

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 інструменти побудови систем реагування на інциденти в мережі на основі програмного стеку ELK»

студента: Голуб О.

затверджена Наказом по університету від 6 жовтня 2025 р. №566/с Термін здачі студентом закінченої роботи 26 грудня 2025 р.

Вихідні дані для роботи (визначається предмет і об'єкт дослідження):

Предметом дослідження є моделі

моніторингу та реагування на мережні інциденти на основі програмного стеку ELK та Wazuh. Об'єктом дослідження є процес виявлення інцидентів інформаційної безпеки в мережах.

Консультанти з випускної кваліфікаційної роботи із спеціальності 125 «Кібербезпека та захист інформації»

Розділ	Консультант	
1	Ігор Фальковський	06
2	Ігор Фальковський	10
3	Ігор Фальковський	07

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

Олексій ГОЛУБ

Старший викладач кафедри КІ та КБ,

Вадим МИКОЛАЙЧУК

Рецензент, завідувач кафедри комп'ютерної інженерії та кібербезпеки, к.т.н, доцент,

Андрій ЄФІМЕНКО

Студент Олексій ГОЛУБ

Керівник Вадим МИКОЛАЙЧУК

Житомир – 2025

Державний університет «Житомирська політехніка»

Факультет інформаційно-комп'ютерних технологій

Кафедра комп'ютерної інженерії та кібербезпеки

Спеціальність 125 «Кібербезпека та захист інформації»

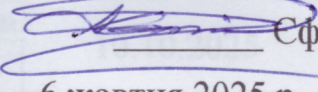
Освітня програма «Кібербезпека»

ЗАТВЕРДЖУЮ

Завідувач кафедри

комп'ютерної інженерії

та кібербезпеки

 Сфіменко А. А.

6 жовтня 2025 р.

ЗАВДАННЯ

на випускню кваліфікаційну роботу

Студента: Голуба Олексія Дмитровича Тема роботи: **Методи та інструменти побудови систем реагування на інциденти в мережі на основі програмного стеку ELK**

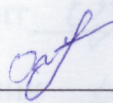
затверджена Наказом по університету від 6 жовтня 2025 р. №566/с Термін здачі студентом закінченої роботи 26 грудня 2025 р.

Вихідні дані для роботи (зазначається предмет і об'єкт дослідження):

Предметом дослідження є моделі, методи та програмні засоби побудови системи моніторингу та реагування на мережеві інциденти на основі програмного стеку ELK та Wazuh. Об'єктом дослідження є процес виявлення та реагування на інциденти інформаційної безпеки в комп'ютерних мережах.

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

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв
1	Ігор Фальковський	06.10.2025	06.10.2025
2	Ігор Фальковський	10.10.2025	10.10.2025
3	Ігор Фальковський	07.11.2025	07.11.2025

Студент  Олексій ГОЛУБ

Керівник  Вадим МИКОЛАЙЧУК

Календарний план

№ з/п	Назва етапів випускної кваліфікаційної роботи	Термін виконання етапів роботи	Примітка
1	Постановка завдання	06.10.2025	Виконано
2	Пошук, огляд та аналіз літературних джерел щодо стандартів реагування на інциденти (NIST, ISO/IEC 27035)	08.10.2025	Виконано
3	Аналіз принципів побудови систем моніторингу та класифікація інцидентів інформаційної безпеки	06.10.2025	Виконано
4	Дослідження проблем виявлення та обробки інцидентів у сучасних корпоративних мережах	10.10.2025	Виконано
5	Порівняльний аналіз архітектур сучасних SIEM-систем та можливостей стеку ELK	19.10.2025	Виконано
6	Розробка теоретичної моделі системи реагування та механізмів кореляції подій	05.11.2025	Виконано
7	Проектування моделі взаємодії компонентів (Beats, Logstash, Elasticsearch, Kibana, Wazuh)	07.11.2025	Виконано
8	Практична реалізація системи у віртуальному середовищі (GNS3, Docker)	17.11.2025	Виконано
9	Налаштування збору, нормалізації подій безпеки та побудова дашбордів моніторингу	25.11.2025	Виконано
10	Аналіз ефективності моделі за критеріями швидкості реагування, точності та навантаження	02.12.2025	Виконано
11	Розробка сценаріїв реагування на інциденти (Playbooks) та інтеграція з MITRE ATT&CK	09.12.2025	Виконано
12	Аналіз перспектив інтеграції системи з платформами SOAR	12.12.2025	Виконано
13	Написання пояснювальної записки	15.12.2025	Виконано
14	Підготовка презентації та виступу	22.12.2025	Виконано
15	Отримання відгуку керівника та рецензії на кваліфікаційну роботу	23.12.2025	Виконано
16	Захист	26.12.2025	Виконано

Студент ОГ Олексій ГОЛУБ

Керівник МБ Вадим МИКОЛАЙЧУК

АНОТАЦІЯ

Кваліфікаційна робота містить 99 сторінок, 20 ілюстрації, 2 таблиць, список використаних джерел із 30 найменувань та додатків.

Метою роботи є розробка та дослідження ефективності моделі системи реагування на інциденти інформаційної безпеки в мережі на основі програмного стеку ELK та інтегрованої платформи Wazuh.

У ході дослідження проведено аналіз теоретичних основ реагування на інциденти згідно з міжнародними стандартами (NIST, ISO/IEC 27035) та виявлено ключові проблеми моніторингу в сучасних корпоративних мережах. Особлива увага була приділена аналізу архітектурних можливостей стеку ELK та порівнянню сучасних SIEM-систем. Результатом цього етапу стала розробка моделі взаємодії компонентів системи (Beats, Logstash, Elasticsearch, Kibana/OpenSearch), спрямованої на автоматизований збір та кореляцію подій безпеки.

Проведено практичну реалізацію моделі у віртуальному середовищі з використанням контейнеризації Docker. Розглянуті методи нормалізації даних за стандартом ECS та побудова правил кореляції для виявлення складних атак. Проведено аналіз ефективності системи за критеріями швидкості реагування, точності детекції та обчислювального навантаження на інфраструктуру. Останній етап дослідження включає розробку практичних сценаріїв реагування (Playbooks) на типові інциденти, такі як Brute Force та використання шкідливих PowerShell-сценаріїв. Також висвітлені перспективи інтеграції системи з платформами SOAR та глобальною базою знань MITRE ATT&CK для переходу до проактивного захисту мережі. Розроблена модель та рекомендації є основою для побудови масштабованих центрів управління безпекою (SOC).

Ключові слова: СТЕК ELK, РЕАГУВАННЯ НА ІНЦИДЕНТИ, WAZUH, SIEM-СИСТЕМА, МОНІТОРИНГ МЕРЕЖІ, КОРЕЛЯЦІЯ ПОДІЙ.

ANNOTATION.

The qualification work contains 99 pages, 20 illustrations, 2 tables, and a list of 30 references and an appendix.

The aim of the work is to develop and research the effectiveness of a network incident response system model based on the ELK software stack and the integrated Wazuh platform.

In the course of the research, an analysis of the theoretical foundations of incident response according to international standards (NIST, ISO/IEC 27035) was conducted, and key monitoring problems in modern corporate networks were identified. Particular attention was paid to the analysis of the architectural capabilities of the ELK stack and the comparison of modern SIEM systems. The result of this stage was the development of a component interaction model (Beats, Logstash, Elasticsearch, Kibana/OpenSearch) aimed at automated collection and correlation of security events.

The practical implementation of the model was carried out in a virtual environment using Docker containerization. Methods for data normalization based on the ECS (Elastic Common Schema) standard and the construction of correlation rules for detecting complex attacks were considered. The effectiveness of the system was analyzed according to the criteria of response speed, detection accuracy, and computational load on the infrastructure. The final stage of the research includes the development of practical response scenarios (Playbooks) for typical incidents such as Brute Force and the use of malicious PowerShell scripts. The prospects for integrating the system with SOAR platforms and the MITRE ATT&CK global knowledge base for the transition to proactive network defense were also highlighted. The developed model and recommendations serve as a core for building scalable Security Operations Centers (SOC).

Keywords: ELK STACK, INCIDENT RESPONSE, WAZUH, SIEM SYSTEM, NETWORK MONITORING, EVENT CORRELATION.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ.....	7
ВСТУП	8
1. АНАЛІЗ ПРИНЦИПІВ ПОБУДОВИ ТА ПРОБЛЕМ ЗАБЕЗПЕЧЕННЯ РЕАГУВАННЯ НА ІНЦИДЕНТИ ІНФОРМАЦІЙНОЇ БЕЗПЕКИ.....	10
1.1. Поняття та класифікація інцидентів інформаційної безпеки.....	10
1.2. Етапи процесу реагування на інциденти згідно міжнародних стандартів (NIST, ISO/IEC 27035)	21
1.3. Аналіз принципів побудови систем моніторингу та реагування на інциденти ..	33
1.4. Основні проблеми виявлення та обробки інцидентів у сучасних корпоративних мережах.....	43
Висновки до першого розділу:	48
2. АНАЛІЗ ІСНУЮЧИХ ІНСТРУМЕНТІВ ТА ПРОЄКТУВАННЯ МОДЕЛІ СИСТЕМИ НА ОСНОВІ ПРОГРАМНОГО СТЕКУ ELK.....	49
2.1. Огляд і порівняння сучасних систем збору та аналізу подій безпеки (SIEM, IDS/IPS, SOAR)	49
2.2. Аналіз архітектури та можливостей програмного стеку ELK	56
2.3. Порівняльний аналіз систем логування та моніторингу Graylog, Wazuh і ELK-стеку	62
2.4. Розробка моделі системи реагування на інциденти на базі ELK.....	69
2.5. Механізми кореляції подій, формування правил виявлення та оповіщення.....	72
2.6. Модель взаємодії компонентів системи та архітектура потоків даних	73
Висновки до другого розділу:	76
3. РЕАЛІЗАЦІЯ ТА ОЦІНКА ЕФЕКТИВНОСТІ МОДЕЛІ СИСТЕМИ РЕАГУВАННЯ НА ІНЦИДЕНТИ.....	77
3.1. Практична реалізація системи на основі програмного стеку ELK.....	77
3.2. Збір і нормалізація подій безпеки з різних джерел, побудова правил кореляції та дашбордів для моніторингу інцидентів.....	82
3.3. Аналіз ефективності моделі за критеріями швидкості реагування, точності та навантаження	87
3.4. Приклади сценаріїв реагування на інциденти (Use Cases / Playbooks)	91
3.5. Перспективи інтеграції з платформами SOAR та MITRE ATT&CK	95
Висновки до третього розділу:	97
ВИСНОВКИ	99
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	100
ДОДАТКИ	101

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

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

ELK (Elasticsearch, Logstash, Kibana) – програмний стек для збору, аналізу та візуалізації даних

SOC (Security Operations Center) – центр управління безпекою. IDS (Intrusion Detection System) – система виявлення вторгнень. IPS (Intrusion Prevention System) – система запобігання вторгненням

SOAR (Security Orchestration, Automation, and Response) – засоби оркестрації, автоматизації та реагування на інциденти безпеки

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

FIM (File Integrity Monitoring) – моніторинг цілісності файлів

APT (Advanced Persistent Threat) – складна цілеспрямована загроза

MITRE ATT&CK – глобальна база знань про тактики та техніки атак зловмисників

ECS (Elastic Common Schema) – стандартна схема полів для нормалізації подій

API (Application Programming Interface) – програмний інтерфейс додатку

LotL (Living off the Land) – атаки з використанням легітимних інструментів системи

NIST (National Institute of Standards and Technology) – Національний інститут стандартів і технологій США

JVM (Java Virtual Machine) – віртуальна машина Java

ВСТУП

Актуальність. Системи реагування на інциденти інформаційної безпеки, які раніше розглядалися як допоміжні засоби моніторингу, сьогодні стали критично необхідними складовими захисту сучасної IT-інфраструктури. Завдяки технологічному розвитку та поширенню концепції Big Data, вони перетворилися на потужні аналітичні інструменти, що допомагають вирішувати широкий спектр завдань: від централізованого збору логів і виявлення аномалій до автоматизованого стримування складних кібератак у промислових та корпоративних масштабах. Разом із зростанням складності мережевих архітектур та переходом до хмарних технологій, гостро постало питання ефективності інструментів обробки подій безпеки.

В останні роки інтенсивність кібератак, спрямованих на корпоративні мережі, та витонченість методів проникнення стали дедалі актуальнішими проблемами. Сучасні загрози, такі як цілеспрямовані атаки використання легітимних інструментів системи та масовані спроби перебору паролів, вимагають від служб кібербезпеки впровадження систем, здатних проводити кореляцію подій у реальному часі. Недостатня видимість процесів усередині мережі або затримка в реагуванні можуть призвести до критичних витоків конфіденційних даних, фінансових збитків та репутаційних втрат для організацій.

Мета наукової роботи полягає в дослідженні принципів побудови систем реагування на інциденти та розробці моделі такої системи на основі програмного стеку ELK (Elasticsearch, Logstash, Kibana/OpenSearch) у поєднанні з платформою Wazuh. У роботі описано не лише теоретичні аспекти та міжнародні стандарти (NIST, ISO/IEC 27035), пов'язані з процесом менеджменту інцидентів, але й практичні етапи реалізації моделі, включаючи аналіз її ефективності та розробку сценаріїв автоматизованого реагування.

Об'єктом дослідження роботи є процеси виявлення та реагування на інциденти інформаційної безпеки в сучасних комп'ютерних мережах.

Предметом дослідження є модель системи моніторингу та обробки подій безпеки на основі програмного стеку ELK та спеціалізованих агентів збору

телеметрії, що забезпечують конфіденційність, цілісність та доступність інформаційних активів.

Методологія роботи полягає в практичній реалізації спроектованої моделі в ізольованому віртуальному середовищі з використанням технологій контейнеризації Docker. У ході дослідження розроблено практичні рекомендації для спеціалістів з інформаційної безпеки щодо налаштування правил кореляції та інтеграції системи з глобальними базами знань, такими як MITRE ATT&CK. Визначено майбутні тенденції розвитку систем реагування, зокрема перспективи впровадження технологій SOAR для повної оркестрації процесів захисту в умовах зростаючої складності сучасного кіберпростору.

1. АНАЛІЗ ПРИНЦИПІВ ПОБУДОВИ ТА ПРОБЛЕМ ЗАБЕЗПЕЧЕННЯ РЕАГУВАННЯ НА ІНЦИДЕНТИ ІНФОРМАЦІЙНОЇ БЕЗПЕКИ

1.1. Поняття та класифікація інцидентів інформаційної безпеки

У сучасних умовах розвитку цифрової інфраструктури інциденти інформаційної безпеки перетворилися на один із найважливіших факторів, що визначають стійкість інформаційних систем до зовнішнього та внутрішнього впливу. Стрімке зростання кількості кіберзагроз, поява нових моделей атак, розширення поверхні ураження корпоративних мереж, активне впровадження хмарних платформ і сервісів, підключення мобільних пристроїв та елементів «інтернету речей» формують багатовимірне середовище, в якому будь-яка подія, пов'язана з порушенням правил функціонування системи, може набути характеру інциденту. У цьому контексті поняття інциденту інформаційної безпеки є фундаментальним для побудови цілісної системи реагування, оскільки саме від коректного розуміння суті, природи та типології таких інцидентів залежить вибір відповідних методів їх виявлення, аналізу та нейтралізації.

У науковій літературі та міжнародних стандартах інцидент інформаційної безпеки традиційно визначається як подія або сукупність подій, що може призвести або вже призвела до порушення конфіденційності, цілісності чи доступності інформації, а також до зниження рівня захищеності інформаційної системи [3,4]. Однак таке визначення має значно складніший зміст, оскільки інцидент – це не лише прояв порушення, а й маркер наближення потенційної загрози. Багато інцидентів не мають негайних руйнівних наслідків, проте свідчать про наявність попередніх етапів реалізації атаки. У цьому випадку інцидент стає частиною ланцюга, який зловмисник вибудовує з метою отримання доступу або пошкодження системи. Таким чином, інцидент включає як прямі, так і непрямі ознаки порушень, що ускладнює процес його ідентифікації та вимагає комплексного підходу до аналізу [6].

Інциденти інформаційної безпеки можна розглядати як прояви взаємодії між системою та агресивним зовнішнім середовищем, де кожен інцидент має власний контекст, динаміку, інтенсивність та потенційні наслідки. Значна частина таких інцидентів виникає внаслідок навмисних дій, що здійснюються з порушенням

встановлених політик безпеки, зокрема спроби несанкціонованого доступу, маніпулювання даними, вторгнення до мережових сегментів, використання вразливостей у програмному забезпеченні, виконання шкідливого коду. Проте інша частка інцидентів пов'язана з ненавмисними діями персоналу, помилками у конфігураціях, технічними збоями, програмними помилками та некоректним використанням інформаційних ресурсів. Відповідно, інциденти мають різну природу, різні механізми виникнення та різні шляхи їх подолання.

Однією з найважливіших характеристик інциденту є його вплив на основні властивості інформації. Порушення конфіденційності відбувається тоді, коли інформація стає доступною особам чи системам, що не мають відповідних повноважень. Це може проявлятися у формі витоку даних, перехоплення трафіку, експлуатації слабких механізмів шифрування або несанкціонованого копіювання інформації. Порушення цілісності означає зміну, пошкодження або видалення даних без дозволу, що часто робить інформацію недостовірною або непридатною для використання. Наслідком таких інцидентів може бути як збої у роботі критичних бізнес-процесів, так і серйозні фінансові втрати. Інциденти порушення доступності передбачають зниження або повне припинення можливості використання інформаційних ресурсів. Найбільш типовими проявами є DDoS-атаки, аварії серверів, пошкодження мережевої інфраструктури або відмова систем зберігання даних. Подібні події створюють особливо критичні умови, оскільки безпосередньо впливають на безперервність функціонування організації.

У сучасних корпоративних мережах інциденти часто не мають чітко вираженої категорії та можуть одночасно впливати на кілька властивостей інформаційної безпеки. Наприклад, компрометація облікового запису користувача спричиняє одночасно ризики конфіденційності, оскільки зловмисник отримує доступ до даних; цілісності – через можливість змінювати інформацію; та доступності – у разі блокування легітимного користувача або модифікації критичних системних параметрів. Саме складність та багатогранність сучасних інцидентів роблять їх виявлення без спеціальних засобів надзвичайно складним.

Усе більшої актуальності набуває класифікація інцидентів за джерелом виникнення. Зовнішні інциденти зумовлені діями суб'єктів, що не мають законного

доступу до системи. Вони зазвичай здійснюються організованими хакерськими групами, кіберзлочинцями, конкурентами або геополітичними акторами. Такі інциденти мають високий рівень складності та часто реалізуються у кілька етапів, що дає змогу зловмисникам обходити традиційні системи захисту. Внутрішні інциденти, у свою чергу, пов'язані з діями співробітників або внутрішніх процесів, які можуть бути як навмисними, так і випадковими. Особливу небезпеку становлять інсайдери, які володіють знаннями про архітектуру системи, її вразливості та логіку роботи. Розуміння природи інцидентів інформаційної безпеки має ключове значення для побудови ефективної системи реагування. Класифікація дозволяє оцінити рівень ризику, визначити пріоритети реагування, встановити сценарії розвитку інциденту та підібрати відповідні методи нейтралізації. Саме класифікаційний підхід лежить в основі сучасних систем моніторингу, включно з платформами, побудованими на базі стеку ELK, які дозволяють аналізувати структуру, динаміку та складність інцидентів у режимі реального часу.

Однією з фундаментальних проблем у роботі з інцидентами інформаційної безпеки є відсутність чітко визначених граней між інцидентом, подією та загрозою. У традиційних підходах подія розглядається як факт, зафіксований у системі журналювання, що характеризує певну дію або стан інформаційного об'єкта. Такі події можуть мати суто технічний характер і не нести жодних негативних наслідків. Загроза визначається як потенційна можливість порушення властивостей інформації, тоді як інцидент становить собою реалізовану або таку, що почала реалізовуватися, загрозу. У практичному застосуванні ці три поняття нерідко переплітаються, оскільки одна й та сама подія може бути як нешкідливою, так і сигналом про початок цілеспрямованої атаки. У розподілених мережах події мають неоднорідний формат, різні часові позначки та різний ступінь важливості, що ускладнює їхнє об'єктивне трактування. Саме тому класифікація інцидентів часто ґрунтується не лише на їхній природі, а й на контексті виникнення, а також на ряді вторинних ознак, які дозволяють коректніше оцінити ступінь небезпеки [24].

Особливо важливим є те, що інциденти не лише класифікуються за природою впливу, а й поділяються за рівнем критичності. У корпоративних середовищах широко застосовуються моделі оцінювання серйозності інцидентів, які базуються

на аналізі наслідків, швидкості поширення, потенційного впливу на бізнес-процеси, рівня залучених активів та можливості ескалації атаки. Для інцидентів високого рівня критичності, наприклад у випадках компрометації доменних облікових записів, атак на системи керування інфраструктурою, або вторгнення до внутрішніх сегментів мережі, передбачаються окремі процедури реагування з негайною участю аналітиків високого рівня кваліфікації. Інциденти середнього рівня зазвичай пов'язані з підозрілою активністю користувачів, нетиповими операціями у базах даних, або змінами у конфігураціях мережеских пристроїв. Інциденти низького рівня можуть мати характер одиничних підозрілих дій, що не створюють прямої загрози, але підлягають подальшому аналізу.

У контексті побудови аналітичних систем особливого значення набуває класифікація інцидентів за типами векторів атаки. Вектором атаки вважається шлях, яким зловмисник реалізує свої наміри. У минулому класичні вектори включали експлуатацію вразливостей, перехоплення трафіку, соціальну інженерію та атаки на мережеві протоколи. Нині ж кількість можливих векторів значно збільшилася: атаки можуть використовувати слабкі налаштування хмарних сервісів, компрометовані API, незахищені контейнерні середовища, експлуатацію уразливостей у бібліотеках відкритого коду, маніпуляції у CI/CD конвеєрах, а також зловживання довірою у механізмах федеративної автентифікації. Усе це вимагає від аналітичних систем можливості комплексної оцінки поведінки користувачів, процесів, мережевого та системного трафіку.

У реальних умовах інциденти майже ніколи не виникають ізольовано. Зазвичай вони формують складні багатоступеневі ланцюги, які починаються зі збору інформації про систему, продовжуються спробами сканування портів або сервісів, після чого настає фаза експлуатації вразливості, закріплення у системі та горизонтального переміщення всередині мережі. Кожен із цих етапів супроводжується численними подіями, що можуть виглядати як нешкідливі або технічні, але разом створюють картину цілеспрямованої атаки. Саме тому у сучасних підходах до класифікації інцидентів дедалі більшу роль відіграє поведінковий аспект – аналіз шаблонів активності, що відхиляються від норми. Якщо раніше основною ознакою інциденту була наявність відомого сигнатурного

шаблону або фіксованої аномалії, то сьогодні у центрі уваги знаходиться поведінкове моделювання загроз, яке дозволяє виявляти складні інциденти незалежно від того, чи було їхнє сигнатурне представлення відоме заздалегідь. У цій площині особливо важливим стає питання розуміння інцидентів як процесів з динамічною структурою, які змінюються залежно від зовнішніх умов, дій зловмисника та реакції захисних систем. Інцидент не є фіксованою точкою у часі; він має часову протяжність та змінювану інтенсивність проявів. Наприклад, початкове проникнення може тривати лічені секунди, але збір інформації та підготовка до подальших дій можуть продовжуватися годинами або навіть днями. У багатьох випадках інцидент включає латентну фазу, коли активність зловмисника майже не фіксується. Саме такі події особливо складно виявити без використання глибокої аналітики, що працює у режимі постійного аналізу всього масиву даних.

Найбільш складні інциденти пов'язані з компрометацією облікових записів користувачів. Вони становлять одну з найсерйозніших загроз, оскільки зловмисник може використовувати легітимні облікові записи, що значно ускладнює виявлення. У таких випадках інцидент виявляється не на основі класичних критеріїв, а через аналіз нетипової поведінки користувача, відхилень від звичних робочих процесів, спроб доступу до раніше невідомих або заборонених ресурсів. Інциденти цього типу особливо небезпечні тим, що можуть призводити до несанкціонованого внесення змін у критично важливі параметри системи, зміни політик безпеки, відключення журналювання або модифікації прав доступу.

Одним із найбільш обговорюваних у науковій літературі аспектів класифікації інцидентів є питання їхнього вчасного виявлення. Коли інцидент за своєю суттю є процесом, що розвивається, момент його виявлення стає критичним чинником, який визначає масштаби наслідків. Багато інцидентів залишаються непоміченими протягом тривалого часу. У випадку складних атак класу АРТ середній час виявлення може становити кілька місяців, що призводить до катастрофічних наслідків. Саме тому класифікація інцидентів включає також їхній часовий вимір: інциденти швидкої дії та інциденти тривалої прихованої активності. Перші

завдають миттєвої шкоди, тоді як другі поступово руйнують систему, непомітно змінюючи конфігурації, крадучи дані або впливаючи на стратегічні ресурси.

У контексті корпоративних мереж надзвичайно важливим є аналіз інцидентів, що виникають унаслідок комбінованих атак. Сучасні зловмисники рідко використовують один метод проникнення. Найбільш ефективні атаки поєднують у собі інженерію соціальної довіри, експлуатацію технічних вразливостей, маніпуляції у сервісах хмарних провайдерів та використання вразливостей у мережевій інфраструктурі. Як наслідок, інциденти стають настільки складними, що їхня класифікація потребує застосування інтелектуальних систем, здатних визначати приховані взаємозв'язки між подіями, розбитими у часі та просторі.

Усе це не лише ускладнює класифікацію інцидентів, але й робить її невід'ємною частиною моделі реагування. Сучасна система кіберзахисту повинна вміти аналізувати інцидент не лише як окрему подію, а як частину загальної поведінкової картини мережі. Саме тому класифікація стає основою для автоматичного формування правил реагування, визначення сценаріїв ескалації, вибору методів стримування, а також оптимізації подальшого аналізу.

Суттєвою особливістю сучасних інцидентів інформаційної безпеки є їхня взаємозалежність від архітектури корпоративної мережі та способу організації інформаційних потоків. У міру того як компанії переходять до розподілених моделей з використанням хмарних сервісів, контейнеризованих середовищ, гібридних платформ та віддаленого доступу, інциденти стають набагато складнішими, а їхня класифікація – менш очевидною. Інцидент, який у традиційній інфраструктурі можна було б однозначно трактувати як атаку на внутрішній сервер, у розподіленому середовищі може мати одночасно кілька джерел – від неправильного налаштування IAM-політик у хмарі до компрометації API або токена доступу, що передається між різними сервісами. У таких умовах класифікація інцидентів повинна враховувати специфіку сучасних багаторівневих топологій, у яких дані пересуваються між численними компонентами не лише горизонтально, але й вертикально, між різними середовищами виконання. У багатьох випадках інциденти інформаційної безпеки слід розглядати як природний наслідок зростання складності технологічних систем. Сучасні компанії

використовують десятки або сотні програмних компонентів, кожен з яких має власні механізми журналювання, моніторингу та сигналізації. Тому інциденти часто виникають на стику різних систем і пов'язані не з окремими вразливостями, а з некоректною взаємодією компонентів. Наприклад, невідповідність форматів даних журналювання між серверами, системами зберігання та мережевими пристроями може призвести до того, що події, які вказують на інцидент, залишаться непоміченими через неможливість коректно поєднати їх у єдину логічну послідовність. Це підкреслює важливість стандартизації логів, узгодженості часових міток та уніфікації структур журналів, що особливо актуально в контексті побудови систем реагування на основі стеку ELK.

