

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 підсистеми централізованого логу-
вання мережних пристроїв організації на базі ELK
stack»

Виконав: студент 4-го курсу групи КБ-22-1
спеціальності 125 «Кібербезпека»



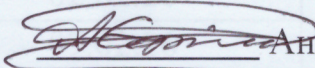
Назар ПЕТЛЮК

Керівник: старший викладач кафедри КІ та КБ



Олександра ПОКОТИЛО

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



Андрій ЄФІМЕНКО

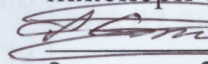
Житомир – 2026

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
2 лютого 2026 р.

ЗАВДАННЯ

на випускн кваліфікаційну роботу бакалавра

Студент: Петлюк Назар Євгенович

Тема роботи: «Проект підсистеми централізованого логування мережних пристроїв організації на базі ELK stack»

затверджена Наказом університету від 2 лютого 2026 р. №58/с

Термін здачі студентом закінченої роботи 12 червня 2026 р.

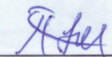
Вихідні дані роботи: метою кваліфікаційної роботи є проектування та реалізація системи мережного моніторингу, дослідження структури логів різних типів мережних пристроїв із використанням стеку Elastic Stack (ELK).

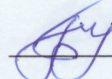
Обладнання та програмне забезпечення:

1. Мережні пристрої: Cisco 2821, switch, MikroTik RB5009UG+S+
2. Ubuntu Server з налаштованим ELK stack
3. Linux PC з необхідними програмним забезпеченням

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

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв
1	Фальковський І. Г	02.02.26	06.02.26 
2	Фальковський І. Г	09.02.26	13.02.26 

Студент  Назар ПЕТЛЮК

Керівник  Олександра ПОКОТИЛО

Календарний план

№ з/п	Назви етапів випускної кваліфікаційної роботи	Термін виконання етапів роботи	Примітка
1	Аналіз отриманого завдання та уточнення мети і задач кваліфікаційної роботи	02.02.2026 – 08.02.2026	Виконано
2	Збір та аналіз літературних джерел щодо мережного моніторингу, логування та SIEM-систем	09.02.2026 – 15.02.2026	Виконано
3	Аналіз об'єкта дослідження (мережні пристрої Cisco, MikroTik, система логування)	16.02.2026 – 21.02.2026	Виконано
4	Постановка задачі дослідження та визначення вимог до системи моніторингу	23.02.2026 – 25.02.2026	Виконано
5	Побудова структурної моделі мережної інфраструктури	26.02.2026 – 03.03.2026	Виконано
6	Вибір інструментів збору, зберігання та аналізу логів	04.03.2026 – 06.03.2026	Виконано
7	Розгортання тестового середовища (GNS3, Ubuntu Server)	07.03.2026 – 12.03.2026	Виконано
8	Налаштування аналітичних панелей, індексів та фільтрації повідомлень у Kibana	13.03.2026 – 17.03.2026	Виконано
9	Розгортання середовища на реальному обладнанні, проведення практичних експериментів, симуляція мережних атак (OSPF, DHCP Starvation), збір логів та написання розділу 3	18.03.2026 – 12.04.2026	Виконано
10	Завершення оформлення пояснювальної записки, підготовка додатків	13.04.2026 – 22.05.2026	Виконано

Студент Назар ПЕТЛЮК

Керівник Олександра ПОКОТИЛО

АНОТАЦІЯ

Кваліфікаційна робота бакалавра присвячена проектуванню та реалізації підсистеми централізованого логування мережних пристроїв організації на базі стеку Elastic Stack (ELK) для забезпечення ефективного моніторингу безпеки.

Метою роботи є розробка та практичне дослідження системи централізованого збору, обробки й аналізу журналів подій мережного обладнання для оперативної ідентифікації кіберзагроз.

Пояснювальна записка містить 78 сторінок, 23 рисунки, 9 таблиць. Список використаних джерел налічує 20 найменувань та займає 3 сторінки.

У роботі визначено ключові вимоги до систем журналювання, проаналізовано структуру повідомлень Syslog та особливості їх передачі через TCP і UDP. Обґрунтовано вибір апаратного забезпечення і компонентів ELK Stack. Розроблено захищену архітектуру передачі телеметрії із використанням шифрування TLS та незалежних конвеєрів Logstash із grok-фільтрами для нормалізації подій.

Для перевірки працездатності здійснено моделювання інфраструктури в середовищі GNS3, після чого архітектуру підсистеми було успішно перенесено в діюче тестове середовище. В інтерфейсі Kibana проведено аналіз ефективності виявлення трьох векторів деструктивного впливу: атак на протокол OSPF із впровадженням фіктивних LSA-пакетів (через FRR), DHCP Starvation (через Yersinia CLI) та підробки логів методом IP-spoofing за допомогою скриптів Python та бібліотеки Scapy. Наведено результати тестування системи та запропоновано методи протидії інцидентам, зокрема впровадження автентифікації MD5 для захисту маршрутизації.

Ключові слова: ЦЕНТРАЛІЗОВАНЕ ЛОГУВАННЯ, ELK STACK, SIEM, МЕРЕЖНИЙ МОНІТОРИНГ, CISCO IOS, ROUTEROS, SYSLOG, МЕРЕЖНІ АТАКИ, OSPF, DHCP STARVATION, LOG SPOOFING, SCAPY, KIBANA.

					ККІКБ.КР.Б – 125 – 13 – ПЗ			
Змн.	Арк.	№ докум.	Підпис	Дата	Проект підсистеми централізованого логування мережних пристроїв орга- нізації	Лім.	Арк.	Аркушів
Розроб.		Петлюк Н.Є.		2024				
Керівник		Покотило О.А.		2024			4	87
Реценз.		Єфіменко А.А.		2024		ФІКТ, Гр. КБ-22-1		
Н. контр.		Покотило О.А.		2024				
Затверд.		Єфіменко А.А.		2024				

ANNOTATION

The bachelor's qualification thesis is devoted to the design and implementation of a centralized network device logging subsystem for an organization based on the Elastic Stack (ELK) to ensure efficient security monitoring.

The purpose of the thesis is the development and practical investigation of a system for centralized collection, processing, and analysis of network equipment event logs for rapid cyber threat identification.

The explanatory note contains 78 pages, 23 figures, and 9 tables. The list of references includes 20 sources and is presented on 3 pages.

The thesis defines key requirements for logging systems and analyzes the Syslog message structure along with its transmission features over TCP and UDP. The choice of hardware and ELK Stack components is justified. A secure telemetry transmission architecture has been developed using TLS encryption and independent Logstash pipelines with grok filters for event normalization.

To verify operability, infrastructure modeling was performed in the GNS3 environment, after which the subsystem architecture was successfully transferred to a live test environment. Practical analysis of three destructive impact vectors was conducted in the Kibana interface: OSPF attacks via malicious LSA injection (using FRR), DHCP Starvation (using Yersinia CLI), and log spoofing via IP spoofing using custom Python scripts and the Scapy library. The testing results are presented, and incident mitigation methods are proposed, including the implementation of MD5 authentication for routing protection.

Keywords: CENTRALIZED LOGGING, ELK STACK, SIEM, NETWORK MONITORING, CISCO IOS, ROUTEROS, SYSLOG, NETWORK ATTACKS, OSPF, DHCP STARVATION, LOG SPOOFING, SCAPY, KIBANA.

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				5
Змн.	Арк.	№ докум.	Підпис	Дата		

ЗМІСТ

ВСТУП	8
РОЗДІЛ 1 ТЕОРЕТИЧНІ ОСНОВИ ЛОГУВАННЯ, СТРУКТУРА ЖУРНАЛІВ ПОДІЙ ТА МОЖЛИВОСТІ ЇХ ВИКОРИСТАННЯ	10
1.1 Поняття логуювання та його роль у забезпеченні інформаційної безпеки...	10
1.2. Механізми логуювання та передача логів	13
1.3. Особливості логуювання на мережних пристроях	16
1.4. Структура логів мережних пристроїв	17
Висновки до першого розділу	20
РОЗДІЛ 2 ВИБІР МЕРЕЖНИХ ПРИСТРОЇВ ТА ІНСТРУМЕНТІВ ДЛЯ ЗБОРУ, ОБРОБКИ Й АНАЛІЗУ ЛОГІВ, ПРОЄКТУВАННЯ МЕРЕЖІ	22
2.1 Вибір мережних пристроїв	22
2.2 Інструменти для збору та обробки даних	27
2.3 Мережний потік логів: передача, формат, захист та обробка	30
2.4 Проєктування мережі	35
Висновки до другого розділу	39
РОЗДІЛ 3 ПРАКТИЧНЕ ДОСЛІДЖЕННЯ СИСТЕМИ ЛОГУВАННЯ ТА АНАЛІЗ ПОДІЙ ІНФОРМАЦІЙНОЇ БЕЗПЕКИ	41
3.1 Атака на OSPF з використанням LSA пакетів.....	42
3.2. Атака DHCP Starvation з використанням discover запитів.....	55
3.3. Підробка логів у системі централізованого логуювання	65
Висновки до третього розділу	72
ВИСНОВКИ.....	74
СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ	77
ДОДАТКИ.....	80

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

SIEM – Security Information and Event Management

ELK Stack – Elasticsearch Logstash Kibana Stack

Elasticsearch – пошукова та аналітична система для зберігання логів

Logstash – інструмент збору, обробки та фільтрації логів

Kibana – веб-інтерфейс для візуалізації та аналізу даних

Cisco IOS – операційна система мережних пристроїв Cisco

RouterOS – операційна система мережних пристроїв MikroTik

Syslog – стандарт протоколу передачі журналів подій

SNMP – Simple Network Management Protocol

TCP – Transmission Control Protocol

UDP – User Datagram Protocol

TLS – Transport Layer Security

FRR – Free Range Routing

OSPF – Open Shortest Path First

LSDB – Link-State Database

LSA – Link-State Advertisement

SPF – Shortest Path First

DHCP – Dynamic Host Configuration Protocol

RFC – Request for Comments

SOC – Security Operation Center

DoS – Denial of Service

NSE – Nmap Scripting Engine

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				7
Змн.	Арк.	№ докум.	Підпис	Дата		

ВСТУП

Актуальність теми. У сучасних умовах розвитку інформаційних технологій та зростання кількості кіберзагроз забезпечення інформаційної безпеки мережних інфраструктур набуває особливої актуальності. Відсутність централізованого контролю за подіями в мережі ускладнює своєчасне виявлення атак і реагування на інциденти, що зумовлює необхідність використання ефективних систем моніторингу та аналізу подій безпеки. Важливу роль у сучасних системах захисту відіграє централізоване логування та аналіз подій, що дозволяє отримати цілісне уявлення про стан мережної інфраструктури. Застосування SIEM-подібних рішень забезпечує збір, зберігання, кореляцію та візуалізацію логів від мережних пристроїв, що дає змогу виявляти аномальну активність, типові мережні атаки та помилки конфігурації. Проєкт підсистеми централізованого логування на базі стеку Elastic Stack (ELK) є ефективним інструментом для побудови надійної архітектури аудиту, моніторингу та оперативного виявлення кіберзагроз у гетерогенному середовищі організації.

Метою кваліфікаційної роботи є проєктування та реалізація системи мережного моніторингу інформаційної безпеки на основі централізованого збору та аналізу логів мережних пристроїв із використанням стеку ELK Stack.

Визначена мета обумовлює наступні завдання:

- проаналізувати теоретичні відомості про формування логів, їхню структуру (стандарти RFC 3164 та RFC 5424), роль у забезпеченні інформаційної безпеки та особливості логування на пристроях Cisco IOS та MikroTik RouterOS;
- обґрунтувати вибір апаратного забезпечення та програмних інструментів для збору, обробки й аналізу журналів подій;
- дослідити механізми передачі, формати, захист потоків даних за допомогою TLS-шифрування та особливості обробки мережних логів;
- спроектувати архітектуру та побудувати структурну модель гетерогенної мережної інфраструктури для централізованої передачі телеметрії;
- розгорнути та налаштувати аналітичний комплекс ELK Stack на базі серверної платформи під управлінням операційної системи Linux Ubuntu Server;

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				8
Змн.	Арк.	№ докум.	Підпис	Дата		

- налаштувати незалежні конвеєри Logstash із grok-фільтрами для нормалізації повідомлень у формат JSON, а також аналітичні панелі, індекси й фільтрацію в Kibana;
- провести практичні експерименти на реальному фізичному обладнанні із симуляції мережних атак: на протокол OSPF через пакет FRR, DHCP Starvation через утиліту Yersinia та підробки логів за допомогою Python/Scapy;
- оцінити ефективність виявлення інцидентів системою моніторингу Kibana та запропонувати й протестувати технічні методи протидії їм.

Об’єктом дослідження є процес забезпечення інформаційної безпеки та моніторингу подій у локальних комп’ютерних мережах.

Предметом дослідження є методи та засоби централізованого збору, агрегації та аналізу логів мережних пристроїв на базі стеку ELK із налаштуванням механізмів реагування на інциденти безпеки.

Дана робота спрямована на створення ефективної підсистеми централізованого логування та моніторингу мережних пристроїв, що забезпечить побудову надійної архітектури аудиту, підвищить рівень захищеності інформаційних ресурсів від потенційних атак та дозволить своєчасно виявляти й нейтралізувати загрози в реальному часі.

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				9
Змн.	Арк.	№ докум.	Підпис	Дата		

РОЗДІЛ 1 ТЕОРЕТИЧНІ ОСНОВИ ЛОГУВАННЯ, СТРУКТУРА ЖУРНАЛІВ ПОДІЙ ТА МОЖЛИВОСТІ ЇХ ВИКОРИСТАННЯ

1.1 Поняття логування та його роль у забезпеченні інформаційної безпеки

Логування — це процес автоматичного запису подій, дій користувачів, системних повідомлень, або змін стану в інформаційній системі мережному обладнанні або програмному забезпеченні. Основною метою логування є забезпечення моніторингу, аудиту, діагностики, а також використання цих даних у подальшому для розслідування інцидентів інформаційної безпеки.

Лог-файли (журнали подій) є структурованими або текстовими файлами, які можуть містити записи про критично важливі події, зокрема:

- Аутентифікація та авторизація користувачів — записи про вхід користувачів, виявлення невдалих спроб входу, входів з незвичних місць, підозрілої активності.
- Мережні з'єднання — фіксація підключень до певного обладнання, запити до мережних ресурсів, передача даних.
- Системні помилки — записи про внутрішні збої програмного забезпечення чи апаратного забезпечення.
- Спроби несанкціонованого доступу — виявлення сканування портів, перебору паролів, спроб експлуатації вразливостей.
- Зміни конфігурації — журналювання змін у налаштуваннях обладнання або систем, зокрема тих, що можуть вплинути на безпеку.

Логування є одним із елементів системи інформаційної безпеки, що дозволяє організаціям постійно контролювати внутрішні та зовнішні події, які відбуваються в межах інформаційної інфраструктури. Завдяки цьому стає можливим виявлення загроз в реальному часі, аналіз інцидентів безпеки, а також впровадження заходів реагування до того, як шкода буде нанесена.

Однією з функцій логування є виявлення атак у реальному часі. Наприклад, інтеграція логів із SIEM-системами (такими як ELK Stack, Splunk, Graylog) або логічними обробниками які дозволяє миттєво ідентифікувати підозрілу активність

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		10

Коли інцидент вже стався, логування дозволяє розслідувати даний випадок. Лог-файли містять хронологічну інформацію про дії користувачів, мережну активність, зміни в конфігураціях систем і навіть дії з боку зловмисників. Завдяки цьому фахівці з кібербезпеки можуть відновити послідовність подій і зрозуміти, як саме була здійснена атака, що постраждало під час інциденту, і яким чином не допустити подібну атаку в майбутньому.

Відстеження хронології подій — одна з найважливіших властивість логів. У складній інфраструктурі події можуть бути взаємопов'язані: зміна налаштувань на маршрутизаторі могла спричинити відмову сервісу, що відкрило новий вектор атаки. Логи дозволяють простежити ці залежності, сформувати повну картину й виявити першопричину проблеми, для її подальшого вирішення.

Ведення журналів також є вимогою стандартів безпеки та аудитів, таких як ISO/IEC 27001, GDPR, PCI DSS. Вони вимагають від організацій мати доказову базу того, що інформація обробляється відповідно до визначених політик безпеки. Логи дозволяють задокументувати відповідність цим вимогам та бути готовими до перевірок з боку регуляторів[13].

Для того щоб логи були ефективним інструментом у сфері безпеки, вони повинні відповідати певним вимогам до якості:

- **Якість** — лог має містити важливу та корисну інформацію: що сталося, де, коли, з ким і як. Наприклад, повідомлення типу «Помилка 5» не дає уявлення про суть проблеми, на відміну від «Не вдалося підключитися до БД: відмовлено у доступі (користувач admin)».
- **Повнота** — логування повинно охоплювати всі ключові компоненти та події. Недостатньо просто зафіксувати, що сталася помилка - потрібно знати, що передувало їй і які були наслідки.
- **Цілісність** — логи мають бути захищені від несанкціонованої модифікації. Їх підробка може приховати факт атаки або спроби втручання. Для цього часто використовують цифрові підписи, контрольні суми, шифрування.

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		11

- Актуальність — події мають фіксуватись у момент або майже відразу після їх виникнення. Затримка в логуванні унеможливорює оперативне реагування на інциденти.

Також, логування відіграє роль у моніторингу працездатності апаратного забезпечення, фіксуючи такі події, як падіння з'єднань на інтерфейсах чи відключення портів. Це дає змогу відстежувати будь-які маніпуляції з конфігурацією — від оновлення списків контролю доступу (ACL) до коригування параметрів маршрутизації, ідентифікуючи при цьому конкретного користувача, який вніс зміни. Такий підхід забезпечує наскрізний аудит роботи ключових мережних служб, зокрема NAT, DHCP, VPN та VLAN, де чітко розуміння того, хто і коли змінив налаштування. Для забезпечення цілісності та зручності подальшого аналізу всі ці дані централізовано зібрані за допомогою стандартних протоколів, як Syslog або SNMP, і передані на сервери обробки логів.

Не менш важливим є виявлення атак на мережу. Сучасне мережне обладнання здатне розпізнавати різні типи загроз, зокрема DoS/DDoS-атаки, ARP-spoofing, сканування портів[18], порушення в роботі протоколів маршрутизації. Такі події логуються із зазначенням усіх важливих технічних деталей.

У сучасних ІТ-системах використовується широкий спектр інструментів для збору та обробки логів. Найбільш базовим є протокол Syslog, який підтримується більшістю Unix-подібних операційних систем та мережного обладнання. Він дозволяє централізовано збирати лог-файли з різних пристроїв і систем[10].

Для глибшого аналізу логів використовуються SIEM-системи, які об'єднують функції збору, зберігання, аналізу, пошуку й візуалізації даних. Одним із таких прикладів є стек ELK. Також широко застосовуються інструменти на кшталт Splunk і Graylog, що дозволяють будувати складні аналітичні інтерактивні панелі управління (dashboard), створювати алерти та автоматизувати реагування на інциденти.

Крім того, для глибшої перевірки мережної активності застосовуються мережні аналізатори — такі як Wireshark або Zeek. Вони дозволяють перехоплювати та детально аналізувати трафік, доповнюючи логування прямим моніторингом мережної взаємодії.

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		12

1.2. Механізми логування та передача логів

Для передачі логів у мережі можуть використовуватися протоколи TCP, та UDP, залежно від вимог до надійності та швидкості передачі даних. TCP забезпечує надійну доставку кожного повідомлення, гарантує їх цілісність і порядок, що важливо для критичних логів, пов'язаних із безпекою або аудитом. Натомість UDP працює швидше і має менші накладні витрати, але не гарантує доставку чи порядок повідомлень, що робить його придатним для передачі великої кількості не критичних логів або для систем, де важлива мінімальна затримка. Отже, вибір між TCP і UDP для логування залежить від конкретних потреб і пріоритетів системи — надійність або швидкість.

TCP — це один із основних транспортних протоколів, він відповідає за надійну передачу даних і розташований між прикладним та мережним рівнями [1]. Його головною особливістю є встановлення передачею «віртуального з'єднання» між відправником і отримувачем, що дає змогу гарантувати доставку кожного пакету даних без помилок і у правильному порядку. У контексті логування TCP забезпечує, що кожне повідомлення буде доставлено та зафіксовано без втрат. Якщо якийсь пакет втрачається чи пошкоджується, TCP автоматично ініціює його повторну передачу. Завдяки цьому можна бути впевненим у повноті та цілісності зібраних логів, що є надзвичайно важливим для аналізу безпеки, розслідування інцидентів або відновлення роботи системи.

TCP забезпечує збереження порядку та хронології передачі повідомлень, що є важливим для аналізу логів і розслідування подій. Для цього протокол використовує процедури встановлення (трьох-етапне рукостискання) та завершення з'єднання, а також механізми контролю потоку даних. Водночас через таку складність TCP створює додаткове навантаження на мережу та систему й збільшує затримку передачі [1].

UDP — спрощує передачу даних, доставляючи пакети (технічно, корисні навантаження) до їх кінцевого пункту без попереднього встановлення непрямого зв'язку [2]. Це означає, що UDP не гарантує доставку повідомлень, їх цілісність чи порядок надходження. Відправник просто відправляє пакети (датаграми) і не очікує

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		13

підтвердження. Такий підхід дозволяє зменшити затримки і навантаження на систему, що робить UDP ідеальним для передачі великих обсягів даних у реальному часі, де швидкість важливіша за абсолютну надійність. У сфері логуювання UDP часто використовується для передачі не критичних, але масових повідомлень, які надсилаються без перешкод і затримок, наприклад коли для тестування мережі вмикають всі логи навіть неважливі для подальшої обробки. Протокол широко застосовується в мережних пристроях для відправки системних логів (Syslog) через порт 514.

