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: «Проєкт захищеної мережної інфраструктури на базі пристроїв Cisco та MikroTik з реалізацією систем виявлення та запобігання вторгненням»

Виконала: студентка 4-го курсу групи КБ-22-3
спеціальності 125 «Кібербезпека»
_____ Катерина БІЛОУС
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 випускну кваліфікаційну роботу бакалавра

Студентка: Білоус Катерина Богданівна

Тема роботи: «Проєкт захищеної мережної інфраструктури на базі пристроїв Cisco та MikroTik з реалізацією систем виявлення та запобігання вторгненням»
затверджена Наказом університету від 2 лютого 2026 р. №58/с

Термін здачі студентом закінченої роботи 22 червня 2026 р.

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

- процес організації інформаційної безпеки мережних інфраструктур;
- методи, засоби та технології побудови й захисту мережі на базі обладнання Cisco та MikroTik;
- механізми реалізації систем виявлення та запобігання вторгненням (IDS/IPS).

Обладнання та програмне забезпечення:

1. Емулятор GNS3
2. Гіпервізори Oracle VirtualBox та VMware Workstation
3. Мережні образи операційних систем MikroTik RouterOS та Cisco IOS
4. Операційні системи Ubuntu Desktop 26.04, Ubuntu Server 26.04
5. Система виявлення та запобігання вторгненням Suricata 8.0.3
6. Програмний комплекс Elasticsearch, Kibana, Filebeat версій 8.19.15

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

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв
1	Воротніков В.В.	02.02.2026	02.02.2026
2	Воротніков В.В.	25.03.2026	25.03.2026
3	Воротніков В.В.	15.04.2026	15.04.2026

Студентка _____ Катерина БІЛОУС

Керівник _____ Олександра ПОКОТИЛО

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назви етапів випускної кваліфікаційної роботи	Термін виконання етапів роботи	Примітка
1	Постановка завдання	до 10.02.2026	Виконано
2	Аналіз отриманого завдання та розробка плану	до 25.03.2026	Виконано
3	Огляд літературних джерел	до 15.04.2026	Виконано
4	Вибір моделей обладнання та програмного забезпечення	до 30.04.2026	Виконано
5	Проєктування топології та налаштування мережі	до 22.05.2026	Виконано
6	Інтеграція IDS/IPS та налаштування додаткових сервісів	до 02.06.2026	Виконано
7	Тестування та перевірка працездатності мережі	до 08.06.2026	Виконано
8	Оформлення пояснювальної записки	до 13.06.2026	Виконано
9	Підготовка доповіді та презентації до захисту	до 20.06.2026	Виконано
10	Захист	22.06.2026	

Студентка _____ Катерина БІЛОУС

Керівник _____ Олександра ПОКОТИЛО

АНОТАЦІЯ

Кваліфікаційна робота бакалавра на тему «Проєкт захищеної мережної інфраструктури на базі пристроїв Cisco та MikroTik з реалізацією систем виявлення та запобігання вторгненням» складається з переліку умовних позначень, вступу, трьох розділів, висновків, списку використаної літератури та додатків. Текстова частина викладена на 87 сторінках друкованого тексту. Список використаних джерел містить 22 найменування і займає 2 сторінки. В роботі наведено 92 рисунки та 5 таблиць.

Метою роботи є проєктування та реалізація захищеної мережної інфраструктури на базі обладнання Cisco та MikroTik, яка забезпечує автоматизоване виявлення та нейтралізацію мережних загроз через інтеграцію систем виявлення та запобігання вторгненням.

У роботі визначено основні завдання для побудови захищеної мережі, проаналізовано сучасні методи забезпечення інформаційної безпеки та принципи функціонування IDS/IPS систем. Виконано порівняльний аналіз популярних систем виявлення вторгнень, що дозволило обґрунтувати вибір Suricata як основного інструменту захисту. Показано розробку мережної топології та схеми адресації, впровадження протоколу динамічної маршрутизації OSPF, механізмів контролю доступу та централізованої автентифікації користувачів. Реалізовано систему збору та візуалізації подій безпеки на базі Elastic Stack. Наведено результати тестування інфраструктури, які підтвердили високу ефективність автоматичного блокування загроз, стабільність маршрутизації та відмовостійкість створеної системи.

Ключові слова: CISCO, IDS, IPS, MIKROTIK, МЕРЕЖНА БЕЗПЕКА, КОМУТАТОР, МАРШРУТИЗАТОР, ТРИРІВНЕВА СТРУКТУРА.

ANNOTATION

The bachelor's qualification thesis on the topic "Design of a Secure Network Infrastructure Based on Cisco and MikroTik Devices with the Implementation of Intrusion Detection and Prevention Systems" consists of a list of abbreviations, an introduction, three chapters, conclusions, a list of references, and appendices. The main text is presented on 87 pages of printed material. The list of references includes 22 sources and occupies 2 pages. The work contains 92 figures and 5 tables.

The aim of the thesis is to design and implement a secure network infrastructure based on Cisco and MikroTik equipment that provides automated detection and mitigation of network threats through the integration of intrusion detection and prevention systems.

The study defines the main tasks for building a secure network, analyzes modern methods of ensuring information security, and examines the principles of IDS/IPS systems. A comparative analysis of popular intrusion detection systems was carried out, which justified the selection of Suricata as the primary security tool. The network topology and addressing scheme were designed, the OSPF dynamic routing protocol was implemented, and access control mechanisms along with centralized user authentication were applied. A system for collecting and visualizing security events based on the Elastic Stack was implemented. The testing results of the infrastructure confirmed the high effectiveness of automated threat blocking, routing stability, and the fault tolerance of the developed system.

Keywords: CISCO, IDS, IPS, MIKROTIK, NETWORK SECURITY, SWITCH, ROUTER, THREE-TIER ARCHITECTURE.

ЗМІСТ

ВСТУП 8
РОЗДІЛ 1 ТЕОРЕТИЧНІ ОСНОВИ ТА ПРИНЦИПИ ПОБУДОВИ МЕРЕЖНОЇ ІНФРАСТРУКТУРИ З ВИКОРИСТАННЯМ ТЕХНОЛОГІЙ IDS/IPS 10
1.1 Основи побудови та захисту комп'ютерних мереж 10
1.2 Технологічні рішення для побудови безпечних мереж 15
1.2.1 Огляд Cisco 16
1.2.2 Огляд MikroTik 18
1.3 Аналіз систем виявлення та запобігання вторгненням 19
1.3.1 Принципи функціонування та архітектура систем IDS/IPS 20
1.3.2 Порівняльний аналіз та вибір IDS/IPS 23
1.4 Постановка технічного завдання 26
РОЗДІЛ 2 ПРОЄКТУВАННЯ МЕРЕЖНОЇ ІНФРАКСТРУКТУРИ 28
2.1 Розробка топології та адресації комп'ютерної мережі 28
2.2 Інтеграція системи виявлення вторгнень Suricata 31
2.2.1 Формування технічних вимог та конфігурації 38
2.2.2 Розробка системи візуалізації 40
2.3 Супутні засоби захисту мережі 43
2.4 Вибір мережного обладнання та ПЗ 44
РОЗДІЛ 3 НАЛАШТУВАННЯ ТА ТЕСТУВАННЯ МЕРЕЖНОЇ ІНФРАСТУКТУРИ 49
3.1 Налагодження обладнання та засобів безпеки 49
3.2 Перевірка роботи та тестування безпеки мережі 71
ВИСНОВКИ 84
СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ 86
ДОДАТКИ 88

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

ARP — Address Resolution Protocol

DHCP — Dynamic Host Configuration Protocol

DNS — Domain Name System

HTTP — HyperText Transfer Protocol

ICMP — Internet Control Message Protocol

IDS — Intrusion Detection System

IP — Internet Protocol

IPS — Intrusion Prevention System

JSON — JavaScript Object Notation

NAT — Network Address Translation

OSPF — Open Shortest Path First

SHA — Secure Hash Algorithm

SSH — Secure Shell

TCP — Transmission Control Protocol

TLS — Transport Layer Security

UDP — User Datagram Protocol

ВСТУП

Актуальність теми. В сучасних умовах цифровізації мережі стали основою функціонування підприємств, установ і організацій. Від їхньої надійності, продуктивності та захищеності прямо залежить стабільність та безперервність роботи інформаційних систем і збереження критично важливих даних.

Кіберзагрози невпинно еволюціонують, а їхня кількість стрімко збільшується. У зв'язку з цим особливого значення набуває створення надійної та захищеної мережної інфраструктури, здатної своєчасно виявляти та запобігати потенційним загрозам. Традиційні засоби захисту, такі як міжмережні екрани та антивірусні рішення, не завжди здатні ефективно протидіяти, що спричиняє необхідність впровадження додаткових механізмів безпеки, наприклад систем виявлення та запобігання вторгненням.

Окремим важливим аспектом розробки такої інфраструктури є вибір мережного обладнання. На ринку представлено багато виробників, і поєднання декількох, зокрема пристроїв Cisco та MikroTik, може бути ефективним рішенням. Використання обладнання Cisco забезпечує високу продуктивність, надійність і масштабованість, тоді як пристрої MikroTik вирізняються гнучкістю налаштування та економічною ефективністю.

Дослідження даного питання є актуальним і має важливе практичне значення для забезпечення належного рівня інформаційної безпеки сучасних підприємств.

Метою кваліфікаційної роботи є проєктування та реалізація захищеної мережної інфраструктури на базі обладнання Cisco та MikroTik, яка забезпечує автоматизоване виявлення та нейтралізацію мережних загроз через інтеграцію систем виявлення та запобігання вторгненням.

Визначена мета обумовлює наступні завдання:

- проаналізувати методи побудови захищених мереж;
- розробити та налаштувати мережу з урахуванням специфіки пристроїв Cisco та MikroTik;

- впровадити та налаштувати IDS/IPS для моніторингу та фільтрації мережного трафіку в реальному часі;
- провести тестування мережі для оцінки працездатності та ефективності впроваджених засобів безпеки.

Об’єктом дослідження є процес організації інформаційної безпеки мережних інфраструктур у середовищі з обладнанням різних виробників.

Предметом дослідження є методи, засоби та технології побудови й захисту мережевої інфраструктури на базі обладнання Cisco та MikroTik та механізми реалізації систем виявлення та запобігання вторгненням.

У процесі виконання кваліфікаційної роботи було розроблено проєкт захищеної мережної інфраструктури на базі обладнання Cisco та MikroTik з інтеграцією системи виявлення та запобігання вторгненням Suricata. Реалізовано багаторівневий захист мережі, централізований моніторинг подій безпеки, а також механізми виявлення та реагування на мережні загрози.

РОЗДІЛ 1 ТЕОРЕТИЧНІ ОСНОВИ ТА ПРИНЦИПИ ПОБУДОВИ МЕРЕЖНОЇ ІНФРАСТРУКТУРИ З ВИКОРИСТАННЯМ ТЕХНОЛОГІЙ IDS/IPS

1.1 Основи побудови та захисту комп’ютерних мереж

Комп'ютерна мережа — це складна система, що об'єднує велику кількість автономних обчислювальних засобів для спільного використання ресурсів та обміну інформацією. У контексті кібербезпеки мережа розглядається як середовище, що піддається постійним деструктивним впливам, тому її побудова вимагає глибокого розуміння як архітектурних стандартів, так і методологій захисту [1].

Ефективна побудова сучасної мережної інфраструктури базується на дотриманні ряду критичних вимог, які гарантують стабільність роботи системи при зростанні навантаження та зміні бізнес-процесів. До ключових критеріїв, що висуваються до архітектури мережі, належать:

- модульність, що передбачає поділ мережі на окремі функціональні блоки;
- масштабованість, яка забезпечує можливість поступового розширення мережі;
- відмовостійкість, що досягається за рахунок резервування каналів і пристроїв;
- продуктивність, яка залежить від пропускної здатності каналів і затримок передачі;
- керованість, що забезпечується централізованими засобами моніторингу та адміністрування.

Фундаментальною основою проектування мереж є еталонна модель взаємодії відкритих систем (Open Systems Interconnection, OSI). Вона розділяє процес передачі даних на сім логічних рівнів, кожен з яких виконує специфічні функції та взаємодіє з сусідніми рівнями через стандартизовані інтерфейси [1].

Поряд із теоретичною моделлю OSI, практична реалізація сучасних мереж базується на стеку протоколів TCP/IP (Transmission Control Protocol/Internet Protocol). На відміну від OSI, цей стек має чотирьохрівневу структуру: прикладний,

транспортний, міжмережний та рівень мережного доступу. Його успіх зумовлений прагматичністю та ефективністю механізму інкапсуляції, де кожен рівень додає свій службовий заголовок, перетворюючи дані на сегменти, пакети та кадри у процесі проходження крізь стек [1].

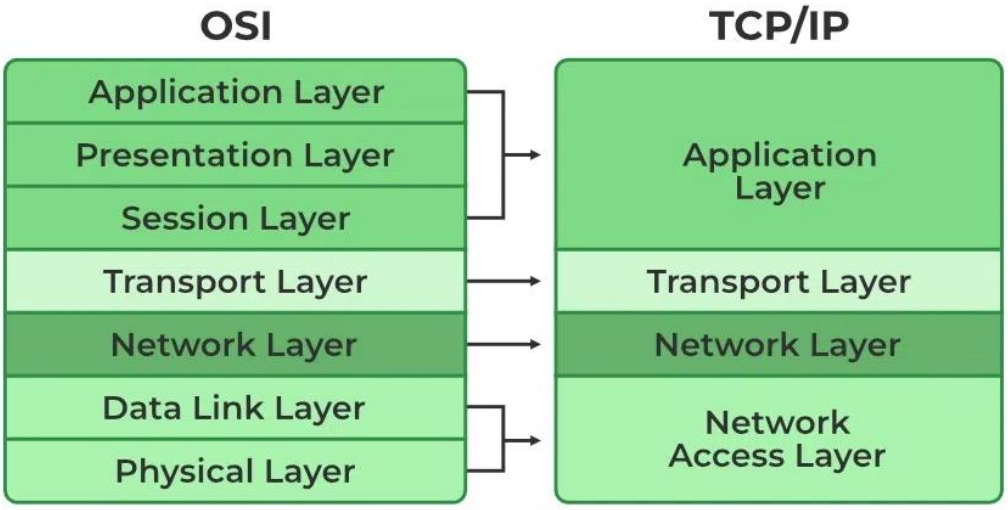


Рисунок 1.1 – Порівняння моделі OSI та стеку TCP/IP

Важливим елементом побудови мереж є її архітектура. Найбільш поширеними є топології «зірка», «шина», «кільце» та їх гібридні варіанти. У сучасних мережах переважає топологія «зірка» або ієрархічна структура, оскільки вона забезпечує кращу відмовостійкість і зручність управління [2].

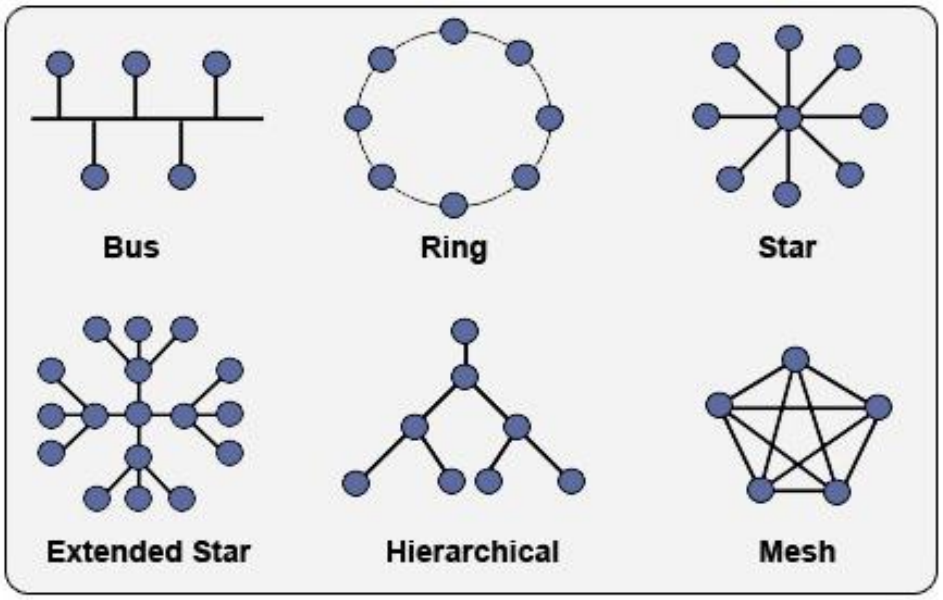


Рисунок 1.2 – Схеми різних типів топологій

Ієрархічна модель передбачає розподіл мережі на функціональні рівні. Такий підхід дозволяє спростити адміністрування, підвищити продуктивність і забезпечити гнучкість масштабування [2]. Крім того, ієрархічна модель сприяє кращій ізоляції збоїв і спрощує локалізацію проблем у мережі. Вона також дозволяє впроваджувати політики безпеки та керування трафіком на різних рівнях, що підвищує загальну ефективність роботи мережі.

Ієрархічна модель мережі включає три основні рівні:

1. Рівень ядра (Core). Забезпечує високошвидкісну передачу даних між сегментами мережі. Основною вимогою є максимальна продуктивність і мінімальні затримки.
2. Рівень розподілу (Distribution). Виконує функції маршрутизації, фільтрації трафіку, реалізації політик безпеки та контролю доступу.
3. Рівень доступу (Access). Забезпечує підключення кінцевих пристроїв і первинний контроль доступу до мережі [2].

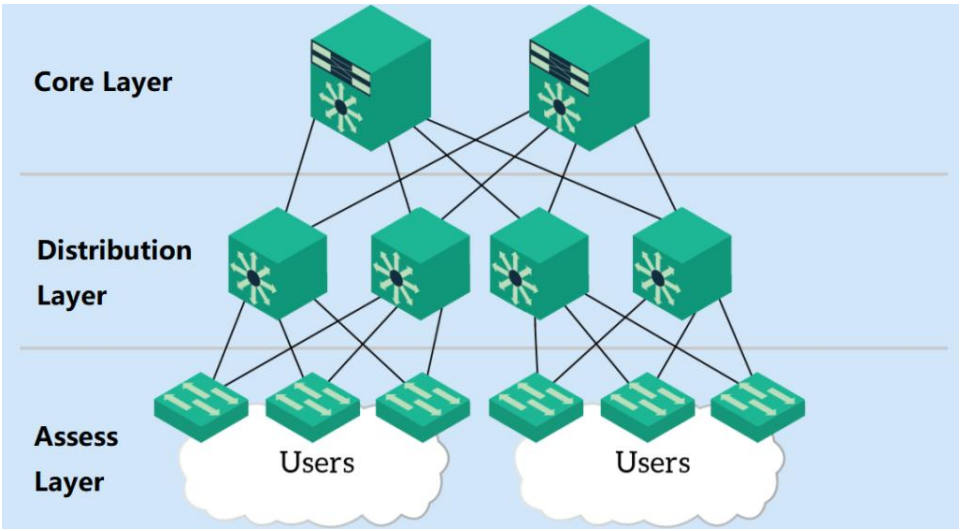


Рисунок 1.3 – Структура ієрархічної моделі

Також важливим аспектом побудови інфраструктури є класифікація мереж, що безпосередньо впливає на вибір засобів захисту [1]. Виділяють наступні типи:

- LAN (Local Area Network);
- WLAN (Wireless Local Area Network);
- MAN (Metropolitan Area Network);
- WAN (Wide Area Network).