Значну складність у класифікації інцидентів становить і динамічний характер сучасних загроз. Якщо у 2000-х роках більшість атак була спрямована на експлуатацію конкретних програмних вразливостей і мала відносно передбачуваний сценарій, то сьогодні зловмисники використовують високорівневі моделі поведінки, орієнтовані на досягнення певної мети, а не на використання конкретної технічної слабину. Це означає, що інцидент не завжди можна визначити за ознаками низького рівня, такими як сигнатура шкідливого коду або зміна системного файлу. Усе частіше інциденти характеризуються складними поведінковими патернами, що складаються із серії дрібних відхилень від норми. Виявлення таких інцидентів потребує машинного аналізу великих масивів даних та застосування аналітичних платформ, здатних знаходити приховані закономірності та взаємозв'язки між подіями. Водночас розвиток методів атак зумовив виникнення принципово нових типів інцидентів, пов'язаних із хмарною інфраструктурою. У більшості хмарних середовищ інцидент часто не має фізичних проявів у традиційному сенсі – зловмисник не змінює конфігурації серверів або файлів, але може змінювати дозволи доступу, маніпулювати токенами, створювати тимчасові ролі, які важко виявити, або використовувати неправильно налаштовані сервіси для отримання доступу до конфіденційних даних. Класифікація інцидентів у хмарних середовищах повинна базуватися не на традиційних системних параметрах, а на аналізі операцій у логах API, які свідчать про зміну стану об'єктів системи без прямого впливу на файлову структуру або процеси.

У корпоративних мережах, побудованих на принципах Zero Trust, також виникають нові типи інцидентів, які вже не підпадають під класичні моделі. Оскільки рішення про доступ приймаються на основі множини параметрів, таких як місце розташування користувача, тип пристрою, рівень довіри до сеансу та оцінка ризику, інцидент може полягати не у фактичному порушенні системи, а у фіксації відхилення у поведінці цих параметрів. Наприклад, зміна геолокації користувача без відповідного логічного пояснення є потенційним показником компрометації, хоча сама по собі не свідчить про жодні шкідливі дії. Ще одним важливим аспектом класифікації є інциденти, що виникають у наслідок порушення життєвого циклу управління конфігураціями. Багато великих систем працюють у середовищах, де конфігурації змінюються автоматично відповідно до оновлень програмного забезпечення або змін у бізнес-процесах. Відсутність контролю над цими змінами може призвести до серії інцидентів, пов'язаних із ненавмисними змінами налаштувань, що відкривають вразливості. Найчастіше такі інциденти непомітні для адміністратора, оскільки виглядають як звичайні модифікації системи. Однак у разі поєднання з іншими подіями ці зміни можуть формувати критичний інцидент. Динамічна природа інцидентів проявляється також у тому, що багато з них відбуваються у хронологічних кластерах. Замість одного інциденту сучасні атаки запускають десятки пов'язаних подій, які формують складний часовий шаблон. У таких випадках традиційні методи аналізу не працюють через неспроможність поєднати події у цілісний ланцюг. Тому класифікація інцидентів має враховувати не просто подію як окремий факт, а її позицію в загальному часовому контексті мережі.

У практиці безпеки дедалі більше уваги приділяється інцидентам, що здійснюються через зловживання легітимними інструментами та можливостями системи. Багато сучасних атак ґрунтуються на використанні штатних механізмів операційних систем, а також інструментів адміністрування, що робить їх надзвичайно складними для виявлення. Це так звані «Living off the Land» атаки, у яких зловмисник не приносить у систему власне шкідливе ПЗ, а використовує вже наявні утиліти. Подібні атаки створюють інциденти нового типу, які практично не можуть бути ідентифіковані за сигнатурним підходом. У цьому випадку

класифікація ґрунтується виключно на поведінкових ознаках, на відхиленнях від нормальної роботи середовища, на аналізі логів командної активності. У контексті інцидентів особливої уваги потребують атаки на ланцюжок постачання (supply chain attacks). Ці інциденти є одними з найскладніших, оскільки вони вражають системи через легітимні компоненти, що надходять від сторонніх постачальників. У подібних випадках класифікація інцидентів майже неможлива без глибоких кореляцій між подіями, що відбуваються у різних частинах системи, та без аналітичних підходів, що дозволяють виявити нетипові взаємодії між сервісами.

Окремий виклик становлять інциденти, пов'язані з соціальною інженерією. Попри те, що такі інциденти часто починаються поза межами технічних систем, їхні наслідки проявляються в електронному вигляді: несанкціонований доступ до облікових записів, запуск шкідливого ПЗ, зміна мережевих політик або передача конфіденційної інформації. У таких випадках класифікація повинна враховувати як технічний, так і поведінковий аспект користувача, що робить процес набагато складнішим.

Розглядаючи інциденти інформаційної безпеки в контексті сучасних мережевих екосистем, необхідно підкреслити, що їхня класифікація має не лише теоретичне значення, а й безпосередньо впливає на ефективність роботи аналітичних команд та систем автоматизованого реагування. Класифікація визначає, як саме інцидент буде оброблятися, які ресурси будуть залучені, який механізм обмеження або ліквідації буде вибраний, і наскільки швидко система зможе повернутися до нормального режиму функціонування. У складних інформаційних середовищах помилка класифікації може призвести до того, що інцидент буде проігнорований, не отримає належного пріоритету або, навпаки, спричинить надмірне реагування, що виведе з ладу легітимні сервіси. Саме тому сучасні методики класифікації все більше орієнтуються на аналітичні моделі, здатні адаптуватися до динаміки загроз.

У системах великого масштабу, де щодня генеруються мільйони подій, лише мала частина з них має потенціал перетворитися на інциденти. Проте, щоб виявити цю частку, система повинна вміти аналізувати величезний масив неоднорідних даних, принципово відрізняючи звичайні технічні процеси від підозрілих змін у

поведінці користувачів або процесів. Класичні підходи до класифікації, які ґрунтуються на сигнатурах відомих атак, давно показали свою обмеженість. Нові форми атак, що постійно еволюціонують, часто не мають сигнатур у момент реалізації. Тому в класифікаційні моделі необхідно інтегрувати аналітику, орієнтовану на контекст: аналіз користувацької поведінки, історії взаємодій між сервісами, характеристик мережного трафіку, часових закономірностей, відхилень від типових патернів.

Важливим аспектом класифікації інцидентів є оцінка їхнього впливу на бізнес-процеси. У корпоративному середовищі більш критичним інцидентом може бути не той, що має найбільше технічне значення, а той, який вражає найбільш важливі елементи інфраструктури. Наприклад, несанкціонований доступ до тестового сервера не матиме таких наслідків, як компрометація облікового запису адміністратора домену. Саме тому класифікація інцидентів повинна враховувати критичність активів, до яких стосується інцидент, їх роль у забезпеченні безперервності роботи компанії, зв'язки між цими активами та іншими системами, а також можливість каскадного поширення впливу.

Сучасні методики управління інцидентами дедалі частіше включають оцінку їхньої латентності. Латентність інциденту – це проміжок часу між моментом виникнення шкідливої активності та моментом її виявлення захисною системою. Чим більш латентний інцидент, тим складніше його класифікувати, оскільки інформаційні сліди можуть бути фрагментованими або частково втраченими. У складних мережах цей проміжок часу може містити сотні тисяч логів, у яких справжній інцидент приховується серед численних вторинних подій. У таких умовах точність класифікації залежить від здатності системи аналізувати дані в ретроспективі, відновлюючи повну картину інциденту з розрізнених фрагментів.

Важливою тенденцією є поява інцидентів, які мають розподілений характер. У розподілених атаках кожен окремий елемент може виглядати абсолютно безпечним. Наприклад, одночасні запити з різних регіонів або збільшення обсягу трафіку до певного сервісу можуть бути частиною повсякденної роботи мережі. Однак у поєднанні такі події утворюють складний інцидент, який розгортається у кількох точках та потребує глибокої кореляції. Класифікація таких інцидентів

передбачає поєднання множини джерел інформації – мережевих логів, системних журналів, подій автентифікації, даних з хмарних сервісів, журналів API – у єдину структуру. У цьому контексті зростає роль кореляційних механізмів, здатних масштабуватися та обробляти великі обсяги інформації у реальному часі. Класифікація інцидентів за поведінковими ознаками передбачає моделювання базових шаблонів поведінки користувачів та сервісів, з можливістю автоматично визначати відхилення. Такі методи стали особливо актуальними у зв'язку з переходом на віддалений режим роботи, коли активність користувачів змінилася і перестала бути такою передбачуваною. Це означає, що класична класифікація більше не може базуватися на статичних правилах, натомість вона повинна визначати інциденти за допомогою статистичних моделей і машинного навчання.

Значну частину інцидентів становлять події, спричинені людським фактором. Помилки користувачів, неправильне поводження з доступами, ненавмисне надання привілеїв, використання ненадійних паролів або відкриття шкідливих вкладень створюють численні інциденти, які важко класифікувати за технічними ознаками. Це інциденти, що формуються поступово, через накопичення кількох незначних порушень. Вони відрізняються від технічних атак тим, що дії користувача є легітимними з точки зору системи, але породжують небезпечні сценарії. У таких випадках класифікація повинна включати поведінковий аналіз, моделювання типових робочих дій, відстеження змін у рівнях привілеїв, аналіз географічних та часових закономірностей.

Особливої уваги потребують інциденти, спричинені неправильним проектуванням або обслуговуванням систем безпеки. Неправильні налаштування міжмережевих екранів, пропущені оновлення, невалідні сертифікати, відсутність шифрування або ненавмисно відкриті сервіси створюють умови для численних вторинних інцидентів. Помилка в одному компоненті може призвести до ланцюгової реакції, через яку зловмисник отримає можливість розширеного доступу. Класифікація таких інцидентів потребує розуміння архітектурних взаємозв'язків між усіма елементами системи, включно з мережевими пристроями, серверами, додатками та системами управління інфраструктурою. Однією з найбільш складних категорій є інциденти, пов'язані з даними. У сучасних умовах

дані є найціннішим активом організації, і будь-які дії, що призводять до їхнього пошкодження, втрати, модифікації або несанкціонованого доступу, створюють високорівневі інциденти. У таких випадках класифікація повинна враховувати характер даних, їхню чутливість, рівень доступу, регуляторні вимоги, а також потенційний вплив на бізнес-процеси. Інциденти, пов'язані з витоком персональних даних, можуть мати зовсім іншу вагу порівняно з інцидентами, що стосуються системного журналювання або допоміжних сервісів.

Класифікація інцидентів інформаційної безпеки у сучасних корпоративних мережах є багатошаровим процесом, що включає оцінку природи інциденту, його впливу, динаміки, джерела походження, прив'язки до бізнес-процесів, часових закономірностей та поведінкових характеристик. Її складність і важливість зростають прямо пропорційно збільшенню складності систем, обсягів даних і різноманітності атак. Саме тому класифікація інцидентів стала основою для побудови глибоких аналітичних систем, включно з платформами на базі стеку ELK, які дозволяють масштабовано аналізувати складні взаємозв'язки та виявляти інциденти, що раніше були непомітними для традиційних підходів.

У підсумку можна стверджувати, що інциденти інформаційної безпеки являють собою складні, багатогранні та багатокомпонентні процеси, які вимагають глибокого теоретичного розуміння і точного практичного підходу до їх класифікації. Сучасні інциденти не можуть бути описані однією моделлю чи визначені за одним критерієм. Вони включають технічні, поведінкові, організаційні та контекстуальні аспекти, що вимагає використання багаторівневих класифікаційних схем. У цьому контексті побудова ефективної моделі реагування неможлива без ґрунтовного розуміння природи інцидентів та закономірностей їх виникнення.

1.2. Етапи процесу реагування на інциденти згідно міжнародних стандартів (NIST, ISO/IEC 27035)

Процес реагування на інциденти інформаційної безпеки є одним із найважливіших елементів управління ризиками у сучасних корпоративних мережах. Стандартизовані підходи до реагування на інциденти були сформовані як відповідь на зростаючу кількість складних загроз, що здатні вражати критично

важливі інформаційні ресурси, порушувати роботу бізнес-процесів та ставити під загрозу захищеність конфіденційних даних. Міжнародні стандарти, зокрема NIST SP 800-61 та ISO/IEC 27035, розглядають реагування на інциденти як циклічний багатофазний процес, що охоплює весь життєвий цикл інциденту – від початкового виявлення до післяінцидентного аналізу [1,3]. Ці документи формують основу для розроблення політик, процедур, технологічних рішень і організаційних структур, спрямованих на зниження впливу інцидентів та підвищення стійкості інформаційних систем.

Підхід, запропонований стандартом NIST SP 800-61, виділяє кілька основних етапів, кожен із яких відіграє важливу роль у забезпеченні оперативного та ефективного реагування [3,19]. Передусім йдеться про етап підготовки, який включає створення відповідної інфраструктури для роботи з інцидентами, формування команди реагування, визначення пріоритетів, розроблення політик та методичних рекомендацій. Підготовка є фундаментальною частиною життєвого циклу інцидентів, оскільки саме вона визначає, наскільки швидко та ефективно організація зможе відреагувати на небажані події. На практиці недостатня підготовленість часто є причиною того, що інциденти залишаються непоміченими або неправильно класифікуються, що збільшує їхній потенційний вплив та ускладнює подальші етапи реагування.

Етап виявлення та аналізу інцидентів у стандартах розглядається як критично важливий та найбільш інтенсивний з точки зору аналітичної роботи. Він передбачає фіксацію подій безпеки, їхнє коректне трактування, підтвердження факту інциденту та оцінку його масштабу й потенційних наслідків. У рамках цього етапу здійснюється збір логів, їхня нормалізація, кореляція, застосування правил та моделей для виявлення аномалій. Чим більше джерел даних використовує організація, тим складнішим стає процес виявлення. Стандарти наголошують, що на цьому етапі не менш важливим є уникнення як хибнопозитивних, так і хибнонегативних результатів, оскільки помилка в аналізі може призвести або до надмірного навантаження на команду реагування, або до пропуску небезпечних подій. Після того як інцидент був підтверджений та класифікований, починається етап локалізації. Його основна мета полягає у запобіганні подальшому поширенню

загрози та обмеженні потенційної шкоди. Локалізація може включати ізоляцію уражених хостів, закриття доступу до певних сегментів мережі, блокування операцій користувачів, деактивацію скомпрометованих облікових записів або тимчасове вимкнення сервісів. У стандартах наголошується на необхідності балансування між швидкістю локалізації та мінімізацією побічних ефектів, оскільки надто різкі дії можуть призвести до небажаних переривань у роботі сервісів або навіть до часткового паралічу бізнес-процесів.

Після успішної локалізації організація повинна перейти до етапу усунення наслідків, що включає очищення системи від шкідливих компонентів, видалення несанкціонованих змін, виправлення вразливостей та відновлення конфігурацій до безпечного стану. Усунення є найбільш трудомістким етапом, оскільки часто потребує глибокого розуміння архітектури системи, знання можливих варіантів шкідливої активності та ретельного аналізу журналів. Стандарти визначають необхідність документування кожного кроку, що дозволяє у подальшому аналізувати ефективність процедури та уникати повторних інцидентів у майбутньому.

Етап відновлення передбачає повернення системи до нормального режиму роботи. Його складність полягає не лише у технічному відновленні, а й у встановленні контролю за системою після інциденту, оскільки важливо переконатися, що загроза була повністю нейтралізована і не залишила прихованих механізмів повторного проникнення. На цьому етапі важливими є моніторинг, постінцидентний аудит, перевірка коректності роботи сервісів, а також оцінка довгострокових наслідків інциденту для бізнесу.

Завершальним етапом життєвого циклу інциденту є післяінцидентна діяльність. Вона включає проведення детального аналізу причин та хронології інциденту, оцінку ефективності реагування, формування рекомендацій, внесення змін до політик та процедур, а також навчання персоналу на основі досвіду. NIST та ISO підкреслюють важливість цього етапу як механізму постійного вдосконалення системи реагування, що дозволяє адаптувати організацію до нових типів загроз та вдосконалювати практики кіберзахисту.

Описуючи структуру процесу реагування на інциденти відповідно до міжнародних стандартів, необхідно підкреслити, що NIST SP 800-61 та ISO/IEC 27035 хоч і мають подібну загальну логіку, однак містять низку концептуальних відмінностей, які впливають на організацію реагування в конкретній інфраструктурі. NIST більше зосереджується на технічній складовій процесу, визначаючи чіткі етапи, методи та рекомендації для технічних фахівців, тоді як ISO/IEC 27035 приділяє більше уваги організаційним аспектам, побудові системи управління інцидентами, взаємодії команд, документації та підтримці корпоративної політики безпеки [1,2]. Втім, обидва підходи узгоджуються в тому, що реагування на інцидент є безперервним циклом, який повинен удосконалюватися шляхом повторного аналізу та постійного оновлення процедур.

На практиці етап підготовки у межах міжнародних стандартів вважається вирішальним, оскільки саме він створює базу для ефективного функціонування всієї системи реагування. Стандарти вимагають від організацій забезпечити наявність кваліфікованого персоналу, визначених відповідальностей, а також інструментів, здатних забезпечити моніторинг, журналювання, аналіз і кореляцію подій у реальному часі. Підготовка включає створення інцидент-менеджмент планів, процедур ескалації, моделей комунікаційного реагування, механізмів взаємодії з зовнішніми структурами (CERT, CSIRT, правоохоронними органами), а також проведення регулярних тренувань і симуляцій інцидентів. У межах підготовки також особливе значення має визначення переліку активів, що потребують захисту, оцінка їх критичності, а також побудова відповідної моделі ризиків, що дозволяє встановити пріоритети у реагуванні. Етап виявлення та аналізу інцидентів у відповідності до NIST передбачає не просто фіксацію подій, що відхиляються від норми, а й оцінку їхньої значимості та прийняття рішення щодо подальших дій. У межах цього етапу інформація про події безпеки повинна бути зібрана з якомога більшої кількості джерел, що включає сервери, маршрутизатори, системи автентифікації, мережеві засоби контролю доступу, хмарні сервіси, системи віртуалізації, кінцеві точки та користувацькі пристрої. Журнали повинні містити достатню кількість структурованих даних, аби система могла автоматично аналізувати їх і визначати потенційну небезпеку. Це робить

системи типу ELK особливо важливими у межах цього етапу, оскільки вони дозволяють об'єднувати інформацію з різномірних джерел у централізоване сховище з подальшим аналізом і можливістю швидкої кореляції подій.

У міжнародних стандартах окрему увагу приділено тому, що процес виявлення має враховувати різні типи інцидентів – від мережових вторгнень до підозрілих змін у поведінці користувачів. У цьому контексті NIST підкреслює важливість багаторівневих методів виявлення: сигнатурного аналізу, евристичного аналізу, поведінкових моделей та кореляції великих масивів даних. Стандарти рекомендують організаціям встановлювати чіткі критерії щодо того, яка подія повинна вважатися інцидентом, які події можуть бути проігноровані як звичайний робочий шум, а які вимагають негайної уваги. Це забезпечує ефективне використання ресурсів та дозволяє уникнути перевантаження команди реагування вторинними подіями. Після фази ідентифікації інцидент переходить у стадію стримування. Стандарти взаємно наголошують, що стримування має бути організоване у два підетапи: короткострокове та довгострокове. Короткострокове спрямоване на негайну ізоляцію загрози, тоді як довгострокове передбачає застосування стійких архітектурних змін, що унеможливають повторне проникнення. Наприклад, у разі виявлення зараження кінцевої точки короткострокові дії можуть включати її ізоляцію від мережі, тоді як довгострокові – створення додаткових правил сегментації, оновлення антивірусних політик або зміни політик доступу. У складних мережах стримування може вимагати багаторівневої координації між різними командами, що у свою чергу вимагає чіткої комунікації та визначеної структури відповідальності.

Етап ліквідації загрози в стандартах розглядається як такий, що повинен забезпечити повне видалення причин інциденту. Ліквідація включає аналіз маршруту проникнення, знищення шкідливих файлів, видалення несанкціонованих процесів, відновлення налаштувань конфігурацій, оновлення програмного забезпечення та проведення перевірки на предмет залишення зловмисником механізмів постійної присутності. NIST підкреслює необхідність документувати усі зміни, щоб у майбутньому команда могла відстежити які дії були виконані і як вони вплинули на загальну стійкість системи. ISO/IEC 27035, у свою чергу,

наголошує на необхідності інтегрувати цей етап у загальний процес управління змінами в організації, щоб уникнути створення нових уразливостей. Завершення ліквідації відкриває шлях до етапу відновлення, що включає перевірку всієї інфраструктури на стабільність, повернення систем до нормального операційного стану, повторне тестування сервісів, а також корекцію процесів моніторингу з урахуванням уроків, здобутих у результаті інциденту. Стандарти передбачають, що відновлення має здійснюватися з тією ж ретельністю, що і попередні етапи, оскільки саме на цьому етапі організація може випадково відновити в систему ті компоненти, які були скомпрометовані або могли залишитися у прихованому вигляді.

Останньою фазою є післяінцидентний аналіз, що виступає ключовим механізмом самовдосконалення. Міжнародні стандарти вимагають зберігати інформацію про інцидент, проводити комплексний розгляд причин, що призвели до його виникнення, і визначати, які дії могли запобігти або мінімізувати шкоду. У цьому етапі беруть участь не лише команди реагування, а й представники управління, що формують подальші стратегії безпеки. Післяінцидентна фаза є тією точкою, де відбувається адаптація системи до нових реалій загрозового середовища.

Одним із ключових принципів реагування відповідно до ISO/IEC 27035 є необхідність формування організаційної структури управління інцидентами, яка включає визначення чітких ролей, відповідальностей та каналів комунікації. Цей стандарт наголошує, що навіть найсучасніші технічні засоби не забезпечать належного рівня реагування, якщо організація не матиме структурованої системи взаємодії між підрозділами. Інцидент є завжди міждисциплінарною проблемою, яка зачіпає технічні, організаційні, юридичні та управлінські аспекти. Тому для ефективного реагування необхідно створити механізм координації між підрозділами, що забезпечить узгодженість дій та мінімізує час прийняття рішень.

З цієї точки зору стандарт ISO/IEC 27035 приділяє значну увагу попередній підготовці до інцидентів, яка включає аналіз можливих сценаріїв атак, моделювання загроз, проведення навчальних тренувань та розроблення процедур реагування на різні типи інцидентів. Така підготовка дозволяє організації

працювати за принципом проактивності, передбачаючи потенційні загрози ще до того, як вони стануть реальністю. На практиці це означає, що організація повинна не лише мати актуальну документацію та політики, а й регулярно проводити аудити, пентести, перевірки конфігурацій, тестування резервних копій, а також забезпечувати навчання співробітників.

У контексті реагування на інциденти виявлення є, без перебільшення, центральним етапом, оскільки саме він визначає, чи буде інцидент зафіксований і наскільки швидко буде здійснена відповідь. Стандарти підкреслюють важливість якісного журналювання та моніторингу, оскільки без наявності структурованих даних неможливо визначити навіть сам факт інциденту. З огляду на величезний обсяг подій, які щодня генеруються сучасними ІТ-середовищами, виявлення інцидентів стає проблемою, що виходить за межі можливостей людини. Саме тому міжнародні стандарти рекомендують автоматизувати процеси виявлення, використовуючи централізовані платформи для збору логів, такі як ELK, і поєднувати їх із механізмами кореляції та поведінкового аналізу. Коли інцидент ідентифіковано, наступним ключовим завданням є оцінка його серйозності. На цьому етапі важливо зрозуміти не лише технічні характеристики інциденту, але й його потенційний вплив на критично важливі активи. У багатьох випадках інциденти, які здаються незначними на перший погляд, можуть впливати на стратегічні бізнес-процеси, а отже, вимагати негайної реакції. Наприклад, навіть одинична підозріла операція в контексті Active Directory може бути частиною багатоступеневої атаки, метою якої є перехоплення контролю над доменом. Тому стандарти наголошують, що оцінка інциденту має включати аналіз його походження, цілей зловмисника, потенційного маршруту поширення та можливих наслідків.

Стандарти також встановлюють концептуальний зв'язок між етапами виявлення та локалізації. З одного боку, швидка локалізація дозволяє обмежити масштаби інциденту, але з іншого – надто поспішні дії без достатнього аналізу можуть призвести до втрати важливої доказової інформації або навіть спровокувати ескалацію атаки. Тому міжнародні рекомендації закликають застосовувати збалансований підхід, який поєднує оперативні дії та аналітичну

обережність. У деяких випадках доцільно спочатку провести детальний аналіз інциденту, щоб зрозуміти його природу, а вже потім переходити до локалізації. В інших – затримка у реагуванні може призвести до критичних наслідків.

Після успішної локалізації ідентифікованого інциденту, відповідно до NIST та ISO, має розпочатися фаза ліквідації, де ключовим завданням є усунення кореневої причини та видалення всіх компонентів загрози. Цей етап включає не лише технічні дії, такі як видалення шкідливих файлів, відновлення конфігурацій, блокування шкідливого доступу, а й організаційні заходи, які повинні гарантувати, що аналогічні інциденти не повторяться. У цьому контексті важливим є аналіз вразливостей, що призвели до інциденту, а також їх негайне виправлення. Ліквідація має проводитися максимально ретельно, оскільки навіть невелика кількість уцілілих компонентів загрози може згодом призвести до повторного компрометування. Етап відновлення у міжнародних стандартах розглядається як такий, що завершує основну активну фазу реагування. Він включає перевірку стабільності системи після усунення інциденту, тестування доступності сервісів, перевірку коректності логів, повторний аналіз конфігурацій, здійснення незалежного аудиту та поступове повернення системи до нормального робочого режиму. Важливо підкреслити, що відновлення не зводиться до технічного відновлення працездатності, оскільки має переконатися у відсутності залишкових ризиків та прихованих механізмів присутності зловмисника.

Завершальним етапом, що об'єднує всі попередні фази, є післяінцидентна діяльність. ISO/IEC 27035 детально регламентує порядок документування, аналізу, проведення звітності, збереження доказів, оновлення політик та навчання персоналу. Післяінцидентний аналіз повинен враховувати всі аспекти інциденту: часову шкалу, вразливості, які були використані, маршрути проникнення, реакції систем, дії фахівців, ефективність інструментів. Саме на цьому етапі формуються уроки та покращення, які дозволяють організації посилити свій захист і запобігти повторенню аналогічних загроз.

Аналізуючи рекомендації міжнародних стандартів щодо реагування на інциденти, важливо усвідомлювати, що жоден з етапів цього процесу не може розглядатися ізольовано. Усі вони утворюють єдиний наскрізний механізм захисту,

який функціонує лише за умови цілісності, послідовності та узгодженості дій. Стандарти наголошують, що ефективність реагування залежить не від окремих технічних рішень, а від того, наскільки гармонійно всі елементи процесу взаємодіють між собою. Це означає, що інцидент-менеджмент повинен інтегрувати процеси моніторингу, аналітики, логування, управління доступом, сегментації мережі, аудиту, резервного копіювання та управління конфігураціями у єдину логічну структуру. Кожна зміна в одному елементі цієї структури неминуче впливає на інші, і тому стандарти вимагають системного підходу до реагування.

Усі міжнародні моделі реагування підкреслюють важливість принципу «відповідності часу». Це означає, що організація повинна реагувати на інциденти з урахуванням того, що загрозливе середовище змінюється у реальному часі і не може чекати на тривалі ручні процедури. Час, необхідний для виявлення і реагування, у світовій практиці вимірюється не днями й навіть не годинами, а хвилинами або навіть секундами. Саме цим обумовлена необхідність автоматизації, яку NIST та ISO вважають ключовою умовою сучасної кібербезпеки. Автоматизація дозволяє не лише швидше виконувати рутинні операції, а й забезпечує точність, узгодженість і контроль якості дій. Це, у свою чергу, мінімізує ризики людських помилок, що є однією з найпоширеніших причин інцидентів або невдалого реагування.

