].

– *Зворотний зв'язок та вдосконалення.* Важливою складовою є механізм зворотного зв'язку: результати розслідувань та реакцій аналітиків використовуються для удосконалення правил та моделей. Зворотний зв'язок може реалізовуватися у вигляді перенавчання моделей на нових даних, уточнення сигнатур або кореляційних алгоритмів. Наприклад, на практиці рекомендується організовувати постійний канал зворотного зв'язку між аналітиками та командою підтримки SIEM для виявлення «сліпих зон» та зменшення хибних спрацьовувань. Такий цикл «continuous improvement» забезпечує поступове підвищення точності виявлення інцидентів і адаптацію системи до нових типів загроз.

На рисунку 2.1 показана типова архітектура SIEM-системи, де події з різних джерел (сервери, мережеві пристрої, IDS/IPS) централізовано збираються, нормалізуються та корелюються для подальшого аналізу і формування сповіщень.

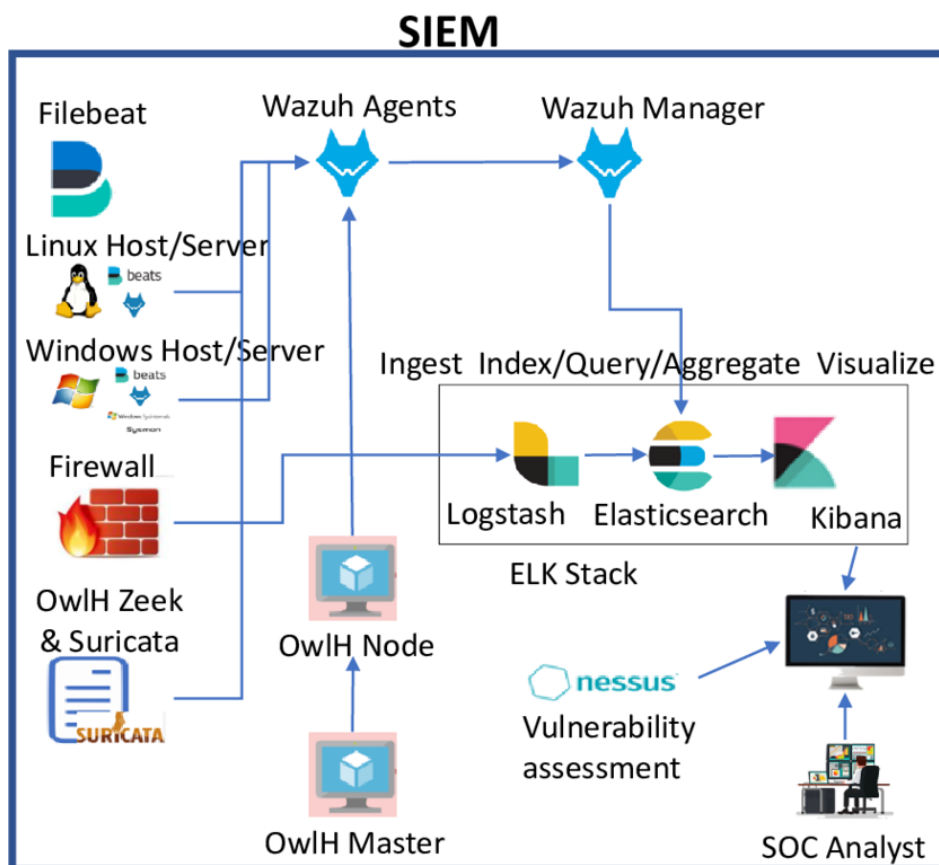


Рисунок 2.1 – Загальна архітектура SIEM-системи: джерела даних, збір, нормалізація, кореляція та візуалізація

Концепція архітектури систем аналізу підозрілої активності базується на стандартах галузі: MITRE ATT&CK забезпечує основу для сценаріїв атак і правил кореляції, а стандарти NIST (SP 800-137) наголошують на безперервному зборі та аналізі даних для управління ризиками. Основна логіка полягає в централізації та уніфікації логів, застосуванні гнучкого кореляційного рушія з поєднанням правилевих і поведінкових методів, а також наявності зручних інтерфейсів із підтримкою зворотного зв'язку.

2.2 Логічна модель мережевої інфраструктури

У рамках розробки архітектури системи виявлення підозрілої активності запропоновано поділ мережі на функціональні зони з чітко визначеними ролями. Такий підхід дозволяє локалізувати та контролювати потоки даних, обмежувати можливості атакуючої сторони та спрощувати процес моніторингу подій безпеки. Як зазначається у фахових рекомендаціях, структурована схема з виділеним сегментуванням підвищує ефективність систем безпеки та засобів моніторингу.

Відповідно до цього підходу мережеву інфраструктуру поділено на три основні логічні зони: зону управління, зону робочих станцій та зону атаки. Така сегментація імітує типову корпоративну IT-інфраструктуру та створює умови для дослідження підозрілої активності в контрольованому середовищі.

У зоні управління (Management zone) розміщено сервер Wazuh Server з IP-адресою 192.168.100.10/24, який виконує роль центрального вузла збору, кореляції та аналізу подій безпеки. Сервер Wazuh функціонує як SIEM-платформа та забезпечує централізований аналіз журналів, що надходять від клієнтських систем і мережевих сенсорів.

Зона робочих станцій (Internal zone) включає хост Windows 11 з IP-адресою 192.168.100.25/24 та сервер Ubuntu 20.04 LTS з IP-адресою 192.168.100.30/24. На цих системах встановлено Wazuh-агенти, які безперервно передають на сервер інформацію про події безпеки, зокрема журнали операційної системи, події автентифікації, змін у файловій системі (FIM) та інші метрики. Сервер Ubuntu додатково

використовується як мережевий сенсор із розгорнутою системою IDS Suricata, що дозволяє здійснювати аналіз мережевого трафіку.

На рисунку 2.2 зображено схему мережевої інфраструктури.

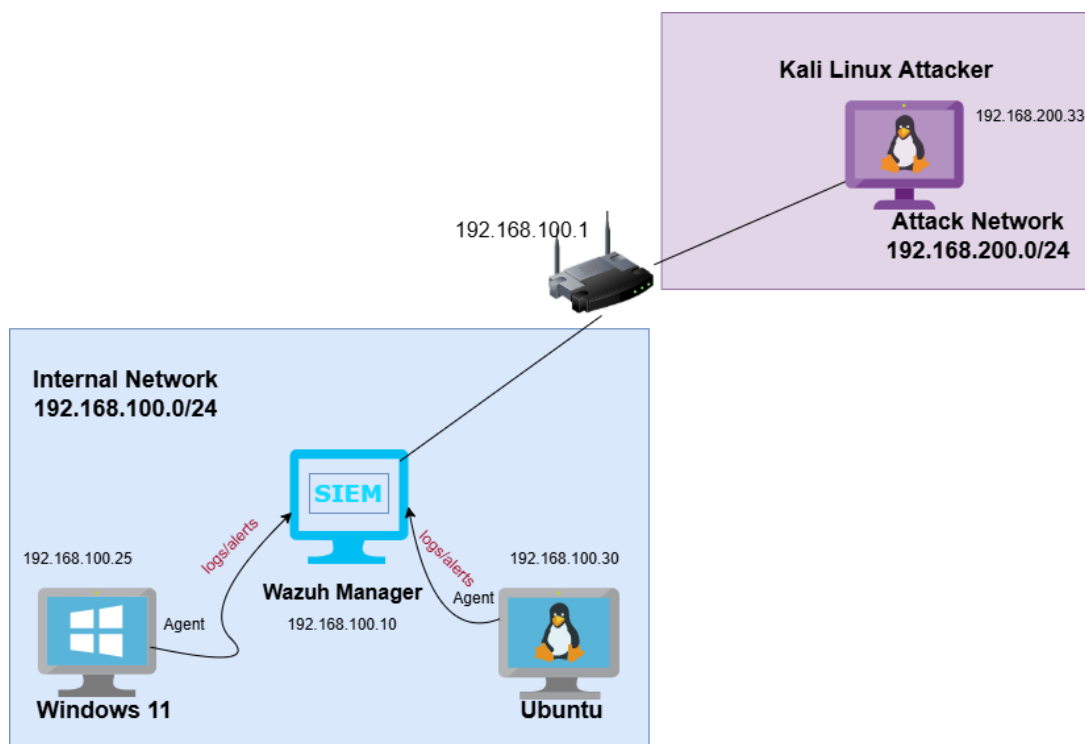


Рисунок 2.2 – Схема ІТ-інфраструктури

Для моделювання дій потенційного зловмисника виділено окрему зону атаки (Attack zone), у якій розміщено систему Kali Linux з IP-адресою 192.168.200.33/24. Винесення атакуючої сторони в окремий мережевий сегмент дозволяє імітувати зовнішні загрози та підвищує реалістичність тестування сценаріїв атак, таких як сканування портів, спроби несанкціонованого доступу та інші дії атакувального характеру.

Маршрутизація між мережевими сегментами здійснюється через шлюз з IP-адресою 192.168.100.1/24, який забезпечує обмін трафіком між зонами відповідно до заданих правил доступу. Забезпечення зв'язку між зонами відбувається через центральний маршрутизатор або програмний міжмережевий екран із кількома інтерфейсами, на якому налаштовується міжзональна фільтрація трафіку. Зокрема,

правила можуть дозволяти передачу журналів від клієнтів до сервера Wazuh (наприклад, через TCP-порт 1514), обмежувати доступ системи Kali Linux лише до клієнтських вузлів і блокувати прямі з'єднання з сервером управління.

Подібний підхід до зонування застосовується в лабораторних і навчальних проєктах, де виділяються окремі мережі для інструментів безпеки та для імітації атакуючого середовища. Така сегментація відповідає принципу найменших привілеїв (least privilege) та зменшує ризик бічного переміщення зловмисника між зонами, що підтверджується профільними дослідженнями.

Схема IP-адресації та розподіл вузлів за мережевими сегментами наведені в таблиці 2.1. Запропонована логічна модель мережі забезпечує централізований збір і аналіз подій безпеки, підвищує видимість мережових взаємодій та дозволяє ефективно досліджувати підозрілу активність у контрольованому середовищі без ризику для критичних компонентів системи.

Таблиця 2.1

Схема IP-адресації

Назва	IP-адреса	Опис
Wazuh Server	192.168.100.10/24	Центральний вузол збору, кореляції та аналізу подій
Windows 11	192.168.100.25/24	Моніторинг дій користувачів і подій ОС
Ubuntu 20.04 LTS	192.168.100.30/24	Сервер з IDS Suricata та NIDS Wireshark
Kali Linux	192.168.200.33/24	Модель атакуючої сторони
Gateway	192.168.100.1/24	Маршрутизація між сегментами

2.3 Обґрунтування вибору технологій та програмних компонентів

Проектування системи аналізу підозрілої активності в IT-інфраструктурі передбачає свідомий вибір технологій, які здатні забезпечити комплексне виявлення, аналіз та інтерпретацію подій безпеки. На цьому етапі важливо не зосереджуватися на конкретній реалізації чи бренді програмного забезпечення, а визначити клас технологій і їх функціональне призначення в межах загальної архітектури системи.

Основним критерієм вибору є відповідність таким вимогам, як – *виявлення підозрілої активності на хостах і в мережі, аналіз реакції системи на атаки та*

оцінка ефективності застосованих методів. Враховуючи це, система має включати кілька взаємодоповнюючих технологічних рівнів, кожен з яких виконує окрему функцію у процесі забезпечення безпеки[9].