Найменшою одиницею є локальна мережа (LAN), яка охоплює обмежену територію та характеризується високою швидкістю за повного контролю власника над середовищем. Її бездротовий аналог (WLAN) забезпечує мобільність, проте через відкритість радіоефіру вимагає посиленого шифрування. Для об'єднання ресурсів у межах міста використовуються міські мережі (MAN), тоді як глобальні мережі (WAN) охоплюють країни та континенти.



Рисунок 1.4 – Взаємозв'язок мереж різного рівня

Процес передачі інформації в мережі постійно піддається загрозам. Вони спрямовані на порушення базових властивостей інформаційної безпеки, серед яких:

- конфіденційність інформації, що полягає у захисті даних від несанкціонованого розголошення;
- цілісність даних, яка забезпечує захист інформації від несанкціонованих змін або знищення;
- доступність ресурсів, що гарантує безперебійну роботу мережних сервісів та систем.

За характером впливу на інформаційні ресурси та інфраструктури атаки прийнято класифікувати на пасивні та активні [3].

Пасивні атаки спрямовані на порушення конфіденційності. Зловмисник виступає як «прослуховувач», який перехоплює дані без їхньої зміни. Оскільки

стан системи не змінюється, такі атаки майже неможливо виявити технічними засобами моніторингу. Основними прикладами є:

- сніфінг пакета, що полягає у захопленні та перегляді вмісту пакетів даних;
- аналіз трафіку, що передбачає вивчення характеристик повідомлень для розкриття топології мережі та логіки взаємодії вузлів;
- сканування портів, яке використовується для виявлення активних мережних служб та потенційних точок входу в систему.

Активні атаки передбачають навмисну зміну мережних потоків або створення фальшивих повідомлень, що призводить до порушення цілісності або доступності системи. На відміну від пасивних, ці атаки спрямовані на деструктивну зміну стану середовища. До ключових векторів активного впливу належать:

- маскарад та спуфінг, що проявляються у спробах зловмисника видати свій пристрій за легітимний вузол шляхом підміни IP-адрес або DNS-записів;
- атаки на доступність (DoS/DDoS), спрямовані на виснаження обчислювальних ресурсів серверів або переповнення каналів зв’язку;
- модифікація повідомлень, яка передбачає перехоплення та несанкціоновану зміну вмісту пакетів під час їхнього транзиту, що порушує цілісність інформації;
- атаки повтором (Replay attacks), що полягають у захопленні порції автентифікаційних даних з її подальшою повторною відправкою для отримання несанкціонованого доступу;
- ARP-отруєння (ARP Poisoning), що використовується для маніпуляції протоколом розпізнавання адрес із метою перенаправлення трафіку на пристрій зловмисника.

Захист комп’ютерних мереж будується за принципом багаторівневості, що передбачає впровадження засобів безпеки на різних рівнях — від фізичного до прикладного. Такий підхід часто називають «глибинною обороною», оскільки він дозволяє створити кілька незалежних бар’єрів на шляху потенційних загроз [4]. Це суттєво знижує ризик компрометації системи навіть у разі обходу одного з рівнів захисту.

До основних технічних засобів захисту комп’ютерних мереж належать:

- міжмережні екрани, які здійснюють контроль і фільтрацію мережного трафіку відповідно до встановлених правил;
- системи виявлення та запобігання вторгненням, що аналізують мережну активність і дозволяють виявляти аномалії;
- криптографічні засоби захисту, які забезпечують шифрування даних під час передачі;
- віртуальні приватні мережі, що створюють захищені канали зв’язку через незахищені мережі;
- системи автентифікації та управління доступом, які контролюють права користувачів [4].

Важливим аспектом є сегментація мережі, яка дозволяє розділити її на окремі ізольовані зони. Це обмежує можливість поширення атак і підвищує контроль над доступом до критичних ресурсів. Сегментація може реалізовуватися за допомогою VLAN, підмереж та зон безпеки [2].

Окрім технічних засобів, значну роль відіграють організаційні заходи. Вони включають розробку політик безпеки, регламентів доступу, процедур реагування на інциденти та управління ризиками. Політика безпеки визначає основні правила використання мережних ресурсів і є основою для впровадження технічних механізмів захисту [3].

Також істотною складовою захисту є постійний моніторинг мережі та аналіз подій безпеки. Для цього використовуються системи журналювання, засоби централізованого збору логів і системи управління подіями безпеки (SIEM). Вони дозволяють своєчасно виявляти підозрілу активність і оперативно реагувати на інциденти, що значно підвищує рівень загальної безпеки мережі.

1.2 Технологічні рішення для побудови безпечних мереж

Технологічні рішення у сфері мережної безпеки спрямовані на забезпечення стабільної та захищеної роботи інформаційної інфраструктури підприємства. Вони дозволяють контролювати мережний трафік, розмежовувати доступ користувачів

до ресурсів, забезпечувати захищений обмін даними та своєчасно виявляти потенційні загрози. Вибір конкретних рішень залежить від масштабів мережі, вимог до безпеки та особливостей функціонування організації.

На сьогодні значного поширення набули рішення компаній Cisco та MikroTik, які пропонують широкий набір засобів для побудови мереж різного рівня складності.

1.2.1 Огляд Cisco

Компанія Cisco є одним із світових лідерів у сфері мережних технологій та інформаційної безпеки. Її рішення широко застосовуються у корпоративних мережах, дата-центрах та телекомунікаційній інфраструктурі [5]. Основною особливістю підходу Cisco є інтеграція функцій мережної взаємодії та безпеки в єдину інфраструктуру.



Рисунок 1.5 – Логотип компанії Cisco

Компанія виробляє широкий спектр мережних пристроїв, серед яких маршрутизатори, комутатори, бездротові точки доступу, міжмережні екрани та контролери бездротових мереж. Маршрутизатори Cisco використовуються для організації взаємодії між різними мережами та забезпечення маршрутизації трафіку. Комутатори застосовуються для побудови локальних мереж та підтримують технології VLAN, QoS, STP та інші механізми керування мережею.

Більшість пристроїв Cisco працюють під керуванням операційних систем Cisco IOS, IOS XE або NX-OS [6]. Дані операційні системи забезпечують підтримку великої кількості мережних протоколів та технологій, включаючи статичну та

динамічну маршрутизацію, VLAN, ACL, VPN, NAT, DHCP, IPv6 та механізми забезпечення якості обслуговування трафіку (QoS). Конфігурація пристроїв може виконуватись через командний рядок (CLI), вебінтерфейс або системи централізованого управління.

Рішення Cisco переважно орієнтовані на середні та великі корпоративні мережі, де важливими є висока відмовостійкість, централізоване управління та масштабованість мережної інфраструктури.

Для забезпечення мережної безпеки Cisco пропонує широкий спектр технологій та програмних платформ. Одним із ключових напрямів є концепція Zero Trust, яка передбачає перевірку кожного користувача та пристрою незалежно від їх розташування у мережі [6]. Такий підхід дозволяє зменшити ризик несанкціонованого доступу та підвищити рівень контролю над інформаційними ресурсами.

Важливе місце у системі безпеки Cisco займають міжмережні екрани Cisco Secure Firewall. Дані рішення забезпечують:

- фільтрацію мережного трафіку;
- контроль доступу між сегментами мережі;
- виявлення шкідливої активності;
- захист від мережних атак;
- підтримку VPN-з’єднань.

Cisco також пропонує систему централізованого контролю доступу Cisco Identity Services Engine (ISE), яка забезпечує автентифікацію користувачів і пристроїв та підтримує сегментацію мережі відповідно до політик безпеки [7].

Для побудови корпоративних мереж компанія використовує технології SD-WAN та SASE, що дозволяють організовувати безпечне з’єднання філій, хмарних сервісів і віддалених користувачів через централізовану систему управління [6]. Також Cisco реалізує засоби автоматизації та аналітики мережної безпеки. Платформи Cisco XDR та Cisco Security Cloud Control дозволяють централізовано відстежувати події безпеки, аналізувати загрози та автоматизувати реагування на інциденти [7].

1.2.2 Огляд MikroTik

Компанія MikroTik є виробником мережного обладнання та розробником операційної системи RouterOS [8]. Її рішення широко використовуються у невеликих корпоративних мережах, провайдерських інфраструктурах та навчальних середовищах. Основною перевагою обладнання MikroTik є поєднання широкого функціоналу та відносно невисокої вартості.



Рисунок 1.6 – Логотип компанії MikroTik

MikroTik виробляє маршрутизатори, комутатори, бездротові точки доступу та спеціалізовані мережні пристрої [8]. Найбільш відомими є пристрої серії RouterBOARD, які працюють під керуванням RouterOS. Такі пристрої можуть виконувати функції маршрутизації, комутації, VPN-сервера, міжмережного екрана та контролю трафіку.

Операційна система RouterOS забезпечує підтримку великої кількості мережних функцій, включаючи статичну та динамічну маршрутизацію, VLAN, VPN, QoS, DHCP, DNS, балансування навантаження, фільтрацію трафіку та керування пропускною здатністю каналів [9]. RouterOS також підтримує роботу з протоколами OSPF, BGP, RIP та MPLS. Для адміністрування обладнання MikroTik використовуються вебінтерфейс, командний рядок та спеціалізована утиліта WinBox [9].

У сфері безпеки RouterOS підтримує міжмережне екранування, NAT, ACL, VPN-технології та фільтрацію пакетів [9]. Вбудований Firewall дозволяє створювати правила контролю трафіку, обмежувати доступ до мережних сервісів та блокувати підозрілу активність. Для захисту мережі можуть використовуватись списки адрес, механізми обмеження швидкості з’єднань та правила захисту від DoS-атак.

MikroTik підтримує різні VPN-технології, серед яких:

- IPsec;
- L2TP;
- SSTP;
- OpenVPN;
- WireGuard.

Це дозволяє організовувати захищений віддалений доступ користувачів та створювати VPN-з'єднання між філіями підприємства. Особливої популярності набуває WireGuard завдяки простоті налаштування та високій продуктивності.

В RouterOS підтримуються механізми розмежування прав доступу користувачів, журналювання подій та обмеження доступу до служб адміністрування [9]. Також важливою особливістю MikroTik є гнучкість налаштування мережної інфраструктури. Адміністратор має можливість детально конфігурувати правила маршрутизації, фільтрації та контролю трафіку.

1.3 Аналіз систем виявлення та запобігання вторгненням

Одним із важливих напрямів забезпечення мережної безпеки є використання систем виявлення та запобігання вторгненням. Дані системи призначені для контролю мережного трафіку та виявлення підозрілої активності, яка може свідчити про спроби несанкціонованого доступу або мережні атаки [10].

У сучасних комп'ютерних мережах традиційних засобів захисту часто недостатньо для забезпечення повноцінного рівня безпеки. Це пов'язано з тим, що багато сучасних атак можуть маскуватися під легітимний трафік або використовувати складні механізми обходу захисту. Саме тому IDS/IPS стали важливою складовою комплексного підходу до захисту мережної інфраструктури.

Системи IDS виконують функцію моніторингу та аналізу подій безпеки. Їх основним завданням є виявлення підозрілої активності та повідомлення адміністратора про потенційну загрозу. На відміну від IDS, системи IPS не лише виявляють атаки, а й можуть автоматично блокувати небезпечний трафік або обмежувати доступ до мережних ресурсів [10].

Використання IDS/IPS дозволяє:

- виявляти мережні атаки та спроби вторгнення;
- контролювати мережний трафік у режимі реального часу;
- фіксувати події безпеки та вести журналювання;
- автоматично реагувати на підозрілу активність;
- підвищувати загальний рівень захисту мережної інфраструктури.

1.3.1 Принципи функціонування та архітектура систем IDS/IPS

Принцип роботи систем IDS/IPS полягає у постійному аналізі мережного трафіку або подій. Основними компонентами архітектури IDS/IPS є:

- сенсори або агенти, які збирають інформацію про мережний трафік чи події системи;
- модуль аналізу, що виконує перевірку та виявлення підозрілої активності;
- база сигнатур або правил виявлення атак;
- система журналювання та формування повідомлень;
- консоль централізованого управління та моніторингу.

Сенсорний рівень цієї структури відповідає за збір даних із різних сегментів мережі. Сенсори можуть працювати в режимі копіювання трафіку або в активному (inline-передача через пристрій). Обраний метод підключення визначає ступінь впливу: від спостереження за подіями без втручання в трафік до безпосередньої фільтрації сесій у реальному часі.

Процес внутрішньої обробки даних починається з етапу нормалізації та препроцесингу, що дозволяє уникнути технік обходу захисту, заснованих на фрагментації пакетів. Після підготовки дані потрапляють до аналітичного ядра. Архітектура цього ядра зазвичай поєднує декілька методів аналізу, серед яких ключовими є сигнатурне детектування та аналіз протокольних аномалій. Сигнатурний аналіз базується на порівнянні вмісту пакетів із базою даних відомих шаблонів атак, що забезпечує високу точність спрацювання при мінімальній кількості помилок. Водночас аналіз аномалій дозволяє системі будувати статистичну модель «нормальної» поведінки мережі та виявляти відхилення [11].

Завершальним рівнем архітектури є підсистема управління та реагування, яка агрегує дані від розподілених сенсорів. Сучасні IDS/IPS підтримують інтеграцію з іншими засобами безпеки, зокрема SIEM-системами, міжмережними екранами та платформами централізованого моніторингу, що дозволяє автоматизувати процес реагування на інциденти.

Ефективність IDS/IPS значною мірою залежить від правильного розташування в мережній інфраструктурі. Найчастіше такі системи встановлюються:

- за міжмережним екраном;
- безпосередньо між внутрішньою мережею та зовнішніми середовищем;
- всередині мережі для моніторингу локального трафіку та виявлення загроз у разі обходу периметра;
- безпосередньо на хостах для контролю локальної активності [11].

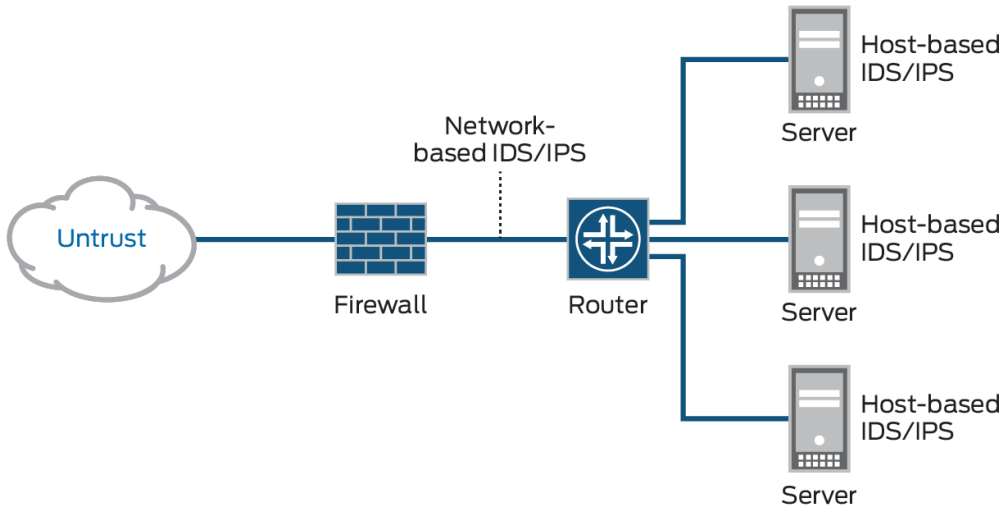


Рисунок 1.7 – Схема застосування систем виявлення та запобігання вторгненням

Існує кілька типів IDS, які класифікуються залежно від їхнього місця розташування або способу виявлення:

- мережна система виявлення вторгнень (NIDS);
- хостова система виявлення вторгнень (HIDS);
- гібридна система виявлення вторгнень;
- система виявлення вторгнень на рівні прикладних протоколів (APIDS);
- система виявлення вторгнень на рівні протоколів (PIDS).

Системи IPS можна класифікувати на такі типи залежно від їхньої спрямованості моніторингу та розгортання:

- мережна система запобігання вторгненням (NIPS);
- хостова система запобігання вторгненням (HIPS);
- бездротова система запобігання вторгненням (WIPS);
- система аналізу мережної поведінки (NBA).

Найбільш поширеними є мережні системи NIDS/NIPS, які встановлюються у ключових точках інфраструктури та аналізують мережний трафік між сегментами мережі, а також хостові системи HIDS/HIPS, що функціонують безпосередньо на окремих пристроях і контролюють локальні події, журнали безпеки та зміни системних файлів. Також використовуються гібридні системи, які поєднують функції мережного та хостового аналізу, що дозволяє забезпечити більш комплексний контроль безпеки. Окремо виділяють системи APIDS, орієнтовані на контроль прикладних протоколів, зокрема HTTP чи SQL, що дозволяє виявляти атаки на вебзастосунки та бази даних. PIDS, у свою чергу, контролюють роботу окремих мережних протоколів і виявляють відхилення від їх нормального функціонування. Бездротові WIPS призначені для захисту Wi-Fi інфраструктури. Системи аналізу мережної поведінки (NBA) аналізують поведінкові характеристики трафіку та дозволяють виявляти аномальні мережні процеси [10, 11].

Практичне використання IDS/IPS супроводжується певними технічними обмеженнями, які необхідно враховувати під час проєктування системи захисту. До основних недоліків належать:

- можливість виникнення хибнопозитивних спрацювань;
- потреба в регулярному оновленні сигнатур;
- залежність ефективності від правильності конфігурації;
- додаткове навантаження на мережну інфраструктуру.

Незважаючи на зазначені обмеження, ці технології залишаються важливим елементом захисту комп’ютерних мереж, оскільки дозволяють своєчасно виявляти інциденти безпеки та зменшувати ризики компрометації інформаційних ресурсів.