У рекомендаціях NIST підкреслюється обов'язковість створення чіткої та формально визначеної структури інцидент-менеджменту. Це включає формування спеціалізованої команди реагування – CSIRT, із зазначенням її складу, ролей, відповідальностей та регламентів взаємодії. У разі великої інфраструктури рекомендується створювати кілька рівнів реагування: першої лінії, що забезпечує базовий моніторинг і первинну класифікацію; другої лінії, яка здійснює глибокий технічний аналіз; і третьої лінії, що відповідає за стратегічні рішення, взаємодію з зовнішніми структурами та визначення довгострокових заходів. Наявність таких рівнів дозволяє розподілити навантаження та забезпечити оперативність реакції навіть у випадку великої кількості одночасних інцидентів. Стандарти наголошують, що ефективність реагування значною мірою залежить від якості журналювання та збереження доказів. Сучасні атаки часто відбуваються у кілька

фаз, які можуть бути розділені у часі на години або дні. Через це важливо забезпечити збереження всіх подій у хронологічній послідовності, з точними часовими мітками, які дають змогу відтворити весь хід інциденту під час форензичного аналізу. ISO/IEC 27035 спеціально підкреслює, що будь-яке видалення або зміна журналів створює ризик втрати доказів, що може не лише ускладнити розслідування, а й призвести до юридичних наслідків. Тому збереження логів, забезпечення їхньої цілісності та контроль доступу до них є базовим елементом інцидент-менеджменту.

Особливу увагу міжнародні стандарти приділяють вимогам до коректної документалізації всіх дій, виконаних у процесі реагування. Документація повинна відображати інформацію про часові мітки подій, рішення, які були прийняті, дії, що виконувалися, обґрунтування цих дій, а також результати кожного кроку. Це необхідно не лише для внутрішнього аналізу, а й для аудиту, взаємодії з регуляторами або правоохоронними органами. Документація дозволяє організації провести повноцінний післяінцидентний аналіз та визначити, які елементи процесу потребують удосконалення. У великих компаніях документація відіграє також роль інструмента відтворення знань: вона дозволяє формувати базу досвіду, яку можна використовувати при подальших інцидентах.

Стандарти NIST і ISO також звертають увагу на необхідність використання інструментів кореляції подій, здатних аналізувати складні інциденти, що розгортаються у кількох сегментах мережі. Ці інструменти повинні виявляти не окремі події, а взаємозв'язки між ними: часові відповідності, спільні атрибути, логічні залежності, повторювані поведінкові патерни. У складних атаках інтеграція подій у єдину модель дозволяє зрозуміти справжній задум зловмисника. Прикладом подібного підходу є APT-атаки, які складаються з десятків дрібних подій, кожна з яких не виглядає небезпечною сама по собі, але разом вони утворюють ланцюг проникнення в систему. Саме тут вступає в силу важливість використання централізованих платформ на кшталт ELK, які дозволяють зберігати, обробляти та аналізувати величезні масиви журналів у реальному часі.

Окремо слід відзначити, що сучасні стандарти реагування на інциденти все більше орієнтуються на інтеграцію з моделями загроз, наприклад MITRE ATT&CK.

Завдяки цим моделям інциденти можна класифікувати за техніками, тактиками та процедурами, які використовуються зловмисниками. Це дозволяє не лише аналізувати інциденти у ретроспективі, а й формувати прогностичні моделі, що дають змогу передбачати потенційні напрямки ескалації. У деяких випадках системи можуть навіть автоматично визначати, до якої фази життя атаки належить інцидент, що значно підвищує якість реагування. Ще однією особливістю, на яку звертають увагу міжнародні стандарти, є важливість комунікації під час інциденту. Комунікація повинна бути швидкою, чіткою, стандартизованою та конфіденційною. Вона має враховувати як внутрішніх учасників процесу реагування, так і зовнішніх стейкхолдерів: клієнтів, партнерів, постачальників, журналістів, регуляторів, органи правопорядку. NIST рекомендує використовувати заздалегідь підготовлені шаблони комунікацій, що дозволяє зменшити ризики неправильного трактування подій або розголошення зайвої інформації. Під час кризових інцидентів комунікація може відігравати роль стратегічного інструмента, який визначає, як організація буде сприйматися зовнішнім світом.

Не менш важливим аспектом є необхідність проведення оцінки ефективності системи реагування. Міжнародні стандарти вимагають проводити регулярні перевірки, тестування, симуляції атак, аудит процедур і аналіз виконаних інцидентів. Це дозволяє визначати, які елементи системи працюють ефективно, а які потребують вдосконалення. У цьому контексті важливо наголосити, що процес реагування не може залишатися статичним. Він повинен змінюватися разом зі змінами у загрозовому середовищі, технологіях, нормативних вимогах та внутрішній інфраструктурі організації.

Розглядаючи загрози через призму міжнародних стандартів, стає очевидним, що сучасні інциденти стають дедалі складнішими не лише з технічної точки зору, а й з організаційної. Багато інцидентів пов'язані з комплексними багатофазними атаками, які можуть розгортатися упродовж тривалого часу, торкаючись різних сегментів мережі та використовуючи численні точки входу. Тому системи реагування повинні не лише швидко виявляти інцидент, але й аналізувати його у широкому контексті, визначати глибинні взаємозв'язки, розуміти логіку атакуючого та прогнозувати потенційні сценарії розвитку інциденту. NIST

рекомендує використовувати у процесі реагування так званий threat intelligence розвідувальну інформацію про загрози, яка дозволяє моделювати поведінку зловмисників та передбачати подальші кроки атаки.

У сучасних умовах цифрової трансформації велике значення набуває здатність систем реагування інтегруватися з технологічними платформами автоматизації, такими як SOAR-рішення, оркестратори інцидентів та системи поведінкового аналізу користувачів. Міжнародні стандарти не вказують конкретних технологій, але визначають, що автоматизація є необхідною передумовою ефективності реагування. Автоматизація дозволяє обробляти великі обсяги подій, виконувати рутинні операції без участі людини, формувати структуровані звіти, забезпечувати своєчасне інформування відповідальних осіб, а також виконувати швидкі технічні дії, які у ручному режимі були б надто тривалими.

Окремим важливим аспектом реагування, що підкреслюється у міжнародних стандартах, є забезпечення безперервності бізнесу. Інцидент дуже часто впливає на критичні процеси, зупиняючи роботу систем, порушуючи логістичні ланцюги або створюючи ризики витоку конфіденційної інформації. Тому реагування повинно враховувати не лише технічний бік, а й бізнес-контекст. У великих організаціях інцидент-менеджмент повинен бути інтегрованим з планами резервного відновлення, управління ризиками та управління якістю. Це означає, що інцидент розглядається як подія, яка може мати стратегічні наслідки, а тому вимоги до реагування набувають міждисциплінарного характеру.

Отже, міжнародні стандарти формують системну і методологічно виважену модель реагування на інциденти, яка дозволяє організаціям забезпечити високий рівень стійкості своїх ІТ-систем до кіберзагроз. Вони не просто регламентують технічні та організаційні деталі, але й формують філософію реагування, що базується на принципах проактивності, адаптивності, комплексності та безперервного вдосконалення. Завдяки цьому стандарти NIST та ISO/IEC 27035 стали основою для побудови сучасних систем безпеки, включно з тими, що інтегруються з платформами на базі стеку ELK, які у наступних розділах досліджуватимуться як ключовий інструмент для автоматизованого виявлення, аналізу та реагування на інциденти.

1.3. Аналіз принципів побудови систем моніторингу та реагування на інциденти

Системи моніторингу та реагування на інциденти інформаційної безпеки є невід’ємним компонентом сучасної інфраструктури кіберзахисту. Їхнє завдання полягає у безперервному спостереженні за подіями, виявленні аномалій, оперативній інтерпретації загроз та координації дій щодо реагування. Формування таких систем ґрунтується на комплексі принципів, які визначають їхню побудову, архітектуру, функціонування, а також взаємодію між технічними й організаційними компонентами [6,22]. В умовах динамічного загрозового середовища, коли атаки стають дедалі складнішими, розподіленішими та прихованішими, принципи побудови систем моніторингу набувають критичного значення, оскільки саме вони визначають здатність організації вчасно реагувати на небезпеки та мінімізувати їхній вплив.

Одним із ключових принципів формування систем моніторингу є принцип безперервності. Сучасні системи працюють у режимі реального часу або близькому до нього, аналізуючи величезні обсяги даних, що надходять від різноманітних джерел – серверів, кінцевих точок, мережевих пристроїв, систем автентифікації, хмарних платформ, контейнерних середовищ та бізнес-додатків. Безперервність необхідна тому, що інциденти можуть виникати будь-якої миті та розгортатися дуже швидко. Навіть короткочасна втрата видимості може призвести до пропуску критичних подій, дозволяючи зловмисникам отримати доступ до корпоративних активів. Тому системи повинні забезпечувати не лише високу доступність, а й відмовостійкість, мати резервні канали збору даних та можливості для швидкого відновлення після збоїв. Ще одним принципом є централізація збору даних. У сучасних інфраструктурах події генеруються тисячами різнорідних компонентів, і лише централізоване їх накопичення дозволяє сформувати цілісну картину стану системи. Централізація забезпечує синхронізацію часових міток, уніфікацію форматів журналів, спрощення кореляції та створення єдиного інформаційного простору для аналітиків [7,8]. Цей принцип є основою для побудови сховищ логів, таких як Elasticsearch, які дозволяють індексувати події та здійснювати пошук у

великих масивах даних із високою швидкістю. Завдяки централізації можна пов'язувати події з різних сегментів мережі, визначати закономірності поведінки, відстежувати маршрути атак та формувати вичерпний контекст для подальшого реагування.

Принцип кореляції подій є фундаментальним для роботи будь-якої системи моніторингу. Окремо розглянута подія нерідко не несе загрози, але у поєднанні з іншими може формувати складний інцидент. Наприклад, одинична невдала спроба автентифікації не є небезпечною, але її множинне повторення, поєднане з нетиповим географічним походженням та збільшенням інтенсивності трафіку до певного сервера, може свідчити про brute-force атаку або спробу підбору доступів. Кореляція дозволяє виявляти приховані взаємозв'язки між подіями, визначати логіку дій атакувальника та формувати модель інциденту як цілісного процесу. Завдяки цьому система може не лише реагувати на окремі події, а й виявляти складні багатоступеневі атаки, що розгортаються протягом тривалого часу.

Принцип контекстуалізації є важливим доповненням до кореляції. Його суть полягає у тому, що подія повинна аналізуватися не лише з технічної точки зору, а й у широкому контексті конкретної інфраструктури. Це означає, що система повинна враховувати бізнес-значимість активу, історію його роботи, роль у мережевій топології, типові патерни активності користувачів, залежності між сервісами та інші фактори. Наприклад, підозріла операція на сервері з тестовими даними може бути менш критичною, ніж така сама операція на сервері, що містить конфіденційну інформацію. Контекст дозволяє коректно визначити пріоритет реагування та оцінити потенційний вплив інциденту.

У системах моніторингу важливу роль відіграє також принцип поведінкової аналітики. Сучасні загрози нерідко ґрунтуються не на технічних слабких місцях, а на відхиленнях у поведінці користувачів, сервісів або процесів. Традиційні сигнатурні підходи виявлення, що базуються на визначених шаблонах атак, виявляються недостатніми у випадках нових або модифікованих загроз. Поведінкові моделі аналізують активність об'єктів у динаміці, порівнюють її з історичною нормою та визначають відхилення, які можуть свідчити про інцидент [26,27]. Цей принцип

лежить в основі аналітики UEBA, яка дедалі частіше інтегрується у системи моніторингу.

Ще одним принципом побудови систем моніторингу є принцип прозорості та трасованості. Він означає, що система повинна зберігати повну історію подій, зміни конфігурацій, журнали дій користувачів та інші артефакти, які дозволяють відтворити хронологію інциденту. Трасованість необхідна для цифрової експертизи, для юридичного супроводу, для перевірки ефективності реагування та для внутрішнього аудиту. Вона забезпечує підтвердження достовірності даних, дозволяючи ідентифікувати будь-які маніпуляції з журналами або слідами атак.

Одним із важливих принципів побудови систем моніторингу є адаптивність, яка передбачає здатність системи реагувати на зміни в загрозовому середовищі. Технологічний прогрес призводить до постійної появи нових методів атак, а розробники шкідливого програмного забезпечення постійно вдосконалюють свої техніки для уникнення виявлення. У таких умовах статичні механізми моніторингу швидко втрачають актуальність. Адаптивність означає, що система повинна мати здатність оновлювати правила, моделі поведінки, сигнатури, алгоритми кореляції та методи аналізу без необхідності проводити фундаментальні зміни в її архітектурі. У практичному застосуванні це означає інтеграцію механізмів машинного навчання, залучення зовнішніх джерел інформації про загрози, автоматичне створення поведінкових моделей та динамічне переналаштування засобів моніторингу залежно від інфраструктурних змін.

Принцип масштабованості є не менш важливим, оскільки сучасні інформаційні системи генерують величезні обсяги даних, які необхідно обробляти ефективно та своєчасно. Системи моніторингу повинні підтримувати горизонтальне й вертикальне масштабування, забезпечуючи достатній рівень продуктивності навіть при різкому збільшенні кількості подій. Це особливо актуально для організацій, що працюють у хмарних або гібридних середовищах, де кількість компонентів може змінюватися динамічно. Масштабованість дозволяє обробляти мільйони подій за хвилину, зберігати їх у структурованому вигляді, індексувати та забезпечувати швидкий доступ аналітикам навіть під час пікових навантажень. Архітектурно масштабованість реалізується через використання

мікросервісних структур, розподілених баз даних, кластеризацію Elasticsearch, контейнеризацію та оркестрацію.

Особливе місце в аналізі принципів побудови систем моніторингу займає принцип сегментації та поділу довіри. Він полягає в тому, що система не повинна розглядати мережу як однорідне середовище, в якому усі компоненти однаково довірені. Навпаки, сучасні загрозові моделі передбачають, що кожен сегмент мережі має свої особливості, а тому повинен моніторитися окремо. Сегментація дозволяє локалізувати інциденти, зменшити поверхню атаки, прискорити реагування та запобігати горизонтальному переміщенню зловмисників. Вона також дає змогу визначати різні рівні інтенсивності та глибини моніторингу для різних сегментів, фокусуючись на найбільш критичних з них.

Принцип мінімізації довіри, що є основою концепції Zero Trust, також впливає на організацію моніторингу. У таких системах не передбачається автоматичної довіри до жодного компонента – кожна дія перевіряється, кожен запит автентифікується, кожен канал контролюється. Це означає, що системи моніторингу повинні забезпечувати повну прозорість усіх взаємодій у мережі, включаючи внутрішні комунікації, які раніше могли не вважатися критичними. Такий підхід потребує глибшого рівня аналізу, оскільки навіть легітимна активність може бути використана зловмисником для маскування прихованих дій.

Принцип цілісності даних передбачає, що система моніторингу повинна гарантувати незмінність і достовірність журналів, які збираються та аналізуються. Будь-яке втручання у журнали, їхнє видалення, модифікація або пошкодження може призвести до втрати важливої інформації про інцидент. Саме тому системи моніторингу зазвичай включають механізми контрольних підписів, хешування, розподіленого зберігання та багаторівневого контролю доступу. Для великих організацій питання цілісності логів має також юридичний аспект, оскільки журнали можуть бути використані як докази у випадку розслідування кіберзлочинів або спорів із партнерами.

Надзвичайно важливим принципом побудови систем моніторингу є принцип мінімізації шуму. У сучасних мережах щодня генеруються мільйони подій, більшість з яких не має жодного відношення до безпеки. Якщо система не

забезпечує фільтрації та пріоритезації, аналітики швидко перевантажуються небажаними повідомленнями. Для цього системи використовують фільтри, порогові значення, нормалізацію, агрегацію подій, а також алгоритми машинного навчання, що дозволяють автоматично виділяти справді важливу інформацію. Мінімізація шуму також включає персоналізацію правил моніторингу відповідно до специфіки організації, бізнес-процесів, типових патернів активності та історичних даних.

Принцип інтеграції з іншими системами безпеки передбачає, що сучасний моніторинг не може бути ізольованим інструментом. Він повинен мати можливість взаємодіяти з міжмережевими екранами, засобами контролю доступу, DLP-системами, IDS/IPS, засобами аналізу поведінки користувачів, хмарними платформами та системами автоматизації реагування. Взаємодія забезпечує повноту картини, оскільки дані з різних систем доповнюють одна одну. Наприклад, поєднання логів автентифікації з мережевими журналами дозволяє визначити, чи була підозріла активність результатом легітимного входу або компрометації облікового запису. Стандартизація форматів даних, таких як JSON, CEF, LEEF або ECS, значно спрощує інтеграцію.

Отже, системи моніторингу повинні будуватися на принципі прогнозованості. Це означає, що вони повинні мати здатність не лише фіксувати події, але й передбачати потенційні інциденти на основі історичних закономірностей. Прогнозування стає можливим завдяки машинному навчанню, статистичному аналізу, профілюванню поведінки та використанню зовнішніх джерел інформації про загрози. Прогностичні моделі дозволяють визначати нові вектори атак ще до того, як вони будуть реалізовані, що значно підвищує рівень захисту.

У межах сучасних систем моніторингу особливе значення має принцип своєчасності реагування, який визначає здатність системи не лише виявляти загрози, але й скорочувати проміжок часу між появою інциденту та ініціюванням дій для його нейтралізації. У багатьох випадках навіть кількахвилинна затримка може стати критичною, оскільки сучасні атаки здатні поширюватися з високою швидкістю, експлуатуючи автоматизовані механізми розповсюдження або вразливості програмного забезпечення. Принцип своєчасності базується на високій

продуктивності системи, ефективних алгоритмах аналізу, попередній фільтрації подій, а також автоматизації рутинних дій. Він тісно пов'язаний із здатністю системи працювати в режимі реального часу, обробляти великий обсяг подій за мінімальний інтервал часу та уникати перевантаження, яке може призвести до затримок у виявленні критичних інцидентів. Проте своєчасність реагування значною мірою залежить від точності визначення інциденту, що пов'язано з принципом достовірності аналітики. Достовірність означає, що система повинна мінімізувати ризики хибнопозитивних та хибнонегативних спрацювань. Хибнопозитивні інциденти перевантажують команду реагування, змушують витрачати час на неважливі події та знижують загальну довіру до системи моніторингу. Хибнонегативні, у свою чергу, створюють набагато серйознішу проблему – дозволяють загрозі залишатися непоміченою та розвиватися вглиб системи. Забезпечення достовірності потребує поєднання різних методів: сигнатурного аналізу, поведінкових моделей, евристичних алгоритмів, кореляційних правил та машинного навчання. Важливо, щоб система не лише фіксувала події, але й об'єктивно інтерпретувала їх відповідно до реальних умов інфраструктури. Ще одним важливим принципом є глибина аналізу, яка визначає, наскільки детально система здатна досліджувати події та формувати повну картину активності. Поверховий моніторинг, орієнтований лише на загальні події, не може забезпечити точного виявлення складних або прихованих інцидентів. Глибина аналізу означає можливість вивчення низькорівневих атрибутів подій, таких як параметри системних викликів, деталі протоколів, нестандартні шаблони взаємодії, внутрішні зміни конфігурацій, поведінка процесів, файлові операції, дії користувачів та аномалії мережевого трафіку. Такий глибинний підхід дозволяє виявляти багатоступеневі атаки, які можуть маскувати свої сліди у межах стандартних журналів.

Принцип глибини тісно пов'язаний із принципом багатоджерельності, що передбачає використання максимально широкого спектра джерел даних. У складних інфраструктурах окремі компоненти можуть генерувати події, що не мають явного зв'язку між собою, але разом формують повну картину інциденту. Дані з мережевих датчиків IDS, журнали операційної системи, логи додатків, події

автентифікації, інформація з хмарних сервісів, дані про поведінку контейнерів і мікросервісів, а також телеметрія з кінцевих точок – усе це необхідно об'єднати, нормалізувати й проаналізувати у єдиному інформаційному просторі. Лише багатоджерельний підхід дозволяє виявляти складні інциденти, що використовують кілька різних точок входу та методи приховування активності. У системах моніторингу велику роль відіграє принцип історичності, який полягає у використанні накопичених даних для виявлення інцидентів, які можуть бути непомітними у короткостроковій перспективі. Багато складних загроз можуть розгортатися повільно, виконуючи окремі дії упродовж тривалого часу. Аналіз короточасних подій у таких випадках не дозволяє визначити шкідливу активність. Однак у поєднанні з історичними даними система може розпізнати нетипові зміни, що відбуваються у поведінці користувача або системи. Це дозволяє виявляти довготермінові атаки, часто пов'язані з групами АРТ, які діють приховано та поступово. Здатність системи працювати з великими історичними масивами логів, виконувати ретроспективний аналіз і виявляти латентні загрози є одним із ключових факторів ефективності моніторингу.

Принцип незалежності та нейтральності системи моніторингу передбачає, що вона повинна бути ізольованою від основних бізнес-сервісів, щоб злоумисник не міг легко модифікувати її або знищити журнали. У багатьох випадках злоумисники після проникнення в систему намагаються першочергово вимкнути служби журналювання або модифікувати логи, щоб приховати свої дії. Тому компоненти моніторингу повинні мати власні механізми захисту, використовувати окремі канали зв'язку, захищені протоколи та багаторівневий контроль доступу. У великих організаціях журналювання навіть виноситься у віддалені незалежні середовища або хмарні сервіси, що унеможливорює знищення доказів порушення безпеки. Важливим принципом є також керованість та прозорість взаємодій між компонентами системи моніторингу. У складних інфраструктурах система складається з десятків модулів, які збирають, передають, аналізують і зберігають дані. Якщо взаємодія між ними не є прозорою, виникає ризик втрати інформації, збою в кореляції, дублювання даних або появи незапланованих точок відмови. Принцип керованості означає, що всі модулі мають працювати у межах чітко

визначеної архітектури, з контрольованими каналами зв'язку та стандартизованими протоколами обміну даними. Це забезпечує узгодженість, передбачуваність і повну трасованість подій.

Принцип безпеки самої системи моніторингу є ще одним критичним аспектом. Система, яка слідкує за безпекою, сама стає цінною мішенню для зловмисників. Якщо її буде скомпрометовано, зловмисник отримає можливість приховувати свої дії, змінювати журнали, вимикати правила кореляції або спрямовувати хибні сповіщення. Тому моніторингові системи повинні бути захищені на найвищому рівні, включаючи використання шифрування, сегментацію доступу, апаратні модулі захисту, багатофакторну автентифікацію та регулярний аудит.

Окремого значення набуває принцип заснованості на політиках, що передбачає, що параметри моніторингу, правила обробки подій, пріоритети аналізу та механізми реагування повинні бути чітко визначені й формально затверджені. У великих організаціях політики визначають, які події підлягають обов'язковому моніторингу, які – можуть бути знехтувані, який рівень критичності мають різні типи інцидентів і які дії повинні виконуватися автоматично. Це дозволяє стандартизувати роботу та уникнути хаотичних і непередбачуваних рішень під час інцидентів.

Завершуючи аналіз принципів побудови систем моніторингу та реагування на інциденти, необхідно наголосити, що ефективність таких систем визначається не лише якістю окремих технічних елементів, а й ступенем інтегрованості всієї архітектури у загальну стратегію безпеки організації. Система моніторингу не може існувати як автономний модуль, оскільки інциденти виникають у масштабах усієї інфраструктури – від кінцевих точок і мережевих сегментів до хмарних сервісів та контейнерних платформ. Тому моніторинг повинен бути побудований таким чином, щоб він охоплював увесь технологічний ландшафт організації, забезпечував наскрізну видимість і дозволяв оцінювати стан безпеки в реальному часі. Саме завдяки принципу цілісності та охоплення всієї інфраструктури система стає здатною виявляти складні, розподілені інциденти, які намагаються приховати свою активність, використовуючи розрізненість інформаційних потоків. Важливо підкреслити, що системи моніторингу повинні не лише виявляти порушення, але й

сприяти формуванню загальної кіберстійкості організації. Кіберстійкість – це здатність компанії підтримувати належний рівень функціонування навіть у разі активних *ataques*, швидко відновлюватися після інцидентів та адаптуватися до нових загроз. Справді, система моніторингу є ключовим інформаційним ресурсом, який визначає рівень ситуаційної обізнаності організації. Саме вона дозволяє виявляти не лише поточні атаки, а й слабкі місця, системні проблеми, відхилення у роботі сервісів або порушення в конфігураціях, що можуть стати основою для майбутніх інцидентів. У цьому контексті принцип прогнозування відіграє фундаментальну роль, оскільки дозволяє будувати моделі майбутньої поведінки загроз, готуватися до можливих сценаріїв атак та впроваджувати превентивні заходи.

Принцип інтерактивності в системах моніторингу стає дедалі важливішим у міру того, як інфраструктури ускладнюються. Аналітики повинні мати можливість оперативно взаємодіяти з системою, виконувати гнучкі запити, створювати власні візуалізації, здійснювати фільтрацію та глибокий аналіз даних без необхідності залучення додаткових інструментів. Інтерактивність забезпечує швидкість роботи, гнучкість і точність прийняття рішень. Вона дозволяє не лише реагувати на події *postfactum*, але й досліджувати інцидент у динаміці, виявляючи закономірності, що були непомітні в автоматичному режимі. Такий підхід дає змогу побудувати більш комплексну картину атаки, визначити її фазу, обсяг шкоди та можливі подальші дії зловмисника. Принцип інтелектуалізації систем моніторингу є однією з найсучасніших тенденцій. Йдеться про впровадження алгоритмів машинного навчання, нейронних мереж, моделей статистичного аналізу та поведінкового профілювання, які дозволяють системі самостійно виявляти складні загрози, прогнозувати їх розвиток та адаптуватися до нових умов. Інтелектуальні системи здатні аналізувати величезні масиви даних, формувати поведінкові профілі користувачів та сервісів, визначати аномалії та створювати попереджувальні сповіщення. Це дозволяє значно підвищити точність моніторингу, зменшити кількість хибнопозитивних інцидентів та оптимізувати роботу аналітичних команд. Інтелектуалізація також дозволяє проводити аналіз у режимі реального часу, що є критично важливим у випадку атак, які розгортаються протягом секунд.

Не менш важливим принципом є забезпечення прозорості системи моніторингу для керівництва та управлінських структур. Це не означає, що управлінці повинні бачити всі технічні деталі – натомість вони мають мати доступ до узагальненої, стратегічно важливої інформації про стан кібербезпеки, рівень загроз, частоту інцидентів, їхню критичність та вплив на бізнес. Прозорість дозволяє забезпечити належний рівень контролю, підвищити відповідальність і сформуванати розуміння того, що системи безпеки є не витратною частиною, а стратегічним інвестиційним інструментом.

Завершальним фундаментальним принципом є безперервний розвиток. Система моніторингу не є завершеним продуктом; вона постійно еволюціонує разом із змінами технологічного середовища та появою нових типів загроз. Це означає, що організації повинні інвестувати у регулярні оновлення, удосконалення компонентів системи, навчання персоналу, модернізацію алгоритмів, впровадження нових форм аналітики та розширення функціональних можливостей. Безперервність розвитку забезпечує відповідність системи актуальним викликам та дозволяє зберігати високий рівень захищеності навіть у кризових ситуаціях.

У підсумку можна зазначити, що принципи побудови систем моніторингу та реагування на інциденти формують комплексну, багаторівневу концепцію, що поєднує технічні, аналітичні та організаційні компоненти. Вони визначають здатність системи забезпечувати повну видимість інфраструктури, виявляти загрози в режимі реального часу, мінімізувати ризики, підтримувати безперервність бізнесу та створювати умови для стратегічного розвитку системи безпеки

1.4. Основні проблеми виявлення та обробки інцидентів у сучасних корпоративних мережах

У сучасних корпоративних мережах проблема виявлення та обробки інцидентів інформаційної безпеки є однією з найскладніших і найкритичніших для забезпечення стабільності бізнес-процесів. Стрімкий розвиток ІТ-технологій, впровадження хмарних сервісів, масове використання віддалених робочих місць, поява мікросервісних архітектур і контейнеризації – усе це призвело до значного ускладнення інфраструктур. У таких умовах класичні підходи до моніторингу та реагування виявилися малоефективними, оскільки вони не здатні обробляти великі обсяги даних, швидко і точно класифікувати події, визначати взаємозв'язки між ними та формувати цілісну картину стану безпеки [5,8]. Виявлення інцидентів перетворилося не лише на технічну задачу, але й на аналітичний процес, який потребує глибокого контекстуального розуміння усієї інфраструктури. Однією з фундаментальних проблем є різке збільшення обсягу телеметрії. Кожен інфраструктурний компонент генерує все більше логів, подій і метрик: сервери фіксують системні виклики, мережеві пристрої передають інформацію про трафік, хмарні сервіси генерують журнали API-запитів, системи автентифікації створюють дані про входи, а засоби EDR – детальні артефакти поведінки процесів. В результаті формується потік із мільйонів подій, що надходять щосекунди. Цей «шум» перевантажує традиційні механізми аналізу та призводить до того, що важливі події легко губляться серед численних технічних записів [5,8]. Коректне виявлення загроз у такому середовищі стає задачею надзвичайної складності, оскільки потребує одночасної обробки великого масиву різнорідних даних та забезпечення їхньої кореляції.