Відповідно до підходів, розглянутих у першому розділі, система аналізу підозрілої активності будується за багаторівневою моделлю та включає рівень централізованого аналізу подій (SIEM), хостовий рівень моніторингу (HIDS) і мережевий рівень виявлення вторгнень (NIDS). У межах даного дослідження реалізація цих рівнів здійснюється з використанням конкретних програмних засобів, обґрунтування вибору яких наведено нижче.

Центральним елементом системи є платформа **Wazuh**, яка поєднує функції SIEM та хостової системи виявлення вторгнень. Wazuh забезпечує збір журналів подій з кінцевих систем, їх нормалізацію, зберігання та кореляційний аналіз. На цьому рівні реалізуються ключові компоненти системи аналізу підозрілої активності, а саме централізований збір і логування даних, сховище подій безпеки, механізми кореляції та первинного аналізу інцидентів. Використання Wazuh дозволяє фіксувати як окремі підозрілі події, так і їх сукупність у вигляді логічно пов'язаних сценаріїв атак.

Хост-орієнтований моніторинг у системі також реалізується за допомогою агентів Wazuh, які встановлюються на контрольовані вузли інфраструктури. Вони забезпечують аналіз журналів автентифікації, контроль цілісності файлів, моніторинг змін конфігурацій та виявлення аномальної поведінки на рівні операційної системи. Таким чином, Wazuh виконує функції HIDS і дозволяє виявляти наслідки атак після проникнення злоумисника, що є критично важливим для аналізу внутрішніх загроз та компрометації облікових записів[7].

Для розширення зони видимості системи та виявлення атак на ранніх етапах у роботі використовується мережевий засіб виявлення вторгнень **Suricata**, який реалізує рівень NIDS. Suricata здійснює аналіз мережевого трафіку в реальному часі та дозволяє виявляти типові мережеві атаки, зокрема сканування портів, спроби

експлуатації сервісів, підозрілу мережеву активність і використання відомих сигнатур атак. Події, згенеровані Suricata, передаються до платформи Wazuh, де вони корелюються з хостовими журналами та іншими подіями безпеки. Таким чином забезпечується зв'язок між мережевими індикаторами атак і їх наслідками на рівні кінцевих систем[2].

Механізми оцінки вразливостей та відповідності безпековим політикам у межах дослідження також реалізуються засобами Wazuh. Платформа дозволяє виявляти застаріле програмне забезпечення, небезпечні конфігурації та відхилення від базових рекомендацій з безпеки. Отримана інформація використовується як контекст для аналізу інцидентів, що підвищує точність інтерпретації подій та дозволяє пов'язувати зафіксовані атаки з конкретними слабкими місцями системи.

Компоненти візуалізації та звітності реалізуються за допомогою інструментів, інтегрованих у стек Wazuh (зокрема веб-інтерфейсу та панелей візуалізації). Вони дозволяють представити результати аналізу у вигляді графіків, таблиць і часових діаграм, що є важливим у дослідницькому контексті для аналізу динаміки атак і оцінки ефективності системи виявлення[18].

Рівень ручного аналізу мережевого трафіку із застосуванням **Wireshark** передбачено для глибокого дослідження окремих інцидентів. Цей інструмент допомагає аналітику переглядати деталі мережевих пакетів та перевіряти сигнали від NIDS. Включення Wireshark до автоматизованої архітектури не передбачене через трудомісткість та відсутність централізованого збору даних.

Загалом, у межах роботи реалізовано багаторівневу систему аналізу підозрілої активності, що включає SIEM/HIDS-рівень на базі Wazuh та мережевий рівень виявлення вторгнень на базі Suricata. Така архітектура забезпечує комплексний підхід до виявлення та аналізу як зовнішніх, так і внутрішніх загроз та створює основу для практичного тестування системи в умовах змодельованих кібератак, що буде продемонстровано в наступному розділі.

Висновки до другого розділу

У другому розділі розроблено концепцію та логічну архітектуру системи аналізу підозрілої активності в ІТ-інфраструктурі. Прийнято рішення про багаторівневу систему моніторингу, що поєднує централізований збір і кореляцію подій (SIEM), хостовий (HIDS) і мережевий (NIDS) рівні виявлення загроз.

Для централізованого аналізу подій обрано Wazuh, який забезпечує збір журналів, їх зберігання, кореляцію та візуалізацію. Хостовий рівень реалізовано агентами Wazuh, що контролюють цілісність файлів, журнали автентифікації та конфігурацій, виявляючи підозрілу активність. Мережевий рівень забезпечує Suricata, яка аналізує трафік і генерує події, інтегровані у Wazuh для повного контексту інцидентів.

Для поглибленого аналізу мережевого трафіку використано Wireshark, що дозволяє детально досліджувати окремі інциденти та перевіряти сигнали від NIDS.

Архітектура передбачає розмежування ролей сервера моніторингу, клієнтських систем і вузла-атакера, а також сегментацію мережі, що забезпечує коректний збір і аналіз даних. Обрані програмні засоби чітко реалізують функціональні компоненти системи аналізу підозрілої активності та створюють основу для практичної реалізації та тестування у третьому розділі.


```

wazuh-user@wazuh-server ~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:e8:c6:c2 brd ff:ff:ff:ff:ff:ff
    altname emp0s9
    inet 192.168.100.10/24 brd 192.168.100.255 scope global eth0
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe08:c6c2/64 scope link proto kernel_ll
        valid_lft forever preferred_lft forever
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:0e:17:5c brd ff:ff:ff:ff:ff:ff
    altname emp0s17
    inet 10.0.2.15/24 metric 1024 brd 10.0.2.255 scope global dynamic eth1
        valid_lft 86196sec preferred_lft 86196sec
    inet6 fd17:625c:1037:2:a00:27ff:fe0e:175c/64 scope global dynamic mngtnpaddr noprefixroute
        valid_lft 86198sec preferred_lft 14198sec
    inet6 fe80::a00:27ff:fe0e:175c/64 scope link proto kernel_ll
        valid_lft forever preferred_lft forever
wazuh-user@wazuh-server ~$

```

Рисунок 3.2 – Налаштування IP-адресації на сервері Wazuh

Після розгортання серверної частини було виконано підключення клієнтських систем. У межах дослідження використовувалися дві кінцеві системи з різними операційними системами, що дозволяє оцінити універсальність підходу до збору та аналізу подій. На кожній клієнтській системі було встановлено агент програмного забезпечення, який забезпечує передачу журналів подій та телеметрії до центрального сервера.

На клієнтській системі під керуванням операційної системи Windows було виконано встановлення агента моніторингу, після чого налаштовано з'єднання із сервером аналізу безпеки. Основним параметром конфігурації є адреса сервера та порт для передачі подій, що забезпечує централізований збір інформації про безпекові події (рис. 3.3-3.5).

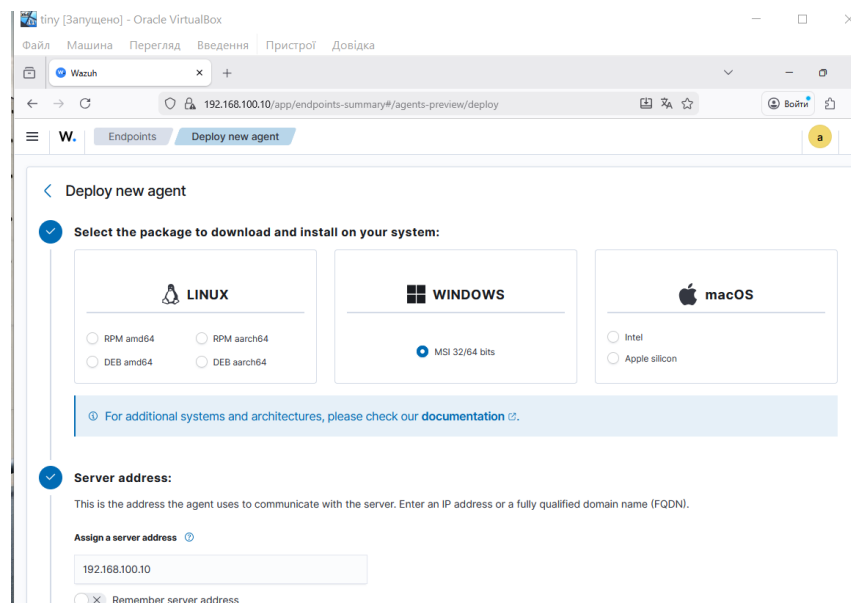


Рисунок 3.3 – Додавання агента моніторингу

Assign an agent name: [?](#)

[?](#) The agent name must be unique. It can't be changed once the agent has been enrolled. [?](#)

Select one or more existing groups: [?](#)

4 Run the following commands to download and install the agent:

```
Invoke-WebRequest -Uri https://packages.wazuh.com/4.x/windows/wazuh-agent-4.14.1-1.msi -OutFile $env:tmp\wazuh-agent; msixec.exe /i $env:tmp\wazuh-agent /q WAZUH_MANAGER='192.168.100.10' WAZUH_AGENT_NAME='win11-client'
```

Рисунок 3.4 – Додавання агента моніторингу

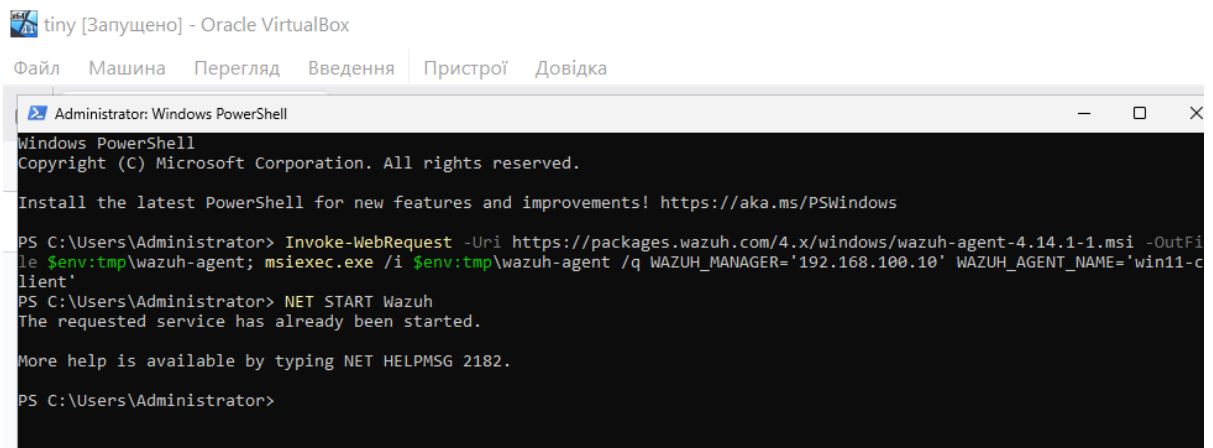


Рисунок 3.5 – Встановлення агента моніторингу

За допомогою додатка Wazuh Agent було запущено агента (рис. 3.6)

