Вінницький національний технічний університет
Факультет інформаційних технологій та комп'ютерної інженерії
Кафедра захисту інформації

МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА
на тему:
«СИСТЕМА УПРАВЛІННЯ БЕЗПЕКОЮ ТА ПОДІЯМИ. ЧАСТИНА 1. МЕТОД ТА ПРОГРАМНИЙ ЗАСІБ ДЛЯ УПРАВЛІННЯ БЕЗПЕКОЮ ТА ПОДІЯМИ»

Виконав: студент 2 курсу, групи БС-21м
спеціальності 125 «Кібербезпека»
_____ В. М. Буняк

Керівник: д. т. н, проф., зав. каф. ЗІ
_____ В. А. Лужецький
«19» ____12____ 2022 р.

Опонент: к. т. н., доц., доц. каф. ПЗ
_____ В. П. Майданюк
«19» ____12____ 2022 р.

Допущено до захисту
Завідувач кафедри ЗІ
д.т.н., проф.
_____ В. А. Лужецький
«19» ____12____ 2022 р.

Вінниця ВНТУ – 2022 року

Вінницький національний технічний університет
Факультет інформаційних технологій та комп'ютерної інженерії
Кафедра захисту інформації
Рівень вищої освіти II (магістерський)
Галузь знань – 12 «Інформаційні технології»
Спеціальність – 125 «Кібербезпека»
Освітньо-професійна програма – «Безпека інформаційних і комунікаційних систем»

ЗАТВЕРДЖУЮ
Завідувач кафедри ЗІ,
д.т.н., проф.
_____ В. А. Лужецький
«15» 09 2022 року

З А В Д А Н Н Я
НА МАГІСТЕРСЬКУ КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ

Буняку Віталію Михайловичу

1. Тема роботи: «Система управління безпекою та подіями. Частина 1. Метод та програмний засіб для управління безпекою та подіями»
керівник роботи: Лужецький Володимир Андрійович, д.т.н., проф., зав. каф. ЗІ,
затверджені наказом ректора ВНТУ від 14 вересня 2022 року №203.

2. Строк подання студентом роботи: 19 грудня 2022 р.

3. Вихідні дані до роботи:
– дослідження методів роботи засобів для управління безпекою та подіями;
– аналіз даних, що надходять у програмний засіб;
– програмна реалізація методів засобу для управління безпекою та подіями;
– тестування розробленого програмного засобу.

4. Зміст текстової частини: Вступ. 1. Аналіз літературних джерел. 2. Розробка методу управління безпекою та подіями. 3. Розробка програмного засобу для управління безпекою та подіями. 4. Економічна частина. Висновки. Список використаних джерел. Додатки.

5. Перелік ілюстративного матеріалу: Система управління безпекою та подіями. Частина 1. Метод та програмний засіб для управління безпекою та подіями. Базові компоненти SIEM (плакат А4). Метод управління безпекою та подіями (плакат А4). Алгоритм роботи засобу управління безпекою та подіями (плакат А4). Архітектура програмного засобу (плакат А4). Результати тестування програмного засобу та його складових (плакат А4). Результати порівняння SIEM (плакат А4)

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийня
1	Лужецький В. А., д.т.н., проф., зав. каф. ЗІ		
2	Лужецький В. А., д.т.н., проф., зав. каф. ЗІ		
3	Лужецький В. А., д.т.н., проф., зав. каф. ЗІ		
4	Лесько О. Й., к.е.н., доц., проф., каф. ЕПВМ		

7. Дата видачі завдання 1 вересня 2022 року.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Зміст етапу	Строк виконання етапів роботи	Примітка
1	Аналіз завдання. Вступ	01.09.2022 – 04.09.2022	
2	Аналіз інформаційних джерел за напрямком магістерської кваліфікаційної роботи	05.09.2022 – 15.09.2022	
3	Науково-технічне обґрунтування	16.09.2022 – 22.09.2022	
4	Розробка технічного завдання	23.09.2022 – 04.10.2022	
5	Аналіз програмних засобів для управління безпекою та подіями	05.10.2022 – 08.10.2022	
6	Аналіз та формування вимог до програмного засобу	09.10.2022 – 16.10.2022	
7	Розробка програмного засобу для управління безпекою та подіями	17.10.2022 – 14.11.2022	
8	Тестування програмного засобу для управління безпекою та подіями	15.11.2022 – 17.11.2022	
9	Розробка розділу економічного обґрунтування доцільності розробки	18.11.2022 – 21.11.2022	
10	Аналіз виконання ТЗ, висновки	22.11.2022 – 24.11.2022	
11	Оформлення пояснювальної записки	25.11.2022 – 29.11.2022	
12	Перевірка магістерської роботи на наявність плагіату	30.11.2022 – 02.12.2022	
13	Попередній захист та доопрацювання МКР	07.12.2022 – 14.12.2022	
14	Представлення МКР до захисту, рецензування	15.12.2022 – 19.12.2022	
16	Захист МКР	22.12.2022 – 26.12.2022	

Студент _____ В. М. Буня

Керівник роботи _____ В. А. Лужецьки

АНОТАЦІЯ

УДК 681.325.5

Буняк В. Система управління безпекою та подіями. Частина 1. Метод та програмний засіб для управління безпекою та подіями. Магістерська кваліфікаційна робота зі спеціальності 125 «Кібербезпека», освітня програма «Безпека інформаційних і комунікаційних систем». Вінниця: ВНТУ, 2022. 79 с.

На укр. мові. Бібліогр.: 45 назв; рис.: 16; табл.: 17.

Магістерська кваліфікаційна робота присвячена розробці методу та програмного засобу для управління безпекою та подіями. Підготовлено науково-дослідне та техніко-економічне обґрунтування доцільності досліджень. У роботі здійснено аналіз існуючих методів управління безпекою та подіями і обґрунтовано вибір для реалізації методу управління безпекою та подіями. Розроблено метод управління безпекою та подіями. Розроблено програмний засіб, який дозволив протестувати запропонований метод.

Ілюстративна частина складається з 7 плакатів з демонстрацією результатів моделювання і проведених досліджень.

В економічному розділі оцінено витрати на розробку.

Ключові слова: інформаційна безпека, SIEM, додаткові модулі.

ABSTRACT

Buniak V. Security and event management system. Part 1. Method and software for security and event management. Master's thesis in the specialty 125 – cybersecurity, educational program - Security of information and communication systems. Vinnytsia: VNTU, 2022. 79 p.

In Ukrainian. Bibliographer: 45 titles; fig.: 16; tabl.: 17.

The master's thesis is devoted to the development of a method and a software tool for security and event management. A scientific-research and technical-economic justification of the feasibility of research has been prepared. The work analyzes the existing security and event management methods and substantiates the choice for the implementation of the security and event management method. A security and event management method has been developed. A software tool was developed that allowed testing the proposed method.

The illustrative part consists of 7 posters with a demonstration of the results of development and conducted research.

In the economic section, an assessment of costs for the development of information technology was made.

Keywords: information security, SIEM, additional modules.

ЗМІСТ

ВСТУП

Кількість кібератак, як і їх масштаби у світі, збільшується з року в рік. У 2022 році, за даними відкритих джерел, вони зросли на 20%, а загальна кількість заражених шкідливим ПЗ пристроїв — майже на 40%. Найчастіше хакери обирають своїми жертвами державні установи, промислові компанії, медицину, освіту та фінансові організації. Головною метою є викрадення персональних, облікових та фінансових даних. З другого краю місці — шифрування даних із подальшим здирництвом.

Від тактики «зламати все, що ламається» зловмисники переходять до цілеспрямованих атак на конкретні бізнеси. Таких випадків минулого року було 60% усіх зафіксованих атак. Цільові атаки можуть здійснюватися скоординовано кількома групами хакерів, після чого ділять виручку.

Зростає популярність APT-атак (англ. Advanced Persistent Threat, тобто «складна постійна загроза» коли зловмисники впроваджуються в системи компанії, довго не дають про себе знати, нарощують свої можливості, а коли їхня діяльність стає помітна, захищатися вже пізно).

Зі збільшенням кількості загроз і атак на операційні системи та на веб-сервери, багато організацій страждають через витік конфіденційної інформації Тому, необхідно мати засоби кібербезпеки і впроваджувати їх у свою систему, щоб можна було зменшити кількість загроз та забезпечити конфіденційність.

Якщо підсумувати – при все глобальнішому оцифровуванні послуг росте і вразливість їх несанкціонованого зламу зі сторони зацікавлених осіб. А рішення SIEM перетворилися на комплексні системи, які забезпечують широку видимість для виявлення областей високого ризику та зосереджені на стратегіях пом’якшення, спрямованих на зменшення витрат і часу реагування на інциденти. Системи безпеки та управління подіями (SIEM) широко розгортаються, як потужний інструмент для запобігання, виявлення та реагування на кібератаки.

Зараз системи SIEM і пов’язані з ними рішення повільно зближуються з інструментами аналізу великих даних. Незважаючи на те, що теперішні SIEM надають потужні функції з точки зору виявлення зв’язків між подіями,

зберігання, візуалізації та продуктивності, а також здатність автоматизувати процес реакції шляхом вибору та розгортання контрзаходів, поточні системи реагування дуже обмежені та контрзаходи вибираються та розгортаються без виконання комплексного аналізу впливу атак і сценаріїв реагування.

Об’єктом дослідження є процеси управління безпекою та подіями.

Предмет дослідження – метод та засіб для управління безпекою та подіями.

Метою магістерської кваліфікаційної роботи є розширення функціональних можливостей систем управління безпекою та подіями, спрямованих на підвищення безпеки веб-серверів, шляхом розробки методу та програмного засобу для управління безпекою та подіями.

Для досягнення мети необхідно розв’язати такі задачі:

– проаналізувати сучасні системи управління безпекою та подіями;

– проаналізувати недоліки сучасних SIEM;

– розробити метод та програмний засіб для управління безпекою та подіями;

– проаналізувати виконані розробки з метою виявлення можливих помилок та їх усунення;

– визначити економічну доцільність від розробки засобу для управління безпекою та подіями.

Наукова новизна полягає в тому, що:

– вперше запропоновано метод для управління безпекою та подіями, що забезпечує розширення функціональних можливостей системи і як наслідок підвищення безпеки веб-серверів.

Практична цінність полягає в тому, що розроблено програмний засіб для управління безпекою та подіями, який відкритий для масштабування завдяки модулям.

1 АНАЛІЗ ЛІТЕРАТУРНИХ ДЖЕРЕЛ

1.1 Огляд та аналіз літературних джерел з теми дослідження

Ризики кібербезпеки, що впливають на промислові системи управління (ІКТ), надзвичайно зросли за останні пару років, головним чином через активізацію національних держав і кіберзлочинців. Зловмисники стали більш досвідченими та небезпечними, а їх належне та своєчасне виявлення стало справжнім викликом. Прикладами поточних інцидентів кібербезпеки, що впливають на ІТ та ІКТ, є [1]: атаки програм-вимагачів; зловмисне програмне забезпечення, яке впливає на здатність комунального підприємства вести бізнес і операції; фішингові кампанії, спрямовані на керівників, помічників керівників, інженерів SCADA, ІТ-адміністраторів або інших привілейованих користувачів; інциденти зламу ділової електронної пошти, включаючи захоплення облікового запису або видавання себе за керівників; витік даних і крадіжки; соціальна інженерія для збору конфіденційної інформації від персоналу.

Згідно з звітом NIST [2], рішення з кібербезпеки в промислових системах керування повинні забезпечувати виявлення поведінкових аномалій у режимі реального часу, забезпечувати швидке управління інцидентами та дозволяти інтелектуальну візуалізацію мережі та всіх її взаємопов’язаних вузлів. Системи безпеки інформації та керування подіями (SIEM) розглядають вищезазначені можливості як вбудовані функції.

Загалом SIEM мають здатність збирати, агрегувати, зберігати та корелювати події, створені керованою інфраструктурою [3]. Вони становлять центральну платформу сучасних операційних центрів безпеки, оскільки вони збирають події з кількох датчиків (системи виявлення вторгнень, антивіруси, брандмауери тощо), корелюють ці події та надають синтетичні перегляди сповіщень для обробки загроз і звітування про безпеку [4]. Окрім цих ключових можливостей, існує багато відмінностей між існуючими системами, які зазвичай відображають різні позиції SIEM на ринку.

Кілька компаній розробили програмні продукти SIEM для виявлення мережевих атак і аномалій в інфраструктурі IT-системи. Серед них можна знайти класичні IT-компанії (наприклад, HP, IBM, Intel, McAfee), інші з більш далекоглядними можливостями (наприклад, AT&T Cybersecurity/AlienVault SIEM), а також багатообіцяючі інструменти, які слід брати до уваги в контексті SIEM (наприклад Splunk).

1.2 Порівняльна характеристика SIEM рішень

У даному підрозділі розглядатимуться найбільш широко використовувані інструменти управління інформацією про безпеку та подіями (комерційні та з відкритим кодом), щоб визначити їх основні характеристики, переваги та обмеження для виявлення поточних сценаріїв атак і реагування на них. Виконається поглиблений аналіз функцій і можливостей поточних SIEM, також проведеться аналіз їхніх обмежень, щоб запропонувати потенційні вдосконалення для створення власного SIEM. Аналіз зовнішніх факторів (наприклад, політичних, економічних, соціальних), які потенційно можуть вплинути на майбутні SIEM у середньостроковій та довгостроковій перспективі, надається як спосіб виявлення чинників і перешкод для нового покоління систем SIEM. Крім того, надається огляд рішень SIEM у критичних інфраструктурах, щоб визначити потенційне використання цих інструментів.

Системи безпеки інформації та керування подіями були розроблені у відповідь, щоб допомогти адміністраторам розробляти політики безпеки та керувати подіями з різних джерел. Зазвичай простий SIEM складається з окремих блоків (наприклад, вихідний пристрій, збір журналів, нормалізація синтаксичного аналізу, механізм правил, зберігання журналів, моніторинг подій), які можуть працювати незалежно один від одного, але без їхньої спільної роботи SIEM не працюватиме функціонувати належним чином [3]. На рис. 1.1 зображено основні компоненти базового рішення SIEM.

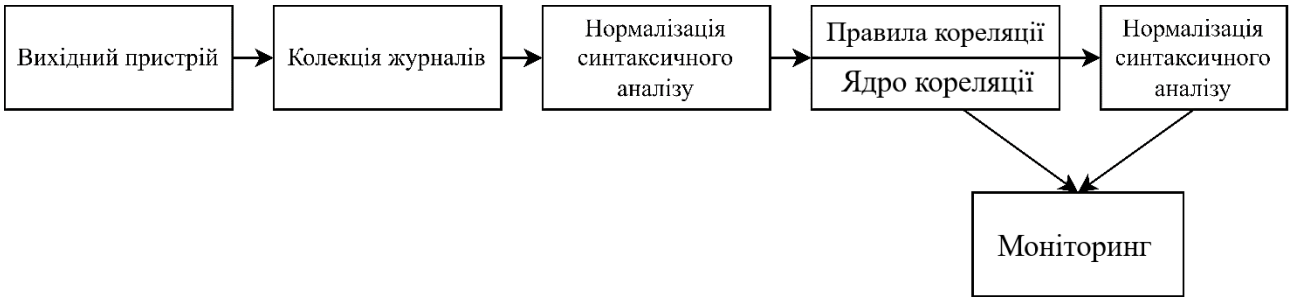


Рисунок 1.1 – Базові компоненти SIEM

Платформи SIEM забезпечують аналіз у реальному часі подій безпеки, створених мережевими пристроями та програмами. Крім того, незважаючи на те, що нове покоління SIEM надає можливості реагування для автоматизації процесу вибору та розгортання контрзаходів, поточні системи реагування вибирають і розгортають заходи безпеки, не виконуючи комплексного аналізу впливу атак і сценаріїв реагування.

Окрім цих спільних рис, поточні SIEM мають відмінності, які класифікують їх як лідерів, претендентів, нішевих гравців або провидців, згідно з річним звітом Gartner SIEM Magic Quadrant. Цей розділ представляє основні рішення SIEM, доступні на ринку на сьогоднішній день, і надає основні переваги та недоліки кожного з них на основі останнього звіту Gartner і дослідницьких робіт, пов’язаних з технологіями SIEM [4]. У табл. 1.1 розглянуто список рішень SIEM, запропонованих компанією Gartner протягом останнього десятиліття в їхньому щорічному звіті Magic Quadrant.

Таблиця 1.1 – Аналіз покриття рішення SIEM базовими функціями

SIEM	2010	2012	2014	2016	2018	2020
ArcSight Enterprise Security Manager	1	1	1	1	1	3
IBM Qradar	2	1	1	1	1	1
McAfee Enterprise Security Manager	3	1	1	1	1	3
LogRhythm Next GEN SIEM	3	1	1	1	1	1
Dell RSA Netwitness	1	2	2	2	2	1

У табл. 1 показано розвиток рішень SIEM (і постачальників SIEM) з 2010 по 2020 рік з кроком у два роки. Варто зауважити, що останній звіт на сьогоднішній день був опублікований у січні 2020 року. У табл. 1 одиницею позначено тих, хто лідирував на ринку, претенденти позначаються двійкою, а

нішеві гравці позначаються трійкою. Це випадок RSA, підрозділу безпеки корпорації EMC (Dell Technologies), який пропонує розвинену платформу NetWitness SIEM; IBM, яка пропонує інструмент під назвою Qradar; NetIQ/Microfocus/ArcSight, що пропонує ArcSight Enterprise Security Manager; McAfee/Intel, що пропонує McAfee Enterprise Security Manager, і LogRhythm, що пропонує платформу Nextgen SIEM.

1.2.1 Класифікація SIEM

Аналіз та оцінка систем безпеки були широко запропоновані в літературі. У той час як деякі дослідження зосереджені на комерційних аспектах, інші зосереджуються на технічних характеристиках, які можна покращити в поточних рішеннях SIEM. Такі організації, як Gartner [5], наприклад, пропонують комерційний аналіз систем SIEM на основі ринку та основних постачальників, для яких щорічно публікується звіт, щоб позиціонувати постачальників SIEM як лідерів ринку, конкурентів або нішевих гравців.

Інші установи безпеки (наприклад, Techtarget і Info-Tech Research Group) широко повідомляли про можливості рішень SIEM і про те, як можна порівнювати та оцінювати постачальників SIEM. Techtarget, з одного боку, випускає періодичні електронні посібники про захист систем SIEM і про те, як визначити стратегію SIEM, управління та успіх на підприємстві [5]. Info-Tech, з іншого боку, надає технічні звіти про ландшафт постачальників SIEM [6], зосереджуючись на перевагах і недоліках основних комерційних SIEM. Обидві організації беруть за основу свого аналізу магічний квадрант Gartner.

Протягом останнього десятиліття Gartner класифікувала рішення SIEM як лідери (організації, які добре працюють у порівнянні зі своїм поточним баченням і мають хороші позиції для завтрашнього дня), провидці (організації, які розуміють, куди рухається ринок або мають бачення зміни ринкових правил, але ще не дуже ефективні), нішеві гравці (організації, які успішно зосереджені на невеликому сегменті або не зосереджені та не випереджають інновацій або перевершують інших), і конкуренти (організації, які працюють добре сьогодні

або можуть домінувати у великому сегменті, але не демонструють розуміння напрямку ринку).

Окрім аналізу загроз, відповідності та керування журналами, розробники SIEM розглядають можливості UEBA та розумні інформаційні панелі як інновації, які слід додати до своїх рішень. Як наслідок, нові системи SIEM допоможуть адміністраторам безпеки за допомогою попередньо створених панелей інструментів, звітів, робочих процесів реагування на інциденти, розширеної аналітики, виявлення зв’язків між подіями та індикаторів безпеки [6]. Крім того, поглиблений аналіз масштабованості SIEM показав, що поточні рішення SIEM потребують покращення таких функцій, як аналіз поведінки, аналіз ризиків і розгортання, візуалізації, зберігання даних і можливості реагування, щоб не відставати від ринку [7].

1.2.2 Засоби SIEM

Беручи до уваги попередню інформацію про SIEM, можна підсумувати деякі з найперспективніших SIEM на сьогодні.

ArcSight Enterprise Security Manager від MicroFocus/HPE/NetIQ надає графічний інтерфейс для команди SOC і набір програм або зовнішніх команд, які допомагають виявлення зв’язків між подіями та/або процесам дослідження. У цього засобу обмежені можливості візуалізації та складні правила виявлення зв’язків між подіями [8]. Інформація, пов’язана з подіями, є незмінною, з очевидними недоліками, коли мова йде про адаптацію продукту до процесів і потреб компанії.

Qradar від IBM можна розгорнути як апаратне, програмне або віртуальне обладнання, а також SaaS у хмарі IBM. Надає інтерфейс користувача для перегляду подій у реальному часі, звітів, правопорушень, інформації про активи та керування продуктами. Засіб пропонує підтримку каналів аналізу загроз, також забезпечує основні можливості реагування, включаючи функції звітування та попередження. Проте моніторинг кінцевої точки для виявлення

загроз і реагування на них або базової цілісності файлів вимагає використання сторонніх технологій.

McAfee Enterprise Security Manager дозволяє використовувати масштабовану та універсальну архітектуру SIEM, забезпечуючи експертизу в режимі реального часу, також комплексний моніторинг трафіку/вмісту додатків і баз даних, розширені правила та виявлення зв’язків між подіями на основі ризиків для виявлення інцидентів у реальному часі. Для останнього вимагає використання додаткових рішень (наприклад McAfee Active Response). Прогностична аналітика та інші вбудовані функції, такі як аналіз поведінки, розвинені погано.

Платформа LogRhythm Next GEN SIEM забезпечує моніторинг кінцевих точок, аналіз мережі, аналітику поведінки користувачів і об’єктів, а також можливості реагування. Може бути розгорнута в пристрої, програмному забезпеченні або віртуальному екземплярі, що підтримує масштабовані децентралізовані архітектури. Не підходить для організацій із критично важливою інфраструктурою, хоча розширення можна розгортати для покращення можливостей SIEM. Вимагає високого ступеня автоматизації та готового вмісту.

Платформа RSA Netwitness від Dell аналізує дані та поведінку людей і процеси в мережі через журнали, пакети та кінцеві точки компанії. Засіб зосереджено на розширеному виявленні загроз. Забезпечує потужні можливості моніторингу операційних технологій. Це вимагає широкого розуміння широти опцій і наслідків для вартості, функціональності та масштабованості.

По суті, усі SIEM мають здатність збирати, зберігати та корелювати події, створені керованою інфраструктурою [9]. Окрім цих ключових можливостей, існує багато відмінностей між існуючими системами, які зазвичай відображають різні позиції SIEM на ринку. У табл. 1.2 структуровано рішення SIEM шляхом розподілення на декілька рівнів. Ця оцінка включає лише базову конфігурацію вибраних рішень SIEM, без додаткових функцій.

Таблиця 1.2 – Аналіз покриття рішення SIEM базовими функціями

Функціональність	ArcSight	Qradar	McAfee	LogRhythm	RSA
Правила виявлення зв’язків між подіями	2	2	3	3	2
Джерела даних	3	3	3	2	3
Обробка в реальному часі	3	3	3	3	3
Обсяг даних	3	2	3	2	2
Візуалізація	1	2	2	2	2
Аналітика даних	2	3	2	3	2
Продуктивність	2	2	3	2	3
Експертиза	1	3	3	2	3
Складність	3	2	2	2	3
Масштабованість	3	3	3	3	3
Аналіз ризиків	1	2	2	2	2
Зберігання	2	2	3	2	2
Вартість	3	3	3	2	3
Стійкість	2	3	3	2	3
Реакція та звіт	1	1	3	3	2
UEBA	3	3	1	3	3
Безпека	3	3	1	1	2

Засоби розподілені на декілька рівнів:

– низький/базовий рівень, де функції погано реалізовані або не реалізовані взагалі;

– середній рівень, де функції частково реалізовані;

– високий/просунутий, де функції повністю реалізовані (для найперспективніших рішень SIEM).

Для кожного з цих рівнів у табл. 1.2 присвоєні номери від 1 до 3 відповідно.

Правила виявлення зв’язків між подіями. Успішне виявлення події за допомогою SIEM залежить від потужності правил виявлення зв’язків між

подіями. Хоча більшість SIEM мають базові правила виявлення зв’язків між подіями, деякі з них мають надійні можливості пошуку та підтримують мови обробки пошуку для написання складних пошуків, які можна використовувати з даними SIEM.

Джерела даних. Однією з ключових особливостей системи SIEM є здатність збирати події з багатьох і різноманітних джерел даних у керованій інфраструктурі. Більшість SIEM підтримують кілька типів джерел даних, включаючи підтримувані датчики та підтримувані типи даних (наприклад, розвідка про загрози). Для інших рішень (наприклад, Qradar, USM) така функція може підтримуватися додатковими компонентами, інтегрованими в SIEM. Ця функція оцінює вихідні джерела даних, які підтримуються, і можливість для SIEM автоматично їх налаштовувати.

Обробка в реальному часі. Ця функція враховує здатність SIEM обробляти дані в реальному часі, які постійно змінюються. Він оцінює можливості керування, моніторингу та конвеєрної обробки в реальному часі, які використовує інструмент для запобігання інцидентам кібербезпеки або реагування на них, а також можливості обчислення продуктивності, які мають SIEM для аналізу мільйонів подій у режимі реального часу. Усі досліджувані SIEM мають розширені можливості обробки в реальному часі.

Обсяг даних. Аналіз великих обсягів даних, що надходять із різних джерел, важливий для отримання більшої інформації про зібрані події та кращого моніторингу. Однак зберігання великих обсягів зібраних даних у оперативній системі SIEM часто є дорогим і недоцільним. Ця функція оцінює можливість поточних систем підтримувати великі обсяги даних для виявлення зв’язків між подіями, індексування та операцій зберігання.

Візуалізація. одним із ключових факторів, які перешкоджають аналізу подій безпеки, є відсутність підтримки належних методів візуалізації даних і незначна підтримка інтерактивного дослідження зібраних даних. Тому важливо розуміти можливості аналізованих систем з точки зору створення нових методів візуалізації даних і користувальницьких інформаційних панелей.

Аналіз даних. новіші версії провідних SIEM підтримують широку інтеграцію з програмними та користувальницькими детекторами аномалій. Ці можливості включають аналіз поведінки співробітників, сторонніх підрядників та інших співробітників організації. Для цього SIEM має включати керування профілями користувачів/додатків і використання методів машинного навчання для виявлення неправильної поведінки.

Продуктивність. Ця функція оцінює продуктивність рішення SIEM з точки зору обчислювальної потужності, можливостей зберігання даних (наприклад, читання/запису), обробки правил виявлення зв'язків між подіями (наприклад, високопродуктивного механізму виявлення зв'язків між подіями), а також пошуку даних, індексування та моніторингу.