Таблиця 1.1

Порівняння TCP та UDP у логуюванні

Характеристика	TCP	UDP
Тип з'єднання	Орієнтований на з'єднання	Без з'єднаний
Надійність доставки	Висока (з підтвердженням і повторною передачею)	Низька (без гарантій)
Порядок доставки	Гарантується	Не гарантується
Навантаження на систему	Високе	Низьке
Затримка	Вища	Мінімальна
Приклади використання	Аудит, безпека, критичні події конфіденційні дані з шифруванням	Моніторинг, менш важливі логи, SNMP traps

Розглядаючи застосування TCP у системах логуювання, ключовим фактором є його здатність гарантувати доставку кожного окремого запису та збереження їхнього хронологічного порядку, оскільки TCP є протоколом із встановленням з'єднання. Такий підхід є безальтернативним у сценаріях, де логуються бізнес-критичні чи юридично значущі події. Типовим прикладом є фінансовий аудит та транзакційні журнали, де втрата хоча б одного запису про переказ коштів чи зміну балансу є критичною для бізнесу. Аналогічно, TCP використовується для журналів безпеки та доступу — фіксації входу в систему, зміни паролів чи спроб доступу до конфіденційних даних. Якщо систему буде атаковано, а логи безпеки загубляться в

мережі, розслідувати інцидент або довести факт зламу буде неможливо. Окрім того, TCP незамінний для систем, що підпадають під суворі регуляторні вимоги (GDPR, HIPAA), де закон зобов'язує компанії зберігати безперервний і точний цифровий слід дій. Також для забезпечення конфіденційності та цілісності логів у публічних або незахищених мережах рекомендується використовувати передачу по TCP із додатковим шифруванням, наприклад, syslog over TLS, що дозволяє захистити інформацію від перехоплення або зміни під час транспортування[11].

На противагу цьому, використання UDP у логуванні підпорядковане ідеї максимальної швидкодії додатка. Це усуває будь-який вплив процесу логування на продуктивність основної системи: навіть якщо центральний збірник логів повністю вийде з ладу, додаток цього не помітить і продовжуватиме працювати без жодних затримок. Цей протокол ідеально підходить для високонавантажених систем та збору телеметрії, де обсяги повідомлень колосальні, а втрата невеликої частини даних ні на що не впливає. Яскравим прикладом є збір метрик продуктивності через системи на зразок StatsD: якщо додаток щосекунди відправляє дані про завантаження процесора чи пам'яті, то втрата 1% цих пакетів під час мережного збою взагалі не змінить загальний графік моніторингу в Grafana. Також UDP є стандартом для IoT-пристроїв та датчиків, які працюють від батарейок та мають слабкі процесори, оскільки підтримання постійної TCP-сесії для них є надто дорогим з погляду енерговитрат.

Таким чином, фінальний вибір протоколу в архітектурі логування завжди зводиться до класифікації самих даних та визначення пріоритетів системи. Якщо лог містить інформацію про помилку рівня Critical або Error, фіксує зміну конфігурації чи фінансову операцію — розробники обирають TCP. Якщо ж мова йде про повідомлення рівня Info чи Debug, трасування запитів у мікросервісах або періодичні метрики, пріоритет зміщується в бік швидкодії, і логування проектується через UDP.

При цьому у сучасних масштабованих системах часто застосовують комбінований або гібридний підхід. Додаток на максимальній швидкості відправляє всі типи логів по UDP на спеціальний легковажний агент (наприклад, Fluent Bit чи

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		15

Logstash), який запущений на тій самій машині. А вже цей локальний агент акумулює дані, сортує їх і надійно, без поспіху, передає на центральний сервер зберігання по протоколу TCP з шифруванням, поєднуючи найкращі сторони обох підходів. У підсумку, вибір протоколу і порту для логування завжди слід робити, враховуючи цей баланс між надійністю, швидкістю та безпекою передачі даних під конкретні завдання системи.

1.3. Особливості логування на мережних пристроях

Маршрутизатори та комутатори Cisco підтримують детальне логування за допомогою протоколу Syslog, що дозволяє відстежувати широкий спектр подій. Зокрема, у логах можуть відображатися події, пов'язані з оновленням маршрутів у протоколах OSPF, BGP, EIGRP, станом мережних інтерфейсів (включно з їх увімкненням, вимкненням, помилками або перевантаженнями), а також повідомлення про зміни у конфігурації пристрою, що дає змогу відслідковувати дії адміністраторів. Окрім цього, у логах фіксуються події, пов'язані з роботою списків контролю доступу (ACL) та трансляції мережних адрес (NAT).

У MikroTik RouterOS реалізована власна система логування, що має гнучкі можливості збирання, фільтрації та форматування логів. MikroTik не обмежується стандартним Syslog, а пропонує локальний журнал, консольний вивід, запис у файл, а також можливість пересилати логи на зовнішні Syslog-сервери [7].

Особливості системи логування MikroTik:

- Повідомлення групуються за різними тематичними групами, наприклад firewall, system, ospf, що допомагає легко фільтрувати і аналізувати потрібні категорії подій.
- Логування підтримує правила фільтрації, які дозволяють вибирати, які саме події і з яким рівнем важливості будуть записуватися або відправлятися.
- Для пересилки логів через Syslog можна налаштувати не тільки рівень важливості повідомлень. Це дає можливість інтегрувати MikroTik у будь-які існуючі системи моніторингу.

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		16

Таким чином, різні виробники мережного обладнання пропонують власні механізми та особливості логування, що відображаються у підтримці унікальних форматів повідомлень, можливостей фільтрації, налаштуванні рівнів важливості і методів передачі логів. Це дозволяє побудувати ефективну систему моніторингу та аудиту мережі, яка відповідатиме конкретним вимогам організації.

1.4. Структура логів мережних пристроїв

В архітектурі пристроїв Cisco реалізовано ієрархічну систему з восьми рівнів важливості, яка потрібна для пріоритезації обробки мережних подій та інцидентів безпеки. Проте варто враховувати певну специфіку найвищого рівня — 0 (emergencies). Попри те, що він сигналізує про найбільш критичні стани, за яких система стає фактично непридатною для експлуатації, такі повідомлення зазвичай не фіксуються на зовнішніх syslog-серверах. Це пояснюється самим характером аварії: коли виникає збій такого масштабу, пристрій часто втрачає здатність підтримувати мережне з'єднання або виконувати будь-які стандартні операції з передачі даних. У контексті Cisco IOS рівень 0 можна вважати прямим аналогом критичних повідомлень про аварійне завершення роботи в UNIX-системах, оскільки він вказує на фатальну нестабільність, за якої механізми дистанційного моніторингу просто перестають функціонувати[6].

Таблиця 1.2

Шкала важливості формат RFC 3164

Рівень	Назва	Опис
0	emergency	система не придатна до роботи
1	alert	Необхідна негайна дія
2	critical	Критичні умови
3	error	Умови помилки
4	warning	Попереджувальні умови
5	notification	Нормальні, але значимі умови
6	informational	Інформаційні повідомлення
7	debugging	Повідомлення для налагодження

На Cisco IOS-пристроях (маршрутизаторах і комутаторах) syslog-повідомлення мають таку базову структуру:

[timestamp]: %<FACILITY>-<SEVERITY>-<MNEMONIC>: <description>[3]

- timestamp: дата та час події (за замовчуванням у форматі Mmm dd:hh:mm:ss[.sss])
- FACILITY: підсистема або модуль, що згенерував повідомлення
- SEVERITY: цифровий рівень важливості від 0 до 7
- MNEMONIC: короткий опис, що відображає тип події
- description: текстовий опис події з додатковими параметрами

Окремої уваги тут можна приділити Facility-кодам та мнемоніки. У системі логування Cisco IOS FACILITY — це ідентифікатор, який вказує на джерело або підсистему, що згенерувала повідомлення. Це може бути апаратний пристрій, програмний модуль або протокол.

Таблиця 1.3

Поширені facility коди Cisco IOS

Facility-код	Опис
CONFIG	Зміни конфігурації пристрою
LINEPROTO	Протоколи каналного рівня інтерфейсів
BGP	Повідомлення протоколу BGP
OSPF	Повідомлення протоколу OSPF
AAA	Аутентифікація, авторизація та облік

У системі логування Cisco IOS мнемоніка (mnemonic) — це короткий код, який ідентифікує тип події або повідомлення. Вона слугує для швидкої класифікації та розуміння суті повідомлення без необхідності читати опис[3].

На пристроях MikroTik RouterOS формат syslog-повідомлень дещо відрізняється від Cisco IOS і має власну структуру. Базовий формат повідомлення виглядає так:

<timestamp> <topic> <message>

- timestamp — дата та час події у різних форматах, залежно від налаштувань.

- topic — одна або кілька тем (topics), що класифікують повідомлення за підсистемами та рівнем важливості.
- message — текстове повідомлення, яке описує подію.

Таблиця 1.4

Поширені мнемонік Cisco IOS

Мнемоніка	Опис
UPDOWN	Стан інтерфейсу змінився (увімкнено/вимкнено)
CHANGED	Статус інтерфейсу було змінено
CONFIG_I	Зміни конфігурації, внесені через консоль або віддалено
SEC_LOGIN	Успішна або невдала спроба входу в систему
LINEPROTO	Зміни в протоколі канального рівня інтерфейсу
LOGIN_SUCCESS	Успішний вхід користувача в систему

У системі логування MikroTik RouterOS структура повідомлень базується на концепції topics (тем), які виконують роль класифікаторів подій та допомагають у впорядкуванні журналів. Кожне повідомлення може мати одну або кілька тем, що дозволяє точно визначити як джерело, так і характер події, що відбулася. Такий підхід забезпечує гнучкість в аналізі, фільтрації, сортуванні логів. Це відрізняє MikroTik від систем, таких як Cisco IOS, де для класифікації повідомлень застосовується виключно числове позначення рівнів важливості[7].

Одним із важливих аспектів тем у MikroTik є рівні важливості, які відповідають загальноприйнятій класифікації за форматом RFC 3164. На відміну від Cisco IOS, де використовуються лише цифри, у MikroTik важливість подій позначається текстовими ключовими словами, такими як critical, error, warning, info, debug тощо. Таке текстове представлення є більш інтуїтивно зрозумілим і зручним для сприйняття. Детальний опис кожного рівня та його відповідність стандартній класифікації подано в таблиці 1.2. Це значно полегшує ідентифікацію серйозності подій під час аналізу логів.

Окрему категорію становлять так звані незалежні теми (facility-independent topics), які класифікують повідомлення за їх важливістю або загальним характером, незалежно від того, яка саме підсистема чи служба їх згенерувала[7].

Незалежні теми

Тема	Опис
critical	Критичні повідомлення
error	Повідомлення про помилки
warning	Попередження
info	Інформаційні повідомлення
debug	Повідомлення для налагодження
packet	Інформація про отримані/відправлені пакети
raw	Сирові дані отриманих/відправлених пакетів

Крім того, існують теми, пов'язані з підсистемами (facility-specific topics), які класифікують повідомлення відповідно до джерела або функціонального модуля, що їх згенерував. Ці теми дають змогу адміністраторам точно визначити, яка частина системи стала джерелом певної події, що є важливим для моніторингу, усунення несправностей та проведення аудиту.

Наприклад, повідомлення може мати теми firewall, info, що вказує на інформаційне повідомлення, пов'язане з брандмауером.

У RouterOS адміністратори можуть гнучко налаштовувати логування, визначаючи теми та дії, як-от збереження в пам'яті, запис на диск, надсилання на пошту чи syslog-сервер. Це дає змогу адаптувати логування до різних задач моніторингу й аудиту.

Для зовнішньої інтеграції підтримуються формати логів: власний, bsd-syslog (RFC 3164) та cef (для SIEM-систем), що забезпечує сумісність із платформами на кшталт Graylog, Splunk чи ELK Stack.

Висновки до першого розділу

У першому розділі було здійснено комплексний аналіз теоретичних засад логування як фундаментального інструменту для забезпечення інформаційної безпеки, безперервного аудиту та моніторингу сучасної мережної інфраструктури. Визначено ключові вимоги до якості, повноти, цілісності та актуальності журналів

подій для забезпечення надійної доказової бази під час розслідування кіберінцидентів.

Проведено детальний порівняльний аналіз транспортних протоколів TCP та UDP у контексті транспортування телеметрії. Обґрунтовано, що вибір між ними є компромісом між швидкістю передачі та гарантованою доставкою: протокол UDP є оптимальним для масових потоків інформаційних повідомлень, тоді як TCP у поєднанні з TLS є обов'язковим для критичних подій безпеки та систем, що підпадають під суворі регуляторні стандарти.

Особливу увагу приділено дослідженню специфіки формування, фільтрації та структурування журналів подій на мережному обладнанні провідних виробників Cisco IOS та MikroTik RouterOS. Досліджено стандарт протоколу Syslog RFC 3164/5424, ієрархію рівнів важливості, а також унікальну концепцію тематичних груп та специфічних вендорних полів у RouterOS.

Отримані теоретичні відомості щодо структури syslog-повідомлень, рівнів важливості та vendor-specific полів таких як FACILITY, MNEMONIC у Cisco або topics у MikroTik мають пряме практичне значення. Надалі ці структури безпосередньо використовуються для розробки grok-шаблонів і регулярних виразів у конвеєрах Logstash з метою нормалізації різноманітного тексту та його перетворення в уніфікований формат JSON. Окрім цього, класифікація рівнів важливості та специфічних маркерів вендорів є базою для побудови індексів Elasticsearch та налаштування аналітичних інструментів обробки логів, створення правил кореляції подій, побудови інтерактивних панелей і систем оперативного виявлення аномалій та мережних атак у Kibana.

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				21
Змн.	Арк.	№ докум.	Підпис	Дата		

РОЗДІЛ 2 ВИБІР МЕРЕЖНИХ ПРИСТРОЇВ ТА ІНСТРУМЕНТІВ ДЛЯ ЗБОРУ, ОБРОБКИ Й АНАЛІЗУ ЛОГІВ, ПРОЄКТУВАННЯ МЕРЕЖІ

2.1 Вибір мережних пристроїв

Маршрутизатор Cisco 2821 Integrated Services Router належить до серії Integrated Services Router (ISR) та призначений для побудови корпоративних мереж малого і середнього масштабу. Пристрій поєднує функції маршрутизації, комутації, захисту трафіку та віддаленого доступу, що робить його придатним для використання як центральний вузол мережної інфраструктури. Завдяки підтримці операційної системи Cisco IOS маршрутизатор забезпечує стабільну роботу великої кількості мережних сервісів, включаючи динамічну маршрутизацію, VPN-з'єднання, контроль доступу та централізоване логування подій.



Рисунок 2.1 – Маршрутизатор Cisco 2821

Архітектура Cisco 2821 побудована на модульному принципі, що дозволяє розширювати функціональні можливості пристрою шляхом встановлення додаткових інтерфейсних модулів. У досліджуваній конфігурації маршрутизатор оснащений двома гігабітними Ethernet-інтерфейсами, чотирма FastEthernet-портами комутації та двома serial-інтерфейсами. Таке поєднання інтерфейсів забезпечує можливість одночасного підключення локальних сегментів мережі, зовнішніх маршрутизаторів та емуляції WAN-з'єднань у лабораторному середовищі. Наявність модулів HWIC та WIC також дозволяє адаптувати пристрій до різних сценаріїв використання без заміни основного обладнання.

Маршрутизатор працює під керуванням Cisco IOS 15.1(4)M1 з образом Advanced Enterprise Services, який підтримує широкий спектр мережних технологій корпоративного рівня. Серед них підтримка протоколів OSPF, BGP, EIGRP, ACL, NAT, QoS, SNMP, Syslog та IPsec VPN. У межах досліджуваного середовища маршрутизатор використовує протокол OSPF для обміну маршрутною інформацією

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		22

між мережними сегментами. Для підвищення рівня безпеки автентифікація OSPF реалізована за допомогою MD5 message-digest ключів, що дозволяє захистити процес обміну маршрутами від несанкціонованого втручання.

Також варто приділити можливостям моніторингу та журналювання подій. Cisco 2821 підтримує централізовану передачу логів через Syslog-сервер, що дозволяє інтегрувати його з системами аналізу та моніторингу мережної безпеки. У досліджуваній конфігурації маршрутизатор надсилає системні повідомлення на віддалені сервери логування через UDP-порт 5515. Це дає змогу фіксувати зміни стану інтерфейсів, події маршрутизації, спроби доступу та інші мережні події для подальшого аналізу у системах типу ELK Stack або інших систем моніторингу.

У межах даної дипломної роботи Cisco 2821 виконує функції центрального маршрутизатора мережі, забезпечуючи маршрутизацію між локальним сегментом та зовнішніми вузлами. Через інтерфейс GigabitEthernet0/0 маршрутизатор підключений до іншого мережного пристрою, а через GigabitEthernet0/1 обслуговує локальну мережу з адресним простором 192.168.41.0/24. Крім маршрутизації, пристрій реалізує DHCP-сервіс для автоматичної видачі IP-адрес клієнтам локальної мережі, що дозволяє централізувати процес мережного адміністрування.

Таблиця 2.1

Основні характеристики Cisco 2821 ISR

Характеристика	Значення
Модель	Cisco 2821 ISR
Операційна система	Cisco IOS 15.1(4)M1
Серія	Cisco ISR 2800
Оперативна пам'ять	247 МБ DRAM
Flash-пам'ять	250 МБ CompactFlash
Gigabit Ethernet	2 порти
FastEthernet	4 порти
Serial-інтерфейси	2 порти
Модульність	HWIC/WIC модулі

Використання Cisco 2821 у лабораторному середовищі дозволило моделювати реальні корпоративні мережі із застосуванням сучасних методів

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		23

маршрутизації, моніторингу та контролю доступу. Незважаючи на належність до попередніх поколінь ISR-маршрутизаторів, дана модель залишається актуальною для навчальних та дослідницьких задач завдяки стабільності роботи, підтримці IOS та широким можливостям інтеграції з системами мережного моніторингу.

Серед сучасних аналогів Cisco 2821 варто виділити платформи Cisco Catalyst 8200 і 8300 Series Edge, а також Cisco ISR 4331. Вони орієнтовані на новітні корпоративні мережі й підтримують технології SD-WAN, автоматизацію, сегментацію та розширені VPN-механізми. Зокрема, серії Catalyst 8200 та 8300 працюють на базі IOS XE, забезпечуючи багатоядерну обробку пакетів, апаратне шифрування та управління через Cisco DNA Center. Маршрутизатор Cisco ISR 4331 є популярним рішенням для філій, що поєднує високу продуктивність із сервісами безпеки, QoS, NetFlow та Syslog. Завдяки високій пропускній здатності, телеметрії та інтеграції з хмарними сервісами, всі ці пристрої значно краще підходять для сучасного моніторингу та швидкого реагування на інциденти, ніж застаріла модель 2821.

Таблиця 2.2

Порівняння сучасних аналогів маршрутизаторів

Модель	Ethernet-інтерфейси	Продуктивність	Підтримка SD-WAN	VPN та безпека	Основне призначення
Cisco 2821 ISR	2 GE + 4 FE	Низька	Ні	IPsec, ACL	Навчальні та базові корпоративні мережі
Cisco ISR 4331	3 GE	Середня	Частково	IPsec, Zone-Based Firewall	Філії та корпоративні мережі
Cisco Catalyst 8200	2–4 GE	Висока	Так	SD-WAN, IP-Sec, Umbrella	Сучасні edge-мережі
Cisco Catalyst 8300	До 8 GE + модулі	Дуже висока	Так	SD-WAN, сегментація, DPI	Центральні вузли та дата-центри

Маршрутизатор MikroTik RB5009UG+S+ є високопродуктивним мережним пристроєм корпоративного класу, який працює під керуванням операційної системи RouterOS v7 та призначений для реалізації функцій маршрутизації, фільтрації трафіку, організації VPN-з'єднань, моніторингу мережі та підтримки динамічних протоколів маршрутизації. Даний пристрій побудований на сучасній ARM64-

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		24

платформі та оснащений процесором який має 4 ядра, що забезпечує високу швидкість навіть при значному мережному навантаженні. У рамках даної роботи використовується RouterOS версії 7.5, яка належить до сучасної гілки RouterOS v7 і містить підтримку WireGuard, оновлені механізми OSPF/BGP, покращену систему routing filter, а також розширені можливості журналювання та моніторингу мережних подій.



Рисунок 2.2 – Маршрутизатор MikroTik RB5009UG+S+

RB5009UG+S+ орієнтований на використання у корпоративних мережах, дата-центрах, ISP-інфраструктурі та навчально-дослідницьких лабораторіях. Пристрій підтримує апаратне прискорення маршрутизації, VLAN, MPLS, VPN-технології та роботу з динамічними протоколами маршрутизації, зокрема OSPF і BGP. У досліджуваному стенді маршрутизатор використовується для організації OSPF-маршрутизації між сегментами мережі, передачі логів до віддаленого syslog/Elastic-сервера. Для підвищення безпеки обміну маршрутною інформацією використовується автентифікація OSPF MD5.

Однією з ключових особливостей RB5009UG+S+ є наявність великої кількості високошвидкісних мережних інтерфейсів. Пристрій має вісім гігабітних Ethernet-портів та один SFP+ порт із підтримкою швидкості до 10 Gbit/s, що дозволяє використовувати його як центральний вузол маршрутизації або агрегації трафіку. Завдяки підтримці Jumbo Frame та збільшеному значенню L2MTU пристрій може ефективно працювати у високонавантажених мережах.

Завдяки поєднанню високої продуктивності, сучасної програмної платформи та можливостей масштабування, MikroTik RB5009UG+S+ є ефективним

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		25

інструментом для побудови мережної інфраструктури. Реалізація передачі системних повідомлень і подій на віддалені сервери (наприклад, syslog або ELK Stack) дозволяє централізовано аналізувати стан мережі, помилки маршрутизації та інциденти безпеки, що є важливим для сучасних систем виявлення загроз. Використання даного пристрою в межах дипломної роботи дає змогу моделювати реальні сценарії функціонування корпоративних мереж, досліджуючи механізми журналювання та оперативного реагування на мережні події в умовах, максимально наближених до реальної експлуатації.

Таблиця 2.3

Основні технічні характеристики MikroTik RB5009UG+S+

Характеристика	Значення
Модель	RB5009UG+S+
Операційна система	RouterOS v7
Архітектура	ARM64
Кількість ядер CPU	4
Оперативна пам'ять	1 GB
Вбудована пам'ять	1 GB NAND
Ethernet порти	8 × Gigabit Ethernet
SFP+ порти	1 × 10G SFP+
Підтримка VPN	IPsec, WireGuard, L2TP, PPTP, SSTP
Максимальний L2MTU	9796 bytes
Журналювання / Syslog	Підтримується
Призначення	Корпоративні мережі, ISP, лабораторні стенди

Вибір комбінації Cisco IOS та MikroTik RouterOS обумовлений принципово різними підходами цих платформ до формування й структурування телеметрії, що дозволяє проаналізувати поведінку систем логування в умовах гетерогенного мережного середовища. Cisco IOS використовує класичний, суворо стандартизований підхід до syslog-повідомлень із фіксованими шаблонами, чіткою структурою засобів і мнемонік. Натомість MikroTik RouterOS базується на гнучкій концепції

тематичних груп та динамічних префіксів, де одна подія може одночасно маркуватися кількома тегами. Крім того, суттєві відмінності існують у механізмах фільтрації: Cisco орієнтована на традиційні рівні важливості, тоді як RouterOS пропонує правила логування для динамічного перенаправлення комбінацій тем на зовнішні сервери.