1.3.2 Порівняльний аналіз та вибір IDS/IPS

Вибір системи виявлення та запобігання вторгненням для побудови системи моніторингу мережного трафіку є важливим етапом проєктування захищеної мережної інфраструктури. У межах даного проєкту IDS/IPS повинна відповідати таким вимогам:

- підтримувати роботу в середовищі емуляції мереж;
- бути доступною для практичного використання без фінансових витрат;
- забезпечувати аналіз мережного трафіку в режимі реального часу;
- мати можливість інтеграції із системами журналювання та централізованого моніторингу.

Враховуючи визначені критерії, для подальшого аналізу було обрано три найбільш поширені рішення: Snort, Suricata та Zeek. Дані системи мають відкритий код, активно використовуються у сфері інформаційної безпеки та можуть бути розгорнуті у віртуальному середовищі.

Snort — довгожитель ринку мережних систем виявлення вторгнень. Використовує набір правил, які допомагають визначити шкідливу мережну активність, зіставляє пакети даних із цими правилами та генерує сповіщення для користувачів [12].



Рисунок 1.8 – Логотип ПЗ Snort

Основні переваги цієї системи полягають у перевіреній часом моделі, величезній базі сигнатур, простому синтаксисі правил та готовій інтеграції із SIEM-системами. Недоліками є неефективність проти атак нульового дня без регулярних

оновлень, базовий аналіз протоколів у старіших версіях, а також обмежена продуктивність на високих швидкостях через однопотокову архітектуру.

Suricata — сучасна багатопотокова система IDS/IPS, що підтримує паралельну обробку мережного трафіку та глибокий аналіз пакетів. Як і Snort, Suricata підтримує сигнатурні правила (багато синтаксису сумісний зі Snort), але додатково має вбудований декодер протоколів, підтримку потокової реконструкції, багатопотокову архітектуру та вбудовану обробку потоків [13].



Рисунок 1.9 – Логотип ПЗ Suricata

Перевагами цього рішення є висока продуктивність завдяки багатопотоковій архітектурі, глибокий аналіз протоколів, а також придатність для використання у великих мережних середовищах. Обмеження полягають у потребі в більш потужному апаратному забезпеченні при високому навантаженні, а також що управління великими наборами правил потребує додаткової уваги та адміністрування.

Zeek — система мережного моніторингу, що орієнтована на поведінковий аналіз мережного трафіку. Система не обмежується сигнатурним аналізом, а забезпечує детальне журналювання мережних подій та глибокий аналіз поведінки мережних вузлів. Платформа інтерпретує сирий пакетний трафік у структуровані логи високого рівня, диференційовані за протоколами, та підтримує запуск спеціалізованих скриптів [14].



Рисунок 1.10 – Логотип ПЗ Zeek

Переваги полягають у гнучкості й глибині аналізу протоколів, наявності потужної мови сценаріїв та високій ефективності при розслідуванні інцидентів завдяки генерації детальних метаданих для SIEM. Недоліками є великі затрати часу на написання скриптів і налаштування, а також менша кількість готових сигнатур через орієнтацію на поведінковий аналіз.

Для проведення порівняльного аналізу основні характеристики систем було наведено в таблиці:

Таблиця 1.1

Порівняння рішень

Критерій	Snort	Suricata	Zeek
Тип системи	NIDS/NIPS	NIDS/NIPS	NIDS (мережний аналізатор)
Основний підхід	Сигнатурний аналіз	Сигнатурний + аномальний аналіз	Поведінковий аналіз
Архітектура обробки	Однопотокова	Багатопотокова	Подієво-орієнтована
Продуктивність	Середня	Висока	Висока
Масштабованість	Обмежена	Висока	Висока
Формат логів	Текстові + unified2	JSON (EVE)	Розширені журнали подій
Аналіз HTTP/SSL	Базовий	Розширений	Дуже глибокий
Готові сигнатури	Дуже велика база	Велика база + Snort сумісність	Обмежено (більше сценарії)
Поріг входження	Низький	Середній	Високий
Призначення	Класичний IDS/IPS	Високопродуктивний IDS/IPS	Мережний аналіз та форензика
Інтеграція з SIEM	Обмежена	Висока	Висока
Відображення подій	Базове	Розширене (JSON для SIEM)	Детальне (логічні події)

З огляду на вимоги даного проєкту, найбільш доцільним є вибір Suricata. На відміну від Snort, який історично орієнтований на однопотокову обробку, Suricata має нативну багатопотокову архітектуру, що дозволяє ефективно використовувати ресурси процесора та уникати втрати пакетів. У порівнянні із системою Zeek, яка є ресурсомісткою платформою переважно для поведінкового аналізу й кіберрозвідки, Suricata пропонує вищу швидкість сигнатурного захисту та повноцінну роботу в активному режимі запобігання атакам. Таким чином, Suricata стала оптимальним вибором, поєднавши високу продуктивність та простоту інтеграції із системами централізованого моніторингу.

1.4 Постановка технічного завдання

Метою проєкту є розробка та моделювання захищеної мережної інфраструктури з інтеграцією системи виявлення та запобігання вторгненням, яка забезпечує моніторинг мережного трафіку та підвищення рівня інформаційної безпеки.

Необхідно спроєктувати мережну архітектуру. Система повинна підтримувати маршрутизацію між сегментами та забезпечувати коректну взаємодію мережних пристроїв. Особлива увага приділятиметься безпеці інфраструктури: планується впровадити систему контролю доступу користувачів та централізований збір логів із мережних пристроїв.

Окремим завданням є впровадження NIDS/NIPS-рішення Suricata на критичних вузлах для аналізу мережного трафіку в режимі реального часу, виявлення підозрілої активності та формування подій безпеки з метою їх подальшої обробки та візуалізації.

Заплановано тестування працездатності мережі та оцінка стабільності роботи системи в умовах навантаження. Також передбачається діагностика роботи IDS/IPS шляхом перевірки її реакції на типові мережні аномалії і генерації відповідних сповіщень про виявлені події безпеки.

Практична реалізація буде здійснюватися на базі спеціалізованої платформи емуляції з використанням віртуальних образів мережного обладнання.

Висновки до першого розділу

У першому розділі розглянуто теоретичні основи побудови комп'ютерних мереж та принципи їх захисту із застосуванням IDS/IPS. Проаналізовано архітектуру мережі, можливі загрози та базові підходи до забезпечення інформаційної безпеки.

Проведено огляд технологічних рішень Cisco та MikroTik. Встановлено, що Cisco забезпечує високу надійність і розширені функції корпоративного рівня, тоді як MikroTik відзначається простотою конфігурації та гнучкістю. Їх поєднання є доцільним для побудови ефективної мережної інфраструктури.

Також проведено аналіз систем IDS/IPS, розглянуто їх принципи роботи, класифікацію та методи виявлення загроз. У результаті порівняння Snort, Suricata та Zeek встановлено, що Snort є простим та добре відомим рішенням із великою базою сигнатур, Zeek забезпечує глибокий поведінковий аналіз, але є складнішим у налаштуванні, тоді як Suricata поєднує високу продуктивність та багатопотокову обробку, що робить її найбільш доцільною для використання в даному проєкті.

РОЗДІЛ 2 ПРОЄКТУВАННЯ МЕРЕЖНОЇ ІНФРАКСТУКТУРИ

2.1 Розробка топології та адресації комп'ютерної мережі

Важливим етапом проєктування захищеної інфраструктури є побудова раціональної топології, яка забезпечує не лише стабільний обмін даними, але й створює умови для ефективного впровадження засобів контролю та моніторингу. Для реалізації поставлених завдань обрано ієрархічну модель побудови мережі, що дозволяє чітко розмежувати зони відповідальності та сегментувати трафік і складається з рівнів ядра, розподілу та доступу. В ході роботи над проєктом було складено таку топологію:

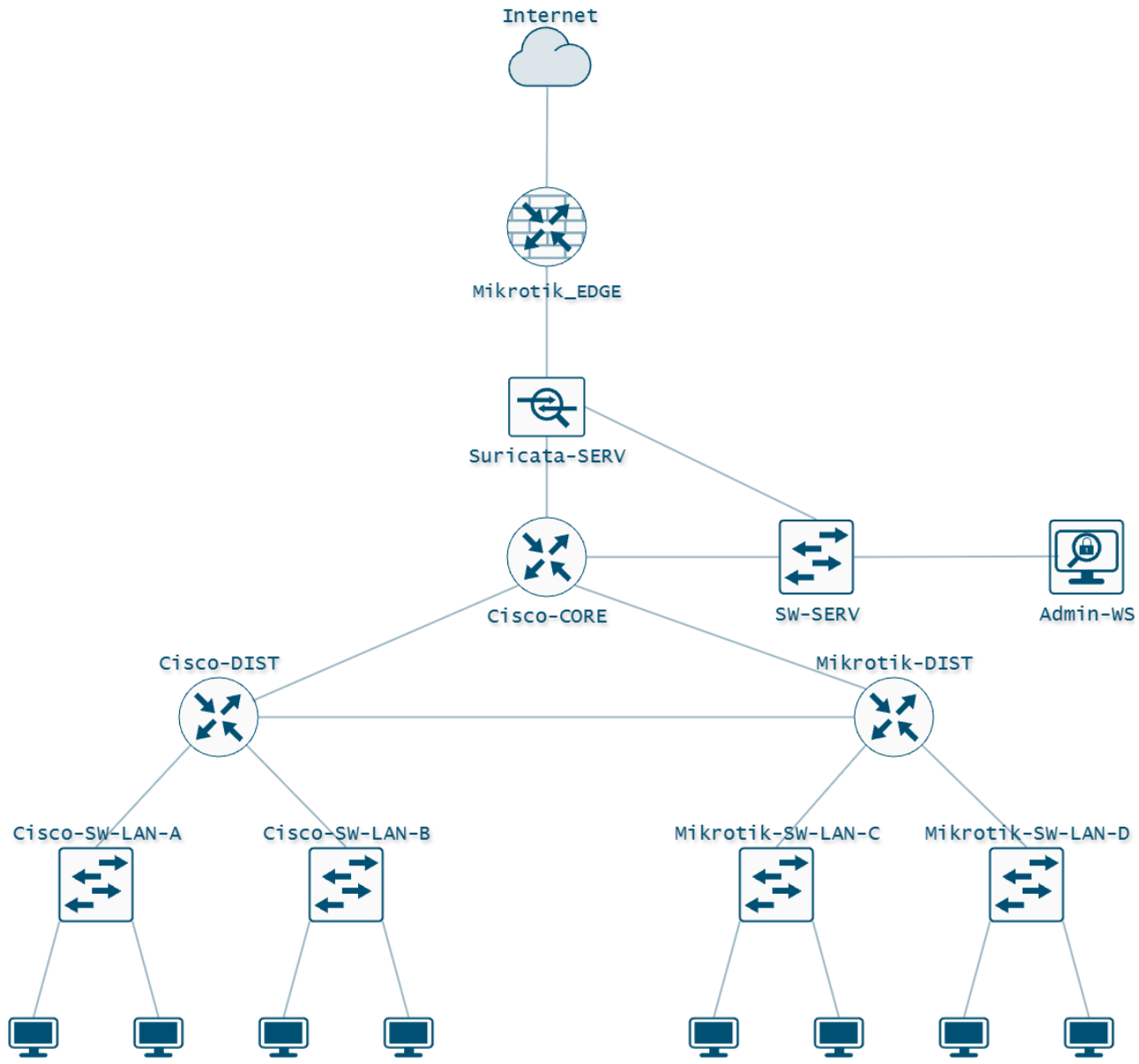


Рисунок 2.1 – Топологія мережі

На периметрі мережі розташований об’єкт «Internet», який у межах даного проєкту є імітацією зовнішнього середовища для відпрацювання сценаріїв взаємодії з глобальною мережею. До цього сегмента підключається маршрутизатор Mikrotik_EDGE, який виступає межовим пристроєм всієї інфраструктури. Він відповідає за трансляцію мережних адрес (NAT), первинну фільтрацію вхідного трафіку та забезпечення логічного зв’язку зі зовнішньою мережею.

Сервер Suricata розміщений між периметровим маршрутизатором та ядром мережі та працює в inline-режимі. Таке розташування дозволяє перехоплювати та аналізувати весь вхідний та вихідний трафік інфраструктури в режимі реального часу.

Центральним елементом внутрішньої інфраструктури на рівні ядра виступає комутатор Cisco-CORE, який виконує роль головної точки агрегації для розподільчих рівнів. До цього маршрутизатора підключається окремий серверний сегмент через комутатор SW-SERV для керування пристроями мережі. Тут розташована робоча станція Admin-WS. Також Suricata-SERV має додаткове з'єднання з цим комутатором, яке призначене для адміністрування та моніторингу стану серверу.

На рівні розподілу задіяно пристрої Cisco-DIST та Mikrotik-DIST, які функціонують як DHCP-сервери у своїх сегментах, забезпечуючи динамічне виділення IP-адрес для кінцевих користувачів. Також вони агрегують трафік від нижчих рівнів та здійснюють обмін маршрутами з ядром мережі для оптимізації передачі даних.

На рівні доступу розгорнуто комутатори Cisco-SW-LAN-A, Cisco-SW-LAN-B, а також MikroTik-SW-LAN-C та MikroTik-SW-LAN-D. Ці пристрої забезпечують безпосереднє фізичне підключення кінцевих користувачів до загальної мережної структури.

Обмін маршрутною інформацією всередині інфраструктури забезпечує протокол OSPF. Для захисту від підміни маршрутних даних та запобігання несанкціонованому втручанню, усі OSPF-процеси використовують криптографічну автентифікацію на базі алгоритму SHA-256.

Побудова раціональної схеми адресації є базовим етапом проєктування, оскільки вона визначає логіку взаємодії всіх її елементів. Чітке розмежування адресних діапазонів дозволяє ізолювати критичні сегменти інфраструктури та забезпечити надійний фундамент для впровадження майбутніх налаштувань. Для організації зв'язку між маршрутизаторами всередині інфракструктури розподіл адрес проводився таким чином, щоб використати мінімальну кількість IP-адрес.

Таблиця 2.1

Адресація підмереж між маршрутизаторами

Пристрій	Мережа	Маска	Префікс
Mikrotik_EDGE ↔ Cisco-CORE	10.0.0.0	255.255.255.252	/30
Cisco-CORE ↔ Cisco-DIST	10.0.0.4	255.255.255.252	/30
Cisco-CORE ↔ Mikrotik-DIST	10.0.0.8	255.255.255.252	/30
Mikrotik-DIST ↔ Cisco-DIST	10.0.0.12	255.255.255.252	/30

Для організації зв’язку серверної підмережі, де передбачено функціонування 4 вузлів, було обрано статичну адресацію. Оптимальним рішенням для такого сегмента є використання мережі з префіксом /29, що надає 6 доступних IP-адрес для хостів.

Таблиця 2.2

Адресація підмережі SERV

Мережа/ Пристрій	IP-адреса	Шлюз	Маска	Префікс
Підмережа SERV	192.168.100.0	-	255.255.255.248	/29
Маршрутизатор Cisco-CORE	192.168.100.1	-	255.255.255.248	/29
SW-SERV	192.168.100.2	192.168.100.1	255.255.255.248	/29
Admin-WS	192.168.100.3	192.168.100.1	255.255.255.248	/29
Suricata-SERV	192.168.100.4	192.168.100.1	255.255.255.248	/29

Для забезпечення зв’язку в сегментах рівня доступу (підмережі A, B, C, D), де передбачено підключення 160 робочих станцій у кожній з підмереж, було обрано використання адресного простору з префіксом /24, що надає 254 доступних IP-адрес. Для управління мережним обладнанням перші адреси діапазону зарезервовані за комутаторами доступу.

Таблиця 2.3

Адресація підмереж A та B

	Підмережа A	Підмережа B
Адреса мережі	192.168.10.0	192.168.20.0
Маска мережі	255.255.255.0	255.255.255.0
Шлюз	192.168.10.1	192.168.20.1
Діапазон адрес комутаторів	192.168.10.2 – 192.168.10.10	192.168.20.2 – 192.168.20.10
Діапазон адрес робочих станцій	192.168.10.11 – 192.168.10.170	192.168.20.11 – 192.168.20.170
Діапазон вільних адрес	192.168.10.171-192.168.10.254	192.168.20.171 – 192.168.20.254

Таблиця 2.4

Адресація підмереж C та D

	Підмережа C	Підмережа D
Адреса мережі	192.168.30.0	192.168.40.0
Маска мережі	255.255.255.0	255.255.255.0
Шлюз	192.168.30.1	192.168.40.1
Діапазон адрес комутаторів	192.168.30.2 – 192.168.30.10	192.168.40.2 – 192.168.40.10
Діапазон адрес робочих станцій	192.168.30.11 – 192.168.30.170	192.168.40.11 – 192.168.40.170
Діапазон вільних адрес	192.168.30.171 – 192.168.30.254	192.168.40.171 – 192.168.40.254

Окремо, для організації з'єднання між зовнішнім середовищем та маршрутизатором Mikrotik_EDGE було виділено підмережу 203.0.113.0/30.

2.2 Інтеграція системи виявлення вторгнень Suricata

Suricata є багатопотоковою системою аналізу мережного трафіку, архітектура якої побудована за модульним принципом. Система виконує нормалізацію мережного трафіку та приведення різнорідних потоків даних до уніфікованого формату для подальшого аналізу. Результати аналізу формуються у вигляді структурованих подій, що містять детальну інформацію про мережну активність. Отримані дані можуть використовуватися як для оперативного моніторингу, так і для подальшого розслідування інцидентів інформаційної безпеки.

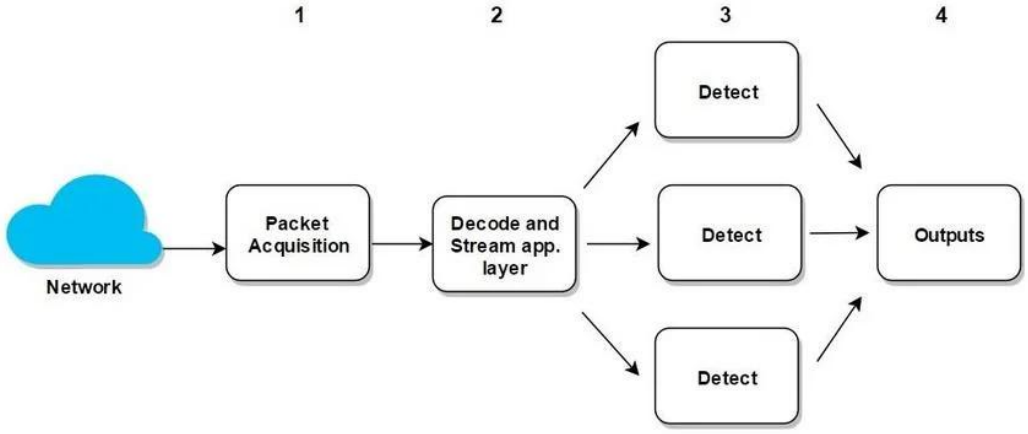


Рисунок 2.2 – Архітектура Suricata

Першим етапом роботи системи є отримання мережних пакетів із мережного інтерфейсу. Для цього можуть використовуватися різні механізми взаємодії з операційною системою, серед яких:

- af-packet є вбудованим механізмом ядра linux для захоплення та передачі мережних пакетів на рівні канального шару;
- pf_ring являє собою високопродуктивний фреймворк захоплення пакетів, розроблений компанією ntop, що забезпечує зменшення навантаження на процесор та підвищення швидкості обробки трафіку порівняно зі стандартними механізмами операційної системи;
- dpdk (data plane development kit) є набором бібліотек і драйверів для високошвидкісної обробки пакетів у просторі користувача, що дозволяє досягати високої продуктивності шляхом обходу частини мережного стеку операційної системи;
- netmap є програмним інтерфейсом для швидкого обміну пакетами між мережним адаптером і прикладними програмами, що забезпечує низькі затримки та високу швидкість обробки трафіку;
- nfqueue є механізмом ядра linux, який дозволяє передавати пакети з мережного стеку операційної системи до простору користувача для подальшої обробки або прийняття рішення щодо їх блокування [15].