Експертиза. Окрім можливостей журналювання, деякі SIEM (наприклад, ArcSight, LogRhythm) пропонують вбудовані мережеві можливості експертизи, які включають захоплення пакетів повного сеансу з мережевих з'єднань, які вважаються шкідливими для перетворення пакетних даних у документи, веб-сторінки, голос через IP та інші розпізнані файли. Деякі інші продукти (наприклад, QRadar, Splunk) можуть зберігати окремі цікаві пакети за запитом аналітика безпеки, але не зберігають автоматично мережеві сеанси [10], а решта досліджуваних рішень не мають вбудованої мережі можливостей експертизи.

Складність. SIEM відомі тим, що їх важко розгортати та керувати ними. Однак важливо розуміти, чи можна аналізовану систему встановити для тестування з невеликими чи помірними зусиллями. З восьми досліджуваних SIEM ArcSight є інструментом найвищої складності для розгортання та керування, тоді як LogRhythm і Splunk розглядаються як прості та зручні інструменти для встановлення, розгортання та використання.

Масштабованість. Ця функція враховує здатність розгортання SIEM зростати не лише з точки зору апаратного забезпечення, але й з точки зору кількості подій безпеки, зібраних на межі інфраструктури SIEM. Нова цифрова трансформація веде до збільшення кількості датчиків і пристроїв (наприклад, серверів, агентів, вузлів), підключених до однієї мережі.

Аналіз ризиків. Останні версії провідних систем SIEM (наприклад, Qradar, LogRhythm, Splunk) містять функції для аналізу ризиків щодо активів керованої інфраструктури. Ця функція оцінює, чи підтримує SIEM аналіз ризиків або його можна інтегрувати із зовнішніми пристроями для цієї мети.

Зберігання. Враховуючи, що SIEM зазвичай зберігають інформацію не більше 90 днів, ця функція оцінює тривалість, протягом якої поточні технології SIEM зберігають дані, що зберігаються у їхніх системах, для подальшої обробки та операцій експертизи.

Вартість. Ця функція оцінює метод ліцензування, пов'язаний із рішенням SIEM (наприклад, корпоративне, безкоштовне, бета-версія, преміум), а також обмеження кількості користувачів, запитів, обсягів індексів, сповіщень, виявлення зв'язків між подіями, звітів, інформаційних панелей і автоматизованих дій щодо виправлення.

Стійкість. Стійкість або відмовостійкість є важливою характеристикою будь-якої критичної системи моніторингу. Важливо розуміти, якими є можливості відмовостійкості існуючих SIEM, наприклад, якщо механізм виявлення зв'язків між подіями підтримує відмовостійкість; спосіб підтримки аварійного відновлення та реплікації в сховищі подій; якщо з'єднувачі підтримують функції високої доступності.

Можливості реагування та звітування. Ця функція вивчає дії, які нативно підтримуються SIEM для реагування на інциденти безпеки (зокрема можливості спільного використання та звітування), а також спосіб, у який такі дії передаються механізму виявлення зв'язків між подіями.

UEBA. Ця функція оцінює, чи надає рішення SIEM власну функцію аналітики поведінки користувачів і суб'єктів (UEBA) або забезпечує інтеграцію зі сторонніми рішеннями UEBA.

Безпека. Ця функція оцінює здатність реалізувати автоматизацію безпеки, а також власні можливості шифрування, наявні в SIEM під час моніторингу, виявлення, виявлення зв'язків між подіями, аналізу та представлення результатів.

1.3 Аналіз обмежень сучасних SIEM

Незважаючи на те, що нове покоління SIEM надає потужні функції з точки зору виявлення зв’язків між подіями, зберігання, візуалізації та продуктивності, а також здатність автоматизувати процес реакції шляхом вибору та розгортання контрзаходів [11], поточні системи реагування дуже обмежені та контрзаходи вибираються та розгортаються без виконання комплексного аналізу впливу атак і сценаріїв реагування [11].

Крім того, більшість SIEM підтримують інтеграцію нових з’єднувачів або аналізаторів для збору подій або даних і надають API або інтерфейси RESTful для збору подій пізніше. Ці механізми дозволяють створювати доповнення та розширення до існуючих систем. Майбутні SIEM повинні використовувати цю функцію, щоб покращити якість подій, що передаються в систему (наприклад, використовуючи нові системи моніторингу або збираючи зовнішні дані з розвідувальних даних з відкритим кодом) через спеціальні конектори, а також надавати нові інструменти візуалізації, збираючи дані з SIEM сховище даних.

1.3.1 Невеликий набір правил виявлення зв’язків між подіями

Хоча поточні SIEM працюють з безліччю даних, жодна з них не має всіх даних, необхідних для обробки та виявлення всіх інцидентів безпеки. Причина полягає в тому, що збирати й обробляти всі необхідні дані невигідно. Як правило, усі SIEM корелюють журнали з VPN, брандмауерів, елементів керування доменом, невдалих з’єднань тощо. Більшість SIEM можуть співвідносити входи, зловмисне програмне забезпечення та веб-журнали, але лише деякі з поточних SIEM корелюють трафік DNS, журнали даних кінцевої точки, та журнали електронної пошти. Як наслідок, неможливо знати, хто кожен є в системі [12].

Ідентичність фрагментована, люди мають спільні облікові записи та різні ролі, пов’язані з одним і тим же користувачем, але згідно із законом не можна розкрити особу даної особи, оскільки це створює проблеми з конфіденційністю, як зазначено в Загальному регламенті захисту даних (GDPR) [12]. Якщо SIEM не в змозі охопити всі дані про користувачів і цінні активи, виявлення зв’язків між

подіями ніколи не працюватиме належним чином, що призведе до великої кількості помилкових позитивних і негативних результатів. Таким чином, наступне покоління SIEM має відповідати вимогам конфіденційності GDPR, надаючи аналітикам достатньо інформації для виявлення інцидентів безпеки [12].

Майбутні SIEM повинні досліджувати та впроваджувати нові неконтрольовані методи, які поєднують статистичний і багатокритеріальний аналіз рішень для автоматичного моделювання додатків і поведінки користувачів, а потім ідентифікують аномалії та відхилення від відомої належної поведінки, які є статистично значущими. Це призведе до розгортання розширених датчиків моніторингу додатків, які будуть передавати системам SIEM різні типи подій, які можна співвіднести з більш традиційними подіями безпеки, зібраними з хостів і мережевих пристроїв.

Завдяки поєднанню подій на основі аномалій із подіями, що надаються традиційнішими евристичними інструментами та інструментами на основі сигнатур, SIEM покращить частоту помилкових спрацьовувань цих компонентів, які традиційно були головним каменем спотикання для їх широкого застосування в реальних операціях.

1.3.2 Основні правила виявлення зв’язків між подіями

Платформи SIEM забезпечують аналіз у реальному часі подій безпеки, які генеруються мережевими пристроями та програмами [3]. Ці системи отримують великі обсяги інформації з різнорідних джерел і обробляють їх на льоту. Таким чином, їх розгортання зосереджено, по-перше, на написанні спеціальних збирачів і трансляторів для отримання інформації та її нормалізації, а по-друге, на написанні правил виявлення зв’язків між подіями для агрегування інформації та зменшення обсягу даних. Цей операційний фокус змушує розробників SIEM надавати пріоритет синтаксису над семантикою та використовувати мови для виявлення зв’язків між подіями, які є бідними з точки зору функцій [13]. Однак

із збільшенням кількості атак і, отже, різноманітності сповіщень, які отримують SIEM, потреба у відповідній обробці цих сповіщень стає важливою.

Поточні правила виявлення зв’язків між подіями SIEM слабкі [13]. Більшість із них використовують базове булеве об’єднання подій, які перевіряють певний шлях атаки (одну з багатьох тисяч можливих). Дуже небагато рішень SIEM мають вбудований розширений механізм виявлення зв’язків між подіями, здатний виконувати історичне виявлення зв’язків між подіями, корисну, наприклад, для перевірки після виявлення атаки нульового дня.

1.3.3 Неможливість зберігання даних про події

Для більшості існуючих рішень SIEM, коли дані заархівовано та виведено з активної системи, SIEM не використовуватиме їх. Крім того, спосіб обробки заархівованих даних або місце їх зберігання чи передачі залежить від користувача, і зазвичай це робиться вручну. Оскільки існують різноманітні варіанти того, де зберігати архівні дані, деякі користувачі SIEM вибирають приєднане сховище, інші використовуватимуть власну розподілену файлову систему, наприклад, розподілену файлову систему Hadoop (HDFS) [14], комерційне хмарне рішення для зберігання даних, як Amazon S3, Amazon Glacier або навіть використовувати операції «scp» на іншому пристрої.

Незалежно від використовуваного рішення для архівування, фактичний процес архівування складається із запуску сценаріїв, які часто створюються на замовлення для конкретного IT-середовища. Таким чином, сценарій, використаний одним клієнтом, може бути не корисним для потреб іншого клієнта, а зміна параметра архівування вимагає переписування сценарію архівування.

Поточні інфраструктури зазвичай зберігають необроблені події протягом обмеженого періоду часу (наприклад, 6 місяців), щоб обмежити простір для зберігання, який використовується для такого архівування (наприклад, 6 ТБ). Враховуючи те, що деякі вдосконалені постійні загрози виявляються через

багато місяців після їх появи в системі [14], таких можливостей зберігання може бути недостатньо, щоб допомогти з певними інцидентами.

Незважаючи на те, що хмара є перспективною, більшість компаній уникають використання хмари через побоювання, пов’язані з конфіденційністю подій (які містять конфіденційну інформацію) і побоювання, пов’язані з довірою таких важливих даних третім особам [14].

Мета майбутніх SIEM має зосереджуватися на пропонуванні безпечного та еластичного рішення для архівування даних незалежно від потреб зберігання даних із можливістю налаштування політик відповідно до вимог збереження.

1.3.4 Недостатня автоматизація процесів реагування на події

Дослідження технологій SIEM традиційно зосереджені на забезпеченні всебічної інтерпретації загроз, зокрема на оцінці їх важливості та відповідному визначенні пріоритетів відповідей. Однак у багатьох випадках реагування на загрози все ще вимагає від людей проведення аналізу та прийняття рішень щодо розуміння загроз, визначення відповідних заходів протидії та їх застосування. Це повільний і дорогий процес, який вимагає високого рівня досвіду, і, незважаючи на це, залишається схильним до помилок. Таким чином, останні дослідження технологій SIEM були зосереджені на здатності автоматизувати процес вибору та розгортання контрзаходів.

Відповідно до Scarfone [14] автоматизовані реакції повинні враховувати: (I) часову шкалу: час, потрібний SIEM для виявлення атаки та спрямування відповідного контролю безпеки для її пом’якшення; (II) безпека: зв’язок між SIEM та іншими засобами безпеки захищений таким чином, щоб запобігти прослуховування та зміни; (III) ефективність: здатність SIEM зупиняти атаки до того, як буде завдано збитків.

1.3.5 Неможливість індивідуального налаштування звітів

Традиційно SIEM підтримують створення директив безпеки для виявлення підозрілої поведінки в системі та повідомлення про тривоги. Однак ці

директиви/правила в принципі можна використовувати для ініціювання дій для модифікації керованої інфраструктури (наприклад, зміна конфігурації брандмауерів або NIDS).

Для деяких SIEM можна використовувати автоматичні тригери для виконання зовнішніх дій (наприклад, надсилання електронних листів, виконання сценаріїв, відкриття заявок), зазвичай через командний рядок. Однак більшість із цих систем не забезпечують попередньо налаштованих і налаштованих дій, які запускаються, коли виконується певна умова чи набір умов. Зазвичай вони зосереджені на створенні, розповсюдженні та управлінні звітами.

Важливою частиною проекту безпеки є глибокий захист [15] з використанням рівнів захисту, які зменшують ймовірність успішної атаки (або принаймні стримують її наслідки). Це вимагає використання різноманітності, включаючи, але не обмежуючись, використанням кількох систем виявлення вторгнень (IDS) і розрізнених розвідувальних даних із відкритим кодом (наприклад, пов’язаної з інфраструктурою інформації про безпеку з відкритих розвідувальних даних, доступних у різних джерелах з Інтернет). Досліджень щодо того, як вибрати серед альтернативних багатошарових засобів захисту, мало; іноді з’являються невідповідні моделі, які спираються на наївне припущення про незалежні збої між різними компонентами [15]. Інженери з безпеки мають мало або зовсім не мають теорії, щоб керувати своїми рішеннями щодо різноманітності, хоча інтуїція без сторонньої допомоги може бути дуже оманливою (наприклад, Littlewood and Wright [15]).

SIEM вже надають функціональність для читання журналів з кількох різних моніторів безпеки та інструментів виявлення на різних рівнях. Майбутні SIEM повинні створювати інструменти, які дозволяють консолідувати результати від кількох різноманітних моніторів подібного типу, які можуть здійснювати моніторинг подібних типів активів. Це допоможе підвищити точність виявлення та зменшити кількість помилкових тривог, про які повідомляють SOC.

Незважаючи на те, що потреба та актуальність для постачальників послуг безпеки, які мають системи звітування про кібербезпеку (CSRS), були визначені майже два десятиліття тому [15], все ще бракує рішень, зосереджених на управлінні та створенні обов'язкового звітування про інциденти відповідно до різних нормативних актів. каркаси. Крім того, незважаючи на те, що зростаюча кількість існуючих нормативно-правових актів і законодавства, що стосуються інцидентів кібербезпеки, викликала потребу в дослідженнях щодо звітування про інциденти кібербезпеки для певних сфер (наприклад, ядерні об'єкти [15], критично важливі для безпеки системи [15]), наразі ця функція є дуже обмежений у більшості комерційних і відкритих SIEM.

1.3.6 Обмежена візуалізація даних

Під час перегляду сучасного стану існуючих SIEM було помічено, що можливості звітування та візуалізації даних обмежені з точки зору підтримки ефективного вилучення корисної інформації з величезної кількості даних, які збирають системи. Незважаючи на те, що всі SIEM пропонують своїм користувачам можливості візуалізації даних, найчастіше візуальні представлення є загальними, не розробленими з урахуванням конкретних потреб користувачів, або навіть занадто рудиментарними, щоб мати значний вплив на те, як використовуються згенеровані дані [15].

Крім того, існуючі системи не мають можливості використовувати різноманітні режими даних, наприклад, результати статистичного моделювання, збори даних OSINT або комплексні моделі поведінки користувачів.

Щоб покращити можливості візуалізації існуючих систем, SIEM повинні зосередитися на гнучких платформах, здатних працювати з декількома джерелами даних, які несуть неоднорідні характеристики, і з даними, які постійно змінюються, тобто потоковими даними в реальному часі. Крім того, візуалізація повинна дозволити аналітикам безпеки краще профілювати систему за допомогою нових уявлень, які передають походження атаки, поточну діяльність, уразливості та характеристику сеансів/користувачів.

2 РОЗРОБКА МЕТОДУ УПРАВЛІННЯ БЕЗПЕКОЮ ТА ПОДІЯМИ

2.1 Обґрунтування вибору напрямку розробки методу

SIEM в основному стосується виявлення загроз, запобігання та керування ними. Метою платформи SIEM є забезпечення обізнаності про ситуацію в реальному часі. Це дозволяє організації вчасно виявляти атаки та реагувати на них.

Його архітектура відіграє вирішальну роль у безперебійній роботі SIEM. Загалом, перед тим, як SIEM буде введено в дію, слід приділити достатньо уваги його налаштуванням і технологічним аспектам.

Однією з невід'ємних частин архітектури управління інформацією про безпеку та подіями є операційний процес, який стоїть за створенням ефективної стратегії пом'якшення кіберзагроз. Однак уся ця інформація може бути надзвичайною і повинен бути спосіб її оптимізувати.

Правильні робочі вказівки можуть дозволити SIEM стати проактивною частиною повної архітектури керування інформацією безпеки підприємства. Інструменти SIEM беруть величезні обсяги даних, зібраних із різноманітних джерел, і об'єднують їх у дієву аналітичну інформацію, яка допомагає захистити бізнес. Плавна інтеграція між архітектурою SIEM і операційними процесами сприяє підвищенню продуктивності системи [16].

2.1.1 Інструменти для збирання даних

Інструменти SIEM збирають журнали, кілька типів даних і події з джерел в IT-системі організації. Після збору даних із цих пристроїв SIEM несе відповідальність за стандартизацію та збереження їх у форматі, який дозволяє легко аналізувати та переглядати. SIEM може збирати дані двома методами:

– Автоматизований збір даних: це включає в себе агента, встановленого на пристрої, і прямі з'єднання для отримання файлів журналу зі сховища (формат syslog) або протоколу потокової передачі подій, а також

зберігання їх в одному легкодоступному місці, щоб організація могла отримати повну видимість загроз що воно стикається.

– Характеристика активів: класифікуючи інфраструктуру інформаційних технологій організації, дуже важливо відокремити або розбити мережу на менші групи активів, які мають певну базову подібність або функцію. Прикладами категорій IT-активів є пристрої, мережі та програми. Це допомагає контролювати мережеву активність і виявляти активи з високим ризиком [17]. На табл. 2.1 зображені типові джерела даних для SIEM.

Таблиця 2.1 – Типові джерела даних SIEM

Програми	Події безпеки	Мережеві логи	Пристрої
Веб-додатки	Трафік брандмауера	Бездротові точки доступу	Мобільні пристрої
Додатки SaaS	Засоби захисту від зловмисного програмного забезпечення	Віртуальні мережі	Персональні ноутбуки або настільні ПК
Інтранет-додатки	Фільтри веб-додатків	Маршрутизатори, комутатори	Спільні робочі місця між людьми

Нижче в табл. 2.2 перераховані найпоширеніші методи та засоби та їх короткий опис.

Таблиця 2.2 – Найпоширеніші методи та засоби SIEM

№	Назва	Опис
1	2	3
1	Збір та агрегація	Збір та агрегації даних з різних джерел – мережевих пристроїв та сервісів, сенсорів систем безпеки, серверів, баз даних, додатків та інших джерел. Цими методами забезпечується консолідація даних для подальшого виявлення критичних даних
a.	Збір та одержання подій джерел	Методи та засоби збору, одержання подій, що реєструються в журналах реєстрації подій різних джерел, включають в себе збір та одержання подій за протоколами syslog, odbc, tftp, snmp, wmi та інші
b.	Збір мережних пакетів	Методи та засоби збору пакетів мережного трафіку
c.	Збір пакетів Netflow	Збір пакетів мережного протоколу Netflow, призначеного для обліку мережевого трафіку, розробленого компанією Cisco Systems.

Продовження таблиці 2.2

1	2	3
d.	Збір пакетів інших протоколів	Збір пакетів різних протоколів, призначених для обліку роботи мережного обладнання та програм
e.	Моніторинг продуктивності та доступності	Збір інформації за різними показниками продуктивності та доступності вузлі
2	Аналіз та виявлення зв’язків між подіями	Методи та засоби, що виконують пошук загальних атрибутів у зібраних даних, зв'язування даних у значні підгрупи. Забезпечують застосування різних технічних прийомів для інтеграції даних із різних джерел перетворення вихідних даних на значну інформацію – інциденти ІБ
3	Сповіщення	Методи та засоби генерації сповіщень (повідомлень) про виявлені інциденти або значущі події. Оповіщення може, як виводитися на екран моніторингу інтерфейсу системи, так і направлено в різні канали оповіщень: електронною поштою, на GSM-шлюз, системи обміну миттєвими повідомленнями (jabber) та інші
4	Візуалізація	Методи та засоби зі спеціального графічного відображення та візуалізації зібраних даних. Наприклад, у вигляді діаграм, які допомагають ідентифікувати аномалії або значні події, відмінні від стандартної поведінки.
5	Зберігання	Методи та засоби застосування довготривалого сховища даних в історичному порядку для виявлення зв’язків між подіями за часом та для забезпечення методів та засобів візуалізації. Довготривале зберігання необхідне проведення комп'ютерно-технічних експертиз, оскільки більшість розслідувань інцидентів проводиться після порушення.
a.	Нормалізація	Приведення зібраних даних до єдиного стандартного виду
6	Експертний аналіз та пошук	Методи та засоби, що забезпечують можливість пошуку по безлічі зібраних даних з різних джерел. Необхідно в рамках виконання комп'ютерно-технічних експертиз та розслідувань.
7	Інші методи та засоби	Методи та засоби, що не входять до жодної з вищеперелічених груп.

Після аналізу збору даних наступним кроком буде аналіз методів управління даними.

2.1.2 Інструменти управління даними

Після завершення збору даних починається процес керування ними. Дані, якщо вони зберігаються належним чином, можуть значно покращити безпеку інформації та функціонування керування подіями [18].

– Зберігання: інструменти SIEM здатні збирати величезну кількість даних. Ці дані можуть зберігатися локально, у хмарі або в обох. Найважливіше те, що місця зберігання повинні мати суворий захист, щоб уникнути будь-якої втрати даних.

– Багаторівневе: розміщення даних має ґрунтуватися на їх актуальності та важливості. Наприклад, гарячі дані, які використовуються для моніторингу безпеки в режимі реального часу, слід розміщувати у високопродуктивному сховищі. З іншого боку, холодні дані, які не можуть бути використані негайно, повинні бути розміщені на недорогих носіях.

– Категоризація: категоризація допомагає покращити продуктивність і компетенцію інструментів SIEM. Це можна зробити, оптимізувавши та індексуючи дані для аналізу, перегляду та дослідження, використовуючи дані про загрози для класифікації ризиків загроз, працюючи з алгоритмами виявлення, щоб зменшити ймовірність помилкових спрацьовувань, і встановлюючи політики для стандартизованих робочих процесів даних.

2.1.3 Інструменти зберігання даних

Інформація про безпеку та інструменти керування подіями містять велику кількість даних, які вимагають певної міри сортування та фільтрації, щоб зменшити непотрібні дані, які займають простір для зберігання в організації. Це, у свою чергу, також допомагає зберігати важливі дані [18].

Крім того, вимоги PCI, DSS, HIPAA та SOX передбачають обов’язкове зберігання критичних журналів від 1 до 7 років. Розумне ставлення до збереження даних також може допомогти проаналізувати тенденції поведінки користувачів і не дуже типові схеми безпеки на майбутнє. SIEM використовує такі методи для мінімізації обсягів журналів:

– Сервери Syslog: Syslog — це гнучкий стандарт журналу, який дозволяє організації зберігати велику кількість даних, зберігаючи їх у стандартизованому форматі. Syslog стискає великі обсяги даних і дозволяє легко шукати ці журнали за допомогою складних інструментів запитів.

– Графік видалення журналу: SIEM автоматично видаляє дані журналу, які зберігалися довше, ніж вимагає політика збереження. Файли журналу зазвичай можна отримати безпосередньо зі сховища у форматі Syslog.

– Фільтрування журналів: програмне забезпечення SIEM організації може фільтрувати журнали, які не завжди потрібні для відповідності вимогам або для цілей криміналістичного аналізу. Адміністратори SIEM можуть фільтрувати журнали за джерелом, типом, часом або іншими визначеними правилами. Крім того, використання фільтрації журналів може значно зменшити обсяг даних, що обробляються.

– Узагальнення: журнали можна узагальнювати, щоб зменшити кількість даних, що зберігаються для кожної події, але зберегти важливі біти, як-от кількість подій, унікальні IP-адреси тощо [19].

2.1.4 Інтеграція процесів у SIEM

При інтеграції процесів управління інформацією про безпеку та подіями з іншими інструментами кібербезпеки синергія забезпечує вищий захист для всієї компанії. Це пов’язано з використанням керованої SIEM платформи, яка інтегрується з різноманітними програмними інструментами, виявляючи, запобігаючи та зберігаючи загрози безпеці, коли вони виникають.

Найкраще в цьому рішенні полягає в тому, що воно не вибирає програмне забезпечення, яке використовує команда, якщо його дані можна безпечно встановити на платформі SIEM. Приклади кількох варіантів інтеграції SIEM включають програмне забезпечення для керування ідентифікацією та доступом, інструменти керування виправленнями, інструменти хмарної безпеки та інструменти керування ризиками сторонніх розробників [20].

2.1.5 Найкращі практики управління інформацією та подіями безпеки

Програмне забезпечення SIEM допомагає організаціям стежити за потенційними загрозами для своїх систем, виконуючи завдання автоматичного моніторингу для забезпечення відповідності галузевим нормам. Оскільки порушення безпеки є серйозною проблемою, важливо, щоб підприємства використовували доступне програмне забезпечення для кібербезпеки, оскільки воно просто допомагає зробити ситуацію більш керованою.

Визначення обсягу. Перш ніж інтегрувати програмне забезпечення SIEM в архітектуру безпеки підприємства, важливо зрозуміти конкретні цілі його впровадження. По-перше, варто визначити, які системи, користувачі, мережі та програми підпадають під моніторинг, і знайти частини даних, які є дуже конфіденційними. Це можна зробити, створивши в програмному забезпеченні правила на основі політики, а потім порівнявши їх із зовнішніми вимогами відповідності, щоб визначити тип інформаційної панелі та звітності, які знадобляться організації.

Ця практика допоможе вирішити, чи вибрати локальне рішення, хмарну реалізацію чи розміщення за допомогою технології віртуалізації. Крім того, належний обсяг також гарантує, що всі важливі аспекти контролюються без збору великої кількості непотрібних даних [21].

Пілотний запуск. Незалежно від того, готова компанія до великого кроку чи ні, розумно почати з маленького кроку. Те ж саме стосується і SIEM. Дуже важливо піддати програмне забезпечення SIEM пілотному запуску, щоб не використовувати несправні інструменти, які можуть зіпсувати системи та сповільнити процеси в довгостроковій перспективі. Щоб уникнути ймовірності помилок, варто почати із впровадження рішення SIEM в одній сфері бізнесу, а потім розширення зони його використання, оцінивши чи дасть воно бажані результати чи ні.

Можна виміряти ефективність, визначивши ключові показники впливу на результат будь-якого впровадження. Ці дані не лише допоможуть визначити, чи

були інвестиції компанії в технологію того варті, але й висвітлять сфери, де можна покращити з точки зору повернення більшої вартості [21].

Правила співвідношення. Програмне забезпечення SIEM пропонує широкий спектр попередньо налаштованих правил виявлення зв'язків між подіями. За бажанням групи безпеки можуть налаштувати програмне забезпечення відповідно до конкретних потреб організації та її клієнтів. Це можна зробити, увімкнувши все за замовчуванням і відстежуючи поведінку програмного забезпечення, щоб визначити обсяги вдосконалення та підвищити ефективність виявлення помилкових спрацьовувань.

Варто розглядати SIEM, як партнера, який допоможе відстежувати підозрілі події та захистити організацію від будь-яких майбутніх загроз. Найкращий спосіб підійти до виявлення зв'язків між подіями — ознайомитися з попередньо налаштованими правилами, а потім налаштувати їх на основі того, що потрібно корелювати, а що ні.

Наприклад, якщо є потреба у отримуванні сповіщення про будь-які інциденти, які можуть поставити під загрозу безпеку веб-сайту компанії, доцільно налаштувати правила виявлення зв'язків між подіями для поширених сповіщень SIEM, пов'язаних із порушеннями безпеки, такими як ін'єкція SQL або міжсайтовий сценарій [21].