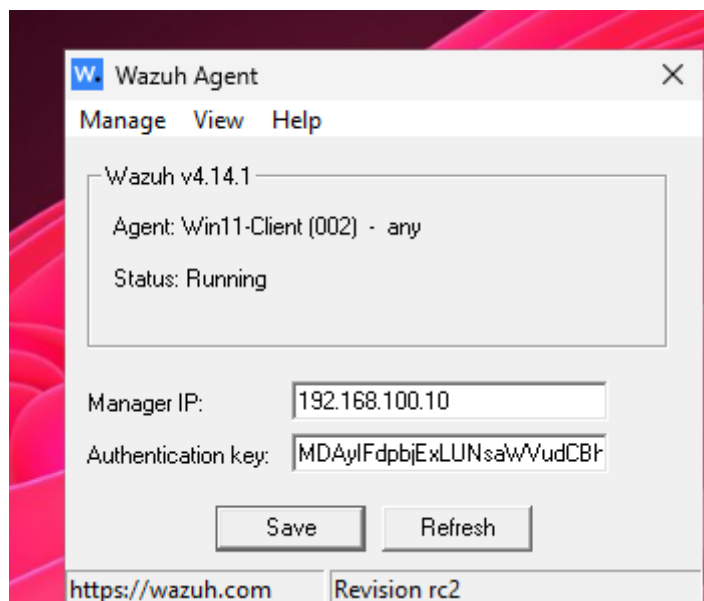


Рисунок 3.6 – Запуск агента моніторингу

Аналогічні дії було виконано для клієнтської системи під керуванням операційної системи Linux. Після встановлення агента було налаштовано передачу системних журналів, подій автентифікації та змін у файловій системі. Це дозволяє здійснювати порівняльний аналіз подій безпеки на різних платформах у межах єдиної системи моніторингу (рис. 3.7-3.9).

```
student@student-VirtualBox:~$ wget https://packages.wazuh.com/4.x/apt/pool/main/w/wazuh-agent/wazuh-agent_4.14.1-1_amd64.deb
--2025-12-13 20:26:59-- https://packages.wazuh.com/4.x/apt/pool/main/w/wazuh-agent/wazuh-agent_4.14.1-1_amd64.deb
Визначення імені packages.wazuh.com (packages.wazuh.com)... 3.174.230.69, 3.174.230.3, 3.174.230.56, ...
Встановлення з'єднання з packages.wazuh.com (packages.wazuh.com)|3.174.230.69|:443... з'єднано.
HTTP-запит надіслано, очікуємо на відповідь... 200 OK
Довжина: 13112530 (13M) [application/vnd.debian.binary-package]
Зберігаємо до «wazuh-agent_4.14.1-1_amd64.deb»

wazuh-agent_4.14.1-1_amd64.deb 100%[=====] 12,50M 4,72MB/s за 2,6s
2025-12-13 20:27:02 (4,72 MB/s) - «wazuh-agent_4.14.1-1_amd64.deb» збережено [13112530/13112530]

[sudo] пароль до student:
Вибір раніше не обраного пакунку wazuh-agent.
(Читання бази даних ... на дану мить встановлено 205106 файлів та каталогів.)
Приготування до розпакування .../wazuh-agent_4.14.1-1_amd64.deb ...
Розпакування wazuh-agent (4.14.1-1)...
Налаштування wazuh-agent (4.14.1-1) ...
Обробка тригерів systemd (245.4-4ubuntu3.2)...
student@student-VirtualBox:~$ sudo systemctl daemon-reload
student@student-VirtualBox:~$ sudo systemctl enable wazuh-agent
Created symlink /etc/systemd/system/multi-user.target.wants/wazuh-agent.service → /lib/systemd/system/wazuh-agent.service.
student@student-VirtualBox:~$ sudo systemctl start wazuh-agent
```

Рисунок 3.7 – Встановлення пакунків Wazuh на Ubuntu

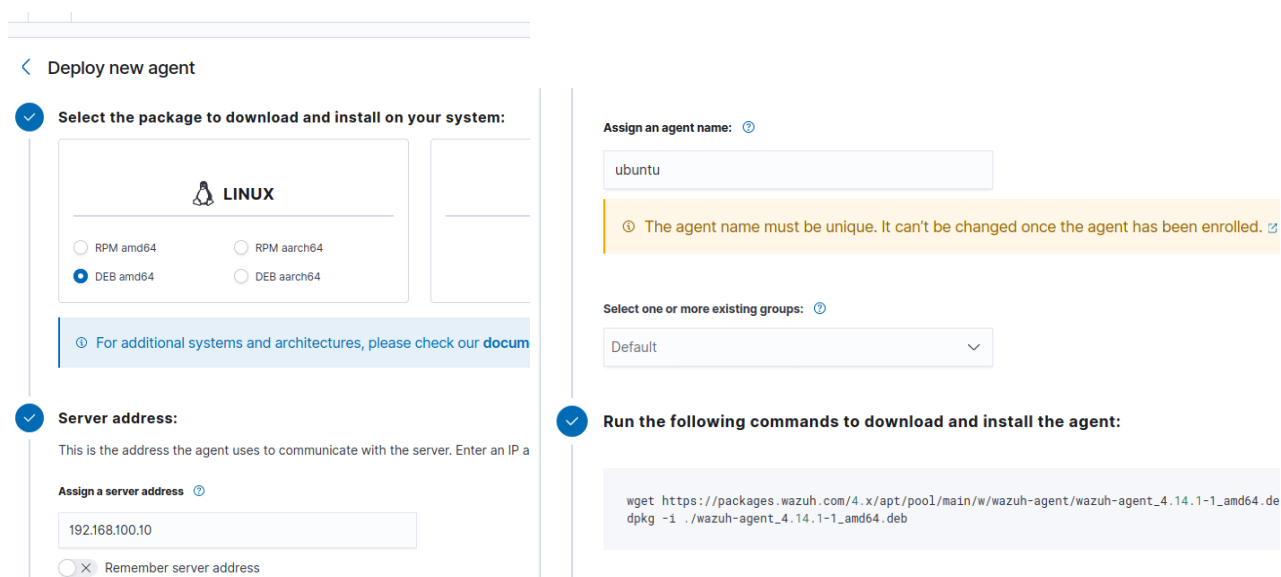


Рисунок 3.8-3.9 – Додавання агента моніторингу

Після підключення клієнтів було виконано перевірку коректності обміну даними між усіма компонентами системи. Для цього аналізувався статус підключених агентів на сервері, а також надходження перших подій безпеки у центральну систему аналізу. Успішне відображення клієнтів та подій підтвердило правильність налаштувань мережевої взаємодії та функціонування системи в цілому (рис. 3.10).

Agents (2)								⊕ Deploy new agent	🔄 Refresh	📄 Export fo
status=active										
☐	ID ↑	Name	IP address	Group(s)	Operating system	Cluster node	Version			
☐	002	Win11-Client	192.168.100.25	default	Microsoft Windows 11 Home 10.0.26200.5670	node01	v4.14.			
☐	004	ubuntu	192.168.100.30	default	Ubuntu 20.04.1 LTS	node01	v4.14.			

Рисунок 3.10 – Агенти

Для забезпечення виявлення підозрілої активності на мережевому рівні в межах досліджуваної інфраструктури на Ubuntu-сервері було розгорнуто мережеву систему виявлення вторгнень (IDS). Даний сервер виконує роль сенсора мережевого трафіку та доповнює хост-орієнтований моніторинг, реалізований на клієнтських системах.

У процесі розгортання було виконано встановлення IDS та базову конфігурацію її роботи відповідно до особливостей тестового середовища (рис. 3.11). Система налаштована на аналіз мережевого трафіку інтерфейсу, через який проходить взаємодія між атакуючою стороною та внутрішніми хостами.

```
student@student-VirtualBox:~$ sudo add-apt-repository ppa:oisf/suricata-stable
Suricata IDS/IPS/NSM stable packages
https://suricata.io/
https://oisf.net/

Suricata IDS/IPS/NSM - Suricata is a high performance Intrusion Detection and Prevention System and Network Security Monitoring engine.

Open Source and owned by a community run non-profit foundation, the Open Information Security Foundation (OISF). Suricata is developed by the OISF, its supporting vendors and the community.

This Engine supports:
- Multi-Threading - provides for extremely fast and flexible operation on multicore
```

Рисунок 3.11 – Встановлення репозиторію Suricata

Для підвищення ефективності виявлення атак було завантажено та активовано набір правил виявлення Emerging Threats, орієнтований на поширені типи мережевих загроз, зокрема сканування портів, спроби перебору облікових даних та використання відомих мережевих сигнатур атак (рис. 3.12-3.13) [22].

```

student@student-VirtualBox:/tmp$ curl -LO https://rules.emergingthreats.net/open/suricata-6
.0.8/emerging.rules.tar.gz
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
           Dload  Upload   Total   Spent    Left     Speed
100 5152k  100 5152k    0     0  152k      0  0:00:33  0:00:33 --:--:-- 549k
student@student-VirtualBox:/tmp$ sudo mkdir -p /etc/suricata/rules
student@student-VirtualBox:/tmp$ sudo tar -xvzf emerging.rules.tar.gz
rules/
rules/BSD-License.txt
rules/LICENSE
rules/boficc_portgrouped.rules

student@student-VirtualBox:/tmp$ sudo mkdir -p /etc/suricata/rules
student@student-VirtualBox:/tmp$ sudo cp etrules/rules/*.rules /etc/suricata/rules/
student@student-VirtualBox:/tmp$ sudo chmod 644 /etc/suricata/rules/*.rules

```

Рисунок 3.12-3.13 – Встановлення правил виявлення Emerging Threats

Наступним кроком є зміна налаштування Suricata у файлі `/etc/suricata/suricata.yaml` на власні. (рис. 3.14)

HOME_NET: "192.168.100.0/24"

EXTERNAL_NET: "any"

```

student@student-VirtualBox: /tmp
GNU nano 4.8 /etc/suricata/suricata.yaml
%YAML 1.1
---
# Suricata configuration file. In addition to the comments describing all
# options in this file, full documentation can be found at:
# https://docs.suricata.io/en/latest/configuration/suricata-yaml.html
# This configuration file was generated by Suricata 8.0.2.
suricata-version: "8.0"

##
## Step 1: Inform Suricata about your network
##

vars:
  # more specific is better for alert accuracy and performance
  address-groups:
    HOME_NET: "192.168.100.0/24"
    #HOME_NET: "[192.168.0.0/16]"
    #HOME_NET: "[10.0.0.0/8]"
    #HOME_NET: "[172.16.0.0/12]"
    #HOME_NET: "any"

    #EXTERNAL_NET: "!$HOME_NET"
    EXTERNAL_NET: "any"

```

Рисунок 3.14 – Зміна файлу `/etc/suricata/suricata.yaml`

Надалі в файлі необхідно вказати шлях до правил: (рис. 3.15):

default-rule-path: `/etc/suricata/rules`

rule-files:

- `"*.rules"`

```

##
## Configure Suricata to load Suricata-Update managed rules.
##

default-rule-path: /etc/suricata/rules

rule-files:
- "*.rules"

```

Рисунок 3.15 – Шлях до правил

У блоці capture (af-packet) задаємо інтерфейс, який Suricata має моніторити. (рис. 3.16).

```
# Linux high speed capture support
af-packet:
- interface: enp0s3
  # Number of receive threads. "auto" u
  #threads: auto
  # Default clusterid. AF_PACKET will l
```

Рисунок 3.16 – Інтерфейс

У файлі агента /var/ossec/etc/ossec.conf додати розділ <localfile> з шляхом до Suricata EVE-логу: (рис. 3.17).

```
student@student-VirtualBox: ~
GNU nano 4.8 /var/ossec/etc/ossec.conf
<localfile>
  <log_format>json</log_format>
  <location>/var/log/suricata/eve.json</location>
</localfile>
```

Рисунок 3.17 – Зміна файлу /var/ossec/etc/ossec.conf

У результаті виконання даного етапу було сформовано працездатну інфраструктуру аналізу підозрілої активності, що складається з центрального сервера аналізу безпеки та декількох клієнтських систем. Отримана конфігурація дозволяє перейти до наступного етапу дослідження — тестування роботи системи в умовах реальних сценаріїв атак та аналізу її реакції на підозрілі події.