Після отримання пакета виконується його декодування. На даному етапі аналізуються заголовки мережних протоколів, визначаються IP-адреси, номери

портів, використовувані протоколи та інші службові параметри. Отримані дані використовуються для подальшої класифікації мережного трафіку.

На відміну від аналізу окремих пакетів, Suricata працює з поняттям мережного потоку (flow). Вони відіграють велику роль у тому, як Suricata організовує дані всередині системи. Потік дещо схожий на з'єднання, за винятком того, що потік є більш загальним поняттям. Усі пакети, які мають однаковий кортеж (Tuple: протокол, IP-адреса джерела, IP-адреса призначення, порт джерела, порт призначення), належать до одного й того самого потоку [15].

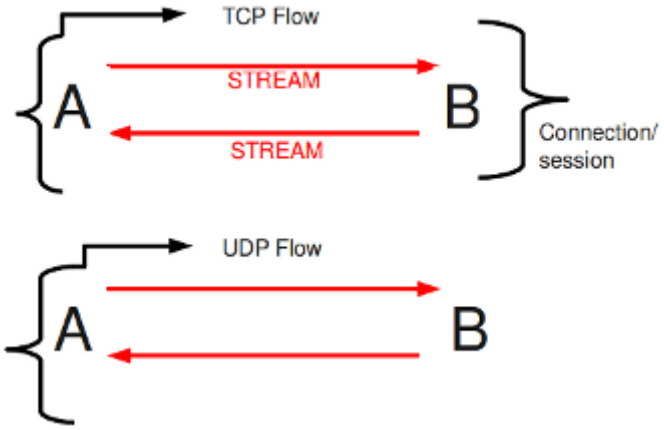


Рисунок 2.3 – Приклад потоку (flow)

Значна частина сучасних мережних атак може бути виявлена лише після аналізу повної сесії обміну даними. Для цього Suricata використовує механізм Stream Reassembly. Під час роботи система виконує збір TCP-сегментів та відновлює початковий потік даних. Завдяки цьому аналізуються не окремі пакети, а фактичний вміст мережного сеансу.

Для кожного потоку Suricata намагатиметься розпізнати протокол прикладного рівня. Визначення протоколу (protocol detection) виконується для потоків TCP та UDP та запускається незалежно для обох напрямків потоку. Потік може змінювати свій протокол прикладного рівня протягом свого існування. У випадку перехоплення потоку з середини (midstream), визначення протоколу може змінити напрямок потоку на протилежний [15]. Підтримуються протоколи, такі як:

- веб-протоколи: http (http1, http2), websocket;
- файлові та мережні служби: ftp, smb, nfs, tftp;

- криптографічні та захищені протоколи: tls (ssl), ssh, ike, krb5;
- поштові протоколи: smtp, imap, pop3;
- служби імен та мережні сервіси: dns, dhcp, dcerpc, snmp, ntp;
- промислові та спеціалізовані протоколи: modbus, dnp3, enip;
- віддалений доступ та керування: rfb, rdp, sip;
- додаткові протоколи: bittorrent-dht.

Suricata виконує визначення прикладного протоколу за кількома послідовними методами. На першому етапі застосовується механізм багатошаблонного зіставлення (multi-pattern matching), який не залежить від мережних портів і дозволяє швидко ідентифікувати протокол за характерними ознаками. Додатково використовується пробний (probing) парсер, який виконує аналіз вмісту пакета за допомогою спеціалізованого коду та працює лише для визначених портів, забезпечуючи більш точну ідентифікацію протоколу. Результатом роботи probing-парсера може бути підтвердження визначеного протоколу, відхилення або необхідність отримання додаткових даних для прийняття рішення. У деяких випадках застосовується механізм очікуваних потоків (expectation), який використовується для коректної обробки специфічних з’єднань, коли тип протоколу визначається на основі контексту мережного обміну.

Основним компонентом системи є рушій виявлення (Detection Engine), який виконує аналіз мережного трафіку на основі заданого набору правил. Detection Engine працює в межах вже сформованих мережних потоків і отримує підготовлені дані від підсистем декодування протоколів та обробки прикладного рівня.

Для проведення перевірки певних даних потоку рушій виявлення має задіяти модуль потоків для отримання даних. Робити це для кожної порції даних, що піддається аналізу, може бути ресурсомістко та ненадійно, тому модуль запитує дані частинами (чанками). Важливо зазначити, що вікно перевірки може бути обмежене певними особливими умовами, такими як досягнення максимальної глибини потоку (stream depth), досягнення кінця потоку тощо. Модуль потоків повинен відстежувати точку, до якої перевірка вже виконана. Це допомагає модулю знати, які дані вже були оброблені і можуть бути висунуті за межі вікна [15].

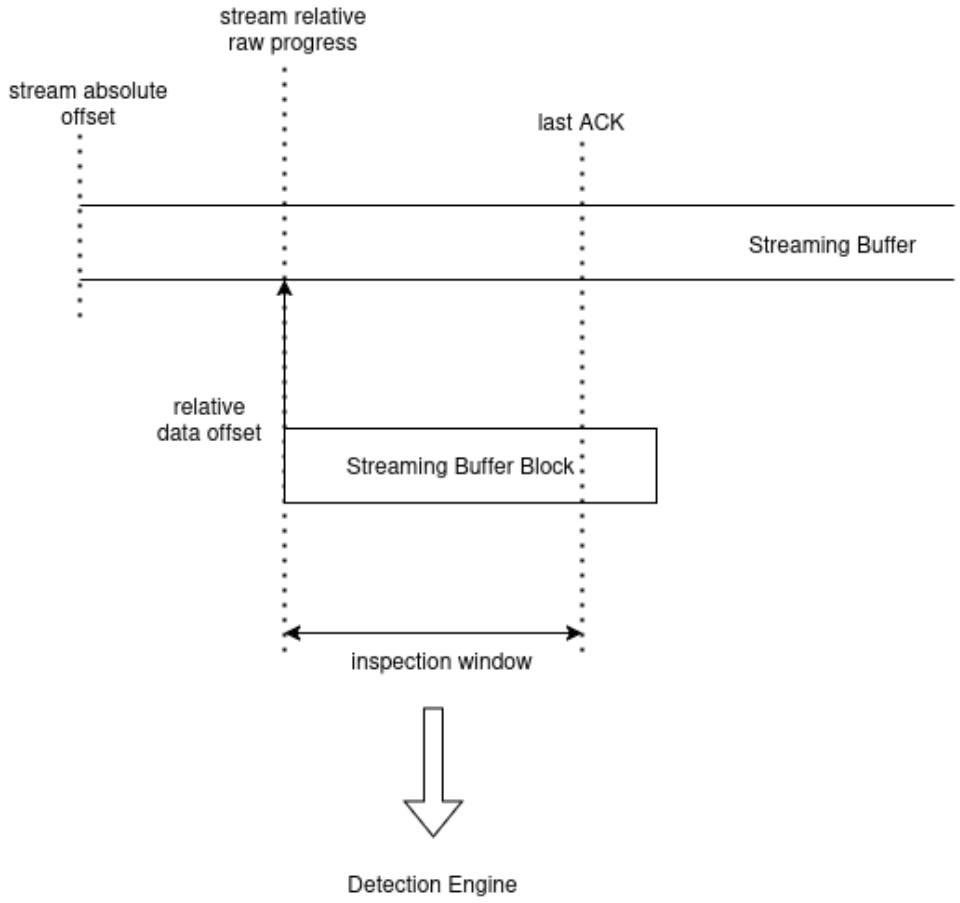


Рисунок 2.4 – Приклад схеми відстеження перевірки

Архітектурно рушій виявлення використовує попередню оптимізацію правил, що дозволяє групувати сигнатури за ознаками протоколів, портів та типів трафіку. Завдяки цьому система не виконує повну перевірку всіх правил для кожного пакета, а застосовує лише релевантні підмножини, що значно підвищує продуктивність.

У Detection Engine застосовуються механізми оптимізації та контекстного аналізу правил, зокрема fast pattern, sticky buffers та flowbits. Механізм fast pattern використовується для визначення ключового фрагмента сигнатури, який застосовується для попередньої фільтрації правил. Sticky buffers дозволяють виконувати перевірку окремих логічних частин прикладних протоколів, а не всього пакета. Flowbits забезпечують збереження стану між правилами в межах одного мережного потоку, що дає можливість реалізовувати багатокрокову логіку.

Правила Suricata є основним механізмом визначення умов виявлення мережних загроз. Вони складаються із дії, заголовка та набору опцій. Дія визначає поведінку системи у випадку спрацювання:

- alert: згенерувати сповіщення;
- pass: припинити подальшу перевірку цього пакета;
- drop: скинути пакет і згенерувати сповіщення;
- reject: надіслати помилку RST (для TCP) або ICMP Unreachable (для UDP) відправнику пакета, який збігся з правилом;
- rejectsrc: діє аналогічно до звичайного reject;
- rejectdst: надіслати пакет помилки RST/ICMP отримувачу пакета, який збігся з правилом;
- rejectboth: надіслати пакети помилки RST/ICMP обом сторонам взаємодії (і відправнику, і отримувачу) [15].

Заголовок визначає параметри мережного трафіку, а опції задають критерії спрацювання. Кожне правило має унікальний ідентифікатор SID та номер ревізії, що використовується для контролю його змін. Правила можуть бути як сигнатурними, так і поведінковими залежно від логіки виявлення. Також використовуються готові набори правил, що містять попередньо визначені сигнатури для виявлення відомих типів мережних загроз.

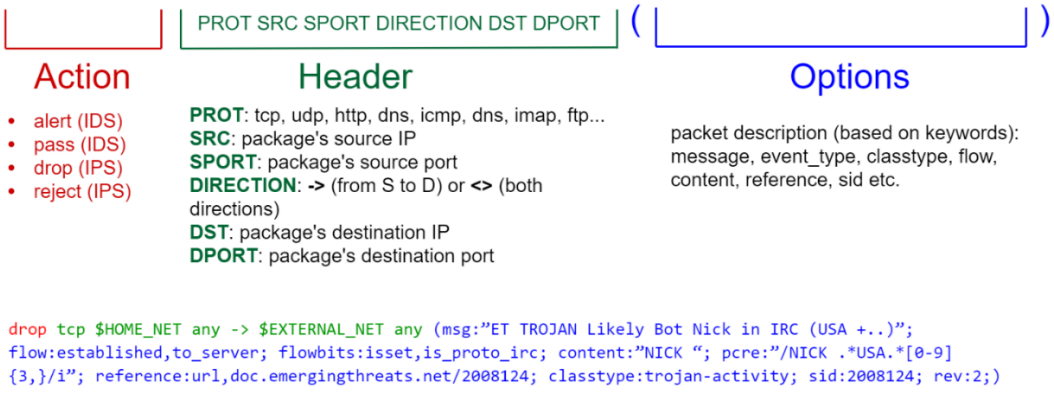


Рисунок 2.5 – Будова та приклад правила

Після завершення аналізу система формує події безпеки. Основним форматом журналювання є Eve JSON, який використовує структуроване представлення даних.

Suricata може генерувати різні типи логів та записів:

- alert (спрацювання сигнатур);
- flow (інформація про мережні потоки);
- http (дані HTTP-сеансів);
- dns (DNS-запити та відповіді);
- tls (інформація про TLS-з'єднання);
- fileinfo (метадані переданих файлів);
- stats (статистика роботи системи);
- anomaly (виявлені аномалії протоколів).


```
{
  "timestamp": "2017-04-07T22:24:37.251547+0100",
  "flow_id": 586497171462735,
  "pcap_cnt": 53381,
  "event_type": "alert",
  "src_ip": "192.168.2.14",
  "src_port": 50096,
  "dest_ip": "209.53.113.5",
  "dest_port": 80,
  "proto": "TCP",
  "metadata": {
    "flowbits": [
      "http.dottedquadhost"
    ]
  },
  "tx_id": 4,
  "alert": {
    "action": "allowed",
    "gid": 1,
    "signature_id": 2018358,
    "rev": 10,
    "signature": "ET HUNTING GENERIC SUSPICIOUS POST to Dotted Quad with Fake Browser 1",
    "category": "Potentially Bad Traffic",
    "severity": 2
  },
  "app_proto": "http"
}
```


Рисунок 2.6 – Приклад формату Eve JSON

Зазначені типи логів є основою для подальшої обробки та аналізу мережних подій у системах моніторингу. Вони дозволяють виконувати кореляцію інцидентів, аналіз мережної активності та формування історії подій інформаційної безпеки. Структурований формат Eve JSON забезпечує зручну інтеграцію з зовнішніми засобами зберігання та візуалізації даних.

2.2.1 Формування технічних вимог та конфігурації

Перед безпосереднім впровадженням системи необхідно провести аналіз вимог. Як професійне рішення класу Enterprise, дана система вирізняється широкою кросплатформеністю, демонструючи стабільну роботу в середовищах Linux, FreeBSD, OpenBSD, macOS та Windows. Проте варто зауважити, що найбільш повний інструментарій реалізований саме в дистрибутивах на базі ядра Linux, тому для даного проєкту було обрано Ubuntu Server LTS. Це рішення обумовлене наявністю найбільш актуальних пакетів у репозиторіях, що гарантує підтримку сучасних протоколів та алгоритмів шифрування без необхідності ручної компіляції ПЗ з вихідних кодів.

Ефективність аналізу трафіку в Suricata базується на багатопотоковій обробці, що дозволяє паралельно розподіляти навантаження на ядра CPU для лінійного зростання пропускної здатності. Оптимальна продуктивність досягається на багатоядерних процесорах за наявності щонайменше 8 ГБ RAM, необхідних для завантаження масивів сигнатур та підтримки буферів перехоплення. На апаратному рівні критичною є підтримка мережними інтерфейсами режиму "promiscuous mode" та технологій багаточерговості (RSS/Multi-queue), що забезпечує коректний розподіл апаратних переривань. Зазначені параметри є галузевим стандартом для розгортання систем у реальних промислових мережах із високою інтенсивністю трафіку.

З огляду на обмежені ресурси основної робочої станції та необхідність одночасного запуску кількох вузлів, для віртуальної машини виділено 2 ГБ оперативної пам’яті. Хоча цей обсяг нижчий за стандартні вимоги, його цілком достатньо для емуляції, оскільки обсяги імітованого трафіку в межах проєкту є значно меншими за навантаження в реальних промислових мережах.

Після уточнення операційно системи та апаратних параметрів віртуальної машини, наступним етапом є вибір режиму перехоплення та обробки пакетів. Було прийнято рішення про використання Suricata в режимі IPS із застосуванням механізму af_packet. Використання цього підходу забезпечує стабільність системи та дозволяє блокувати загрози без залучення сторонніх утиліт на кшталт iptables.

Завдяки механізмам ядра рішення демонструє високу відмовостійкість, зберігаючи цілісність мережного стеку навіть у разі збоїв.

Для побудови ефективної системи захисту необхідно визначити перелік загроз, які підлягають блокуванню на периметрі мережі. Основна увага приділяється виявленню експлуатації вразливостей, підозрілих мережних з'єднань, шкідливої активності в веб-трафіку, а також використанню відомих індикаторів компрометації. Окремо враховується необхідність моніторингу DNS-активності, TLS-з'єднань та передачі потенційно небезпечних файлів. З огляду на необхідність високої точності, найоптимальнішим варіантом є комбінований набір сигнатур із перевірених джерел, а саме:

- Emerging Threats (ET) Open;
- Abuse.ch SSL Blacklist (SSLS);
- Abuse.ch URLhaus;
- Positive Technologies (PT Research).

Кожен із обраних наборів відіграє специфічну роль у загальній архітектурі безпеки. При цьому логіка реагування розділяється залежно від критичності загрози: частина правил налаштовується на блокування для нейтралізації явних атак, тоді як інша частина функціонує в режимі сповіщення для збору аналітики без ризику порушити роботу легітимних сервісів.

Основою виступає Emerging Threats (ET) Open — один із найбільш динамічних відкритих наборів сигнатур, що фокусується на виявленні активності ботнетів, експлуатації вразливостей та проведенні брутфорс-атак. Його використання дозволяє закрити базові потреби моніторингу периметра від масових загроз. Для протидії більш специфічним методам компрометації залучаються ресурси платформи Abuse.ch. Зокрема, база SSL Blacklist (SSLS) дозволяє системі розпізнавати шкідливі TLS-сертифікати, що використовуються для зв'язку з командними серверами (C2), без необхідності дешифрації трафіку. Паралельно з цим, URLhaus забезпечує реагування на спроби взаємодії з хостингами, які поширюють шкідливе програмне забезпечення. Завершальним елементом захисту є набір правил від Positive Technologies (PT Research). Ці сигнатури орієнтовані на

виявлення складних цільових атак та новітніх експлойтів, які часто залишаються поза увагою стандартних баз.

Наступним кроком стало встановлення критеріїв до інформативності записів, що накопичуються. Від системи вимагається фіксація повного контексту інцидентів, включаючи корисне навантаження пакетів, метадані прикладного рівня, параметри DNS-запитів, деталі TLS-сесій та вміст HTTP-тіла. У межах моніторингу файлів система повинна забезпечувати розпізнавання типів даних та генерацію їхніх хеш-сум за алгоритмами MD5, SHA-1 і SHA-256, що дозволить оперативно ідентифікувати шкідливі об’єкти. Крім того, система має бути здатна логувати усі відкинуті пакети із фіксацією фінального вердикту фільтрації.