Таким чином, інтеграція настільки відмінних форматів в єдину підсистему на базі ELK stack дозволяє практично вирішити проблему нормалізації різнорідних сирих даних. Це дає змогу розробити універсальні конвеєри обробки, які зводять унікальні поля кожного вендора до єдиної JSON-схеми для подальшого аналізу та виявлення аномалій у Kibana.

2.2 Інструменти для збору та обробки даних

У рамках даної курсової роботи для збору, обробки та візуалізації логів було розгорнуто стек Elastic Stack (відомий як ELK Stack) на базі Ubuntu Server. До складу цього стека входять:

- Elasticsearch – пошукова та аналітична система, яка зберігає та індексує дані;
- Logstash — інструмент для збору, обробки та пересилання логів;
- Kibana — веб-інтерфейс для візуалізації та аналізу даних, що зберігаються в Elasticsearch;

У межах проєктованої підсистеми ці компоненти не працюють ізольовано, а утворюють єдиний конвеєр обробки телеметрії. Взаємодія між ними побудована за лінійною схемою: мережне обладнання виступає джерелом даних, Logstash виконує роль шлюзу обробки та нормалізації, Elasticsearch слугує центральним аналітичним сховищем, а Kibana забезпечує рівень представлення даних для адміністратора безпеки. Такий підхід розділяє обчислювальні процеси збору, індексації та візуалізації для того щоб кожен із модулів можна було замінити та налаштувати незалежно від інших.

Elasticsearch — це високопродуктивна, масштабована пошукова та аналітична система з відкритим кодом, яка дозволяє зберігати, індексувати та здійснювати

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				27
Змн.	Арк.	№ докум.	Підпис	Дата		

швидкий пошук по великих об'ємах даних. Вона є основою стеку Elastic Stack і використовується у багатьох сферах, від моніторингу журналів безпеки до аналітики бізнес-даних у режимі реального часу. Elasticsearch побудований на основі бібліотеки Apache Lucene і працює як розподілена база даних документів, де всі дані зберігаються у форматі JSON.

Основні компоненти:

- Індекс (Index) — аналог бази даних, де зберігаються документи;
- Документ (Document) — одиниця збереженої інформації (наприклад, лог або запис про подію);
- Шард (Shard) — частина індексу, яка дозволяє масштабувати зберігання на кілька вузлів;
- Репліка (Replica) — копія шару, яка забезпечує відмовостійкість та балансування навантаження.

При надходженні даних, наприклад із Logstash, кожен лог-трук перетворюється на документ JSON і відправляється до Elasticsearch, де він автоматично індексується для подальшого пошуку. Користувач може надсилати складні запити через REST API або за допомогою Kibana, яка слугує візуальним інтерфейсом для роботи з Elasticsearch.

Архітектурно система Elasticsearch посідає центральне місце у взаємодії компонентів, не приймаючи логи безпосередньо від маршрутизаторів Cisco чи MikroTik, оскільки сирий текстовий формат Syslog є неструктурованим. Замість цього Elasticsearch очікує винятково HTTP POST-запити із готовими JSON-документами від Logstash. Після індексації даних Elasticsearch відкриває свій REST API порт 9200 для Kibana, яка надсилає аналітичні запити для побудови часових зрізів, мінімізуючи навантаження на дискову підсистему за рахунок використання заздалегідь налаштованих індексів.

Elasticsearch володіє можливостями повнотекстового пошуку, які роблять його хорошим інструментом для аналізу логів, подій і не структурованих даних. Він підтримує розширений синтаксис запитів, що включає булеві оператори, регулярні вирази, fuzzy-пошук (розмитий пошук із урахуванням помилок у словах),

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				28
Змн.	Арк.	№ докум.	Підпис	Дата		

score (діапазони) та фільтри. Завдяки цьому користувачі можуть будувати як прості, так і надзвичайно складні запити, що дозволяє знаходити точні збіги або виявляти закономірності в потоках даних. Це особливо корисно в безпекових сценаріях, коли важливо миттєво знаходити критичні події серед мільйонів логів.

Однією з ключових переваг Elasticsearch є робота майже в реальному часі. Хоча повна миттєвість обробки не гарантується через технічну затримку в кілька секунд (через механізм індексації), така швидкість є достатньою для оперативного реагування на інциденти. Це дозволяє використовувати Elasticsearch у системах сповіщення, моніторингу продуктивності, безпеки та експлуатаційного стану інфраструктури.

Функції агрегації та аналітики в Elasticsearch реалізовані через механізм агрегацій, що дозволяє виконувати зведені операції над великими обсягами даних. Наприклад, можливо створювати гістограми, будувати часові діаграми, аналізувати відхилення від середнього, групувати події за певними критеріями та рахувати кількість записів за певний період часу. Такий функціонал є незамінним для аналітиків, фахівців SOC та DevOps-інженерів, які щодня працюють із великими обсягами журналів подій.

Завдяки кластерній архітектурі Elasticsearch надзвичайно легко масштабувати. Його можна запускати як на одному сервері, так і в розподіленому середовищі з десятків або сотень нод. Кожен індекс автоматично розділяється на шардові частини, які розміщуються на різних вузлах, що забезпечує рівномірне навантаження. При зростанні обсягу даних достатньо додати нову ноду до кластера, і Elasticsearch автоматично перерозподіляє дані для оптимальної роботи.

Надійність Elasticsearch досягається через вбудовану систему реплікації. Кожен шард може мати одну або кілька копій (реплік), які зберігаються на інших вузлах. У випадку виходу з ладу однієї машини Elasticsearch автоматично переключається на репліку, забезпечуючи безперервність обробки запитів. Такий підхід дозволяє гарантувати високу доступність сервісу без простоїв, що необхідно для забезпечення нормативних стандартів безпеки та відмовостійкості.

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		29

Ще однією вагомою перевагою є здатність Elasticsearch інтегруватися з широким спектром інструментів збору та транспортування логів. Зокрема, він тісно інтегрується з Logstash і Beats (Filebeat, Metricbeat, Packetbeat, Auditbeat), а також із системами Kafka, Fluentd, Prometheus, Grafana. Це дозволяє створювати повноцінну інфраструктуру моніторингу, де дані можуть надходити з різноманітних джерел: серверів, маршрутизаторів, міжмережних екранів, додатків, контейнерів, хмарних сервісів.

Elasticsearch доступний у декількох редакціях. Базова (Basic) версія є безкоштовною і містить більшість функцій, необхідних для збирання, зберігання та пошуку даних, включно з REST API, базовою візуалізацією в Kibana, HTTPS, RBAC (розмежування прав доступу), моніторингом та попереднім налаштуванням. Для підприємств, які потребують розширених можливостей, доступні платні редакції Platinum та Enterprise. Вони надають розширені механізми безпеки (SAML, LDAP, шифрування даних у стані спокою), автоматичне виявлення аномалій за допомогою машинного навчання, інтеграцію з SIEM, audit-логи та офіційну технічну підтримку.

Важливо відзначити, що Elasticsearch має одну з найбільших спільнот серед open-source проєктів. Його підтримує компанія Elastic, яка регулярно випускає оновлення, вдосконалює функціональність і розвиває навколо нього повноцінну екосистему.

У результаті, Elasticsearch є продуктивним, масштабованим і гнучким інструментом для обробки логів і аналітики. Його функціонал дозволяє створювати складні системи моніторингу, які здатні працювати в режимі майже реального часу, інтегруватися з різними джерелами, масштабуватися відповідно до навантаження та забезпечувати високу надійність. Саме тому в межах даної роботи Elasticsearch був обраний як основа для побудови системи збору, обробки й аналізу журналів подій.

2.3 Мережний потік логів: передача, формат, захист та обробка

Логи (журнали подій) — це записи про події, які можуть бути як структуровані так і неструктуровані, що відбуваються у комп'ютерних системах, мережному

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		30

обладнанні, програмному забезпеченні або службах. Вони є ключовим джерелом даних для аудиту, діагностики, моніторингу, кіберзахисту та форензики. Збір логів передбачає не лише фіксацію подій на джерелі, але й їх централізовану доставку, обробку та зберігання — зазвичай на лог-сервері або в системі моніторингу, або локально.

Існує кілька основних типів логів, кожен з яких виконує певну функцію у контексті моніторингу, аудиту або безпеки інформаційної системи. Системні логи, такі як `syslog` або `journald`, відображають функціонування операційної системи, повідомляючи про запуск служб, оновлення, помилки ядра та інші події на рівні ОС. Вони є базовим джерелом інформації про стан хост-машини.

Мережні логи — формуються маршрутизаторами, комутаторами, міжмережними екранами та іншими мережними пристроями. Вони містять інформацію про проходження мережного трафіку, стан інтерфейсів, спроби підключень, зміни в конфігурації, маршрутизацію, та інші ключові події, які впливають на функціонування мережі.

Аудиторські логи — зосереджуються на фіксації дій користувачів. Вони записують події, пов'язані з авторизацією, входом і виходом із системи, доступом до ресурсів або виконанням адміністративних команд. Вони особливо важливі для забезпечення відповідності політикам безпеки та стандартам (наприклад, ISO/IEC 27001, GDPR)[13].

Застосункові логи — генеруються програмним забезпеченням і зазвичай включають події запуску, помилки, виключення, дії користувача та стан взаємодії з базою даних чи API. Вони критично важливі для розробників і адміністраторів при усуненні несправностей та оптимізації роботи програм.

Окрему категорію становлять безпекові логи, які включають спроби несанкціонованого доступу, виявлення вторгнень, зміни політик безпеки, роботу IDS/IPS систем тощо. Вони дозволяють виявляти підозрілу активність і є основою для реагування на інциденти.

Також виділяють транзакційні логи, які найчастіше зустрічаються в банківських, бухгалтерських та інших системах з високим рівнем відповідальності. Вони

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		31

відстежують кожну зміну стану системи або об'єкта, фіксуючи дії користувача, час, тип операції та інші параметри.

Щодо формату логів, то в більшості випадків вони передаються у текстовому вигляді. Це може бути як звичайний неструктурований текст, так і структуровані формати — наприклад, JSON або XML. Найбільш поширеним стандартом для передачі логів у середовищі UNIX-подібних систем є Syslog (відповідно до RFC 5424). Цей формат включає ключові атрибути події: рівень важливості (наприклад, info, warning, critical), дату і час, ідентифікатор джерела (хост, програма), а також текстове повідомлення про подію. Завдяки своїй універсальності, Syslog підтримується більшістю сучасних систем і мережного обладнання.

У процесі побудови системи моніторингу та реагування на інциденти важливо забезпечити їх надійну, безпечну та своєчасну передачу до центрального серверу обробки. Цей процес включає різні протоколи, способи передачі та підходи до шифрування даних. Нижче наведено основні методи, за допомогою яких логи можуть передаватися через мережу:

- Syslog (UDP/514 або TCP/514) — класичний метод. UDP використовується для швидкої, але ненадійної доставки (без підтвердження доставки), TCP — для гарантованої передачі. Також можна використовувати інші порти, при необхідності, наведено стандартні порти за замовчуванням.
- Syslog over TLS (TCP/6514) — захищена версія, в якій дані передаються зашифрованими через TLS, що гарантує конфіденційність логів під час пересилання.
- Beats (наприклад, Filebeat) — спеціалізовані агенти, що встановлюються на вузли і передають логи у Logstash або Elasticsearch по протоколу Beats over TLS, що забезпечує захист і підтримку повторної відправки при збоях.
- Logstash Forwarder / Lumberjack Protocol — старіший протокол доставки логів, також із підтримкою TLS.
- Kafka / Redis / MQTT — брокери повідомлень, які використовуються в масштабованих або асинхронних системах передачі логів.

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		32

- API (HTTP POST/PUT) — деякі застосунки надсилають логи у JSON-форматі через REST-запити до централізованого лог-сервера.

Передача логів в реальному часі зазвичай здійснюється за допомогою спеціалізованих агентів або демонів, які працюють у фоновому режимі. Ці компоненти (наприклад, rsyslog, syslog-ng, journald, Filebeat, Fluentd, nxlog) постійно стежать за системними або прикладними лог-файлами та, у разі появи нових записів, миттєво пересилають їх на сервер обробки. Агенти можуть працювати як у push-режимі, так і в pull-режимі, залежно від архітектури системи збору логів. Вони також можуть попередньо обробляти записи — виконувати фільтрацію, нормалізацію, додавати метадані або навіть змінювати формат логів (наприклад, перетворення з звичайного тексту в JSON).

У зв'язку з тим, що лог-файли можуть містити конфіденційну інформацію — наприклад, імена користувачів, IP-адреси, шляхи до файлів, параметри команд, або навіть сесійні ключі — надзвичайно важливо забезпечити зашифровану передачу цих даних через мережу. Найбільш поширеним протоколом шифрування є TLS версії 1.2 або 1.3, який застосовується у таких рішеннях, як Syslog over TLS, Filebeat → Logstash, або Logstash → Elasticsearch. Крім шифрування, для підвищення довіри до джерела логів часто використовуються X.509 сертифікати, які дозволяють автентифікувати як клієнта, так і сервер збору логів, щоб унеможливити підробку якось із компонентів, для подальшої підміни.

Більше того, у системах на кшталт Elasticsearch/Kibana передбачена гнучка система керування доступом: автентифікація може здійснюватися через LDAP, SAML, API-ключі, токени або вбудовані механізми. Це дозволяє обмежити коло користувачів, які мають доступ до чутливої інформації, або контролювати права на читання, запис і перегляд логів.

Таким чином, правильно налаштована передача логів повинна гарантувати три ключові властивості:

- Конфіденційність — виключає можливість перехоплення або перегляду логів злоумисниками.
- Цілісність — захищає від змін у логах під час їх передачі.

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		33

- Автентичність — підтверджує достовірність джерела логів і запобігає підміні.

Реалізація таких заходів необхідна для відповідності міжнародним стандартам інформаційної безпеки.

Процес обробки логів у побудованій системі починається з їх генерації на мережних пристроях — маршрутизаторах MikroTik, Cisco. Ці пристрої формують лог-повідомлення, що містять інформацію про маршрутизацію, події безпеки, стани інтерфейсів, спроби підключення тощо. Передача логів до сервера збору та обробки відбувається мережею за допомогою протоколів UDP або TCP.

На сервері Ubuntu з розгорнутою системою обробки логів першим етапом є прийом логів через Logstash. Це гнучкий інструмент для збору, фільтрації й трансформації лог-повідомлень. Logstash може використовувати різноманітні фільтри (grok, kv, dissect), які дозволяють розбивати текстові повідомлення на структуровані поля - наприклад, виділяти IP-адресу, подію, логін користувача або рівень важливості. Також він може додавати теги, класифікувати події за типом, відфільтровувати зайві записи, виконувати анонімізацію та формувати уніфікований формат для подальшого аналізу.

Оброблені логи у форматі JSON передаються в Elasticsearch, що дозволяє індексувати та зберігати великі обсяги структурованих даних. Вона надає інструменти для пошуку, агрегації, фільтрації та аналізу логів у реальному часі. Передача даних здійснюється через HTTP POST-запити.

Для візуалізації даних використовується Kibana — веб-інтерфейс, який дозволяє створювати інформативні графіки, діаграми, таблиці, фільтри та сповіщення. Завдяки Kibana користувачі можуть виявляти аномалії, простежувати зміни, отримувати аналітичні висновки або будувати автоматичні оповіщення про потенційні загрози у простому та зручному форматі, без необхідності перерахувати кожне лог-повідомлення.

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		34

2.4 Проектування мережі

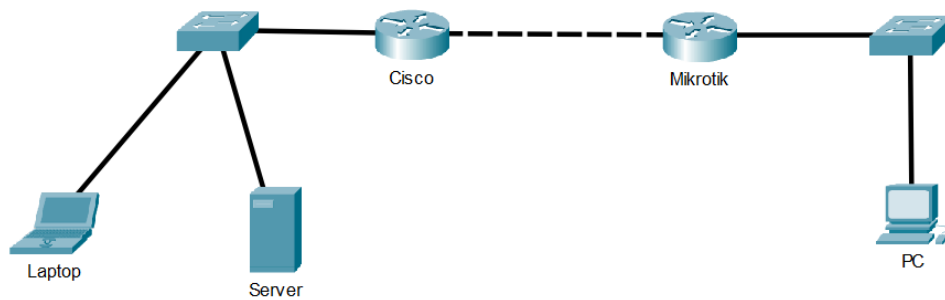


Рисунок 2.3 – Топологія мережі

Інфраструктура мережі умовно поділена на три ключові сегменти: магістраль маршрутизації (транзитну зону), локальні підмережі кінцевих пристроїв та зону централізованого моніторингу, така топологія була обрана для зручності централізованого збору та аналізу логів у межах побудованого стенду. Головною особливістю цієї архітектури є її динамічність: конфігурація мережі, метрики та топологічні параметри навмисно змінювалися в процесі дослідження. Це було необхідним етапом для створення різноманітних, репрезентативних та гетерогенних системних логів, що відображають як штатні процеси конфігурування, так і різні вектори аномальної чи деструктивної мережної активності.

Центральна ланка топології побудована на базі двох маршрутизаторів, які з'єднані між собою безпосередньо через службову транзитну підмережу 193.41.11.0/30. Для забезпечення зв'язку між різними сегментами на пристроях налаштовано протокол динамічної маршрутизації OSPF. Кожен із маршрутизаторів керує власною ізольованою локальною підмережею. Перший локальний сегмент розгорнуто в межах адресного простору 192.168.41.0/24 і призначено для імітації стандартного користувацького середовища. Друга локальна підмережа функціонує в межах адресного простору 194.41.11.0/24 і забезпечує підключення сервісних та інфраструктурних елементів. Така сегментація дозволяє розмежувати трафік та дослідити специфіку журналювання маршрутизаційних змін у кожній підмережі окремо, на обладнанні від різних виробників та дослідити їх особливості, відмінності та проаналізувати їх.

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		35

IP-адресація мережі

Підмережа / Пристрій	Інтерфейс/Мережний адаптер/Шлюз	IP-адреса	Маска підмережі	Префікс
LAN Cisco	192.168.41.0	-	255.255.255.0	/24
LAN transfer	193.41.11.0	-	255.255.255.252	/30
LAN Mikrotik	194.41.11.0	-	255.255.255.0	/24
Cisco 2821	GigabitEthernet0/0	193.41.11.1	255.255.255.252	/30
	GigabitEthernet0/1	192.168.41.1	255.255.255.0	/24
Mikrotik RB5009UG+S+	ether3	193.41.11.2	255.255.255.252	/30
	ether4	194.41.11.1	255.255.255.0	/24
Ubuntu server (ELK stack)	FastEthernet0	192.168.41.5	255.255.255.0	/24
Laptop	FastEthernet0	DHCP		

Зона кінцевих пристроїв містить ноутбук під управлінням операційної системи Linux. Цей пристрій використовується як інструмент для проведення тестів на проникнення, симуляції дій зломисника та штучного наповнення журналів подій специфічними алертами безпеки завдяки встановленому комплексу програмного забезпечення:

- FRR (Free Range Routing) — серверний пакет маршрутизації з відкритим кодом, який дозволяв ноутбуку імітувати поведінку суміжного маршрутизатора та маніпулювати анонсами динамічних маршрутів, що провокувало фіксацію зміни маршрутів у системних логах[16];
- Yersinia CLI — спеціалізована утиліта, призначена для реалізації атак на каналному та мережному рівнях моделі OSI. За її допомогою здійснювалося моделювання атак на протокол DHCP з метою повного виснаження пулу доступних адрес (DHCP starvation attack)[14, 17];
- Python з бібліотекою Scapy — інструментарій для низькорівневого проектування мережних пакетів. За допомогою власних скриптів виконувалася генерація фальшивого трафіку та логів із підміною вихідної IP-адреси (IP spoofing), що дозволило перевірити стійкість систем захисту[12].

Зона централізованого моніторингу та аналізу представлена виділеним сервером під управлінням операційної системи Linux Ubuntu, на якому розгорнуто аналітичний комплекс ELK Stack. Цей сервер виконує роль SIEM-системи, забезпечуючи в реальному часі централізований прийом, агрегацію, індексацію та графічну візуалізацію всіх згенерованих логів і телеметрії, що надходять від маршрутизаторів та дослідницьких станцій під час проведення експериментів.

Фундаментом для збереження та швидкого пошуку зібраної інформації є підсистема Elasticsearch. Конфігурація Elasticsearch є основним етапом налаштування сервера логування, оскільки цей компонент безпосередньо відповідає за зберігання, індексацію та обробку великих обсягів даних. В основі його конфігурації лежать параметри, що визначають місце зберігання даних та журналів подій, їх реплікацію та балансування навантаження, що дозволяє ефективно організувати файлову систему і забезпечити максимально швидкий доступ до індексованої інформації. При налаштуванні важливо, щоб усі шляхи зберігання були вказані коректно, з урахуванням прогнозованого обсягу даних та наявності вільного дискового простору на сервері[5].

Для забезпечення стійкості та масштабованості Elasticsearch окремо налаштовуються параметри, які визначають початкові вузли кластера та мережні інтерфейси. Відкритий доступ до сервісу з будь-яких IP-адрес спрощує його інтеграцію з іншими компонентами системи та клієнтськими застосунками, проте вимагає підвищеної уваги до налаштувань безпеки. Таке мережне конфігурування дозволяє масштабувати систему в майбутньому, додаючи нові вузли без значних змін у поточній архітектурі.

З огляду на це, особливу увагу приділено безпековим аспектам. Впровадження підсистеми безпеки дозволяє забезпечити надійну автентифікацію користувачів, контроль доступу та шифрування даних під час їх передачі між зовнішніми клієнтами. Це є важливим для запобігання несанкціонованому доступу та захисту конфіденційної телеметрії. Практичне використання цифрових сертифікатів для протоколів SSL/TLS підвищує загальний рівень захисту, гарантуючи цілісність і конфіденційність мережних з'єднань.