3.2 Тестування виявлення підозрілої активності на різних рівнях

З метою перевірки ефективності розробленої системи аналізу підозрілої активності було проведено серію тестових сценаріїв, що імітують типові дії злоумисника в IT-інфраструктурі. Тестування виконувалося як на хостовому, так і на мережевому рівнях, що дозволило оцінити комплексність підходу до виявлення загроз.

У межах дослідження використовувалася змодельована лабораторна інфраструктура, яка включала сервер централізованого моніторингу, клієнтські системи

та окрему атакувальну машину. Усі дії атакуючої сторони виконувалися контролювано з метою аналізу реакції системи та фіксації відповідних подій безпеки.

Виявлення невдалих спроб автентифікації в ОС Windows

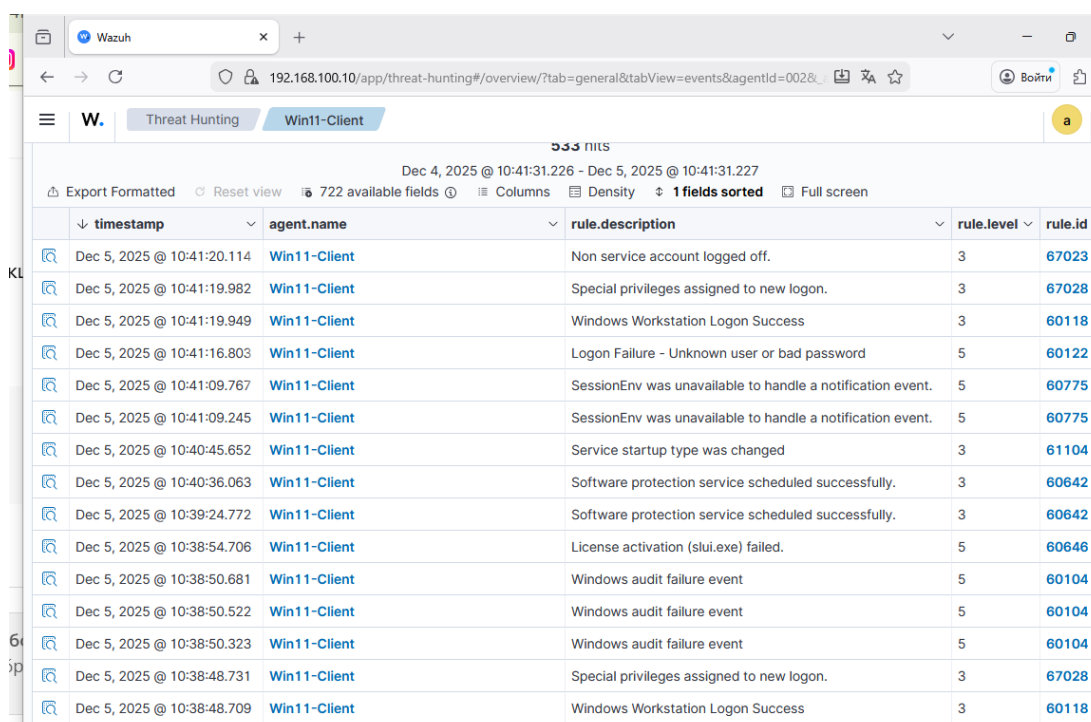
Першим етапом тестування стало моделювання підозрілої активності на хостовому рівні шляхом здійснення багаторазових невдалих спроб входу до операційної системи Windows 11. Такі дії є типовим прикладом атак на облікові записи та можуть свідчити про спроби підбору паролів або несанкціонованого доступу.

```
PS C:\Users\Administrator> net user testuser1 P@ssword123 /add
The command completed successfully.

PS C:\Users\Administrator> _
```

Рисунок 3.18 – Створення користувача на ОС Windows 11

Під час виконання тесту в журналі подій Windows були зафіксовані записи про помилки автентифікації, які автоматично передавалися агентом моніторингу на сервер аналізу подій. Система коректно ідентифікувала повторювані невдалі логіни та сформувала відповідні сповіщення безпеки.



The screenshot shows the Wazuh Threat Hunting interface in a web browser. The URL is 192.168.100.10/app/threat-hunting#/overview/?tab=general&tabView=events&agentId=002&... The interface displays a table of events for the agent 'Win11-Client'. The table has columns for timestamp, agent.name, rule.description, rule.level, and rule.id. The events include logon failures, session events, and service startup changes.

timestamp	agent.name	rule.description	rule.level	rule.id
Dec 5, 2025 @ 10:41:20.114	Win11-Client	Non service account logged off.	3	67023
Dec 5, 2025 @ 10:41:19.982	Win11-Client	Special privileges assigned to new logon.	3	67028
Dec 5, 2025 @ 10:41:19.949	Win11-Client	Windows Workstation Logon Success	3	60118
Dec 5, 2025 @ 10:41:16.803	Win11-Client	Logon Failure - Unknown user or bad password	5	60122
Dec 5, 2025 @ 10:41:09.767	Win11-Client	SessionEnv was unavailable to handle a notification event.	5	60775
Dec 5, 2025 @ 10:41:09.245	Win11-Client	SessionEnv was unavailable to handle a notification event.	5	60775
Dec 5, 2025 @ 10:40:45.652	Win11-Client	Service startup type was changed	3	61104
Dec 5, 2025 @ 10:40:36.063	Win11-Client	Software protection service scheduled successfully.	3	60642
Dec 5, 2025 @ 10:39:24.772	Win11-Client	Software protection service scheduled successfully.	3	60642
Dec 5, 2025 @ 10:38:54.706	Win11-Client	License activation (slui.exe) failed.	5	60646
Dec 5, 2025 @ 10:38:50.681	Win11-Client	Windows audit failure event	5	60104
Dec 5, 2025 @ 10:38:50.522	Win11-Client	Windows audit failure event	5	60104
Dec 5, 2025 @ 10:38:50.323	Win11-Client	Windows audit failure event	5	60104
Dec 5, 2025 @ 10:38:48.731	Win11-Client	Special privileges assigned to new logon.	3	67028
Dec 5, 2025 @ 10:38:48.709	Win11-Client	Windows Workstation Logon Success	3	60118

Рисунок 3.19 – Події Wazuh про невдалий вхід

Тестування brute-force атаки на мережеві сервіси

Наступним етапом стало моделювання brute-force атаки з використанням атакувальної системи Kali Linux. Атака була спрямована на сервіс віддаленого доступу, розгорнутий на клієнтській машині, та виконувалася за допомогою спеціалізованих інструментів для перебору облікових даних (рис. 3.20-3.21).

У процесі атаки система зафіксувала велику кількість однотипних спроб автентифікації за короткий проміжок часу. Ці події були об'єднані в єдиний логічний інцидент, що дозволило відрізнити атаку від випадкових помилок користувача (рис. 3.22-3.23) [24].

```
student@student-VirtualBox:~$ sudo systemctl enable --now ssh
Synchronizing state of ssh.service with SysV service script with /lib/systemd/sy
stemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable ssh
student@student-VirtualBox:~$ sudo adduser testuser
Додається користувач "testuser"...
Додається нова група "testuser" (1003)...
Додається новий користувач "testuser" (1001) з групою "testuser"...
Створюється домашня директорія "/home/testuser"...
Копіюються файли з "/etc/skel"..
Новий пароль :
Повторіть новий пароль :
passwd: пароль було успішно змінено
Зміна інформації про користувача testuser
Введіть нове значення або натисніть ENTER для значення за замовчанням
Повне ім'я []:
Номер кімнати []:
Робочий телефон []:
Домашній телефон []:
Інше []:
Чи є ця інформація вірною? [Т/Н]
student@student-VirtualBox:~$
```

Рисунок 3.20 – Створення користувача на ubuntu

```
(kali㉿kali)-[~]
$ hydra -l testuser -P /home/kali/pass.txt ssh://192.168.100.30
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in
military or secret service organizations, or for illegal purposes (this is n
on-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-12-13 17:
34:59
[WARNING] Many SSH configurations limit the number of parallel tasks, it is r
ecommended to reduce the tasks: use -t 4
[DATA] max 16 tasks per 1 server, overall 16 tasks, 19 login tries (l:1/p:19)
, ~2 tries per task
[DATA] attacking ssh://192.168.100.30:22/
[22][ssh] host: 192.168.100.30 login: testuser password: test123
1 of 1 target successfully completed, 1 valid password found
[WARNING] Writing restore file because 1 final worker threads did not complet
e until end.
[ERROR] 1 target did not resolve or could not be connected
[ERROR] 0 target did not complete
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-12-13 17:
35:06
```

Рисунок 3.21 – Реалізація brute force атаки

```

student@student-VirtualBox:~$ sudo tail -f /var/log/auth.log
Dec 14 00:35:06 student-VirtualBox sshd[6804]: Connection closed by authenticating user t
estuser 192.168.200.33 port 33138 [preauth]
Dec 14 00:35:06 student-VirtualBox sshd[6804]: PAM 1 more authentication failure; logname
= uid=0 euid=0 tty=ssh ruser= rhost=192.168.200.33 user=testuser
Dec 14 00:35:06 student-VirtualBox sshd[6800]: Connection closed by authenticating user t
estuser 192.168.200.33 port 33094 [preauth]
Dec 14 00:35:06 student-VirtualBox sshd[6800]: PAM 1 more authentication failure; logname
= uid=0 euid=0 tty=ssh ruser= rhost=192.168.200.33 user=testuser
Dec 14 00:35:06 student-VirtualBox sshd[6798]: Connection closed by authenticating user t
estuser 192.168.200.33 port 33078 [preauth]
Dec 14 00:35:06 student-VirtualBox sshd[6798]: PAM 1 more authentication failure; logname
= uid=0 euid=0 tty=ssh ruser= rhost=192.168.200.33 user=testuser
Dec 14 00:35:30 student-VirtualBox dbus-daemon[580]: [system] Failed to activate service
'org.bluez': timed out (service_start_timeout=25000ms)
Dec 14 00:35:57 student-VirtualBox sudo: pam_unix(sudo:session): session closed for user
root
Dec 14 00:36:19 student-VirtualBox sudo: student : TTY=pts/1 ; PWD=/home/student ; USER=
root ; COMMAND=/usr/bin/tail -f /var/log/auth.log
Dec 14 00:36:19 student-VirtualBox sudo: pam_unix(sudo:session): session opened for user
root by (uid=0)

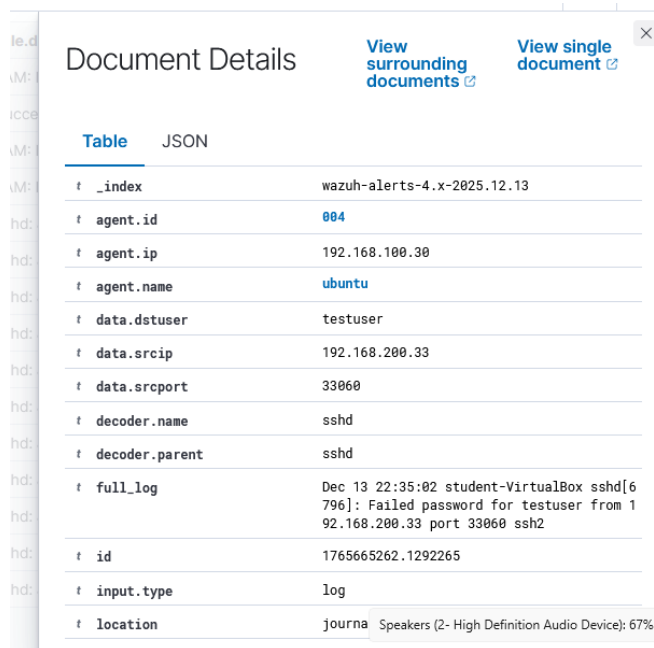
```