Вимоги відповідності. Реєстрація безпеки із системи SIEM може забезпечити ваш бізнес важливою інформацією для демонстрації відповідності стандартам безпеки. Однак, якщо компанія не знає, чи ці правила випередили час, можна ненавмисно витратити гроші на систему SIEM, яка навіть не відповідає мінімальним вимогам безпеки.

Ось чому було б гарною практикою створити окремий документ із переліком усіх IT-регуляторів (HIPAA, GDPR, HITECH), яких потрібно дотримуватися, а потім узгодити ці вимоги з потенційним рішенням SIEM, яке розробляється.

У цьому випадку найкраще зв'язатися з постачальниками, які пропонують вбудовані функції, які підтримують точні вимоги щодо відповідності. Це

допоможе вибрати постачальників і дізнатися про вимоги до аудиту, включаючи необхідний обсяг даних журналу та час, протягом якого дані журналу залишаються сумісними [21].

Постійний моніторинг. Інструменти SIEM є важливою частиною безпеки мережі. Наявність їх на місці може допомогти швидко виявити атаки або вторгнення, які інакше могли б залишитися непоміченими програмним забезпеченням моніторингу. Однак надійне рішення SIEM вимагатиме реєстрації якомога більшої кількості даних із додатків і служб компанії, щоб мати достатньо інформації для успішного позначення незвичайної активності, коли вона виникає.

SIEM відстежує будь-які аномалії, такі як незвичайна поведінка користувачів у системах, спроби віддаленого входу та системні збої. Цей підхід дає можливість завчасно вжити профілактичних заходів щодо потенційної вразливості [21].

Комплексний план реагування на інцидент. Такі переваги, як моніторинг у реальному часі, сповіщення про виявлення IT-загроз і швидке реагування на інциденти безпеки, є важливим аспектом, який приходить разом із впровадженням програмного забезпечення SIEM. Однак, щоб належним чином реагувати на ці інциденти, організація, яка впроваджує SIEM, повинна прийняти правильні плани.

Варто переконатись, що є потужний план гри з іменами відповідних людей, детальним описом процесів ескалації та підходів до усунення несправностей. Це забезпечує зменшення кількості порушень і мінімізацію будь-яких можливих питань/проблем або принаймні їх вирішення якомога швидше [21].

Правильне розгортання. Розгортання програмного забезпечення може бути складним; ніхто ніколи не впевнений, як це буде, чи всі відповідатимуть добре і чи не зламається він несподівано. Можна подумати, що зроблено правильний вибір щодо інструменту SIEM, але якщо його не вдасться розгорнути – можуть виникнути додаткові проблеми.

Організація повинна переконатися, що вся необхідна інфраструктура та обладнання є на місці та готові до роботи. Крім того, важливо шукати попереджувальні знаки, такі як втома або помилкові тривоги, щоб зрозуміти, чи існують потенційні перешкоди для розгортання [21].

Межі мережі. У мережі та навколо неї існує декілька незахищених зон, які можуть призвести до кібератак. Уразливі зони існують на межі мереж, і їх потрібно ретельно контролювати програмним забезпеченням SIEM.

Ці вразливі зони включають брандмауери, маршрутизатори, порти та бездротові точки доступу. Найкраще регулярно реєструвати зміни в мережі та іншу інформацію, щоб проблеми були негайно враховані, як тільки вони виникають, і можна було вжити профілактичних заходів [21].

Пробні запуски. В ідеальному світі всі сповіщення безпеки, які генерує SIEM, були б принципово правильними та виявляли б атаки та події, коли вони відбуваються, у реальному часі. Однак насправді це не завжди так.

Шанси отримати помилкові спрацьовування нечувані. Наприклад, SIEM може ідентифікувати сканер уразливостей як агресивного зловмисника, позначати різні сповіщення про загрози та надсилати потік сповіщень. Тому доцільно провести тестовий запуск процесу інтеграції SIEM [21].

Регулярні огляди та моніторинг. Останнє, але не менш важливе, варто регулярно оцінювати всі кроки, описані вище, щоб переконатися, що все належним чином підтримується та налаштовано. Це включає перевірку функціональності SIEM, визначення того, чи може інфраструктура задовольнити поточні та майбутні потреби, а також оптимізацію всього, що вважається за потрібне, на основі результатів тестування продуктивності.

Крім того, дуже важливо постійно оновлювати всі інші інструменти безпеки, а також саму SIEM, оскільки час від часу з’являються нові вразливості. З огляду на те, що зловмисники стають розумнішими, ніж будь-коли, важливо залишатися напоготові та вживати заходів для пом’якшення будь-яких нових загроз, які можуть зустрітися на вашому шляху.

Програмні засоби управління безпекою та подіями забезпечили повну інфраструктуру безпеки для захисту організацій та їхніх мереж. Вони дозволили організаціям стати розумнішими у виявленні та запобіганні загрозам, допомагаючи командам безпеки швидко вирішувати найнагальніші проблеми, дотримуючись при цьому нормативних норм. SIEM, безсумнівно, є одним із найкорисніших способів посилити безпеку даних і гарантувати, що ваша організація не стане жертвою кібератак [21].

Взявши до уваги всі вищеперераховані методи та найкращі практики, які використовуються в сучасних SIEM, було вирішено розширити функціональні можливості SIEM шляхом автоматизації збору, категоризації та узагальнення даних. При цьому передбачається додаткова інтеграція завдяки спеціалізованим модулям.

2.2 Розробка методу для управління безпекою та подіями

Застосовуючи SIEM як службу, потрібно її правильно розгорнути. Також перед початком застосування системи потрібно опрацювати декілька запитань.

Потрібно спочатку відповісти, де буде впроваджуватись SIEM. Також потрібно визначити бази даних, цифрові активи та мережеві зони, які потребують нагляду та раннього виявлення загроз. Також потрібно знати повне покриття мережі.

– чи планується впровадити SIEM одним ударом? Натомість планується впроваджувати це за критерієм розташування, повільно та акуратно;

– чи є хороша команда безпеки? Вони будуть стежити за процесом SIEM. Для бездоганної роботи він потребує постійної уваги та перевірки даних;

– також потрібно визначити та зрозуміти проблему, яку потрібно позначити, для якої потрібне рішення.

Організація використовує SIEM як службу для багатозадачності. Його основною функцією є керування журналами. Але деякі з них також використовують його для дотримання різноманітних правил, таких як HIPPA, PCI, SOX і GDPR. Від агрегування даних можна відстежувати дані в часі. Це

також допомагає відстежувати розвиток організації. Його також можна використовувати як інструмент для складання бюджету [22].

Для успішного впровадження процесу SIEM потрібна активна команда безпеки в організації. Це також вимагає залучення всієї робочої сили організації. Співробітники є основою організації. Вони відіграють важливу роль у виконанні будь-якої функції організації. Працівник повинен бути уважним і працювати швидко. Вони можуть негайно ідентифікувати атаку та будь-яку підозрілу сторону у світі цифрових гаджетів.

Для успішного впровадження процесу SIEM організація повинна мати трьох членів команди [23]. Команди наступні:

– команда безпеки – повинна бути сформована команда для активної роботи. Вони відомі як співробітники служби безпеки. Вони будуть поширювати всю інформацію та сповіщення. Це допоможе організації вжити заходів проти онлайн-загроз;

– оперативна група – вони працюватимуть онлайн. Вони перевірять журнали, події та інші інциденти безпеки. Це забезпечить їм швидке вирішення всіх подібних проблем;

– команда відповідності – це важлива команда для організації. Вони оброблятимуть дані. Вони також складатимуться з галузевими та державними правилами.

Незалежно від того, чи потрібно піклуватися про IT-інфраструктуру чи фінансову інформацію організації, впровадження процесу SIEM є дуже важливим. Це запобіжить атакам зловмисного програмного забезпечення та надасть негайну допомогу.

У сучасних рішеннях застосовуються такі основні методи контролю функціонування з точки зору забезпечення доступності систем:

– збір та агрегація різноманітних даних, показників та лічильників про використання апаратних ресурсів системи, як правило за допомогою встановлюваних агентів на контрольованих вузлах або з використанням

протоколу SNMP (рівень споживання CPU, пам'ять, жорстких дисків, мережевих адаптерів та інших даних);

– аналіз та виявлення зв’язків між подіями зібраних даних для визначення або попередження досягнення порогових значень показників продуктивності та доступності з метою реагування чи запобігання позаштатним ситуаціям функціонування систем;

– автоматизоване виконання заздалегідь запрограмованих тестів, які перевіряють функціонування різних параметрів сервісів за заданим сценарієм. Успішне виконання таких тестових сценаріїв дозволяє підтверджувати доступність сервісів та систем на різних рівнях;

– автоматизоване реагування системи у вигляді виконання заданих скриптів, програм чи завдань при виявленні значних відхилень показників на етапі виявлення зв’язків між подіями;

– генерації оповіщень (повідомлень) про виявлені відхилення у продуктивності та доступності систем. Оповіщення може, як виводитися на екран моніторингу інтерфейсу системи, так і направлено в різні канали оповіщень: електронною поштою, на GSM-шлюз, системи обміну миттєвими повідомленнями (наприклад, jabber) та інші;

– візуалізація даних, що збираються у вигляді діаграм, що допомагають ідентифікувати аномалії або значні відхилення, відмінні від стандартної поведінки систем;

– зберігання зібраних даних у базі даних [24].

Рішення щодо моніторингу продуктивності та доступності є невід'ємною частиною повноцінної системи моніторингу ІБ. Такі рішення мають застосовуватися як моніторингу ІТ-систем, які забезпечують роботу технологічних процесів компанії, так контролю функціонування підсистем ІБ, які забезпечують захист цих ІТ-систем з метою забезпечення доступності обох.

Крім зазначеного функціоналу, SIEM-системи можуть також оснащуватися додатковими функціями, такими як управління ризиками та вразливістю, інвентаризація ІТ-активів, побудова звітів та діаграм тощо.

Автоматизоване реагування на інцидент також можна налаштувати, для цього використовують системи IRP (Incident Response Platform), які можуть без участі людини, наприклад, заблокувати зламаний обліковий запис або відключити інфікований ПК від мережі [24].

Розроблена методика матиме такі особливості:

1. збір даних: інструмент збору інформації збирає дані в реальному часі.
2. агрегація даних: дані співвідносяться з подібними подіями, щоб полегшити їх аналіз для людей. Програмне забезпечення робить інформацію легшою для використання та читання людьми, щоб спростити процес.
3. аналіз: дані аналізуються на наявність загроз, щоб сповістити адміністраторів/офіцерів безпеки. За допомогою ряду аналітичних засобів потенційно небезпечні дані відокремлюються від непроблемних даних, а адміністратори/офіцери безпеки сповіщаються про потенційні загрози.
4. виявлення та усунення порушень: порушення, виявлені шляхом збору та аналізу даних, ідентифікуються та виправляються.

У даному розділі було наведено обґрунтування вибору напряму розробки методу. Проаналізовано та наведено методи, які використовують популярні сучасні SIEM, їхню роботу з процесами та безпекою. Наведено приклади їхнього використання, а також їхні переваги та недоліки. За результатами аналізу було розроблено метод для управління безпекою та подіями. Завдяки аналізу було розглянуто можливість розширювати функціонал методу і засобу додатковими модулями.

3 РОЗРОБКА ПРОГРАМНОГО ЗАСОБУ ДЛЯ УПРАВЛІННЯ БЕЗПЕКОЮ ТА ПОДІЯМИ

3.1 Програмна реалізація засобу для управління безпекою та подіями

Для складання випробувального стенду будуть використовуватися open-source рішення та власні додатки на PHP та C# [25,26]. Архітектура програмного засобу представлена рис. 3.1.

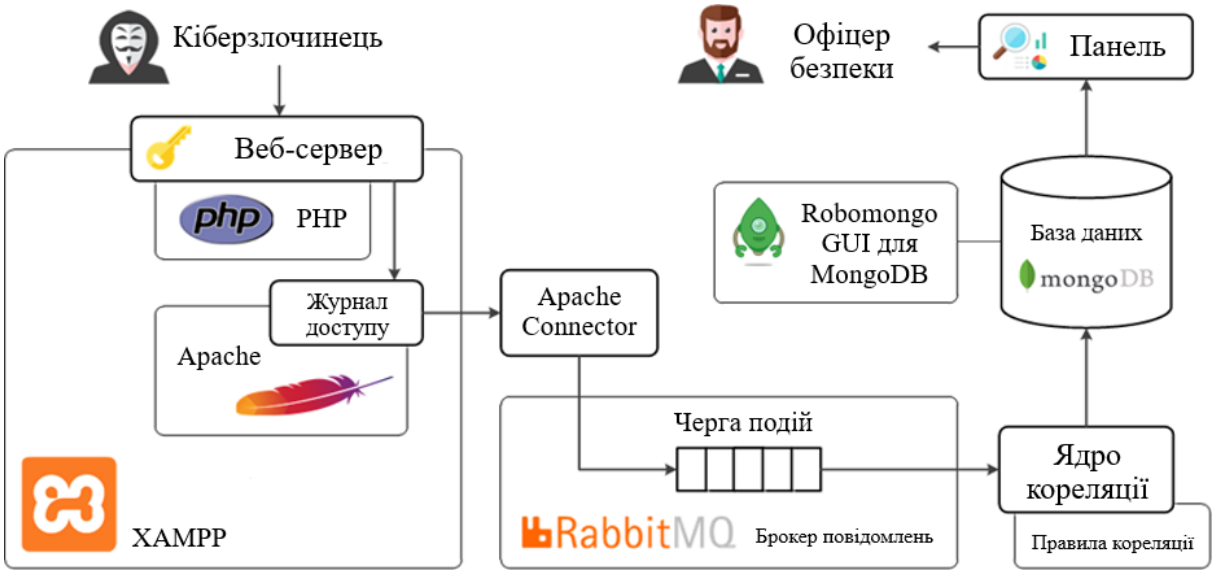


Рисунок 3.1 – Архітектура програмного засобу

Завдання SIEM системи – сповістити адміністратора безпеки про спробу реалізації такої атаки. Роботу програмного забезпечення можна звести до алгоритму на рис 3.2.

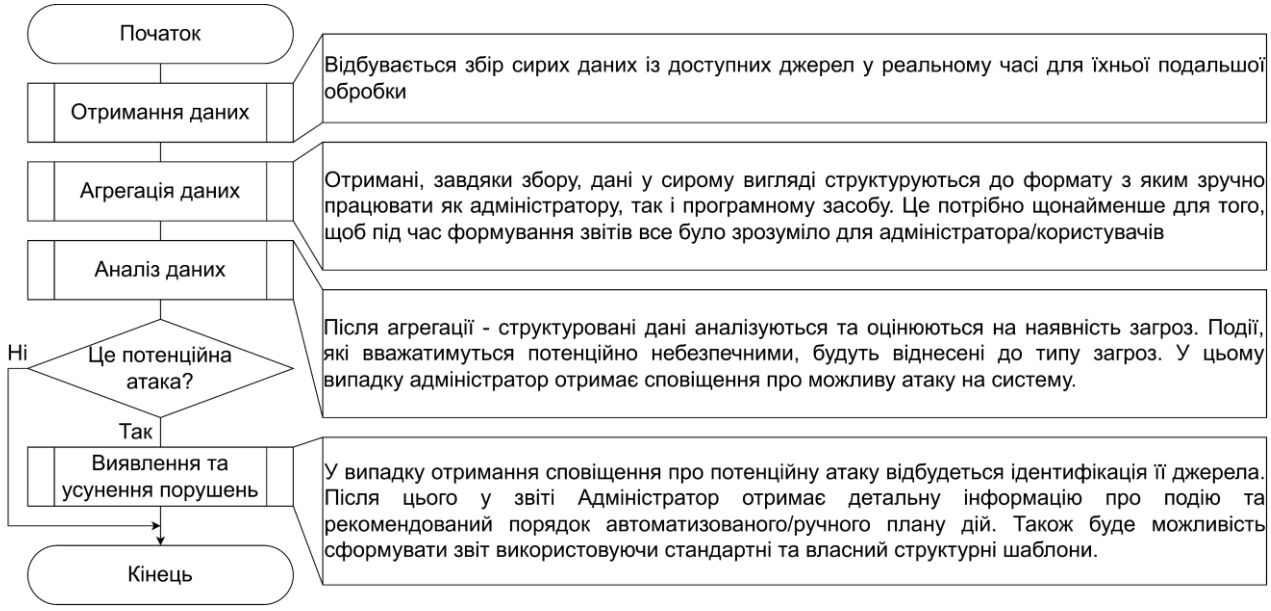


Рисунок 3.2 – Алгоритм засобу управління безпекою та подіями

3.1.1 Встановлення та налаштування веб-сервера Apache

Як веб-сервер використовуватиметься збірку XAMPP for Windows. Для перевірки правильності встановлення веб-сервера достатньо відкрити панель керування XAMPP Control Panel та запустити модуль Apache. Після цього у браузері за адресою «http://127.0.0.1/» відкриється вітальна сторінка проекту XAMPP [27].

3.1.2 Встановлення та налаштування сховища даних MongoDB

Для організації сховищ даних пропонується використовувати документну базу даних MongoDB. Адже вивчення комерційних SIEM систем дозволяє зробити висновок про те, що більшість провідних виробників разом із традиційними SQL базами даних у своїх рішеннях застосовують технології NoSQL/NewSQL [28].

Варто перевірити працездатність сервера, створивши в колекції новий документ з полем {a:1}. Спроба знайти документ у колекції має завершитися успіхом (рис. 3.3).

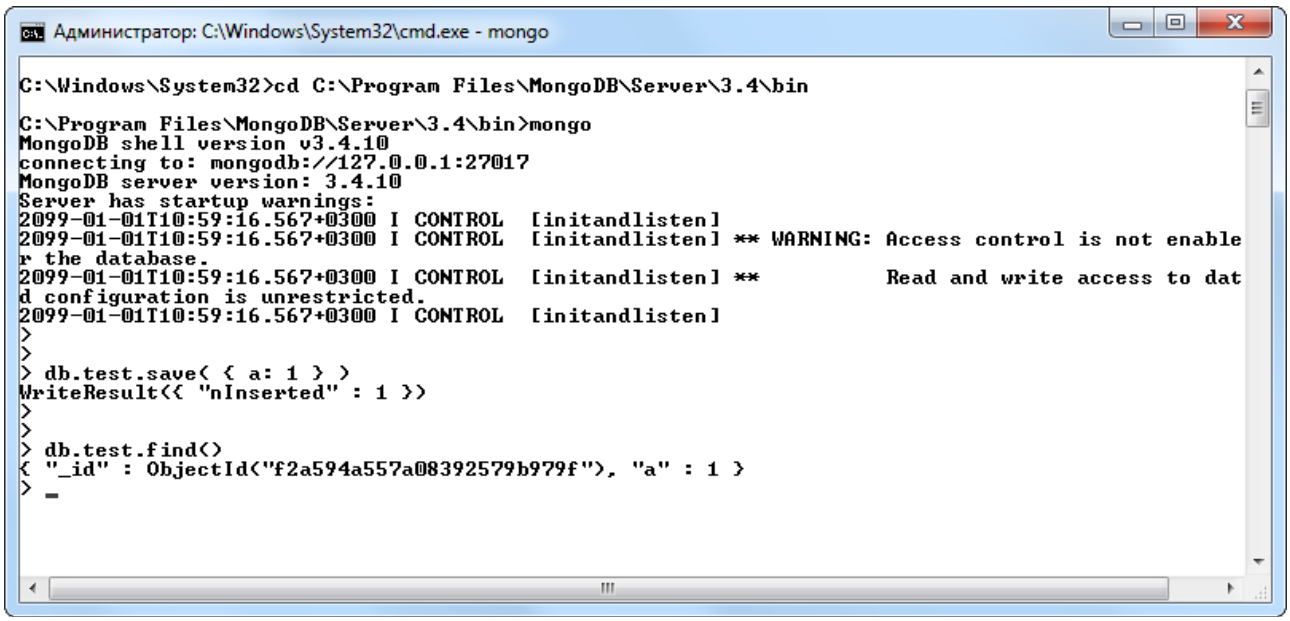


Рисунок 3.3 – Вигляд консолі для перевірки MongoDB

Варто зауважити, що при додаванні документа до колекції сервер MongoDB за замовчуванням документ доповнює полем _id типу ObjectId.

"Доповнення" це унікальне (не можна плутати з хешем, у нього правила формування інші) і дозволяє однозначно ідентифікувати документ у колекції.

Варто виконати ще одну перевірку, більш наочну. Варто завантажити один із доступних засобів адміністрування, наприклад Robomongo, і створити підключення до сервера MongoDB (рис. 3.4). У колекції повинен бути створений документ з полем { a: 1 } [29].

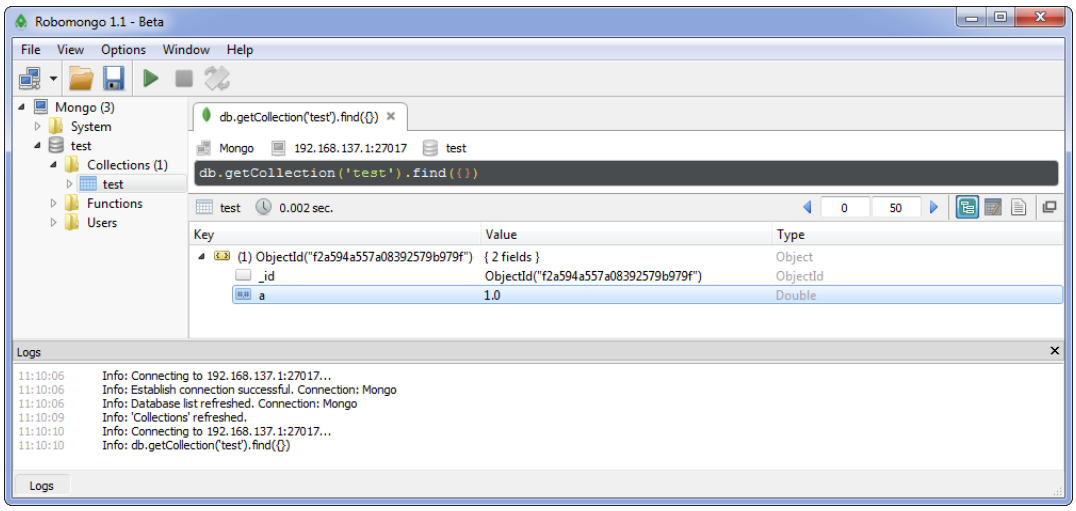


Рисунок 3.4 – Вигляд засобу адміністрування Robomongo

На рис. 3.3 замість очікуваної адреси 127.0.0.1 вказана адреса 192.168.137.1 – для експериментів використовується розподілений випробуваль-ний стенд, що складається з декількох фізичних робочих станцій і віртуальних машин. Але складнощів при встановленні всіх компонентів на одній робочій станції в хостовій операційній системі не повинно бути.

3.1.3 Встановлення та налаштування брокера повідомлень RabbitMQ

Для організації обміну даними між компонентами системи буде використано брокер повідомлень RabbitMQ. Це одне з найпоширеніших рішень подібного класу за версією stackshare.io. Також потрібно використати Erlang, попередньо перевіривши залежність версій. Для візуального контролю за функціонуванням брокера повідомлень буде використаний відповідний плагін Management plugin [30]. Після запуску плагіна в браузері за адресою буде доступна панель адміністрування (рис. 3.5).

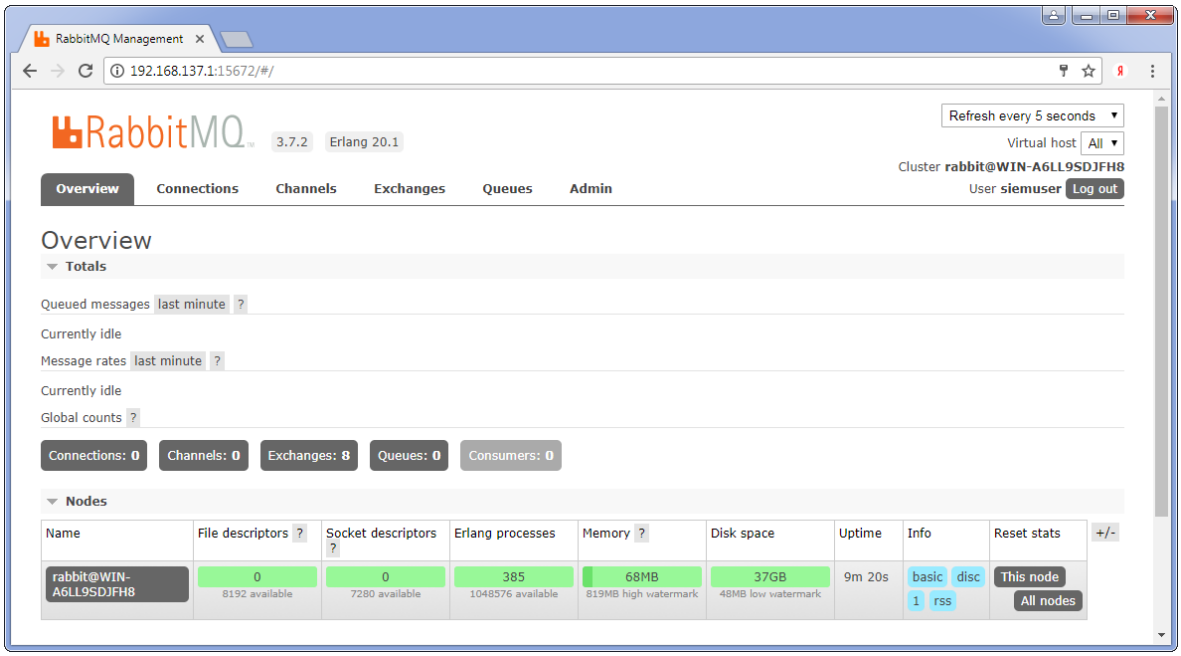


Рисунок 3.5 – Вигляд панелі адміністрування брокера

Під час збирання стенду буде розумно встановили сервер обміну повідомленнями на вузлі з адресою 192.168.137.1. Якщо потрібно віддалене підключення до панелі адміністрування, потрібно буде створити нового користувача та надати йому відповідні дозволи, наприклад:

```
rabbitmqctl add_user siemuser siempass
rabbitmqctl set_user_tags siemuser administrator
rabbitmqctl set_permissions -p / siemuser ".*" ".*" ".*"
```

Після перевірки підключення до панелі адміністрування встановлення та налаштування брокера повідомлень вважатиметься успішним. Перевірка обміну повідомленнями буде реалізована на наступних етапах [31].

3.2 Розробка веб-програми для захисту

Варто розробити веб-додаток, що реалізує функцію автентифікації користувача. Надихаємося ідеями мінімалізму в розмітці, створюється макет програми з використанням Bootstrap та PHP [32,25].

Скрипт admin.php запитує ім'я користувача та пароль та порівнює їх з відповідними параметрами адміністративного облікового запису. Якщо ім'я користувача та пароль не співпадають, виводиться повідомлення про помилку (рис. 3.6).