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		37

Компонент Logstash виконує роль центрального елемента для збору, попередньої обробки та передачі логів у систему зберігання Elasticsearch. Завдяки архітектурі конвеєрів (pipelines), Logstash здатний приймати потоки даних із різних джерел, застосовувати до них необхідні трансформації і фільтрацію, а також ефективно перенаправляти їх до кінцевої точки. Такий підхід дозволяє адаптувати систему під специфіку різних типів пристроїв та різномірних форматів логів[4].

З метою підвищення зручності та гнучкості налаштування логіки обробки, конфігурації Logstash розбиті на кілька окремих незалежних конвеєрів, кожен з яких відповідає за обробку конкретного типу логів. Це дозволяє ізольовано налаштовувати прийом і парсинг повідомлень від різноманітних джерел та застосовувати індивідуальні шаблони розбору повідомлень, правила трансформації, а також додавати специфічні теги для кращої подальшої категоризації даних. Це суттєво підвищує якість даних, що передаються до бази Elasticsearch, і значно полегшує їх подальший аналіз. При цьому для забезпечення безпеки і надійності взаємодії, Logstash налаштовано на роботу з Elasticsearch виключно через захищений протокол HTTPS із обов'язковою автентифікацією[15, 20].

Kibana виконує роль інтерактивного веб-інтерфейсу для візуалізації та аналізу даних, що зберігаються в Elasticsearch. Важливою частиною налаштування Kibana є забезпечення доступу до її інтерфейсу з будь-яких мережних адрес. Це дозволяє аналітикам та адміністраторам з різних систем підключатися до сервісу без додаткових локальних обмежень, що є корисним для організацій із віддаленими робочими місцями чи кількома підрозділами.

Для доступу до веб-інтерфейсу Kibana визначено відповідний мережний порт, а її взаємодія з Elasticsearch захищена протоколом HTTPS та обов'язковою автентифікацією користувачів. Водночас для ефективної діагностики й адміністрування сервісу налаштовано внутрішнє логування у форматі JSON та використання PID-файлу, що дозволяє оперативно моніторити помилки, контролювати системні процеси та безпечно виконувати перезапуски системи.

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		38

Висновки до другого розділу

У другому розділі було виконано обґрунтування вибору апаратних засобів, програмних інструментів та спроектовано топологію мережної інфраструктури для реалізації підсистеми централізованого логування. Як базове мережне обладнання обрано маршрутизатори корпоративного рівня Cisco 2821 та MikroTik RB5009UG+S+, що дає змогу дослідити відмінності формування та обробки журналів подій в операційних системах Cisco IOS та RouterOS v7. Для збору, накопичення й аналітики подій безпеки обрано стек Elastic Stack (ELK) у складі Elasticsearch, Logstash та Kibana. Даний вибір зумовлений високою масштабованістю, продуктивністю обраного програмного забезпечення та його здатністю функціонувати в режимі майже реального часу, формуючи надійну основу для побудови повноцінного SIEM-подібного рішення.

Детально проаналізовано особливості формування мережних потоків журналів подій, зокрема використання загальноприйнятого стандарту Syslog (RFC 5424). З огляду на потенційну наявність конфіденційних даних у телеметрії (таких як IP-адреси чи облікові записи), обґрунтовано впровадження протоколів шифрування TLS версій 1.2 або 1.3 разом із цифровими сертифікатами X.509 для автентифікації джерел і забезпечення конфіденційності даних під час їх передачі каналами зв'язку. Окрім цього, розроблено архітектуру конвеєризації Logstash із розподілом обробки на окремі незалежні конвеєри та застосуванням фільтрів grok, що забезпечує гнучке структурування, нормалізацію та трансформацію різноманітних текстових повідомлень від пристроїв у єдиний аналітичний формат JSON.

На основі проведеного аналізу було успішно спроектовано топологію мережі, здійснено її попереднє моделювання в емуляторі GNS3 та повністю реалізовано діюче тестове середовище на базі фізичного мережного обладнання під управлінням операційної системи Ubuntu Server. Створена модель інфраструктури містить чітко розмежовані зони мережного ядра, централізованого моніторингу та дослідницьких станцій під управлінням Linux. Інтеграція спеціалізованого програмного забезпечення (серверного пакета маршрутизації FRR, утиліти Yersinia CLI та скриптів Python із бібліотекою Scapy) дозволила симулювати широкий спектр мережних

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		39

атак та аномалій для штучного наповнення системи логами безпеки різних рівнів важливості. Таким чином, результати проєктування другого розділу повністю сформували практичну та захищену базу для безпосереднього дослідження журналів подій та аналізу їхніх структурних особливостей у наступній частині роботи.

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				40
Змн.	Арк.	№ докум.	Підпис	Дата		

РОЗДІЛ 3 ПРАКТИЧНЕ ДОСЛІДЖЕННЯ СИСТЕМИ ЛОГУВАННЯ ТА АНАЛІЗ ПОДІЙ ІНФОРМАЦІЙНОЇ БЕЗПЕКИ

Практична частина дослідження спрямована на комплексну перевірку розгорнутої підсистеми централізованого логування в умовах, максимально наближених до реальної корпоративної інфраструктури. Для проведення стрес-тестування та верифікації аналітичних можливостей стеку ELK було обрано три різнорівневі вектори деструктивного впливу: атака на динамічний протокол маршрутизації OSPF, атака на критично важливу мережну службу DHCP та безпосередній деструктивний вплив на саму підсистему збору телеметрії за допомогою підробки журналів подій. Такий набір атак дозволяє оцінити стійкість та ефективність SIEM-подібного рішення на різних рівнях моделі OSI — від канального та мережного до прикладного. Вибір саме цих атак зумовлений їхньою небезпекою для стабільності мережі та здатністю провокувати специфічні, інтенсивні повідомлень, що дозволяє перевірити здатність системи моніторингу захищати власну довіру до джерел даних та ідентифікувати спроби зловмисника приховати свої сліди.

Під час проведення експериментів тестування системи здійснюється за кількома ключовими аспектами. Першочергово перевіряється якість та оперативність виявлення аномалій і маркерів компрометації у реальному часі за допомогою розроблених grok-фільтрів Logstash та правил індексації. Важливим критерієм оцінки є реакція системи на високоінтенсивні потоки даних та її здатність зберігати високу швидкодію обробки подій без втрати інформації. Крім того, досліджується цілісність логів, надійність транспортування телеметрії, а також загальна ефективність захисту підсистеми від спроб низькорівневих маніпуляцій із заголовками мережних пакетів. Результати цього комплексного тестування дозволять підтвердити працездатність спроектованої архітектури та сформулювати практичні рекомендації щодо протидії мережним інцидентам.

Розгортання системи, симуляція зазначених мережних аномалій та збір аналітичної телеметрії проводилися виключно на реальному фізичному мережному обладнанні, змонтованому в лабораторному стійковому шасі. Використання справжніх апаратних комплексів замість повністю віртуалізованих або емульованих

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		41

середовищ дозволило забезпечити абсолютну автентичність згенерованих системних журналів та фіксувати реакції операційних систем на деструктивні впливи з урахуванням специфіки обробки сигналів на фізичному й каналному рівнях. Для цього було налаштовано апаратний стенд, архітектура якого повністю відповідає затвердженій гетерогенній топології мережі. До складу інфраструктури увійшли два магістральні маршрутизатори різних виробників (Cisco 2821 та MikroTik RB5009UG+S+), два керовані комутатори серії Cisco Catalyst, мобільна дослідницька станція для генерації тестових впливів та виділений сервер під управлінням ОС Linux Ubuntu з аналітичним стеком ELK. Усі елементи з'єднані фізичними патч-кордами та налаштовані на миттєве перенаправлення своїх системних подій на IP-адресу сервера моніторингу 192.168.41.5, що гарантує тотожність умов дослідження реальній корпоративній мережі.

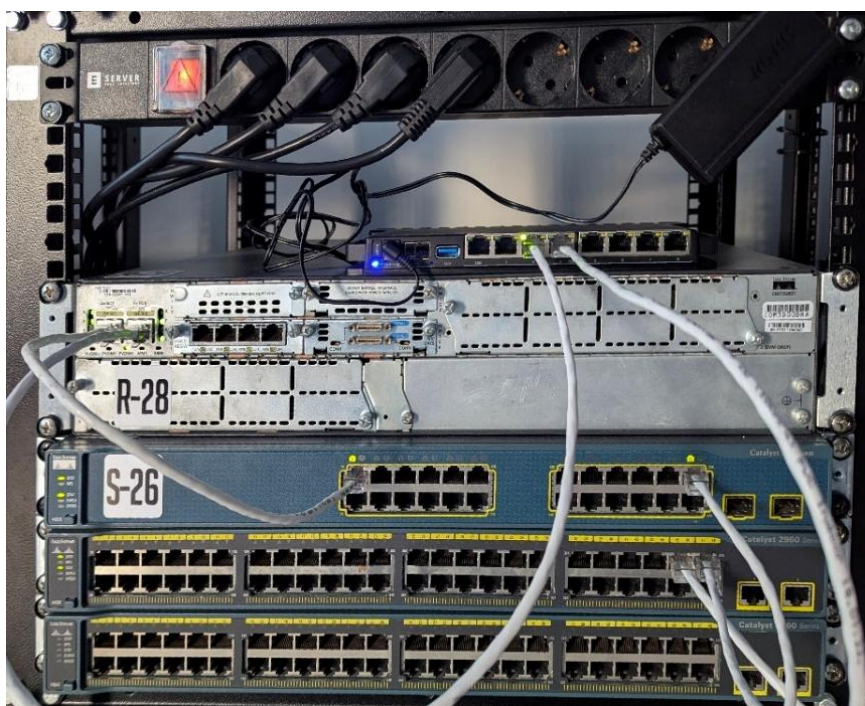


Рисунок 3.1 – Фізичний стенд дослідження

3.1 Атака на OSPF з використанням LSA пакетів

OSPF є одним із поширених протоколів внутрішньої маршрутизації (IGP), який базується на алгоритмі стану каналів. Його основна функція полягає в побудові та підтримці карти мережі, що дозволяє кожному маршрутизатору самостійно розраховувати найкоротші маршрути до всіх доступних мереж [9, с. 6]. Для цього

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		42

кожен маршрутизатор має власну базу даних стану зв'язків (LSDB), до якої надходить інформація від інших маршрутизаторів у вигляді спеціальних пакетів — LSA [9, с. 6, 13]. У цих пакетах міститься опис безпосередньо підключених інтерфейсів, їхні метрики, префікси мереж та інші характеристики, що дозволяє маршрутизатору вираховувати найкоротший шлях пакету до кінцевої точки [9, с. 21].

Після того як маршрутизатор отримує та синхронізує LSDB із сусідніми пристроями, він запускає алгоритм Dijkstra (SPF), який будує дерево найкоротших шляхів від цього маршрутизатора до всіх інших відомих мереж [9, с. 6, 21]. Ці розрахунки є локальними — кожен маршрутизатор виконує їх самостійно, ґрунтуючись лише на отриманій інформації [9, с. 6]. Результатом роботи SPF-алгоритму є формування таблиці маршрутизації [9, с. 21, 161].

Перед тим як маршрутизатори зможуть обмінюватися LSA, вони встановлюють логічну суміжність (adjacency) [9, с. 52]. Для цього вони надсилають одне одному Hello-повідомлення — спеціальні пакети, що містять параметри зони (Area ID), інтервал обміну, таймери, ідентифікатор маршрутизатора (Router ID), методи аутентифікації та інші важливі параметри [9, с. 52]. Якщо два маршрутизатори мають відповідні параметри, вони вважаються суміжними, після чого починається процес обміну базами LSDB [9, с. 53].

LSA-пакети мають декілька типів, кожен з яких виконує свою функцію у підтримці топології [9, с. 42, 111]. Кожне LSA має унікальний ідентифікатор та порядковий номер (sequence number), який постійно збільшується при оновленні інформації [9, с. 142]. Коли маршрутизатор отримує LSA з вищим порядковим номером, він вважає цей запис актуальнішим, замінює стару інформацію у своїй LSDB та запускає перерахунок маршрутизаторів за SPF-алгоритмом [9, с. 142–143]. Цей процес є безперервним і забезпечує високу адаптивність мережі до змін: у разі падіння каналу, зміни метрики або підключення нових пристроїв топологія автоматично оновлюється і трафік переорієнтовується через нові шляхи [9, с. 138].

Отже знаючи ці особливості, можна використати гнучкість і довіру проти цього протоколу, використовуючи стандартні методи які створюють потенційні

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		43

вектори атаки, що й було використано у дослідженій атаці за допомогою інструмента FRR.

Атака на протокол OSPF за допомогою програмного комплексу маршрутизації FRR, розгорнутого на робочій станції під управлінням ОС Linux, базувався на вразливості цього протоколу до несанкціонованих або підроблених повідомлень LSA, які OSPF використовує для обміну інформацією про топологію між учасниками мережі. Основна ідея атаки полягає в тому, що контрольований дослідником Linux-ноутбук за допомогою FRR ініціює роботу фейкового маршрутизатора в загальному домені маршрутизації. Цей квазі-пристрій впроваджує в мережу сфальсифіковані LSA-анонси з навмисно завищеним порядковим номером та обраним Router ID. Це змушує сусідні легітимні маршрутизатори прийняти ці пакети як більш актуальні, внаслідок чого в їхніх базах даних LSDB відбувається заміна відомостей про топологію, що спотворює реальну карту мережі.

Під час реалізації цього вектору впливу, налаштований сервіс FRR формує спеціально згенеровані пакети типу LSA Type 1 (Router LSA) або Type 2 (Network LSA). Створювані фейковим маршрутизатором повідомлення точно копіюють ключові характеристики легітимного сегмента мережі, зокрема ідентифікатор зони, маску підмережі, пріоритет та інші системні опції. Проте атакуючий вузол через конфігурацію OSPF-демона в FRR вставляє в пакет штучно збільшене значення sequence number, що є критичним критерієм у системі синхронізації LSDB. Завдяки цьому апаратні маршрутизатори мережі розцінюють підроблений анонс як новіший та з більшим пріоритетом, автоматично оновлюють власні бази даних і заміщують коректні маршрутні записи.

Процес деструктивного впливу через FRR реалізується переважно шляхом активного встановлення фальшивої суміжності (adjacency) із легітимними пристроями за допомогою Hello-пакетів та подальшого обміну технологічними даними. Програмний маршрутизатор на базі Linux повністю імітує повноцінну участь у процесі маршрутизації з певним Router ID, створюючи для легітимних пристроїв ілюзію існування нових альтернативних шляхів. Це дозволяє зловмиснику анонсувати фіктивні або підконтрольні йому технологічні маршрути (здійснюючи route

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		44

poisoning або перенаправлення трафіку), викликати конфлікти ідентифікаторів LSA у мережі та провокувати постійне обчислення алгоритму SPF, що призводить до критичної нестабільності роботи всього OSPF-домену.

Наслідки реалізації такої атаки через програмний комплекс FRR мають критичний характер для безпеки та стабільності всієї інфраструктури. По-перше, якщо фейковий маршрутизатор на базі Linux успішно анонсує дезінформаційні маршрути, легітимний трафік мережі перенаправляється через підконтрольний атакуючому вузол. Це створює ідеальні умови для перехоплення, аналізу або несанкціонованої модифікації конфіденційних даних за допомогою атаки типу «людина посередині» (Man-in-the-Middle, MitM). По-друге, можлива повна втрата зв'язку між окремими сегментами мережі (спрямування трафіку в «чорну діру») через відсутність або викривлення достовірних записів у таблицях маршрутизації апаратних пристроїв. Окрім цього, виникає ситуація, коли реальні маршрутизатори починають безперервно перераховувати топологію мережі за алгоритмом SPF через постійне надходження суперечливих OSPF LSA-повідомлень від FRR. Це викликає критичне навантаження на центральні процесори Cisco та MikroTik, призводячи до деградації мережної продуктивності та відмови в обслуговуванні (DoS).

В умовах створеної апаратної інфраструктури було здійснено атаку на домен маршрутизації OSPF за допомогою програмного комплексу FRR, розгорнутого на підконтрольній робочій станції. У ході атаки через FRR було реалізовано запуск фейкового маршрутизатора та ін'єкцію несанкціонованих повідомлень LSA з модифікованими параметрами, що призвело до дестабілізації суміжності пристроїв та штучного викривлення мережної топології. Головною метою цього етапу досліджень було визначення специфіки подій, які генеруються апаратним обладнанням у момент атаки, фіксація аномалій у поведінці пристроїв та оцінка ефективності їхнього виявлення й візуалізації за допомогою розгорнутого веб-інтерфейсу Kibana.

Після завершення конфігурування OSPF-демона у FRR, процес якого є прямолінійним і полягає у визначенні параметрів зони та автентифікації для встановлення зв'язку з легітимними вузлами, було активовано трансляцію підроблених

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		45

анонсів. Основну увагу в експерименті було зосереджено безпосередньо на аналізі наслідків деструктивного впливу та верифікації тих телеметричних маркерів, які дозволяють системному адміністратору оперативно ідентифікувати атаку на рівень керування (control-plane) в реальному часі.

Практично відразу після активації фейкового маршрутизатора FRR і початку масової розсилки модифікованих LSA-повідомлень, на цільовому апаратному маршрутизаторі Cisco R1 із системною IP-адресою 192.168.41.1 розпочалися процеси безперервного перерахунку маршрутів. Це миттєво спровокувало реєстрацію аномальної активності в SIEM-системі. Як чітко демонструє аналітичний дашборд Kibana, за короткий проміжок часу від вузла 192.168.41.1 надійшло рівно 104 записи логів.

Аналіз розподілу подій на візуалізаціях виявив такі ключові телеметричні індикатори:

- Мнемоніка UPDOWN склала найбільшу частку — 38.46% (40 записів), що свідчить про нестабільність інтерфейсів або протокольних процесів під час атаки.
- Мнемоніка ADJCHG зайняла 23.08% (24 записи), прямо вказуючи на постійне та циклічне скидання й перебудову станів суміжності (adjacency) між легітимним маршрутизатором та фейковим сусідом.
- Серед системних тегів домінують OSPF (24 записи), LINEPROTO (20 записів) та LINK (20 записів).

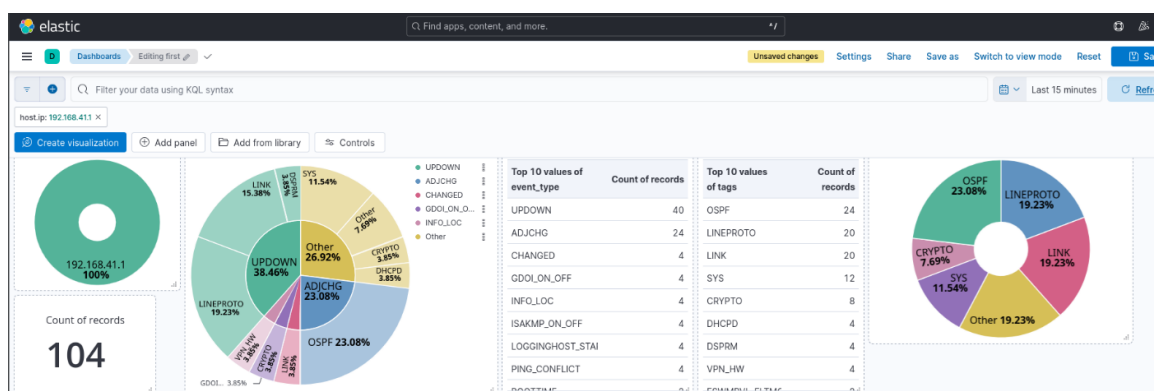


Рисунок 3.2 – Наслідки атаки на OSPF з використанням LSA пакетів від обладнання з системою Cisco IOS

Поява такої високої щільності системних сповіщень із технологічними темами OSPF та ADJCHG сигналізує про серйозну потенційну небезпеку для працездатності та безпеки всієї мережної інфраструктури. Подібна аномальна концентрація подій за короткий проміжок часу зазвичай свідчить або про грубі помилки при початковому конфігуруванні параметрів стикування, або про раптові апаратні чи програмні збої під час синхронізації баз даних (LSDB) між маршрутизаторами, або ж про реалізацію цілеспрямованої кібератаки, спрямованої на викривлення маршрутної карти. У будь-якому випадку, така поведінка домену маршрутизації провокує безперервне обчислення топології, що загрожує відмовою в обслуговуванні (DoS) через перевантаження процесорів апаратних пристроїв і вимагає негайного втручання адміністратора безпеки.

Для того щоб детально розібратися у характері зафіксованої аномалії та визначити її точну першопричину, необхідно перейти від загальних статистичних метрик дашборду до аналізу безпосереднього вмісту сирих лог-файлів. З цією метою в SIEM-системі використовується функціонал глибинного аналізу подій у реальному часі. На наступному знімку екрана (рис. 3.3) продемонстровано інтерфейс вкладки Discover у Kibana, який дозволяє вивчити покрокову хронологію та детальну текстову структуру конкретних syslog-повідомлень, що генерувалися пристроєм під час деструктивного впливу.