Рисунок 3.22 – Сповіщення про невдалі входи

timestamp	agent.name	rule.description	rule.level	rule.id
Dec 14, 2025 @ 00:35:40.164	ubuntu	PAM: Login session opened.	3	5501
Dec 14, 2025 @ 00:35:40.118	ubuntu	Successful sudo to ROOT executed.	3	5402
Dec 14, 2025 @ 00:35:18.102	ubuntu	PAM: Login session closed.	3	5502
Dec 14, 2025 @ 00:34:26.113	ubuntu	PAM: Login session closed.	3	5502
Dec 14, 2025 @ 00:34:26.107	ubuntu	sshd: authentication failed.	5	5760
Dec 14, 2025 @ 00:34:26.104	ubuntu	sshd: authentication failed.	5	5760
Dec 14, 2025 @ 00:34:26.096	ubuntu	sshd: authentication failed.	5	5760
Dec 14, 2025 @ 00:34:26.085	ubuntu	sshd: authentication failed.	5	5760
Dec 14, 2025 @ 00:34:22.039	ubuntu	sshd: authentication failed.	5	5760
Dec 14, 2025 @ 00:34:22.036	ubuntu	sshd: authentication failed.	5	5760
Dec 14, 2025 @ 00:34:22.033	ubuntu	sshd: authentication failed.	5	5760
Dec 14, 2025 @ 00:34:22.030	ubuntu	sshd: authentication failed.	5	5760
Dec 14, 2025 @ 00:34:22.027	ubuntu	sshd: authentication failed.	5	5760
Dec 14, 2025 @ 00:34:22.024	ubuntu	sshd: brute force trying to get access to the system....	10	5763
Dec 14, 2025 @ 00:34:22.021	ubuntu	sshd: authentication failed.	5	5760

Рисунок 3.23 – Події Wazuh про невдалий вхід та підозра на brute force атаку

Під час експерименту з підбору облікових даних через SSH система Wazuh зафіксувала багаторазові невдалі спроби автентифікації з однієї IP-адреси. На основі кореляції подій у журналі автентифікації було згенеровано сповіщення про атаку типу brute-force з підвищеним рівнем критичності, що підтверджує ефективність механізмів аналізу та кореляції подій у SIEM-системі.



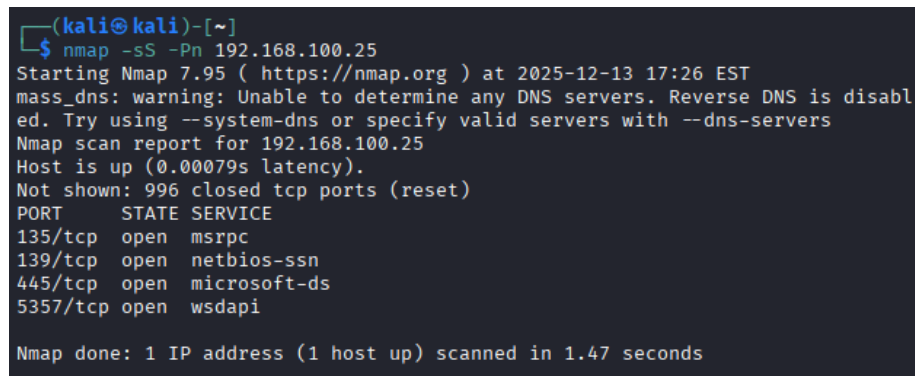
The screenshot shows a 'Document Details' window in Wazuh. It displays a table of fields for a specific alert. The fields include agent information (agent.id: 004, agent.name: ubuntu), source information (data.srcip: 192.168.200.33, data.srcport: 33060), decoder details (decoder.name: sshd), and the full log message: 'Dec 13 22:35:02 student-VirtualBox sshd[6796]: Failed password for testuser from 192.168.200.33 port 33060 ssh2'. Other fields like id, input.type, and location are also visible.

Field	Value
_index	wazuh-alerts-4.x-2025.12.13
agent.id	004
agent.ip	192.168.100.30
agent.name	ubuntu
data.dstuser	testuser
data.srcip	192.168.200.33
data.srcport	33060
decoder.name	sshd
decoder.parent	sshd
full_log	Dec 13 22:35:02 student-VirtualBox sshd[6796]: Failed password for testuser from 192.168.200.33 port 33060 ssh2
id	1765665262.1292265
input.type	log
location	Journalctl -o short -n 1000 -u sshd.service

Рисунок 3.24 – Сповіщення про невдалий вхід

Виявлення мережевої розвідки (nmap)

Для перевірки можливостей мережевого моніторингу було виконано сканування портів і сервісів із використанням інструменту nmap. Подібні дії є типовим етапом підготовки атаки та дозволяють зловмиснику зібрати інформацію про доступні сервіси й потенційні вразливості.



```
(kali@kali)-[~]
$ nmap -sS -Pn 192.168.100.25
Starting Nmap 7.95 ( https://nmap.org ) at 2025-12-13 17:26 EST
mass_dns: warning: Unable to determine any DNS servers. Reverse DNS is disabled. Try using --system-dns or specify valid servers with --dns-servers
Nmap scan report for 192.168.100.25
Host is up (0.00079s latency).
Not shown: 996 closed tcp ports (reset)
PORT      STATE SERVICE
135/tcp    open  msrpc
139/tcp    open  netbios-ssn
445/tcp    open  microsoft-ds
5357/tcp   open  wsdapi
Nmap done: 1 IP address (1 host up) scanned in 1.47 seconds
```

Рисунок 3.25 – Сканування відкритих портів за допомогою nmap

У процесі експерименту було здійснено мережеву розвідку з використанням інструмента Nmap. Система мережевого виявлення вторгнень Suricata зафіксувала спроби сканування портів та сформувала відповідні сигнатурні повідомлення. Логи IDS були передані до SIEM-системи Wazuh, де події були централізовано зібрані та

відображені у вигляді інцидентів безпеки, що підтверджує ефективність інтеграції IDS та SIEM [23].

```
student@student-VirtualBox:~$ sudo tail -f /var/log/suricata/fast.log
12/14/2025-00:26:37.663489    ** [1:2010937:3] ET SCAN Suspicious inbound to MySQL port 3
306 ** [Classification: Potentially Bad Traffic] [Priority: 2] {TCP} 192.168.200.33:636
48 -> 192.168.100.25:3306
12/14/2025-00:26:37.670296    ** [1:2010935:3] ET SCAN Suspicious inbound to MSSQL port 1
433 ** [Classification: Potentially Bad Traffic] [Priority: 2] {TCP} 192.168.200.33:636
48 -> 192.168.100.25:1433
12/14/2025-00:26:37.674119    ** [1:2010936:3] ET SCAN Suspicious inbound to Oracle SQL p
ort 1521 ** [Classification: Potentially Bad Traffic] [Priority: 2] {TCP} 192.168.200.3
3:63648 -> 192.168.100.25:1521
12/14/2025-00:26:38.770867    ** [1:2010935:3] ET SCAN Suspicious inbound to MSSQL port 1
433 ** [Classification: Potentially Bad Traffic] [Priority: 2] {TCP} 192.168.200.33:636
50 -> 192.168.100.25:1433
12/14/2025-00:26:38.854158    ** [1:2002910:6] ET SCAN Potential VNC Scan 5800-5820 **
[Classification: Attempted Information Leak] [Priority: 2] {TCP} 192.168.200.33:63648 ->
192.168.100.25:5801
12/14/2025-00:26:38.923789    ** [1:2010939:3] ET SCAN Suspicious inbound to PostgreSQL p
ort 5432 ** [Classification: Potentially Bad Traffic] [Priority: 2] {TCP} 192.168.200.3
3:63648 -> 192.168.100.25:5432
```

Рисунок 3.26 – Журнал Suricata про потенційну nmap атаку

→ → 192.168.100.10/app/threat-hunting#/overview/?tab=general&view=events&agentId=0048...

W. Threat Hunting ubuntu

	timestamp	agent.name	rule.description	rule.level	rule.id
	Dec 14, 2025 @ 00:27:19.230	ubuntu	Suricata: Alert - ET SCAN Potential VNC Scan 5800-5820	3	86601
	Dec 14, 2025 @ 00:27:19.230	ubuntu	Suricata: Alert - ET SCAN Suspicious inbound to PostgreSQL p...	3	86601
	Dec 14, 2025 @ 00:27:17.238	ubuntu	Suricata: Alert - ET SCAN Suspicious inbound to PostgreSQL p...	3	86601
	Dec 14, 2025 @ 00:27:17.236	ubuntu	Suricata: Alert - ET SCAN Suspicious inbound to MSSQL port 1...	3	86601
	Dec 14, 2025 @ 00:27:17.234	ubuntu	Suricata: Alert - ET SCAN Suspicious inbound to MSSQL port 1...	3	86601
	Dec 14, 2025 @ 00:27:17.231	ubuntu	Suricata: Alert - ET SCAN Suspicious inbound to PostgreSQL p...	3	86601
	Dec 14, 2025 @ 00:27:17.229	ubuntu	Suricata: Alert - ET SCAN Suspicious inbound to MySQL port 3...	3	86601
	Dec 14, 2025 @ 00:27:17.229	ubuntu	Suricata: Alert - ET SCAN Suspicious inbound to Oracle SQL po...	3	86601
	Dec 14, 2025 @ 00:25:59.174	ubuntu	Suricata: Alert - ET SCAN Suspicious inbound to PostgreSQL p...	3	86601
	Dec 14, 2025 @ 00:25:59.173	ubuntu	Suricata: Alert - ET SCAN Potential VNC Scan 5800-5820	3	86601
	Dec 14, 2025 @ 00:25:59.170	ubuntu	Suricata: Alert - ET SCAN Suspicious inbound to MSSQL port 1...	3	86601
	Dec 14, 2025 @ 00:25:57.215	ubuntu	Suricata: Alert - ET SCAN Suspicious inbound to Oracle SQL po...	3	86601
	Dec 14, 2025 @ 00:25:57.215	ubuntu	Suricata: Alert - ET SCAN Suspicious inbound to MSSQL port 1...	3	86601
	Dec 14, 2025 @ 00:25:57.170	ubuntu	Suricata: Alert - ET SCAN Suspicious inbound to MySQL port 3...	3	86601
	Dec 14, 2025 @ 00:25:33.214	ubuntu	PAM: Login session opened.	3	5501

Rows per page: 15

Рисунок 3.27 – Журнал подій Wazuh

Для додаткової верифікації мережевої активності та детального аналізу переданих пакетів було використано інструмент Wireshark. За його допомогою здійснено перехоплення мережевого трафіку під час виконання сканування (рис. 3.28), що дозволило візуально підтвердити наявність характерних TCP SYN-запитів та інших ознак мережевої розвідки. Використання Wireshark у даному експерименті доповнює автоматизовані засоби виявлення та забезпечує глибший аналіз окремих мережевих інцидентів на пакетному рівні.