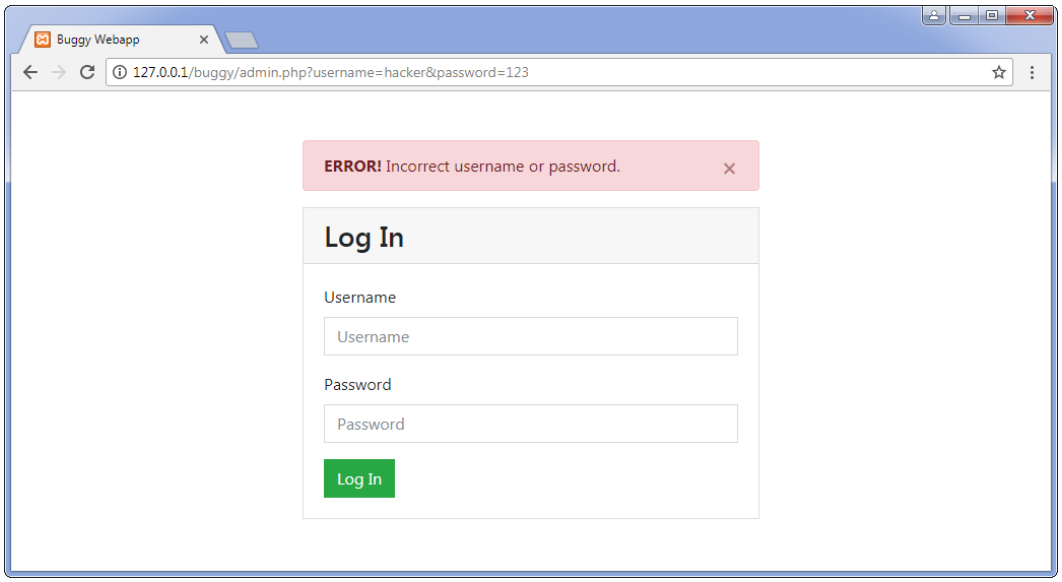


Рисунок 3.6 – Вигляд вікна авторизації при введенні неправильних даних

У разі правильно введених імені користувача та пароля надається доступ до адміністративного розділу (3.7).

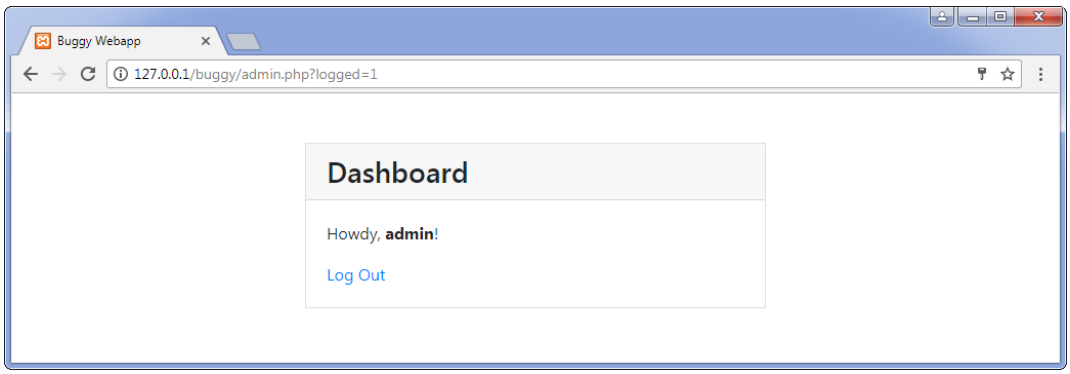


Рисунок 3.7 – Вигляд надання доступу до адміністративного розділу

З панелі керування XAMPP буде запущений веб-сервер, після чого веб-додаток буде доступним у браузері за адресою «http://127.0.0.1/buggy/admin.php».

3.3 Розробка конектора для веб-сервера Apache

Для відстеження звернення до веб-програми, що захищається та щоб спробувати виявити підозрілі дії користувачів варто переглянути журнал доступу (журнал звернень) веб-сервера. У даному випадку використання журналу звернень зберігається у файлі access.log. Після кількох завантажень веб-застосунку в браузері варто звернутись до вмісту файлу журналу:

```
127.0.0.1 - [01/Jan/2099:12:30:39 +0300] "GET /buggy/admin.php?username=hacker&password=123 HTTP/1.1" 200 2040 "http://127.0.0.1/buggy/admin.php" "Mozilla/5.0 (Windows NT 6.1; Win64; x64) appleWebKit/537.36 (KHTML, like Gecko) Chrome/61.0.3163.91 Safari/537.36"
```

Формат запису визначається в конфігураційному файлі httpd.conf веб-сервера Apache (приблизний шлях у XAMPP – «c: xampp apache conf httpd.conf»):

```
LogFormat "%h%l%u%t\"%r\"%>s%b \"%{Referer}i\" \"%{User-agent}i\"" combined
customLog "logs/access.log" combined
```

Варто познайомитись з детальним описом формату та приготувати схему мепінгу та приступати до розробки конектора. Варто пам’ятати, що питання ефективності коду більш важливі, ніж простота викладу. Загальний опис алгоритму:

– На початковому етапі відбуватиметься звернення до журналу доступу access.log та відбуватиметься запам’ятовування розміру файлу.

– Далі в безкінечному циклі з паузою між ітераціями відбуватиметься відслідковування зміни розміру файлу. При збільшенні розміру відбувається зчитування з файлу останніх доданих рядків та передавання у чергу повідомлень RabbitMQ, тепер відбувається запам’ятовування нового розміру файлу.

– Файл access.log можна перезаписати. Варто врахувати випадок зменшення розміру файлу.

Стиль форматування коду витримуватиметься згідно з прикладами Джеффрі Ріхтера та Стіва Макконнелла [33,34].

Для складання конектора відбувається підключення до проекту бібліотеки .NET/C# RabbitMQ Client Library. Для тестування запуску конектора з одночасною перевіркою працездатності брокера повідомлень відповідно до наступного сценарію:

1. Запуск програми ApacheConnector. Конектор починає відстежувати зміни розміру файлу access.log.

2. У браузері кілька разів оновлюється сторінка веб-програми Buggy Webapp, при цьому журнал доступу веб-сервера буде дописано рядки, що відповідають зверненням браузера до веб-сервера.

3. ApacheConnector виявить зміни розміру файлу і надішле останні рядки в чергу брокера повідомлень RabbitMQ (рис. 3.8).

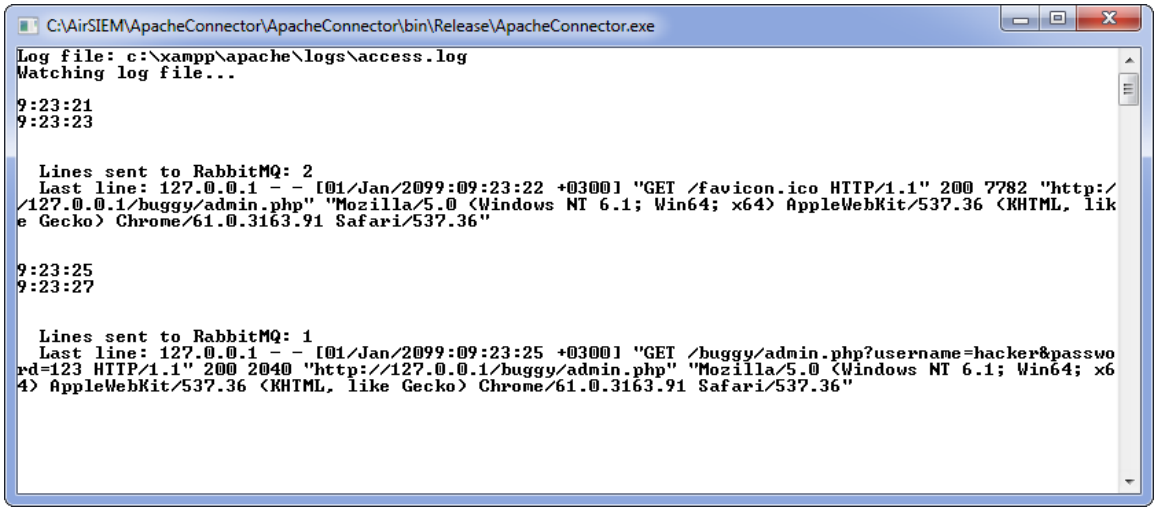


Рисунок 3.8 – Консоль брокера під час надсилання повідомлення

Якщо все правильно налаштовано, у панелі адміністрування RabbitMQ з’явиться додана черга з ненульовим навантаженням (рис. 3.9).

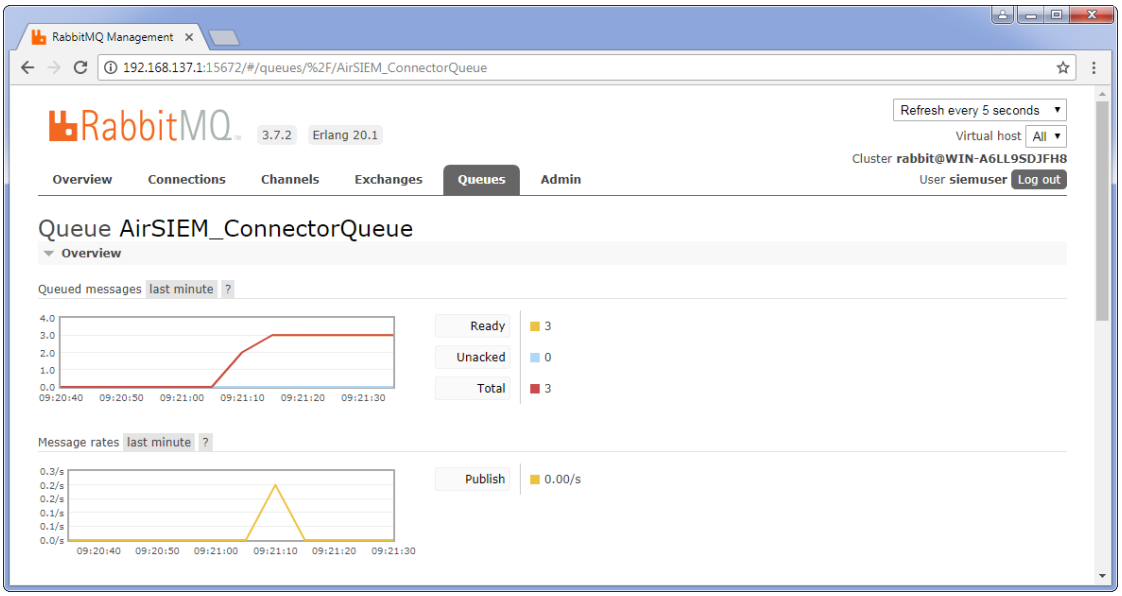


Рисунок 3.9 – Вигляд черги з ненульовим навантаженням.

Налагодивши роботу брокера повідомлень та конектора Apache можна переходити до розробки ядра виявлення зв’язків між подіями.

3.4 Розробка ядра виявлення зв’язків між подіями

Отже, до цього етапу було налаштовано підсистему збору подій безпеки від одного джерела – веб-сервера Apache [35]. Конектор ApacheConnector відстежуватиме зміни журналу звернень access.log і надсилатиме останні рядки в чергу брокера повідомлень RabbitMQ [36]. Наступний етап – розробка ядра виявлення зв’язків між подіями (обробника подій). Але попередньо, як було зазначено раніше, пропонується оцінити швидкість запису та читання зі сховища даних MongoDB. Очікується, що цей компонент являтиме собою конвеєр системи і визначить верхню межу продуктивності.

3.4.1 Оцінка продуктивності сховища даних MongoDB

Тестування продуктивності виконуватиметься простим способом – спочатку буде оцінена швидкість послідовного запису в сховищі (поодинокі документи та пакети документів), а потім буде оцінена швидкість випадкового читання документа з колекції. Для роботи з MongoDB до проекту необхідно підключити .NET Driver для MongoDB [37].

Після запуску програми можна оцінити результати тестування за допомогою рис. 3.10.

```
InsertMany by 1: 20000 ops in 13.52 seconds (1479.15 ops/sec) => 1479.15 docs/sec
InsertMany by 2: 10000 ops in 7.11 seconds (1406.24 ops/sec) => 2812.48 docs/sec
InsertMany by 5: 4000 ops in 3.20 seconds (1250.93 ops/sec) => 6254.66 docs/sec
InsertMany by 10: 2000 ops in 2.08 seconds (960.88 ops/sec) => 9608.78 docs/sec
InsertMany by 50: 400 ops in 1.15 seconds (347.68 ops/sec) => 17384.15 docs/sec
InsertMany by 100: 200 ops in 1.06 seconds (188.32 ops/sec) => 18832.26 docs/sec
InsertMany by 250: 80 ops in 0.95 seconds (83.87 ops/sec) => 20968.05 docs/sec
InsertMany by 500: 40 ops in 0.92 seconds (43.67 ops/sec) => 21835.64 docs/sec
InsertMany by 1000: 20 ops in 0.93 seconds (21.39 ops/sec) => 21391.22 docs/sec
InsertMany by 5000: 4 ops in 1.00 seconds (3.99 ops/sec) => 19936.32 docs/sec
InsertMany by 10000: 2 ops in 1.01 seconds (1.97 ops/sec) => 19730.96 docs/sec
InsertMany by 20000: 1 ops in 1.01 seconds (0.99 ops/sec) => 19832.80 docs/sec
Find: 10000 ops in 6.17 seconds (1620.21 ops/sec)
```

Рисунок 3.10 – Результати тестування швидкості MongoDB

Розподіл швидкості припадає на випадок відправки 500 подій в одному пакеті (блоці), швидкість послідовного запису при цьому – 21835 документів в секунду. Швидкість випадкового читання з колекції – 1620 документів на секунду. Враховуючи той факт, що тестова система організована з

використанням персональних комп'ютерів нетопових комплектацій, результати цілком влаштовують. Вважатиметься, що отримані в експерименті значення швидкості запису та читання дозволять сховищу даних MongoDB обробити заплановане у навантаження.

3.4.2 Формування набору правил виявлення зв’язків між подіями

Ядро виявлення зв’язків між подіями під час вступу чергової події безпеки намагається застосувати щодо нього заздалегідь завантажені правила обробки (правила виявлення залежностей між окремими подіями, правила виявлення зв’язків між подіями). Для цього буде сформовано набір правил.

Для опису правил використовуватиметься синтаксис правил виявлення зв’язків між подіями, запропонований у проекті OSSEC, із мінімальними змінами. Для прикладу моделюватиметься спрощений сценарій атаки типу «перебір пароля» щодо веб-програми, що потрібно захистити. Для цього прикладу складено відповідний набір правил, що знаходитиметься у файлі test_webapp_rules.xml [38].

Формат набору правил мінімально відрізняється від формату XML. Наприклад, допускається наявність у файлі кількох кореневих елементів. Такий документ не є коректним документом XML, ця особливість врахована при розборі файлу засобами, що надаються простором імен System.XML (рис. 3.11).

```
<group name="web-app">

  <rule id="100000" level="0">
    <match>/buggy/</match>
    <description>Access to BUGGY webapp</description>
  </rule>

  <rule id="100001" level="0">
    <if_sid>100000</if_sid>
    <match>password</match>
    <description>Attempt to login to BUGGY webapp</description>
  </rule>

  <rule id="100002" level="1" frequency="3" timeframe="5">
    <if_matched_sid>100001</if_matched_sid>
    <same_source_ip/>
    <description>Brute force trying to login to BUGGY webapp</description>
  </rule>

</group>
```

Рисунок 3.11 – Вміст файлу для виявлення зв’язків між подіями

Перше правило. Логіка правила 100000 наступна: оповіщати систему безпеки про всі спроби звернення до веб-застосунку Buggy Webapp, що захищається. Для цього відстежується рядок /buggy/ у всіх зверненнях до веб-сервера. Рівень значущості у правила «100000» нульовий, критичність у відстежуваних зверненнях відсутня, спрацьовування правила використовуватимуться для побудови складніших ланцюжків правил.

Елемент <rule> визначає правило. Атрибут id елемента <rule> визначає ідентифікатор правила. Ідентифікатори будуть вибрані з діапазону, який рекомендується проектом OSSEC для «авторських» правил: >=100000. Атрибут рівня елемента <rule> визначає рівень значущості правила. Мінімальне значення – 0, максимальне значення – 16. Елемент <match> задає підрядок для пошуку в рядку повідомлення, що обробляється. Елемент <description> задає опис правила, яке відображатиметься як оповіщення для адміністратора безпеки. Випадок спрацьовування такого правила перевіряється просто – якщо підрядок, вказаний в елементі <match>, буде виявлений у рядку повідомлення, що обробляється, ядро виявлення зв’язків між подіями сформує оповіщення безпеки.

Друге правило. В описі правила «100001» є новий елемент <if_sid> з ідентифікатором правила «100000», який накладає додаткову умову спрацьовування – необхідно, щоб до цього спрацювало правило «100000».

Логіка правила «100001»: якщо рядок, що обробляється, має відношення до доступу до веб-програми і при цьому у зверненні до веб-сервера виявлено рядок «password» (можливо свідчить про передачу пароля у форму введення імені користувача та пароля), то сповістити систему безпеки про спробу отримання адміністративного доступу – «Attempt to login to webapp». Правило «100001» дозволяє виявляти залежності між окремими подіями безпеки та коректно може називатися правилом виявлення зв’язків між подіями.

Третє правило. В описі правила «100002» є новий елемент <if_matched_sid> з ідентифікатором правила «100001», який накладає додаткову умову спрацьовування – необхідно, щоб правило «100001» спрацювало не менше 3 разів (атрибут frequency=«3» елемента <rule>) протягом останніх =«5»).

Порожній елемент <same_source_ip/> вказує на те, що при підрахунку спрацьовувань правила, заданого елементом <if_matched_sid>, враховуються лише спрацьовування з відповідними IP-адресами джерела.

Логіка правила «100002»: якщо протягом останніх 5 секунд з однієї і тієї ж адреси зафіксовано безліч спроб (3 і більше) отримання доступу до веб-додатку, то сформувати оповіщення з більш високим рівнем значущості level=«1» про спробу підбору пароля доступу – «Brute force trying to login to BUGGY webapp».

Набір правил виявлення зв’язків між подіями для прикладу сформований, наступним кроком буде перехід до безпосередньої реалізації обробника подій.

3.4.3 Реалізація ядра виявлення зв’язків між подіями

Програмна реалізація ядра виявлення зв’язків між подіями є найскладнішою і об'ємною частиною, по суті визначаючи всю логіку обробки подій безпеки SIEM системи. У загальному вигляді логіка роботи ядра виявлення зв’язків між подіями описується так:

– Ядро виявлення зв’язків між подіями прослуховує чергу повідомлень.

– При надходженні чергового повідомлення (події) ядро намагається застосувати щодо нього заздалегідь завантажені правила обробки подій (правила виявлення зв’язків між подіями).

– У разі застосування одного з правил до дії, що надійшла, ядро при необхідності формує інцидент безпеки і зберігає його в колекції alerts сховища даних MongoDB.

Реалізація буде достатньо об'ємною, тому для зручності налагодження буде правильно підключити до проекту систему логування NLog за допомогою Nuget менеджера. Крім того, для роботи з брокером повідомлень потрібно підключити пакет RabbitMQ.Client, а для взаємодії зі сховищем даних пакет MongoDB.Driver [39].

Згідно інструкції з налаштування логера NLog, варто додати до проекту конфігураційний файл nlog.config та вказати необхідність його копіювання у вихідний каталог [40].

Для перевірки правильності роботи ядра виявлення зв’язків між подіями необхідно переконатися, що у разі виявлення сценаріїв атак (заданих відповідними правилами) ядро формує інциденти безпеки і зберігає в колекції alert сховища даних MongoDB.

3.5 Розробка консолі адміністратора безпеки

Для зручності роботи адміністратора безпеки буде передбачене відповідне рішення – консоль керування. Функціонал консолі буде обмежений переглядом сформованих інцидентів безпеки, варіант реалізації – веб-додаток PHP. Щоб PHP міг взаємодіяти з MongoDB, потрібно виконати дві дії: до інтерпретатора PHP підключити відповідне розширення, а до веб-додатку – відповідну бібліотеку.

3.5.1 Підключення розширення php_mongodb.dll до веб-сервера

Варто згадати, що збірка XAMPP for Windows використовується як веб-сервер. За умовчанням інтерпретатор PHP не знає про існування сховища MongoDB, виправити це можна інсталяцією драйвера MongoDB PHP Driver on Windows [41].

Перейшовши на сайт pecl.php.net/package/mongodb у пошуках підходящої версії драйвера php_mongodb.dll варто обрати останню стабільну версію, Windows, PHP Version 7.1, thread safe, x86. Вибраний архів – php_mongodb-1.4.0-7.1-ts-vc14-x86.zip, в архіві потрібний файл php_mongodb.dll [41].

Скопіювавши драйвер php_mongodb.dll в папку розширень PHP. За замовчуванням "c:\xampp\php\ext" варто подати до файлу налаштувань PHP (за замовчуванням «c:xamppphp.ini») рядок extension=php_mongodb.dll в розділ підключення розширень.

Варто перезапустити Apache (найпростіше – з панелі XAMPP), щоб на наступних етапах не витрачати зайвий час на розбір помилок типу Uncaught Error: Class 'MongoDBDriverManager' not found'.

3.5.2 Підключення до веб-додатку бібліотеки MongoDB PHP Library

Є необхідність підключити до проекту консолі адміністратора бібліотеку MongoDB PHP Library [42].

Вивчаючи офіційну документацію рекомендувався найпростіший спосіб – за допомогою Composer [43]. Після створення папки для майбутнього проекту – варто виконати в папці команду установки пакету mongodb. Результат виконання команди зображений на рис. 3.12.

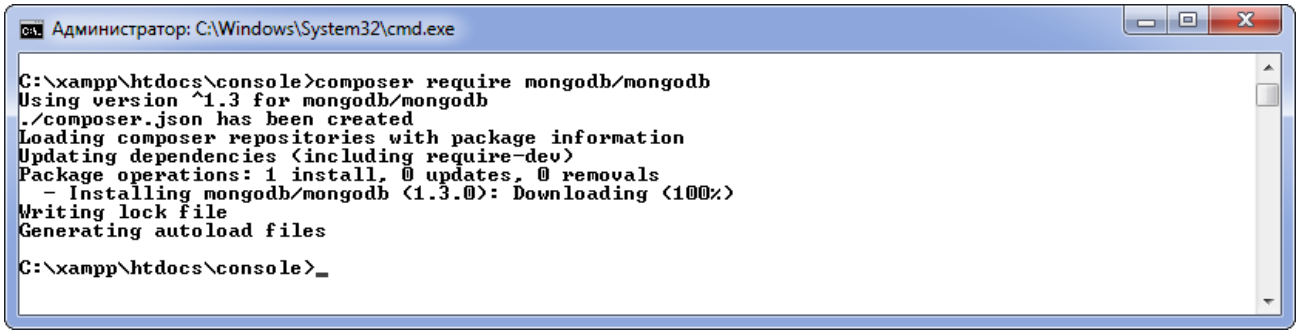
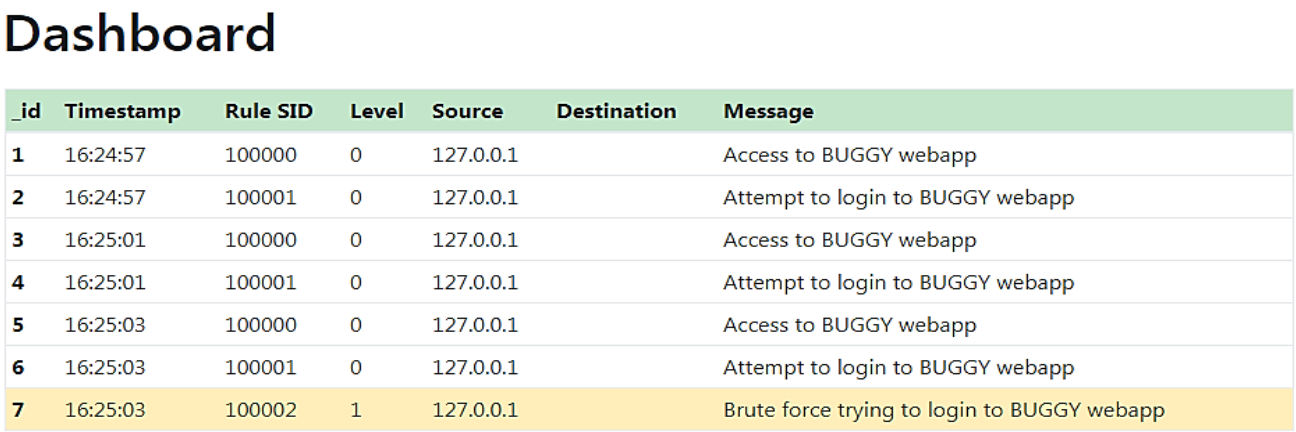


Рисунок 3.12 – Вигляд результату використання команди установки пакету

Composer завантажить необхідні файли та додасть їх у проект. Щоб проект міг використовувати функціонал бібліотеки MongoDB PHP Library, до коду потрібно додати наступний рядок:

```
require_once __DIR__ . "/vendor/autoload.php";
```

Таблиця може бути непустою при першому запуску, якщо ядро виявлення зв’язків між подіями під час тестових запусків сформувало інциденти безпеки та зберегло відповідні документи в колекції alerts сховища даних MongoDB (рис. 3.13).


Dashboard

_id	Timestamp	Rule SID	Level	Source	Destination	Message
1	16:24:57	100000	0	127.0.0.1		Access to BUGGY webapp
2	16:24:57	100001	0	127.0.0.1		Attempt to login to BUGGY webapp
3	16:25:01	100000	0	127.0.0.1		Access to BUGGY webapp
4	16:25:01	100001	0	127.0.0.1		Attempt to login to BUGGY webapp
5	16:25:03	100000	0	127.0.0.1		Access to BUGGY webapp
6	16:25:03	100001	0	127.0.0.1		Attempt to login to BUGGY webapp
7	16:25:03	100002	1	127.0.0.1		Brute force trying to login to BUGGY webapp

Рисунок 3.13 – Вигляд вікна консолі

Налагодивши роботу веб-сервера, MongoDB та консолі адміністратора безпки у цілому – можна переходити до перевірки працездатності системи.

3.6 Перевірка працездатності розробленої системи

Отже, розробка системи SIEM завершена. На заключному етапі варто виконати покрокове налагодження ядра виявлення зв’язків між подіями і переконатися в правильності роботи всього рішення в цілому.

Розглянемо підсумковий тестовий сценарій у загальному вигляді:

1. Зловмисник відкриває сторінку admin.php веб-програми і кілька разів намагається підібрати ім'я користувача та пароль адміністратора.

2. Після кількох безуспішних спроб зловмисник підбирає облікові дані адміністратора – «admin/admin» – та отримує доступ до адміністративного розділу.

3. Розроблена SIEM система в результаті обробки зареєстрованих подій безпеки із застосуванням заданих правил виявлення зв’язків між подіями виявляє реалізацію сценарію атаки, формує інцидент безпеки та сповіщає адміністратора.