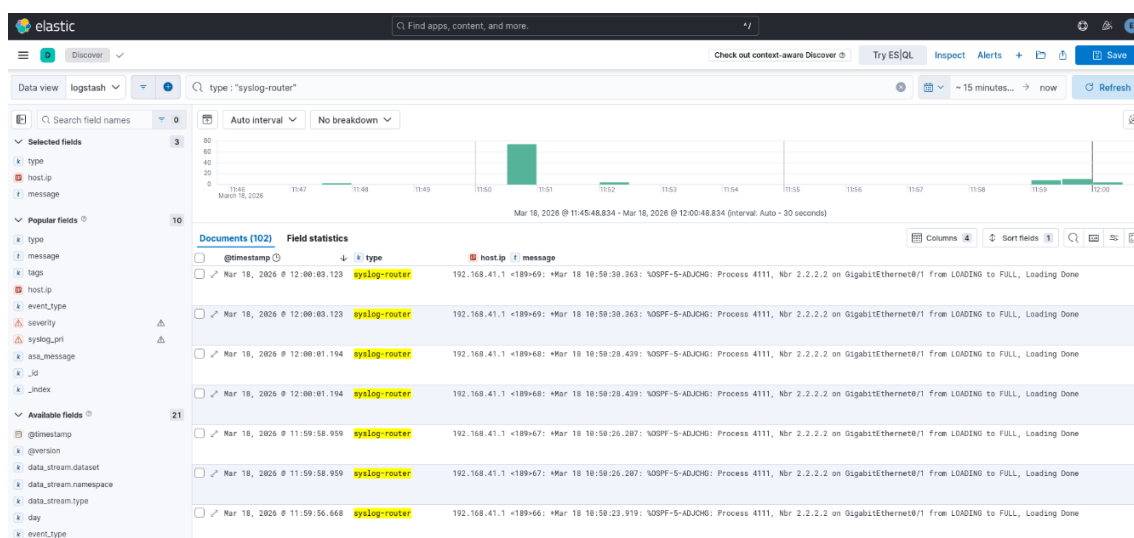


Рисунок 3.3 – Вкладки Discover у Kibana під час атаки на OSPF з використанням LSA

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				47
Змн.	Арк.	№ докум.	Підпис	Дата		

Як чітко видно на інтерфейсі вкладки Discover (рис. 3.4), на цільовому маршрутизаторі Cisco R1 з IP-адресою 192.168.41.1 фіксуються численні записи з типом даних syslog-router та тегом OSPF. Системні сповіщення типу: *Mar 18 10:50:30.363: %OSPF-5-ADJCHG: Process 4111, Nbr 2.2.2.2 on GigabitEthernet0/1 from LOADING to FULL, Loading Done свідчать про постійні та циклічні зміни стану суміжностей (adjacency). Це означає, що під впливом зовнішніх ін'єкцій від фейкового маршрутизатора FRR пристрій Cisco змушений багаторазово переходити між різними фазами OSPF-сусідства (зокрема, з LOADING у FULL). Такі «флапінги» суміжностей підтверджують, що легітимний вузол постійно перезапущає процес синхронізації через надходження нових суперечливих LSA, що призводить до критичного виснаження його апаратних ресурсів (CPU та RAM).

Прямі наслідки цього деструктивного впливу на таблицю маршрутизації відображені на рисунку 3.4, де представлено вивід команди show ip route маршрутизатора Cisco R1 після початку активної фази атаки.

```

R1#show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       + - replicated route, % - next hop override

Gateway of last resort is not set

    2.0.0.0/32 is subnetted, 1 subnets
O      2.2.2.2 [110/11] via 192.168.41.10, 00:00:10, GigabitEthernet0/1
    99.0.0.0/32 is subnetted, 100 subnets
O E2   99.41.11.0 [110/30] via 192.168.41.10, 00:00:10, GigabitEthernet0/1
O E2   99.41.11.1 [110/30] via 192.168.41.10, 00:00:10, GigabitEthernet0/1
O E2   99.41.11.2 [110/30] via 192.168.41.10, 00:00:10, GigabitEthernet0/1
O E2   99.41.11.3 [110/30] via 192.168.41.10, 00:00:10, GigabitEthernet0/1
O E2   99.41.11.4 [110/30] via 192.168.41.10, 00:00:10, GigabitEthernet0/1
O E2   99.41.11.5 [110/30] via 192.168.41.10, 00:00:10, GigabitEthernet0/1
O E2   99.41.11.6 [110/30] via 192.168.41.10, 00:00:10, GigabitEthernet0/1
O E2   99.41.11.7 [110/30] via 192.168.41.10, 00:00:10, GigabitEthernet0/1
O E2   99.41.11.8 [110/30] via 192.168.41.10, 00:00:10, GigabitEthernet0/1
O E2   99.41.11.9 [110/30] via 192.168.41.10, 00:00:10, GigabitEthernet0/1
O E2   99.41.11.10 [110/30] via 192.168.41.10, 00:00:10, GigabitEthernet0/1
O E2   99.41.11.11 [110/30] via 192.168.41.10, 00:00:10, GigabitEthernet0/1
O E2   99.41.11.12 [110/30] via 192.168.41.10, 00:00:10, GigabitEthernet0/1
O E2   99.41.11.13 [110/30] via 192.168.41.10, 00:00:10, GigabitEthernet0/1

```

Рисунок 3.4 – Таблиця маршрутизації після надсилання хибних LSA пакетів.

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		48

Цей вивід наочно демонструє успішність реалізованого вектору зламу: у таблиці з'явилася величезна кількість штучно впроваджених маршрутів типу O E2 (OSPF External Type 2) для мереж у діапазоні 99.41.11.0/32. Усі ці фіктивні маршрути прописані через LAN-інтерфейс GigabitEthernet0/1 із наступним хопом (next-hop) на IP-адресу 192.168.41.10, яка належить підконтрольній робочій станції під керуванням Linux із запущеним комплексом FRR.

Враховуючи, що час оновлення цих записів становить лічені секунди (наприклад, 00:00:10), це доводить факт успішної ін'єкції зовнішніх LSA в момент активації фейкового вузла. У штатному режимі функціонування мережа не може генерувати таку велику кількість надлишкових та дрібних маршрутів за такий короткий проміжок часу, що робить цей аномальний показник головним критерієм для миттєвого виявлення control-plane атак адміністратором безпеки.

З метою порівняльного аналізу було проведено аналогічну атаку на маршрутизаторі іншого виробника для визначення особливостей її впливу на пристрої під управлінням RouterOS 7. Для забезпечення об'єктивності та коректності експерименту дослідницьку станцію було перенесено в іншу підмережу, де виконано ідентичну послідовність дій.

Практично відразу після активації нелегітимного маршрутизатора FRR та початку масової розсилки модифікованих LSA-повідомлень, на цільовому пристрої MikroTik під управлінням RouterOS 7 розпочалися процеси обробки аномального обсягу маршрутної інформації та штучного масового створення нових записів у таблиці маршрутизації, що миттєво спровокувало реєстрацію критичної активності підсистемою централізованого логування. Як чітко демонструє аналітичний дашборд Kibana (рис. 3.5), за короткий проміжок часу від досліджуваного пристрою надійшло 3064 записи логів, з яких 2216 стосуються безпосередньо протоколу OSPF. Наведене на рисунку графічне представлення результатів успішної реалізації атаки відображає масив системних повідомлень різних типів, дозволяючи детально проаналізувати їхній кількісний розподіл, а також класифікувати категорії та рівні важливості логів, що надходять від маршрутизатора, для максимально швидкої локалізації та усунення проблеми адміністратором мережі.

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		49

Розподіл повідомлень за типом івенту має домінування OSPF-трафіку, найбільшу частку серед усіх зареєстрованих подій займають повідомлення протоколу OSPF — 2 216 записів, що становить 72.32% від загального обсягу. Також можна побачити що майже кожен лог який мав тип OSPF має тег route, що дає адміністратору інформацію що дані логи пов’язані саме з додавання нових маршрутів. Решта 27.68% (848 записів) розподілені між загальними інформаційними сповіщеннями, з яких найбільша частина це SSH оскільки в момент атаки було активне з’єднання по SSH.

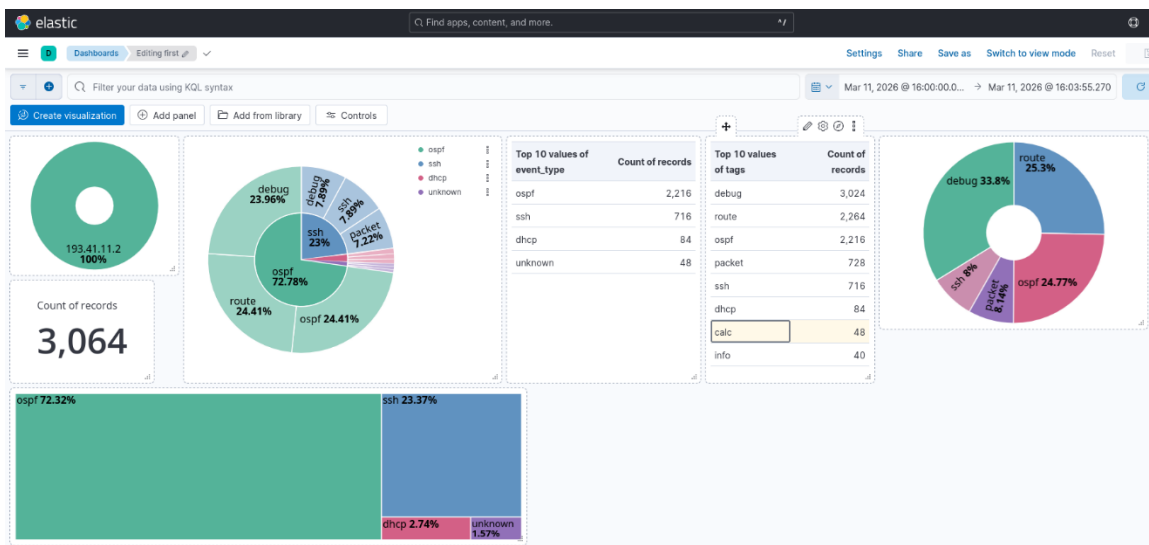


Рисунок 3.5 – Наслідки атаки на OSPF з використанням LSA пакетів від обладнання з системою RouterOS

Аналіз первинних часових міток на початковій стадії фіксації аномалії свідчить про успішне прийняття цільовим пристроєм модифікованих записів про маршрути, що надходили від зовнішнього джерела з ідентифікатором 2.2.2.2. Отримання копій LSA з вищими номерами послідовності ініціювало в OSPF-процесі Mikrotik стандартну процедуру оновлення бази даних лінійних зв’язків (LSDB). Відповідно до базових алгоритмів функціонування протоколу Link-State, маршрутизатор негайно переходив до каскадного розповсюдження цих даних через усі активні інтерфейси OSPF-сусідам, що відображено у логах як масовий процес лавиноподібної розсилки (flooding). Тобто ситуація аналогічна як була і з маршрутизатором на Cisco IOS.

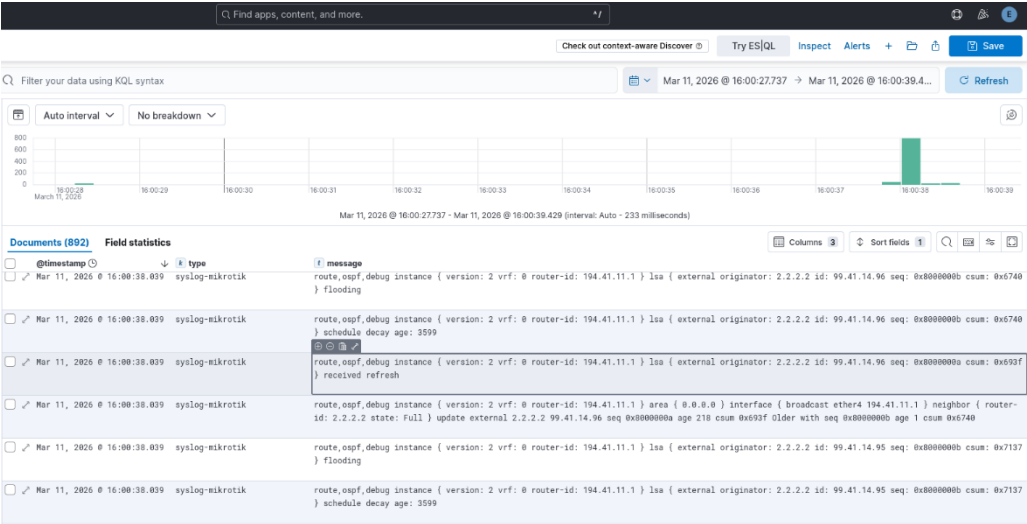


Рисунок 3.6 – Візуалізація та аналіз мережних подій пов’язаних з OSPF на мережних пристроях з системою RouterOS

Особливий науковий інтерес викликає фіксація системних повідомлень із параметром штучного обмеження життєвого циклу пакета(рис 3.7) — schedule decay age: 3599. Даний маркер безпосередньо вказує на виникнення критичного конфлікту легітимності маршрутної інформації всередині мережного сегмента. Подібна аномалія виникає тоді, коли легітимний маршрутизатор або внутрішні захисні таймери OSPF намагаються виконати процедуру примусового очищення бази даних шляхом виставлення максимального віку пакета MaxAge. Проте, зловмисний вузол на базі FRR циклічно генерував нові пакети з іншими параметрами, що створило стан race condition(гонки замовчувань) і призвело до безперервного шторму оновлень.

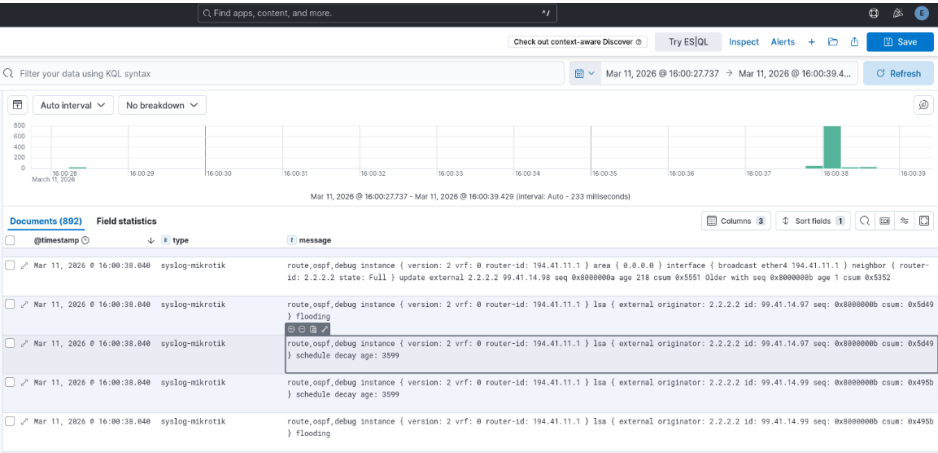


Рисунок 3.7 – Візуалізація та аналіз мережних подій пов’язаних з OSPF на мережних пристроях з системою RouterOS

Наслідки описаного шторму маршрутизації чітко простежуються на завершальному етапі логування, де зафіксовано різке зростання щільності записів про початок обчислювальних операцій. Через постійну зміну топологічної карти мережі операційна система маршрутизатора була змушена лавиноподібно запускати процедуру перерахунку дерева найкоротших шляхів за алгоритмом Дейкстри. Численні системні відмітки щодо ініціалізації розрахунків, побудови дерева найкоротших шляхів (calc spt), а також подальшої обробки сумарних та зовнішніх маршрутів (process summaries, process externals) свідчать про критичне навантаження на обчислювальні ресурси пристрою.

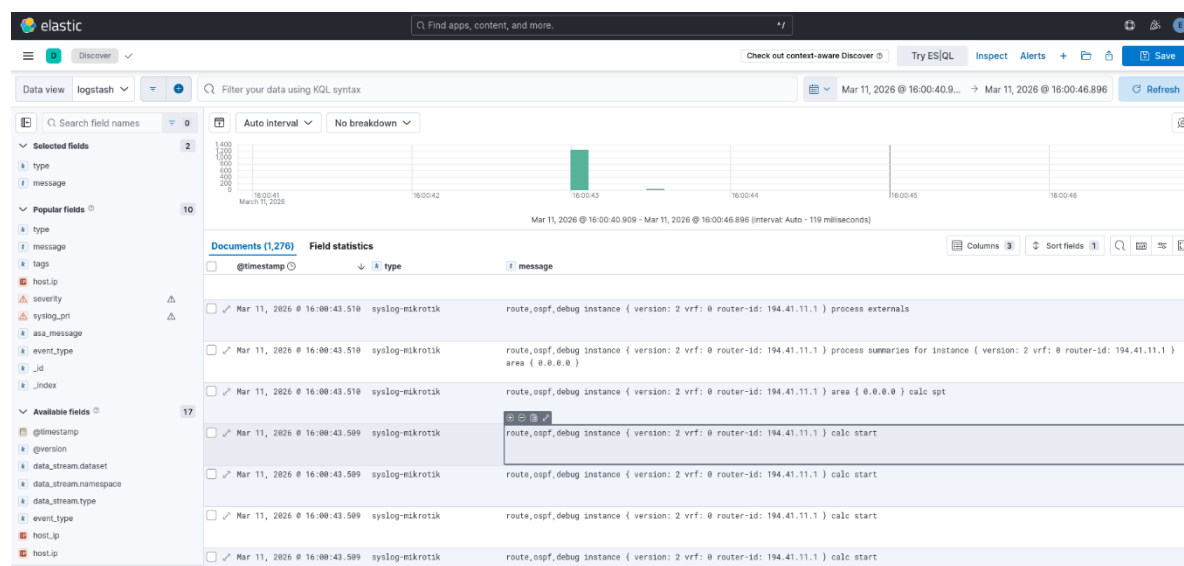


Рисунок 3.8 – Візуалізація та аналіз мережних подій пов’язаних з OSPF на мережних пристроях з системою RouterOS

Візуалізація стану таблиці маршрутизації відображає кінцеву фазу деструктивного впливу на Mikrotik, отруєння таблиці маршрутизації. Порівняльний аналіз двох послідовних викликів команди чітко демонструє динаміку наповнення бази даних пристрою фіктивними записами. На початковому етапі в системі фігурували лише легітимні маршрути, зокрема безпосередньо підключені мережі (прапорці DAC — Dynamic, Active, Connected) та один запис, отримані по OSPF (прапорець DAo — Dynamic, Active, OSPF) з метрикою 110(рис 3.9, переш застосування команди «/ip route print»).

```

[admin@MikroTik] > /ip route print
Flags: D - DYNAMIC; A - ACTIVE; c, o, y - COPY
Columns: DST-ADDRESS, GATEWAY, DISTANCE
      DST-ADDRESS      GATEWAY      DISTANCE
DAo 192.168.41.0/24    193.41.11.1%ether3    110
DAc 193.41.11.0/30     ether3              0
DAc 194.41.11.0/24     ether4              0
[admin@MikroTik] > /ip route print
Flags: D - DYNAMIC; A - ACTIVE; c, o, y - COPY
Columns: DST-ADDRESS, GATEWAY, DISTANCE
      DST-ADDRESS      GATEWAY      DISTANCE
DAo 2.2.2.2/32         194.41.11.10%ether4    110
DAo 99.41.14.0/32      194.41.11.10%ether4    110
DAo 99.41.14.1/32      194.41.11.10%ether4    110
DAo 99.41.14.2/32      194.41.11.10%ether4    110
DAo 99.41.14.3/32      194.41.11.10%ether4    110
DAo 99.41.14.4/32      194.41.11.10%ether4    110
DAo 99.41.14.5/32      194.41.11.10%ether4    110
DAo 99.41.14.6/32      194.41.11.10%ether4    110
DAo 99.41.14.7/32      194.41.11.10%ether4    110
DAo 99.41.14.8/32      194.41.11.10%ether4    110

```

Рисунок 3.9 – Таблиця маршрутизація до та після атаки за допомогою LSA-пакетів

Наступний зріз стану системи фіксує упровадження великого масиву аномальних маршрутів типу хост (з маскою /32)(рис 3.9, друге застосування команди «/ip route print»). Усі нові записи мають префікс 99.41.14.X/32 та вказують на шлюз 194.41.11.10%ether4, що відповідає інтерфейсу взаємодії з атакуючим програмним маршрутизатором FRR. Згенерована зловмисним вузлом послідовність хостових маршрутів була успішно оброблена OSPF-процесом і встановлена в операційну систему як активна (прапорці DАо), що свідчить про повну компрометацію поточного табличного простору маршрутизації.

Таким чином, експериментально підтверджено, що використання інструментарію FRR для упровадження хибних LSA-пакетів є ефективним методом деструктивного впливу, здатним дестабілізувати функціонування будь-яких пристроїв третього рівня моделі OSI, незалежно від вендора чи конкретного виробника мережного обладнання. Оскільки описана вразливість криється в архітектурних

особливостях самого протоколу OSPF, за відсутності належних механізмів захисту вплив призводить до масштабного збою в роботі мережної інфраструктури. У реальних умовах функціонування телекомунікаційних систем цей деструктивний процес викликає тривалу нестабільність топології мережі та критичне вичерпання ресурсів центрального процесора маршрутизаторів, що тягне за собою втрату Hello-пакетів, розірвання суміжності між пристроями та масштабну відмову в обслуговуванні (DoS) для всього легітимного користувацького трафіку.

У межах досліджуваного сегмента також реалізовано перехід до криптографічної автентифікації на основі алгоритму хешування MD5. Математична стійкість даного підходу базується на обчисленні унікального автентифікаційного заголовка для кожного OSPF-пакета, який містить ідентифікатор ключа, номер послідовності для захисту від атак повторення та безпосередньо хеш-суму Message Digest. Це дозволяє в ході подальшого моніторингу чітко розмежувати легітимний обмін даними та спроби стороннього втручання, та показати яким чином можна не допускати дану атаку на LSA-пакети[19].