2.2.2 Розробка системи візуалізації

У процесі роботи система Suricata формує значну кількість журналів, аналіз яких у текстовому вигляді є малоефективним, особливо при тривалому моніторингу мережі. Тому до системи візуалізації було сформовано такі основні вимоги:

- централізований збір журналів безпеки;
- можливість роботи з журналами Suricata у форматі Eve JSON;
- пошук та фільтрація подій за різними параметрами;
- відображення статистичних даних у графічному вигляді;
- можливість роботи в режимі реального часу.

Для реалізації зазначених вимог було розглянуто декілька можливих варіантів. Зокрема, розглядалася можливість використання локальних консольних інструментів, таких як Evebox або jq, які демонструють високу швидкість роботи при мінімальних витратах ресурсів, проте не здатні забезпечити глибокий аналіз чи побудову складних кореляційних зв'язків між подіями за тривалі проміжки часу. Також було проаналізовано професійні SIEM-системи, на кшталт Wazuh або Graylog. Попри потужні інструменти автоматизації, ці платформи орієнтовані на комплексне управління безпекою, що створює надмірну складність для конкретного завдання візуалізації та потребує комплекного налаштування правил

кореляції. Як підсумок, для реалізації системи візуалізації було обрано стек Elastic Stack, що складається з Elasticsearch, Filebeat та Kibana.

Filebeat є легковаговим агентом збору журналів, який належить до сімейства Beats. Основним його призначенням є моніторинг файлів журналів та автоматична передача нових записів до системи централізованого зберігання даних [16].



Рисунок 2.7 – Логотип Beats

У межах даного проєкту Filebeat використовується для збору інформації з журналів формату Eve JSON з Suricata. Важливою перевагою є наявність готового модуля інтеграції, який забезпечує автоматичне розпізнавання структури журналів та їх нормалізацію відповідно до стандарту Elastic Common Schema. Крім того, агент характеризується низьким споживанням процесорних ресурсів та оперативної пам'яті.

Elasticsearch — це розподілена пошукова та аналітична платформа з відкритим вихідним кодом. Як платформа пошуку, вона зберігає структуровані, неструктуровані та векторні дані в режимі реального часу, забезпечуючи швидкий гібридний і векторний пошук, роботу аналітики спостережуваності та безпеки, а також функціонування високопродуктивних, точних і релевантних додатків на базі ШІ [17].



Рисунок 2.8 – Логотип Elasticsearch

Elasticsearch є центральним компонентом Elastic Stack та виконує функції сховища даних. Після отримання журналів від Filebeat усі події індексуються в Elasticsearch та стають доступними для подальшого аналізу. Адміністратор може виконувати пошук за IP-адресами, часовими інтервалами, типами подій, рівнем критичності або іншими параметрами.

Kibana є веборієнтованою платформою для візуалізації та аналізу даних, що зберігаються в Elasticsearch. Саме цей компонент забезпечує взаємодію користувача із системою моніторингу через графічний інтерфейс [18].



Рисунок 2.9 – Логотип Kibana

За допомогою Kibana адміністратор може здійснювати пошук подій, переглядати журнали безпеки, аналізувати статистичні дані та створювати інформаційні панелі для моніторингу стану мережі. Візуалізація може виконуватися у вигляді таблиць, гістограм, часових графіків, кругових діаграм та інших елементів представлення інформації.

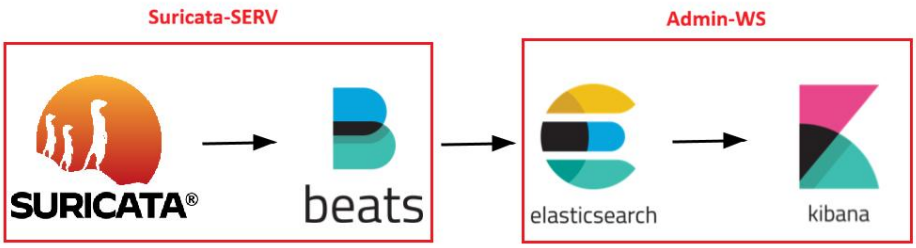


Рисунок 2.10 – Схема взаємодії компонентів системи

У розробленій системі компоненти Elastic Stack розподілені між двома віртуальними машинами. На сервері Suricata-SERV розгортається система Suricata та агент Filebeat. Сформовані журнали подій зчитуються Filebeat і передаються мережею до сервера Admin-WS. На робочій станції адміністратора встановлюються компоненти Elasticsearch та Kibana.

2.3 Супутні засоби захисту мережі

Окрім впровадження системи виявлення та запобігання вторгнень, проєктом передбачено комплекс додаткових заходів, спрямованих на захист мережної інфраструктури.

По-перше, використання FreeRADIUS. Це ПЗ є відкритою реалізацією протоколу RADIUS (Remote Authentication Dial-In User Service) та широко застосовується для контролю доступу користувачів і мережних пристроїв до інформаційних систем [19]. Система забезпечує перевірку облікових даних, визначення рівнів доступу користувачів, а також облік подій автентифікації (Authentication, Authorization, Accounting). Сервіс FreeRADIUS встановлений на машині Admin-WS, яка виконує роль централізованої робочої станції для моніторингу й керування елементами інфраструктури.



Рисунок 2.11 – Логотип FreeRADIUS

Планується створення двох основних облікових записів: netadmin та netviewer. Користувач netadmin має розширені адміністративні права для керування мережним обладнанням та його конфігурацією. Натомість netviewer використовується для перегляду стану мережі та моніторингу без можливості

внесення змін до налаштувань пристроїв. Щоб забезпечити резервний метод доступу в разі недоступності сервера RADIUS буде створений локальний користувач на пристроях.

По-друге, для забезпечення централізованого моніторингу та аналізу подій журнали FreeRADIUS, а також мережного обладнання передаватимуться до Elastic Stack за допомогою Filebeat. Для журналів FreeRADIUS використовуватиметься конвеєр обробки даних (ingest pipeline), який забезпечуватиме парсинг полів, нормалізацію даних та приведення записів до уніфікованого формату для подальшого аналізу й візуалізації в Kibana. Для повідомлень MikroTik також застосовуватиметься «ingest pipeline», конфігурацію якого взято з офіційної документації виробника [9]. Обробка системних журналів пристроїв Cisco здійснюватиметься за допомогою модуля «cisco-ios» для Filebeat, який забезпечує автоматичне розпізнавання структури повідомлень, виділення ключових полів та їх нормалізацію.

По-третє, на прикордонному рівні буде здійснюватися фільтрація мережного трафіку для обмеження небажаних з’єднань і контролю доступу до внутрішніх ресурсів. На рівні центрального маршрутизатора будуть впроваджені списки контролю доступу для суворого обмеження трафіку до серверної зони. На рівні комутаційного обладнання планується реалізація механізмів контролю підключення кінцевих пристроїв, захисту від використання неавторизованого мережного обладнання та запобігання підміні мережних адрес. Додатково, підключення по SSH та Winbox буде дозволено виключно з адміністративного сегмента мережі, що дозволить зменшити ризик несанкціонованого доступу до конфігурації обладнання та підвищити загальний рівень безпеки мережної інфраструктури.

2.4 Вибір мережного обладнання та ПЗ

Моделювання мережної топології буде реалізовано у середовищі GNS3, яке, на відміну від класичних симуляторів, забезпечує повноцінну емуляцію реальних операційних систем мережних пристроїв [20]. Запуск базової віртуальної машини

GNS3 VM здійснено на платформі гіпервізора VMware Workstation, що забезпечує найвищу стабільність роботи емулятора. Водночас кінцеві робочі станції та сервери моніторингу розгорнуто за допомогою Oracle VM VirtualBox.

Машину Suricata-Serv буде розгорнуто на базі Ubuntu Server 26.04 для забезпечення високої продуктивності та стабільної роботи в inline-режимі. Робочу станцію Admin-WS реалізовано на Ubuntu Desktop 26.04, що надає графічний інтерфейс для аналізу журналів подій та керування мережею.

Вибір мережного обладнання базується на функціональних вимогах до побудови багаторівневої мережної інфраструктури, зокрема продуктивності, масштабованості та підтримки засобів маршрутизації і сегментації трафіку. Для реалізації маршрутизаційного рівня використовуються пристрої, орієнтовані на високу швидкість обробки мережних пакетів, підтримку динамічних протоколів маршрутизації та можливість розширення функціональності. У даному проєкті застосовуються маршрутизатори Cisco ISR 4331 та MikroTik CCR2004-16G-2S+.

Cisco ISR 4331 є представником інтелектуальних маршрутизаторів Cisco серії ISR 4000 (Integrated Services Routers), який працює під керуванням операційної системи Cisco IOS і забезпечує високу пропускну здатність завдяки розділенню площин керування та передачі даних (data і control planes).



Рисунок 2.12 – Модель Cisco ISR 4331

Пристрій обладнаний 3 вбудованими WAN/LAN інтерфейсами 10/100/1000 Mbps, 2 слотами для модулів NIM (Network Interface Modules) та 1 слотом ISC для встановлення сигнального процесора (PVDM4). Маршрутизатор забезпечує

номінальну пропускну здатність від 100 до 300 Мбіт/с із ліцензійним шейпінгом трафіку (Performance License) та реалізує апаратну підтримку списків контролю доступу (Extended ACL), технологій забезпечення якості обслуговування (QoS/HQoS) та механізмів інспекції трафіку (Zone-Based Firewall).

MikroTik CCR2004-16G-2S+ є високопродуктивним промисловим маршрутизатором серії Cloud Core Router, який побудований на базі потужного багатоядерного ARM-процесора. Пристрій обладнаний 16 гігабітними портами Ethernet та двома високошвидкісними портами 10G SFP+. Функціонуючи під керуванням операційної системи RouterOS v7.20.6 з підтримкою 64-бітної архітектури, маршрутизатор забезпечує апаратне прискорення обробки IPsec-трафіку, динамічну маршрутизацію (OSPFv2/v3, BGP), гнучке керування таблицями маршрутизації (зокрема полісі-маршрутизацію PBR), а також реалізацію Stateful Firewall із підтримкою трекінгу з'єднань (Connection Tracking) та механізмів трансляції адрес (Source/Destination NAT).

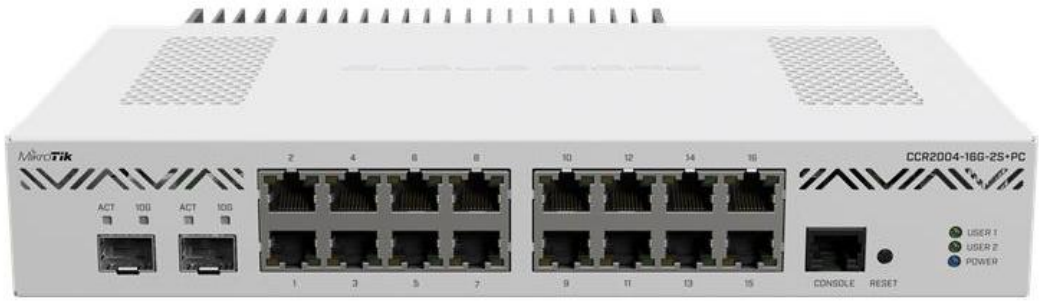


Рисунок 2.13 – MikroTik моделі CCR2004-16G-2S+

Для побудови комутаційного рівня ключовими критеріями є кількість портів, пропускна здатність інтерфейсів та підтримка механізмів сегментації мережі. У проєкті використовуються комутатори Cisco WS-C2960+24LC-S та MikroTik CRS328-24P-4S+RM.

Комутатор Cisco WS-C2960+24LC-S належить до серії Catalyst 2960, яка призначена для побудови рівня доступу в корпоративних мережах. Комутатор має 24 порти 10/100/1000Base-T та 4 фіксовані гігабітні порти SFP для магістральних з'єднань. Працюючи під керуванням ПЗ Cisco IOS із набором функцій LAN Base, пристрій підтримує базу даних на 16 000 MAC-адрес, до 1023 активних віртуальних

локальних мереж, агрегацію каналів Link Aggregation (LACP/PAgP), сімейство протоколів Spanning Tree (PVST+, Rapid PVST+, MSTP) та інструменти безпеки портів (Port Security, DHCP Snooping, Dynamic ARP Inspection). Він ідеально підходить для організації мережного доступу в офісах та навчальних закладах, забезпечуючи надійне підключення та простоту адміністрування.



Рисунок 2.14 – Комутатор серії Cisco Catalyst 2960

MikroTik CRS328-24P-4S+RM є керованим комутатором корпоративного рівня, що поєднує функціональність Layer 2 та частково Layer 3 пристроїв. Він оснащений 24 Gigabit Ethernet портами з підтримкою PoE-out та 4 SFP+ 10G uplink-портами, що забезпечує високу швидкість передачі даних між сегментами мережі.



Рисунок 2.15 – Комутатор MikroTik CRS328-24P-4S+RM

Пристрій має апаратну підтримку комутації (switch chip), що дозволяє обробляти трафік на швидкості лінії без значного навантаження на CPU. Підтримуються VLAN (802.1Q), STP/RSTP, mirror-порти, а також базові функції маршрутизації в RouterOS. CRS328 також підтримує керування живленням через PoE-out, що дозволяє підключати та живити мережні пристрої (наприклад, точки доступу або IP-камери) без додаткових блоків живлення.

Висновок до другого розділу

У другому розділі було виконано проєктування мережної інфраструктури та визначено основні засоби забезпечення її безпеки. Розроблено топологію комп’ютерної мережі та схему адресації, що забезпечують логічний розподіл мережних ресурсів і можливість подальшого масштабування інфраструктури.

Сформовано технічні вимоги до інтеграції системи виявлення вторгнень Suricata, визначено особливості її конфігурації та обрано набори правил для виявлення мережних загроз. Також розроблено систему збору, обробки та візуалізації подій безпеки на основі Elastic Stack, що забезпечує централізований моніторинг мережної активності та інцидентів інформаційної безпеки.

Окрім цього, визначено супутні засоби захисту мережі, зокрема централізовану автентифікацію користувачів, журналювання подій мережного обладнання та додаткові механізми мережної безпеки. На завершення здійснено вибір програмного забезпечення та мережного обладнання, яке відповідає вимогам проєкту та забезпечує можливість реалізації запропонованих рішень у віртуальному середовищі.

РОЗДІЛ 3 НАЛАШТУВАННЯ ТА ТЕСТУВАННЯ МЕРЕЖНОЇ ІНФРАСТУКТУРИ

3.1 Налагодження обладнання та засобів безпеки

Першочерговим етапом розгортання інфраструктури є налаштування фізичних і логічних інтерфейсів. Цей крок передбачає призначення IP-адрес відповідно до розробленого плану адресації, активацію портів та створення віртуальних інтерфейсів зворотного зв'язку (Loopback) для забезпечення стабільної маршрутизації. Як приклад конфігурації інтерфейсів нижче наведено лістинги налаштувань. Інші пристрої налаштовуються аналогічно.

Лістинг налаштувань маршрутизатора Cisco-DIST:

```
Cisco-DIST(config)#int gi0/0
Cisco-DIST(config-if)#desc LINK-TO-CISCO-CORE
Cisco-DIST(config-if)#ip address 10.0.0.6 255.255.255.252
Cisco-DIST(config-if)#no shutdown
Cisco-DIST(config-if)#exit
Cisco-DIST(config)#int gi0/1
Cisco-DIST(config-if)#desc LINK-TO-MIKROTIK-DIST
Cisco-DIST(config-if)#ip address 10.0.0.13 255.255.255.252
Cisco-DIST(config-if)#no shutdown
Cisco-DIST(config-if)#exit
Cisco-DIST(config)#int gi0/2
Cisco-DIST(config-if)#desc LINK-TO-LAN-A
Cisco-DIST(config-if)#ip address 192.168.10.1 255.255.255.0
Cisco-DIST(config-if)#no shutdown
Cisco-DIST(config-if)#exit
Cisco-DIST(config)#int gi0/3
Cisco-DIST(config-if)#desc LINK-TO-LAN-B
Cisco-DIST(config-if)#ip address 192.168.20.1 255.255.255.0
Cisco-DIST(config-if)#no shutdown
Cisco-DIST(config-if)#exit
Cisco-DIST(config)#int loopback0
Cisco-DIST(config-if)#ip address 2.2.2.1 255.255.255.255
Cisco-DIST(config-if)#exit
```

Лістинг налаштувань маршрутизатора Mikrotik-DIST:

```
[admin@Mikrotik-DIST] > ip address add address=10.0.0.10/30 interface=ether1
[admin@Mikrotik-DIST] > interface set ether1 comment="LINK-TO-CISCO-CORE"
[admin@Mikrotik-DIST] > ip address add address=10.0.0.14/30 interface=ether2
[admin@Mikrotik-DIST] > interface set ether2 comment="LINK-TO-CISCO-DIST"
[admin@Mikrotik-DIST] > ip address add address=192.168.30.1/24 interface=ether3
[admin@Mikrotik-DIST] > interface set ether3 comment="LINK-TO-LAN-C"
[admin@Mikrotik-DIST] > ip address add address=192.168.40.1/24 interface=ether4
[admin@Mikrotik-DIST] > interface set ether4 comment="LINK-TO-LAN-D"
[admin@Mikrotik-DIST] > ip address add address=2.2.2.2/32 interface=lo
```

Наступним етапом побудови є налаштування динамічної маршрутизації на основі протоколу OSPF. Захист забезпечено за допомогою сучасного стійкого алгоритму хешування HMAC-SHA-256. У межах конфігурування процесів

маршрутизації для всіх пристроїв було визначено унікальні ідентифікатори в межах магістральної зони (Area 0 / backbone). При цьому на інтерфейсах, що дивляться у бік користувацьких підмереж, активовано пасивний режим, який повністю блокує витік службового OSPF-трафіку за межі контрольованого периметра. Приклади налаштування маршрутизації наведено нижче, всі інші пристрої налаштовуються аналогічно:

Лістинг налаштувань маршрутизатора Cisco-CORE:

```
Cisco-CORE(config)#key chain OSPF_EDGE
Cisco-CORE(config-keychain)#key 1
Cisco-CORE(config-keychain-key)#key-string dAkoK8ijMyaF
Cisco-CORE(config-keychain-key)#cryptographic-algorithm hmac-sha-256
Cisco-CORE(config-keychain-key)#exit
Cisco-CORE(config-keychain)#exit
Cisco-CORE(config)#key chain OSPF_CISCO_DIST
Cisco-CORE(config-keychain)#key 1
Cisco-CORE(config-keychain-key)#key-string nE7dcA2lQ8zo
Cisco-CORE(config-keychain-key)#cryptographic-algorithm hmac-sha-256
Cisco-CORE(config-keychain-key)#exit
Cisco-CORE(config-keychain)#exit
Cisco-CORE(config)#key chain OSPF_MIKROTIK_DIST
Cisco-CORE(config-keychain)#key 1
Cisco-CORE(config-keychain-key)#key-string cYssO6trGzz
Cisco-CORE(config-keychain-key)#cryptographic-algorithm hmac-sha-256
Cisco-CORE(config-keychain-key)#exit
Cisco-CORE(config-keychain)#exit
Cisco-CORE(config)#router ospf 1
Cisco-CORE(config-router)#router-id 1.1.1.2
Cisco-CORE(config-router)#log-adjacency-changes
Cisco-CORE(config-router)#passive-interface gi0/3
Cisco-CORE(config-router)#exit
Cisco-CORE(config)#int gi0/0
Cisco-CORE(config-if)#ip ospf authentication key-chain OSPF_EDGE
Cisco-CORE(config-if)#ip ospf 1 area 0
Cisco-CORE(config-if)#exit
Cisco-CORE(config)#int gi0/1
Cisco-CORE(config-if)#ip ospf authentication key-chain OSPF_CISCO_DIST
Cisco-CORE(config-if)#ip ospf 1 area 0
Cisco-CORE(config-if)#exit
Cisco-CORE(config)#int gi0/2
Cisco-CORE(config-if)#ip ospf authentication key-chain OSPF_MIKROTIK_DIST
Cisco-CORE(config-if)#ip ospf 1 area 0
Cisco-CORE(config-if)#exit
Cisco-CORE(config)#int gi0/3
Cisco-CORE(config-if)#ip ospf 1 area 0
Cisco-CORE(config-if)#exit
```

Лістинг налаштувань маршрутизатора Mikrotik-DIST:

```
[admin@Mikrotik_DIST] > /routing ospf instance add \
\... name=ospf-main \
\... router-id=2.2.2.2
[admin@Mikrotik_DIST] > /routing/ospf/area add \
\... name=backbone \
\... area-id=0.0.0.0 \
\... instance=ospf-main
[admin@Mikrotik_DIST] > /routing ospf interface-template add \
```

```
\... interfaces=ether1 \
\... area=backbone \
\... auth=sha256 \
\... auth-key=" cYssO6trGzz"  \
\...  auth-id=1
[admin@Mikrotik_DIST] > /routing ospf interface-template add \
\... interfaces=ether2 \
\... area=backbone \
\... auth=sha256 \
\... auth-key=" eJ5wU9sjE6qd" \
\...  auth-id=1
[admin@Mikrotik_DIST] > /routing ospf interface-template add \
\... interfaces=ether3 \
\... area=backbone \
\... passive
[admin@Mikrotik_DIST] > /routing ospf interface-template add \
\... interfaces=ether4 \
\... area=backbone \
\... passive
```

Маршрутизатори рівня розподілу налаштовуємо як DHCP-сервери. У межах цього процесу для кожної підмережі задано адресний простір, шлюзи за замовчуванням та час оренди.