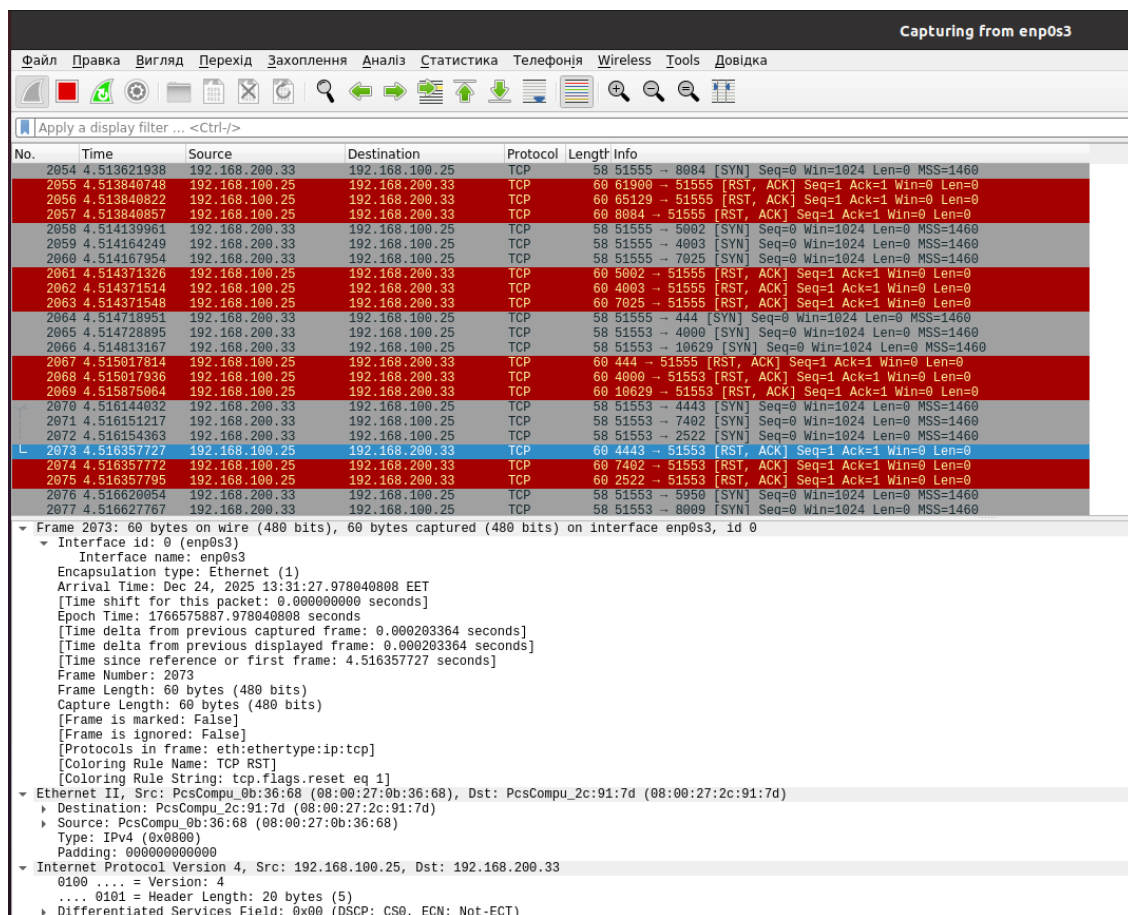


Рисунок 3.28 – Перехоплення мережевого трафіку за допомогоюWireshark

Перевірка реакції на тестовий шкідливий файл (EICAR)

Для перевірки можливостей виявлення шкідливого програмного забезпечення було використано тестовий файл EICAR, що є загальноприйнятим стандартом для перевірки антивірусних та IDS/IPS-систем. Після створення файлу система Wazuh зафіксувала його як потенційно небезпечний об'єкт та згенерувала відповідне сповіщення з високим рівнем критичності, що підтверджує коректну роботу механізмів виявлення шкідливого ПЗ (рис. 3.29).

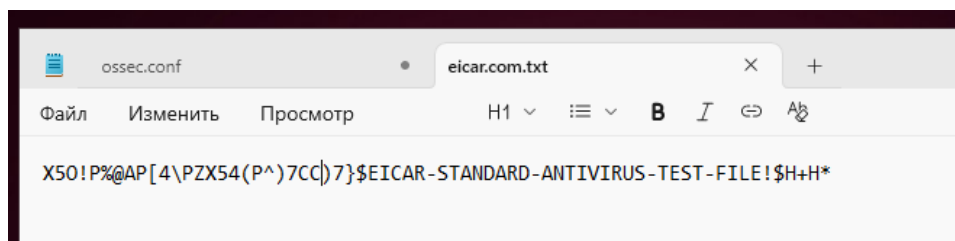


Рисунок 3.29 – Створення тестового файлу EICAR

Після створення файлу система зафіксувала відповідну подію та передала інформацію на сервер аналізу (рис. 3.30-3.31). Це підтвердило здатність системи реагувати на появу підозрілих об'єктів у файловій системі.

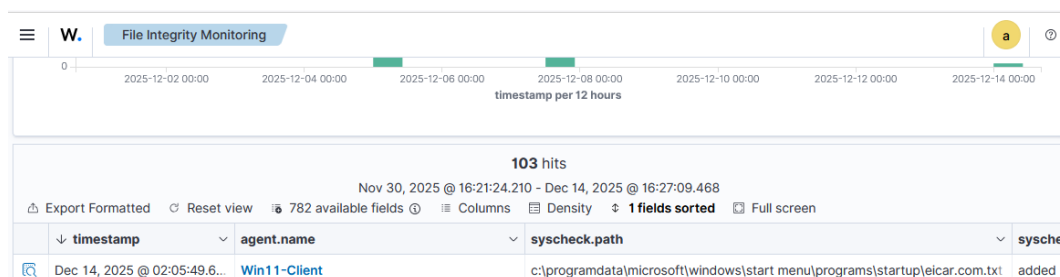


Рисунок 3.30 – Повідомлення роботи FIM про створення файлу

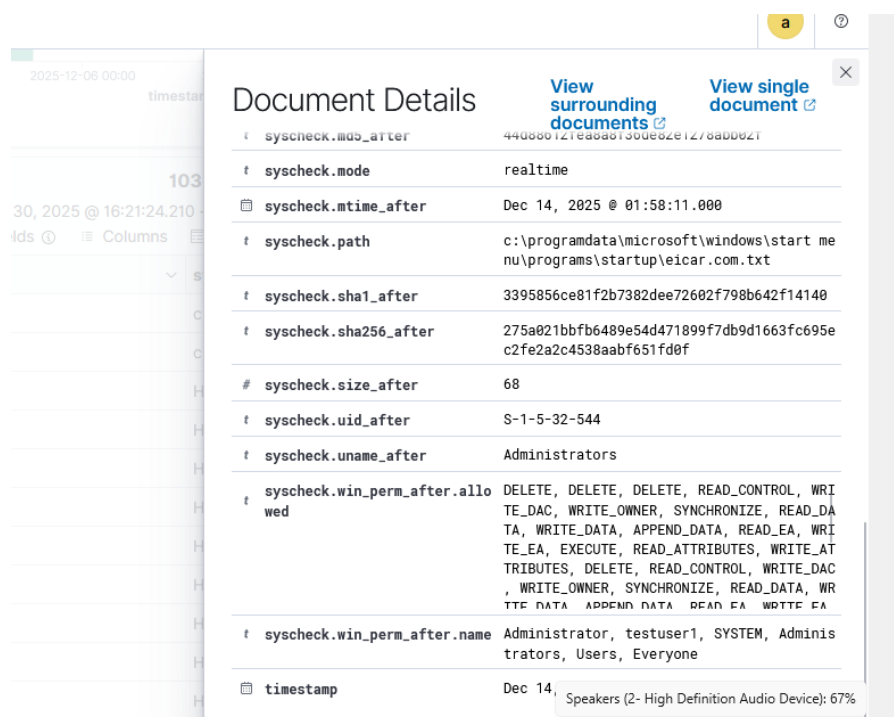


Рисунок 3.31 – Детальна інформація про подію

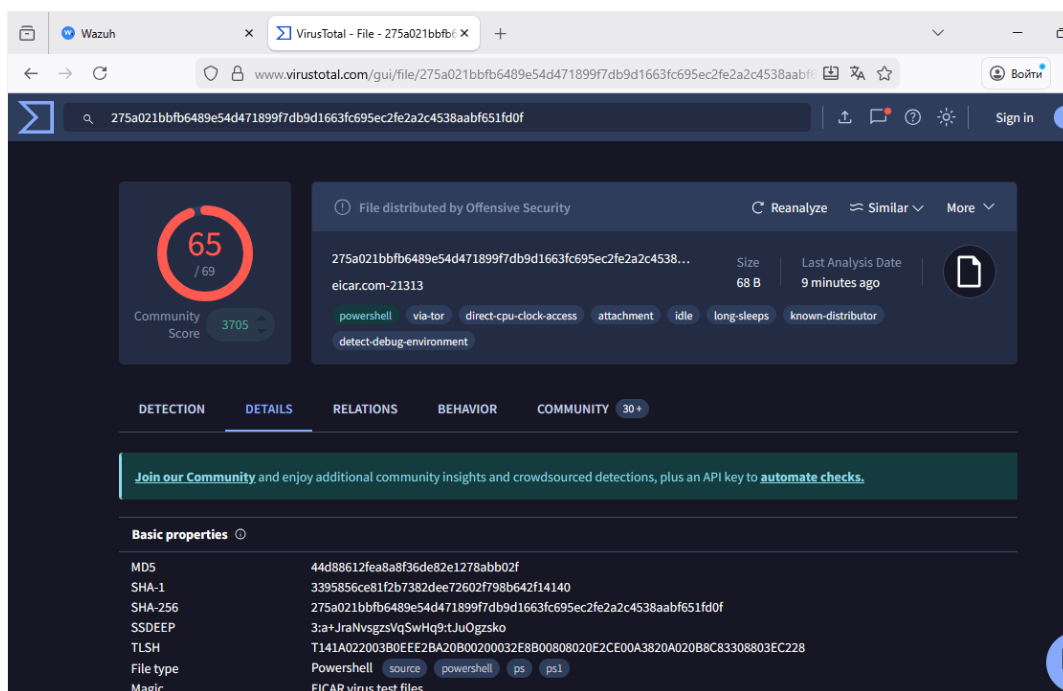


Рисунок 3.32 – Перевірка вмісту хеша сервісом VirusTotal

У ході експерименту було зафіксовано роботу модуля Vulnerability Detection системи Wazuh, який автоматично виявляє відомі CVE у встановленому програмному забезпеченні. Система не лише фіксує наявність вразливостей, але й коректно відображає факт їх усунення після оновлення компонентів, що підтверджує ефективність безперервного моніторингу стану безпеки хостів (рис. 3.33-3.35).

timestamp	agent.name	data.vulnerability.cve	data.vulnerability.severity	data.vulnerability.package
Dec 14, 2025 @ 01:12:32.773	ubuntu	CVE-2024-1724	High	snapt
Dec 14, 2025 @ 01:12:32.762	ubuntu	CVE-2022-3328	High	snapt
Dec 14, 2025 @ 01:12:32.752	ubuntu	CVE-2021-44731	High	snapt
Dec 14, 2025 @ 01:12:32.741	ubuntu	CVE-2021-4120	High	snapt
Dec 14, 2025 @ 01:12:32.735	ubuntu	CVE-2021-44730	High	snapt
Dec 14, 2025 @ 01:12:32.719	ubuntu	CVE-2024-29068	Medium	snapt
Dec 14, 2025 @ 01:12:32.708	ubuntu	CVE-2021-3155	Low	snapt
Dec 14, 2025 @ 01:12:32.697	ubuntu	CVE-2020-27352	High	snapt
Dec 14, 2025 @ 01:12:32.686	ubuntu	CVE-2023-1523	Critical	snapt
Dec 14, 2025 @ 01:12:32.676	ubuntu	CVE-2024-29069	High	snapt
Dec 14, 2025 @ 01:12:32.637	ubuntu	CVE-2021-41617	High	openssh-client
Dec 14, 2025 @ 01:12:32.626	ubuntu	CVE-2025-26465	Medium	openssh-client
Dec 14, 2025 @ 01:12:32.615	ubuntu	CVE-2025-32728	Low	openssh-client
Dec 14, 2025 @ 01:12:32.604	ubuntu	CVE-2021-28041	High	openssh-client
Dec 14, 2025 @ 01:12:32.593	ubuntu	CVE-2023-48795	Medium	openssh-client

Рисунок 3.33 – Vulnerability Detection

W.