Експериментальна перевірка стійкості інфраструктури до повторного впливу з боку інструментарію FRR відобразила якісні зміни у системних логах. При спробі зловмисного вузла здійснити ін'єкцію підроблених Type-5 LSA без знання секретного ключа, атаковані маршрутизатори ідентифікували такі пакети як невалідні на етапі перевірки заголовка пакетів (OSPF Packet Header Validation). Оскільки зловмисник не здатний згенерувати коректне значення Message Digest, операційні системи миттєво відкидали модифіковані пакети. Цей етап захисної реакції та фіксації критичних повідомлень ілюструють системні лонги Cisco IOS, наведені на рисунках (Рис.3.10) де зафіксовано аварійне скидання суміжності з неавторизованим сусідом 2.2.2.2 та перехід інтерфейсу у стан DOWN через порушення політики безпеки.

```
<189>54: *Mar 18 10:50:28.439: %OSPF-5-ADJCHG: Process 4113,
Nbr 2.2.2.2 on FastEthernet1/0 from FULL to DOWN, Neighbor
Down: Interface down or detached
```

Рис 3.10. Невдала спроба аутентифікації Cisco IOS

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		54

Паралельно з цим, якісні зміни захисного характеру було зафіксовано і в системних журналах маршрутизатора під керуванням операційної системи MikroTik RouterOS. При спробі несанкціонованого інструментарію FRR надіслати модифікований пакет, підсистема логування Mikrotik (індикатори route, ospf, warning) миттєво згенерувала повідомлення про критичну невідповідність параметрів безпеки (Рис. 3.11). Системна подія з інформаційним вмістом authentication failed from 195.41.13.1 type mismatch чітко свідчить про те, що інтерфейс ether1 маршрутизатора з ідентифікатором 1.1.1.1 зафіксував спробу встановлення суміжності або передачі LSA з дефектним типом автентифікації. Даний запис підтверджує, що ядро операційної системи Mikrotik заблокувало обробку вхідного пакета ще на рівні аналізу службового заголовка OSPF, не допустивши його впливу на внутрішню базу даних лінійних зв'язків (LSDB).

```
route,ospf,warning default-inst { version: 2 router-id:
1.1.1.1 } backbone { 0.0.0.0 } interface { broadcast
195.41.13.2%ether1 } authentication failed from 195.41.13.1
type mismatch
```

Рисунок 3.11 – Невдала спроба автентифікації RouterOS

3.2. Атака DHCP Starvation з використанням discover запитів

Поряд із дестабілізацією маршрутної інфраструктури, загрозу для цілісності та доступності локальної мережі становлять вектори атак, спрямовані на протокол DHCP. Будучи сервісом для автоматичного призначення мережних реквізитів, DHCP за замовчуванням не має вбудованих засобів автентифікації клієнтів чи серверів, що створює передумови для зловмисного втручання. Для комплексного оцінювання ризиків у межах цієї роботи було обрано інструментарій Yersinia CLI, який дозволяє моделювати сценарії порушення безпеки DHCP, такі як вичерпання доступних адрес (DHCP Starvation) та подальшого впровадження нелегітимного сервера (DHCP Spoofing). Цей етап присвячено детальному аналізу механіки цих атак, оцінці їхніх наслідків та фіксації їхніх ознак за допомогою засобів моніторингу.

Протокол DHCP функціонує на прикладному рівні моделі OSI та використовує як транспортний протокол UDP із виділеними портами 67 для сервера, та 68 для

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		55

клієнта. Оскільки на початковому етапі підключення новий пристрій у мережі ще не має чинної IP-адреси та не знає архітектури мережі, пошук доступного DHCP-сервера здійснюється виключно шляхом надсилання широкомовних запитів на канальному та мережному рівнях [8, с. 7, 13].

У штатному режимі процес динамічного отримання мережних реквізитів базується на взаємодії пакетів стандарту DORA. Першим кроком є надсилання клієнтом широкомовного пакету DHCP Discover [8, с. 7-8, 13]. Реагуючи на цей запит, DHCP-сервер виконує такі послідовні дії:

1. Аналіз ідентифікатора: Сервер зчитує вміст поля CHADDR (Client Hardware Address) всередині пакету, яке містить унікальну фізичну MAC-адресу пристрою-відправника [8, с. 9, 15].

2. Модифікація таблиці розподілу: Сервер звертається до внутрішньої таблиці динамічних зв'язків (DHCP binding table) і фіксує в ній запит від цієї MAC-адреси, автоматично виділяючи під неї вільну IP-адресу із наявних доступних [8, с. 11, 26].

3. Формування пропозиції: Після фіксації запису в системній таблиці, сервер генерує та відправляє назад у мережу пакет DHCP Offer. Цей пакет містить запропоновану IP-адресу, маску підмережі, адресу шлюзу за замовчуванням, адреси DNS-серверів, час оренди адреси [8, с. 9, 15].

Саме цей архітектурний алгоритм попереднього резервування адрес стає об'єктом експлуатації під час атаки типу DHCP Starvation, моделювання якої виконується за допомогою спеціалізованого інструментарію Yersinia CLI.

Суть атаки полягає у інтенсивній генерації та масовому розсиланні аномально великої кількості фіктивних пакетів DHCP Discover. При цьому Yersinia у кожному згенерованому пакеті динамічно підробляє (проводить spoofing) MAC-адресу джерела в полі CHADDR [8, с. 9, 15].

Отримуючи такий потік широкомовних запитів, легітимний DHCP-сервер сприймає кожен пакет як звернення від нового фізичного пристрою. Він послідовно зчитує згенеровані зловмисником підроблені MAC-адреси, вносить їх у свою

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		56

таблицю розподілу та переводить відповідні IP-адреси з вільних в статус зарезервованих [8, с. 26]. Далі сервер штатно відправляє пакети DHCP Offer у мережу [8, с. 15].

Проте, оскільки реальних клієнтів із такими фізичними адресами в сегменті не існує, зворотні пакети DHCP Request на сервер ніколи не повертаються. Незважаючи на відсутність фінального підтвердження, зарезервовані сервером IP-адреси залишаються заблокованими у таблиці розподілу на час дії внутрішнього таймауту очікування. Через високу швидкість атаки це призводить до повного вичерпання всього адресного простору за короткий проміжок часу, оскільки генерувати дані пакети дуже просто.

У результаті атаки таблиця розподілу легітимного сервера повністю заповнюється фіктивними записами, хоча фактично жодна з адрес не була остаточно призначена реальному робочому пристрою. Як наслідок, коли новий легітимний користувач підключається до мережі та надсилає свій легітимний запит DHCP Discover, сервер не здатний запропонувати йому мережні налаштування через відсутність вільних ресурсів. Це спричиняє стан відмови в обслуговуванні на рівні базової мережної конфігурації.

Окрім ізоляції клієнтів, такий стан штучного дефіциту створює ідеальні умови для реалізації наступного вектора — атаки DHCP Spoofing (Rogue DHCP Server). Оскільки легітимний сервер повністю нейтралізований і не відповідає на запити, зломисник отримує можливість розгорнути в мережному сегменті власний нелегітимний DHCP-сервер. Цей сервер починає перехоплювати запити нових користувачів і видавати їм модифіковані параметри, підроблену адресу шлюзу або DNS, що дозволяє зломиснику реалізувати атаку типу «людина посередині», для перехоплення або модифікації трафіку.

Практична верифікація та аналіз деструктивних аномалій від атаки DHCP Starvation базувалися на аналітиці розгорнутого комплексу. Завдяки цьому вдалося детально дослідити зміни у системних журналах Cisco IOS безпосередньо в момент реєстрації безпекових інцидентів.

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		57

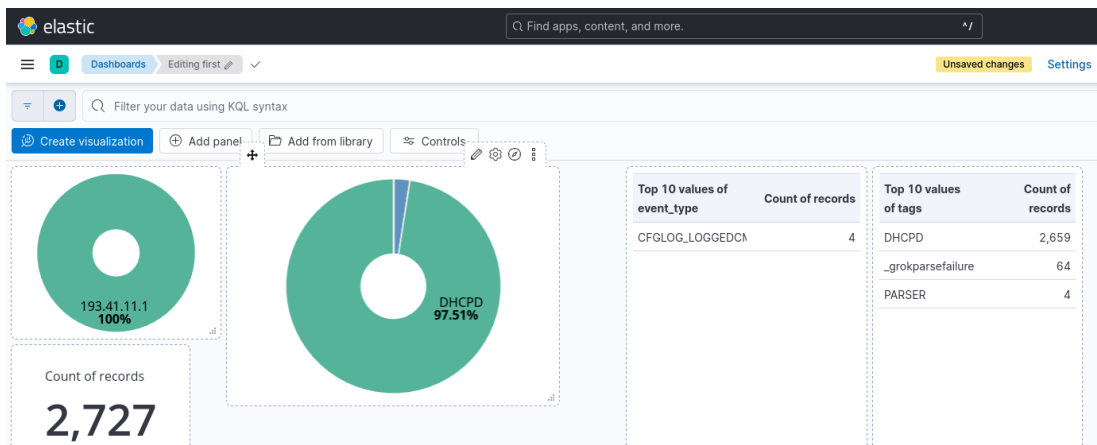


Рисунок 3.12 – Інформаційна панель системи моніторингу під час атаки на протокол DHCP у обладнанні з системою Cisco IOS

У момент активації інструменту Yersinia та запуску інтенсивного генерації широкомовними запитами DHCP Discover, аналітична платформа зафіксувала миттєву аномальну активність. На загальному дашборді моніторингу (Рисунок 3.12) чітко візуалізовано масштабне перевантаження сервісу: за короткий проміжок часу загальна кількість подій стрімко зросла до 2 727 записів. При цьому аналіз розподілу тегів показує, що 97,51% від усього обсягу транзитного syslog-трафіку (2 659 записів) припадає саме на підсистему DHCPD. Така концентрація однорідних повідомлень є першим прямим маркером проведення цілеспрямованої атаки на вичерпання ресурсів сервера.

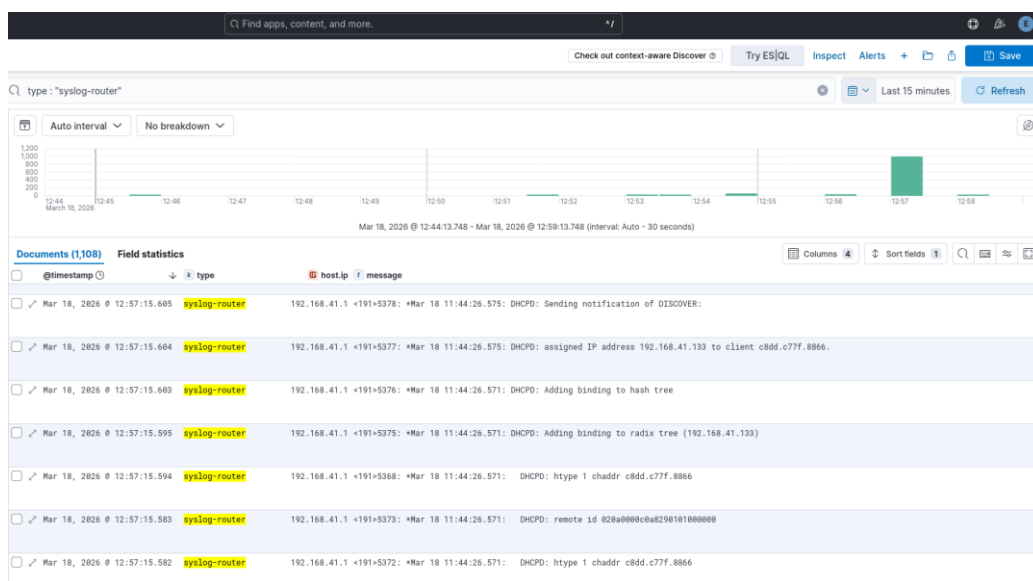


Рисунок 3.13 – Моніторинг подій DHCP-сервера, процес DHCP DISCOVER та призначення IP-адреси у обладнання з системою Cisco IOS

Аналіз часового зрізу системних повідомлень на Рисунку 3.13 дозволяє детально реконструювати покрокову послідовність дій та внутрішню логіку функціонування DHCP-сервера операційної системи Cisco IOS під час обробки зловмисного запиту. Хронологічний аналіз телеметрії (знизу догори за часовими мітками поля @timestamp) відображає таку послідовність виконання операцій ядром маршрутизатора:

- 12:57:15.582 — Ініціалізація та зчитування L2-параметрів: Системний демон переходить у стан обробки пакета й фіксує його базові фізичні характеристики за допомогою повідомлення DHCPD: htype 1 chaddr c8dd.c77f.8866. Процесор ідентифікує тип мережної архітектури (htype 1 — Ethernet) та вилучає згенеровану утилітою штучного флуду апаратну MAC-адресу клієнта (chaddr).
- 12:57:15.583 — Десеріалізація службових опцій: На наступній мілісекунді фіксується лог DHCPD: remote id 020a0000c0a8290101000000. Маршрутизатор виконує синтаксичний аналіз додаткових полів запиту (зокрема, Option 82 / Remote ID), перевіряючи легітимність параметрів ретрансляції.
- 12:57:15.594 — Повторна верифікація транзакції: Операційна система повторно звіряє дескриптори пакета DHCPD: htype 1 chaddr c8dd.c77f.8866 перед безпосереднім виділенням обчислювальних ресурсів під нову сесію оренди.
- 12:57:15.595 — Резервування у префіксному дереві: Повідомлення DHCPD: Adding binding to radix tree (192.168.41.133) фіксує початок впливу атаки на оперативну пам'ять. Маршрутизатор динамічно виділяє вільну IP-адресу 192.168.41.133 та ініціює створення нового вузла у внутрішній деревовидній структурі (radix tree) ядра ОС, що відповідає за загальну маршрутизацію адресних префіксів.
- 12:57:15.603 — Індксація у хеш-таблиці: Запис DHCPD: Adding binding to hash tree відображає процедуру оптимізації пошуку. Для забезпечення швидкого доступу за парою MAC-IP ядро Cisco IOS дублює створену прив'язку та записує її у внутрішні хеш-структури пам'яті (hash tree).
- 12:57:15.604 — Формальне призначення адреси: Повідомлення DHCPD: assigned IP address 192.168.41.133 to client c8dd.c77f.8866. сигналізує про

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		59

завершення формування локального запису оренди. Тимчасовий логічний зв’язок між підробленою MAC-адресою та доступним IP-ресурсом вважається закріпленим у базі даних сервера.

Наведений послідовність логів є доказом успішної реалізації атаки DHCP Starvation на рівні логіки ядра ОС. Отримані дані демонструють, що на один дискретний запит зломисника маршрутизатор змушений виконувати складний алгоритм обробки, який супроводжується інтенсивним внесенням змін у динамічні структури даних оперативної пам’яті (radix tree та hash tree). За умов високошвидкісного повторення цієї послідовності тисячі разів на секунду невідворотно призводить до критичного вичерпання RAM/CPU пристрою та повного блокування доступних адрес.

Додатковий аналіз часових зрізів системних повідомлень на Рисунок 3.14 та Рисунок 3.15 дозволяє виявити критично важливу особливість життєвого циклу виділених IP-адрес під час компрометації сервісу. Зафіксовані у журналах події виду DHCPD: the lease for address 192.168.41.213 has expired та DHCPD: returned 192.168.41.176 to address pool LAN свідчать про автоматичний запуск внутрішніх захисних та очищувальних механізмів операційної системи Cisco IOS. Наведені логи показують, що зарезервовані внаслідок штучного флуду IP-адреси не блокуються сервером назавжди. Після завершення встановленого періоду очікування вони повертаються до категорії доступних і знову можуть легітимно використовуватись кінцевими користувачами мережного сегмента.

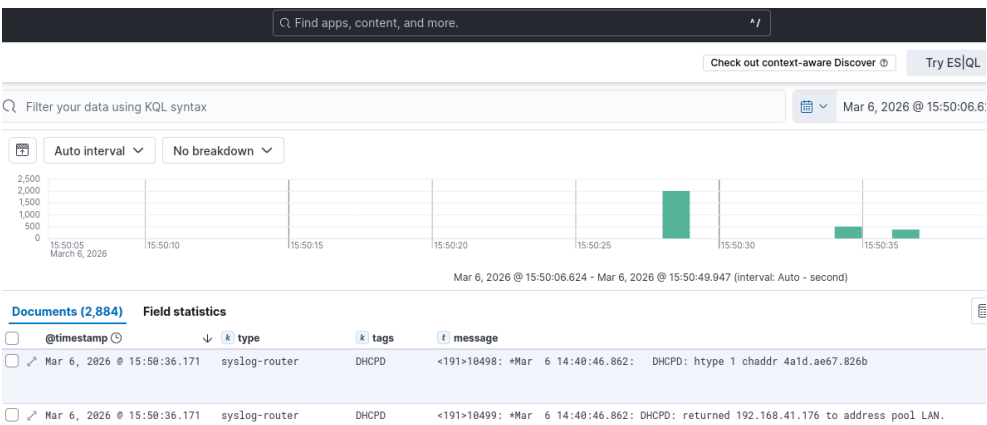


Рисунок 3.14 – Моніторинг подій DHCP-сервера, процес повернення IP-адреси до списку доступних у обладнання з системою Cisco IOS

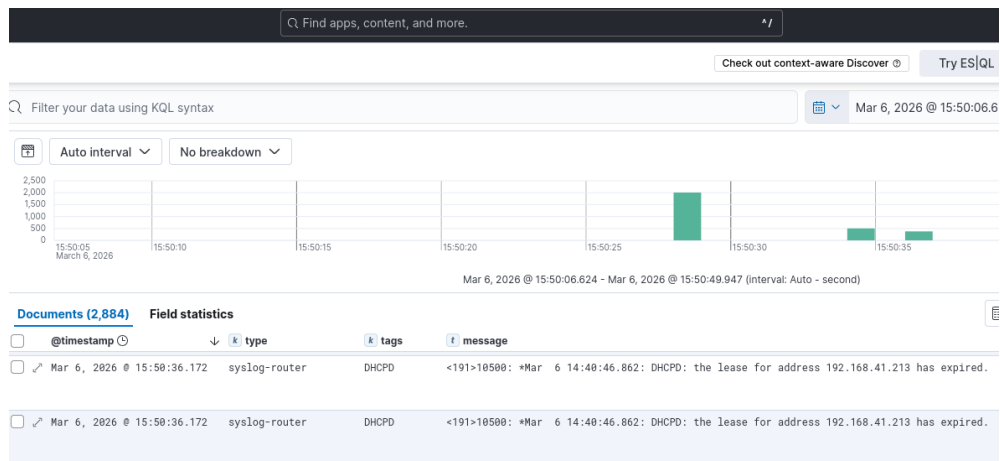


Рисунок 3.15 – Моніторинг подій DHCP-сервера, процес повернення IP-адреси до списку доступних у обладнання з системою Cisco IOS

Отримані телеметричні дані дозволяють оцінити практичні часові межі ефективності розглянутого вектору деструктивного впливу. Незважаючи на те, що метод атаки DHCP Starvation є вкрай ефективним для миттєвого та повного вичерпання адресного простору, тривалість фази повної відмови в обслуговуванні (DoS) обмежена алгоритмами обробки незавершених транзакцій. Оскільки атакуюче програмне забезпечення ігнорує транзитні відповіді сервера (DHCP Offer) і принципово не надсилає фінального підтвердження про успішне отримання мережних налаштувань (DHCP Request), операційна система очікує завершення стандартного 5-хвилинного інтервалу, після чого примусово анулює тимчасову прив'язку й оголошує IP-адресу вільною.

З точки зору безпекового аудиту та моделювання дій зловмисника, наявність цього п'ятихвилинного таймауту накладає обмеження на реалізацію подальших етапів кібератаки (таких як розгортання Rogue DHCP-сервера або перехоплення трафіку методом Man-in-the-Middle), тому для досягнення стабільного результату атакуюча сторона змушена обирати між двома тактиками. Перша — тактика швидкої дії, що передбачає миттєву реалізацію наступних фаз компрометації мережі суворо в межах виявленого п'ятихвилинного вікна, доки легітимні клієнти позбавлені можливості оновити конфігурацію. Друга — тактика перманентного шторму, яка полягає у проведенні інтенсивної атаки безперервно для підтримки DHCP-сервера у стані постійного флуду паралельно з виконанням інших етапів вторгнення.

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				61
Змн.	Арк.	№ докум.	Підпис	Дата		

Переходячи до дослідження аналогічного деструктивного впливу на інший тип мережного операційного забезпечення, а саме MikroTik RouterOS, варто проаналізувати зведену телеметрію безпекових подій. На Рисунку 3.16 представлено дашборд у Kibana, який відображає загальний план атаки та фіксує структурний розподіл повідомлень, що надходили під час виконання атаки.

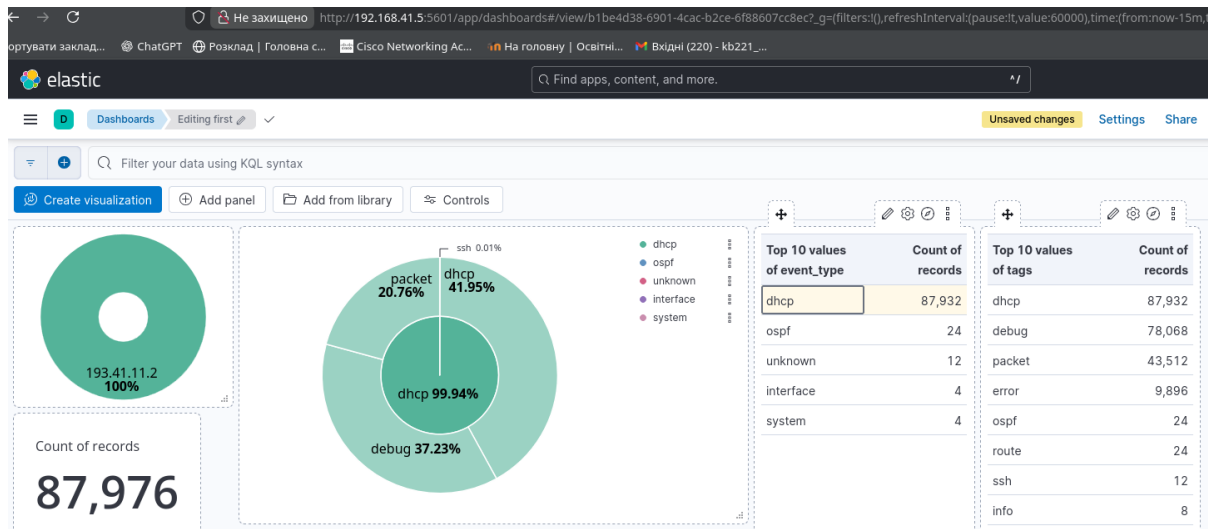


Рисунок 3.16 – Інформаційна панель системи моніторингу під час атаки на протокол DHCP у обладнанні з системою RouterOS

Індикатор загальної кількості записів фіксує критично високу щільність аномального трафіку — усього за аналізований часовий проміжок з хоста маршрутизатора під керуванням RouterOS було отримано 87 976 записів. Абсолютна більшість цих повідомлень належить безпосередньо до підсистеми DHCP, що підтверджується аналізом відповідного тегу dhcp, який присвоєно 87 932 документам. Найбільший інтерес для безпекового аудиту викликає фіксація значного обсягу записів із тегом error, кількість яких у системних журналах сягає 9 896 записів. Така висока концентрація помилок у підсистемі DHCP свідчить про виникнення збоїв у обробки пакетів, спричинених повним вичерпанням доступного адресного простору внаслідок зловмисної атаки.