Пропонується виконати тестовий сценарій та розглянути послідовність дій SIEM системи:

Попередня підготовка. Для "чистоти" експерименту варто очистити сховище даних. Для цього відбувається під’єднання до MongoDB за допомогою Robomongo та видаляється база даних AirSIEM. Також запускається ApacheConnector.

Моделювання дій зловмисника. Веб-додаток відкривається у браузері за адресою http://127.0.0.1/buggy/admin.php. Протягом 5 секунд робляться три спроби підбору імені користувача та пароля адміністратора.

Перевірка роботи конектора. У вікні програми ApacheConnector перевіряється факт читання журналу звернень веб-сервера та передачі зареєстрованих подій безпеки до черги брокера повідомлень. Наявність навантаження перевіряється на панелі адміністрування брокера повідомлень

RabbitMQ («http://127.0.0.1:15672/»): у черзі ConnectorQueue мають з'явитися повідомлення.

Перевірка ядра виявлення зв'язків між подіями. Далі процес обробки вступає ядро виявлення зв'язків між подіями, тому запускається додаток. Правильність роботи оцінюватимемо за лог-файлом Logs/log.txt (шлях щодо вихідного каталогу складання) логгера NLog [39]. На попередньому етапі завантажуються правила виявлення зв'язків між подіями:

```
ParseRuleDir start
ParseRulesFromXML handles file: test_webapp_rules.xml
  3 rules processed: 100000, 100001, 100002
ParseRuleDir: total 1 files processed
ParseRuleDir: total 3 rules processed
ParseRuleDir stop
```

Після завантаження правил виконується побудова ланцюжків правил:

```
CheckDependencies start
Dependencies:
       100000 children => 100001
       100001 children => 100002
 100002 children =>
CheckDependencies stop
```

Далі створюються черги до підрахунку числа спрацьовувань тих правил, ідентифікатори яких зустрічаються в елементах <if_matched_sid>. У тестовому наборі міститься 3 правила, серед яких лише одне правило 100002 підраховує спрацьовування іншого правила – з ідентифікатором 100001. Черга створюється для ідентифікатора 100001

```
GenerateQueueList start
Created 1 queues:
  [100001, FireQueue object => iD=[100001], count=[0], timeFrame=[5 sec],
maxSize=[1000]]
GenerateQueueList stop
```

Потім ініціалізується та запускається «підписник», який «слухає» чергу ConnectorQueue і передає прийняті повідомлення обробникові:

```
RabbitMQconsumer init
RabbitMQconsumer start
```

Після першої спроби зловмисника підібрати ім'я користувача та пароль конектор ApacheConnector передає черговий рядок з журналу звернень access.log

веб-сервера в чергу RabbitMQ, «підписник» отримує повідомлення та передає на обробку:

```
RMQ message received: apacheconnector:127.0.0.1 - - [01/Jan/2099:16:24:57 +0300] "GeT /buggy/admin.php?username=1&password...
```

З повідомлення формується нормалізована подія безпеки (примірник класу SecurityEvent):

```
SecurityEvent object => timestamp=[16:24:57], source=[127.0.0.1], destination=[], message=[GET /buggy/admin.php?username=1&password=1 HTTP/1.1]
```

Далі ядро виявлення зв’язків між подіями застосовує завантажені правила до події, що обробляється. При першому «проході» проглядаються ті правила, спрацьовування яких залежить від інших правил:

```
Check rule 100000 - access to BUGGY webapp
  Check <match>/buggy/</match>
  Rule 100000 matched
  ALERT: LEVEL 0 - access to BUGGY webapp
```

Правило 100000 спрацьовує, формується інцидент безпеки, оповіщення передається в MongoDB і відображається в консолі адміністратора (під час налагодження відображатимуться всі оповіщення, навіть з нульовим рівнем критичності).

Далі рекурсивно проглядаються правила, спрацьовування яких залежить від спрацьовування правила 100000 (рис. 3.14).

```
  Check the child rules
  Check rule 100001 - Attempt to login to BUGGY webapp
    Check <if_sid>100000</if_sid>
    Check <match>password</match>
    Rule 100001 matched
    ALERT: LEVEL 0 - Attempt to login to BUGGY webapp
    Matched rule is queue tracked
    Enqueue item: FireQueueItem object => timestamp=[16:24:57], source=[127.0.0.1], destination=[]
    FireQueue object => ID=[100001], count=[1], timeFrame=[5 sec], maxSize=[1000]
      1: FireQueueItem object => timestamp=[16:24:57], source=[127.0.0.1], destination=[]
    Check the child rules
    Check rule 100002 - Brute force trying to login to BUGGY webapp
      Check <if_matched_sid>100001</if_matched_sid>
        QueueDictionary.CheckIfMatched start
          counter++ => counter=[1]
          counterSameSourceIP++ => counterSameSourceIP=[1]
        Rule 100001 QueueDictionary.CheckIfMatched == FALSE
    Rule 100002 not matched
    Check rule 100002: OK
    Check the child rules: OK
  Check rule 100001: OK
  Check the child rules: OK
Check rule 100000: OK
```

Рисунок 3.14 – Вигляд розгляду правил

В результаті обробки спрацьовує правило 100001, у чергу підрахунку FireQueue заноситься перше спрацювання. Коли кількість спрацьовувань дорівнюватиме 3, виконається одна з умов спрацьовування правила 100002.

```
RMQ message received: apacheconnector:127.0.0.1 - - [01/Jan/2099:16:25:02 +0300]
"GET /buggy/admin.php?username=1&password...
SecurityEvent object => timestamp=[16:25:02], source=[127.0.0.1], destination=[],
message=[GeT /buggy/admin.php?username=1&password=1 HTTP/1.1]
Check rule 100000 - access to BUGGY webapp
  check <match>/buggy/</match>
  Rule 100000 matched
  ALERT: LEVEL 0 - access to BUGGY webapp
  Check the child rules
  Check rule 100001 - attempt to login to BUGGY webapp
    Check <if_sid>100000</if_sid>
    Check <match>password</match>
    Rule 100001 matched
    ALERT: LeVeL 0 - attempt to login to BUGGY webapp
```

Після спрацювання правила 100001 варто перевірити, чи не підраховується кількість спрацьовувань у будь-якій черзі підрахунку FireQueue. Для ідентифікатора правила 100001 така черга існує:

```
Matched rule is queue tracked
    Enqueue    item:    FireQueueItem    object    =>    timestamp=[16:25:02],
source=[127.0.0.1], destination=[]
    FireQueue object => ID=[100001], count=[3], timeFrame=[5 sec], maxSize=[1000]
      1:  FireQueueItem  object  =>  timestamp=[16:24:57],  source=[127.0.0.1],
destination=[]
      2:  FireQueueItem  object  =>  timestamp=[16:25:01],  source=[127.0.0.1],
destination=[]
      3:  FireQueueItem  object  =>  timestamp=[16:25:02],  source=[127.0.0.1],
destination=[]
```

Перевірка сховища даних. Перевіряється формування оповіщень. Для цього варто підключитись до MongoDB за допомогою Robomongo та переглянути колекцію документів alerts бази даних AirSIEM. Колекція має бути не пустою.

Моделювання дій адміністратора безпеки. Варто відкрити у браузері консоль адміністратора безпеки за адресою: http://127.0.0.1/console/index.php. Якщо всі компоненти SIEM системи налаштовані правильно, після виконання тестового сценарію вікно консолі повинне мати вигляд, як на рис. 3.15.

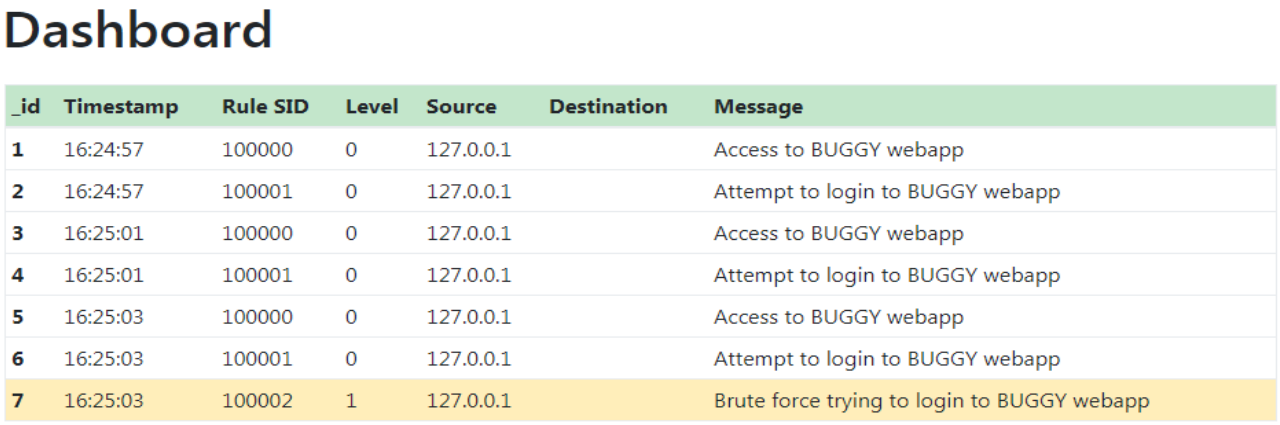


Рисунок 3.15 – Вигляд вікна консолі

Варто підкреслити, виявлено не лише окремі події, а виділено послідовність дій зловмисника та ідентифіковано сценарій атаки. У консолі адміністратора це рядок під номером 7, виділений кольором, з повідомленням про виявлену реалізацію сценарію атаки типу «перебір пароля». Мітка часу 16:25:03 встановлена ядром виявлення зв’язків між подіями, значення відрізняється від мітки часу події, що відповідає третій спробі отримання доступу зловмисника.

Якщо сповіщення з нульовим рівнем значущості розглядати як помилкові спрацьовування, можна вимкнути їхнє виведення в консоль і знизити цим навантаження на оператора.

Підсумки. Адміністратора безпеки повідомлено про підозрілі дії зловмисника. Розроблена система SIEM виконала поставлене завдання.

4 ЕКОНОМІЧНА ЧАСТИНА

Науково-технічна розробка має право на існування та впровадження, якщо вона відповідає вимогам часу, як в напрямку науково-технічного прогресу та і в плані економіки. Тому для науково-дослідної роботи необхідно оцінювати економічну ефективність результатів виконаної роботи.

Магістерська кваліфікаційна робота на тему «Система управління безпекою та подіями. Частина 1. Метод та програмний засіб для управління безпекою та подіями» відноситься до науково-технічних робіт, які орієнтовані на виведення на ринок (або рішення про виведення науково-технічної розробки на ринок може бути прийнято у процесі проведення самої роботи), тобто коли відбувається так звана комерціалізація науково-технічної розробки. Цей напрямок є пріоритетним, оскільки результатами розробки можуть користуватися інші споживачі, отримуючи при цьому певний економічний ефект. Але для цього потрібно знайти потенційного інвестора, який би взявся за реалізацію цього проекту і переконати його в економічній доцільності такого кроку.

Для наведеного випадку нами мають бути виконані такі етапи робіт:

1) проведено комерційний аудит науково-технічної розробки, тобто встановлення її науково-технічного рівня та комерційного потенціалу;

2) розраховано витрати на здійснення науково-технічної розробки;

3) розрахована економічна ефективність науково-технічної розробки у випадку її впровадження і комерціалізації потенційним інвестором і проведено обґрунтування економічної доцільності комерціалізації потенційним інвестором.

4.1 Проведення комерційного та технологічного аудиту науково-технічної розробки

Метою проведення комерційного і технологічного аудиту дослідження за темою «Система управління безпекою та подіями. Частина 1. Метод та програмний засіб для управління безпекою та подіями» є оцінювання науково-

технічного рівня та рівня комерційного потенціалу розробки, створеної в результаті науково-технічної діяльності.

Оцінювання науково-технічного рівня розробки та її комерційного потенціалу рекомендується здійснювати із застосуванням 5-ти бальної системи оцінювання за 12-ма критеріями, наведеними в табл. 4.1 [44].

Таблиця 4.1 – Рекомендовані критерії оцінювання науково-технічного рівня і комерційного потенціалу розробки та бальна оцінка

Бали (за 5-ти бальною шкалою)					
1					
№	0	1	2	3	4
2	3	4	5	6	7
Технічна здійсненність концепції					
1	Достовірність концепції не підтверджена	Концепція підтверджена експертними висновками	Концепція підтверджена розрахунками	Концепція перевірена на практиці	Перевірено працездатність продукту в реальних умовах
Ринкові переваги (недоліки)					
2	Багато аналогів на малому ринку	Мало аналогів на малому ринку	Кілька аналогів на великому ринку	Один аналог на великому ринку	Продукт не має аналогів на великому ринку
3	Ціна продукту значно вища за ціни аналогів	Ціна продукту дещо вища за ціни аналогів	Ціна продукту приблизно дорівнює цінам аналогів	Ціна продукту дещо нижче за ціни аналогів	Ціна продукту значно нижче за ціни аналогів
4	Технічні та споживчі властивості продукту значно гірші, ніж в аналогів	Технічні та споживчі властивості продукту трохи гірші, ніж в аналогів	Технічні та споживчі властивості продукту на рівні аналогів	Технічні та споживчі властивості продукту трохи кращі, ніж в аналогів	Технічні та споживчі властивості продукту значно кращі, ніж в аналогів
5	Експлуатаційні витрати значно вищі, ніж в аналогів	Експлуатаційні витрати дещо вищі, ніж в аналогів	Експлуатаційні витрати на рівні експлуатаційних витрат аналогів	Експлуатаційні витрати трохи нижчі, ніж в аналогів	Експлуатаційні витрати значно нижчі, ніж в аналогів
Ринкові перспективи					
6	Ринок малий і не має позитивної динаміки	Ринок малий, але має позитивну динаміку	Середній ринок з позитивною динамікою	Великий стабільний ринок	Великий ринок з позитивною динамікою

Продовження таблиці 4.1

1					
2	3	4	5	6	7
Практична здійсненність					
7	Активна конкуренція великих компаній на ринку	Активна конкуренція	Помірна конкуренція	Незначна конкуренція	Конкурентів немає
8	Відсутні фахівці як з технічної, так і з комерційної реалізації ідеї	Необхідно наймати фахівців або витрачати значні кошти та час на навчання наявних фахівців	Необхідне незначне навчання фахівців та збільшення їх штату	Необхідне незначне навчання фахівців	Є фахівці з питань як з технічної, так і з комерційної реалізації ідеї
9	Потрібні значні фінансові ресурси, які відсутні. Джерела фінансування ідеї відсутні	Потрібні незначні фінансові ресурси. Джерела фінансування відсутні	Потрібні значні фінансові ресурси. Джерела фінансування є	Потрібні незначні фінансові ресурси. Джерела фінансування є	Не потребує додаткового фінансування
10	Необхідна розробка нових матеріалів	Потрібні матеріали, що використовують ся у військово промисловому комплексі	Потрібні дорогі матеріали	Потрібні досяжні та дешеві матеріали	Всі матеріали для реалізації ідеї відомі та давно використовуютьс я у виробництві
11	Термін реалізації ідеї більший за 10 років	Термін реалізації ідеї більший за 5 років. Термін окупності інвестицій більше 10-ти років	Термін реалізації ідеї від 3-х до 5-ти років. Термін окупності інвестицій більше 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій менше 3-х років
12	Необхідна розробка регламентних документів та отримання великої кількості дозвільних документів на виробництво та реалізацію продукту	Необхідно отримання великої кількості дозвільних документів на виробництво та реалізацію продукту, що вимагає значних коштів та часу	Процедура отримання дозвільних документів для виробництва та реалізації продукту вимагає незначних коштів та часу	Необхідно тільки повідомлення відповідним органам про виробництво та реалізацію продукту	Відсутні будь- які регламентні обмеження на виробництво та реалізацію продукту

Результати оцінювання науково-технічного рівня та комерційного потенціалу науково-технічної розробки потрібно звести до табл. 4.2.

Таблиця 4.2 – Результати оцінювання науково-технічного рівня і комерційного потенціалу розробки експертами

Критерії	Експерт (ПІБ, посада)		
	1	2	3
	Бали:		
1. Технічна здійсненність концепції	4	5	4
2. Ринкові переваги (наявність аналогів)	2	2	1
3. Ринкові переваги (ціна продукту)	1	1	1
4. Ринкові переваги (технічні властивості)	1	2	2
5. Ринкові переваги (експлуатаційні витрати)	2	2	2
6. Ринкові перспективи (розмір ринку)	3	4	3
7. Ринкові перспективи (конкуренція)	3	2	3
8. Практична здійсненність (наявність фахівців)	5	5	5
9. Практична здійсненність (наявність фінансів)	3	4	3
10. Практична здійсненність (необхідність нових матеріалів)	4	5	5
11. Практична здійсненність (термін реалізації)	5	4	4
12. Практична здійсненність (розробка документів)	4	5	4
Сума балів	37	41	37
Середньоарифметична сума балів $СБ_c$	38,3		

За результатами розрахунків, наведених в табл. 4.2, зробимо висновок щодо науково-технічного рівня і рівня комерційного потенціалу розробки. При цьому використаємо рекомендації, наведені в табл. 4.3 [44].

Таблиця 4.3 – Науково-технічні рівні та комерційні потенціали розробки

Середньоарифметична сума балів СБ ,розрахована на основі висновків експертів	Науково-технічний рівень та комерційний потенціал розробки
41…48	Високий
31…40	Вище середнього
21…30	Середній
11…20	Нижче середнього
0…10	Низький

Згідно проведених досліджень рівень комерційного потенціалу розробки за темою «Система управління безпекою та подіями. Частина 1. Метод та програмний засіб для управління безпекою та подіями» становить 38,3 бала, що, відповідно до табл. 4.3, свідчить про комерційну важливість проведення даних досліджень (рівень комерційного потенціалу розробки вище середнього).

4.2 Оцінювання рівня новизни розробки

Виводячи на ринок новинку розробник вважає, що тієї новизни, якою наділена нова розробка є достатньо для того, щоб вона була сприйнята споживачем як нова. Але це не завжди так, в силу того, що споживач і розробник неоднозначно визначать її рівень новизни. Тому доцільним є визначення рівня новизни розробки отриманої в результаті досліджень за темою «Система управління безпекою та подіями. Частина 1. Метод та програмний засіб для управління безпекою та подіями».

Саме визначення рівня і ступеня інтегральної новизни є найбільш актуальним, оскільки її рівень визначає ступінь однакового позитивного сприйняття новизни розробки як виробником, так і споживачем, а отже і ринком в цілому, а це, у свою чергу, є гарантією того, що новинка знайде своє місце на ринку, користуватиметься попитом у споживачів і забезпечить відшкодування витрат, зазнаних товаровиробником під час розроблення та виробництва технічної розробки [45].

Рівень новизни нової продукції розраховуємо експертним методом шляхом протиставлення нової продукції та її аналогів, що існують в даний час на ринку, за чинниками що визначають її значення, в системі «краще-гірше». Рівень новизни встановлюємо відносно рівня аналога (або продукту, що досить близький до аналога).

Для визначення *i*-го виду новизни, застосуємо чинники, які впливають на її рівень. Кожен чинник *i*-го виду новизни розраховуємо в балах. Більша кількість набраних балів свідчить про більший рівень новизни. Для оцінювання рівня новизни використаємо думки експертів, які встановлюють визначені бали відповідним чинникам. Бал відповідності проставляється в діапазоні від (-5 – значно гірше аналога до +5 – значно краще аналога). Результати попереднього оцінювання зведемо до відповідного листа оцінювання (табл. 4.4).

Таблиця 4.4 – Лист оцінювання рівня новизни експертами

Види та чинники		Бали та експерти		
		Експерт 1	Експерт 2	Експерт 3
1		2	3	4
Споживча новизна	Питома вага 0,232	Максимальний бал $Б_{i\ MAX}$		25
1. Зміна поведінкових звичок споживача		4	4	4
2. Ступінь задоволення потреб і запитів		5	5	5
3. Спосіб задоволення потреби		4	4	4
4. Формування нової потреби		4	4	4
5. Формування нового споживача		1	1	1
Середній бал експертів $Б_{i\ отр}$		18		
Товарна новизна	Питома вага 0,21	Максимальний бал $Б_{i\ MAX}$		30
1. Параметричні зміни показників продукції				
1.1. Якісні		4	4	4
1.2. Технічні		3	4	3
1.3. Економічні		3	3	3
1.4. Сервісні		4	4	4
2. Якість продукції по відношенню до конкурентів		3	3	3
3. Функціональні зміни		4	4	4
Середній бал експертів $Б_{i\ отр}$		21		
Виробнича новизна	Питома вага 0,042	Максимальний бал $Б_{i\ MAX}$		25
1. Рівень унікальності товару для підприємства		5	5	5
2. Рівень унікальності для галузі		3	3	3
3. Рівень унікальності товару для країни		1	1	1
4. Зміна виробничої системи		4	4	4
5. Відносно існуючого асортименту		3	2	3
Середній бал експертів $Б_{i\ отр}$		16		
Прогресивна новизна	Питома вага 0,2	Максимальний бал $Б_{i\ MAX}$		25
1. Зміна технології виготовлення		4	4	4
2. Рівень застосування нових компонентів і матеріалів		2	2	2
3. Зміна технологічного принципу дії виробу		1	2	1
4. Зміна конструктивного виконання		3	3	3
5. Рівень застосування інновацій		2	2	2
Середній бал експертів $Б_{i\ отр}$		12		
Ринкова новизна	Питома вага 0,1	Максимальний бал $Б_{i\ MAX}$		20
1. Новий виріб на новому ринку		0	0	0
2. Новий виріб на відомому ринку		4	4	4
3. Модернізований виріб		2	2	2
4. Нова модель		3	3	3
Середній бал експертів $Б_{i\ отр}$		9		

Продовження таблиці 4.4

1		2	3	4
Екологічна новизна	Питома вага 0,035	Максимальний бал $Б_{i\ MAX}$		20
1. Рівень екологічної чистоти технології виробництва		5	5	5
2. Рівень впровадження мало- та безвідходних технологій		5	5	5
3. Рівень екологічно небезпечних режимів експлуатації продукції		5	5	5
4. Рівень забруднення навколишнього середовища		5	5	5
Середній бал експертів $Б_{i\ отр}$		20		
Соціальна новизна	Питома вага 0,036	Максимальний бал $Б_{i\ MAX}$		20
1. Використання нового товару приводить до покращення стану здоров'я нації		0	0	0
2. Використання нового товару приводить до зростання доходів населення		0	0	0
3. Виробництво нового товару приводить до збільшення (зменшення) кількості робочих місць на підприємстві		4	5	4
4. Виробництво нового товару приводить до підвищення кваліфікації персоналу		3	3	3
Середній бал експертів $Б_{i\ отр}$		7		
Маркетингова новизна	Питома вага 0,145	Максимальний бал $Б_{i\ MAX}$		20
1. Нові методи маркетингових досліджень		0	0	0
2. Вживання нових стратегій сегментації ринку		3	3	3
3. Вибір нової маркетингової стратегії обхвату і розвитку цільового сегмента		1	1	1
4. Побудова нових каналів збуту		2	1	1
Середній бал експертів $Б_{i\ отр}$		5		

Значення *i*-го виду новизни розрахуємо за формулою [45]:

$$I_i = \frac{Б_{i\,отр}}{Б_{i\,MAX}}, \tag{4.1}$$

де $Б_{i\ отр}$ – отримана кількість балів за шкалою оцінок чинників, що визначають *i*-й вид новизни;

$Б_{i\ MAX}$ – максимальна кількість балів, що може бути отримана за *i*-м видом новизни.

Загальний рівень інтегральної новизни розраховуємо шляхом перемноження отриманого значення *i*-го виду новизни на її вагомість, причому вагомість *i*-го виду новизни визначаємо експертним методом, за формулою [45]:

$$N_{інт} = \sum_{i}^{n} W_i \cdot I_i, \tag{4.2}$$

де $N_{інт}$ – рівень інтегральної (сукупної) новизни;

W_i – вагомість (питома вага) i-го виду новизни;

n – загальна кількість видів новизни.

$$N_{інт} = (0{,}232 \cdot 18/25)+(0{,}21 \cdot 21/30)+(0{,}042 \cdot 16/25)+(0{,}2 \cdot 12/25)+(0{,}1 \cdot 9/20)+$$
$$(0{,}035 \cdot 20/20)+(0{,}036 \cdot 7/20)+(0{,}145 \cdot 5/20) = 0{,}573.$$

Отримане значення інтегрального рівня новизни зіставляємо зі шкалою, що наведена в табл. 4.5 [44].

Таблиця 4.5 – Рівні новизни нового товару та їхня характеристика

Рівні новизни товару	Значення інтегральної новизни	Характеристика товару	Вид нового товару
Найвища	1,00	Абсолютно новий товар	Новий товар, що наділений ознаками інноваційності (інноваційний товар)
Висока	0,8…0,99	Товар, який не має аналогів	
Значуща	0,6…0,79	Принципова зміна споживчих властивостей товару	
Достатня	0,4…0,59	Принципова технологічна модифікація товару	
Незначна	0,2…0,39	Кардинальна зміна параметрів	Новий товар
Помилкова	0,00…0,19	Малоістотна модифікація	

Згідно табл. 4.5 розробка відповідає рівню при значенні інтегральної новизни 0,573 – достатня новизна; за характеристикою: принципова технологічна модифікація товару; вид розробки – новий товар, що наділений ознаками інноваційності (інноваційний товар).

4.3 Розрахунок витрат на проведення науково-дослідної роботи

Витрати, пов’язані з проведенням науково-дослідної роботи на тему «Система управління безпекою та подіями. Частина 1. Метод та програмний засіб для управління безпекою та подіями», під час планування, обліку і калькулювання собівартості науково-дослідної роботи групуємо за відповідними статтями.

4.3.1 Витрати на оплату праці

До статті «Витрати на оплату праці» належать витрати на виплату основної та додаткової заробітної плати керівникам відділів, лабораторій, секторів і груп, науковим, інженерно-технічним працівникам, конструкторам, технологам,

креслярам, копіювальникам, лаборантам, робітникам, студентам, аспірантам та іншим працівникам, безпосередньо зайнятим виконанням конкретної теми, обчисленої за посадовими окладами, відрядними розцінками, тарифними ставками згідно з чинними в організаціях системами оплати праці.

Основна заробітна плата дослідників. Витрати на основну заробітну плату дослідників ($З_о$) розраховуємо у відповідності до посадових окладів працівників, за формулою [44]:

$$З_о = \sum_{i=1}^{k} \frac{M_{ni} \cdot t_i}{T_p}, \quad (4.3)$$

де k – кількість посад дослідників залучених до процесу досліджень;

M_{ni} – місячний посадовий оклад конкретного дослідника, грн;

t_i – число днів роботи конкретного дослідника, дн.;

T_p – середнє число робочих днів в місяці, T_p=21 дні.

$$З_о = 17000{,}00 \cdot 31 / 21 = 25095{,}24 \; грн.$$

Проведені розрахунки зведемо до табл. 4.6.