Лістинг налаштувань маршрутизатора Cisco-DIST:

```
Cisco-DIST(config)#ip dhcp pool LAN-A
Cisco-DIST(dhcp-config)#network 192.168.10.0 255.255.255.0
Cisco-DIST(dhcp-config)#default-router 192.168.10.1
Cisco-DIST(dhcp-config)#lease 0 12
Cisco-DIST(dhcp-config)#exit
Cisco-DIST(config)#ip dhcp excluded-address 192.168.10.1 192.168.10.10
Cisco-DIST(config)#ip dhcp excluded-address 192.168.10.171 192.168.10.254
Cisco-DIST(config)#ip dhcp pool LAN-B
Cisco-DIST(dhcp-config)#network 192.168.20.0 255.255.255.0
Cisco-DIST(dhcp-config)#default-router 192.168.20.1
Cisco-DIST(dhcp-config)#lease 0 12
Cisco-DIST(dhcp-config)#exit
Cisco-DIST(config)#ip dhcp excluded-address 192.168.20.1 192.168.20.10
Cisco-DIST(config)#ip dhcp excluded-address 192.168.20.171 192.168.20.254
```

На пристроях Mikrotik надалі налаштування виконуються через утиліту WinBox з робочої станції адміністратора. Процес було реалізовано за допомогою інтерактивного майстра налаштування (DHCP Setup).

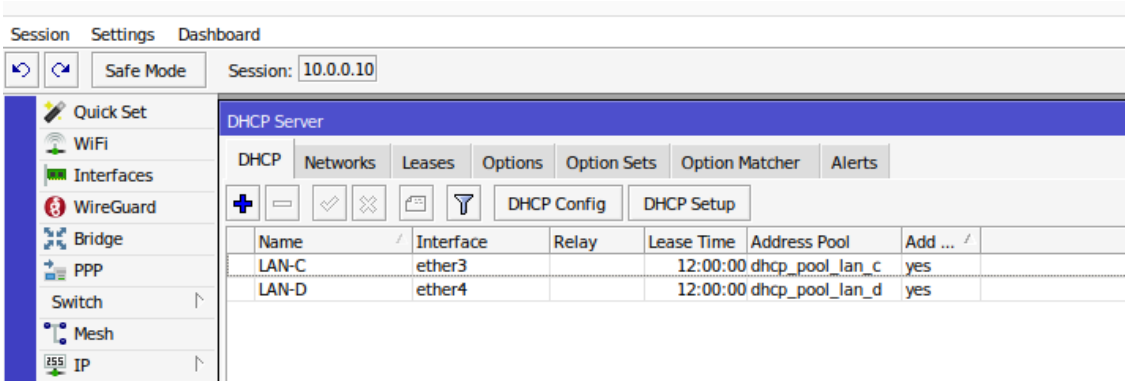


Рисунок 3.1 – Результат налаштування DHCP

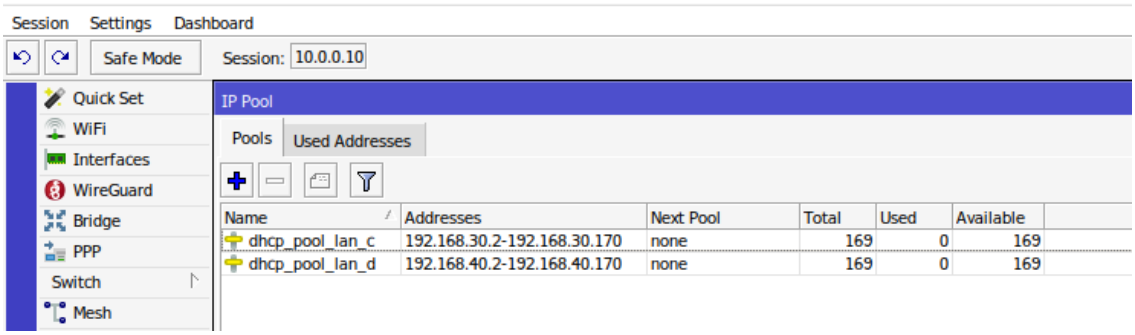


Рисунок 3.2 – Результат налаштування діапазонів адрес

Для забезпечення проактивного захисту мережного периметра та виявлення потенційних загроз в інфраструктурі було розгорнуто систему виявлення та запобігання вторгненням Suricata версії 8.0.3. Інсталяцію виконано стандартним методом за допомогою пакетного менеджера після попереднього оновлення компонентів операційної системи.

Основні параметри роботи Suricata знаходяться у конфігураційному файлі — /etc/suricata/suricata.yaml. Спочатку було деталізовано змінну HOME_NET шляхом внесення всіх підмереж інфраструктури, включаючи технологічні лінки магістральної маршрутизації, користувацькі сегменти та інші внутрішні адреси.

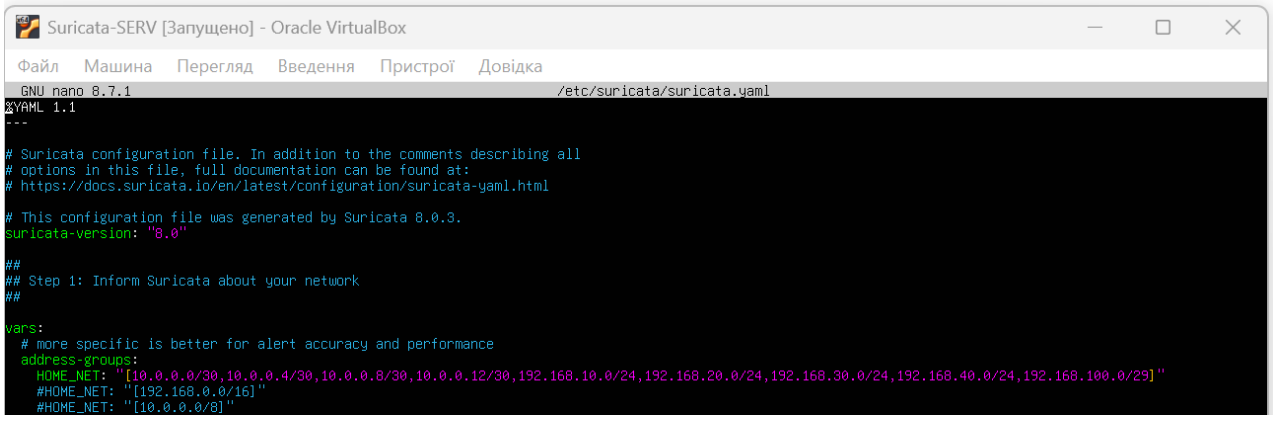


Рисунок 3.3 – Зміна значень HOME_NET

З метою забезпечення повної видимості процесів у мережі та подальшої кореляції логів, налаштування підсистеми розширеного логування EVE log було повністю приведено у відповідність до проєктних рішень, детально визначених у другому розділі роботи.

Для забезпечення безпосереднього перехоплення й фільтрації трафіку проєкт передбачає роботу через рушій af-packet. Функціонування системи переведено в

активний режим inline за допомогою директиви copy-mode: ips, для чого між інтерфейсами enp0s3 та enp0s8 налаштовано взаємне копіювання і транзит трафіку із застосуванням кластеризації типу cluster_flow для балансування навантаження.

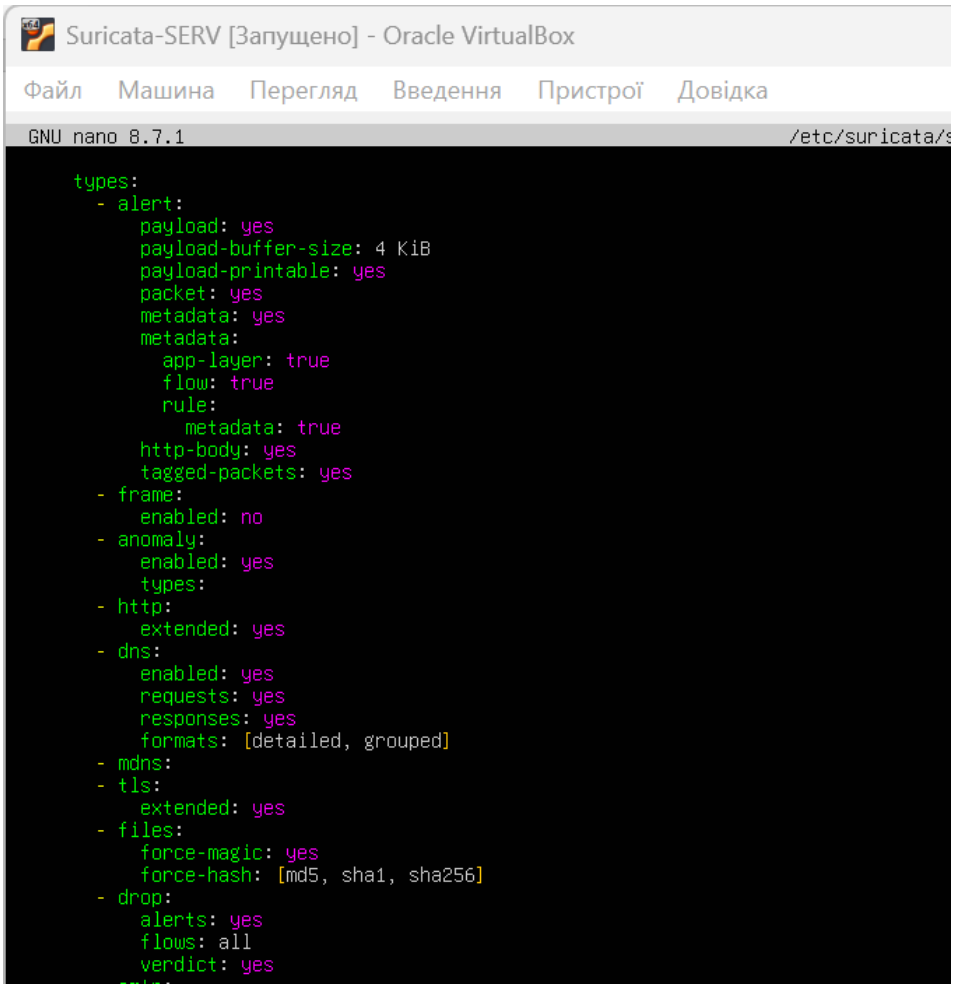


Рисунок 3.4 – Налаштування параметрів логування

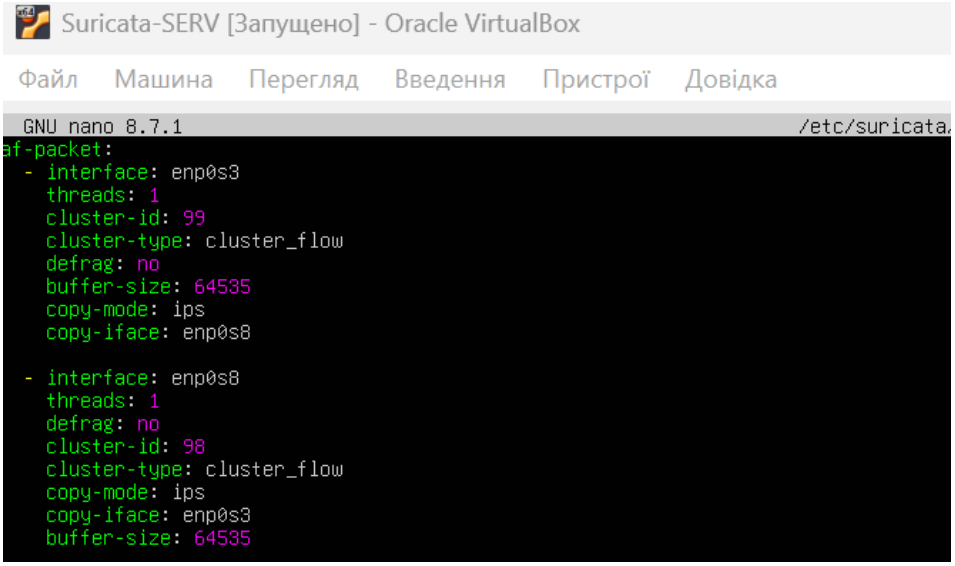


Рисунок 3.5 – Налаштування параметрів af-packet

Наступним кроком після конфігурування основного файла налаштувань став процес підключення та актуалізації сигнатурних баз системи. Для активації конкретних джерел та наборів правил, що були обґрунтовані на етапі проєктування у другому розділі роботи, використали команду «sudo suricata-update enable-source назва_набору», після чого за допомогою штатної утиліти «sudo suricata-update» здійснено їх безпосереднє скачування та інтеграцію.

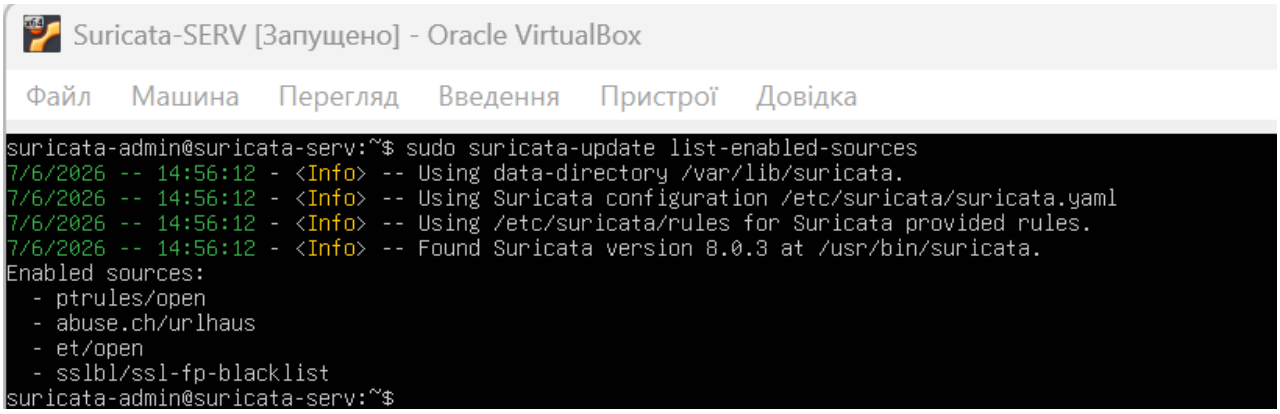


Рисунок 3.6 – Підвантажені набори сигнатур

Після підключення наборів сигнатур було здійснено їх адаптацію для роботи в активному режимі запобігання вторгненням (IPS). Оскільки за замовчуванням більшість правил націлені лише на фіксацію подій (alert), критично важливі класи загрози було примусово переведено в режим блокування (drop). Модифікація виконувалася за визначеним принципом:

- всі правила з репозиторіїв SSLBL та URLhaus;
- усі сигнатури критичного (першого) класу пріоритетності (зокрема trojan-activity, shellcode-detect, exploit-kit та attempted-user);
- окремі загрози другого класу правила (зокрема suspicious-login, misc-attack, denial-of-service та coin-mining);
- правила, які за метаданими мають середній або високий рівень достовірності спрацьовування у поєднанні з високим чи критичним рівнем важливості.

Для цієї автоматичної модифікації діючі регулярні вирази замін було внесено до конфігураційного файла /etc/suricata/modify.conf.