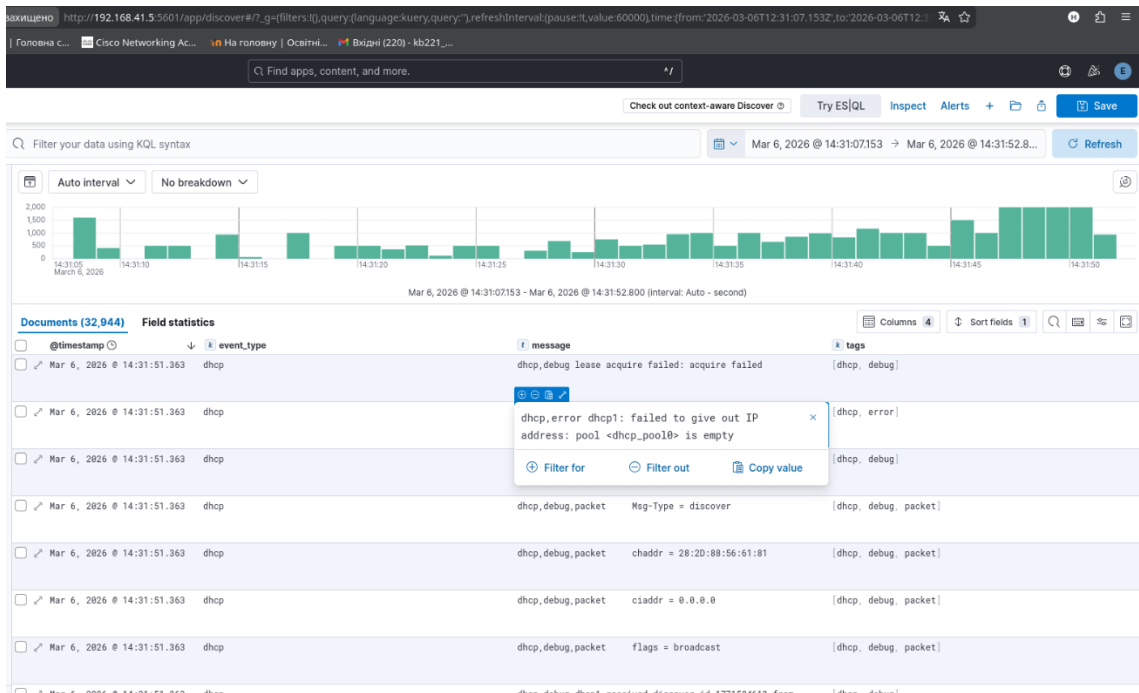


Рисунок 3.17 – Моніторинг подій DHCP-сервера, процес DHCP DISCOVER та призначення IP-адреси у обладнання з системою RouterOS

Аналіз часового зрізу системних повідомлень на Рисунку 3.17 дозволяє детально реконструювати покрокову послідовність дій та внутрішню логіку функціонування DHCP-сервера операційної системи MikroTik RouterOS під час обробки зловмисного запиту. Хронологічний аналіз телеметрії у вкладці Discover відображає таку послідовність виконання операцій ядром маршрутизатора:

- 14:31:51.363 — Реєстрація вхідного запиту: Системний демон DHCP фіксує факт фізичного надходження пакета повідомленням `dhcp, debug, packet dhcp1 received discover id...` від інфраструктури, що перебуває під атакою. Процесор маршрутизатора виділяє первинний потік обробки для нової транзакції.
- 14:31:51.363 — Аналіз прапорців передачі: Наступний службовий лог `dhcp, debug, packet flags = broadcast` відображає етап зчитування заголовків пакета. Сервер визначає, що запит має широкомовний характер, що є стандартним для процедури пошуку DHCP-серверів у сегменті.
- 14:31:51.363 — Верифікація статусу клієнта: Запис телеметрії `dhcp, debug, packet ciaddr = 0.0.0.0` показує, що поточна IP-адреса відправника є нульовою. Це підтверджує, що пристрій заявляє про повну відсутність мережних налаштувань та ініціює процедуру їхнього первинного отримання.

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				63
Змн.	Арк.	№ докум.	Підпис	Дата		

- 14:31:51.363 — Ідентифікація апаратної адреси: Повідомлення dhcp, debug, packet chaddr = 28:2D:88:56:61:81 відображає десеріалізацію фізичного ідентифікатора. Ядро RouterOS вилучає згенеровану утилітою штучного флуду апаратну MAC-адресу клієнта (chaddr) для її подальшої реєстрації у базі даних оренди.

- 14:31:51.363 — Остаточна класифікація типу повідомлення: Запис dhcp, debug, packet Msg-Type = discover завершує етап синтаксичного аналізу полів пакета. Операційна система остаточно інтерпретує транзакцію як пошуковий запит DHCP DISCOVER.

- 14:31:51.363 — Фіксація критичного вичерпання ресурсів адресного простору: Повідомлення про помилку з тегом error виду dhcp, error dhcp1: failed to give out IP address: pool <dhcp_pool0> is empty фіксує момент повного краху розподілу адрес. Маршрутизатор намагається виділити логічний ресурс із локального адресного простору <dhcp_pool0>, але виявляє його стовідсоткове вичерпання фіктивними прив'язками, створеними під час атаки DHCP Starvation.

- 14:31:51.363 — Анулювання транзакції оренди: Фінальний лог dhcp, debug lease acquire failed: acquire failed сигналізує про завершення обробки запиту з деструктивним результатом. Процедура призначення мережних параметрів остаточно скасовується, підтверджуючи стійкий стан відмови в обслуговуванні (DoS) як для зловмисних, так і для будь-яких легітимних запитів користувачів.

Зіставлення телеметрії безпекових подій обох платформ підтверджує, що вразливість протоколу DHCP перед загрозою штучного вичерпання адресного простору виникає незалежно від архітектури операційної системи конкретного маршрутизатора. При цьому розгортання централізованого логування у стеку Elastic дозволяє повністю нівелювати відмінності у синтаксисі журналів різних вендорів, оскільки інструменти фільтрації, пошуку та аналізу часових зрізів у Kibana трансформують розрізнені текстові потоки Cisco IOS та MikroTik RouterOS у єдині, структуровані маркери компрометації. Таке агрегування даних прискорює локалізацію мережних аномалій на ранніх етапах, що дозволяє оптимізувати роботу адміністратора безпеки та плавно змістити фокус дослідження з базових інфраструктурних

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		64

сервісів мережного рівня на деструктивні впливи прикладного характеру, зокрема на телеметрію компрометації систем автентифікації та вебатак.

3.3. Підробка логів у системі централізованого логування

У системах забезпечення інформаційної безпеки журналювання подій є дуже важливим інструментом для моніторингу стану системи, виявлення аномалій та розслідування інцидентів. Проте ефективність цього інструменту повністю залежить від цілісності та автентичності даних, що передаються й зберігаються. Якщо сервіси збору та транспортування логів не захищені за допомогою криптографічних сертифікатів та механізмів суворої автентифікації, вони стають вразливими до ряду критичних атак, зокрема до підробки та навмисного викривлення даних.

Реалізація атаки зловмисником зазвичай починається з етапу розвідки. Цей процес дозволяє визначити архітектуру мережі, виявити відкриті порти та ідентифікувати запущені сервіси й версії їхнього програмного забезпечення. У випадку виявлення незахищених каналів передачі службових даних, атакуюча сторона отримує можливість безперешкодно впроваджувати хибну інформацію безпосередньо у потоки журналювання.

Основна небезпека маніпулювання лог-файлами та масового надсилання хибних записів полягає не лише у приховуванні дій зловмисника, а й у створенні штучного інформаційного шуму для системного адміністратора або SOC-аналітика. Отримуючи сотні сфальсифікованих критичних сповіщень, адміністратор втрачає можливість швидко локалізувати реальну проблему та правильно оцінити вектор атаки. Такий дезорієнтуючий фактор суттєво збільшує час реагування на інцидент, що надає зловмиснику додаткове часове вікно для виконання неправомірних дій — від підвищення привілеїв у системі до несанкціонованого вивантаження конфіденційних даних — ще до того, як буде впроваджено перші захисні заходи.

Для практичного підтвердження цієї тези було реалізовано експериментальне дослідження за моделлю «сірого ящика» (Grey Box). За умовами цього сценарію, умовний зловмисник перебуває у віддаленій мережі, але заздалегідь володіє інформацією про загальну топологію внутрішньої інфраструктури об'єкта захисту. На

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		65

початковому етапі атаки зловмисник здійснив послідовне сканування підмереж локальної мережі, в результаті чого було успішно виявлено та локалізовано цільову IP-адресу сервера — 192.168.41.5. З метою виявлення активних служб та потенційних точок входу на цьому хості було проведено комплексне сканування всіх 65535 TCP-портів із залученням вбудованих скриптів NSE, а також додаткове сканування UDP-портів для пошуку служб збору метрик і журналів.

```
Starting Nmap 7.98 ( https://nmap.org ) at 2026-03-18 09:33 +0200
NSE: Loaded 158 scripts for scanning.
NSE: Script Pre-scanning.
Initiating NSE at 09:33
Completed NSE at 09:33, 0.00s elapsed
Initiating NSE at 09:33
Completed NSE at 09:33, 0.00s elapsed
Initiating NSE at 09:33
Completed NSE at 09:33, 0.00s elapsed
Initiating Ping Scan at 09:33
Scanning 192.168.41.5 [2 ports]
Completed Ping Scan at 09:33, 0.00s elapsed (1 total hosts)
Initiating Connect Scan at 09:33
Scanning 192.168.41.5 [65535 ports]
Discovered open port 22/tcp on 192.168.41.5
Discovered open port 80/tcp on 192.168.41.5
Discovered open port 9300/tcp on 192.168.41.5
Discovered open port 9200/tcp on 192.168.41.5
Discovered open port 5601/tcp on 192.168.41.5
Completed Connect Scan at 09:33, 3.28s elapsed (65535 total ports)
```

Рисунок 3.18 – Базове сканування цільового порту

Детальний аналіз конфігурації виявлених сервісів дозволяє зрозуміти логіку подальших дій зловмисника у віддаленій мережі. З одного боку, результати сканування свідчать про наявність базових налаштувань безпеки всередині самого сховища даних: порти Elasticsearch 9200 та 9300 використовують шифрування SSL/TLS та повертають помилку 401 при спробі прямого доступу до REST API. Це означає, що віддалений атакуючий не може безпосередньо підключитися до бази даних, щоб видалити або модифікувати вже наявні індекси логів без дійсних адміністративних облікових даних.

З іншого боку, архітектура системи містить критичні вразливості на етапах передачі та візуалізації інформації. Веб-інтерфейс аналітики Kibana (порт 5601) та веб-сервер Apache (порт 80) функціонують через незахищений протокол HTTP, що робить трафік вразливим до перехоплення. Проте точкою входу для реалізації атаки з дезорієнтації є виявлені UDP-порти 5515 та 5516. У типових

інфраструктурах збору подій безпеки (наприклад, на базі рішень Logstash або Rsyslog) такі порти конфігуруються для потокового приймання логів від віддалених пристроїв.

Оскільки протокол UDP працює без встановлення постійного з'єднання, він повністю вразливий до атаки типу IP-spoofing. Знаючи топологію мережі, зломисник отримує можливість надсилати на ці UDP-порти спеціально сформовані пакети, підставляючи у заголовок IP-адреси довірених пристроїв. Сервер збору логів сприйме ці повідомлення як легітимні та внесе їх у загальну базу даних.

Таблиця 3.1

Результати мережного сканування цільового хоста

Порт / Протокол	Статус	Сервіс	Версія програмного забезпечення / Додаткові дані
22 / TCP	Відкритий	SSH	OpenSSH 8.9p1 Ubuntu 3ubuntu0.5 (Ubuntu Linux)
80 / TCP	Відкритий	HTTP	Apache httpd 2.4.52 (Ubuntu) — сторінка за замовчуванням
5601 / TCP	Відкритий	HTTP	Elasticsearch Kibana (назва сервера: savelastic)
9200 / TCP	Відкритий	SSL/HTTP	Elasticsearch REST API 7.0+ (плагін Shield; автентифікація)
9300 / TCP	Відкритий	SSL	Elasticsearch Transport (внутрішній транспортний рівень)
5515 / UDP	Відкритий / фільтрований	Unknown	Невідома служба (потенційний порт прийому логів)
5516 / UDP	Відкритий / фільтрований	Unknown	Невідома служба (потенційний порт прийому логів)

Таким чином, виявлена конфігурація дозволяє зломиснику перейти від етапу розвідки до безпосереднього моделювання атаки — генерації штучного лог-спаму. Це повністю підтверджує описану раніше загрозу: імітуючи фейкові критичні збої або атаки в інших сегментах мережі, зломисник створює «димову завісу» для SOC-аналітика, виграючи час для реального несанкціонованого доступу до системи.

Наступним кроком дослідження став етап безпосереднього моделювання атаки, для реалізації якої було розроблено спеціалізований інструментарій мовою програмування Python із використанням бібліотеки з низькорівневим маніпулювання мережними пакетами Scapy. Технічна суть розробленого сценарію полягає в автоматизації процесу IP- spoofing та високочастотної сфальсифікованих UDP-пакетів у напрямку цільового сервера 192.168.41.5. За допомогою конструктора пакетів Scapy для кожного повідомлення динамічно формувався заголовок мережного рівня IP, куди скрипт підставляв адреси легітимного обладнання: 193.41.11.2 для імітації маршрутизатора MikroTik (порт призначення 5516) та 192.168.41.1 / 192.168.41.2 для імітації пристроїв Cisco (порт призначення 5515). Вихідні UDP-порти відправника генерувалися випадковим чином за допомогою функції `gandm.randint(32768, 61000)` для імітації реального мережного трафіку та обходу можливих статичних фільтрів. Текстове навантаження (Payload) містило заздалегідь підготовлені шаблони критичних системних помилок. Цикл симуляції був налаштований на автоматичне виконання ітерацій із мінімальним інтервалом затримки, що дозволило забезпечити високу щільність атаки та згенерувати інтенсивний потік лог-спаму за надзвичайно короткий проміжок часу.

У результаті запуску розробленого Python-скрипта було зафіксовано суттєві деструктивні зміни в інформаційному полі системи моніторингу. Аналіз панорами моніторингу в інтерфейсі Kibana безпосередньо в момент атаки демонструє стрімке та аномальне зростання загальної кількості оброблених записів до 4412 повідомлень за короткий 15-хвилинний проміжок часу. На основі аналізу розподілу джерел встановлено, що система SIEM успішно сприйняла підроблені пакети, зафіксувавши надходження журналів нібито від трьох різних легітимних пристроїв: 193.41.11.2 (54,13%), 192.168.41.2 (23,66%) та 192.168.41.1 (22,21%). Головною технічною особливістю цієї фази атаки стало навмисне порушення структури даних зловмисником. Статистика розподілу тегів свідчить, що мітки `_grokparsefailure` та `%{tag}` охоплюють по 29,06% від загального обсягу логів (по 2024 записи). Наявність такої кількості помилок парсингу вказує на те, що синтаксис надісланих Python-скриптом текстових рядків навмисно не відповідав конфігураційним

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		68

регулярним виразам (Grok-патернам) у Logstash[15]. Це призвело до збою в системі нормалізації даних, змушуючи конвеєр Logstash витратити значні обчислювальні ресурси процесора на обробку виняткових ситуацій та заповнювати базу даних Elasticsearch неструктурованим інформаційним сміттям.

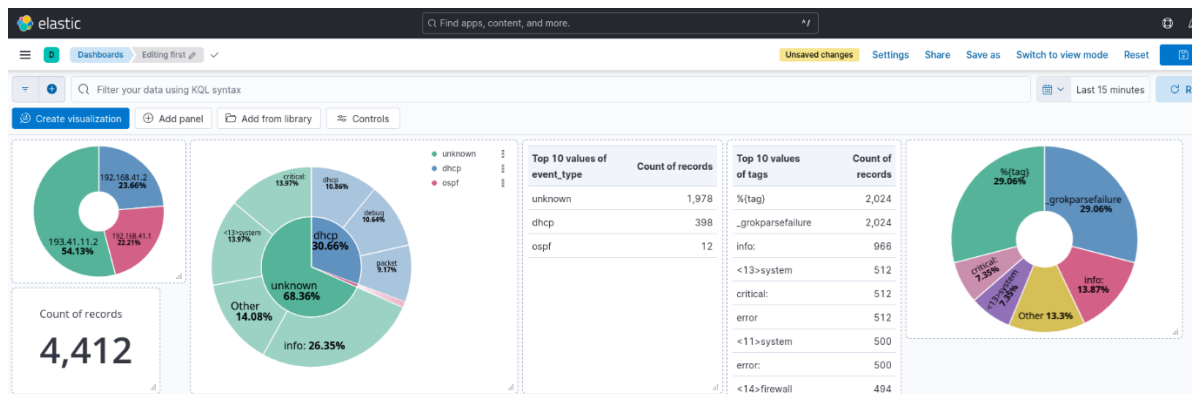


Рисунок 3.19 – Інформаційна панель під час генерації нелегітимних логів

Більш детальний аналіз вмісту повідомлень на вкладці Discover дозволяє виявити конкретні дезорієнтуючі чинники, які відображаються на моніторі адміністратора і вимагають негайного реагування. Усі згенеровані записи є нелегітимними, проте їхній зміст імітує одночасне настання кількох критичних інцидентів в інфраструктурі. По-перше, записи від імені джерела 192.168.41.1 містять повідомлення про аварійне завершення роботи веб-сервера Nginx на критичному вузлі Server-01 (%SYS-2-CRITICAL: Service 'nginx' terminated unexpectedly). По-друге, логи від джерела 193.41.11.2 сигналізують про апаратний збій мережі у вигляді відключення магістрального інтерфейсу маршрутизатора (interface ether1 link down). По-третє, потокові записи від джерела 192.168.41.2 імітують роботу міжмережного екрана Cisco (Router-HQ), який масово блокує несанкціоновані спроби передачі трафіку з внутрішнього хоста 172.16.0.5.

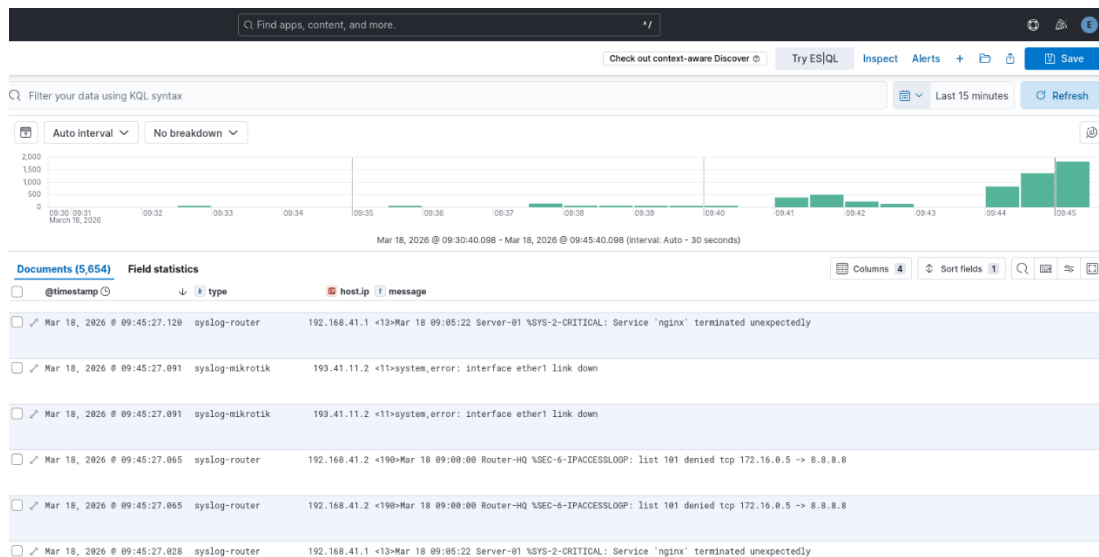


Рисунок 3.20 – Перегляд логів під час їх нелегітимного генерування

Сформований зловмисником потік даних чинить комплексний дезорієнтуючий та психологічний вплив на персонал захисту інформаційних систем. Побачивши масове виникнення помилок `_grokparsefailure`, системний адміністратор першочергово припустить наявність внутрішнього збою в конфігураціях самого стеку Elastic або некоректну зміну форматів виводу логів після планового оновлення ПЗ маршрутизаторів. Замість моніторингу безпеки, аналітик почне витратити час на дебаг конвеєра Logstash та виправлення Grok-патернів, яких насправді міняти не потрібно. Одночасно з цим, поява фантомних повідомлень про падіння інтерфейсів MikroTik та зупинку сервісів Nginx змусить адміністратора хаотично перевіряти фізичну доступність пристроїв у мережі. Оскільки в реальності всі ці вузли працюють у штатному режимі, персонал захисту потрапляє у замкнене коло діагностики неіснуючих несправностей. Таким чином, експеримент наочно доводить, що відсутність криптографічного захисту та автентифікації на етапі транспортування логів дозволяє зловмиснику повністю захопити контроль над інформаційним полем адміністратора і виграти час для непомітного виконання неправомірних дій через інші канали.

Для нівелювання виявлених архітектурних уразливостей та унеможливлення проведення атак типу IP-spoofing та генерації хибних логів необхідно впровадити комплексну стратегію захисту транспортного рівня системи логування. Першочерговим кроком є повна відмова від протоколу UDP на користь гарантованого

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				70
Змн.	Арк.	№ докум.	Підпис	Дата		

з'єднання TCP. Перехід на TCP дозволяє встановити сесію з тристороннім рукошлякуванням, що захищає від базової підміни IP-адрес наосліп. Однак, для захисту від більш складних атак у віддаленій мережі та забезпечення конфіденційності й цілісності даних, транспортний канал необхідно захистити за допомогою криптографічного протоколу TLS. Найбільш ефективним рішенням у цьому контексті є розгортання двосторонньої автентифікації. За такої архітектури не лише клієнтський пристрій перевіряє автентичність сервера збору логів, але й сам сервер вимагає та валідує цифровий сертифікат від кожного маршрутизатора чи сервера, що намагається передати дані. Це виключає можливість підробки як самого сервера моніторингу, так і джерела генерації подій, оскільки без наявності легітимного закритого ключа та сертифіката інжекція лог-спаму стає технічно неможливою.