Таблиця 4.6 – Витрати на заробітну плату дослідників

Найменування посади	Місячний посадовий оклад, грн	Оплата за робочий день, грн	Число днів роботи	Витрати на заробітну плату, грн
Керівник проекту	17000,00	809,52	31	25095,24
Інженер-розробник програмного забезпечення	16500,00	785,71	31	24357,14
Науковий консультант з проблем управління безпекою	16450,00	783,33	6	4700,00
Лаборант	6800,00	323,81	15	4857,14
Всього				59009,52

Основна заробітна плата робітників. Витрати на основну заробітну плату робітників ($З_р$) за відповідними найменуваннями робіт НДР на тему «Система управління безпекою та подіями. Частина 1. Метод та програмний засіб для управління безпекою та подіями» розраховуємо за формулою:

$$З_p = \sum_{i=1}^{n} C_i \cdot t_i \,, \qquad (4.4)$$

де C_i – погодинна тарифна ставка робітника відповідного розряду, за виконану відповідну роботу, грн/год;

t_i – час роботи робітника при виконанні визначеної роботи, год.

Погодинну тарифну ставку робітника відповідного розряду C_i можна визначити за формулою:

$$C_i = \frac{М_М \cdot K_i \cdot K_c}{T_p \cdot t_{зм}} \,, \qquad (4.5)$$

де $М_М$ – розмір прожиткового мінімуму працездатної особи, або мінімальної місячної заробітної плати (в залежності від діючого законодавства), приймемо *Мм*=6700,00 грн;

K_i – коефіцієнт міжкваліфікаційного співвідношення для встановлення тарифної ставки робітнику відповідного розряду (табл. Б.2, додаток Б) [44];

K_c – мінімальний коефіцієнт співвідношень місячних тарифних ставок робітників першого розряду з нормальними умовами праці виробничих об'єднань і підприємств до законодавчо встановленого розміру мінімальної заробітної плати.

T_p – середнє число робочих днів в місяці, приблизно T_p = 21 дн;

$t_{зм}$ – тривалість зміни, год.

$C_1 = 6700{,}00 \cdot 1{,}10 \cdot 1{,}65 / (21 \cdot 8) = 72{,}38$ *грн.*

$З_{р1} = 72{,}38 \cdot 7{,}50 = 542{,}88$ *грн.*

Таблиця 4.7 – Величина витрат на основну заробітну плату робітників

Найменування робіт	Тривалість роботи, год	Розряд роботи	Тарифний коефіцієнт	Погодинна тарифна ставка, грн	Величина оплати на робітника грн
1	2	3	4	5	6
Підготовка робочого місця інженера-розробника програм-ного забезпечення	7,50	2	1,10	72,38	542,88
Інсталяція програм-ного забезпечення середовища розробки	5,65	3	1,35	88,83	501,92

Продовження таблиці 4.7

1	2	3	4	5	6
Формування кодів програмних блоків управління безпекою та подіями	12,00	5	1,70	111,87	1342,39
Формування бази даних дослідження	20,00	2	1,10	72,38	1447,68
Контроль проходження програмних експериментів	6,00	4	1,50	98,71	592,23
Всього					4427,10

Додаткова заробітна плата дослідників та робітників. Додаткову заробітну плату розраховуємо як 10 … 12% від суми основної заробітної плати дослідників та робітників за формулою:

$$З_{дод} = (З_о + З_р) \cdot \frac{Н_{дод}}{100\%}, \quad (4.6)$$

де $Н_{дод}$ – норма нарахування додаткової заробітної плати. Приймемо 10%.

$З_{дод}$ = (59009,52 + 4427,10) · 10 / 100% = 6343,66 грн.

4.3.2 Відрахування на соціальні заходи

Нарахування на заробітну плату дослідників та робітників розраховуємо як 22% від суми основної та додаткової заробітної плати дослідників і робітників за формулою:

$$З_н = (З_о + З_р + З_{дод}) \cdot \frac{Н_{зп}}{100\%} \quad (4.7)$$

де $Н_{зп}$ – норма нарахування на заробітну плату. Приймаємо 22%.

Зн = (59009,52 + 4427,10 + 6343,66) · 22 / 100% = 15351,66 грн.

4.3.3 Сировина та матеріали

До статті «Сировина та матеріали» належать витрати на сировину, основні та допоміжні матеріали, інструменти, пристрої та інші засоби і предмети праці, які придбані у сторонніх підприємств, установ і організацій та витрачені на проведення досліджень за темою «Система управління безпекою та подіями. Частина 1. Метод та програмний засіб для управління безпекою та подіями».

Витрати на матеріали (*М*), у вартісному вираженні розраховуються окремо по кожному виду матеріалів за формулою:

$$M = \sum_{j=1}^{n} H_j \cdot Ц_j \cdot К_j - \sum_{j=1}^{n} B_j \cdot Ц_{вj} , \qquad (4.8)$$

де H_j – норма витрат матеріалу *j*-го найменування, кг;

n – кількість видів матеріалів;

$Ц_j$ – вартість матеріалу *j*-го найменування, грн/кг;

$К_j$ – коефіцієнт транспортних витрат, ($К_j$ = *1,1 ... 1,15*);

B_j – маса відходів *j*-го найменування, кг;

$Ц_{вj}$ – вартість відходів *j*-го найменування, грн/кг.

$M_1 = 3,0 \cdot 210,00 \cdot 1,1 - 0 \cdot 0 = 693,00$ *грн.*

Проведені розрахунки зведемо до табл. 4.8.

Таблиця 4.8 – Витрати на матеріали

Найменування матеріалу, марка, тип, сорт	Ціна за 1 кг, грн	Норма витрат, кг	Величина відходів, кг	Ціна відходів, грн/кг	Вартість витраченого матеріалу, грн
1	2	3	4	5	6
Папір А4 500 аркушів клас-С Crystal Print&Copy UPM	210,00	3,0	0	0	693,00
Папір офісний Офіс Центр А5 80г/м2 500 аркушів клас С	129,00	4,0	0	0	567,60
Прибор настільний 13 предметів 6300-01 Buromax чорний	220,00	3,0	0	0	726,00
Набір канцелярський офісний FAX	210,00	3,0	0	0	693,00
Картридж для принтера	1100,00	1,0	0	0	1210,00
Диск оптичний CD-RW	22,00	3,0	0	0	72,60
USB флеш накопичувач Transcend 64Gb JetFlash 700 (TS64GJF700)	279,00	1,0	0	0	306,90
Всього					4269,10

Після розрахунку витрат на сировину та матеріали можна переходити до розрахунку витрат на комплектуючі.

4.3.4 Розрахунок витрат на комплектуючі

Витрати на комплектуючі *($К_в$)*, які використовують при проведенні НДР на тему «Система управління безпекою та подіями. Частина 1. Метод та програмний засіб для управління безпекою та подіями» відсутні.

4.3.5 Спецустаткування для наукових (експериментальних) робіт

До статті «Спецустаткування для наукових (експериментальних) робіт» належать витрати на виготовлення та придбання спецустаткування необхідного для проведення досліджень, також витрати на їх проектування, виготовлення, транспортування, монтаж та встановлення.

Балансову вартість спецустаткування розраховуємо за формулою:

$$B_{спец} = \sum_{i=1}^{k} Ц_i \cdot C_{пр.i} \cdot K_i \ , \qquad (4.9)$$

де $Ц_i$ – ціна придбання одиниці спецустаткування даного виду, марки, грн;

$C_{пр.i}$ –кількість одиниць устаткування відповідного найменування, які придбані для проведення досліджень, шт.;

K_i – коефіцієнт, що враховує доставку, монтаж, налагодження устаткування тощо, ($K_i = 1,10…1,12$);

k – кількість найменувань устаткування.

$B_{спец} = 9850,00 \cdot 1 \cdot 1,1 = 10835,00$ *грн.*

Отримані результати зведемо до табл. 4.9.

Таблиця 4.9 – Витрати на придбання спецустаткування по кожному виду

Найменування устаткування	Кількість, шт	Ціна за одиницю, грн	Вартість, грн
Відеокарта GeForce RTX 3060	1	9850,00	10835,00
Зовнішній жорсткий диск 2.5" 1TB Seagate	1	1890,00	2079,00
Відеопам'ять об'ємом 12 ГБ	1	2650,00	2915,00
Всього			15829,00

Після розрахунку витрат на спецустаткування для наукових робіт можна переходити до розрахунку витрат на програмне забезпечення для наукових (експериментальних) робіт.

4.3.6 Програмне забезпечення для наукових (експериментальних) робіт

До статті «Програмне забезпечення для наукових (експериментальних) робіт» належать витрати на розробку та придбання спеціальних програмних засобів і програмного забезпечення, (програм, алгоритмів, баз даних) необхідних для проведення досліджень, також витрати на їх проектування, формування та встановлення.

Балансову вартість програмного забезпечення розраховуємо за формулою:

$$B_{прг} = \sum_{i=1}^{k} Ц_{iпрг} \cdot C_{прг.i} \cdot K_i \ , \qquad (4.10)$$

де $Ц_{iпрг}$ – ціна придбання одиниці програмного засобу даного виду, грн;

$C_{прг.i}$ – кількість одиниць програмного забезпечення відповідного найменування, які придбані для проведення досліджень, шт.;

K_i – коефіцієнт, що враховує інсталяцію, налагодження програмного засобу тощо, ($K_i = 1,10…1,12$);

k – кількість найменувань програмних засобів.

$B_{прг} = 7860,00 \cdot 1 \cdot 1,1 = 8646,00$ *грн.*

Отримані результати зведемо до табл. 4.10.

Таблиця 4.10 – Витрати на придбання програмних засобів по кожному виду

Найменування програмного засобу	Кількість, шт	Ціна за одиницю, грн	Вартість, грн
Прикладне програмне забезпечення розробки	1	7860,00	8646,00
Середовище програмування Visual Studio Community	1	900,00	990,00
Всього			9636,00

Після розрахунку витрат на сировину та матеріали можна переходити до розрахунку витрат на комплектуючі.

4.3.7 Амортизація обладнання, програмних засобів та приміщень

В спрощеному вигляді амортизаційні відрахування по кожному виду обладнання, приміщень та програмному забезпеченню тощо, розраховуємо з використанням прямолінійного методу амортизації за формулою:

$$A_{обл} = \frac{Ц_б}{T_в} \cdot \frac{t_{вик}}{12}, \quad (4.11)$$

де $Ц_б$ – балансова вартість обладнання, програмних засобів, приміщень тощо, які використовувались для проведення досліджень, грн;

$t_{вик}$ – термін використання обладнання, програмних засобів, приміщень під час досліджень, місяців;

$T_в$ – строк корисного використання обладнання, програмних засобів, приміщень тощо, років.

$A_{обл}$ = (40284,00 · 2) / (2 · 12) = 3357,00 грн.

Проведені розрахунки зведемо до табл. 4.11.

Таблиця 4.11 – Амортизаційні відрахування по кожному виду обладнання

Найменування обладнання	Балансова вартість, грн	Строк корисного використання, років	Термін використання обладнання, місяців	Амортизаційні відрахування, грн
Персональний комп'ютер проведення розробки та моделювання Комп'ютер Vinga Wolverine A5003 (I5M16G3060T.A5003)	40284,00	2	2	3357,00
Робоче місце інженера-розробника ПЗ	8560,00	5	2	285,33
Пристрої передачі даних (роутер безпровідний)	6750,00	4	2	281,25
Пристрій виводу інформації	6840,00	5	2	228,00
Оргтехніка	7250,00	4	2	302,08
Приміщення лабораторії	620400,00	20	2	5170,00
ОС Windows 10	8370,00	2	2	697,50
Прикладний пакет Microsoft Office 2016	7825,00	2	2	652,08
Всього				10973,25

Після розрахунку витрат на амортизацію обладнання, програмних засобів та приміщень можна переходити до розрахунку витрат на паливо та енергію для науково-виробничих цілей.

4.3.8 Паливо та енергія для науково-виробничих цілей

Витрати на силову електроенергію (B_e) розраховуємо за формулою:

$$B_e = \sum_{i=1}^{n} \frac{W_{yi} \cdot t_i \cdot Ц_e \cdot K_{вni}}{\eta_i}, \quad (4.12)$$

де W_{yi} – встановлена потужність обладнання на визначеному етапі розробки, кВт;

t_i – тривалість роботи обладнання на етапі дослідження, год;

$Ц_e$ – вартість 1 кВт-години електроенергії, грн; (вартість електроенергії визначається за даними енергопостачальної компанії), приймемо Ц$_e$ = 6,20 грн;

$K_{вni}$ – коефіцієнт, що враховує використання потужності, $K_{вni} < 1$;

η_i – коефіцієнт корисної дії обладнання, $\eta_i < 1$.

$B_e = 0,32 \cdot 240,0 \cdot 6,20 \cdot 0,95 / 0,97 = 476,16$ *грн.*

Проведені розрахунки зведемо до табл. 4.12.

Таблиця 4.12 – Витрати на електроенергію

Найменування обладнання	Встановлена потужність, кВт	Тривалість роботи, год	Сума, грн
1	2	3	4
Персональний комп'ютер проведення розробки та моделювання Комп'ютер Vinga Wolverine A5003	0,32	240,0	476,16
Робоче місце інженера-розробника ПЗ	0,15	240,0	223,20
Пристрої передачі даних	0,01	600,0	37,20
Пристрій виводу інформації	0,50	11,0	34,10
Оргтехніка	0,62	4,5	17,30
Всього			787,96

Після розрахунку витрат на паливо та енергію для науково-виробничих цілей можна переходити до розрахунку витрат на службові відрядження.

4.3.9 Службові відрядження

До статті «Службові відрядження» дослідної роботи на тему «Система управління безпекою та подіями. Частина 1. Метод та програмний засіб для управління безпекою та подіями» належать витрати на відрядження штатних працівників, працівників організацій, які працюють за договорами цивільно-правового характеру, аспірантів, зайнятих розробленням досліджень, відрядження, пов'язані з проведенням випробувань машин та приладів, а також витрати на відрядження на наукові з'їзди, конференції, наради, пов'язані з виконанням конкретних досліджень.

Витрати за статтею «Службові відрядження» розраховуємо як 20…25% від суми основної заробітної плати дослідників та робітників за формулою:

$$B_{св} = (З_о + З_р) \cdot \frac{Н_{св}}{100\%}, \quad (4.13)$$

де $Н_{св}$ – норма нарахування за статтею «Службові відрядження», приймемо $Н_{св}$ = 21%.

$B_{св}$ = (59009,52 + 4427,10) · 21 / 100% = 13321,69 грн.

4.3.10 Витрати на роботи, які виконують сторонні підприємства, установи і організації

Витрати за статтею «Витрати на роботи, які виконують сторонні підприємства, установи і організації» розраховуємо як 30…45% від суми основної заробітної плати дослідників та робітників за формулою:

$$B_{сп} = (З_о + З_р) \cdot \frac{Н_{сп}}{100\%}, \quad (4.14)$$

де $Н_{сп}$ – норма нарахування за статтею «Витрати на роботи, які виконують сторонні підприємства, установи і організації», приймемо $Н_{сп}$= 30%.

$B_{сп}$ = (59009,52 + 4427,10) · 30 / 100% = 19030,99 грн.

4.3.11 Інші витрати

До статті «Інші витрати» належать витрати, які не знайшли відображення у зазначених статтях витрат і можуть бути віднесені безпосередньо на собівартість досліджень за прямими ознаками.

Витрати за статтею «Інші витрати» розраховуємо як 50…100% від суми основної заробітної плати дослідників та робітників за формулою:

$$I_в = (З_о + З_р) \cdot \frac{Н_{ів}}{100\%}, \quad (4.15)$$

де $Н_{ів}$ – норма нарахування за статтею «Інші витрати», приймемо $Н_{ів}$ = 60%.

$I_в$ = (59009,52 + 4427,10) · 60 / 100% = 38061,97 грн.

4.3.12 Накладні (загальновиробничі) витрати

Витрати за статтею «Накладні (загальновиробничі) витрати» розраховуємо як 100…150% від суми основної заробітної плати дослідників та робітників за формулою:

$$В_{нзв} = (З_о + З_р) \cdot \frac{Н_{нзв}}{100\%}, \quad (4.16)$$

де $Н_{нзв}$ – норма нарахування за статтею «Накладні (загальновиробничі) витрати», приймемо $Н_{нзв}$ = 105%.

$В_{нзв}$ = (59009,52 + 4427,10) · 105 / 100% = 66608,45 грн.

Витрати на проведення науково-дослідної роботи на тему «Система управління безпекою та подіями. Частина 1. Метод та програмний засіб для управління безпекою та подіями» розраховуємо як суму всіх попередніх статей витрат за формулою:

$$В_{заг} = З_о + З_р + З_{дод} + З_н + М + К_в + В_{спец} + В_{прг} + А_{обл} + В_е + В_{св} + В_{сп} + І_в + В_{нзв}. \quad (4.17)$$

$В_{заг}$= 59009,52 +4427,10 +6343,66 +15351,66291 +4269,10 +0,00 +15829,00 + 9636,00 + 10973,25 +787,96 +13321,69 +19030,99 +38061,97 +66608,45 = =263650,36 грн.

Загальні витрати *ЗВ* на завершення науково-дослідної (науково-технічної) роботи та оформлення її результатів розраховується за формулою:

$$ЗВ = \frac{В_{заг}}{\eta}, \quad (4.18)$$

де η – коефіцієнт, який характеризує етап (стадію) виконання науково-дослідної роботи, приймемо η =0,9.

ЗВ = 263650,36 / 0,9 = 292944,85 грн.

4.4 Розрахунок економічної ефективності науково-технічної розробки при її можливій комерціалізації потенційним інвестором

Результати дослідження проведені за темою «Система управління безпекою та подіями. Частина 1. Метод та програмний засіб для управління безпекою та подіями» передбачають комерціалізацію протягом 4-х років реалізації на ринку.

В цьому випадку майбутній економічний ефект буде формуватися на основі таких даних:

ΔN – збільшення кількості споживачів продукту, у періоди часу, що аналізуються, від покращення його певних характеристик;

Таблиця 4.13 – Вигляд збільшення споживачів за показником

Показник	1-й рік	2-й рік	3-й рік	4-й рік
Збільшення кількості споживачів, осіб	500	850	950	800

N – кількість споживачів які використовували аналогічний продукт у році до впровадження результатів нової науково-технічної розробки, приймемо 7500 осіб;

$Ц_б$ – вартість програмного продукту у році до впровадження результатів розробки, приймемо 8500,00 грн;

$\pm\Delta Ц_о$ – зміна вартості програмного продукту від впровадження результатів науково-технічної розробки, приймемо 1250,25 грн.

Можливе збільшення чистого прибутку у потенційного інвестора $\Delta П_i$ для кожного із 4-х років, протягом яких очікується отримання позитивних результатів від

можливого впровадження та комерціалізації науково-технічної розробки, розраховуємо за формулою [44]:

$$\Delta П_i = (\pm \Delta Ц_o \cdot N + Ц_o \cdot \Delta N)_i \cdot \lambda \cdot \rho \cdot (1 - \frac{\vartheta}{100}), \quad (4.19)$$

де λ – коефіцієнт, який враховує сплату потенційним інвестором податку на додану вартість. У 2022 році ставка податку на додану вартість складає 20%, а коефіцієнт $\lambda = 0{,}8333$;

ρ – коефіцієнт, який враховує рентабельність інноваційного продукту). Приймемо ρ =30%;

ϑ – ставка податку на прибуток, який має сплачувати потенційний інвестор, у 2022 році $\vartheta = 18\%$;

Збільшення чистого прибутку 1-го року:

$\Delta П_1$ =(1250,25·7500,00+9750,25·500)·0,83·0,3·(1-0,18/100%)=2909973,36 грн.

Збільшення чистого прибутку 2-го року:

$\Delta П_2$ =(1250,25·7500,00+9750,25·1350)·0,83·0,3·(1-0,18/100%)=4602158,50 грн.

Збільшення чистого прибутку 3-го року:

$\Delta П_3$ =(1250,25·7500,00+9750,25·2300)·0,83·0,3·(1-0,18/100%)=6493424,24 грн.

Збільшення чистого прибутку 4-го року:

$\Delta П_4$ =(1250,25·7500,00+9750,25·3100)·0,83·0,3·(1-0,18/100%)=8086069,08 грн.

Приведена вартість збільшення всіх чистих прибутків *ПП*, що їх може отримати потенційний інвестор від можливого впровадження та комерціалізації науково-технічної розробки:

$$ПП = \sum_{i=1}^{T} \frac{\Delta П_i}{(1+\tau)^t}, \quad (4.20)$$

$ПП$ =2909973,36/(1+0,24)^1+4602158,50/(1+0,24)^2+6493424,24/(1+0,24)^3+

+8086069,08/(1+0,24)^4=2346752,71+2993079,15+3405718,30+3420193,73=

=12165743,88 грн.

Величина початкових інвестицій PV, які потенційний інвестор має вкласти для впровадження і комерціалізації науково-технічної розробки:

де $k_{інв}$ – коефіцієнт, що враховує витрати інвестора на впровадження науково-технічної розробки та її комерціалізацію, приймаємо $k_{інв}$=2,3;

$ЗВ$ – загальні витрати на проведення науково-технічної розробки та оформлення її результатів, приймаємо 292944,85 грн.

$PV = k_{інв} \cdot ЗВ$ = 2,3 · 292944,85 = 673773,15 грн.

Абсолютний економічний ефект $E_{абс}$ для потенційного інвестора від можливого впровадження та комерціалізації науково-технічної розробки становитиме:

$$E_{абс} = ПП - PV \tag{4.22}$$

Внутрішня економічна дохідність інвестицій $E_в$, які можуть бути вкладені потенційним інвестором у впровадження та комерціалізацію науково-технічної розробки:

$$E_в = \sqrt[Tж]{1 + \frac{E_{абс}}{PV}} - 1, \tag{4.23}$$

де $E_{абс}$ – абсолютний економічний ефект вкладених інвестицій, 11491970,73 грн;

PV – теперішня вартість початкових інвестицій, 673773,15 грн;

$T_ж$ – життєвий цикл науково-технічної розробки, тобто час від початку її розробки до закінчення отримування позитивних результатів від її впровадження, 4 роки.

$E_в = \sqrt[Tж]{1 + \frac{E_{абс}}{PV}} - 1$ = (1+11491970,73/673773,15)$^{1/4}$= 1,06.

Мінімальна внутрішня економічна дохідність вкладених інвестицій $\tau_{мін}$:

$$\tau_{мін} = d + f, \tag{4.24}$$

де d – середньозважена ставка за депозитними операціями в комерційних банках; в 2022 році в Україні d =0,12;

f – показник, що характеризує ризикованість вкладення інвестицій, приймемо 0,3.

$\tau_{мін}$ = 0,12+0,3 = 0,42 < 1,06 свідчить про те, що внутрішня економічна дохідність інвестицій $E_в$, які можуть бути вкладені потенційним інвестором у впровадження та комерціалізацію науково-технічної розробки вища мінімальної внутрішньої дохідності. Тобто інвестувати в науково-дослідну роботу за темою «Система управління безпекою та подіями. Частина 1. Метод та програмний засіб для управління безпекою та подіями» доцільно.

Період окупності інвестицій $T_{ок}$ які можуть бути вкладені потенційним інвестором у впровадження та комерціалізацію науково-технічної розробки:

$$T_{ок} = \frac{1}{E_в}, \tag{4.25}$$

$T_{ок}$ = 1 / 1,06 = 0,94 р.

$T_{ок}$ < 3-х років, що свідчить про комерційну привабливість науково-технічної розробки і може спонукати потенційного інвестора профінансувати впровадження даної розробки та виведення її на ринок.

Висновки до розділу. Згідно проведених досліджень рівень комерційного потенціалу розробки за темою «Система управління безпекою та подіями. Частина 1. Метод та програмний засіб для управління безпекою та подіями» становить 38,3 бала, що, свідчить про комерційну важливість проведення даних досліджень (рівень комерційного потенціалу розробки вище середнього).

Загальні витрати на завершення науково-дослідної (науково-технічної) роботи та оформлення її результатів складає 292944,85 грн. Також термін окупності становить 0,94 р., що менше 3-х років, що свідчить про комерційну привабливість науково-технічної розробки і може спонукати потенційного інвестора профінансувати впровадження даної розробки та виведення її на ринок.

Отже можна зробити висновок про доцільність проведення науково-дослідної роботи за темою «Система управління безпекою та подіями. Частина 1. Метод та програмний засіб для управління безпекою та подіями».

ВИСНОВКИ

У ході виконання магістерської кваліфікаційної роботи було виконано усі поставлені задачі. У результаті виконання роботи був проведений аналіз існуючих засобів для управління безпекою і подіями та визначено основні моменти їх функціонування та використання, а саме:

- ArcSight
- Qradar
- McAfee
- LogRhythm

Проаналізовані одні з основних існуючих методів для управління безпекою і подіями. На основі їхніх переваг розроблено новий метод для управління безпекою і подіями. Було проведено дослідження методу на помилки та дослідження його стійкості.

На основі методу було розроблено архітектуру додатку, який виконує функції управління безпекою та подіями. Оброблені усі відомі помилки при користуванні програмним засобом та проаналізовано його відповідність поставленим завданням. Розроблений програмний засіб огорнуто в графічний інтерфейс для забезпечення зручності при користуванні.

Проведено тестування програмного засобу на існування помилок та вразливостей. Визначено, що помилки у програмному засобі обробляються за допомогою аналізу отриманих програмним засобом даних, у разі чого користувач буде повідомлений при їх виявленні. У ході тестування було з'ясовано, що програмний засіб працює правильно і відповідно до поставлених завдань. При роботі програмного засобу помилок не було виявлено.