Threat Hunting

ubuntu

a

	timestamp	agent.name	rule.description	rule.level	rule.id
	Dec 14, 2025 @ 01:12:32.686	ubuntu	The CVE-2023-1523 that affected snapd was solved due to an...	3	23502
	Dec 14, 2025 @ 01:12:32.676	ubuntu	The CVE-2024-29069 that affected snapd was solved due to a...	3	23502
	Dec 14, 2025 @ 01:12:32.637	ubuntu	The CVE-2021-41617 that affected openssh-client was solved...	3	23502
	Dec 14, 2025 @ 01:12:32.626	ubuntu	The CVE-2025-26465 that affected openssh-client was solved...	3	23502
	Dec 14, 2025 @ 01:12:32.615	ubuntu	The CVE-2025-32728 that affected openssh-client was solved...	3	23502
	Dec 14, 2025 @ 01:12:32.604	ubuntu	The CVE-2021-28041 that affected openssh-client was solved...	3	23502
	Dec 14, 2025 @ 01:12:32.593	ubuntu	The CVE-2023-48795 that affected openssh-client was solved...	3	23502
	Dec 14, 2025 @ 01:12:32.581	ubuntu	The CVE-2023-38408 that affected openssh-client was solved...	3	23502
	Dec 14, 2025 @ 01:12:32.571	ubuntu	The CVE-2023-51385 that affected openssh-client was solved...	3	23502
	Dec 14, 2025 @ 01:12:30.754	ubuntu	CVE-2023-50495 affects ncurses-term	7	23504

Рисунок 3.34-3.35 – Vulnerability Detection

Результати проведеного тестування свідчать про те, що система забезпечує виявлення підозрілої активності як на хостовому, так і на мережевому рівнях. Поєднання журналів операційних систем, мережевого аналізу та контролю цілісності файлів дозволяє формувати цілісне уявлення про інциденти безпеки та ефективно фіксувати дії атакуючої сторони на різних етапах атаки.

Таблиця 3.1

Тип атак та реакція системи

№	Тип атаки / дії	Джерело атаки	Зафіксована подія	Рівень виявлення	Реакція системи
1	Невдалі спроби входу в ОС Windows	Локальний користувач / Kali Linux	Багаторазові помилки автентифікації (Windows Event Log)	HIDS	Подія передана агентом на сервер аналізу, сформовано алерт про підозрілу активність
2	Brute-force атака на сервіс автентифікації	Kali Linux	Серія повторюваних спроб входу за короткий час	HIDS	Виявлено шаблон brute-force, події згруповано в інцидент
3	Сканування портів (nmap)	Kali Linux	Нетипова кількість мережевих з'єднань до різних портів	NIDS	IDS зафіксувала ознаки мережевої розвідки, згенеровано сповіщення
6	Зміна файлів у контрольованих каталогах	Windows / Ubuntu	Модифікація та створення файлів	HIDS (FIM)	Зафіксовано подію порушення цілісності файлів
7	Створення тестового файлу EICAR	Windows / Ubuntu	Виявлення підозрілого файлу	HIDS	Подія зареєстрована та передана для подальшого аналізу

8	Комплексна атака (мережа + хост)	Kali Linux	Кореляція мережевих та хостових подій	HIDS + NIDS	Забезпечено цілісне бачення інциденту безпеки
---	----------------------------------	------------	---------------------------------------	-------------	---

Наведені результати тестування підтверджують здатність розробленої системи виявляти підозрілу активність на різних рівнях IT-інфраструктури. Хост-орієнтовані механізми дозволяють фіксувати дії на рівні операційної системи, тоді як мережеві засоби забезпечують виявлення активних атак ще на етапі їх виконання. Кореляція подій з різних джерел підвищує точність виявлення інцидентів і зменшує ймовірність пропуску складних багатокрокових атак.

3.3 Аналіз виявлених інцидентів з використанням MITRE ATT&CK

Для систематизації та інтерпретації виявлених інцидентів безпеки в межах даного дослідження використано модель MITRE ATT&CK, яка описує типові тактики та техніки дій зловмисника під час кібератак. Застосування даної моделі дозволяє зіставити зафіксовані події безпеки з етапами атаки та оцінити характер підозрілої активності.

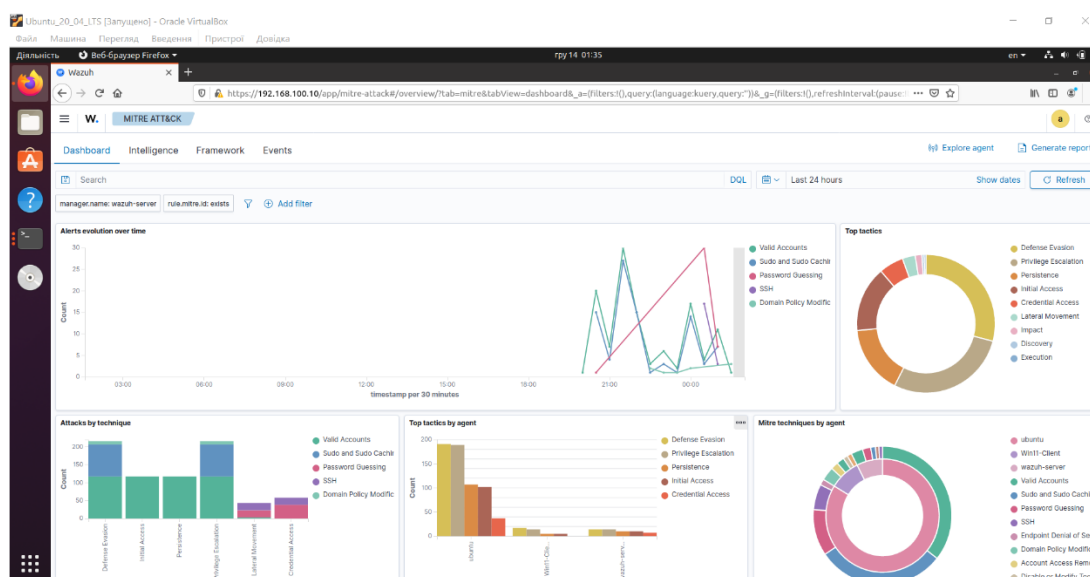


Рисунок 3.36 – Діаграми моделі MITRE ATT&CK

У процесі тестування було зафіксовано низку інцидентів, пов'язаних із несанкціонованими спробами доступу до систем. Зокрема, під час проведення атак пе-

ребору облікових даних та використання коректних, але скомпрометованих облікових записів, система моніторингу ідентифікувала відповідні техніки моделі MITRE ATT&CK.

Однією з таких технік є T1110.001 (Password Guessing), яка відповідає спробам перебору паролів шляхом багаторазових невдалих автентифікацій. Дані події були зафіксовані на рівні хостового моніторингу операційної системи Windows та коректно класифіковані в системі аналізу подій безпеки (рис. 3.37).

timestamp	agent.name	rule.mitre.id	rule.mitre.tactic	rule.description
Dec 14, 2025 @ 00:34:22.0...	ubuntu	T1110.001 T1021.004	Credential Access, Lateral Movement	sshd: authentication failed.
Dec 14, 2025 @ 00:34:22.0...	ubuntu	T1110.001 T1021.004	Credential Access, Lateral Movement	sshd: authentication failed.
Dec 14, 2025 @ 00:34:22.0...	ubuntu	T1110.001 T1021.004	Credential Access, Lateral Movement	sshd: authentication failed.
Dec 14, 2025 @ 00:34:22.0...	ubuntu	T1110.001 T1021.004	Credential Access, Lateral Movement	sshd: authentication failed.
Dec 14, 2025 @ 00:34:22.0...	ubuntu	T1110	Credential Access	sshd: brute force trying to get access to the s
Dec 14, 2025 @ 00:34:22.0...	ubuntu	T1110.001 T1021.004	Credential Access, Lateral Movement	sshd: authentication failed.
Dec 14, 2025 @ 00:34:22.0...	ubuntu	T1110.001 T1021.004	Credential Access, Lateral Movement	sshd: authentication failed.
Dec 14, 2025 @ 00:34:22.0...	ubuntu	T1110.001 T1021.004	Credential Access, Lateral Movement	sshd: authentication failed.
Dec 14, 2025 @ 00:34:22.0...	ubuntu	T1110.001 T1021.004	Credential Access, Lateral Movement	sshd: authentication failed.
Dec 14, 2025 @ 00:34:22.0...	ubuntu	T1110.001 T1021.004	Credential Access, Lateral Movement	sshd: authentication failed.
Dec 14, 2025 @ 00:34:22.0...	ubuntu	T1110.001 T1021.004	Credential Access, Lateral Movement	sshd: authentication failed.
Dec 14, 2025 @ 00:34:22.0...	ubuntu	T1110.001	Credential Access	PAM: User login failed.
Dec 14, 2025 @ 00:34:22.0...	ubuntu	T1110.001	Credential Access	PAM: User login failed.
Dec 14, 2025 @ 00:34:22.0...	ubuntu	T1110.001	Credential Access	PAM: User login failed.

Рисунок 3.37 – Виявлення підбору паролів T1110.001

Крім того, у результаті аналізу подій автентифікації було виявлено інциденти, що відповідають техніці T1078 (Valid Accounts), яка характеризує використання дійсних облікових даних для доступу до системи (рис. 3.38). Такі інциденти є особливо небезпечними, оскільки дії зловмисника можуть маскуватися під легітимну поведінку користувача [25].