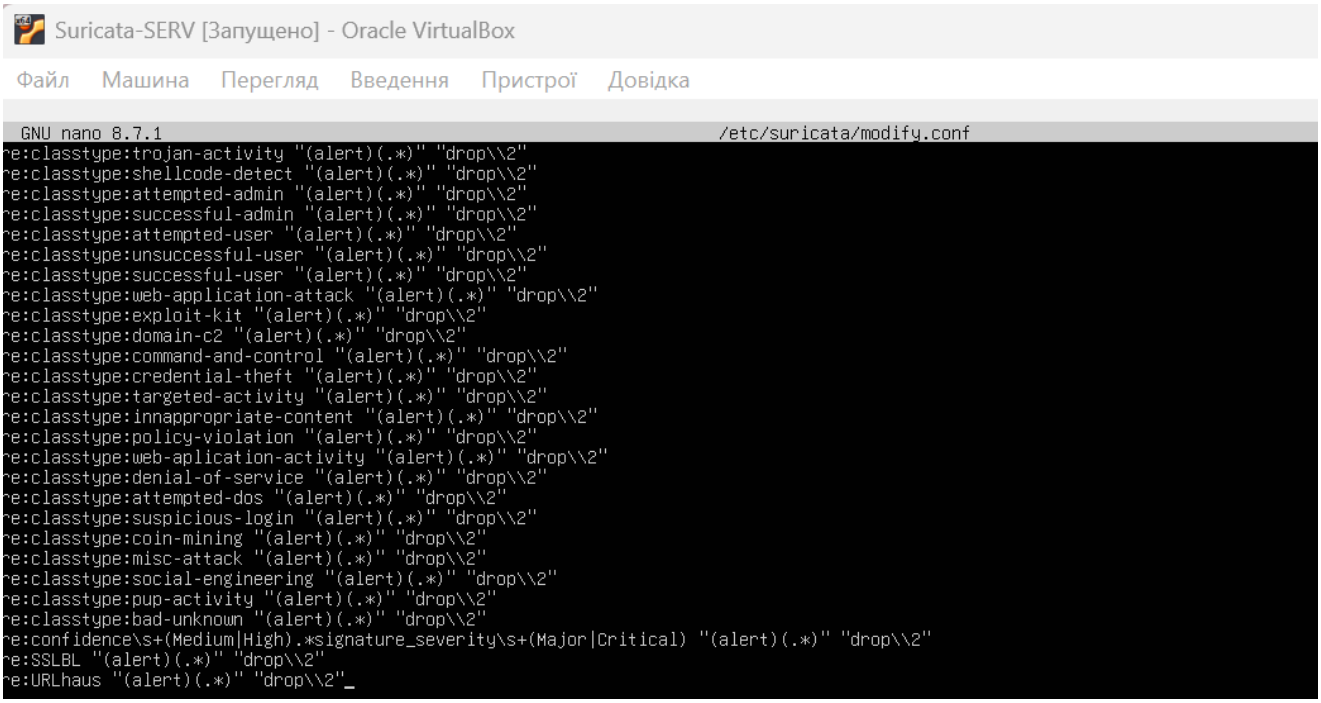


Рисунок 3.7 – Вміст файлу /etc/suricata/modify.conf

Далі перезавантажили саму систему Suricata для того, щоб зміни вступили в дію. Під час моніторингу логів у реальному часі було виявлено правило з ідентифікатором 2200121 («SURICATA Ethertype unknown»), яке спрацьовувало занадто часто та створювало багато зайвих повідомлень. Оскільки ці спрацьовування були помилковими, було прийнято рішення вимкнути це правило, додавши його ідентифікатор до конфігураційного файла /etc/suricata/disable.conf.



Рисунок 3.8 – Вивід файлу /var/log/suricata/fast.log



Рисунок 3.9 – Вміст файлу /etc/suricata/disable.conf

Процес побудови системи моніторингу безпеки та аналізу мережних подій продовжився розгортанням компонентів централізованого збору логів на робочій станції Admin-WS. Роботу було розпочато з системи розподіленого пошуку та аналізу даних Elasticsearch версії 8.19.15. Спочатку імпортували офіційний криптографічний ключ, додали відповідний репозиторій до системи та виконали безпосереднє розгортання пакунків. Після завершення інсталяції здійснено перечитування конфігурації менеджеру системних служб, а сам сервіс безпеки додано до автозавантаження ОС та ініціалізовано для постійної роботи.

Головним файлом конфігурації сервісу є /etc/elasticsearch/elasticsearch.yml. У ньому виконали базові налаштування параметрів, а також активували вбудовані модулі безпеки, включаючи автентифікацію. Після редагування сервіс було перезавантажено.

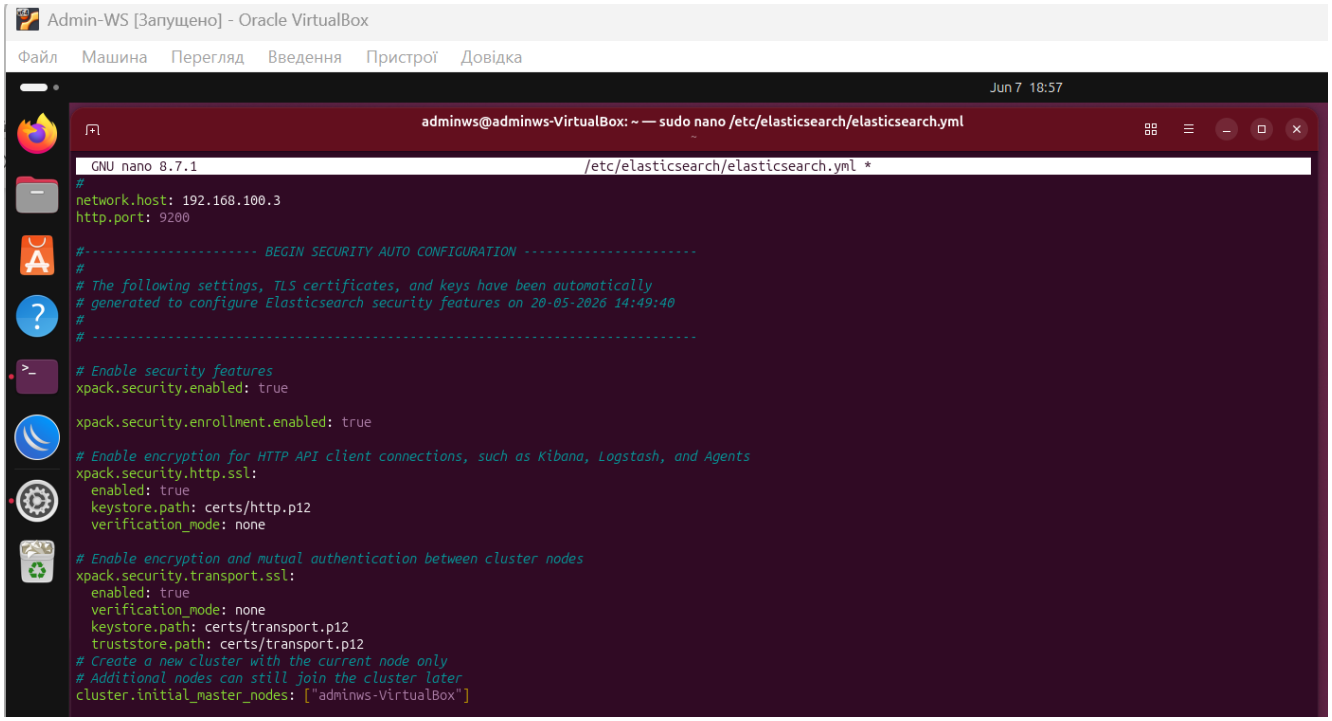


Рисунок 3.10 – Вміст конфігураційного файлу Elasticsearch

Далі провели ініціалізацію облікових записів. За допомогою вбудованої утиліти «elasticsearch-reset-password» у директорії /usr/share/elasticsearch/bin виконано скидання та генерацію нових паролів для суперкористувача elastic, який відповідає за повне адміністрування сховища, а також для системного облікового запису kibana_system.

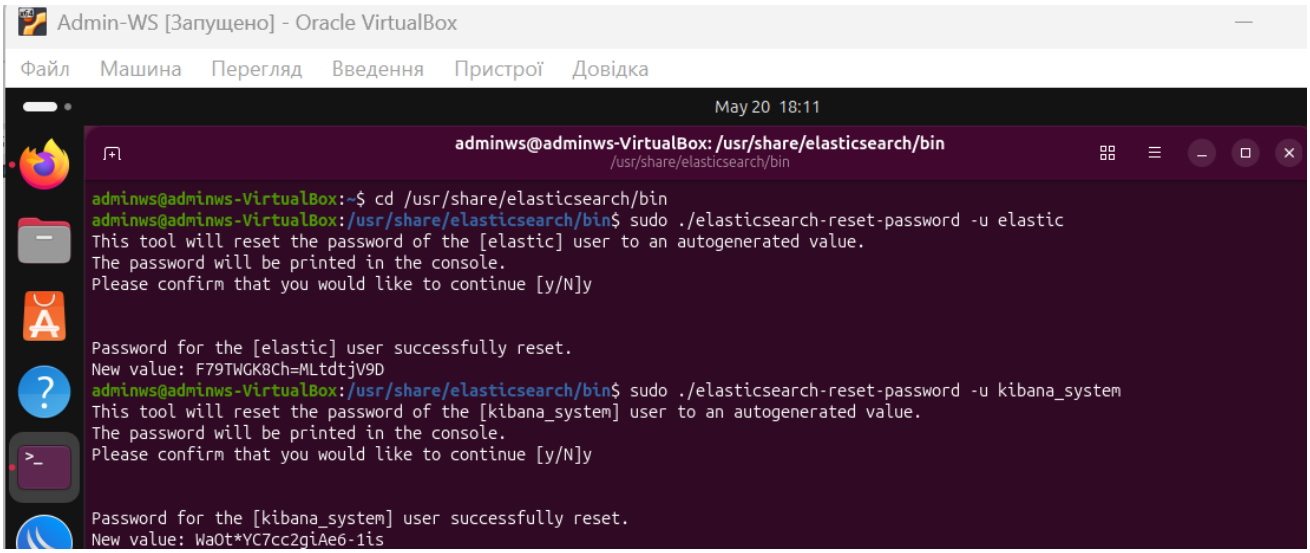


Рисунок 3.11 – Процес генерації паролів

Після завершення базового налаштування сховища було розпочато підготовку платформи візуалізації Kibana. На основі раніше доданих репозиторіїв виконано розгортання пакунка системи в ОС. Для забезпечення захисту збережених об'єктів, звітів та сесій користувачів у веб-інтерфейсі було здійснено перехід до директорії /usr/share/kibana/bin, де за допомогою вбудованої утиліти «kibana-encryption-keys» згенеровано унікальні ключі шифрування.

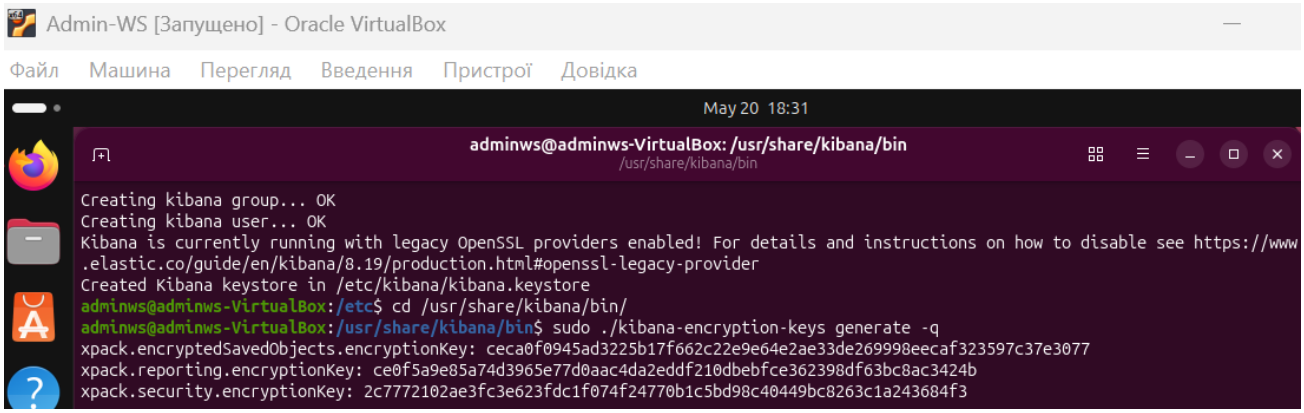


Рисунок 3.12 – Генерація ключів

Для забезпечення коректної взаємодії платформи з базою даних було внесено необхідні зміни до головного конфігураційного файла /etc/kibana/kibana.yml. У ньому визначено IP-адресу та робочий порт сервісу, вказано точний шлях для підключення до Elasticsearch, а також пароль службового облікового запису разом із раніше згенерованими ключами шифрування. Після збереження конфігурації сервіс Kibana було додано до автозавантаження системи та успішно запущено.

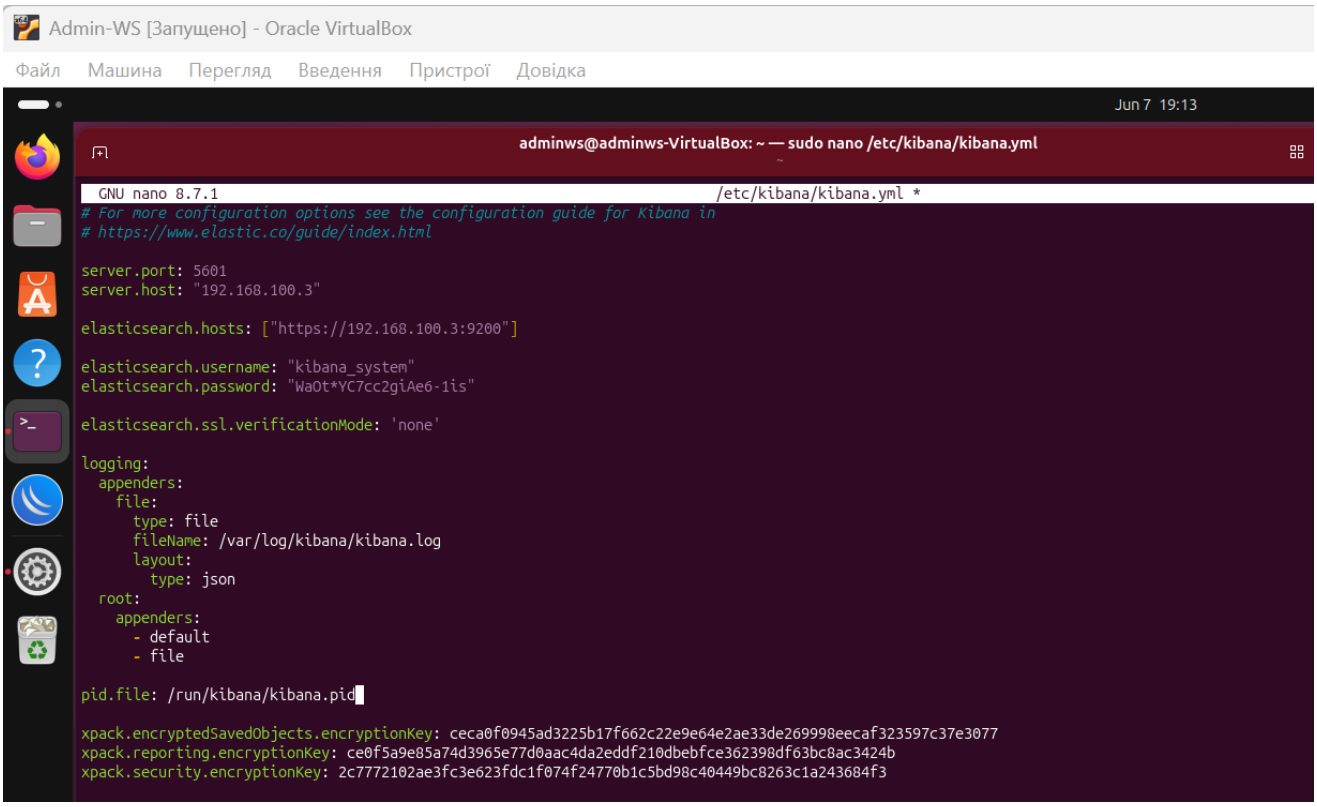


Рисунок 3.13 – Вміст конфігураційного файлу Kibana

Наступним кроком став запуск збору даних на машині Suricata-SERV. Успішно завантаживши агент Filebeat, ми приступили до налаштування параметрів у файлі /etc/filebeat/filebeat.yml. Активували модуль suricata, визначили параметри підключення до Kibana та Elasticsearch. Службу filebeat було додано до автозавантаження системи та запущено в роботу. Це дозволило повністю завершити етап налаштування підсистеми моніторингу для сенсора Suricata, забезпечивши автоматичну і безперервну відправку подій безпеки у сховище Elasticsearch.

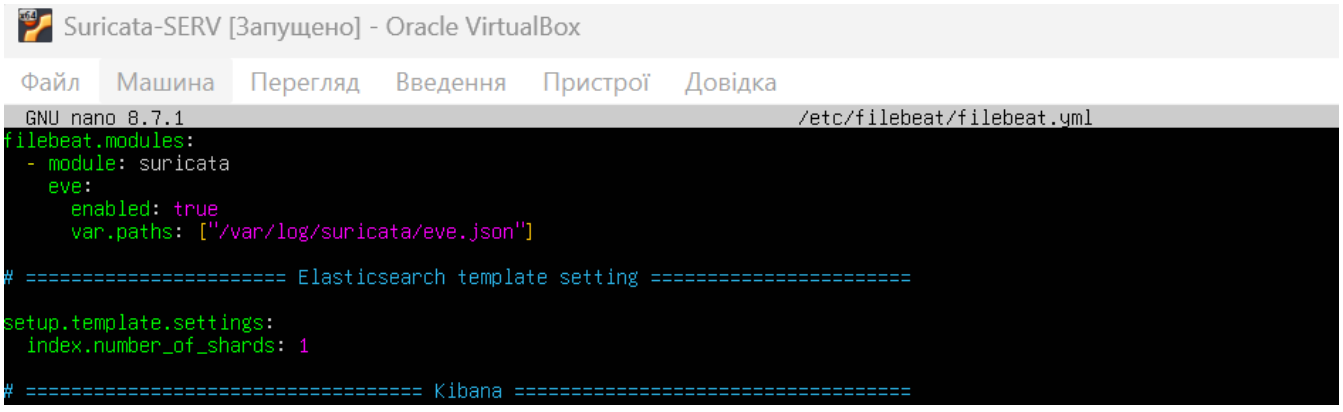


Рисунок 3.14 – Вміст конфігураційного файлу Filebeat

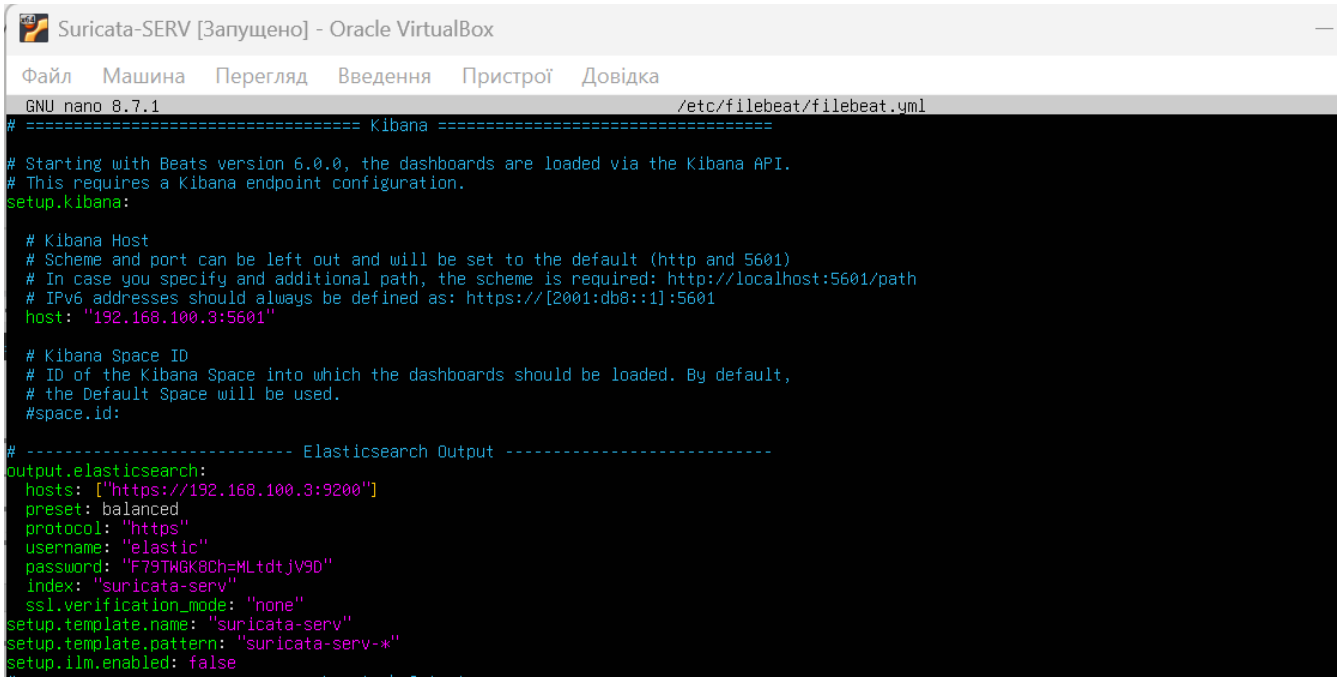


Рисунок 3.15 – Вміст конфігураційного файлу Filebeat

Після успішного встановлення пакетів FreeRADIUS на робочій станції Admin-WS було проведено налаштування сервера. У конфігураційному файлі /etc/freeradius/3.0/clients.conf було зареєстровано всі активні пристрої як AAA-клієнти.