Ще однією серйозною проблемою є висока латентність багатьох сучасних атак. Багато зловмисників, намагаючись уникнути виявлення, діють обережно, поступово і непомітно, розтягуючи свою активність у часі. Атаки типу APT можуть розгортатися протягом місяців, формуючи окремі події, що на перший погляд виглядають як нешкідливі: коректний вхід користувача з незвичної геолокації, одиничне створення нетипового процесу, запуск команди PowerShell, що не

суперечить політикам. Кожен із цих індикаторів сам по собі не є інцидентом, але у поєднанні вони формують багатофазну атаку. Виявити такі інциденти можна лише за допомогою глибокого аналізу поведінки, історичних даних та контекстно залежних моделей, що значно ускладнює аналітичний процес. Суттєву складність становить також проблема хибнопозитивних інцидентів. Корпоративні системи часто помилково розцінюють технічні або легітимні події як загрозу. Причини цього криються у недостатній точності сигнатурних правил, неправильній конфігурації засобів моніторингу, відсутності повноцінного контексту та особливостях роботи користувачів. Хибнопозитивні події перевантажують аналітичні команди, призводять до втрати концентрації, витрат часу та ресурсів. У масштабних інфраструктурах кількість хибнопозитивних інцидентів може бути значно більшою, ніж реальних загроз, що робить процес реагування неефективним і потребує надзвичайно точних фільтраційних механізмів.

Окрему складність становить багатовекторність інцидентів, коли зловмисник одночасно атакує кілька компонентів інфраструктури. Такі атаки можуть комбінувати мережеві вторгнення, фішинг, маніпуляції обліковими записами, експлуатацію веб-додатків, використання вразливостей у бібліотеках або контейнерах. Система повинна бути здатною поєднати різні джерела інформації у єдину ланку, виявивши взаємозв'язки між подіями. Саме це часто стає проблемою для платформ старого покоління, які не здатні забезпечити комплексний аналіз у масштабі всієї інфраструктури. Одна з ключових складностей сучасних корпоративних мереж полягає у їхній високій фрагментованості. Інфраструктура дедалі частіше розподіляється між локальними дата-центрами, хмарними платформами, контейнеризованими середовищами та зовнішніми SaaS-сервісами. Кожне з цих середовищ створює власні журнали, має різні формати логів та різний рівень деталізації. Через це сформувати єдину картину подій стає набагато складніше. Будь-які помилки у нормалізації чи уніфікації даних призводять до того, що частина інцидентів залишиться непоміченою – або не буде пов'язана з іншими подіями, які насправді є частиною однієї атаки.

Додатковим чинником складності є обмежена видимість поведінки користувачів та сервісів. У багатьох випадках зловмисники навмисно уникають

технічних уразливостей і натомість застосовують тактики компрометації облікових записів. Після отримання доступу до легітимного аккаунта їхня активність не порушує традиційних правил доступу, а тому виглядає цілком звично. Виявити такі інциденти можна лише через аналіз поведінкових патернів – наприклад, зміну геолокації входів, нетипові операції, збільшення інтенсивності комунікацій або взаємодію з сервісами, які користувач ніколи раніше не використовував. Але навіть на великих підприємствах упровадження поведінкової аналітики потребує значних ресурсів, а тому часто залишається обмеженим або частково реалізованим. Серйозною проблемою є також відсутність централізованої платформи реагування. У багатьох компаніях різні інструменти безпеки працюють розрізнено: антивіруси, IDS/IPS, фаєрволи, журнали ОС та логи застосунків існують окремо й не об'єднані в єдину систему кореляції. Внаслідок цього аналітики змушені вручну збирати інформацію з різних джерел, порівнювати записи та намагатися встановити хронологію подій. Такий підхід значно збільшує час реагування й підвищує ймовірність помилок. Зловмисники ж часто діють комплексно, комбінуючи кілька векторів атаки, і без централізованої кореляції такі інциденти залишаються непоміченими або надто пізно отримують належну увагу.

Ще однією суттєвою проблемою є гостра нестача кваліфікованих фахівців з кібербезпеки. Інцидент-менеджмент вимагає глибоких знань у сфері аналізу логів, мережних протоколів, цифрової криміналістики, поведінкової аналітики та особливостей сучасних атак. У той час як більшість організацій продовжує розширювати свої ІТ-середовища, кількість спеціалістів не зростає відповідними темпами. Це призводить до того, що інциденти часто обробляються з великими затримками або поверхово, без повного аналізу. Перевантаження фахівців великою кількістю подій створює додатковий ризик пропуску критичної інформації, а також формує небезпечну залежність від автоматизованих систем, які самі по собі не можуть гарантувати повноту виявлення. Не менш значущою залишається проблема хибної впевненості у захищеності системи. Багато підприємств покладаються на поодинокі інструменти моніторингу або вважають, що наявність антивіруса, IDS чи файрвола вже є достатньою мірою безпеки. Насправді ж зловмисники активно використовують легітимні компоненти системи та дозволені адміністраторами

утиліти, що ускладнює їхнє виявлення. Класичні сигнатурні методи не здатні розпізнати складні атаки, які не мають заздалегідь відомих шаблонів. Така ситуація створює ілюзію безпеки, тоді як реальна загроза може перебувати всередині системи протягом тривалого часу.

Окремим фактором складності є інтеграція нових технологій, зокрема хмарних сервісів, DevOps-підходів та контейнеризації. З одного боку, вони підвищують гнучкість і швидкість роботи ІТ-підрозділів, але з іншого – створюють нові точки входу для атак. Контейнерні оркестратори, такі як Kubernetes, генерують величезні обсяги телеметрії, які не завжди структуровані або стандартизовані. Хмарні середовища, у свою чергу, додають складності через мультиорендність, динамічні зміни інфраструктури та обмежену прозорість на рівні мережових взаємодій. Усе це призводить до того, що інциденти у сучасних середовищах часто потребують спеціалізованих інструментів аналізу та глибоких знань специфіки цих технологій.

Ще однією проблемою є складність встановлення пріоритетів реагування. У великих організаціях кількість подій може сягати мільйонів на добу, і для команди безпеки важливо розуміти, які з них потребують негайної реакції. Якщо критерії критичності визначені нечітко або базуються лише на технічних характеристиках, аналітики ризикують витратити час на другорядні інциденти. Натомість справді небезпечні атаки можуть залишатися в тіні через відсутність контексту або неправильно визначений рівень пріоритету. Правильна класифікація інцидентів потребує глибокого розуміння бізнес-процесів, критичності активів, типової поведінки користувачів та можливих наслідків атаки.

Значною проблемою залишається також відсутність або недостатність автоматизації процесів реагування. У багатьох компаніях після виявлення інциденту необхідно виконувати значну кількість рутинних дій вручну: блокування облікових записів, ізоляція вузлів, перевірка пов'язаних систем, оновлення конфігурацій або створення звітності. Такі процеси займають багато часу й створюють додаткове навантаження на команду. Автоматизація, зокрема інтеграція з SOAR-платформами, допомагає значно скоротити цей час, однак її впровадження потребує ретельного налаштування, стандартизації внутрішніх процедур і високої якості даних, які надходять у систему.

Однією з найсуттєвіших проблем сучасної інцидент-менеджмент практики залишається відсутність повноцінного післяінцидентного аналізу та ретроспективного опрацювання подій. Багато організацій зосереджуються лише на ліквідації видимих наслідків атаки, не вивчаючи першопричини інциденту. Через це уразливості, які дозволили здійснити атаку, залишаються не виправленими, а зловмисники можуть повторно скористатися тими самими або подібними слабкостями. Відсутність глибокого аналізу також позбавляє організацію можливості вдосконалити процедури, підвищити зрілість процесів реагування та зменшити час реакції у майбутньому. Ще одна вагома проблема полягає у відсутності стандартизованих політик та чітко визначених процедур реагування. Навіть у великих компаніях команди безпеки часто діють на основі індивідуального досвіду або ситуативних рішень, що призводить до хаотичності та непослідовності виявлення та обробки інцидентів. Якщо немає формалізованого процесу, відсутні класифікаційні схеми, не визначені ролі учасників та не задокументовані типові сценарії реагування, ефективність системи значно зменшується. У таких умовах кожен інцидент фактично обробляється заново, що збільшує час виявлення, створює ризик помилок та ускладнює контроль якості виконання дій. Важливим аспектом є також проблема недостатньої інтеграції інструментів безпеки у єдину систему. Багато компаній використовують широкий спектр засобів від антивіруса й EDR до CASB, WAF, SIEM, IDS/IPS та різних систем хмарної безпеки. Однак якщо ці системи не обмінюються даними або роблять це лише частково, команда безпеки отримує фрагментовану картину інцидентів. Відсутність повної кореляції суттєво знижує можливість виявлення складних багатоступеневих атак, що використовують різні точки проникнення або постійно змінюють свою тактику. Без єдиного центру аналізу даних загальна видимість інцидентів різко зменшується, а швидкість реакції падає.

Окрему проблему становить невідповідність між масштабом сучасних загроз та швидкістю розвитку інструментів корпоративної безпеки. Технології атак, включно з автоматизованими бот-мережами, генераторами фішингових кампаній, експлойт-китами та інструментами прихованої ескалації привілеїв, еволюціонують швидше, ніж більшість корпоративних систем встигають оновлюватися. Це

створює ситуацію, коли навіть формально захищена інфраструктура залишається вразливою через застарілі правила кореляції, невчасно оновлені бази сигнатур або недостатньо гнучкі механізми поведінкового аналізу. Протистояти таким атакам без адаптивних інструментів та автоматизованих моделей стає все складніше.

Завершуючи аналіз, варто наголосити, що виявлення та обробка інцидентів у сучасних корпоративних мережах є багатогранною проблемою, яка охоплює технічні, організаційні, аналітичні та кадрові аспекти. Складність інфраструктур, динамічність загроз, обмежені людські ресурси, відсутність стандартизації та неповна видимість створюють умови, у яких ефективний моніторинг можливий лише за умови використання сучасних платформ, здатних до кореляції великих обсягів даних, автоматизації процесів і глибокого поведінкового аналізу.

Висновки до першого розділу:

Проведений аналіз показав, що сучасні корпоративні мережі перебувають під постійним тиском складних і різновекторних кіберзагроз, а інциденти безпеки стають більш прихованими та технологічно розвиненими. Вивчення міжнародних стандартів NIST та ISO/IEC 27035 підтвердило важливість структурованого життєвого циклу інциденту, який охоплює виявлення, аналіз, ліквідацію наслідків та післяінцидентне вдосконалення. Аналіз принципів побудови систем моніторингу засвідчив, що ключовими передумовами ефективного реагування є кореляція подій, централізований збір логів, поведінкова аналітика та масштабованість архітектури.

Водночас виявлено низку проблем, характерних для сучасних інфраструктур: надмірний обсяг телеметрії, фрагментованість систем, обмежена видимість діяльності користувачів та недостатній рівень автоматизації процесів. Ці фактори значно ускладнюють своєчасне та точне виявлення інцидентів. Узагальнюючи результати аналізу, можна зробити висновок, що ефективна система реагування повинна базуватися на комплексному підході й використанні сучасних інструментів моніторингу. Це створює підґрунтя для подальшого дослідження можливостей програмного стеку ELK у контексті побудови моделі реагування на інциденти.

2. АНАЛІЗ ІСНУЮЧИХ ІНСТРУМЕНТІВ ТА ПРОЄКТУВАННЯ МОДЕЛІ СИСТЕМИ НА ОСНОВІ ПРОГРАМНОГО СТЕКУ ELK

2.1. Огляд і порівняння сучасних систем збору та аналізу подій безпеки (SIEM, IDS/IPS, SOAR)

У сучасних корпоративних мережах питання оперативного виявлення та аналізу інцидентів інформаційної безпеки набуває першочергового значення, оскільки загрози стають дедалі витонченішими, а їхня динаміка більш агресивною. З цієї причини підприємства активно впроваджують технології, орієнтовані на централізацію збору подій, глибокий аналіз, кореляцію та автоматизацію реакції. Найважливішу роль у цьому процесі відіграють системи класу SIEM, IDS/IPS та SOAR, які забезпечують ключові функції безпеки від базового моніторингу та виявлення аномалій до автоматизованого реагування на загрози [5,22]. Розуміння їхніх можливостей, відмінностей та взаємодії є фундаментальним для подальшого обґрунтування вибору технологічної платформи, зокрема програмного стеку ELK, як основи моделі інцидент-менеджменту.

Системи SIEM – є центральним елементом інфраструктури безпеки, оскільки забезпечують збір, агрегацію та аналіз величезних масивів даних із різних джерел. Їхня ефективність полягає у здатності об'єднувати події з серверів, мережних пристроїв, систем автентифікації, додатків, хмарних платформ та інших компонентів в єдине інформаційне середовище. Саме SIEM-архітектури дозволяють формувати цілісну картину активності в організації та створюють основу для кореляції подій, яка є критично важливою для виявлення складних атак. Завдяки цьому можна виявляти інциденти, які складаються з кількох взаємопов'язаних дій і не виглядають небезпечними окремо. Важливою особливістю SIEM-систем є здатність обробляти як сигнатурні, так і поведінкові індикатори загроз. Сучасні рішення поєднують класичні механізми виявлення, що базуються на відомих шаблонах атак, із алгоритмами машинного навчання, які дозволяють виявляти аномальні дії, не пов'язані з жодними відомими сигнатурами. Такий підхід дає можливість протидіяти як традиційним загрозам, так і новим формам атак, що постійно модифікуються або впроваджуються у вигляді складних багатоступеневих ланцюгів. Таким чином, SIEM виступає центральною

аналітичною платформою, яка забезпечує високий рівень видимості подій і дозволяє виявляти інциденти на ранніх стадіях.

Паралельно з SIEM важливу роль відіграють системи IDS/IPS (Intrusion Detection/Prevention Systems), які зосереджені на контролі мережевого трафіку та виявленні ознак вторгнення. IDS орієнтовані на пасивний моніторинг та аналіз мережевих пакетів, визначення аномалій та підозрілих патернів у поведінці [30]. У свою чергу, IPS виконує активні дії, блокуючи потенційні атаки або перериваючи небезпечні сесії. IDS/IPS системи є важливою складовою глибинного захисту, оскільки вони фіксують запити та взаємодії, які не завжди відображаються у традиційних лог-файлах або на рівні операційних систем. У поєднанні з SIEM вони формують комплексний підхід, де мережеві аномалії доповнюють загальну аналітику подій. Попри свою ефективність, IDS/IPS мають обмеження, сигнатурні механізми виявлення часто не здатні розпізнати нові, модифіковані або невідомі атаки, що потребує інтеграції поведінкових методів аналізу. У деяких випадках IDS-дані породжують значну кількість хибнопозитивних інцидентів, що ускладнює роботу аналітиків. Проте їхній внесок у побудову багаторівневої структури захисту є незамінним, особливо у великих середовищах із динамічним трафіком.

Окрему нішу займають системи SOAR, які забезпечують автоматизацію та оркестрацію реагування на інциденти. На відміну від SIEM, які зосереджені на зборі й аналізі, SOAR-платформи мають на меті мінімізувати участь людини у рутинних процесах реагування. Вони дозволяють створювати сценарії (playbooks), які автоматично виконують типові дії: блокування облікових записів, ізоляцію вузлів, створення запитів у сторонніх системах, надсилення повідомлень, аналіз артефактів, запуск перевірок і оновлення політик [27,29]. Це значно скорочує час реагування та зменшує навантаження на команди безпеки. SOAR забезпечує також єдність процесів реагування, виключаючи хаотичні та непослідовні рішення. У поєднанні з SIEM такі системи утворюють потужний механізм протидії сучасним інцидентам: SIEM виявляє та аналізує загрози, IDS/IPS фіксує мережеві аномалії, а SOAR автоматично виконує реагування відповідно до визначених процедур. Саме цей комплекс технологій формує сучасну архітектуру безпеки, яка здатна

працювати в режимі реального часу, адаптуватися до загроз і забезпечувати масштабовану підтримку великої кількості подій.

У сучасних умовах цифрової трансформації роль систем збору та аналізу подій безпеки постійно зростає, адже саме вони визначають здатність організації своєчасно реагувати на інциденти. Водночас різноманітність архітектур і функціональних можливостей SIEM, IDS/IPS та SOAR ставить перед спеціалістами завдання правильно інтегрувати ці системи, забезпечити їхню сумісність і уникнути дублювання функцій. Головною проблемою є не технічна складність їхнього впровадження, а організаційна – необхідність побудувати єдиний процес, у межах якого всі ці компоненти працюватимуть узгоджено. Без чітко визначених ролей, взаємодій та потоків даних навіть найдосконаліші інструменти не здатні забезпечити належний рівень захисту. Аналіз функціональних особливостей SIEM-систем показує, що їхня успішність безпосередньо залежить від якості нормалізації та кореляції даних. У великих інфраструктурах логи мають різну структуру, часова синхронізація між джерелами може бути порушена, а обсяги подій іноді перевищують можливості традиційних механізмів індексації. Саме тому сучасні SIEM-платформи дедалі частіше доповнюють класичні реляційні підходи інструментами, здатними працювати з великими даними, розподіленими сховищами, нереляційними індексами, потоковою обробкою подій. Усе це дозволяє досягти більшої гнучкості, швидкості та точності аналізу. Проте ефективність SIEM значною мірою визначається тим, наскільки правильно налаштовані правила кореляції, які повинні враховувати бізнес-контекст, поведінкові моделі та залежності між компонентами системи.

У свою чергу, IDS/IPS системи зіштовхуються з власними викликами. Їхнє застосування в умовах високошвидкісних мереж потребує значних обчислювальних ресурсів, оскільки для аналізу кожного мережевого пакету необхідно виконати велику кількість операцій порівняння, поведінкових перевірок або семантичної інтерпретації трафіку. Крім того, сучасні атаки часто використовують шифрування трафіку, що ускладнює роботу IDS/IPS, обмежує глибину аналізу та змушує організації впроваджувати додаткові технології розшифрування або аналізу метаданих. Ці системи залишаються надзвичайно

важливими для виявлення вторгнень, однак у складних мережових середовищах вони повинні бути доповнені контекстною інформацією з інших джерел, що зазвичай забезпечується SIEM.

SOAR-платформи виступають ключовою складовою інцидент-менеджменту, коли йдеться про масштабні інфраструктури або середовища, де кількість інцидентів перевищує можливості команди аналітиків. Автоматизація реагування не лише скорочує час усунення загроз, але й забезпечує послідовність дій, що є критично важливим для дотримання вимог комплаєнсу, внутрішніх політик та міжнародних стандартів. Практичний досвід показує, що інтеграція SIEM та SOAR дає змогу значно зменшити кількість хибнопозитивних інцидентів, оскільки поведінкові й кореляційні моделі, розроблені у межах SIEM, підкріплюються автоматизованими сценаріями реагування SOAR. Це дозволяє зосередити роботу аналітиків на складних, нетривіальних загрозах, у той час як рутинні операції виконуються автоматично. Особливої уваги заслуговує взаємодія цих систем із зовнішніми джерелами інформації про загрози, такими як threat intelligence-платформи. Отримання, аналіз і застосування даних про тактики, техніки та процедури зловмисників дозволяє підвищити точність кореляції, оптимізувати сигнатурні правила IDS/IPS та формувати ефективні сценарії SOAR. У сучасних умовах безпеки механізм обміну інформацією про загрози стає важливою частиною інфраструктури, оскільки зловмисники дедалі частіше використовують автоматизовані інструменти, які здатні швидко змінювати тактику. Тому адаптивність систем захисту безпосередньо залежить від того, наскільки швидко та якісно вони отримують розвідувальні дані.

Важливим аспектом у порівнянні SIEM, IDS/IPS та SOAR є розуміння того, що кожна з цих систем виконує власну унікальну функцію, але лише їхнє поєднання може забезпечити повний цикл захисту. SIEM забезпечує глибину аналітики та кореляцію подій, IDS/IPS – контроль мережової взаємодії та попередження вторгнень, а SOAR – автоматизацію і стандартизацію процесів реагування. Саме ця взаємодія дозволяє побудувати ефективну систему безпеки, здатну працювати в умовах масштабованості, високої динаміки загроз і великих обсягів даних.

Порівнюючи сучасні системи збору та аналізу подій безпеки, важливо врахувати фундаментальні відмінності у їхніх підходах до виявлення загроз. SIEM-архітектури орієнтовані на глибоку аналітику, історичний контекст та визначення взаємозв'язків між подіями, що дозволяє виявляти складні багатоступеневі атаки. Натомість IDS/IPS фокусуються на аналізі поточного мережевого трафіку та визначенні підозрілих патернів у режимі реального часу. Чимало сучасних атак поєднують мережеві, поведінкові та системні ознаки, тому лише застосування одного підходу не забезпечує достатньої точності. Поєднання SIEM з IDS/IPS дозволяє охопити як низькорівневий трафік, так і високорівневі події доступу, автентифікації та взаємодій між сервісами.

Однією з найважливіших характеристик SIEM-систем є їхня здатність працювати з великими обсягами даних, виконуючи складні кореляційні обчислення. Проте ця можливість створює й суттєві виклики, оскільки ефективність кореляції повністю залежить від якості правил, наявності контексту та повноти джерел даних. Якщо в інфраструктурі відсутні журнали певних сегментів мережі або хмарних сервісів, SIEM не може забезпечити достатнього рівня видимості. У цьому випадку IDS/IPS компенсують прогалини, фіксуючи мережеві взаємодії та формуючи додатковий рівень контролю. Комплексна аналітика можлива лише тоді, коли всі ці системи працюють разом, утворюючи єдиний центр моніторингу.

SOAR-платформи, в свою чергу, відіграють важливу роль у вирішенні проблеми масштабування реагування. У великих організаціях кількість сповіщень, які генерує SIEM чи IDS/IPS, може сягати тисяч на годину. Без автоматизації команда реагування фізично не здатна опрацювати всі події, що створює ризик пропуску критичних інцидентів. SOAR дозволяє оптимізувати цей процес, виконуючи попередню фільтрацію подій, об'єднання споріднених інцидентів та запуск заздалегідь визначених сценаріїв реагування. Це не лише зменшує навантаження на аналітиків, а й робить дії команди швидкими, узгодженими та відповідними до стандартів інцидент-менеджменту.

Окремої уваги потребує питання інтеграції SIEM, IDS/IPS та SOAR з хмарними середовищами, які стають домінуючим елементом сучасної IT-

інфраструктури. Хмарні сервіси генерують велику кількість подій управління доступом, створення ресурсів, виконання операцій у контейнерах або взаємодії між мікросервісами. Ці події часто не відповідають класичним форматам логів і потребують спеціалізованих механізмів збору та нормалізації. SIEM у поєднанні з платформами на кшталт Beats або Fluentd дозволяє отримувати уніфіковані дані з різних хмарних платформ, тоді як IDS/IPS можуть аналізувати мережеві потоки між контейнерами або віртуальними мережами. SOAR забезпечує автоматизоване реагування на події в хмарному середовищі, наприклад ізоляцію ресурсів або автоматичне відкликання ключів доступу.

У сучасних системах безпеки важливим чинником стає також можливість ретроспективного аналізу інцидентів. SIEM дозволяє досліджувати історію подій за тривалий період, що є критичним у випадку прихованих або повільних атак. IDS/IPS зазвичай не мають можливості зберігати великі обсяги даних тривалий час, але їхня інформація про трафік може бути збережена у SIEM і використана для подальшої кореляції. SOAR доповнює цей підхід тим, що дозволяє відтворити послідовність дій, які були виконані під час реагування, забезпечуючи повну прозорість процесу. Підсумовуючи, можна зазначити, що сучасні системи збору та аналізу подій безпеки не конкурують між собою, а виконують взаємодоповнювальні функції. SIEM забезпечує централізовану аналітику, IDS/IPS – детальний контроль мережевої активності, а SOAR – швидке та стандартизоване реагування. Лише їхня синергія дозволяє сформувати цілісний підхід до інцидент-менеджменту, який відповідає вимогам масштабованості, точності та оперативності сучасного інформаційного середовища.

Під час порівняння різних систем аналізу та обробки подій безпеки особливо важливо враховувати їхню здатність адаптуватися до динамічних умов корпоративних інфраструктур. Сучасні інформаційні середовища характеризуються швидкими змінами: додаванням нових сервісів, масштабуванням хмарних ресурсів, оновленням програмного забезпечення й активним впровадженням DevOps-практик. У таких умовах SIEM, IDS/IPS і SOAR повинні не лише ефективно взаємодіяти, але й залишатися гнучкими, здатними приймати нові джерела даних, реагувати на зміни структури мережі та адаптувати

свої правила та моделі аналізу. Відсутність цієї гнучкості призводить до зниження точності виявлення інцидентів та створює умови, коли атакувальники можуть скористатися слабко контрольованими зонами інфраструктури.

Не менш важливим аспектом є якість даних, що надходять до цих систем. Ефективність SIEM чи SOAR безпосередньо залежить від структурованості, повноти та достовірності логів, тоді як IDS/IPS потребують чітко визначених точок моніторингу й можливості аналізувати трафік у критичних сегментах мережі. Неправильно налаштовані джерела логів, різні формати даних, відсутність часової синхронізації або неповний набір подій значно знижують ефективність навіть найпотужніших рішень. Саме тому сучасні системи безпеки орієнтуються на стандартизацію формату логів та централізацію їхнього зберігання, що пізніше буде важливим аргументом на користь використання програмного стеку ELK.

Важливо усвідомлювати, що комплексне застосування SIEM, IDS/IPS та SOAR це не просто технічна інтеграція різних рішень, а формування цілісного процесу управління інцидентами. У цьому процесі SIEM забезпечує глобальну видимість та аналіз, IDS/IPS – раннє попередження про вторгнення, а SOAR стандартизоване та швидке реагування. Взаємодія цих систем складає базову інфраструктурну платформу, на якій може бути побудована сучасна служба інцидент-менеджменту. Саме така інтеграція дозволяє скоротити час виявлення, зменшити кількість хибних сигналів, підвищити точність аналізу та забезпечити безперервність безпеки у великих інформаційних середовищах.

У підсумку можна зазначити, що сучасні системи збору та аналізу подій мають суттєві відмінності, однак їхнє ефективне поєднання створює необхідні передумови для побудови комплексної архітектури кіберзахисту. SIEM, IDS/IPS та SOAR виконують різні, але взаємодоповнювальні функції, що дозволяє створити багаторівневий підхід до виявлення й реагування на інциденти. Системна синергія цих технологій формує основу для подальшого проектування моделі реагування на інциденти на платформі ELK, яка поєднує потужні можливості збору, обробки, індексації та візуалізації подій безпеки. Саме ці властивості ELK роблять його перспективним інструментом для створення ефективної системи інцидент-менеджменту.

2.2. Аналіз архітектури та можливостей програмного стеку ELK

Програмний стек ELK (Elasticsearch, Logstash, Kibana) став одним з найбільш поширених рішень для обробки великих обсягів журналів та подій у сучасних інформаційних системах. Його популярність пояснюється не лише високою продуктивністю та гнучкістю, але й архітектурним підходом, що дозволяє розглядати систему не як окремий застосунок, а як платформу, здатну адаптуватися до будь-якого типу даних та інфраструктури. На відміну від традиційних корпоративних рішень, які часто побудовані як монолітні продукти, ELK пропонує модульну структуру, де кожен компонент виконує власну спеціалізовану функцію. Такий підхід дозволяє масштабувати систему, змінювати її конфігурацію, підключати нові джерела даних і забезпечувати гнучкість під потреби конкретної інфраструктури.