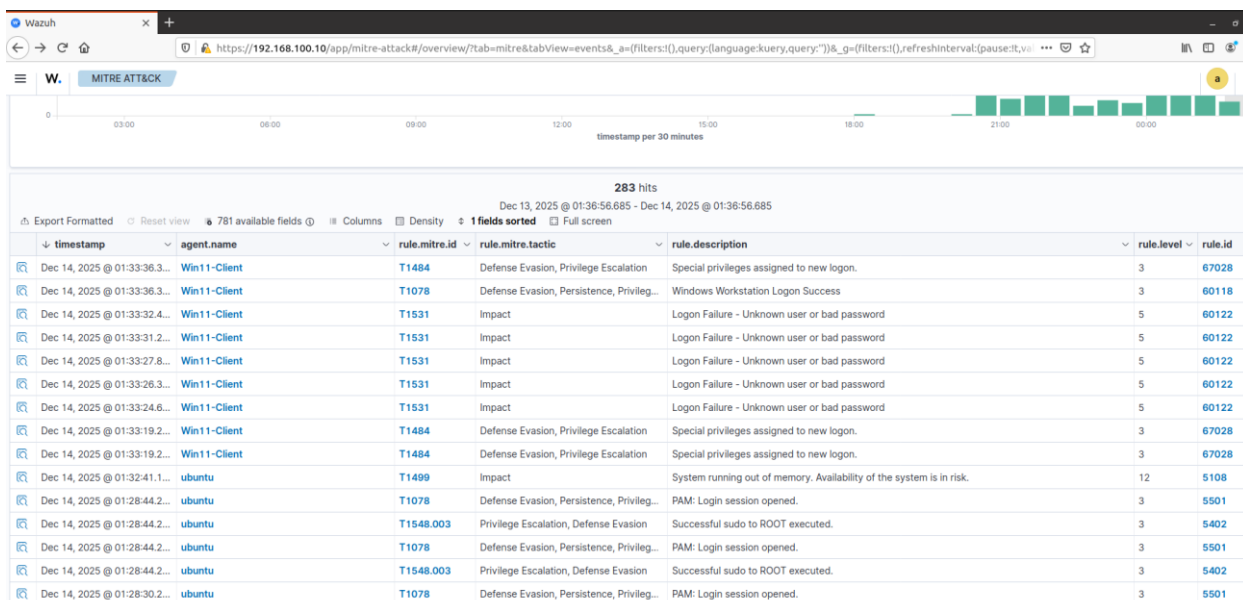


Рисунок 3.38 – Виявлення спроб входу T1078

Таблиця 3.2

Таблиця подій MITRE ATT&CK

Джерело	Подія	Опис	MITRE Tactic	MITRE Technique
Suricata	Nmap scan	Виявлено мережеву розвідку	Discovery	T1046
Wazuh	SSH failures	Перебір паролів	Credential Access	T1110.001
Wazuh	Login success	Спроба доступу	Initial Access	T1078

Хоча система Suricata не підтримує автоматичне зіставлення подій з тактиками MITRE ATT&CK, отримані сигнатури атак були проаналізовані та класифіковані вручну відповідно до моделі MITRE. Такий підхід відповідає практиці SOC-аналітиків і дозволяє пов'язати мережеві та хостові події у єдиний сценарій атаки.

Висновки до третього розділу

У третьому розділі було виконано розгортання системи моніторингу подій безпеки та підключення клієнтських хостів у тестовому середовищі. Проведено налаштування серверної частини системи, забезпечено збір логів з операційних систем Windows та Linux, а також інтеграцію з мережевими засобами виявлення атак.

У ході тестування було змодельовано різні сценарії підозрілої активності, зокрема мережеве сканування, спроби несанкціонованого доступу та атаки типу brute-force. Аналіз результатів показав, що система здатна виявляти події безпеки як на хостовому, так і на мережевому рівнях, а також забезпечує їх кореляцію та централізований аналіз.

Окрему увагу приділено аналізу виявлених інцидентів з використанням моделі MITRE ATT&CK, що дозволило класифікувати дії атакуючого та співвіднести їх з відомими тактиками і техніками атак. Отримані результати підтвердили ефективність комплексного підходу до аналізу підозрілої активності та досягнення поставленої мети роботи.

ВИСНОВКИ

У процесі виконання кваліфікаційної роботи було досягнуто поставленої мети – досліджено та реалізовано підхід до аналізу підозрілої активності в ІТ-інфраструктурі з використанням сучасних методів і технологій моніторингу подій безпеки.

У першому розділі проведено аналіз сучасних загроз інформаційній безпеці та розглянуто основні підходи до виявлення підозрілої активності в ІТ-інфраструктурі. Досліджено методи та технології аналізу подій безпеки, зокрема системи IDS/IPS, SIEM-платформи, засоби логування та моніторингу мережевого трафіку. Визначено їхні переваги та обмеження з огляду на практичне застосування у завданнях виявлення інцидентів безпеки, що дозволило обґрунтувати доцільність використання комплексного підходу до аналізу подій.

У другому розділі здійснено проектування логічної моделі системи виявлення підозрілої активності. Було визначено архітектуру рішення, підходи до збору та обробки журналів подій, а також логіку взаємодії основних компонентів системи. Прийнято рішення про багаторівневу систему моніторингу, що поєднує централізований збір і кореляцію подій (SIEM), а також HIDS та NIDS.

У третьому розділі реалізовано та експериментально перевірено розроблене рішення. У результаті проведених досліджень підтверджено, що система дозволяє ефективно виявляти підозрілу активність як на хостовому, так і на мережевому рівнях. Поєднання різних методів аналізу, кореляції подій та використання MITRE ATT&CK забезпечує комплексний підхід до виявлення та аналізу інцидентів інформаційної безпеки, що дозволяє своєчасно ідентифікувати потенційні загрози.

Таким чином, результати кваліфікаційної роботи підтверджують, що поставлені завдання виконано в повному обсязі, а отримані результати можуть бути використані як основа для впровадження систем моніторингу та виявлення підозрілої активності в реальних ІТ-інфраструктурах, а також для подальших досліджень у сфері інформаційної безпеки.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. CERT-UA минулого року опрацювала 4315 кіберінцидентів. URL: <https://cip.gov.ua/ua/news/cert-ua-minulogo-roku-opracyuvala-4315-kiberinci-dentiv>
2. Signature Based IDS vs Anomaly Based IDS: Understanding the Difference. Which is Best for Your Needs? URL: <https://fidelissecurity.com/cybersecurity-101/learn/signature-based-vs-anomaly-based-ids/>
3. Security Information and Event Management (SIEM) Guide. URL: <https://phoenixnap.com/blog/siem-security-information-event-management>
4. What Is UEBA? URL: <https://www.fortinet.com/uk/resources/cyberglossary/what-is-ueba>
5. Suricata vs snort: detailed guide to the programs. URL: <https://medium.com/@red-fanatic7/suricata-vs-snort-detailed-guide-to-the-programs-c331cff452a1>
6. Suricata vs Snort: A Comprehensive Review. URL: <https://www.stationx.net/suricata-vs-snort/>
7. The Best Free Open Source SIEM Tools. URL: <https://www.comparitech.com/net-admin/open-source-siem-tools/>
8. Graylog Security vs Splunk Enterprise Security Which SIEM Should I Choose? URL: <https://graylog.org/graylog-vs-splunk-siem/>
9. Top 13 Open Source SIEM Tools. URL: <https://aimultiple.com/open-source-siem>
10. Chapter 1. Introduction. URL: https://www.wireshark.org/docs/wsug_html_chunked/ChapterIntroduction.html
11. The Hidden Weapon in Your Network Security Arsenal: NetFlow. URL: <https://www.netflowlogic.com/netflow-for-network-security-a-powerful-tool-for-threat-detection-and-mitigation/>
12. Systemd-journald vs. syslog-ng - Blog - syslog-ng Community - syslog-ng Community. URL: <https://www.syslog-ng.com/community/b/blog/posts/systemd-journald-vs-syslog-ng>

13. Decoding Windows event logs: A definitive guide for incident responders. URL: <https://www.hackthebox.com/blog/decoding-windows-event-logs-a-definitive-guide-for-incident-responders>
14. How does journald differ from traditional syslog? URL: <https://fastneuron.com/forum/showthread.php?tid=6984>
15. The Open Source Security Platform. URL: <https://wazuh.com/>
16. What Is Zeek? URL: <https://docs.zeek.org/en/master/about.html>
17. OSSEC vs Wazuh - What's the Difference? URL: <https://trunc.org/ossec/ossec-vs-wazuh>
18. Top Cybersecurity Statistics: Facts, Stats and Breaches for 2025 URL: <https://www.fortinet.com/resources/cyberglossary/cybersecurity-statistics>
19. Wazuh · Security-Onion-Solutions/security-onion Wiki · GitHub URL: <https://github.com/security-onion-solutions/security-onion/wiki/Wazuh>
20. Virtual machine (VM) URL: <https://documentation.wazuh.com/current/deployment-options/virtual-machine/virtual-machine.html#import-and-access-the-virtual-machine>
21. Network IDS integration. URL: <https://documentation.wazuh.com/current/proof-of-concept-guide/integrate-network-ids-suricata.html>
22. Adding a Comprehensive Wazuh SIEM and Network Intrusion Detection System (NIDS) to the VirtualBox Lab. URL: <https://benheater.com/virtualbox-lab-wazuh-siem-and-nids/#:~:text=note%20the%20MAC%20address%2C%20as,Allow%20All>
23. Responding to network attacks with Suricata and Wazuh XDR. URL: <https://wazuh.com/blog/responding-to-network-attacks-with-suricata-and-wazuh-xdr/>
24. Detecting a brute-force attack. URL: <https://documentation.wazuh.com/current/proof-of-concept-guide/detect-brute-force-attack.html>

25. ATT&CK Matrix for Enterprise. URL: <https://attack.mitre.org/>
26. Introduction — Security Onion Documentation 2.4 documentation. URL: <https://docs.securityonion.net/en/2.4/introduction.html>
27. Setting Up a Comprehensive Open-Source Security Operations Center: A Practical Guide | E-SPIN Group. URL: <https://www.e-spincorp.com/documentation/open-source-soc-setup-guide/>
28. Do more for your organization with Elastic. URL: <https://www.elastic.co/cloud/platform-migration>
29. Infigo SIEM Content. URL: <https://classic.splunkbase.splunk.com/app/7175/>

ДОДАТКИ

Методи виявлення підозрілої активності

Підхід	Принцип дії	Переваги	Обмеження	Приклади застосування
Сигнатурний аналіз	Пошук відомих шаблонів атак у подіях; порівняння з базою наперед визначених сигнатур.[1]	Висока точність для відомих атак; швидке спрацювання; мінімум хибних позитивів на легітимні дії.[2]	Не виявляє нові/модифіковані загрози без оновлення бази; потребує регулярного оновлення правил; можливе обходження сигнатур.	Мережеві IDS типу Snort, Suricata; антивірусні сканери зі сигнатурними базами (AV).[5]
Евристичний поведінковий аналіз	Моніторинг поведінки користувачів, процесів, мережевого трафіку; виявлення відхилень від нормальної активності.[1]	Може виявити нові невідомі атаки за аномальними ознаками; адаптується до змін у середовищі; не залежить від бази сигнатур.[2]	Більше хибних спрацювань (аномалії не завжди означають атаку); потребує налаштування базових профілів; ресурсомісткість постійного аналізу.	Host-based HIDS/EDR з аналізом процесів; мережевий поведінковий аналіз (NBA) трафіку; евристичні модулі в антишкідливому ПЗ.
Статистичний аналіз і ML (UEBA)	Збір великих обсягів даних про активність; побудова статистичних моделей поведінки; застосування алгоритмів ШІ	Виявлення складних, багатокрокових та інсайдерських загроз; здатність ловити «тихі» атаки; зниження навантаження на аналітиків.	Висока складність і вартість впровадження; потрібен час для навчання моделей; потребує експертів для підтримки; генерує лише оповіщення (не блокує атаку).	UEBA-модулі в SIEM (Splunk UBA, QRadar User Behavior Analytics); окремі UEBA-платформи (Exabeam, Varonis); ML-алгоритми в системах моніторингу.[4]
Кореляція подій (SIEM-системи)	Централізований збір логів з різних джерел; аналіз та співставлення (кореляція) подій за наперед визначеними правилами і сценаріями атак.	Комплексний огляд усієї інфраструктури; виявлення мультиетапних атак, що проявляються на різних системах; зниження «шуму» через фільтрацію та пріоритизацію подій.	Залежність від повноти і якості вхідних даних та правил; потребує значних ресурсів (обробка великих логів); складність налаштування і підтримки; висока вартість для підприємств.	SIEM-платформи: Splunk Enterprise Security, IBM QRadar, ArcSight; відкриті рішення на базі ELK Stack (Elastic) + Wazuh; інші SOC-центри з кореляцією подій.[3]