Практична реалізація запропонованого захисного механізму передбачає створення виділеного внутрішнього центру сертифікації. Як відображено в інженерній практиці розгортання захищеного стеку, спочатку за допомогою утиліти OpenSSL генерується кореневий ключ та самопідписаний сертифікат. На основі цього кореня створюються сертифікати для компонентів інфраструктури, зокрема генерується запис на підпис сертифіката (CSR) для сервера Logstash із прив'язкою до його реальної IP-адреси. Отриманий закритий ключ сервера конвертується у сумісний формат PKCS#8 для забезпечення коректної інтеграції з конвеєром даних. У конфігураційному файлі Logstash створюється новий вхідний канал, який замість UDP прослуховує TCP-порт. У цьому блоці конфігурації обов'язково активується параметр захисту `ssl_enable => true`, а також прописуються шляхи до сертифіката сервера, приватного ключа та ланцюжка довіри кореневого СА. Для переходу від базового одностороннього шифрування до повноцінного mTLS з режимом обов'язкової валідації реєр-вузлів.

На стороні мережного обладнання, що виступає джерелом журналів подій, здійснюється дзеркальне налаштування довіри до розгорнутого центру сертифікації. У режимі глобальної конфігурації пристрою створюється криптографічна точка довіри за допомогою команди `crypto pki trustpoint ELK-CA`, де вимикається автоматична перевірка відкликання сертифікатів для ізольованих мереж. Після цього

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		71

виконується процедура автентифікації точки довіри, під час якої в термінал пристрою вноситься легітимний сертифікат кореневого СА у форматі Base64. Фінальним етапом безпечної перебудови архітектури є переведення служби системного логування на роботу через захищений транспортний протокол. Таке налаштування змушує маршрутизатор ініціювати TLS-рукоштовування з Logstash перед відправкою будь-якого повідомлення, що гарантує цілісність інформаційного поля SIEM-системи та ускладнює успішність раніше продемонстровану атаку з дезорієнтації персоналу захисту.

Висновки до третього розділу

У третьому розділі було виконано практичне дослідження розгорнутої підсистеми централізованого логування мережних пристроїв та проведено комплексний аналіз подій інформаційної безпеки в умовах симуляції критичних загроз. Результати експериментальних випробувань підтвердили, що мережне обладнання під управлінням різних операційних систем, зокрема Cisco IOS та MikroTik RouterOS, є однаково вразливим до деструктивних впливів на каналному та мережному рівнях моделі OSI. Моделювання атак на протокол динамічної маршрутизації OSPF через ін'єкцію нелегітимних LSA-пакетів, атак типу DHCP Starvation шляхом масової відправки штучними Discover-запитами, а також спроб прямої підробки системних журналів (log spoofing) засвідчили високі ризики порушення цілісності та доступності інфраструктури за відсутності належного моніторингу телеметрії.

Доведено, що використання централізованого збору логів дозволяє повністю нівелювати перевагу зловмисника в часі та забезпечує активний захист інформаційної системи організації. Завдяки консолідації подій у єдиній точці моніторингу системний адміністратор або аналітик безпеки отримує можливість фіксувати аномальну активність безпосередньо в момент, коли атакуючий тільки починає здійснювати перші кроки втручання в систему та розвідку мережі. Таке рішення забезпечує миттєву ідентифікацію скомпрометованої ланки інфраструктури, дозволяючи чітко визначити проблемний сегмент, оперативно локалізувати загрозу та

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		72

вчасно прийняти ефективні керуючі рішення щодо захисту периметра ще до того, як дії зловмисника призведуть до незворотних наслідків чи повної відмови у сервісах.

Крім того, впровадження аналітичної платформи дозволило кардинально спростити процеси обробки, представлення та інтерпретації великих масивів системних журналів, що надходять від гетерогенних пристроїв. Завдяки інтеграції сучасних інструментів швидкого повнотекстового пошуку, динамічної фільтрації за структурованими полями, агрегації схожих подій та наочних інтерактивних панелей відпала потреба у тривалому ручному аналізі розрізнених файлів конфігурацій. Це суттєво підвищує загальну ефективність адміністрування, мінімізує середній час реакції на інциденти кібербезпеки та дозволяє оперативно викривати навіть замасковані спроби підробки або видалення логів.

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				73
Змн.	Арк.	№ докум.	Підпис	Дата		

ВИСНОВКИ

У випускній кваліфікаційній роботі бакалавра успішно вирішено актуальне науково-практичне завдання щодо проєктування та практичної реалізації підсистеми централізованого логування мережних пристроїв організації на базі стеку Elastic Stack (ELK). Головним результатом роботи відповідно до поставленої мети стало створення комплексної SIEM-подібної системи мережного моніторингу та аналізу телеметрії безпеки в реальному часі. Розроблена архітектура забезпечує збір журналів подій від різноманітного (гетерогенного) мережного обладнання, їхню централізовану обробку, нормалізацію та інтерактивну візуалізацію, що дозволяє оперативно ідентифікувати кіберзагрози та аномалії в інфраструктурі організації.

Під час виконання роботи було реалізовано повний перелік закладених функцій, які демонструють високий ступінь відповідності вихідному технічному завданню. Зокрема, було успішно виконано обґрунтування та вибір апаратних засобів, базою для яких стали маршрутизатори корпоративного рівня Cisco 2821 та MikroTik RB5009UG+S+, що функціонують під управлінням операційних систем Cisco IOS та RouterOS v7 відповідно. У межах підсистеми агрегації та збереження даних розгорнуто стек інструментів ELK, що включає пошукову систему Elasticsearch, інтелектуальний конвеєр обробки Logstash та аналітичну панель Kibana. Для забезпечення безпеки потоків телеметрії впроваджено захищену архітектуру передачі повідомлень за стандартом Syslog із використанням протоколів шифрування TLS, цифрових сертифікатів X.509 для взаємної автентифікації вузлів, а також HTTPS-шифрування для взаємодії компонентів і доступу до веб-інтерфейсу.

Важливою реалізованою функцією підсистеми є гнучка конвеєризація та нормалізація різноманітних даних. Конфігурація Logstash була розділена на окремі незалежні конвеєри, у межах яких застосовано спеціалізовані grok-фільтри. Це дозволило успішно трансформувати неструктуровані текстові повідомлення від пристроїв Cisco та MikroTik у стандартизований аналітичний формат JSON із виділенням ключових полів, що суттєво полегшило подальший пошук, агрегацію та побудову кореляційних зв'язків.

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		74

Для перевірки працездатності спроектованої топології мережі спочатку було здійснено її попереднє моделювання в емуляційному середовищі GNS3, після чого інфраструктуру повністю перенесено та розгорнуто на реальному лабораторному фізичному стенді. Використання справжнього апаратного комплексу дозволило забезпечити абсолютну автентичність системних журналів та зафіксувати точні реакції мережного обладнання на деструктивні впливи на фізичному та каналному рівнях. На базі створеного стенда за допомогою інтеграції спеціалізованого програмного забезпечення (серверного пакета маршрутизації FRR, утиліти Yersinia CLI та низькорівневих скриптів Python із бібліотекою Scapy) проведено практичні експерименти з симуляції трьох векторів деструктивного впливу: атак на протокол динамічної маршрутизації OSPF із впровадженням фіктивних LSA-пакетів, штучного виснаження адресного простору методом DHCP Starvation, а також підробки журналів подій методом IP-spoofing.

Практична цінність отриманих результатів полягає у створенні готового до розгортання, масштабованого та відмовостійкого SIEM-подібного рішення з відкритим вихідним кодом, яке дозволяє службам моніторингу безпеки та системним адміністраторам безперервно контролювати стан інфраструктури. У ході дослідження Kibana-дашбордів було виявлено унікальні телеметричні аномалії та зафіксовано часові межі ефективності атак. Зокрема, визначено, що операційна система Cisco IOS використовує 5-хвилинний таймаут для незавершених DHCP-транзакцій, після чого автоматично запускає захисні механізми та повертає адреси в пул, що накладає обмеження на дії злоумисника. На основі отриманих результатів тестування розроблено конкретні практичні пропозиції щодо підвищення ефективності та безпеки об'єкта дослідження, зокрема: впровадження криптографічної автентифікації MD5 для захисту маршрутизації OSPF, суворе обмеження доступу до портів збору логів, а також перехід на шифровані протоколи передачі телеметрії Syslog over TLS.

Напрямки подальшого наукового дослідження та вдосконалення розробленого програмного комплексу пов'язані з розширенням його аналітичних можливостей та автоматизацією процесів реагування. Перспективним кроком є розробка та

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				75
Змн.	Арк.	№ докум.	Підпис	Дата		

інтеграція комплексних правил кореляції подій, які дозволять автоматично генерувати сповіщення про складні багатовекторні атаки без безпосередньої участі оператора. Окрім цього, доцільним є впровадження модулів машинного навчання для виявлення прихованих аномалій, що не піддаються статичним фільтрам, а також розширення архітектури шляхом інтеграції з агентами кінцевих точок для наскрізного аудиту хостів. Нарешті, розвиток системи передбачає реалізацію концепції SOAR через створення автоматизованих скриптів, які у випадку фіксації критичних логів безпеки самостійно блокуватимуть атакуючі IP-адреси на брандмауерах Cisco та MikroTik.

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				76
Змн.	Арк.	№ докум.	Підпис	Дата		

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Beschokov M. Transmission control protocol - TCP [Електронний ресурс] / Mukhadin Beschokov // wallarm. – Режим доступу: <https://www.wallarm.com/what/transmission-control-protocol-tcp> (дата звернення: 08.12.2025).
2. Beschokov M. User datagram protocol [Електронний ресурс] / Mukhadin Beschokov // Wallarm. – Режим доступу: <https://www.wallarm.com/what/user-datagram-protocol> (дата звернення: 08.12.2025).
3. Cisco IOS system messages guide, release 15.0SY - overview [Електронний ресурс] // Cisco Systems. – Режим доступу: https://www.cisco.com/c/en/us/td/docs/ios/15_0sy/system/messages/15sysmg/sm15syov r.html (дата звернення: 12.01.2026).
4. Create a Logstash pipeline [Електронний ресурс] // Elastic. – Режим доступу: <https://www.elastic.co/docs/reference/logstash/creating-logstash-pipeline> (дата звернення: 10.03.2026).
5. Elastic Search documentation [Електронний ресурс] // Elastic. – Режим доступу: <https://www.elastic.co/docs/solutions/search> (дата звернення: 09.03.2026).
6. Molenaar R. Cisco IOS syslog messages [Електронний ресурс] / Rene Molenaar // NetworkLessons.com. – Режим доступу: <https://networklessons.com/system-management/cisco-ios-syslog-messages> (дата звернення: 09.03.2026).
7. Normunds R. Diagnostics, monitoring and troubleshooting [Електронний ресурс] / R. Normunds // help.mikrotik. – Режим доступу: <https://help.mikrotik.com/docs/spaces/ROS/pages/328094/Log> (дата звернення: 26.12.2025).
8. . Dynamic host configuration protocol [Електронний ресурс]. – На заміну RFC 1541 ; чинний від 1997-03-01. – Вид. офіц. – Bucknell University : RFC Editor, 1997. – 45 с. – Режим доступу: <https://www.rfc-editor.org/rfc/rfc2131.html> (дата звернення: 06.04.2026).

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				77
Змн.	Арк.	№ докум.	Підпис	Дата		

9. RFC 2328. OSPF version 2 [Електронний ресурс]. – На заміну RFC 2178 ; чинний від 1998-04-01. – Вид. офіц. – Fremont : RFC Editor, 1998. – 244 с. – Режим доступу: <https://www.rfc-editor.org/rfc/rfc2328.html> (дата звернення: 24.03.2026).

10. RFC 5424. The Syslog Protocol [Електронний ресурс]. – На заміну RFC 3164 ; чинний від 2009-03-01. – Вид. офіц. – Fremont : RFC Editor, 2009. – 38 с. – Режим доступу: <https://www.rfc-editor.org/rfc/rfc5424.html> (дата звернення: 10.02.2026).

11. RFC 5425. Transport Layer Security (TLS) Transport Mapping for Syslog [Електронний ресурс]. – чинний від 2009-03-01. – Вид. офіц. – Fremont : RFC Editor, 2009. – 15 с. – Режим доступу: <https://www.rfc-editor.org/rfc/rfc5425.html> (дата звернення: 08.04.2026).

12. Scapy documentation: Usage [Електронний ресурс] // Scapy. – Режим доступу: <https://scapy.readthedocs.io/en/latest/usage.html> (дата звернення: 15.03.2026).

13. ДСТУ ISO/IEC 27001:2015. Інформаційні технології. Методи захисту системи. Системи управління інформаційною безпекою. Вимоги (ISO/IEC 27001:2013, IDT). – На заміну ДСТУ ISO/IEC 27001:2010 ; чинний від 2016-12-01. – Вид. офіц. – Київ : ДП «УкрНДНЦ», 2016. – 32 с

14. Khalil M. What is DHCP starvation [Електронний ресурс] / Mohammed Khalil // Deepstrike. – Режим доступу: <https://deepstrike.io/blog/what-is-dhcp-starvation> (дата звернення: 12.04.2026).

15. Grok filter plugin [Електронний ресурс] // Elastic. – Режим доступу: <https://www.elastic.co/docs/reference/logstash/plugins/plugins-filters-grok> (дата звернення: 07.03.2026).

16. FRRouting Documentation: OSPFv2 [Електронний ресурс] // FRRouting. – Режим доступу: <https://docs.frrouting.org/en/latest/ospfd.html> (дата звернення: 28.03.2026).

17. Berrueta A. Layer 2 Attacks [Електронний ресурс] / Andres Berrueta // Black Hat. – Режим доступу: https://blackhat.com/presentations/bh-europe-05/BH_EU_05-Berrueta_Andres/BH_EU_05_Berrueta_Andres.pdf (дата звернення: 03.04.2026).

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				78
Змн.	Арк.	№ докум.	Підпис	Дата		

18. Lyon G. Nmap reference guide [Електронний ресурс] / Gordon Lyon // Nmap. – Режим доступу: <https://nmap.org/book/man.htm> (дата звернення: 15.03.2026).

19. OSPF authentication configuration guide [Електронний ресурс] // Cisco Systems. – Режим доступу: <https://www.cisco.com/c/en/us/support/docs/ip/open-shortest-path-first-ospf/13697-25.html> (дата звернення: 18.03.2026).

20. Ruby filter plugin [Електронний ресурс] // Elastic. – Режим доступу: <https://www.elastic.co/docs/reference/logstash/plugins/plugins-filters-ruby> (дата звернення: 10.03.2026).

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				79
Змн.	Арк.	№ докум.	Підпис	Дата		

ДОДАТКИ

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				80
Змн.	Арк.	№ докум.	Підпис	Дата		

Конфігурація маршрутизатора з системою Cisco IOS

```
!  
! Last configuration change at 11:43:52 UTC Wed Mar 18 2026 by admin  
version 15.1  
service timestamps debug datetime msec  
service timestamps log datetime msec  
no service password-encryption  
!  
hostname R1  
!  
boot-start-marker  
boot-end-marker  
!  
!  
logging discriminator NO_FAN mnemonics drops FAN_LOW_RPM  
logging buffered discriminator NO_FAN 4096  
logging console discriminator NO_FAN  
enable password admin  
!  
no aaa new-model  
!  
!  
dot11 syslog  
ip source-route  
!  
!  
ip cef  
!  
ip dhcp excluded-address 192.168.41.1  
ip dhcp excluded-address 192.168.41.5  
!  
ip dhcp pool LAN  
network 192.168.41.0 255.255.255.0  
default-router 192.168.41.1  
dns-server 192.168.41.1  
!  
!  
ip domain name lab.local  
no ipv6 cef  
!  
multilink bundle-name authenticated  
!  
!  
voice-card 0  
!  
crypto pki token default removal timeout 0  
!  
license udi pid CISCO2821 sn FTX1119A30D  
archive  
log config
```

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				81
Змн.	Арк.	№ докум.	Підпис	Дата		

```

logging enable
notify syslog contenttype plaintext
hidekeys
username admin password 0 admin
!
redundancy
!
interface GigabitEthernet0/0
description mikrotik
ip address 193.41.11.1 255.255.255.252
ip ospf authentication message-digest
ip ospf message-digest-key 1 md5 *****
ip ospf network point-to-point
duplex auto
speed auto
!
interface GigabitEthernet0/1
description LAN
ip address 192.168.41.1 255.255.255.0
duplex auto
speed auto
!
interface Serial0/0/0
no ip address
shutdown
no fair-queue
clock rate 125000
!
interface Serial0/0/1
no ip address
shutdown
clock rate 125000
!
interface FastEthernet0/1/0
no ip address
!
interface FastEthernet0/1/1
no ip address
!
interface FastEthernet0/1/2
no ip address
!
interface FastEthernet0/1/3
no ip address
!
interface Vlan1
no ip address
!
router ospf 4111
area 0 authentication message-digest
network 192.168.41.0 0.0.0.255 area 0
network 193.41.11.0 0.0.0.3 area 0

```

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				82
Змн.	Арк.	№ докум.	Підпис	Дата		

```

!
ip forward-protocol nd
no ip http server
no ip http secure-server
!
logging trap debugging
logging host 192.168.41.5 transport udp port 5515 discriminator NO_FAN
logging host 194.41.11.5 transport udp port 5515 discriminator NO_FAN
!
control-plane
!
mgcp profile default
!
line con 0
line aux 0
line vty 0 4
  login local
  transport input ssh
!
scheduler allocate 20000 1000
end

```

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				83
Змн.	Арк.	№ докум.	Підпис	Дата		

Конфігурація маршрутизатора з системою RouterOS

```
# jan/02/1970 04:33:25 by RouterOS 7.5
# software id = Z5A1-VKUU
#
# model = RB5009UG+S+
# serial number = HD8082PGZ07
/interface wireless security-profiles
set [ find default=yes ] supplicant-identity=MikroTik
add authentication-types=wpa2-psk mode=dynamic-keys name=testProfile supplicant-identity=""
/ip pool
add name=dhcp_pool0 ranges=194.41.11.2-194.41.11.254
/ip dhcp-server
add address-pool=dhcp_pool0 interface=ether4 name=dhcp1
/routing ospf instance
add disabled=no name=ospf-main router-id=194.41.11.1
/routing ospf area
add disabled=no instance=ospf-main name=backbone
/system logging action
add name=elasticsearch remote=192.168.41.5 remote-port=5516 target=remote
/ip address
add address=193.41.11.2/30 interface=ether3 network=193.41.11.0
add address=194.41.11.1/24 interface=ether4 network=194.41.11.0
/ip dhcp-server network
add address=194.41.11.0/24 dns-server=194.41.11.1 gateway=194.41.11.1
/routing ospf interface-template
add area=backbone auth=md5 auth-id=1 auth-key=***** disabled=no interfaces=ether3
networks=193.41.11.0/30 \
    type=ptp
add area=backbone disabled=no interfaces=ether4 networks=194.41.11.0/24
/system logging
add action=elasticsearch topics=info
add action=elasticsearch topics=warning
add action=elasticsearch topics=error
add action=elasticsearch topics=critical
add action=elasticsearch topics=debug
```

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				84
Змн.	Арк.	№ докум.	Підпис	Дата		

Конфігурація Elasticsearch

path.data: /var/lib/elasticsearch
path.logs: /var/log/elasticsearch
xpack.security.enabled: true
xpack.security.enrollment.enabled: true
xpack.security.http.ssl:
 enabled: true
 keystore.path: certs/http.p12
xpack.security.transport.ssl:
 enabled: true
 verification_mode: certificate
 keystore.path: certs/transport.p12
 truststore.path: certs/transport.p12
cluster.initial_master_nodes: ["node-1"]
http.host: 0.0.0.0
network.host: 0.0.0.0

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				85
Змн.	Арк.	№ докум.	Підпис	Дата		

Конфігурація Logstash

```

input {
  udp {
    port => 5515
    type => "syslog-router"
  }
}
filter {
  if [type] == "syslog-router" {
    grok {
      match => {
        "message" => "<{%INT:syslog_pri}>{%INT:seq}: \*{%SYSLOGTIMESTAMP:timestamp}:
        %%{%WORD:tag}-{%INT:severity}-{%WORD:event_type}: {%GREEDYDATA:message_text}"
      }
    }
    mutate {
      convert => {
        "severity" => "integer"
        "syslog_pri" => "integer"
      }
    }
  }
  mutate {
    add_tag => [ "%{tag}" ]
  }
}
output {
  stdout { codec => rubydebug }
  elasticsearch {
    hosts => ["https://localhost:9200"]
    user => "elastic"
    password => "SecretPassword"
    ssl => true
    ssl_certificate_verification => false
  }
}

input {
  udp {
    port => 5516
    type => "syslog-mikrotik"
  }
}
filter {
  if [type] == "syslog-mikrotik" {
    grok {
      match => { "message" => "^(?<topics_raw>[^\s]+)\s" }
    }
    mutate {
      split => { "topics_raw" => "," }
    }
  }
}

```

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				86
Змн.	Арк.	№ докум.	Підпис	Дата		

```

    add_field => { "tags" => "%{topics_raw}" }
  }
  mutate {
    split => { "tags" => "," }
    remove_field => [ "topics_raw" ]
  }
  ruby {
    code => '
      tags = event.get("tags")
      if tags.is_a?(Array)
        priority = ["ospf", "bgp", "firewall", "system", "interface", "dhcp"]
        etype = tags & priority
        event.set("event_type", etype[0]) unless etype.empty?
      end
    '
  }
  if ![event_type] {
    mutate { add_field => { "event_type" => "unknown" } }
  }
}
}
output {
  stdout { codec => rubydebug }
  elasticsearch {
    hosts => ["https://localhost:9200"]
    user => "elastic"
    password => "SecretPassword"
    ssl => true
    ssl_certificate_verification => false
  }
}

```

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				87
Змн.	Арк.	№ докум.	Підпис	Дата		

Конфігурація Kibana

```
server.host: "0.0.0.0"
server.port: 5601
elasticsearch.hosts: ["https://localhost:9200"]
elasticsearch.username: "kibana_system"
elasticsearch.password: "SecretPassword"
elasticsearch.ssl.verificationMode: "none"
logging:
  appenders:
    file:
      type: file
      fileName: /var/log/kibana/kibana.log
      layout:
        type: json
  root:
    appenders: [file]
pid.file: /run/kibana/kibana.pid
```

		Петлюк Н.Є.			ККІКБ.КР.Б – 125 – 13 – ПЗ	Арк.
		Покотило О.А.				88
Змн.	Арк.	№ докум.	Підпис	Дата		