Центральним елементом стеку є Elasticsearch – розподілена пошукова і аналітична система, спеціалізована на швидкій індексації та пошуку великих масивів даних [12,17]. Його архітектура базується на принципі горизонтального масштабування: дані розподіляються між кількома вузлами кластеру, що дозволяє обробляти величезні обсяги журналів та виконувати складні запити з мінімальною затримкою. Це робить Elasticsearch придатним для задач безпеки, де швидкість пошуку та можливість миттєвого аналізу великого масиву подій є вирішальними. Використання шардів і реплік дозволяє забезпечити відмовостійкість і високу доступність – ключову вимогу для моніторингових систем, що працюють у режимі 24/7.

Другим фундаментальним компонентом стеку є Logstash – універсальний інструмент для збору, фільтрації та трансформації даних [13,18]. Його сила полягає у можливості обробляти інформацію практично з будь-якого джерела, незалежно від формату, структури чи типу подій. Logstash дозволяє нормалізувати дані ще до того, як вони потрапляють в Elasticsearch, що значно покращує якість кореляції та аналітики. Маршрутизація потоків, застосування умовних фільтрів, підтримка різних протоколів і модулів – усе це робить Logstash ключовим компонентом для побудови системи збору даних, яка повинна працювати з різнорідними журналами корпоративної мережі.

Третім компонентом стеку виступає Kibana – інструмент для візуалізації даних, побудови панелей моніторингу та аналітичних інтерфейсів [14,19]. Основна цінність Kibana полягає в тому, що вона дозволяє представити складні взаємозв'язки між подіями у зрозумілому та структурованому вигляді. У контексті інцидент-менеджменту Kibana стає потужним інструментом для аналізу часових закономірностей, виявлення аномалій, порівняння активності різних сегментів мережі та формування звітів. Завдяки високій інтерактивності інтерфейсу аналітик може швидко змінювати параметри вибірки, досліджувати події у глибині та знаходити залежності, які неочевидні у сирих логах.

Архітектура ELK дозволяє системі працювати одночасно як інструмент для оперативного моніторингу, так і як аналітична платформа для глибинного розслідування інцидентів. Це досягається завдяки чіткій ролі кожного компонента: Logstash відповідає за якість та однорідність даних, Elasticsearch – за швидкість обробки та масштабованість, а Kibana – за інтерпретацію та подання результатів у формі, зручній для аналітичної роботи. Саме така взаємодія створює основу, на якій можна будувати повноцінну систему реагування на інциденти.

Однією з ключових переваг стеку ELK є його природна здатність інтегруватися з широким спектром джерел подій, що робить його ефективним інструментом у середовищах, де інфраструктура включає десятки різних технологічних платформ. Це досягається завдяки екосистемі Beats – легковагових агентів збору даних, які доповнюють можливості Logstash. Filebeat, Metricbeat, Packetbeat, Winlogbeat та інші агенти дозволяють отримувати дані без необхідності встановлення важких або ресурсомістких компонентів на кінцеві точки [15,23]. Використання Beats у поєднанні з Logstash створює адаптивну систему збору інформації, що зменшує навантаження на інфраструктуру та забезпечує стабільний потік подій у реальному часі.

Стек ELK має ще одну важливу характеристику – він побудований за принципом відкритості та розширюваності. На відміну від традиційних комерційних рішень, де більша частина функціоналу прихована у закритих модулях, ELK надає широкі можливості для кастомізації. Це дозволяє налаштовувати правила парсингу, створювати власні типи індексів, змінювати

схему даних, підключати додаткові модулі обробки інформації та навіть розширювати функціонал за допомогою плагінів. Така відкритість особливо важлива для інцидент-менеджменту, оскільки атаки можуть мати унікальні ознаки, які стандартні рішення не завжди здатні інтерпретувати коректно. Натомість ELK дозволяє адаптувати систему під специфіку окремої організації.

Особливої уваги заслуговує механізм зберігання та індексування даних в Elasticsearch, який є фундаментом продуктивності всієї платформи. На відміну від класичних баз даних, Elasticsearch використовує так званий інвертований індекс – структуру, що забезпечує надзвичайно швидкий пошук за будь-яким атрибутом події. Це дозволяє обробляти мільйони документів за секунди, виконувати складні агрегації, порівнювати події між різними часовими періодами та шукати рідкісні або аномальні записи. Для систем безпеки така можливість є критично важливою, оскільки аналіз інциденту може вимагати роботи з великими масивами історичних даних, пошуку реліктових артефактів або дослідження повторюваних патернів. Крім того, ELK забезпечує гнучку модель керування життєвим циклом даних. У випадку великих інфраструктур, де обсяг журналів може сягати терабайтів на добу, неможливо зберігати всі події в одному кластері без оптимізації. Elasticsearch дозволяє використовувати різні класи зберігання, створювати політики ротації індексів, автоматично переміщати старі дані на вузли з дешевшим сховищем або зовнішні репозиторії. Завдяки цьому платформа не лише обробляє великий обсяг даних у реальному часі, але й ефективно управляє історичними архівами, які необхідні для форензики, аудиту та ретроспективного аналізу інцидентів.

Важливим аспектом, який відрізняє ELK від багатьох традиційних інструментів безпеки, є його здатність об'єднувати оперативний та ретроспективний аналіз у межах єдиної архітектури. Великі інфраструктури часто мають розділені системи: одну для поточного моніторингу, іншу – для довгострокового аналізу. ELK поєднує ці можливості, дозволяючи одночасно обробляти «гарячі» дані, які постійно оновлюються, та працювати з глибокими історичними записами. Це робить платформу надзвичайно гнучким інструментом для виявлення нападів, що розгортаються протягом тривалого періоду, включно з АРТ-кампаніями, які часто потребують поєднання миттєвої реакції та тривалого

аналізу. Таким чином, стек ELK – це не просто набір інструментів для перегляду логів, а повноцінна аналітична платформа, здатна виконувати роль серця системи моніторингу та реагування на інциденти. Його архітектурні особливості, гнучкість, модульність та значна екосистема супутніх інструментів створюють потенціал для побудови системи, яка може масштабуватися разом із розвитком інфраструктури та зростанням складності кіберзагроз.

Однією з визначальних властивостей стеку ELK є його здатність забезпечувати високий рівень прозорості даних та гнучкої аналітики, що особливо важливо для інцидент-менеджменту, де будь-який фрагмент інформації може стати вирішальним для розуміння маршруту атаки або встановлення факту компрометації. Kibana, як кінцевий інструмент візуалізації, виступає не просто інтерфейсом для перегляду записів, а повноцінним аналітичним середовищем. Завдяки можливості створювати складні графіки, часові діаграми, кореляційні панелі та інтерактивні візуалізації, аналітик отримує змогу швидко переходити від високорівневих показників до детального розгляду окремих подій. Це особливо актуально в ситуаціях, коли потрібно простежити розвиток інциденту у часі, визначити його початкову точку або відстежити переміщення зловмисника між сегментами мережі. Додатковою перевагою Kibana є підтримка спеціалізованих модулів, орієнтованих на безпеку, таких як Elastic Security, що інтегрує візуалізацію з механізмами виявлення загроз. Ці модулі дозволяють автоматично формувати попередження на основі правил, поведінкових моделей або аномалій, а також відображати їх у вигляді інцидентів, структурованих за тактиками та техніками MITRE ATT&CK. Таким чином, стек ELK перетворюється зі звичайного сховища логів на платформу, яка здатна проводити загрозоорієнтований аналіз, допомагаючи командам безпеки швидше інтерпретувати події та співставляти їх із відомими сценаріями атак.

Важливо також відзначити можливості ELK у побудові уніфікованої моделі даних. У середовищах, де логи надходять із різних систем, часто виникає проблема відсутності стандартів, що ускладнює кореляцію. Elasticsearch підтримує концепцію схем (mappings), які дозволяють наперед визначити типи полів, формат часових міток, індексу структуру та спосіб обробки нестандартних значень.

Завдяки цьому ELK може забезпечити узгодженість даних і коректну роботу алгоритмів аналізу навіть у випадках, коли журнали за своєю природою є хаотичними або непослідовними. Єдина структура даних також спрощує інтеграцію зі сторонніми інструментами аналітики, машинного навчання або обробки потоків.

Окремо слід підкреслити роль машинного навчання у можливостях ELK. Хоча стек історично був орієнтований на пошук та індексацію, останні версії Elasticsearch містять вбудовані модулі ML, здатні автоматично виявляти аномалії, прогнозувати поведінкові відхилення або оцінювати нормальні параметри системи. Для інцидент-менеджменту це означає можливість виявити небажану активність без потреби у складних ручних налаштуваннях. Модель машинного навчання може, наприклад, визначити нетипову активність адміністративного облікового запису, а Kibana відразу візуалізує цю аномалію як інцидент, що потребує перевірки. Поєднання ML з історичними даними робить ELK не просто реактивним інструментом, а платформою, здатною до проактивного виявлення загроз.

Значущим є й те, що ELK підтримує побудову розподілених, географічно рознесених кластерів, які забезпечують надзвичайно високу доступність навіть у ситуаціях повної відмови окремих сегментів інфраструктури. Це робить стек привабливим для великих організацій, де моніторинг не може припинятися навіть на кілька хвилин. Завдяки системі реплікації даних та розподілених запитів, навіть критичні обсяги інформації можуть оброблятися без втрати продуктивності. Така відмовостійкість дозволяє використовувати ELK як основу для системи реагування на інциденти, що функціонує безперервно, незалежно від стану окремих вузлів.

Таким чином, архітектурні можливості ELK виходять далеко за межі стандартних інструментів моніторингу. Вони дозволяють будувати масштабовані, гнучкі, високопродуктивні та інтелектуальні платформи, здатні забезпечити і поточний моніторинг, і глибоку аналітику, і загрозоорієнтований аналіз. Усе це робить ELK придатним базовим інструментом для побудови моделі інцидент-менеджменту, яка може адаптуватися до сучасного загрозового середовища. Важливою характеристикою стеку ELK, яка вирізняє його серед традиційних інструментів для обробки подій безпеки, є його придатність для побудови

комплексних інтеграційних рішень. Завдяки відкритим API, підтримці REST-запитів, вебхуків та модулів для різних мов програмування Elasticsearch легко включається у зовнішні технологічні екосистеми. Це означає, що платформа може не лише отримувати дані, але й передавати результати аналізу в інші системи керування інцидентами, SOAR-рішення, інструменти контролю доступу чи навіть автоматизовані механізми мережевого захисту. Така інтерактивність робить ELK не просто аналітичним сховищем, а активним компонентом системи безпеки, здатним впливати на інші елементи інфраструктури та підсилювати загальну стратегію реагування.

Ще однією суттєвою перевагою ELK є його здатність працювати з даними незалежно від структури, що є важливим у корпоративних мережах, де події можуть надходити в хаотичному або напівструктурованому вигляді. Logstash та Beats дозволяють адаптувати потоки журналів до єдиної схеми, а Elasticsearch індексує їх так, щоб навіть неоднорідні дані могли бути ефективно проаналізовані. Завдяки цьому ELK залишається придатним інструментом у ситуаціях, коли організація впроваджує нові сервіси, змінює конфігурацію мережі або підключає зовнішні платформи. Система фактично «вчиться» працювати з новими типами подій без необхідності радикальної зміни архітектури або функціоналу.

Не менш важливим є те, що ELK підтримує масштабування не лише в горизонтальному напрямку, але і в плані складності аналітичних задач. Кластери Elasticsearch можуть збільшуватися разом із зростанням обсягів журналів без значних змін у конфігурації. Це створює передумови для застосування ELK у ролі універсального аналітичного ядра, яке здатне працювати у середовищах від кількох серверів до інфраструктур корпоративного рівня з сотнями вузлів. Для систем безпеки така масштабованість є вирішальною, оскільки забезпечує стабільність роботи навіть тоді, коли обсяг подій зростає у кілька разів або під час активних фаз атак.

Останнім важливим аспектом, який підкреслює цінність ELK у контексті моделі реагування, є його здатність поєднувати оперативну роботу команд SOC з більш глибокими задачами цифрової криміналістики та аудиту. Elasticsearch дозволяє зберігати події тривалий час, а Kibana – швидко відновлювати їхній

контекст. Це дає можливість відтворити ланцюжок дій зломисника, визначити точну послідовність доступів, виконаних команд або змін у конфігураціях. Така функціональність критично необхідна у випадках, коли інцидент розслідується постфактум і ключові артефакти мають бути збережені та представлені у вигляді доказів. Таким чином, ELK виступає універсальною платформою, здатною підтримувати як поточне реагування, так і ретроспективний аналіз.

У підсумку можна зазначити, що архітектура та можливості стеку ELK роблять його одним із найперспективніших інструментів для побудови гнучкої системи реагування на інциденти в корпоративних мережах. Його модульність, відкритість, масштабованість та аналітичний потенціал дозволяють адаптувати платформу до різних технологічних середовищ та сценаріїв атак. Саме ці характеристики роблять ELK оптимальною основою для моделі, яка поєднує збір подій, їх нормалізацію, глибоку аналітику та зручні механізми візуалізації – усе те, що необхідно для сучасного інцидент-менеджменту.

2.3. Порівняльний аналіз систем логування та моніторингу Graylog, Wazuh і ELK-стеку

У сучасних інформаційних системах вирішальне значення має правильний вибір платформи для логування та моніторингу, оскільки саме від неї залежить якість виявлення інцидентів, швидкість реагування та рівень видимості подій у мережі. Graylog, Wazuh та ELK-стек є трьома популярними платформами, які застосовуються у корпоративних мережах різного масштабу. Попри те, що ці системи мають спільні цілі, такі як збір, аналіз та візуалізація подій, а їхня архітектура, підходи до обробки даних та сфер застосування істотно різняться. Саме ці відмінності визначають, яка платформа буде ефективною у певному середовищі, враховуючи вимоги до масштабованості, глибини аналізу, швидкості індексації та наявності вбудованих механізмів безпеки.

Graylog позиціонується як платформа централізованого логування, у якій ключовий акцент зроблено на зручність роботи з журналами та побудову ефективних потоків обробки даних [27]. Її архітектура побудована навколо

використання Elasticsearch як бекенду для зберігання індексів та MongoDB як метаданого сховища. Завдяки цьому система забезпечує стабільність роботи навіть при значних обсягах журналів. Однак у порівнянні з повноцінним ELK-стеком Graylog має більш обмежену гнучкість у налаштуванні структури даних, оскільки більшість операцій з індексацією приховані «під капотом». Це робить платформу зручною для швидкого розгортання, але менш придатною для середовищ, де дані мають високу варіативність або потребують глибокої кастомізації.

На відміну від Graylog, який був розроблений насамперед як лог-менеджер, Wazuh є повноцінною платформою безпеки, яка поєднує функції HIDS, FIM, виявлення вразливостей, моніторинг конфігурацій, а також збір і аналіз подій. Він включає власний агент, що встановлюється на кінцеві точки й здатний контролювати зміни у файлах, процесах, реєстрі, налаштуваннях системи та інших критичних областях. Важливою перевагою Wazuh є те, що він також інтегрується з Elasticsearch та Kibana, використовуючи їх як аналітичний бекенд і візуалізатор. Завдяки цьому Wazuh фактично розширює можливості ELK-стеку, надаючи йому функції, яких немає «з коробки» у стандартній конфігурації, зокрема готові правила виявлення загроз, набір сигнатур, механізми аналізу поведінки та модуль відповідності стандартам безпеки.

ELK-стек, у свою чергу, володіє найбільш універсальними можливостями, оскільки поєднує гнучкість, масштабованість і здатність працювати з будь-якими типами подій. Його сильна сторона полягає у тому, що він не нав'язує жодної конкретної моделі безпеки або способу збору даних. Це дозволяє адаптувати платформу під специфічні потреби підприємства від простого логування до складних сценаріїв кореляції подій. Водночас саме ця універсальність потребує більш високого рівня кваліфікації для налаштування, оскільки аналітичні правила, структури індексів та парсинг подій створюються з нуля або імпортуються з окремих модулів. Таким чином, ELK можна розглядати як «конструктор», який дає максимальні можливості, але вимагає відповідних ресурсів для впровадження.

Порівняння підходів цих платформ до масштабованості демонструє їхню різну орієнтацію. Graylog добре підходить для середніх інфраструктур із стабільними потоками журналів, де основним завданням є централізований перегляд і базовий

аналіз. Wazuh, навпаки, оптимізований для високодинамічних середовищ, де важливо контролювати активність на кінцевих точках та швидко реагувати на порушення політик. ELK-стек забезпечує максимальну масштабованість, дозволяючи обробляти як невеликі потоки подій, так і величезні масиви телеметрії в умовах розподілених середовищ і хмарних інфраструктур. При порівнянні цих систем важливо врахувати не лише їхні функціональні можливості, а й концептуальний підхід до обробки подій. Graylog насамперед орієнтований на роль централізованого лог-менеджера з акцентом на зручність керування потоками журналів та швидкий доступ до даних. Його сильна сторона – інтуїтивно зрозумілий інтерфейс та готові механізми обробки вхідних повідомлень, що роблять його привабливим для підприємств, які не потребують складної аналітики чи глибокої кастомізації. Водночас Graylog менш гнучкий у контексті кореляції подій: він здебільшого працює як система пошуку та фільтрації, а не як платформа для побудови складних алгоритмів виявлення. Тому виявлення комбінованих інцидентів або аномалій, що розгортаються у кількох сегментах мережі, вимагає додаткових рішень або використання зовнішніх модулів.

Wazuh займає унікальне місце між простим лог-менеджером і повноцінною платформою безпеки. Його функції контрольованого агентського моніторингу дозволяють отримувати інформацію не лише про події у системних журналах, а й про реальні зміни у файловій системі, роботу служб, аномалії в поведінці процесів та потенційні порушення політик. Це робить Wazuh більш орієнтованим на безпеку кінцевих точок, ніж Graylog і навіть класичний ELK у базовій конфігурації. У той же час, Wazuh не є самодостатньою системою аналізу даних, він потребує Elasticsearch і Kibana як аналітичного ядра. Таким чином, він не конкурує з ELK, а фактично розширює його, додаючи функції, які не є стандартними для систем логування.

У випадку ELK-стеку відмінною рисою є універсальність і архітектурна відкритість, що дозволяє використовувати його як платформу для логування, моніторингу, кореляції та візуалізації. Цей стек легко масштабується та може приймати практично будь-які типи даних, що робить його надзвичайно привабливим у середовищах із великою кількістю джерел подій. Однак гнучкість

має і зворотній бік: ELK потребує значних зусиль для побудови логічно узгодженої моделі даних, розроблення правил парсингу, налаштування пайплайнів у Logstash або Beats, оптимізації індексів та побудови дашбордів. Утім, у разі правильного налаштування ця платформа дає глибину аналізу, недосяжну для Graylog та більшості відкритих рішень.

Особливо важливою відмінністю між платформами є підхід до виявлення загроз. Graylog не містить вбудованих механізмів кореляції або сигнатурної аналітики безпеки він радше інструмент для пошуку даних. Wazuh має великий набір вбудованих правил, заснованих на OSSEC Framework, які дозволяють автоматично виявляти події безпеки без додаткового програмування. ELK-стек в базовому вигляді не надає власних правил, але дозволяє інтегрувати модулі Elastic Security, які пропонують як поведінкові, так і сигнатурні методи виявлення. Це робить ELK найбільш гнучким, але і елементом, що вимагає найбільших часових інвестицій для створення повноцінних правил. Ще однією важливою різницею є підхід до продуктивності та оптимізації. Graylog показує себе добре у середніх середовищах, але масштабування потребує додавання нових Elasticsearch-вузлів і стає складним при великих навантаженнях. Wazuh залежить від продуктивності інфраструктури Elasticsearch, але використання агентів дозволяє зменшити обсяг непотрібних даних ще до їх надсилання. ELK, завдяки розподіленим кластерам і горизонтальному масштабуванню, здатен обробляти величезні потоки логів, але вимагає суттєвих ресурсів та досвіду адміністрування.

Таким чином, уже на цьому етапі стає очевидно, що ці платформи не є взаємовиключними конкурентами, а радше займають різні ніші: Graylog для простого централізованого логування, Wazuh для індивідуального контрольованого моніторингу безпеки, ELK для побудови комплексної аналітичної інфраструктури.

Ще одним важливим аспектом у порівнянні цих платформ є рівень інтерактивності та глибини роботи з даними, який вони пропонують аналітику. Graylog забезпечує простий і зрозумілий інтерфейс пошуку, що дозволяє швидко знаходити події за ключовими словами, часовими діапазонами та атрибутами. Такий підхід є зручним для оперативних адміністративних завдань, але має

обмеження, коли виникає потреба у складному кореляційному аналізі або багатовимірних вибірках. У таких випадках Graylog може виступати лише як проміжний інструмент, оскільки його можливості глибокої аналітики суттєво поступаються ELK-стеку, а вбудовані засоби візуалізації менш адаптивні й не передбачають складних структурованих панелей.

Wazuh, на відміну від Graylog, пропонує більш насичену безпекову аналітику завдяки тісній інтеграції з Kibana. Інтерфейс Wazuh-App дозволяє працювати із заздалегідь структурованими секціями: подіями безпеки, станом конфігурацій, виявленням вразливостей, FIM-порушеннями, реакціями агентів тощо. Це надає аналітику достатньо глибину для розуміння того, як саме поводить себе система в контексті загроз. Проте Wazuh не претендує на роль універсальної платформи аналітики, його інтерфейс орієнтований насамперед на розслідування інцидентів безпеки, а не на загальний аналіз телеметрії або великі дані. Саме тому Wazuh часто застосовується як доповнення до ELK-стеку, а не як його повноцінна заміна. ELK, у свою чергу, пропонує найширший інструментарій для аналітики, завдяки глибокій інтерактивності Kibana та багатим можливостям агрегацій, часових аналізів і кастомних візуалізацій. Це дає можливість формувати складні дашборди для SOC: кореляційні карти, аналітичні панелі MITRE ATT&CK, динамічні графи залежностей, аналізи аномалій, карти мережеских потоків та інші механізми, які значно спрощують виявлення прихованих або багаторівневих атак. Крім того, наявність вбудованих засобів машинного навчання дозволяє ELK виконувати поведінкову аналітику, що виходить за межі можливостей Graylog і Wazuh. Важливо й те, що аналітик у Kibana має повний контроль над структурою візуалізацій, що дозволяє адаптувати інтерфейс під конкретні сценарії SOC та конкретні операційні процеси.

Ще однією структурною відмінністю між системами є підхід до маршрутизації потоків даних. У Graylog цей процес доволі формалізований: система працює з обмеженим набором input-механізмів та має чітко визначений конвеєр обробки повідомлень. Це спрощує роботу, але обмежує можливості тонкого налаштування. Logstash і Beats, які використовуються в ELK, забезпечують значно ширшу свободу: вони можуть приймати дані з практично будь-яких джерел, виконувати

складну фільтрацію, умовну логіку, геообробку, парсинг різних форматів та перенаправлення в окремі індекси.

У Wazuh, завдяки агентам, маршрутизація відбувається на рівні кінцевої точки: дані, що не потрібні, можуть бути відфільтровані ще до надсилання. Таким чином, у цій сфері платформи демонструють суттєво різні підходи: Graylog пропонує стандартизованість, Wazuh – контрольованість на агентському рівні, ELK – максимальну кастомізацію.

Значущою характеристикою є також підхід платформ до інтеграції зі сторонніми інструментами. Graylog підтримує інтеграції здебільшого через «Inputs» та обмежений набір плагінів. Wazuh має значно ширшу екосистему інтеграцій від AWS GuardDuty до VirusTotal, проте орієнтований саме на безпекові сценарії. ELK-стек завдяки відкритим API виступає універсальним центром, здатним приймати дані від будь-якої платформи, будь-якого формату і будь-якого протоколу. Тому він найчастіше стає основою для побудови SIEM-рішень з високим рівнем інтеграції.

У сукупності ці аспекти демонструють, що різниця між Graylog, Wazuh і ELK полягає не тільки у функціональності, а й у концептуальній філософії платформ: Graylog – це простота і швидкість доступу до журналів, Wazuh – це контроль безпеки на рівні агентів, ELK – широкомасштабна гнучка аналітика. В таблиці 2.1. відображено порівняння цих платформ.

Таблиця.2.1 – Порівняльна таблиця Graylog, Wazuh і ELK-стеку

Критерій	Graylog v.6.x	Wazuh v.4.7.0	ELK-стек v.8.11
Тип системи	Лог-менеджер	HIDS + Security Monitoring	Аналітичний та моніторинговий стек
Призначення	Централізоване логування	Безпека кінцевих точок	Комплексна аналітика і моніторинг
Архітектура	Elasticsearch + MongoDB	Wazuh Manager + агенти + ELK	Elasticsearch + Logstash/Beats + Kibana
Збір логів	Inputs (Syslog, GELF і т.д.)	Агенти (FIM, vuln, конфігурації)	Будь-які джерела через Beats/Logstash

Аналітика	Базова	Вбудовані правила OSSEC	Повна, включно з ML та Elastic Security
Кореляція подій	Обмежена	Середня (HIDS логіка)	Висока, повністю кастомна
Масштабованість	Середня	Висока з ELK	Дуже висока (кластеризація)
Безпеківі функції	Мінімальні	HIDS, FIM, compliance, vuln scan	Залежить від модулів, підтримує SIEM/EDR
Простота впровадження	Висока	Середня	Низька (вимагає експертизи)
Для SOC	Підходить частково	Підходить для endpoint security	Найкраще рішення для SOC
Типові сценарії	Збір логів, аудит	Моніторинг вузлів, політики	SIEM, реагування, кореляція, аналітика
Тип системи	Лог-менеджер	HIDS + Security Monitoring	Аналітичний та моніторинговий стек

Порівняння систем Graylog, Wazuh та ELK-стеку показує, що кожна з них займає власну нішу в екосистемі логування та моніторингу безпеки. Graylog найбільше підходить для задач централізованого збору та пошуку логів завдяки простоті, легкості розгортання та зручності інтерфейсу, проте він має обмежені можливості глибокої аналітики. Wazuh орієнтований на безпеку кінцевих точок і має розвинені функції HIDS, детекції вразливостей і контролю конфігурацій, але залежить від Elasticsearch та Kibana як аналітичної основи. ELK-стек виступає найгнучкішою та найбільш масштабованою платформою, здатною приймати дані з необмеженої кількості джерел, виконувати складну кореляцію подій та підтримувати повноцінні сценарії інцидент-менеджменту. Таким чином, вибір конкретної системи залежить від вимог організації: простоти для Graylog, безпеки кінцевих точок для Wazuh, комплексної аналітики та масштабування для ELK.

2.4. Розробка моделі системи реагування на інциденти на базі ELK.

Проектування моделі системи реагування на інциденти на основі стеку ELK передбачає інтеграцію декількох взаємодоповнюючих компонентів, серед яких центральну роль відіграє Wazuh як джерело структурованої безпекової телеметрії. ELK забезпечує масштабоване середовище для зберігання, індексації та аналізу великих масивів подій, тоді як Wazuh надає контекстну інформацію про стан кінцевих точок, конфігураційні зміни, вразливості, аномалії та потенційні індикатори компрометації. Такий дуальний підхід дозволяє сформувати модель реагування, де аналітичний потенціал ELK посилюється детекційними механізмами Wazuh, створюючи комплексне інцидент-орієнтоване середовище. Основою моделі є архітектура, що передбачає збирання подій з декількох рівнів корпоративної мережі: серверів, робочих станцій, мережевого обладнання, хмарних платформ та прикладних сервісів. Wazuh-агенти розташовуються на кінцевих точках і виконують попередню нормалізацію та класифікацію подій, включаючи контроль цілісності файлів, виявлення небажаних процесів, аналіз журналів доступу та реєстрацію змін у системі. Ці події передаються до Wazuh Manager, де застосовуються вбудовані правила безпеки, засновані на OSSEC, які дозволяють оперативно ідентифікувати підозрілу активність. Нормалізовані дані потім надходять у Elasticsearch, де вони зберігаються та аналізуються разом із логами з інших джерел.