Було проведено комерційних аудит розробки, розраховано узагальнений коефіцієнт якості розробки, проаналізовані витрати, яких потребує розробка програмного засобу. Після аналізу всіх витрати, було пораховано, що окупність розробленої системи становить ~0,94 року, що менше трьох років і це дає змогу зробити висновок, що розробка програмного засобу є доцільною.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. WaterISAC 15 Cybersecurity Fundamentals for Water and Wastewater Utilities. Best Practices to reduce Exploitable Weaknesses and Attacks. WaterIsac. URL: https://www.waterisac.org/system/files/articles/15%20Cybersecurity%20Fundamentals%20%28WaterISAC%29.pdf (дата звернення: 05.10.2022).
2. CyberX NIST Recommendations for IoT & ICS Security. An Executive Summary. CyberX Labs. URL: https://cyberx-labs.com/resources/nist-recommendations-for-iot-ics-security (дата звернення: 06.10.2022).
3. Security Information and Event Management (SIEM) Implementation. McGraw-Hill Higher Education. URL: https://scholar.google.com/scholar_lookup?title=Security+Information+and+Event+Management+(SIEM)+Implementation&author=D.+Miller&author=S.+Harris&author=A.+Harper&author=S.+Van+Dyke&author=C.+Blask&publication_year=2010 (дата звернення: 06.10.2022).
4. New types of Alert Correlation for Security Information and Event Management Systems; Proceedings of the 8th International Conference on New Technologies, Mobility and Security, NTMS. IEEE Xplore. URL: https://ieeexplore.ieee.org/abstract/document/7792462 (дата звернення: 07.10.2022).
5. Bryant B.D., Saiedian H. Improving SIEM alert metadata aggregation with a novel kill-chain based classification model. Science Direct. URL: https://www.sciencedirect.com/ science/article/pii/ S016740482030095X (дата звернення: 08.10.2022).
6. Nicolett M., Kavanagh K.M. Magic Quadrant for Security Information and Event Management, Gartner Technical Report: DocPlayer. URL: http://docplayer.net/2407833-Magic-quadrant-for-security-information-and-event-management.html (дата звернення: 11.10.2022).
7. Nicolett M., Kavanagh K.M. Magic Quadrant for Security Information and Event Management, Gartner Technical Report. Novell. URL: https://www.novell.com/docrep/documents/yuufbom4u2/gartner_magic_quadrant_siem_report_may2011.pdf. (дата звернення: 13.10.2022).
8. Nicolett M., Kavanagh K.M. Magic Quadrant for Security Information and Event Management, Gartner Technical Report. Bank Info Security. URL: https://www.bankinfosecurity.com/ whitepapers/2012-gartner-magic-quadrant-for-siem-w-602 (дата звернення: 15.10.2022).
9. Nicolett M., Kavanagh K.M. Magic Quadrant for Security Information and Event Management, Gartner Technical Report. Gartner. URL: https://www.gartner.com/en/ documents/2477018/magic-quadrant-for-security-information-and-event-manage. (дата звернення: 15.10.2022)

10. Nicolett M., Kavanagh K.M., Rochford O. Magic Quadrant for Security Information and Event Management, Gartner Technical Report. Bw digitronik. URL: https://www.bwdigitronik.ch/application/files/5814/5450/7565/www.gartner.com.com.pdf (дата звернення: 17.10.2022).
11. Kavanagh K.M., Rochford O. Magic Quadrant for Security Information and Event Management, Gartner Technical Report. Gartner. URL: https://www.gartner.com/en/documents/3097022/magic-quadrant-for-security-information-and-event-manage (дата звернення: 17.10.2022).
12. Kavanagh K.M., Rochford O., Bussa T. Magic Quadrant for Security Information and Event Management, Gartner Technical Report. Orange Cyberdefense. URL: https://securelink.net/wp-content/uploads/sites/7/2016-Magic-Quadrant-for-SIEM.pdf (дата звернення: 17.10.2022).
13. Kavanagh K.M., Bussa T. Magic Quadrant for Security Information and Event Management, Gartner Tec hnical Report. Gartner. URL: https://www.gartner.com/ en/documents/3834683/magic-quadrant-for-security-information-and-event-manage (дата звернення: 19.10.2022).
14. Kavanagh K.M., Sadowski T.B.G. Magic Quadrant for Security Information and Event Management, Gartner Technical Report. Gartner. URL: https://virtualizationandstorage.files.wordpress.com/2018/03/magic-quadrant-for-security-information-and-event-3-dec-2018.pdf (дата звернення: 22.10.2022).
15. Kavanagh K.M., Sadowski T.B.G. Magic Quadrant for Security Information and Event Management, Gartner Technical Report. Gartner. URL: https://www.gartner.com/ en/documents/3981040/magic-quadrant-for-security-information-and-event-mana-ge (дата звернення: 22.10.2022).
16. Security Information and Event Management (SIEM): Analysis, Trends, and Usage in Critical Infrastructures. National Library. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/ PMC8309804 (дата звернення: 23.10.2022).
17. Security information and event management (SIEM). Imperva. URL: https://www.imperva.com/learn/application-security/siem (дата звернення: 24.10.2022).
18. SIEM Architecture: Technology, Process and Data. Exabeam. URL: https://www.exabeam.com/explainers/siem/siem-architecture (дата звернення: 25.10.2022).
19. What is security information and event management (SIEM). Tech Target. URL: https:// www.techtarget.com/searchsecurity/definition/security-information-and-event-management-SIEM (дата звернення: 26.10.2022).
20. What is the SIEM process. Comodo. URL: https://www.comodo.com/what-is/the-siem-process.php (дата звернення: 27.10.2022).

21. How SIEM Works, Best Practices for Implementation & More. Digital Guardian. URL: https://digitalguardian.com/blog/what-siem-how-it-works-best-practices-implementation-more (дата звернення: 29.10.2022).
22. How to choose a SIEM suite. Tech Beacon. URL: https://learn.techbeacon.com/units/how-choose-siem-suite (дата звернення: 30.10.2022).
23. How to structure your CSIRT or SOC team. InfoSec. URL: https://resources.infosecinstitute.com/career/how-to-structure-your-csirt-or-soc-team (дата звернення: 01.11.2022).
24. Building Your Incident Response Platform Hub. SecurityIntelligence. URL: https://securityintelligence. com/building-your-incident-response-platform-hub (дата звернення: 02.11.2022).
25. PHP: Hypertext Preprocessor. PHP. URL: https://www.php.net. (дата звернення: 04.11.2022).
26. C# Tutorial. W3Schools. URL: https://www.w3schools.com/cs/index.php (дата звернення: 06.11.2022).
27. Complete Guide to What is XAMPP. EDUCBA. URL: https://www.educba.com/what-is-xampp (дата звернення: 07.11.2022).
28. MongoDB: The Developer Data Platform. MongoDB. URL: https://www.mongodb.com (дата звернення: 09.11.2022).
29. Understanding RoboMongo (Robo 3T): A Comprehensive Guide. Hevo. URL: https://hevodata.com/learn/robomongo (дата звернення: 10.11.2022).
30. The RabbitMQ Management Interface. CloudAMQP. URL: https://www.cloudamqp.com/blog/part3-rabbitmq-for-beginners_the-management-interface.html (дата звернення: 11.11.2022).
31. Rabbitmqctl – command line tool for managing a RabbitMQ. Ubuntu. URL: https://manpages.ubuntu.com/manpages/trusty/man1/rabbitmqctl.1.html (дата звернення: 13.11.2022).
32. Bootstrap Tutorial - An Ultimate Guide for Beginners. TutorialRepublic.URL: https://www. tutorialrepublic.com/twitter-bootstrap-tutorial (дата звернення: 14.11.2022).
33. Code Complete. GoodReads. URL: https://www.goodreads.com/book/show/4845.Code_Complete (дата звернення: 15.11.2022).
34. Code Complete: A Practical Handbook of Software Construction. Academia.edu. URL: https://www.academia.edu/34799610/Code_complete (дата звернення: 15.11.2022).
35. Apache HTTP Server Documentation. Apache. URL: https://httpd.apache.org/docs (дата звернення: 16.11.2022).
36. Documentation: Table of Contents – RabbitMQ. RabbitMQ. URL: https://www.rabbitmq.com/documentation.html (дата звернення: 16.11.2022).

37. MongoDB C#/.NET Driver. MongoDB. URL: https://www.mongodb.com/docs/drivers/csharp (дата звернення: 17.11.2022).
38. Manual – OSSEC Documentation. OSSEC. URL: https://www.ossec.net/docs/manual/index.html (дата звернення: 19.11.2022).
39. NLog Tutorial – The essential guide for logging from C#. Elmah. URL: https://blog.elmah.io/nlog-tutorial-the-essential-guide-for-logging-from-csharp (дата звернення: 20.11.2022).
40. NLog configuration options. NLog. URL: https://nlog-project.org/config (дата звернення: 21.11.2022).
41. Installing the MongoDB PHP Driver on Windows – Manual. PHP. URL: https://www.php.net/manual/en/mongodb.installation.windows.php (дата звернення: 24.11.2022).
42. Using the PHP Library for MongoDB (PHPLIB) – Manual. PHP. URL: https://www.php.net/manual/en/mongodb.tutorial.library.php (дата звернення: 26.11.2022).
43. Creating, Reading, Updating, and Deleting MongoDB Documents with PHP. MongoDB. URL: https://www.mongodb.com/developer/languages/php/php-crud (дата звернення: 28.11.2022).
44. Методичні вказівки до виконання економічної частини магістерських кваліфікаційних робіт / Уклад. : В. О. Козловський, О. Й. Лесько, В. В. Кавецький. – Вінниця : ВНТУ, 2021. – 42 с.
45. Кавецький В. В. Економічне обґрунтування інноваційних рішень: практикум / В. В. Кавецький, В. О. Козловський, І. В. Причепа – Вінниця : ВНТУ, 2016. – 113 с.

ДОДАТКИ

Додаток А. Результат перевірки роботи на плагіат

ПРОТОКОЛ ПЕРЕВІРКИ МАГІСТЕРСЬКОЇ КВАЛІФІКАЦІЙНОЇ РОБОТИ НА НАЯВНІСТЬ ТЕКСТОВИХ ЗАПОЗИЧЕНЬ

Назва роботи: Система управління безпекою та подіями. Частина 1. Метод та програмний засіб для управління безпекою та подіями

Автор роботи: Буняк Віталій Михайлович

Тип роботи: магістерська кваліфікаційна робота

Підрозділ кафедра захисту інформації ФІТКІ
(кафедра, факультет)

Показники звіту подібності Unicheck

Оригінальність – 95,79%. Схожість – 4,21%.

Аналіз звіту подібності (відмітити потрібне):

√ 1. Запозичення, виявлені у роботі, оформлені коректно і не містять ознак плагіату.

□ 2. Виявлені у роботі запозичення не мають ознак плагіату, але їх надмірна кількість викликає сумніви щодо цінності роботи і відсутності самостійності її виконання автором. Роботу направити на розгляд експертної комісії кафедри.

□ 3. Виявлені у роботі запозичення є недобросовісними і мають ознаки плагіату та/або в ній містяться навмисні спотворення тексту, що вказують на спроби приховування недобросовісних запозичень.

Особа, відповідальна за перевірку _____ (підпис) Каплун В. А. (прізвище, ініціали)

Ознайомлені з повним звітом подібності, який був згенерований системою Unicheck щодо роботи.

Автор роботи _____ (підпис) Буняк В. М. (прізвище, ініціали)

Керівник роботи _____ (підпис) Лужецький В. А. (прізвище, ініціали)

Додаток Б. Лістинг програми

```
using MongoDB.Bson;
using MongoDB.Driver;
using NLog;
using System;
using System.Configuration;
using System.Text;

namespace AirSIEM
{
    class Program
    {
        private static Logger logger =
LogManager.GetCurrentClassLogger();

        private static CorrelationEngine
engine;
        private static RabbitMQConsumer
consumer;



        private static string rabbitUri =
GetAppSetting("RabbitMQUri",
"amqp://localhost");
        private static string queueName =
GetAppSetting("QueueName",
"AirSIEM_ConnectorQueue");



        private static bool
translateAlertsToDB =
GetAppSetting("TranslateAlertsToDB",
"true").Equals("true") ? true : false;
        private static string
MongoDBConnectionString =
GetAppSetting("MongoDBConnectionString",
"mongodb://localhost:27017");

        public static string
GetAppSetting(string key, string def = "not
exists")
        {
            if
(ConfigurationManager.AppSettings[key] !=
null)
                return
ConfigurationManager.AppSettings[key].ToStrin
g();
            return def;
        }

        static void Main(string[] args)
        {
            try
            {
                logger.Debug("----- AirSIEM
start -----");



                engine = new
CorrelationEngine();
                engine.onAlertReceived += new
CorrelationEngine.onReceiveAlert(handleAlert)
;
engine.ParseRuleDir(GetAppSetting("RuleFolder
"));



engine.GenerateQueueList(engine.ruleList, ref
engine.fireDictionary);



                consumer = new
RabbitMQConsumer(rabbitUri, queueName);
                consumer.onMessageReceived +=
new
RabbitMQConsumer.onReceiveMessage(handleMessa
ge);
                consumer.Consume();

                engine.LogStatistics();
                logger.Debug("----- AirSIEM
stop -----");
            }
            catch (Exception ex)
            {
                logger.Error("Exception: " +
ex.ToString());
            }
        }



        public static void
handleMessage(byte[] utfMessage)
        {
            try
            {
                var message =
Encoding.UTF8.GetString(utfMessage);
                if
(message.Contains("ApacheConnector:"))
                {
                    SecurityEvent
securityEvent = new SecurityEvent(message,
LogMessageType.ApacheLog);

logger.Trace(securityEvent.ToString());



engine.ProcessMessage(securityEvent);
                }
            }
            catch (Exception ex)
            {
                logger.Warn("HandleMessage
exception: " + ex.ToString());
            }
        }



        public static void
handleAlert(SecurityEvent securityEvent, Rule
rule)
        {
            logger.Trace("{0}  Rule {1}
matched", Assistant.GetPadding(), rule.ID);
```

```
            logger.Trace("{0}  ALERT: LEVEL
{1} - {2}", Assistant.GetPadding(),
rule.level, rule.description);
            Console.WriteLine("ALERT: LEVEL
{1} - {2}", Assistant.GetPadding(),
rule.level, rule.description);

            try
            {
                if (translateAlertsToDB)
                {
                    DateTime UnixEpoch = new
DateTime(1970, 1, 1, 0, 0, 0,
DateTimeKind.Utc);
                    long timeStamp =
(long)(DateTime.UtcNow -
UnixEpoch).TotalSeconds;



                    var client = new
MongoClient(MongoDBConnectionString);
                    var dataBase =
client.GetDatabase("AirSIEM");
                    var collection =
dataBase.GetCollection<BsonDocument>("alerts"
);

                    var alert = new
BsonDocument
                    {
                        {
"matching_rule_SID", rule.ID },
                        { "message",
rule.description },
                        { "src_IP",
securityEvent.srcIP },
                        { "src_port",
securityEvent.srcPort },
                        { "dest_IP",
securityEvent.destIP },
                        { "dest_port",
securityEvent.destPort },
                        { "log_string",
securityEvent.logString },
                        { "level",
rule.level},
                        { "timestamp",
DateTime.Now.ToString("HH:mm:ss") },
                        { "rule_chain", new
BsonArray(new[] { 12345, 1234, 123 }) }
                    };


collection.InsertOne(alert);
                }
            }
            catch (Exception ex)
            {
                logger.Warn("HandleAlert
exception: " + ex.ToString());
            }
        }
    }
}
```

```
using NLog;
using System;
using System.Globalization;

namespace AirSIEM
{
    public enum LogMessageType
    {
        ApacheLog = 1
    }

    public class SecurityEvent
    {
        private static Logger logger =
LogManager.GetCurrentClassLogger();


        public string logString = "";
        public string message = "";
        public DateTime time;
        public string srcIP = "";
        public int srcPort = 0;
        public string destIP = "";
        public int destPort = 0;


        public SecurityEvent(string
logMessage, LogMessageType messageType)
        {
            try
            {
                logString = logMessage;

                if (messageType ==
LogMessageType.ApacheLog)
                {
                    string messageString =
logMessage.Replace("ApacheConnector:", "");

                    int splitPos;
                    string timeString = "";
                    string requestLine = "";
                    string referer = "";
                    string userAgent = "";


                    splitPos =
messageString.IndexOf(" ");
                    srcIP =
messageString.Substring(0, splitPos);
                    messageString =
messageString.Substring(splitPos + 1);


                    splitPos =
messageString.IndexOf(" ");
                    messageString =
messageString.Substring(splitPos + 1);


                    splitPos =
messageString.IndexOf(" ");
                    messageString =
messageString.Substring(splitPos + 1);


                    splitPos =
messageString.IndexOf("]");

                    timeString =
messageString.Substring(1, splitPos - 7);
```

```
                    time =
DateTime.ParseExact(timeString,
"dd/MMM/yyyy:HH:mm:ss",

CultureInfo.InvariantCulture);
                    messageString =
messageString.Substring(splitPos + 3);


                    splitPos =
messageString.IndexOf("\"");
                    requestLine =
messageString.Substring(0, splitPos);
                    messageString =
messageString.Substring(splitPos + 2);


                    splitPos =
messageString.IndexOf(" ");
                    messageString =
messageString.Substring(splitPos + 1);


                    splitPos =
messageString.IndexOf(" ");
                    messageString =
messageString.Substring(splitPos + 2);


                    splitPos =
messageString.IndexOf("\"");
                    referer =
messageString.Substring(0, splitPos);
                    messageString =
messageString.Substring(splitPos + 3);



                    splitPos =
messageString.IndexOf("\"");
                    userAgent =
messageString.Substring(0, splitPos);

                    message = requestLine;
                }
            }
            catch (Exception ex)
            {
                logger.Warn("SecurityEvent
exception: {0}", ex.ToString());
            }
        }

        public override string ToString()
        {
            return
String.Format("SecurityEvent object =>
timestamp=[{0}], source=[{1}],
destination=[{2}], message=[{3}]",
                time.ToString("HH:mm:ss"),
srcIP, destIP, message);
        }
    }
}

using NLog;
using System;
using System.Collections.Generic;
using System.Diagnostics;
```

```
namespace AirSIEM
{
    public static class Assistant
    {
        private static Logger logger =
LogManager.GetCurrentClassLogger();

        public static string GetPadding()
        {

            int callNum = 0;

            try
            {
                StackTrace stackTrace = new
StackTrace();
                StackFrame[] stackFrames =
stackTrace.GetFrames();

                foreach (var stackFrame in
stackFrames)
                {
                    string source =
stackFrame.ToString();


                    int n = 0;
                    string subString =
"ApplyRule";
                    while ((n =
source.IndexOf(subString, n,
StringComparison.InvariantCulture)) != -1)
                    {
                        n +=
subString.Length;
                        callNum++;
                    }


                    n = 0;
                    subString =
"CheckIfMatched";
                    while ((n =
source.IndexOf(subString, n,
StringComparison.InvariantCulture)) != -1)
                    {
                        n +=
subString.Length;
                        callNum++;
                    }
                }
            }
            catch (Exception ex)
            {
                logger.Warn("GetPadding
exception: " + ex.ToString());
            }

            return "".PadRight((callNum - 1)
* 2, ' ');
        }
    }
}

using NLog;
using System;
using System.Collections.Generic;
using System.Linq;
```

```
namespace AirSIEM
{
    public class FireQueue
    {
        private static Logger logger =
LogManager.GetCurrentClassLogger();

        public Queue<FireQueueItem> Queue;
        public int ID;
        public int timeFrame;
        public int maxSize;

        public FireQueue(int vID, int
vTimeFrame, int vMaxSize = 1000)
        {
            Queue = new
Queue<FireQueueItem>();
            ID = vID;
            timeFrame = vTimeFrame;
            maxSize = vMaxSize;
        }

        public void Enqueue(SecurityEvent
securityEvent)
        {
            FireQueueItem item = new
FireQueueItem(securityEvent.time,
securityEvent.srcIP,
                securityEvent.srcPort,
securityEvent.destIP,
securityEvent.destPort);


            if (Queue.Count == maxSize)
            {
                logger.Trace("{0}  Dequeue
item (>maxSize): {1}",
Assistant.GetPadding(),
Queue.First().ToString());
                Queue.Dequeue();
            }

            Queue.Enqueue(item);
            logger.Trace("{0}  Enqueue item:
{1}", Assistant.GetPadding(),
item.ToString());


            while (true)
            {
                if ((item.time -
Queue.First().time).TotalSeconds > timeFrame)
                {
                    logger.Trace("{0}
Dequeue item (>timeFrame): {1}",
Assistant.GetPadding(),
Queue.First().ToString());
                    Queue.Dequeue();
                }
                else break;
            }

            ShowQueueBrief();
        }

        public void Dequeue()
        {
            Queue.Dequeue();
        }

        public bool
CheckIfMatched(SecurityEvent securityEvent,
Rule rule)
        {


logger.Trace(Assistant.GetPadding() +
"QueueDictionary.CheckIfMatched start");

            int counter = 0;
            int counterSameSourceIP = 0;

            for (int i = Queue.Count - 1; i
>= 0; i--)
            {

                if ((securityEvent.time -
Queue.ElementAt(i).time).TotalSeconds >
rule.timeFrame) break;
                counter++;
                logger.Trace("{0}  counter++
=> counter=[{1}]", Assistant.GetPadding(),
counter);


                if (rule.sameSourceIP)
                {
                    if
(securityEvent.srcIP.Equals(Queue.ElementAt(i
).srcIP))
                    {

counterSameSourceIP++;
                        logger.Trace("{0}
counterSameSourceIP++ =>
counterSameSourceIP=[{1}]",

Assistant.GetPadding(), counterSameSourceIP);
                    }
                    if (counterSameSourceIP
>= rule.frequency) return true;
                }


                if (!rule.sameSourceIP)
                    if (counter >=
rule.frequency) return true;
            }

            return false;
        }

        public void ShowQueueBrief()
        {

logger.Trace(Assistant.GetPadding() + "  " +
this.ToString());

            int index = 0;
            foreach (FireQueueItem item in
Queue)
            {
                index++;

logger.Trace(Assistant.GetPadding() + "    "
+ index + ": " + item.ToString());
```

```
            }
        }


using System.Collections.Generic;

namespace AirSIEM
{
    public class Rule
    {


        public int ID;
        public int level;
        public int frequency;
        public int timeFrame;
        public int ignore;


        public int ifSID;

        public int ifMatchedSID;

        public string sourceIP;
        public string destIP;
        public bool sameSourceIP = false;

        public string match;
        public string description;
        public string parentFile;
        public string XML;

        public List<int> children;

        public bool HasParent()
        {
            if (ifSID != 0 || ifMatchedSID !=
0) return true;
            else return false;
        }

        public bool HasChidren()
        {
            if (children.Count > 0) return
true;
            else return false;
        }
    }
}

using NLog;
using System;
using System.Collections.Generic;
using System.IO;
using System.Xml;

namespace AirSIEM
{
    public class CorrelationEngine
    {
        private static Logger logger =
LogManager.GetCurrentClassLogger();

        public int maxCount = 10000;
        public Dictionary<int, FireQueue>
fireDictionary = new Dictionary<int,
FireQueue>();
        public Dictionary<int, Rule> ruleList
= new Dictionary<int, Rule>();
        public Dictionary<int, int> matchList
= new Dictionary<int, int>();


        public delegate void
onReceiveAlert(SecurityEvent securityEvent,
Rule rule);
        public event onReceiveAlert
onAlertReceived;

        public void ParseRuleDir(string
rulePath)
        {
            int fileNum = 0;
            int ruleNum = 0;

            logger.Debug("ParseRuleDir start:
{0}", rulePath);

            try
            {
                if (!rulePath.Equals(""))
                {
                    string[] fileEntries =
Directory.GetFiles(rulePath, "*.xml",
SearchOption.AllDirectories);
                    ruleList = new
Dictionary<int, Rule>();

                    foreach (string fileName
in fileEntries)
                    {
                        FileInfo file = new
FileInfo(fileName);
                        List<Rule>
fileRuleList = new List<Rule>();

ParseRulesFromXML(file.FullName, ref
fileRuleList);

                        foreach (Rule rule in
fileRuleList)

ruleList.Add(rule.ID, rule);

                        fileNum++;
                        ruleNum +=
fileRuleList.Count;
                    }
                }
            }
            catch (Exception ex)
            {
                logger.Warn("ParseRuleDir
exception: {0}", ex.ToString());
            }

            logger.Trace("ParseRuleDir: total
{0} files processed", fileNum);
            logger.Trace("ParseRuleDir: total
{0} rules processed", ruleNum);
            logger.Debug("ParseRuleDir
stop");

            CheckDependencies(ref ruleList);
        }

        public void ParseRulesFromXML(string
fileName, ref List<Rule> ruleList)
```

```
        {
            string logString = "";

            try
            {

logger.Trace("ParseRulesFromXML handles file:
" + fileName);

                FileInfo file = new
FileInfo(fileName);


                string xml =
File.ReadAllText(fileName);
                xml = "<root>" + xml +
"</root>";


                using (XmlReader preReader =
XmlReader.Create(new StringReader(xml)))
                {
                    while
(preReader.ReadToFollowing("var"))
                    {
                        string varName =
preReader.GetAttribute("name");
                        string varValue =
preReader.ReadString();


                        xml = xml.Replace("$"
+ varName, varValue);
                    }
                }

                XmlDocument doc = new
XmlDocument();
                doc.LoadXml(xml);



                XmlNode groupNode =
doc.SelectNodes("root/group")[0];
                string groupName =
groupNode.Attributes["name"].Value.ToString()
;


                foreach (XmlNode node in
doc.SelectNodes("root/group/rule"))
                {
                    Rule rule = new Rule();
                    rule.children = new
List<int>();

                    rule.ID =
(node.Attributes["id"] == null) ? 0 :
int.Parse(node.Attributes["id"].Value);
                    logString += rule.ID + "
";

                    rule.level =
(node.Attributes["level"] == null) ? 0 :
int.Parse(node.Attributes["level"].Value);
                    rule.frequency =
(node.Attributes["frequency"] == null) ? 0 :
int.Parse(node.Attributes["frequency"].Value)
;
                    rule.timeFrame =
(node.Attributes["timeframe"] == null) ? 0 :
int.Parse(node.Attributes["timeframe"].Value)
;

                    rule.ifSID =
(node.SelectSingleNode("if_sid") == null) ? 0
:
int.Parse(node.SelectSingleNode("if_sid").Inn
erText);
                    rule.ifMatchedSID =
(node.SelectSingleNode("if_matched_sid") ==
null) ? 0 :
int.Parse(node.SelectSingleNode("if_matched_s
id").InnerText);

                    rule.sourceIP =
(node.SelectSingleNode("srcip") == null) ? ""
: node.SelectSingleNode("srcip").InnerText;
                    rule.destIP =
(node.SelectSingleNode("dstip") == null) ? ""
: node.SelectSingleNode("dstip").InnerText;
                    rule.sameSourceIP =
(node.SelectSingleNode("same_source_ip") ==
null) ? false : true;

                    rule.match =
(node.SelectSingleNode("match") == null) ? ""
: node.SelectSingleNode("match").InnerText;
                    rule.description =
(node.SelectSingleNode("description") ==
null) ? "" :
node.SelectSingleNode("description").InnerTex
t;
                    rule.parentFile =
file.Name;
                    rule.XML = node.OuterXml;

                    ruleList.Add(rule);
                }

                logString =
logString.Trim().Replace(" ", ", ");
                logger.Trace("  {0} rules
processed: {1}", ruleList.Count, logString);
            }
            catch (Exception ex)
            {

logger.Warn("ParseRulesFromXML exception (log
{0}): {1}", logString, ex.ToString());
            }
        }

        public void CheckDependencies(ref
Dictionary<int, Rule> ruleList)
        {
            logger.Debug("CheckDependencies
start");


            foreach (KeyValuePair<int, Rule>
ruleKVP in ruleList)
            {
                Rule rule = ruleKVP.Value;

                if (rule.ifSID != 0)

ruleList[rule.ifSID].children.Add(rule.ID);
```