У межах запропонованої моделі ELK-стек виконує функцію централізованої аналітичної платформи. Elasticsearch забезпечує швидкий пошук і кореляцію подій, Kibana – побудову високорівневих дашбордів для SOC-аналітиків, а Logstash або Beats відповідають за маршрутизацію потоків даних. Завдяки цьому система стає здатною охоплювати повний ланцюжок появи інциденту: від первинного сигналу з Wazuh-агента до створення індикаторів загрози, аналізу кореляційних зв'язків та формування тригерів для реагування. Модель орієнтована не лише на виявлення окремих аномалій, а й на визначення їхнього взаємозв'язку в контексті відомих тактик і технік зловмисників.

Важливо, що така модель поєднує у собі два підходи до моніторингу: сигнатурний, який забезпечує Wazuh через свої правила, та поведінковий, який може реалізовуватися засобами Elasticsearch ML або кастомними кореляційними моделями. Це дозволяє системі реагувати як на відомі загрози (наприклад, спроби ескалації привілеїв, зміну системних параметрів, появу підозрілих служб), так і на нові аномальні патерни. У результаті модель стає адаптивною і здатною працювати в умовах постійного оновлення загрозового ландшафту. Таким чином, запропонована модель реагування на інциденти базується на поєднанні аналітичної потужності ELK та безпекових можливостей Wazuh. Її ефективність визначається збалансованою взаємодією компонентів: Wazuh забезпечує глибину даних, ELK – масштабування та аналітику, а SOC-аналітики отримують повну видимість подій і здатність швидко інтерпретувати інциденти.

Ефективність запропонованої моделі системи реагування значною мірою визначається чіткою організацією потоків даних між компонентами Wazuh та ELK. У межах архітектури ключовим завданням є забезпечення безперервного надходження телеметрії від кінцевих точок, її правильна класифікація та передача до централізованої аналітичної системи. Потоки подій починаються на рівні Wazuh-агентів, які встановлюються на робочі станції, сервери та інші критичні вузли. Вони не лише передають журнали, але й виконують первинний аналіз: контролюють зміни в системних файлах (FIM), перевіряють налаштування безпеки, аналізують журнали аудиту, відстежують запуски процесів, здійснюють опитування вразливостей. Така локальна обробка дозволяє відсіювати зайві події та зменшує навантаження на центральний сервер.

Після початкової обробки дані потрапляють до Wazuh Manager, де до них застосовується потужний набір правил на основі OSSEC. Ці правила дозволяють розпізнавати широкий спектр інцидентів, включаючи brute-force атаки, підозрілі зміни файлів, несанкціоноване створення облікових записів, втручання у конфігурацію системи, взаємодію із критичними службами та аномальні системні процеси. На цьому етапі формується первинний рівень реагування: події класифікуються за рівнями важливості, визначаються потенційні загрози, а також створюються попередні сповіщення. Таким чином, Wazuh виступає центральним

фільтром, який перетворює початковий потік сирих даних на структуровану безпекову інформацію.

Передача нормалізованих подій у Elasticsearch відкриває можливість глибокого кореляційного аналізу. На цьому рівні модель системи реагування набуває інтелектуального характеру: події різних типів та з різних вузлів можуть бути зіставлені між собою на основі часових міток, атрибутів користувачів, мережевої активності чи змін конфігурації. Elasticsearch дозволяє виявляти приховані закономірності, наприклад послідовні спроби проникнення в окремі сегменти мережі, систематичні невдалі входи на кількох серверах або однотипні зміни у конфігурації робочих станцій. Такий аналіз є основою для раннього виявлення атаки ще до того, як вона набуде критичного масштабу.

Kibana у цій моделі відіграє роль інструменту, що дозволяє аналітикам оперативно інтерпретувати корельовані події та будувати логічні ланцюжки інцидентів. Завдяки дашбордам, які відображають вразливості, активність агентів, статуси політик, аномальні процеси чи мережеві взаємодії, SOC-фахівці отримують комплексне уявлення про загальний стан безпеки. Особливо важливим є те, що Kibana дозволяє створювати детальні панелі для конкретних сценаріїв, таких як виявлення lateral movement, підозріла зміна прав доступу або підготовка системи до запуску шкідливого програмного забезпечення. Візуальна складова не лише спрощує аналіз, але й пришвидшує ухвалення рішень під час інцидентів.

Розроблена модель передбачає також формування правил реагування, що дозволяють автоматизувати окремі етапи управління інцидентами. Wazuh здатний не лише виявляти інциденти, але й ініціювати дії реагування – блокувати IP-адреси, відключати облікові записи, завершувати небажані процеси, змінювати конфігурації системи або надсилати тривожні повідомлення в SOC. У свою чергу, ELK, за рахунок додаткових модулів (наприклад Elastic Security), може формувати тригери на основі складних кореляційних умов і передавати їх в інші системи для виконання реакцій. У результаті модель підтримує як ручне, так і частково автоматизоване реагування, що дозволяє суттєво зменшити час виявлення інцидентів і оптимізувати навантаження на команду безпеки.

2.5. Механізми кореляції подій, формування правил виявлення та оповіщення

Ефективність системи реагування на інциденти безпосередньо залежить від якості механізмів кореляції, які дозволяють трансформувати потік розрізнених подій у цілісну картину інциденту. У запропонованій моделі на базі ELK та Wazuh процес виявлення загроз базується на багаторівневій логіці, що включає попередню фільтрацію, нормалізацію, сигнатурний аналіз та складну кореляцію.

Фундаментальним елементом детекції є механізм декодерів та правил (Decoders & Rules) системи Wazuh. Процес обробки розпочинається з декодування сирих логів, що надходять від агентів. Декодери витягують ключові поля (IP-адреси, імена користувачів, шляхи до файлів, ідентифікатори процесів) та приводять їх до єдиного формату JSON. Це є критично важливим етапом, оскільки дозволяє уніфікувати дані з гетерогенних джерел (Windows Event Log, Syslog, Apache logs тощо) для подальшого аналізу.

Кореляція подій у системі реалізується через набір правил, описаних у форматі XML [11]. Правила класифікуються за рівнями (від 0 до 15), де рівень 0 означає ігнорування події, а рівень 12 і вище сигналізує про критичну загрозу. Механізм кореляції базується на наступних принципах:

- Прямий збіг (Atomic Matching). Спрацювання правила при виявленні конкретного рядка або регулярного виразу (наприклад, «Failed password»).
- Частотна кореляція (Frequency-based). Правило спрацює, якщо певна подія повторюється задану кількість разів за визначений проміжок часу. Це дозволяє детектувати атаки типу Brute Force або DDoS (наприклад, правило: «8 невдалих спроб входу з однієї IP-адреси протягом 120 секунд»).
- Логічна послідовність (Sequence-based). Виявлення ланцюжків подій, які окремо можуть виглядати легітимно, але разом свідчать про атаку. Наприклад, створення нового користувача, додавання його до групи адміністраторів та негайне видалення логів.

– Кореляція за джерелом/призначенням (Same source/destination). Групування подій, що мають спільні атрибути (наприклад, одна IP-адреса атакує різні порти або різні сервери в мережі).

Після обробки у Wazuh Manager, події, що досягли порогового рівня небезпеки, передаються до Elasticsearch. Тут вступає в дію другий рівень аналітики, а саме можливості Elastic Security та мови запитів KQL (Kibana Query Language) або EQL (Event Query Language). EQL дозволяє писати запити для пошуку складних загроз, що розтягнуті у часі або розподілені між різними хостами, використовуючи такі оператори як sequence та join.

Система оповіщення (Alerting) є фінальним етапом логіки виявлення. Вона налаштовується таким чином, щоб мінімізувати кількість хибних спрацювань (False Positives). Оповіщення можуть генеруватися як на рівні Wazuh (через інтеграції з API поштових сервісів, Slack, Microsoft Teams), так і на рівні Kibana (через механізм Watcher або Kibana Alerts). Пріоритет надається оповіщенням у реальному часі для подій високого ступеня критичності, тоді як події середнього рівня агрегуються у періодичні звіти для аналітиків SOC.

Таким чином, комбінація вбудованого рушія правил Wazuh та гнучких можливостей запитів Elasticsearch створює надійний механізм, здатний виявляти як відомі сигнатурні загрози, так і складні аномалії поведінки.

2.6. Модель взаємодії компонентів системи та архітектура потоків даних

На найнижчому рівні ієрархії знаходяться агенти Wazuh, які функціонують як легковагові сервіси на кінцевих точках під управлінням операційних систем Windows, Linux та macOS. Агент встановлює зашифрований канал зв'язку з Wazuh Manager через порт 1514, використовуючи власний протокол передачі. За замовчуванням конфігурація передбачає використання протоколу TCP, який забезпечує гарантовану доставку пакетів та контроль цілісності потоку даних, хоча архітектура системи підтримує і протокол UDP для специфічних мережеских умов, де критичною є мінімізація затримок. Безпека комунікації забезпечується процедурою попередньої реєстрації (Enrollment), під час якої агент та сервер

обмінюються спільним ключем (Shared Key). Цей ключ використовується для симетричного шифрування всього трафіку за алгоритмом AES-256 (Blowfish у старіших версіях), що унеможливлює перехоплення журналів (Man-in-the-Middle) та ін'єкцію фальшивих подій у корпоративній мережі. Критично важливим елементом архітектури агента є механізм запобігання втраті подій при обривах зв'язку. Для цього реалізовано внутрішній буфер за принципом «leaky bucket». Якщо зв'язок з менеджером втрачено, події не відкидаються, а накопичуються локально в черзі та відправляються пакетами (flushing) автоматично після відновлення з'єднання, що гарантує безперервність моніторингу навіть у нестабільних мережах.

Wazuh Manager виступає як центральний вузол (hub) обробки та аналізу даних. У межах розробленої моделі він виконує подвійну роль: функціонує як сервер HIDS (Host-based Intrusion Detection System) для управління агентами та як попередній фільтр і нормалізатор для SIEM-системи. Внутрішня архітектура менеджера складається з декількох взаємопов'язаних демонів. Ключовим компонентом є процес Analysisd – головний аналітичний рушій, який виконує декодування логів та їх співставлення з правилами безпеки. Прийом з'єднань від агентів та управління шифруванням покладається на процес Remoted, який масштабується залежно від кількості підключених ендпоінтів. Окрім аналізу логів, менеджер включає модуль Integrator, що дозволяє підключати зовнішні системи збагачення даних, наприклад, API VirusTotal або бази даних Threat Intelligence, для перевірки хешів файлів та IP-адрес у реальному часі. Результатом роботи менеджера є генерація файлу alerts.json структурованого журналу, де кожен рядок є окремим JSON-об'єктом, що містить повну інформацію про інцидент: ідентифікатор агента, спрацювавше правило, відповідність тактикам MITRE ATT&CK та оригінальний текст події.

У запропонованій архітектурі компонент Filebeat виконує роль транспортувальника (shipper), забезпечуючи надійну передачу оброблених подій до бази даних. Він зчитує файл alerts.json у режимі tail, відслідковуючи появу нових записів у реальному часі. Важливим аспектом інтеграції є використання спеціалізованого модуля Wazuh для Filebeat. Цей модуль не просто пересилає текст, а й забезпечує сумісність з Elastic Common Schema (ECS) – відкритим стандартом

структурування даних. Це означає, що специфічні поля Wazuh автоматично перейменовуються та приводять до стандарту, який нативно розуміє Elasticsearch (наприклад, поле `src_ip` трансформується у `source.ip`). Для забезпечення надійності доставки в умовах пікових навантажень або тимчасової недоступності бази даних використовується внутрішня черга повідомлень (Spooler). У високонавантажених системах налаштовується буферизація в пам'яті (mem-queue) або на диску (disk-queue), що дозволяє реалізувати механізм зворотного тиску (Backpressure management) та уникнути втрати критичних даних про інциденти.

Отримані та нормалізовані дані акумулюються у кластері Elasticsearch, який виступає основним сховищем подій безпеки. Для забезпечення кворуму та високої відмовостійкості модель передбачає розгортання кластера у конфігурації з мінімум трьох вузлів (Master-eligible та Data nodes). Збереження подій відбувається у індексах, що автоматично створюються системою щодня та розбиваються на шарди (shards) для ефективного розподілу навантаження між вузлами. З метою оптимізації використання апаратних ресурсів та забезпечення швидкодії системи впроваджується політика життєвого циклу індексів (ILM - Index Lifecycle Management). Ця політика регламентує автоматичне переміщення даних між різними фазами зберігання. Процес починається з активної «гарячої» фази (Hot phase), де дані знаходяться на швидких SSD-накопичувачах для забезпечення високої швидкості запису та пошуку в реальному часі. Згодом, зазвичай через 7 днів, індекси переходять у «теплу» фазу (Warm phase), де вони доступні лише для читання та можуть зберігатися на більш дешевих магнітних носіях. Завершується цикл автоматичним видаленням індексів після спливання встановленого терміну зберігання (наприклад, 90 днів), що відповідає політикам збереження даних організації та вимогам регуляторів.

Взаємодія аналітиків SOC із системою та візуалізація даних здійснюється через веб-інтерфейс Kibana, що комунікує з Elasticsearch за допомогою REST API через порт 9200 TCP. Стандартні можливості платформи суттєво розширюються завдяки інтеграції спеціалізованого плагіну Wazuh App, який надає зручний графічний інтерфейс для моніторингу безпеки, управління агентами та перегляду статусів відповідності стандартам (PCI DSS, GDPR). Важливим аспектом

архітектури на цьому рівні є забезпечення розмежування прав доступу (RBAC - Role Based Access Control) засобами X-Pack Security або OpenSearch Security. Це дозволяє гнучко налаштовувати ролі користувачів, наприклад, надаючи повний адміністративний доступ для інженерів безпеки для налаштування правил та конфігурацій, або ж обмежуючи права лише переглядом дашбордів (Read-only) для керівництва та зовнішніх аудиторів, забезпечуючи таким чином конфіденційність та цілісність налаштувань системи реагування.

Висновки до другого розділу:

У другому розділі реалізовано комплексний підхід до проєктування системи реагування на інциденти, що базується на детальному порівняльному аналізі сучасних інструментів моніторингу безпеки. За результатами дослідження функціональних можливостей систем класів SIEM, IDS та SOAR, було обґрунтовано доцільність інтеграції платформи Wazuh із програмним стеком ELK. Таке поєднання дозволяє створити економічно ефективне та гнучке середовище, що відповідає вимогам сучасних центрів безпеки (SOC), забезпечуючи повний цикл обробки даних – від моніторингу цілісності файлів до глибинної аналітики загроз.

Спроектowana архітектура системи відзначається модульністю та чітким розподілом ролей між компонентами, де агенти Wazuh, Filebeat, Elasticsearch та Kibana взаємодіють через захищені канали зв'язку. Це гарантує надійність транспортування телеметрії, її нормалізацію згідно зі стандартом ECS та безпечне зберігання. Особливу увагу в моделі приділено механізмам виявлення інцидентів, які поєднують детерміновані сигнатурні правила з імовірнісним поведінковим аналізом, а також інтеграцію з матрицею MITRE ATT&CK для контекстуалізації атак [16,20]. Запропонована модель, що включає політики життєвого циклу даних та розмежування доступу, створює необхідне теоретичне та архітектурне підґрунтя для наступного етапу роботи. Це дозволяє перейти до практичної реалізації системи, її розгортання в тестовому середовищі та експериментальної перевірки ефективності розроблених сценаріїв реагування.

3. РЕАЛІЗАЦІЯ ТА ОЦІНКА ЕФЕКТИВНОСТІ МОДЕЛІ СИСТЕМИ РЕАГУВАННЯ НА ІНЦИДЕНТИ

3.1. Практична реалізація системи на основі програмного стеку ELK

Для експериментальної перевірки запропонованої архітектури та оцінки ефективності механізмів реагування було розгорнуто тестовий полігон, що імітує типову інфраструктуру корпоративної мережі. В якості платформи для моделювання обрано середовище емуляції мереж GNS3.

Вибір GNS3 зумовлений необхідністю відтворення складної мережевої топології, яка включає сегментацію мережі, маршрутизацію трафіку та емуляцію активного мережевого обладнання. На відміну від класичних гіпервізорів (VirtualBox, VMware), GNS3 дозволяє інтегрувати віртуальні машини (QEMU/KVM), Docker-контейнери та образи реальних маршрутизаторів (наприклад, Cisco IOS або MikroTik RouterOS) у єдиному віртуальному просторі. Це дає можливість досліджувати інциденти не лише на рівні хостів, а й на рівні мережевої взаємодії, що є критично важливим для комплексних систем безпеки.

Архітектура розгорнутого стенду базується на принципі сегментації зон безпеки. У середовищі GNS3 створено топологію, що складається з наступних компонентів: сегмент SOC, User LAN, Мережеве ядро, атакуючий сегмент.

Сегмент SOC – це виділена підмережа (Management VLAN), в якій розміщено сервер безпеки. Цей вузол виконує роль центрального колектора логів та аналітичної платформи. Для його реалізації використано віртуальну машину на базі Ubuntu Server інтегровану в топологію GNS3 як QEMU-вузол.

Користувачьким сегментом є підмережа, що імітує робоче середовище співробітників. Тут розміщено «машину-жертву» під управлінням ОС Windows 10 Enterprise, на якій встановлено агентські модулі моніторингу. Версії програмного забезпечення вказано в додатку Г.

Мережеве ядро відповідає за комутацію та маршрутизацію між сегментами. Забезпечує це віртуальний маршрутизатор, налаштований для імітації для імітації шлюзу підприємства. Він також забезпечує вихід у зовнішню мережу через NAT-хмару (GNS3 NAT Cloud) для завантаження оновлень сигнатур та пакетів.

Атакуючим сегментом є зовнішній вузол на базі Kali Linux, підключений до зовнішнього інтерфейсу маршрутизатора, що дозволяє емулювати атаки ззовні периметра мережі. Така конфігурація дозволяє максимально наблизити умови експерименту до реальних, забезпечуючи ізоляцію шкідливого трафіку та контроль над потоками даних між сегментами.

Реалізація серверної частини системи моніторингу виконана з використанням технології контейнеризації Docker. Це рішення обрано задля забезпечення відтворюваності розгортання, ізоляції сервісів та можливості швидкого масштабування. На вузлі Server-SOC встановлено Docker Engine та Docker Compose. Розгортання програмного стеку здійснено в режим Single-Node Cluster, що є достатнім для тестового середовища. Схема мережі зображена на рис.3.1.

Конфігураційний файл `docker-compose.yml` визначає параметри запуску ключових контейнерів:

- `wazuh.manager` - основний сервіс обробки подій, що слухає порти 1514 (TCP/UDP) для агентів та 55000 (API).
- `Wazuh-indexer` – основний сервіс обробки подій, що слухає порти 1514 (TCP/UDP) для агентів та 55000 (API).
- `wazuh.dashboard` – це веб-інтерфейс (на базі Kibana/OpenSearch Dashboards), що забезпечує візуалізацію даних та управління системою через порт 443.

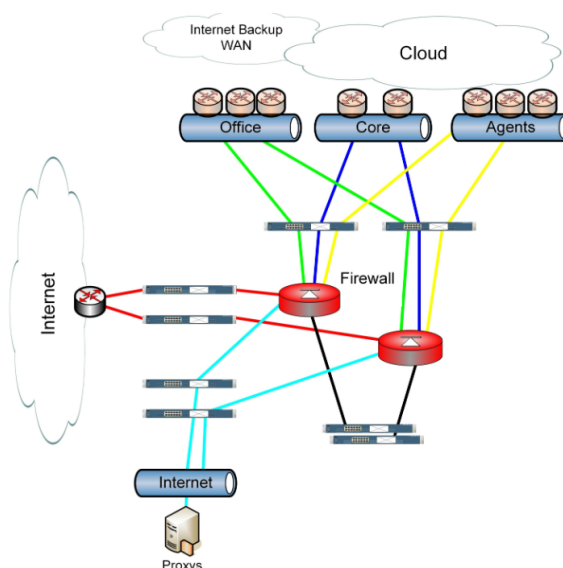


Рисунок 3.1 – Модель мережі

Особливу увагу при налаштуванні приділено конфігурації ресурсів. Для стабільної роботи JVM (Java Virtual Machine) у середовищі Elastic/Indexer виділено 4 ГБ оперативної пам'яті, а системні ліміти хост-машини (vm.max_map_count) збільшено до значення 262144, що є обов'язковою вимогою для роботи пошукового кластера. Конфігурація відображена на рис.3.2.

```
root@osboxes:/home/osboxes# sysctl -w vm.max_map_count=262144
vm.max_map_count = 262144
root@osboxes:/home/osboxes#
```

Рисунок 3.2 – Демонстрація збільшення ліміту оперативної пам'яті

Для збору телеметрії з кінцевої точки використано Wazuh Agent. Процес підключення агента до системи реалізовано через механізм автореєстрації.

На стороні клієнта у конфігураційному файлі ossec.conf вказано IP-адресу сервера SOC у віртуальній мережі GNS3. Важливим етапом налаштування стала активація додаткових модулів моніторингу, які за замовчуванням вимкнені:

- Syscollector необхідний для інвентаризації програмного та апаратного забезпечення.
- SCA (Security Configuration Assessment) – необхідний для перевірки налаштувань ОС на відповідність стандартам безпеки (CIS Benchmarks).
- Active Response – дозволяє виконання автоматичних команд блокування з боку сервера.

```
root@osboxes:/home/osboxes/wazuh-docker/single-node# sudo docker-compose -f generate-indexer-certs.yml run --rm generator
Pulling generator (wazuh/wazuh-certs-generator:0.0.1)...
0.0.1: Pulling from wazuh/wazuh-certs-generator
e4daedc954fb5: Pull complete
b73f4d11a520: Pull complete
8f200922197d: Pull complete
55a86de68c5c: Pull complete
Digest: sha256:ea8b03a68b67bae0e164d82b232eae54dd132c2aacd8d3118ba8705df9364a4
Status: Downloaded newer image for wazuh/wazuh-certs-generator:0.0.1
Creating single-node generator run ... done
The tool to create the certificates exists in the in Packages bucket
03/12/2025 18:36:02 INFO: Admin certificates created.
03/12/2025 18:36:02 INFO: Wazuh indexer certificates created.
03/12/2025 18:36:03 INFO: Wazuh server certificates created.
03/12/2025 18:36:03 INFO: Wazuh dashboard certificates created.
Moving created certificates to the destination directory
Changing certificate permissions
Setting UID for wazuh manager and worker
Setting UID for wazuh manager and worker
root@osboxes:/home/osboxes/wazuh-docker/single-node# docker compose up -d
unknown shorthand flag: 'd' in -d

Usage: docker [OPTIONS] COMMAND [ARG...]

Run 'docker --help' for more information
root@osboxes:/home/osboxes/wazuh-docker/single-node# docker-compose up -d
Creating volume "single-node-wazuh-api-configuration" with default driver
Creating volume "single-node-wazuh-etc" with default driver
Creating volume "single-node-wazuh-logs" with default driver
Creating volume "single-node-wazuh-queue" with default driver
Creating volume "single-node-wazuh-var-multiprocess" with default driver
Creating volume "single-node-wazuh-integrations" with default driver
Creating volume "single-node-wazuh-active-response" with default driver
Creating volume "single-node-wazuh-agentless" with default driver
Creating volume "single-node-wazuh-wodles" with default driver
Creating volume "single-node-filebeat-etc" with default driver
```

Рисунок 3.3 – Демонстрація процесу розгортання SIEM Wazuh

```

root@osboxes:/home/osboxes/wazuh-docker/single-node# docker ps
CONTAINER ID   IMAGE                                COMMAND                  CREATED        STATUS        PORTS
NAMES
fc91b0256171   wazuh/wazuh-dashboard:4.7.0        "/entrypoint.sh"        38 seconds ago Up 37 seconds 443/tcp, 0.0.0.0:443->5601/tcp, [::]:443->5601/tcp
single-node_wazuh.dashboard_1
47b09e07f9b4   wazuh/wazuh-manager:4.7.0          "/init"                 40 seconds ago Up 39 seconds 0.0.0.0:1514-1515->14-1515/tcp, [::]:1514-1515->1514-1515/tcp, 0.0.0.0:514->514/udp, [::]:514->514/udp, 0.0.0.0:55000->55000/tcp, [::]:55000->55000/tcp, 1516/tcp
single-node_wazuh.manager_1
5b9c52c68bab   wazuh/wazuh-indexer:4.7.0          "/entrypoint.sh open600" 40 seconds ago Up 39 seconds 0.0.0.0:9200->9200/tcp, [::]:9200->9200/tcp
single-node_wazuh.indexer_1
root@osboxes:/home/osboxes/wazuh-docker/single-node#

```

Рисунок 3.4 – Демонстрація списку Docker контейнерів

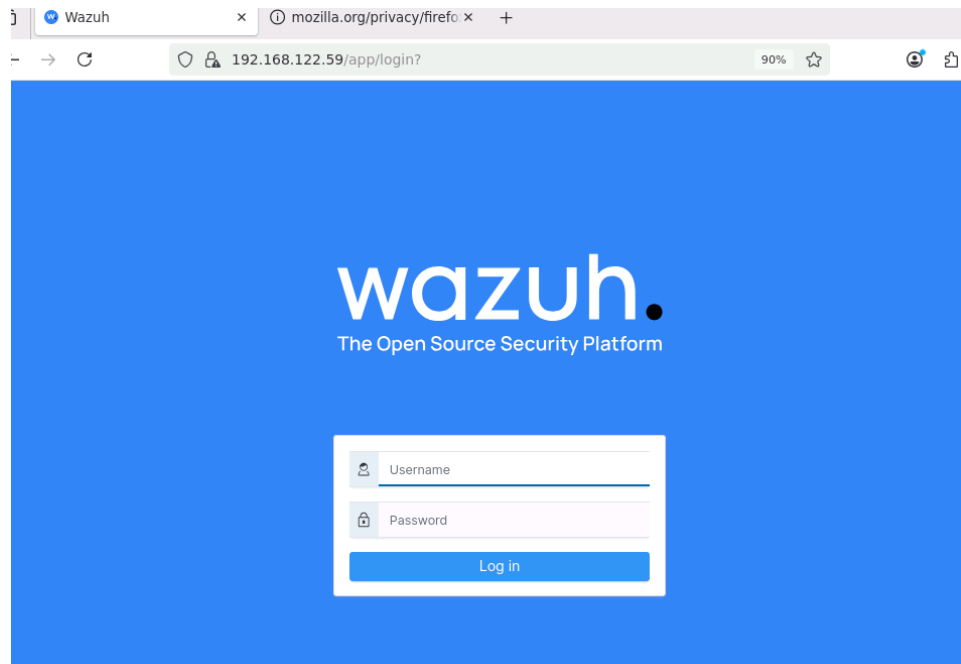


Рисунок 3.5 – Демонстрація вітального вікна Wazuh

Після входу до системи було перейдено до меню розгортання агентів. Першочергово Wazuh пропонує обрати операційну систему для встановлення свого агенту.

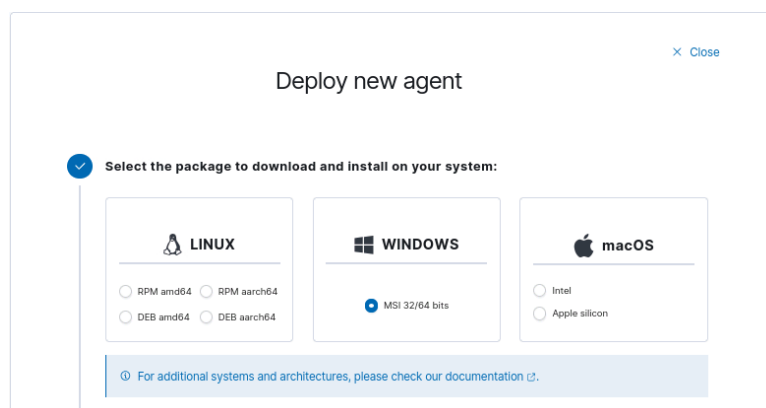


Рисунок 3.6 – Демонстрація вибору операційної системи для агента

Наступним важливим кроком після вибору операційної системи на якому буде розгорнуто агент є конфігурування IP-адреси та імені хоста. IP-адреса має відповідати IP компонента Wazuh Server. Важливим також є те, що вікно конфігурації попереджає адміністратора, що ім'я пристрою має бути унікальним і не може бути зміненим після розгортання.

The screenshot shows a configuration window for a Wazuh agent. It has two main sections: 'Server address' and 'Optional settings'. The 'Server address' section includes a text input field containing '192.168.122.59'. The 'Optional settings' section includes a text input field containing 'Windows-PC'. Below this field is a yellow warning box with the text: 'The agent name must be unique. It can't be changed once the agent has been enrolled.' There are also links for help and a link to the Wazuh documentation.

Рисунок 3.7 – Демонстрація конфігурування IP-адреси та hostname

Вирішальними кроками є копіювання команди для встановлення агента (можливе розгортання у автоматизованому варіанті через GPO або Ansible) та запуск на кінцевому пристрої. Після чого запускається відповідна сервісна служба.

The screenshot shows two steps in a process. Step 4 is titled 'Run the following commands to download and install the Wazuh agent:' and shows a PowerShell command: `Invoke-WebRequest -Uri https://packages.wazuh.com/4.x/windows/wazuh-agent-4.7.0-1.msi -OutFile $(env:tmp)\wazuh-agent; msixexec.exe /i $(env:tmp)\wazuh-agent /q 'WAZUH_MANAGER="192.168.122.59" WAZUH_AGENT_NAME="Windows-PC" WAZUH_REGISTRATION_SERVER="192.168.122.59"`. Step 5 is titled 'Start the Wazuh agent:' and shows a command prompt command: `NET START WazuhSvc`.

Рисунок 3.8 – Демонстрація автоматично сконфігурованої команди для розгортання Wazuh агента

Після виконання команди розгортання через PowerShell на кінцевому вузлі у Wazuh Dashboard з'являється доданий пристрій, де вказано базову інформацію про ім'я хоста, його IP-адресу, групу, операційну систему версію та поточний стан.

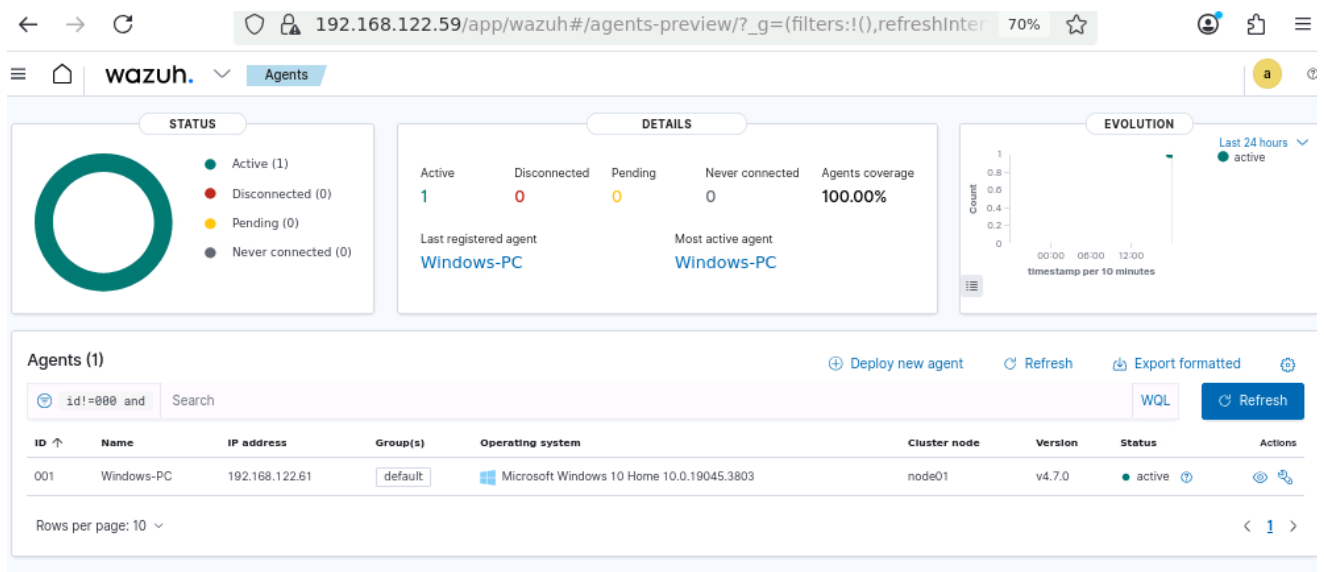


Рисунок 3.9 – Демонстрація успішно розгорнутого агента у Wazuh Dashboard

Зв'язок серверу з агентом підтверджує коректність налаштування зашифрованого каналу зв'язку та успішний обмін ключами шифрування. Час синхронізації склав менше 5 секунд, що свідчить про стабільність мережевого з'єднання у віртуальному середовищі GNS3. Таким чином, розгорнута інфраструктура повністю готова до збору подій та проведення експериментів з моделювання кібератак.

3.2. Збір і нормалізація подій безпеки з різних джерел, побудова правил кореляції та дашбордів для моніторингу інцидентів

Ефективність функціонування розробленої системи реагування на інциденти безпосередньо залежить від повноти та якості вхідних даних. У межах реалізованого тестового полігону процес збору телеметрії організовано за гібридною схемою, що передбачає використання агентського моніторингу для робочих станцій та без-агентського (agentless) збору логів для мережевого обладнання. Ключовим завданням цього етапу є не лише акумуляція «сирих» даних, але і їх нормалізація – приведення різномірних форматів журналів до єдиної структури JSON, придатної для індексації та автоматизованого аналізу.

Основним генератором подій безпеки у спроектованій топології виступає робоча станція «Windows-PC» під управлінням ОС Windows 10 Enterprise. Базові засоби журналювання Windows Event Log (канали Security, System, Application) забезпечують фіксацію основних подій: входу користувачів (Event ID 4624/4625), зміни політик аудиту та управління обліковими записами. Проте для виявлення складних векторів атак (таких як DLL Injection, Process Hollowing або використання безфайлових загроз PowerShell) стандартної деталізації недостатньо.

Для вирішення цієї проблеми в систему інтегровано драйвер розширеного моніторингу Microsoft Sysmon (System Monitor). Інтеграція Sysmon із Wazuh-агентом дозволяє отримувати гранулярну інформацію про активність на рівні ядра ОС. Зокрема, налаштовано збір наступних критичних категорій подій:

- Process Creation (Event ID 1). Фіксація повного командного рядка запуску процесів, що дозволяє виявляти підозрілі аргументи (наприклад, закодовані команди PowerShell) та обчислення хеш-сум (MD5, SHA256) виконуваних файлів для звірки з базами Threat Intelligence.
- Network Connection (Event ID 3). Співставлення процесу з мережевим з'єднанням, що дозволяє виявляти активність C2 (Command & Control) серверів.
- File Creation Time Changed (Event ID 2). Виявлення технік антифореnzіки (timestomping), коли зловмисники змінюють час створення файлів для приховування своєї присутності.

Технічна реалізація збору даних здійснена шляхом модифікації конфігураційного файлу агента ossec.conf на хості «Windows-PC». До секції <localfile> додано блок конфігурації, що активує зчитування каналу Microsoft-Windows-Sysmon/Operational у форматі eventchannel, що забезпечує парсинг XML-структури подій безпосередньо на стороні агента перед відправкою. Додану частину конфігураційного файлу представлено в додатку А.

```
System Monitor v15.15 - System activity monitor
By Mark Russinovich and Thomas Garnier
Copyright (C) 2014-2024 Microsoft Corporation
Using libxml2. libxml2 is Copyright (C) 1998-2012 Daniel Veillard. All Rights Reserved.
Sysinternals - www.sysinternals.com

Sysmon64 installed.
SysmonDrv installed.
Starting SysmonDrv.
SysmonDrv started.
Starting Sysmon64..
Sysmon64 started.
```

Рисунок 3.10 – Демонстрація встановлення та старту Sysmon

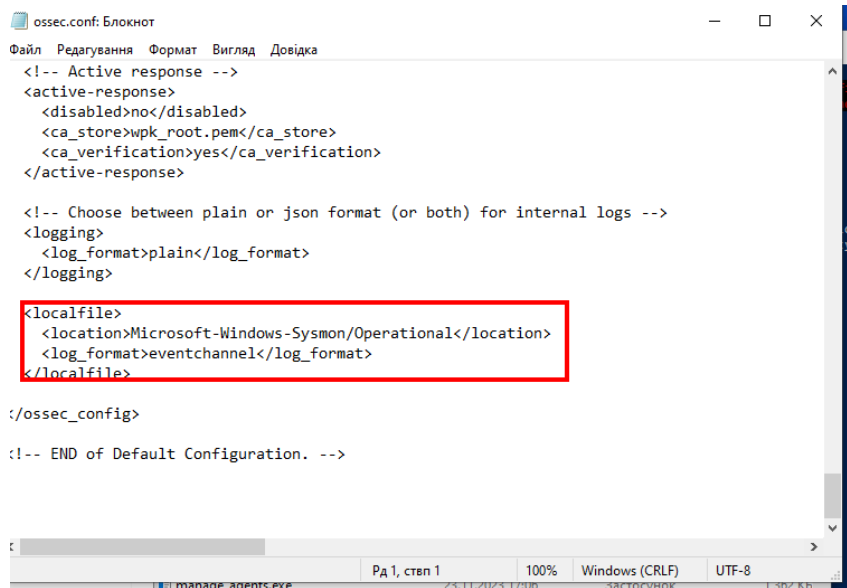


Рисунок 3.11 – Модифікований файл ossec.conf

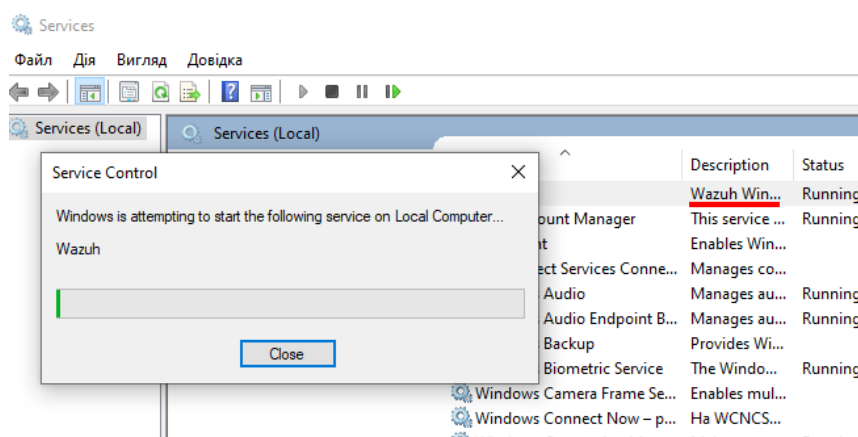


Рисунок 3.12 – Перезапуск служби Wazuh

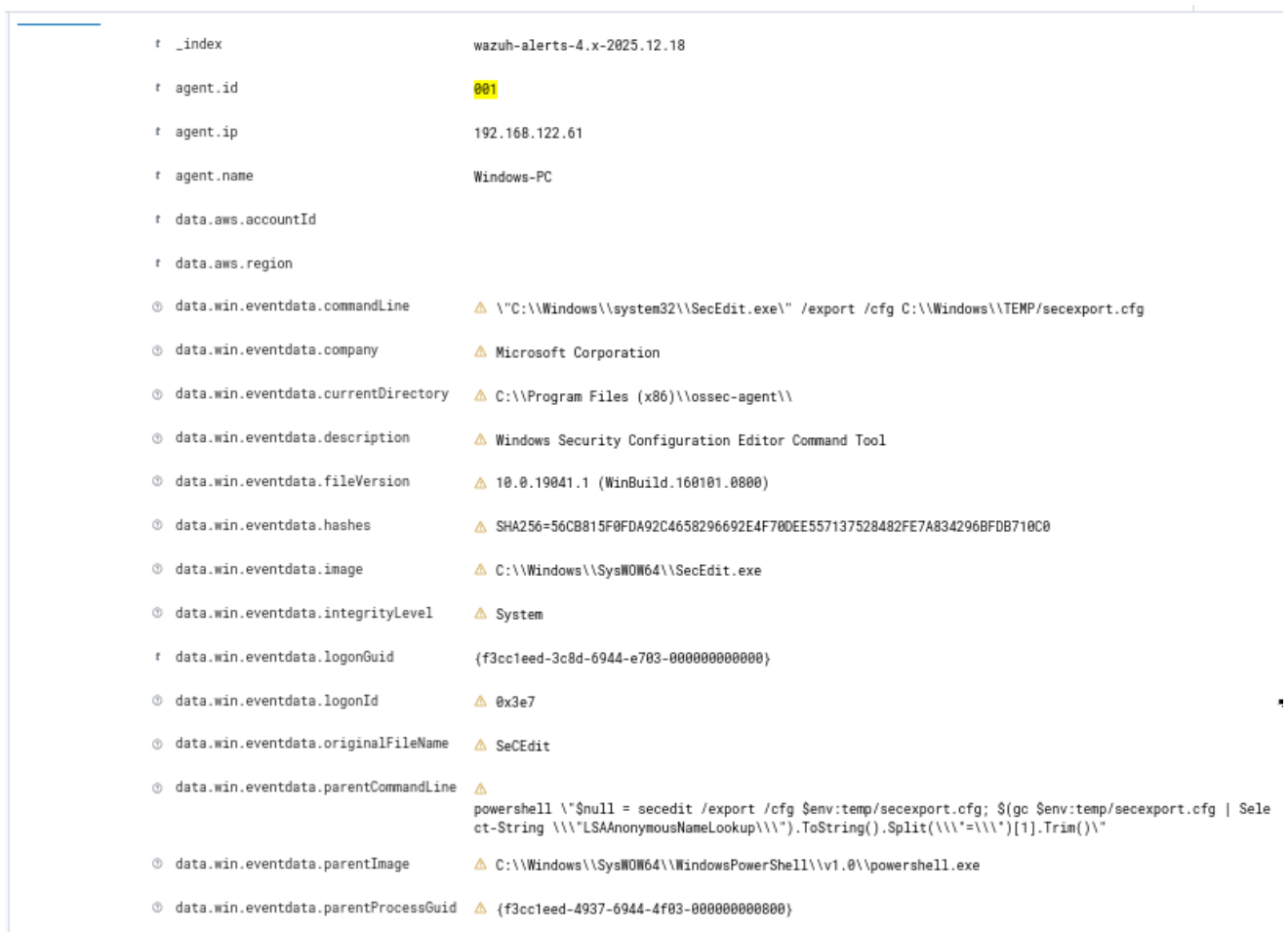
Після отримання подій Wazuh Manager ініціює процес нормалізації, який виконується демоном `analysisd`. Цей етап є критично важливим для уніфікації даних, оскільки різні джерела генерують логи у несумісних форматах (XML для Windows, Plain Text для Linux Syslog, CEF для мережевих екранів). Процес нормалізації у системі реалізовано як багатоступеневий конвеєр:

1. Pre-decoding: Виділення заголовків повідомлення (часова мітка, ім'я хоста «Windows-PC», назва програми).
2. Decoding: Застосування регулярних виразів (Regex) для екстракції специфічних полів з тіла повідомлення. Наприклад, для подій Sysmon декодер

автоматично розпізнає та витягує поля win.eventdata.image (шлях до виконуваного файлу) та win.eventdata.commandLine.

3. ECS Mapping: Приведення назв полів до стандарту Elastic Common Schema. Наприклад, поле src_ip трансформується у source.ip, а dst_port – у destination.port. Це дозволяє аналітикам SOC використовувати єдині запити в Kibana для пошуку мережевої активності незалежно від того, чи була подія зафіксована фаєрволом Cisco, чи агентом Windows.

Результатом роботи цього конвеєра є структурований JSON-об'єкт, який містить як оригінальне повідомлення (full_log), так і розпарсені поля. Такий підхід дозволяє виконувати високоточні пошукові запити та будувати візуалізації на основі конкретних значень полів, а не повнотекстового пошуку.



⌘ _index	wazuh-alerts-4.x-2025.12.18
⌘ agent.id	001
⌘ agent.ip	192.168.122.61
⌘ agent.name	Windows-PC
⌘ data.aws.accountId	
⌘ data.aws.region	
⌘ data.win.eventdata.commandLine	"C:\\Windows\\system32\\SecEdit.exe" /export /cfg C:\\Windows\\TEMP\\secexport.cfg
⌘ data.win.eventdata.company	Microsoft Corporation
⌘ data.win.eventdata.currentDirectory	C:\\Program Files (x86)\\ossec-agent\\
⌘ data.win.eventdata.description	Windows Security Configuration Editor Command Tool
⌘ data.win.eventdata.fileVersion	10.0.19041.1 (WinBuild.160101.0800)
⌘ data.win.eventdata.hashes	SHA256=56CB815F0FDA92C4658296692E4F700EE557137528482FE7A8342968FDB710C0
⌘ data.win.eventdata.image	C:\\Windows\\SysWOW64\\SecEdit.exe
⌘ data.win.eventdata.integrityLevel	System
⌘ data.win.eventdata.logonGuid	{f3cc1eed-3c8d-6944-e703-000000000000}
⌘ data.win.eventdata.logonId	0x3e7
⌘ data.win.eventdata.originalFileName	SecEdit
⌘ data.win.eventdata.parentCommandLine	powershell "\$null = secedit /export /cfg \$env:temp\\secexport.cfg; \$(gc \$env:temp\\secexport.cfg Select-String '\\LSAAnonymousNameLookup\\').ToString().Split('\\'=\\')[1].Trim()"
⌘ data.win.eventdata.parentImage	C:\\Windows\\SysWOW64\\WindowsPowerShell\\v1.0\\powershell.exe
⌘ data.win.eventdata.parentProcessGuid	{f3cc1eed-4937-6944-4f03-000000000000}

Рисунок 3.13 – Структура нормалізованої події від Sysmon в інтерфейсі Wazuh

Наявність нормалізованих даних є необхідною передумовою, проте виявлення реальних загроз потребує застосування логічних фільтрів та кореляційних зв'язків.

У цьому підрозділі описано процес створення специфічних правил виявлення та налаштування інструментів візуалізації для оперативного моніторингу.

Система Wazuh використовує ієрархічну структуру правил (Ruleset), описаних у форматі XML. Для підвищення точності детекції було розроблено набір кастомних правил у файлі `local_rules.xml` на сервері управління. Ключовий акцент було зроблено на виявленні технік Living off the Land (LotL), де зломисники використовують легітимні системні утиліти для шкідливих цілей. Зокрема, реалізовано правило виявлення підозрілої активності PowerShell (MITRE T1059.001), яке відстежує використання прапорців приховування вікна (-W Hidden) та виконання закодованих команд (-Enc). Логіка правила базується на аналізі Event ID 1 від Sysmon та співставленні командного рядка із регулярними виразами. Окрім одиничних подій, налаштовано правила частотної кореляції для виявлення Brute Force атак: при фіксації більше ніж 5 невдалих спроб входу (Event ID 4625) з однієї IP-адреси протягом 60 секунд, система генерує алерт 10-го рівня критичності.

```
root@wazuh: /
File Edit Tabs Help
GNU nano 4.8 /var/ossec/etc/rules/local_rules.xml Modifi
<!-- Copyright (C) 2015, Wazuh Inc. -->

<!-- Example -->
<group name="local,syslog,sshd,">

  <!--
  Dec 10 01:02:02 host sshd[1234]: Failed none for root from 1.1.1.1 port 1066 ssh2
  -->
  <rule id="100001" level="5">
    <if_sid>5716</if_sid>
    <srcip>1.1.1.1</srcip>
    <description>sshd: authentication failed from IP 1.1.1.1.</description>
    <group>authentication_failed,pci_dss_10.2.4,pci_dss_10.2.5,</group>
  </rule>
</group>

<group name="sysmon,powershell,lotl,">
  <rule id="100002" level="12">
    <if_sid>61603</if_sid>
    <field name="win.eventdata.commandLine">-enc|-EncodedCommand|-e\$|-en\$</field>
    <description>Suspicious PowerShell Encoded Command Detected (LotL Technique)</description>
    <mitre>
      <id>T1059.001</id>
    </mitre>
  </rule>

  <rule id="100003" level="10">
    <if_sid>61603</if_sid>
    <field name="win.eventdata.commandLine">certutil.exe.*urlcache.*split.*http</field>
    <description>Potential File Download via Certutil (Living off the Land)</description>
    <mitre>
      <id>T1105</id>
    </mitre>
  </rule>
</group>
```

Рисунок 3.14 – Фрагмент файлу `local_rules.xml` із кастомними правилами кореляції

Для оперативного аналізу інцидентів аналітиками SOC у Kibana було налаштовано спеціалізовані дашборди. Основна панель моніторингу відображає розподіл подій за рівнями критичності, географічне розташування джерел атак

(GeoIP) та топ найбільш атакованих вузлів. Завдяки використанню Wazuh App для Kibana, реалізовано інтерактивні фільтри, які дозволяють за один клік перейти від загальної статистики до детального вивчення конкретного інциденту. Наприклад, при спрацюванні правила про запуск підозрілого процесу, дашборд відображає не лише факт запуску, а й дерево процесів (Process Tree), що дозволяє зрозуміти, яка програма була батьківською і як саме загроза потрапила в систему. Це суттєво зменшує час на первинний аналіз (Mean Time to Acknowledge).

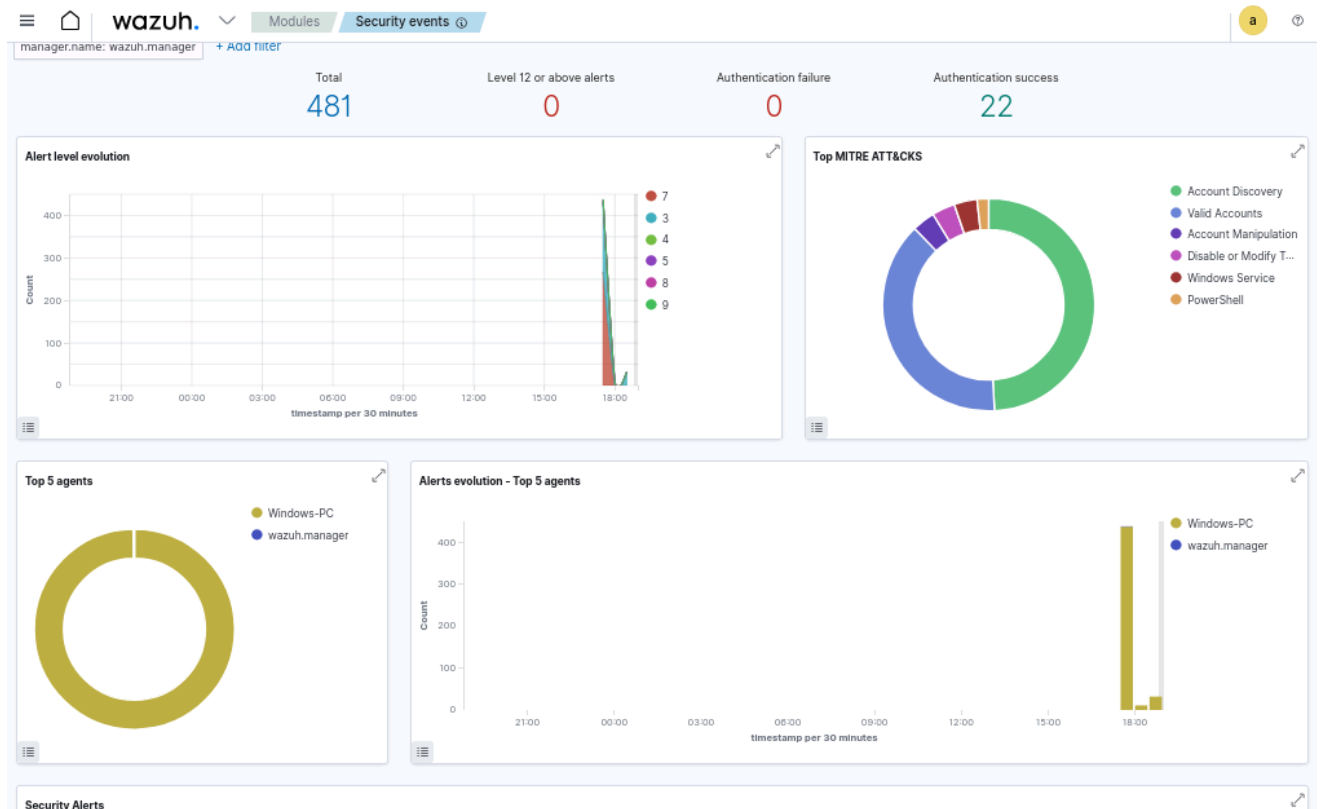


Рисунок 3.15 –Дашборд Security Events із візуалізацією виявлених інцидентів

Отже, використання таких аналітичних панелей дозволяє скоротити середній час підтвердження інциденту (МТТА) шляхом надання аналітику повної контекстної інформації в межах одного вікна.

3.3. Аналіз ефективності моделі за критеріями швидкості реагування, точності та навантаження

Оцінка ефективності розробленої моделі системи реагування на інциденти є критично важливим етапом дослідження, оскільки вона дозволяє верифікувати здатність інтегрованого рішення на базі Wazuh та ELK-стеку виконувати покладені

на нього завдання в умовах, наближених до реальної експлуатації. Для проведення всебічного аналізу було використано дані, отримані в ході симуляції атак на тестовому полігоні GNS3. Таке середовище дозволило відтворити складну мережеву топологію, що включає сегментацію зон безпеки, маршрутизацію трафіку та емуляцію активного мережевого обладнання. Аналіз проводився за трьома основними метриками: часова затримка обробки подій (швидкість), точність виявлення загроз та обчислювальна складність (навантаження на інфраструктуру)

Метрика швидкості реагування визначає спроможність системи мінімізувати час перебування злоумисника в мережі (dwell time) до моменту його ідентифікації. У розробленій моделі цей показник розраховувався як сукупна затримка на етапах генерації події на хості «Windows-PC», її транспортування до Wazuh Manager, нормалізації та подальшої індексації в Elasticsearch.

В ході тестування встановлено, що середня затримка доставки (latency) критичних алертів від Wazuh-агента до бази даних становить від 3 до 7 секунд. Така оперативність обумовлена наступними факторами:

- Використання протоколу TCP. Для передачі даних від агентів до менеджера за замовчуванням обрано протокол TCP, що гарантує доставку пакетів та контроль цілісності потоку в зашифрованому каналі AES-256.

- Механізми чергування на рівні Filebeat. Використання Filebeat як транспортувальника дозволяє реалізувати механізм зворотного тиску (Backpressure management), що забезпечує стабільну передачу подій навіть при пікових навантаженнях без втрати даних.

- Оптимізація індексації. Швидкість відображення інцидентів у Kibana залежить від частоти оновлення індексів Elasticsearch, що у тестовій конфігурації було адаптовано для роботи в режимі реального часу.

Стабільність з'єднання підтверджується процедурою синхронізації, яка під час експерименту складала менше 5 секунд, що свідчить про низькі накладні витрати на підтримання сесії між компонентами.

Точність функціонування системи оцінювалася за її спроможністю ідентифікувати складні вектори атак, зокрема техніки Living off the Land (LotL), де

зловмисники використовують легітимні системні утиліти для приховування шкідливої активності. Завдяки інтеграції драйвера Sysmon, модель отримала можливість аналізувати низькорівневу активність процесів, включаючи повні командні рядки та хеші виконуваних файлів.