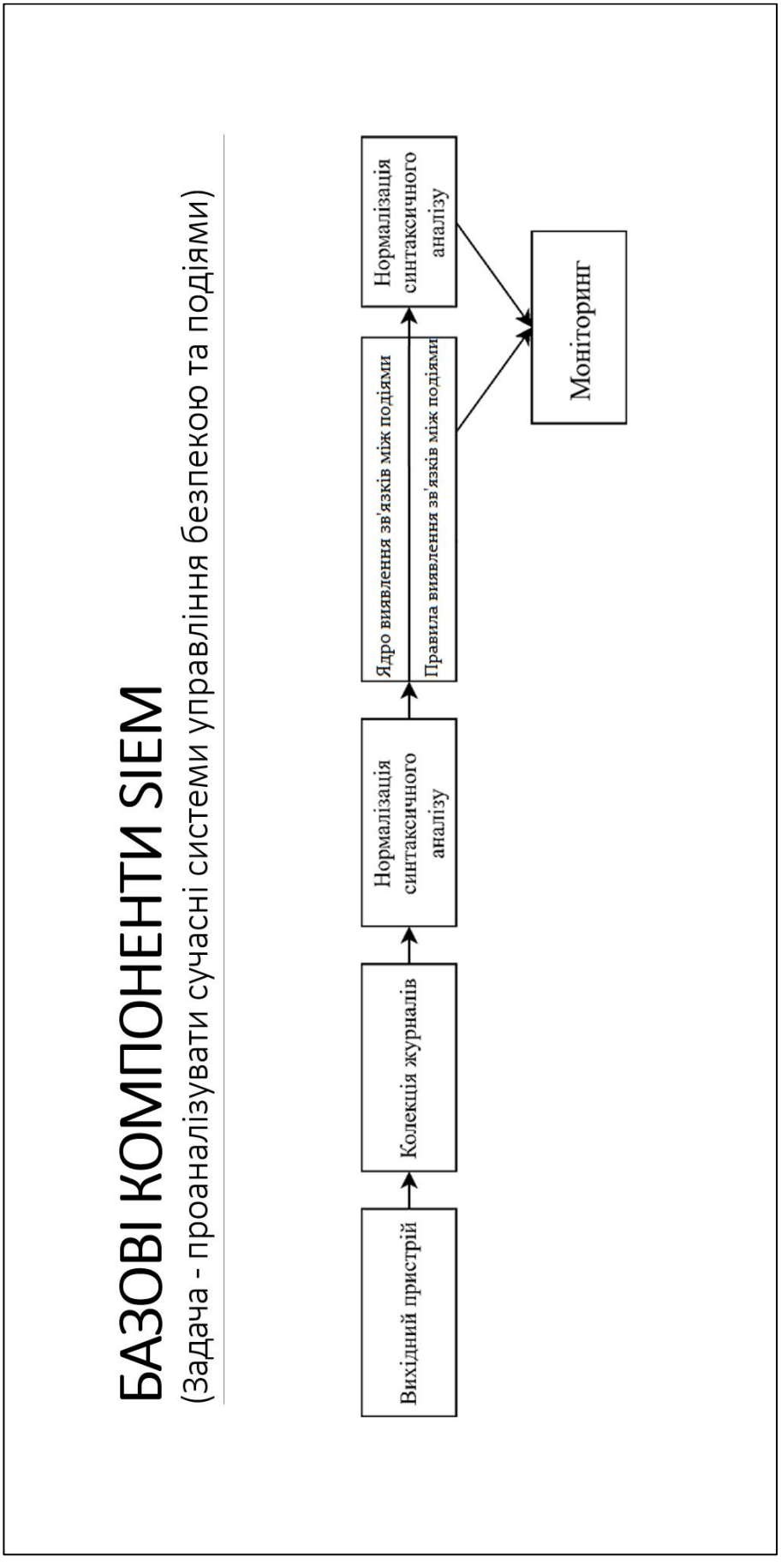
```
                if (rule.ifMatchedSID != 0)
ruleList[rule.ifMatchedSID].children.Add(rule
.ID);
            }

            logger.Trace("Dependencies: ");

            foreach (KeyValuePair<int, Rule>
ruleKVP in ruleList)
            {
                string logString = "";
                foreach (int item in
ruleKVP.Value.children) logString += item + "
";
                logger.Trace("  {0} children
=> {1}", ruleKVP.Value.ID,
logString.Trim().Replace(" ", ", "));
            }

            logger.Debug("CheckDependencies
stop");
        }

        public void
GenerateQueueList(Dictionary<int, Rule>
ruleList,
            ref Dictionary<int, FireQueue>
queueDictionary)
        {
            logger.Debug("GenerateQueueList
start");
            try
            {
                foreach (KeyValuePair<int,
Rule> ruleKVP in ruleList)
                {

                    if
(ruleKVP.Value.frequency == 0) continue;


                    int ID =
ruleKVP.Value.ifMatchedSID;


                    if
(queueDictionary.ContainsKey(ID))
                    {
queueDictionary[ID].timeFrame =
Math.Max(ruleKVP.Value.timeFrame,

queueDictionary[ID].timeFrame);
                    }
                    else
                    {
                        FireQueue queue = new
FireQueue(ID, ruleKVP.Value.timeFrame);

queueDictionary.Add(ID, queue);
                    }
                }

                logger.Trace("Created {0}
queues:", queueDictionary.Count);
                foreach (KeyValuePair<int,
FireQueue> kvp in queueDictionary)
                    logger.Trace("  " +
kvp.ToString());
            }
            catch (Exception ex)
            {

logger.Warn("GenerateQueueList exception: " +
ex.ToString());
            }

            logger.Debug("GenerateQueueList
stop");
        }

        public void
ProcessMessage(SecurityEvent securityEvent)
        {
            foreach (KeyValuePair<int, Rule>
ruleKVP in ruleList)
            {
                Rule rule = ruleKVP.Value;
                if (!rule.HasParent())
ApplyRule(securityEvent, rule);
            }
        }

        public void ApplyRule(SecurityEvent
securityEvent, Rule rule)
        {
            string padding =
Assistant.GetPadding();
            logger.Trace("{0}Check rule {1} -
{2}", padding, rule.ID, rule.description);

            if (CheckIfMatched(ref
securityEvent, ref rule) == true)
            {
                MatchRule(securityEvent,
rule);
                if (rule.HasChidren())
                {
                    logger.Trace(padding + "
Check the child rules");
                    foreach (int item in
rule.children)

ApplyRule(securityEvent, ruleList[item]);
                    logger.Trace(padding + "
Check the child rules: OK");
                }
            }
            else
            {
                logger.Trace("{0}Rule {1} not
matched", padding, rule.ID);
            }

            logger.Trace("{0}Check rule {1}:
OK", padding, rule.ID);
        }

        public bool CheckIfMatched(ref
SecurityEvent securityEvent, ref Rule rule)
        {
            string padding =
Assistant.GetPadding();


            if (rule.ifSID != 0)
            {
```

```
                logger.Trace("{0}Check
<if_sid>{1}</if_sid>", padding, rule.ifSID);

                if
(!matchList.ContainsKey(rule.ifSID)) return
false;
                if (matchList[rule.ifSID] ==
0) return false;
            }


            if (!rule.sourceIP.Equals(""))
            {
                logger.Trace(padding + "
Check <same_source_ip/>");
                if
(!securityEvent.srcIP.Equals(rule.sourceIP))
return false;
            }
            if (!rule.match.Equals(""))
            {
                logger.Trace("{0}Check
<match>{1}</match>", padding, rule.match);

                bool check = false;
                string[] parts =
rule.match.Split(new Char[] { '|' },
StringSplitOptions.RemoveEmptyEntries);

                foreach (string part in
parts)
                {
                    if
(securityEvent.message.IndexOf(part,
StringComparison.OrdinalIgnoreCase) >= 0)
                    {
                        check = true;
                        break;
                    }
                }
                if (!check) return false;
            }
            if (rule.ifMatchedSID != 0)
            {
                logger.Trace("{0}Check
<if_matched_sid>{1}</if_matched_sid>",
padding, rule.ifMatchedSID);
                if
(matchList.ContainsKey(rule.ifMatchedSID))
                {
                    if
(matchList[rule.ifMatchedSID] == 0) return
false;
                    if
(!fireDictionary.ContainsKey(rule.ifMatchedSI
D)) return false;
                    if
(fireDictionary[rule.ifMatchedSID].CheckIfMat
ched(securityEvent, rule))
                    {
                        logger.Trace("{0}
Rule {1} QueueDictionary.CheckIfMatched ==
TRUE", padding, rule.ifMatchedSID);
                    }
                    else
                    {
                        logger.Trace("{0}
Rule {1} QueueDictionary.CheckIfMatched ==
FALSE", padding, rule.ifMatchedSID);
                        return false;
                    }
                }
            }
            return true;
        }

        public void MatchRule(SecurityEvent
securityEvent, Rule rule)
        {

            onAlertReceived(securityEvent,
rule);
            if
(fireDictionary.ContainsKey(rule.ID))
            {

logger.Trace(Assistant.GetPadding() + "
Matched rule is queue tracked");

fireDictionary[rule.ID].Enqueue(securityEvent
);
            }
            if
(matchList.ContainsKey(rule.ID))
            {
                matchList[rule.ID]++;
                if (matchList[rule.ID] >
maxCount) matchList[rule.ID] = maxCount;
            }
            else
                matchList.Add(rule.ID, 1);
        }

        public void LogStatistics()
        {
            logger.Debug("MatchList stat: ");
            foreach (KeyValuePair<int, int>
kvp in matchList)
                logger.Trace("  Rule {0}
fires {1} times", kvp.Key, kvp.Value);
        }
    }
}

using System.Reflection;
using System.Runtime.CompilerServices;
using System.Runtime.InteropServices;


[assembly: AssemblyTitle("AirSIEM")]
[assembly: AssemblyDescription("")]
[assembly: AssemblyConfiguration("")]
[assembly: AssemblyCompany("MICROSOFT")]
[assembly: AssemblyProduct("AirSIEM")]
[assembly: AssemblyCopyright("Copyright ©
MICROSOFT 2018")]
[assembly: AssemblyTrademark("")]
[assembly: AssemblyCulture("")]

 [assembly: ComVisible(false)]

 [assembly: Guid("5cdef6d8-8aab-42cc-aab2-
dddc3e0effb0")]

 [assembly: AssemblyVersion("1.0.0.0")]
[assembly: AssemblyFileVersion("1.0.0.0")]
```

ІЛЮСТРАТИВНА ЧАСТИНА

СИСТЕМА УПРАВЛІННЯ БЕЗПЕКОЮ ТА ПОДІЯМИ. ЧАСТИНА 1. МЕТОД ТА ПРОГРАМНИЙ ЗАСІБ ДЛЯ УПРАВЛІННЯ БЕЗПЕКОЮ ТА ПОДІЯМИ

БАЗОВІ КОМПОНЕНТИ SIEM



МЕТОД УПРАВЛІННЯ БЕЗПЕКОЮ ТА ПОДІЯМИ

(Задача - розробити метод для управління безпекою та подіями)

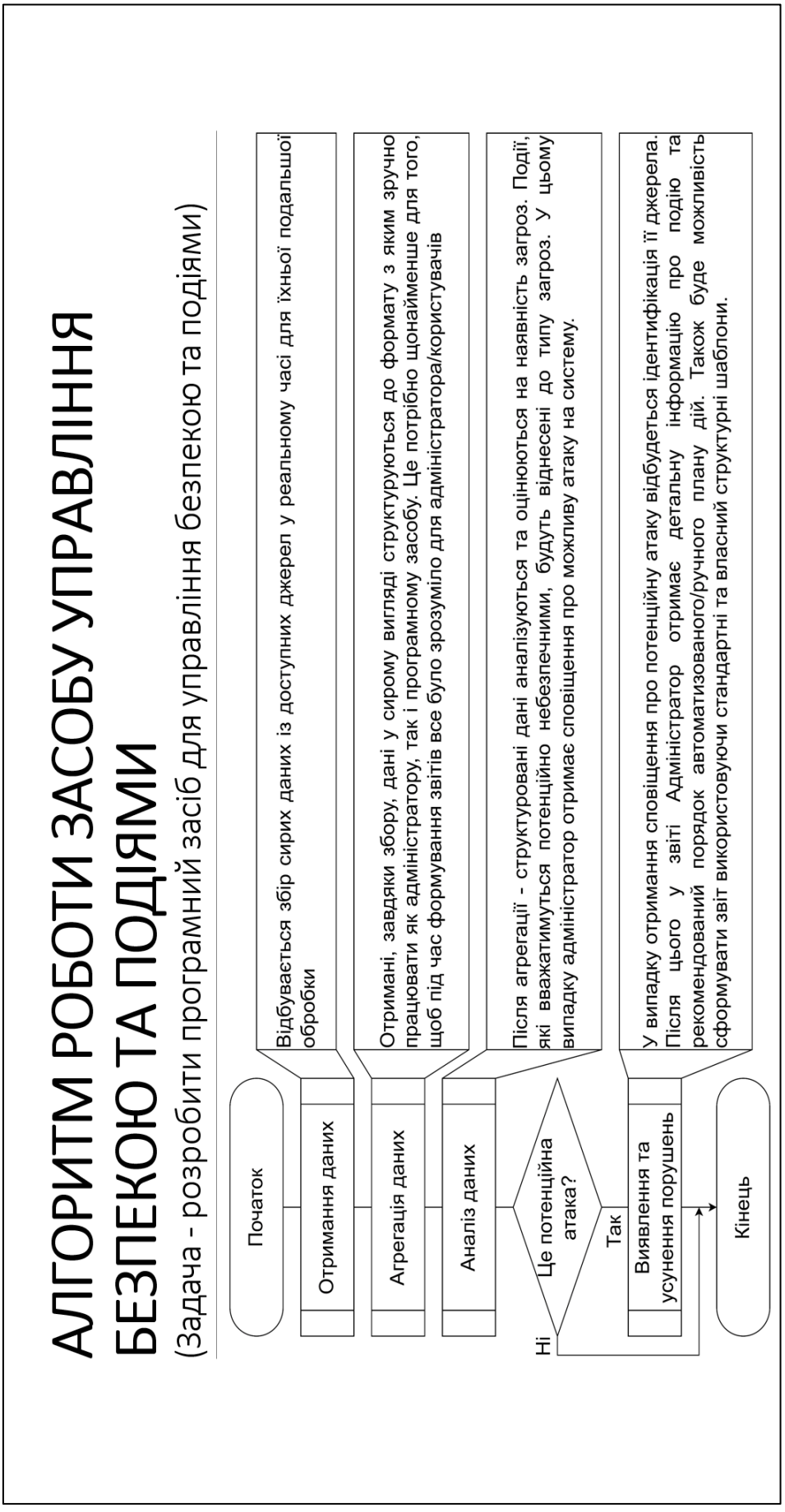
1. **Збір даних:** інструмент збору інформації збирає дані в реальному часі.

2. **Агрегація даних:** дані співвідносяться з подібними подіями, щоб полегшити їх аналіз для людей. Програмне забезпечення робить інформацію легшою для використання та читання людьми, щоб спростити процес.

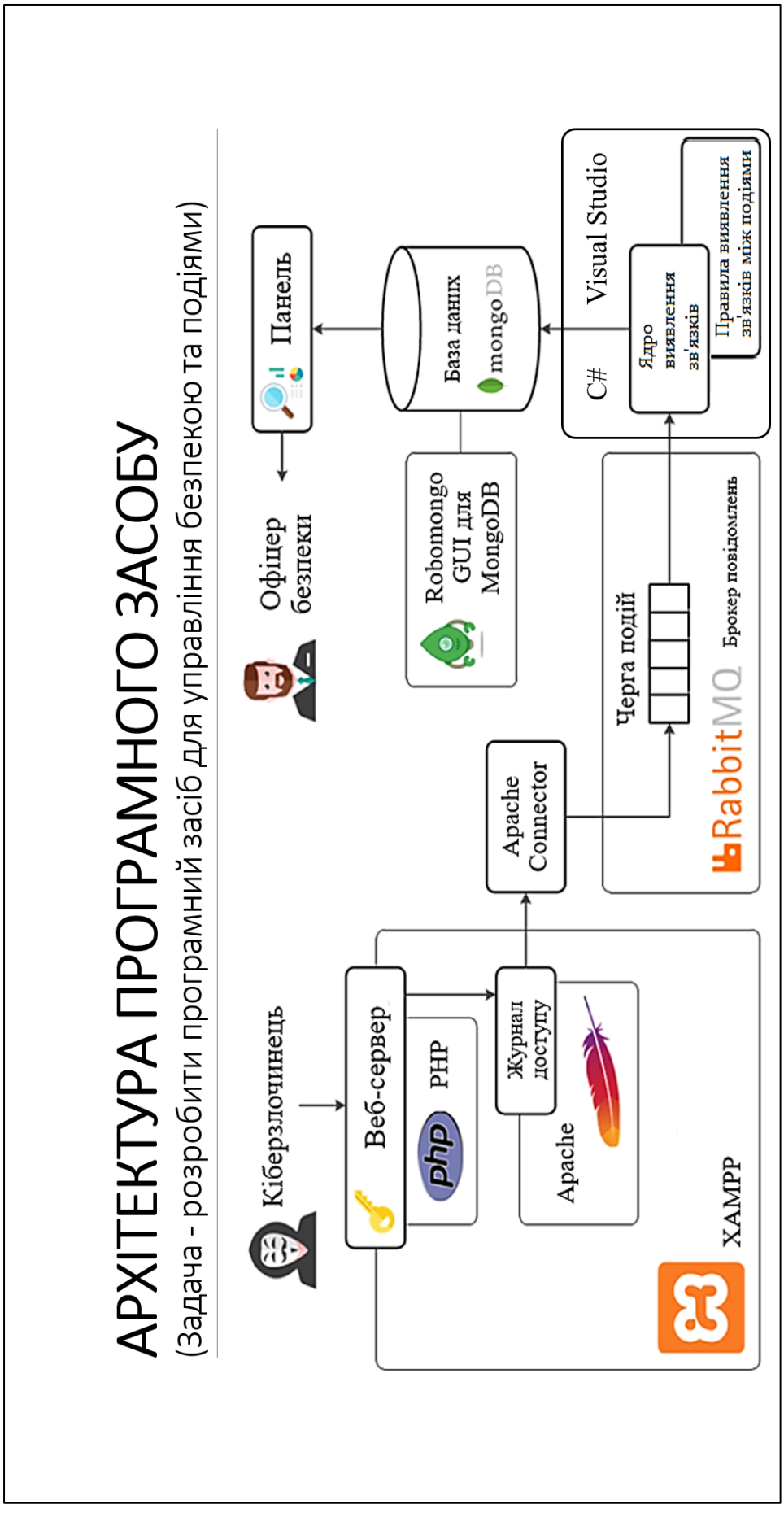
3. **Аналіз:** дані аналізуються на наявність загроз, щоб сповістити адміністраторів/офіцерів безпеки. За допомогою ряду аналітичних засобів потенційно небезпечні дані відокремлюються від непроблемних даних, а адміністратори/офіцери безпеки сповіщаються про потенційні загрози.

4. **Виявлення та усунення порушень:** порушення, виявлені шляхом збору та аналізу даних, ідентифікуються та виправляються.

АЛГОРИТМ РОБОТИ ЗАСОБУ УПРАВЛІННЯ БЕЗПЕКОЮ ТА ПОДІЯМИ

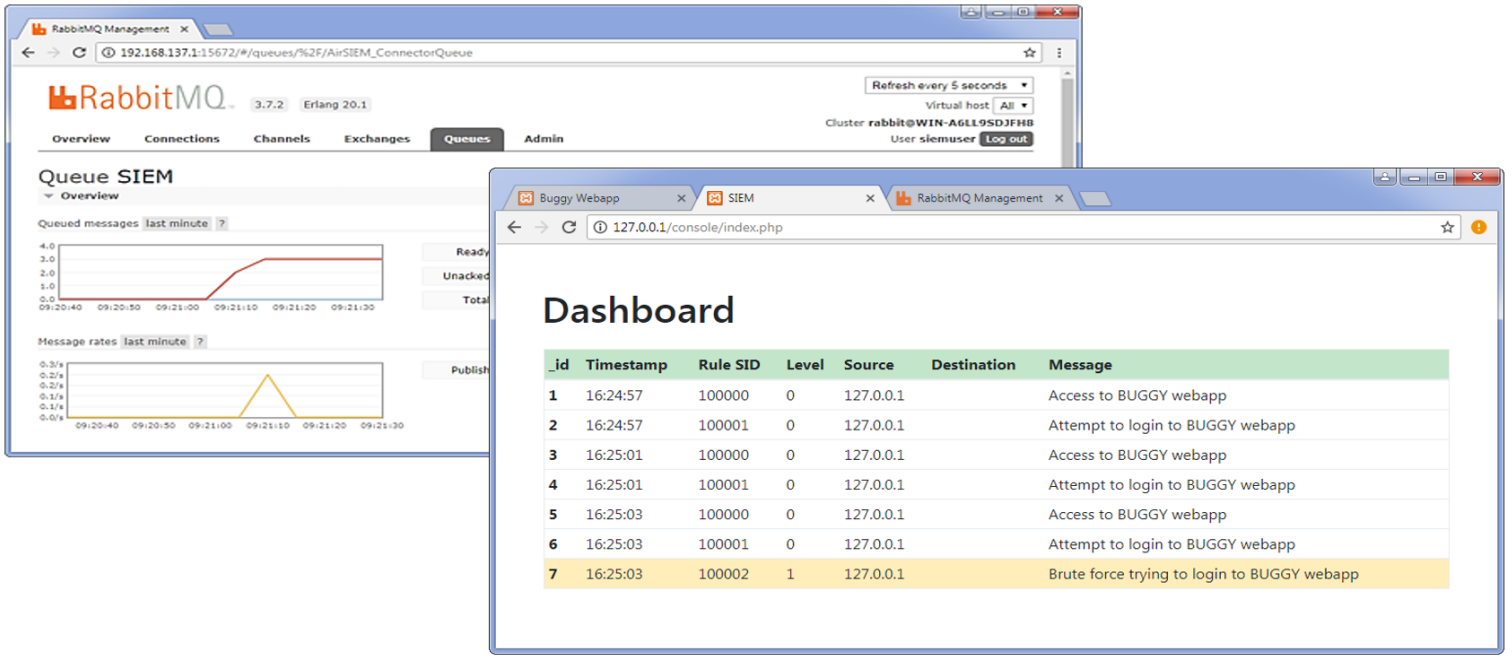


АРХІТЕКТУРА ПРОГРАМНОГО ЗАСОБУ



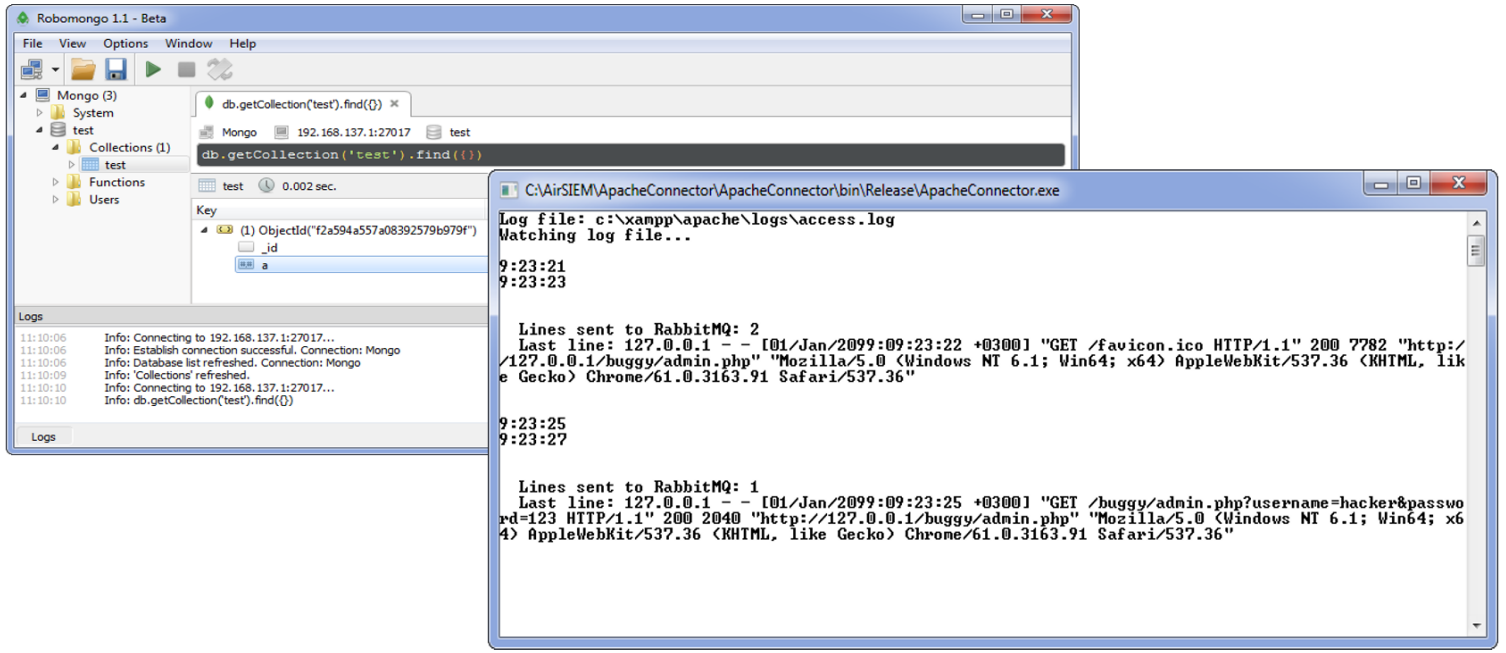
РЕЗУЛЬТАТИ ТЕСТУВАННЯ ПРОГРАМНОГО ЗАСОБУ ТА ЙОГО СКЛАДОВИХ

(Задача - проаналізувати виконані розробки з метою виявлення можливих помилок та їх усунення)



РЕЗУЛЬТАТИ ТЕСТУВАННЯ ПРОГРАМНОГО ЗАСОБУ ТА ЙОГО СКЛАДОВИХ

(Задача - проаналізувати виконані розробки з метою виявлення можливих помилок та їх усунення)



РЕЗУЛЬТАТИ ПОРІВНЯННЯ SIEM

Функціональність	ArcSight	Qradar	McAfee	LogRhythm	RSA	Розроблена SIEM
Правила виявлення зв'язків між подіями	2	2	3	3	2	**3**
Джерела даних	3	3	3	2	3	**3**
Обробка в реальному часі	3	3	3	3	3	**3**
Обсяг даних	3	2	3	2	2	**3**
Візуалізація	1	2	2	1	1	**2**
Аналітика даних	2	3	2	3	2	**2**
Продуктивність	2	2	3	2	3	**3**
Експертиза	1	2	2	2	3	**3**
Складність	3	2	2	2	3	**1**
Масштабованість	2	2	2	1	1	**3**
Аналіз ризиків	1	2	2	2	2	**3**
Зберігання	2	2	3	2	2	**3**
Вартість	3	3	3	2	3	**3**
Стійкість	2	3	3	2	3	**3**
Реакція та звіт	1	1	3	3	2	**2**
Безпека	2	3	1	1	2	**3**