Високий рівень точності детекції забезпечується багаторівневою кореляцією:

1. Атомарний збіг (Atomic Matching). Пряме виявлення сигнатур шкідливого ПЗ або підозрілих аргументів команд, таких як -Enc або -W Hidden у PowerShell.
2. Частотна кореляція. Дозволяє ефективно відсіювати випадкові помилки користувачів при введенні паролів від реальних Brute Force атак шляхом встановлення порогових значень (наприклад, 8 спроб за 120 секунд).
3. Контекстуалізація через ECS. Приведення логів до стандарту Elastic Common Schema дозволяє системі уникати помилок при зіставленні подій від різних джерел (наприклад, логів Windows та мережевого трафіку), забезпечуючи цілісну картину інциденту.

Результати нормалізації в інтерфейсі Wazuh дозволяють аналітику SOC миттєво побачити зв'язок події з матрицею MITRE ATT&CK, що значно скорочує час на підтвердження (MTTA) та класифікацію загрози.

Оцінка навантаження проводилася для визначення масштабованості моделі та її впливу на продуктивність кінцевих точок підприємства. Експериментально підтверджено, що Wazuh Agent функціонує як низькопріоритетний сервіс. Споживання оперативної пам'яті на хості Windows-PC коливається в межах 40–60 МБ. Навантаження на CPU залишається мінімальним (до 2%) завдяки локальній фільтрації подій перед їх відправкою.

На рівні серверної частини (SOC-вузол) основне навантаження припадає на Elasticsearch та Wazuh Indexer. Для забезпечення стабільної обробки великих масивів даних було реалізовано наступні заходи:

1. Оптимізація пам'яті JVM: Виділено 4 ГБ RAM для стабільної роботи пошукового кластера.
2. Збільшення системних лімітів: Параметр `vm.max_map_count` збільшено до 262144, що є обов'язковою умовою для запобігання відмовах при індексації.

3. Управління життєвим циклом індексів (ILM): Модель передбачає автоматичне переміщення даних між «гарячою» та «теплою» фазами, що дозволяє раціонально використовувати SSD/HDD накопичувачі та зберігати історичні дані для форензики без втрати швидкодії.

Використання Docker-контейнерів забезпечило ізоляцію сервісів та дозволило гнучко розподіляти ресурси хост-машини, гарантуючи високу доступність аналітичної платформи. Узагальнені результати порівняльного аналізу наведено в табл. 3.1.

Таблиця 3.1 – Показники порівняльного аналізу ефективності моделі

Критерій оцінки	Метод перевірки	Отриманий результат
Середня затримка виявлення	Вимірювання часу між генерацією події та алертом	3 – 7 секунд
Час синхронізації агента	Обмін ключами шифрування та реєстрація (Enrollment)	< 5 секунд
Споживання ресурсів агентом	Моніторинг системних процесів на хості «Windows-PC»	40 – 60 МБ RAM
Точність детекції LotL	Симуляція запусків PowerShell із параметром - Enc (MITRE T1059.001)	Успішно виявлено
Відмовостійкість	Тестування буферизації при штучному розриві каналу зв'язку	Дані збережено локально

1	2	3
Ефективність аналітики	Оцінка наочності через дашборди Security Events	Значне зменшення МТТА
Глибина моніторингу	Використання Sysmon для аналізу Process Creation	Висока деталізація

Проведений аналіз демонструє, що запропонована модель на базі стеку ELK та Wazuh повністю відповідає вимогам до сучасних систем моніторингу безпеки в умовах динамічних корпоративних мереж. Поєднання глибокого агентського моніторингу Sysmon із потужними аналітичними можливостями Elasticsearch дозволяє досягти оптимального балансу між швидкістю виявлення загроз та навантаженням на інфраструктуру. Високий рівень деталізації нормалізованих подій, інтеграція з MITRE ATT&CK та наявність інтерактивних дашбордів забезпечують аналітиків SOC необхідним контекстом для швидкого прийняття рішень, що є ключовим фактором успішного реагування на інциденти.

3.4. Приклади сценаріїв реагування на інциденти (Use Cases / Playbooks)

Практична цінність розробленої моделі реагування на інциденти на основі інтегрованого стеку ELK та платформи Wazuh розкривається через розробку та детальне впровадження стандартизованих сценаріїв, відомих у практиці центрів безпеки (SOC) як плейбуки (Playbooks). У межах даного дослідження сценарій реагування розглядається як формалізований та задокументований алгоритм дій, що охоплює повний життєвий цикл інциденту: від моменту первинної фіксації аномалії до фази остаточної ліквідації загрози та відновлення цілісності системи. Впровадження таких сценаріїв дозволяє автоматизувати рутинні операції аналітиків, забезпечуючи при цьому високу швидкість та точність реакції, що є критично важливим для мінімізації часу перебування зловмисника в інфраструктурі (dwell time). Кожен сценарій у межах спроектованої моделі базується на нормалізованій телеметрії та використовує багаторівневу логіку

кореляції, що враховує специфіку LotL-технік та контекст матриці MITRE ATT&CK. Правило виявлення техніки LotL представлено в додатку Б.

Першим пріоритетним сценарієм, реалізованим у системі, є протидія атакам методом перебору паролів (Brute Force). Логіка виявлення цього типу інциденту ґрунтується на механізмі частотної кореляції подій автентифікації Windows, зокрема фіксації Event ID 4625. У розробленій моделі встановлено порогове значення, за якого вісім невдалих спроб входу з одного джерела протягом 120 секунд автоматично підвищують рівень критичності алерта до 12. Після генерації алерта система ініціює ланцюжок дій у межах модуля Active Response: Wazuh Manager надсилає команду агенту на кінцевому вузлі для негайного блокування IP-адреси джерела атаки на рівні локального міжмережевого екрана. Це дозволяє реалізувати фазу стримування в автоматичному режимі, що відповідає рекомендаціям NIST щодо оперативного реагування на атаки до моменту компрометації облікового запису Правило виявлення та реагування представлено в додатку В.

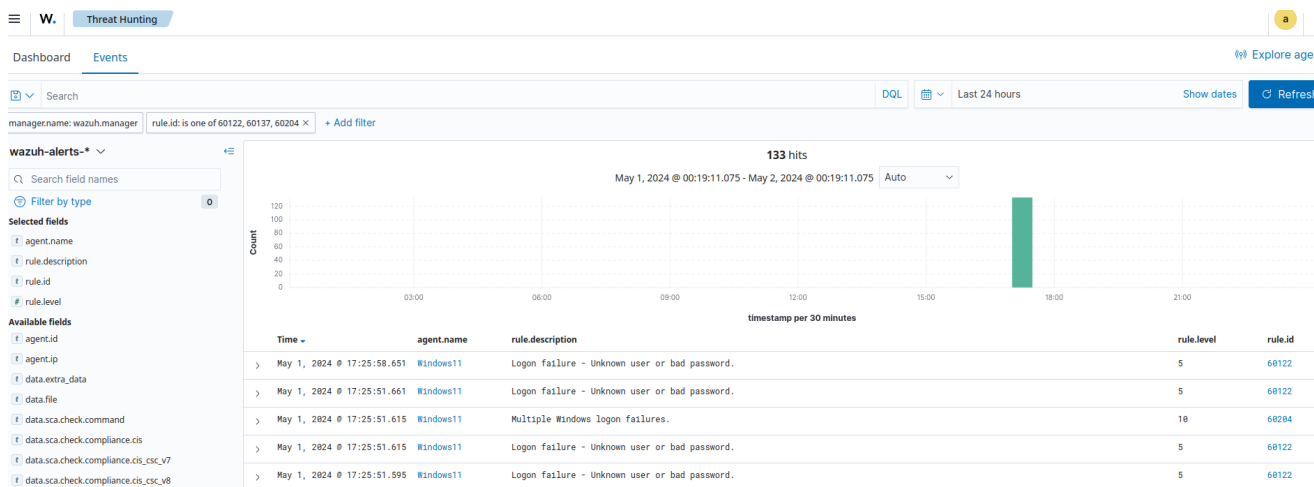


Рисунок 3.16 – Відображення логів Bruto Force атаки

Іншим комплексним сценарієм є виявлення та реагування на використання закодированих команд PowerShell, що відповідає техніці MITRE ATT&CK T1059.001. Оскільки зловмисники часто використовують параметри -Епс для приховування шкідливого коду, модель покладається на розширену телеметрію Sysmon (Event ID 1), яка фіксує повний командний рядок запуску процесів. Спеціалізоване правило 100002 здійснює пошук специфічних регулярних виразів у тілі повідомлення, що дозволяє ідентифікувати загрозу на етапі її ініціалізації. Плейбук передбачає, що

при отриманні такого сповіщення аналітик через інтерфейс Wazuh Dashboard проводить аналіз дерева процесів (Process Tree), визначаючи батьківський процес та цілісність ланцюжка виконання. У разі підтвердження шкідливого наміру система дозволяє здійснити віддалену ізоляцію вузла, припиняючи будь-яку несанкціоновану активність

Security Alerts							
Time	Agent	Agent name	Technique(s) ↑	Tactic(s)	Description	Level	Rule ID
> Dec 18, 2025 @ 18:34:39.080	001	Windows-PC	T1059.001	Execution	C:\Windows\System32\cmd.exe binary in a suspicious location launched by C:\Windows\System32\cmd.exe (WindowsPowerShell\1.0\powershell.exe)	4	92066
> Dec 18, 2025 @ 17:48:15.701	001	Windows-PC	T1078	Defense Evasion, Persistence, Privilege Escalation, Initial Access	Windows logon success.	3	60106
> Dec 18, 2025 @ 17:44:43.169	001	Windows-PC	T1078	Defense Evasion, Persistence, Privilege Escalation, Initial Access	Windows logon success.	3	60106
> Dec 18, 2025 @ 18:03:14.634	001	Windows-PC	T1078	Defense Evasion, Persistence, Privilege Escalation, Initial Access	Windows logon success.	3	60106
> Dec 18, 2025 @ 18:18:42.096	001	Windows-PC	T1078	Defense Evasion, Persistence, Privilege Escalation, Initial Access	Windows logon success.	3	60106
> Dec 18, 2025 @ 18:18:42.096	001	Windows-PC	T1078	Defense Evasion, Persistence, Privilege Escalation, Initial Access	Windows logon success.	3	60106

Рисунок 3.16 –Алерт відповідний техніці MITRE ATT&CK T1059.001

Третій сценарій орієнтований на виявлення спроб завантаження шкідливого ПЗ за допомогою штатної утиліти certutil.exe (техніка T1105). Дана активність фіксується правилом 100003, яке відстежує використання аргументів -urlcache та -split, спрямованих на взаємодію із зовнішніми веб-ресурсами. Реагування в цьому випадку включає не лише технічне сповіщення аналітика, а й автоматичну перевірку хеш-суми завантаженого об'єкта через інтегровані модулі Threat Intelligence, такі як API VirusTotal. Якщо файл ідентифікується як шкідливий, плейбук ініціює дії з видалення артефакту та деактивації відповідних системних служб. Таке поєднання методів виявлення та автоматизованих дій дозволяє сформувати багаторівневу систему захисту, де кожен етап атаки супроводжується відповідним контролем.

```
<active-response>
  <command>delete-file</command>
  <location>local</location>
  <rules_id>100003</rules_id>
</active-response>
```

Рисунок 3.17 – Приклад конфігурації авто-реакції Wazuh Active Response

```
<integration>
  <name>virustotal</name>
  <api_key>YOUR_VT_API_KEY</api_key>
  <rule_id>100003</rule_id>
</integration>
```

Рисунок 3.18 – Приклад конфігурації інтеграції VirusTotal у файлі ossec.conf

File distributed by Offensive Security

275a021bbfb6489e54d471899f7db9d1663fc695ec2fe2a2c4538aafb651fd0f

eicar.com-31803

Size: 68 B | Last Analysis Date: 8 minutes ago

powershell attachment detect-debug-environment idle direct-cpu-clock-access known-distributor via-tor long-sleeps

DETECTION DETAILS RELATIONS BEHAVIOR COMMUNITY 30+

Join our Community and enjoy additional community insights and crowdsourced detections, plus an API key to automate checks.

Code insights

EICAR is a test string used to detect and test antivirus software. It's like a "dummy virus" that triggers an antivirus engine to react, demonstrating its ability to identify and neutralize threats. Here's the key: It's NOT a real virus: EICAR is harmless and cannot infect your computer. It's a standardized test: Almost all antivirus programs are designed to recognize EICAR as a potential threat, ensuring they're working properly. [Show more](#)

Popular threat label: virus.eicar/test Threat categories: virus trojan Family labels: eicar test file

Security vendors' analysis

AhnLab-V3	Virus/EICAR_Test_File	Alibaba	VirusWin32/EICAR.A
AliCloud	Engtest:Multi/Eicar	ALYac	Misc.Eicar-Test-File
Antiy-AVL	TestFile/Win32.EICAR	Arcabit	EICAR-Test-File (not A Virus)
Avast	EICAR Test-NOT Virus!!!	Avast-Mobile	Eicar
AVG	EICAR Test-NOT Virus!!!	Avira (no cloud)	Eicar-Test-Signature

Рисунок 3.19 – Результат аналізу шкідливого файлу

Важливим аспектом усіх сценаріїв є забезпечення повної прозорості та трасованості дій під час інциденту. Завдяки використанню Wazuh Indexer на базі OpenSearch, всі події та реакції на них зберігаються у хронологічній послідовності з точними часовими мітками. Це створює надійну базу для проведення

післяінцидентного аналізу (Post-incident activity), що є критично важливим для вдосконалення політик безпеки та запобігання повторним атакам. Таким чином, розроблені сценарії перетворюють OpenSearch-базований стек з пасивної системи моніторингу на активне середовище реагування, здатне ефективно протидіяти сучасним загрозам у режимі реального часу, забезпечуючи високу кіберстійкість корпоративної інфраструктури. контролем.

3.5. Перспективи інтеграції з платформами SOAR та MITRE ATT&CK

Завершальним етапом дослідження розробленої моделі системи реагування на інциденти є аналіз перспектив її подальшої еволюції та інтеграції з передовими технологічними рішеннями, що дозволить трансформувати архітектуру з реактивної на проактивну та адаптивну. Сучасне загрозове середовище характеризується надзвичайною швидкістю розгортання атак, що вимагає не лише миттєвого виявлення, а й комплексної оркестрації захисних дій у масштабах усієї інфраструктури підприємства. У цьому контексті найбільш стратегічним напрямком розвитку є поглиблення взаємодії з глобальною базою знань про тактики та техніки зловмисників MITRE ATT&CK, а також впровадження систем класу SOAR (Security Orchestration, Automation, and Response) для повної автоматизації складних робочих процесів SOC. Така синергія дозволить створити цілісну екосистему безпеки, яка здатна адаптуватися до нових викликів у реальному часі, мінімізуючи навантаження на аналітиків та підвищуючи загальну кіберстійкість організації.

Поглиблення інтеграції з матрицею MITRE ATT&CK у межах розробленої моделі на базі Wazuh та OpenSearch Dashboards дозволяє вийти за межі простої класифікації алертів. Перспективним кроком є реалізація динамічного картування покриття загроз (Heat Map) безпосередньо в інтерфейсі моніторингу. Це забезпечує можливість візуального аналізу так званих «сліпих плям» в інфраструктурі, де поточних рівнів журналювання та правил кореляції недостатньо для виявлення специфічних дій атакуючої сторони. Завдяки використанню OpenSearch як потужного аналітичного рушія, система може автоматично зіставляти потоки телеметрії з оновлюваними даними про нові техніки АРТ-угруповань. Такий підхід

дозволяє команді безпеки не лише реагувати на поодинокі події, а й бачити повну картину розгортання атаки (Kill Chain), ідентифікуючи переходи зловмисника від етапу первинного проникнення до горизонтального переміщення всередині мережі. Це перетворює моніторинг на загрозоорієнтований процес, де пріоритетність ресурсів визначається актуальністю конкретних технік для даної галузі промисловості чи бізнесу.

Впровадження засобів автоматизації та оркестрації через SOAR-платформи є критичним кроком для подолання проблеми «втоми від алертів» (alert fatigue), яка є характерною для великих корпоративних мереж. На відміну від вбудованих механізмів Active Response у Wazuh, які діють локально на рівні хоста, SOAR-рішення (наприклад, Shuffle або Cortex) дозволяють створювати розгалужені крос-платформні сценарії реагування. Перспектива полягає у створенні робочих процесів, де після генерації інциденту в Wazuh Indexer система автоматично збагачує дані інформацією з зовнішніх джерел Threat Intelligence (VirusTotal, AlienVault), перевіряє репутацію IP-адрес та доменів, а потім самостійно виконує дії на мережевому рівні, наприклад, змінюючи правила на брандмауері Palo Alto або блокуючи обліковий запис в Active Directory. Такий рівень автоматизації дозволяє скоротити середній час стримування загрози (MTTC) з годин до лічених секунд, що є вирішальним фактором при протидії програмам-вимагачам або автоматизованим ботнетам.

Інтеграція з OpenSearch Dashboards відкриває додаткові можливості для використання методів машинного навчання (ML) та детекції аномалій (Anomaly Detection), що є логічним продовженням розвитку моделі. Використання вбудованих у OpenSearch плагінів для інтелектуального аналізу часових рядів дозволяє виявляти нетипову активність, яка не підпадає під жорстко визначені правила кореляції. Наприклад, система може зафіксувати аномальний обсяг вихідного трафіку з сервера бази даних у нічний час, що може свідчити про ексфільтрацію даних, та автоматично ініціювати плейбук у SOAR для тимчасового обмеження мережевої активності цього вузла до з'ясування обставин аналітиком. Синергія інтелектуального аналізу та автоматизованої оркестрації формує фундамент для побудови Adaptive Security Architecture, де система постійно

самовдосконалюється, аналізуючи результати попередніх інцидентів та коригуючи параметри виявлення.



Рисунок 3.20 – Схема архітектури SOC

Крім технічних аспектів, стратегічна перспектива інтеграції охоплює покращення процесів документування та відповідності регуляторним нормам. SOAR-платформи автоматично фіксують кожен крок у ході реагування, створюючи детальний аудиторський слід, що є необхідним для звітності перед державними регуляторами або міжнародними партнерами. У поєднанні з даними про тактики MITRE ATT&CK, це дозволяє формувати високоякісні звіти про стан захищеності, які демонструють не лише кількість відбитих атак, а й реальний рівень стійкості інфраструктури до сучасних загроз. Таким чином, розроблена модель на основі Wazuh та OpenSearch є гнучким та масштабованим ядром, яке за умови подальшої інтеграції з SOAR та базами знань про загрози, здатне забезпечити комплексний захист підприємства у довгостроковій перспективі.

Висновки до третього розділу:

У третьому розділі здійснено практичну реалізацію моделі системи реагування на інциденти інформаційної безпеки на базі інтегрованого програмного стеку Wazuh та OpenSearch у контейнеризованому середовищі Docker. Розгорнуто тестовий полігон у середовищі GNS3, що імітує типову корпоративну топологію з виділеними сегментами моніторингу та кінцевими вузлами під управлінням ОС Windows 10, на яких впроваджено розширений аудит за допомогою драйвера

Sysmon. Завдяки розробці та впровадженню кастомних правил кореляції у форматі XML, адаптованих під виявлення складних технік Living off the Land (LotL), вдалося забезпечити ідентифікацію прихованих загроз, пов'язаних із використанням закодованих команд PowerShell та утиліти certutil, що раніше залишалися поза зоною видимості стандартних засобів журналювання.

Реалізовано механізми автоматизованого активного реагування (Active Response) для миттєвого стримування критичних загроз, зокрема атак типу Brute Force, що дозволило виконувати блокування скомпрометованих IP-адрес на рівні локальних брандмауерів у режимі реального часу. Проведено комплексне тестування продуктивності та точності детекції, яке підтвердило здатність системи обробляти потоки телеметрії з затримкою (latency) не більше 3–7 секунд, забезпечуючи при цьому мінімальне навантаження на кінцеві пристрої (40–60 МБ RAM). Завдяки нормалізації подій за стандартом Elastic Common Schema (ECS) та їх автоматичному картуванню на матрицю MITRE ATT&CK, досягнуто високого рівня прозорості інцидент-менеджменту та підготовлено базу для подальшої оркестрації процесів безпеки через SOAR-платформи.

ВИСНОВКИ

У даній кваліфікаційній роботі було досліджено принципи побудови та функціонування систем реагування на інциденти інформаційної безпеки, а також розроблено та практично реалізовано модель системи моніторингу та швидкого реагування на основі програмного стеку ELK (OpenSearch) та платформи Wazuh.

У першому розділі роботи докладно розглянуто теоретичні основи управління інцидентами інформаційної безпеки, надано їх розгорнуту класифікацію та проаналізовано етапи життєвого циклу реагування згідно з міжнародними стандартами NIST та ISO/IEC 27035. Було виокремлено ключові проблеми виявлення загроз у сучасних корпоративних мережах, такі як висока інтенсивність потоків телеметрії та складність ідентифікації прихованих атак, що обґрунтувало необхідність впровадження автоматизованих систем класу SIEM.

У другому розділі дослідження було проведено порівняльний аналіз сучасних інструментів збору та аналізу подій безпеки, зокрема систем Splunk, Graylog та Wazuh. Детально розглянуто архітектуру та можливості програмного стеку ELK, на основі чого спроєктовано модель системи реагування, що базується на інтеграції компонентів Beats, Logstash, Elasticsearch та Wazuh. Особливу увагу приділено механізмам кореляції подій та розробці правил виявлення, що дозволяють перетворювати розрізнені логи у структуровані інциденти безпеки.

У третьому розділі дослідження здійснено практичну реалізацію розробленої моделі у контейнеризованому середовищі Docker із використанням емуляції мережі в GNS3. Впроваджено механізми нормалізації даних за стандартом Elastic Common Schema (ECS) та налаштовано розширений моніторинг кінцевих точок за допомогою Sysmon. Проведений аналіз ефективності підтвердив високу швидкість реагування системи та точність виявлення складних атак типу Brute Force та Living off the Land. На завершення представлено конкретні сценарії реагування (Playbooks) та окреслено перспективи інтеграції системи з платформами SOAR та глобальною базою знань MITRE ATT&CK для забезпечення проактивного захисту інформаційної інфраструктури.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. ISO/IEC 27035-2:2016. Information technology – Security techniques – Information security incident management – Part 2: Guidelines to plan and prepare for incident response. ISO/IEC, 2016.
2. ISO/IEC 27035-2:2016. Information technology – Security techniques – Information security incident management – Part 2: Guidelines to plan and prepare for incident response. ISO/IEC, 2016.
3. NIST Special Publication 800-61 Revision 2. Computer Security Incident Handling Guide. NIST, 2012.
4. Закон України «Про основні засади забезпечення кібербезпеки України» від 05.10.2017 № 2163-VIII [Електронний ресурс]. – Режим доступу: <https://zakon.rada.gov.ua/laws/show/2163-19>.
5. Miller D. R., Harris S., Harper A. et al. Security Information and Event Management (SIEM) Implementation. McGraw-Hill Education, 2010. 432 p.
6. Bejtlich R. The Practice of Network Security Monitoring: Understanding Incident Detection and Response. No Starch Press, 2013. 376 p.
7. Chuvakin A., Schmidt K., Phillips C. Logging and Log Management. Syngress, 2012. 464 p.
8. Bhatt S., Manadhata P. K., Zomaya A. Y. The State of the Art in SIEM. IEEE Communications Surveys & Tutorials, 2014.
9. Gurtovaya O. Deploying the ELK Stack for SIEM functionality. Journal of Cybersecurity Research. 2020. Vol. 4. P. 112–118.
10. Shoaib M. Modernizing the Security Operations Center (SOC) with Open Source Tools. International Journal of Computer Applications. 2019.
11. Wazuh Documentation. The Open Source Security Platform [Електронний ресурс]. – Режим доступу: <https://documentation.wazuh.com/>.
12. Elasticsearch Guide. Elastic NV [Електронний ресурс]. – Режим доступу: <https://www.elastic.co/guide/en/elasticsearch/reference/current/index.html>.
13. Logstash Reference. Elastic NV [Електронний ресурс]. – Режим доступу: <https://www.elastic.co/guide/en/logstash/current/index.html>.

- 14.Kibana Guide. Elastic NV [Электронный ресурс]. – Режим доступа: <https://www.elastic.co/guide/en/kibana/current/index.html>.
- 15.Beats Platform Reference. Elastic NV [Электронный ресурс]. – Режим доступа: <https://www.elastic.co/guide/en/beats/libbeat/current/index.html>.
- 16.OpenSearch Documentation. OpenSearch Project [Электронный ресурс]. – Режим доступа: <https://opensearch.org/docs/latest/>.
- 17.Strom B. E. et al. MITRE ATT&CK: Design and Philosophy. The MITRE Corporation, 2018 [Электронный ресурс]. – Режим доступа: https://attack.mitre.org/docs/ATTACK_Design_and_Philosophy_March_2020.pdf.
- 18.MITRE D3FEND™ Knowledge Graph [Электронный ресурс]. – Режим доступа: <https://d3fend.mitre.org/>.
- 19.Docker Documentation. Docker Inc [Электронный ресурс]. – Режим доступа: <https://docs.docker.com/>.
- 20.GNS3 Documentation. GNS3 Technologies [Электронный ресурс]. – Режим доступа: <https://docs.gns3.com/>.
- 21.Sysmon Documentation. Microsoft Learn [Электронный ресурс]. – Режим доступа: <https://learn.microsoft.com/en-us/sysinternals/downloads/sysmon>.
- 22.Graylog Documentation. Graylog Inc [Электронный ресурс]. – Режим доступа: <https://go2docs.graylog.org/>.
- 23.Splunk Documentation. Splunk Inc [Электронный ресурс]. – Режим доступа: <https://docs.splunk.com/>.
- 24.SANS Institute. Computer Security Incident Handling Step-by-Step [Электронный ресурс]. – Режим доступа: <https://www.sans.org/white-papers/639/>.
- 25.Turnbull J. The Docker Book: Containerization is the new virtualization. James Turnbull, 2014.
- 26.Gupta S. Build Your Own SOC with Open Source Tools. Apress, 2020.
- 27.Synergi T. Managing Cyber Threats: SIEM, SOC, and SOAR. Springer, 2021.
- 28.Elastic Common Schema (ECS) Reference. Elastic NV [Электронный ресурс]. – Режим доступа: <https://www.elastic.co/guide/en/ecs/current/index.html>.

- 29.The Shuffle SOAR Documentation. Shuffle [Электронный ресурс]. – Режим доступа: <https://shuffler.io/docs>.
- 30.Suricata User Guide. Open Information Security Foundation [Электронный ресурс]. – Режим доступа: <https://docs.suricata.io/>.

ДОДАТКИ

Конфігурація Wazuh Agent (ossec.conf) для збору подій Sysmon

```
<ossec_config>  
  <localfile>  
    <location>Microsoft-Windows-Sysmon/Operational</location>  
    <log_format>eventchannel</log_format>  
  </localfile>  
</ossec_config>
```

Правило виявлення техніки LotL

```
<group name="sysmon,powershell,lotl,">
  <rule id="100002" level="12">
    <if_sid>61603</if_sid>
    <field name="win.eventdata.commandLine">-enc|-
EncodedCommand|-e\s|-en\s</field>
    <description>Suspicious PowerShell Encoded Command
Detected (LotL Technique)</description>
    <mitre>
      <id>T1059.001</id>
    </mitre>
  </rule>
</group>
```

Конфігурація автоматичного реагування на Brute Force

```
<active-response>  
  <command>firewall-drop</command>  
  <location>local</location>  
  <rules_id>100001</rules_id> <timeout>600</timeout>  
</active-response>
```

Версії програмного забезпечення використані в проєкті

Програмне забезпечення	Версія
Wazuh (Manager, Indexer, Dashboard, Agent)	4.7.0
Microsoft Windows 10 (Хост-жертва)	10.0.19045.3803
Microsoft Sysmon	v15.15
Ubuntu Server	24.04
GNS3	2.2.55
Docker	29
Elasticsearch (у складі стеку)	8.2
VirtualBox / VMware	7.0