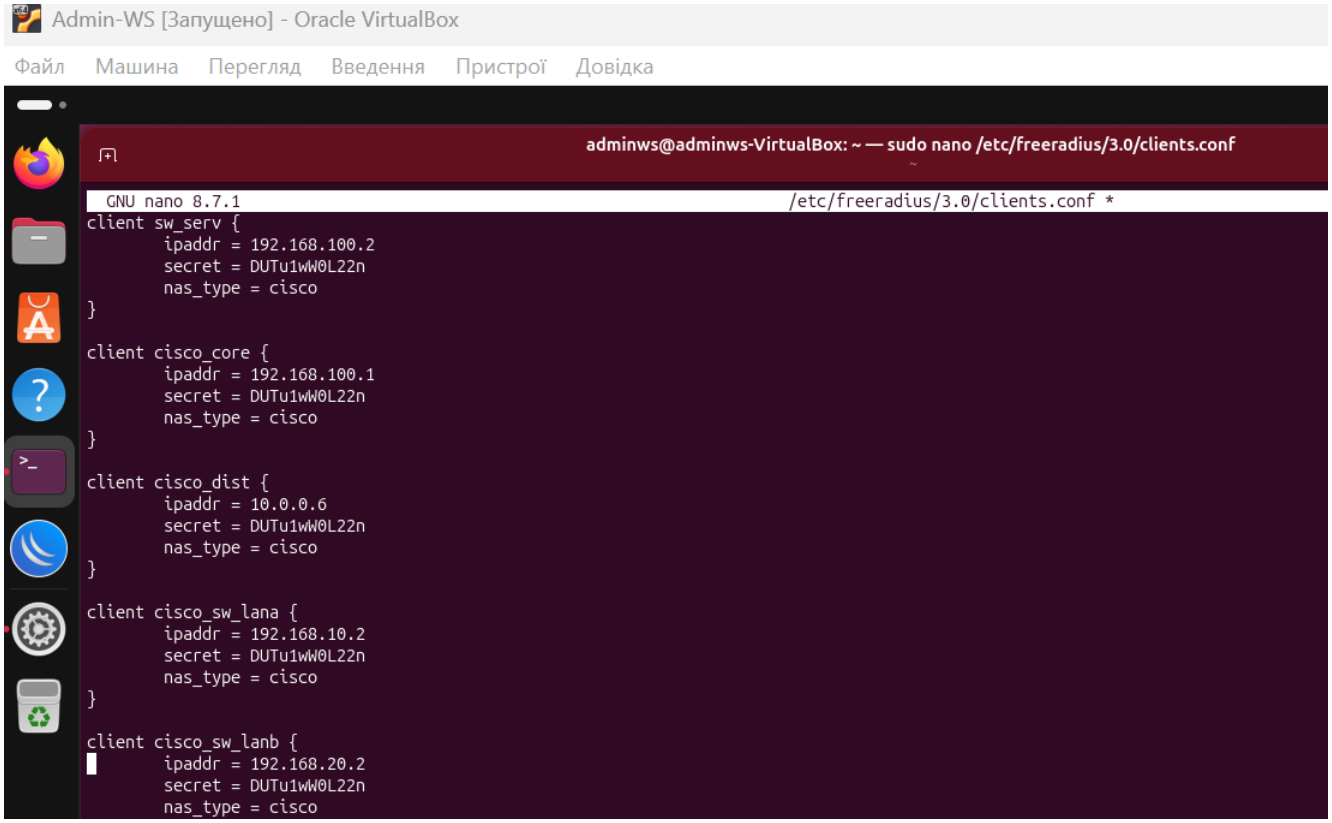


Рисунок 3.16 – Частковий вміст файлу /etc/freeradius/3.0/clients.conf

Далі у файлі /etc/freeradius/3.0/users створили два облікові записи.

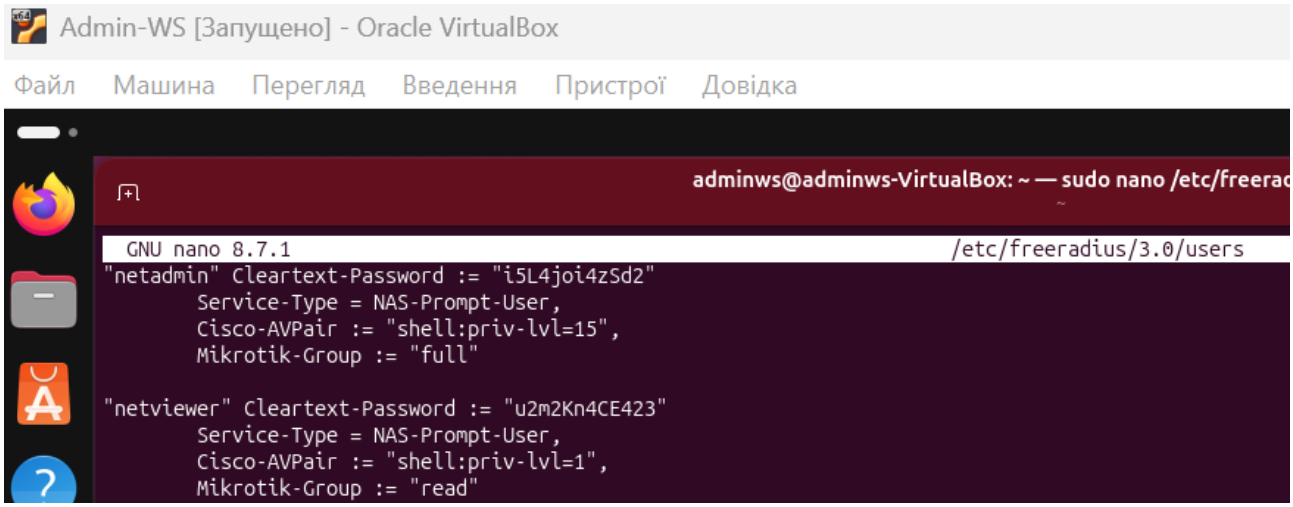


Рисунок 3.17 – Створені користувачі у файлі /etc/freeradius/3.0/users

На завершальному етапі у файлі /etc/freeradius/3.0/radiusd.conf оптимізували підсистему логування, де активували фіксацію спроб автентифікації та увімкнули запис паролів у системні журнали як для успішних, так і для відхилених запитів. Після збереження всіх внесених змін службу freeradius було запущено у роботу.

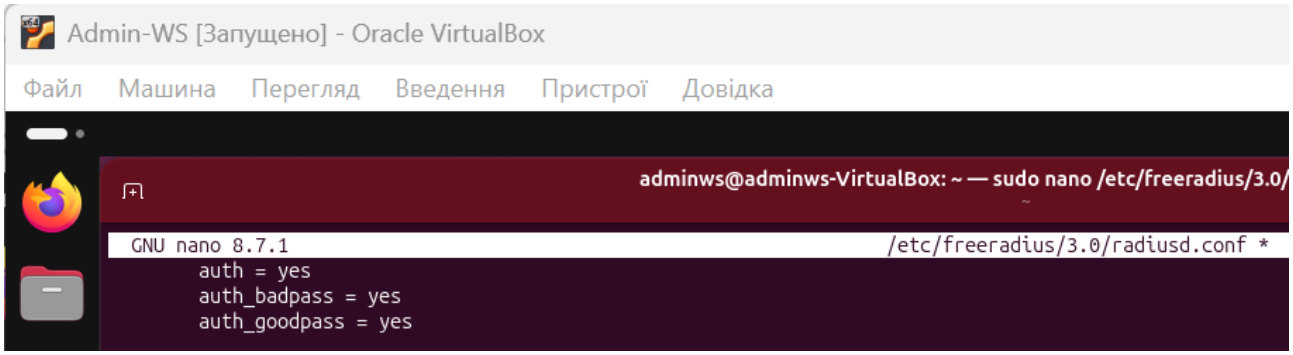


Рисунок 3.18 – Змінені параметри у файлі /etc/freeradius/3.0/radiusd.conf

Централізовану автентифікацію разом із комплексними заходами безпеки, було реалізовано на кожному мережному пристрої. Обмежили доступ до інтерфейсів керування виключно адміністративною підмережею 192.168.100.0/29, увімкнули захищений протокол SSHv2 із жорсткими таймаутами сесій, повністю заблокували вразливі протоколи та налаштували резервні локальні профілі на випадок втрати зв'язку з AAA-сервером.

Лістинг налаштувань комутатора SW-SERV:

```
SW-SERV(config)#ip access-list standard MGMT-ACCESS
SW-SERV(config-std-nacl)#permit 192.168.100.0 0.0.0.7
SW-SERV(config-std-nacl)#deny any
```

```
SW-SERV(config-std-nacl)#exit
SW-SERV(config)#aaa new-model
SW-SERV(config)#enable secret cisco12345
SW-SERV(config)#service password-encryption
SW-SERV(config)#username localadmin privilege 15 secret MysticLion
SW-SERV(config)#radius server Radius-Server
SW-SERV(config-radius-server)#$s ipv4 192.168.100.3 auth-port 1812 acct-port 1813
SW-SERV(config-radius-server)#key DUTu1wW0L22n
SW-SERV(config-radius-server)#exit
SW-SERV(config)#aaa group server radius RadiusGroup
SW-SERV(config-sg-radius)#server name Radius-Server
SW-SERV(config-sg-radius)#exit
SW-SERV(config)#aaa authentication login default group RadiusGroup local
SW-SERV(config)#aaa authorization exec default group RadiusGroup local
SW-SERV(config)#aaa authorization console
SW-SERV(config)#aaa accounting exec default start-stop group RadiusGroup
SW-SERV(config)#ip domain-name bilous.com
SW-SERV(config)#crypto key generate rsa modulus 4096
SW-SERV(config)#ip ssh version 2
SW-SERV(config)#ip ssh time-out 30
SW-SERV(config)#ip ssh authentication-retries 5
SW-SERV(config)#line vty 0 4
SW-SERV(config-line)#transport input ssh
SW-SERV(config-line)#login authentication default
SW-SERV(config-line)#authorization exec default
SW-SERV(config-line)#access-class MGMT-ACCESS in
SW-SERV(config-line)#exit
SW-SERV(config)#line con 0
SW-SERV(config-line)#login authentication default
SW-SERV(config-line)#authorization exec default
SW-SERV(config-line)#exit
SW-SERV(config)#login block-for 60 attempts 5 within 30
SW-SERV(config)#exit
```

Налаштування маршрутизатора Mikrotik-DIST:

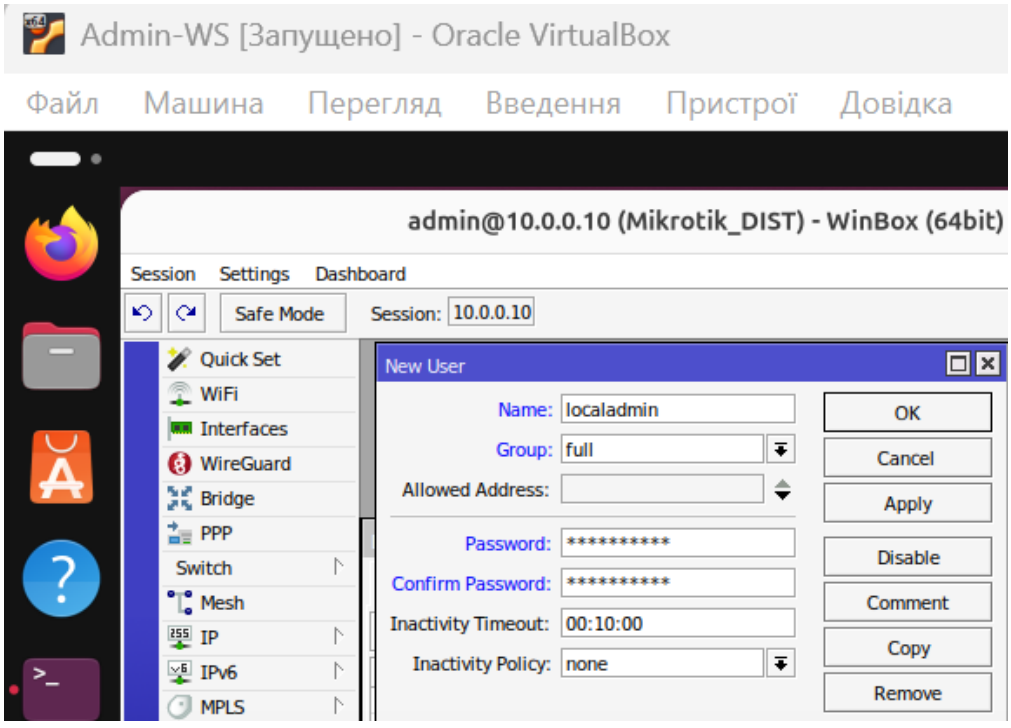


Рисунок 3.19 – Створення локального запису

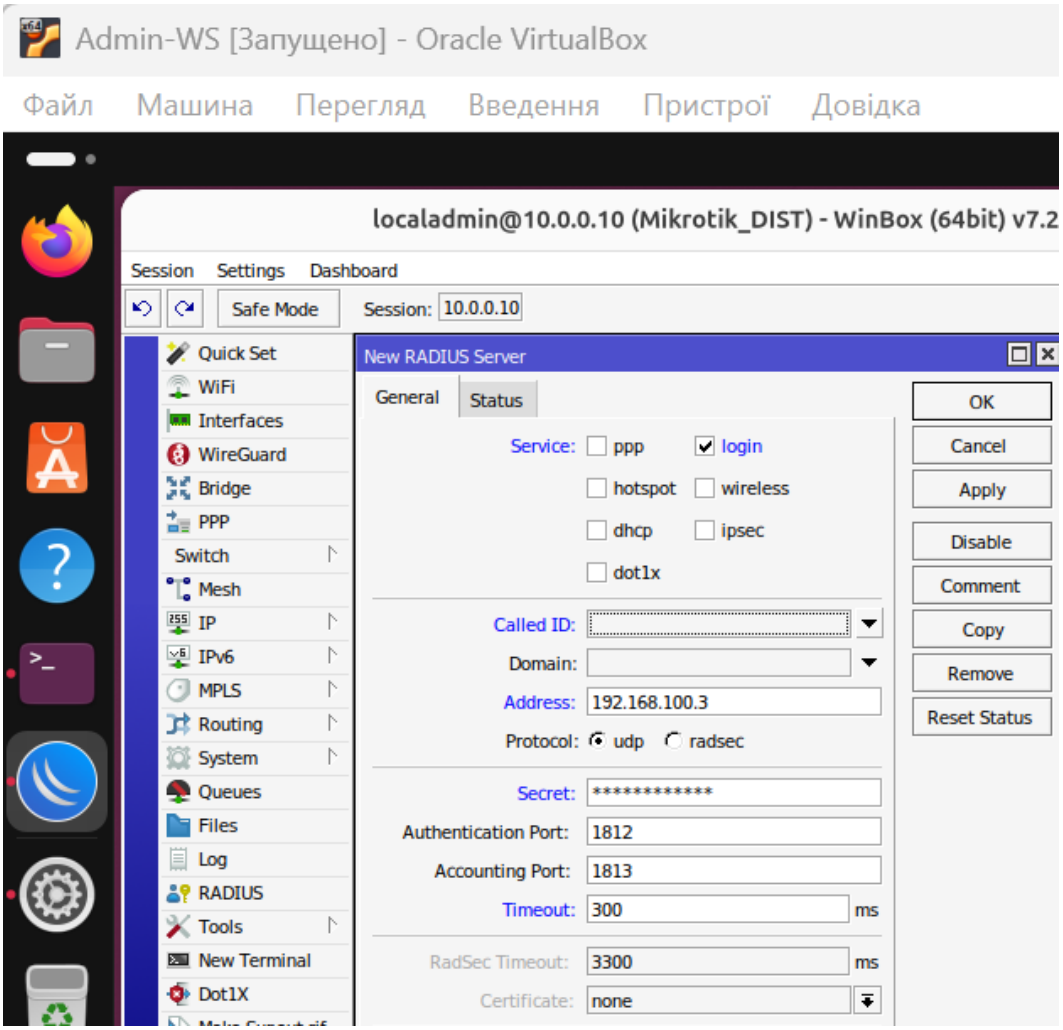


Рисунок 3.20 – Налаштування клієнта RADIUS

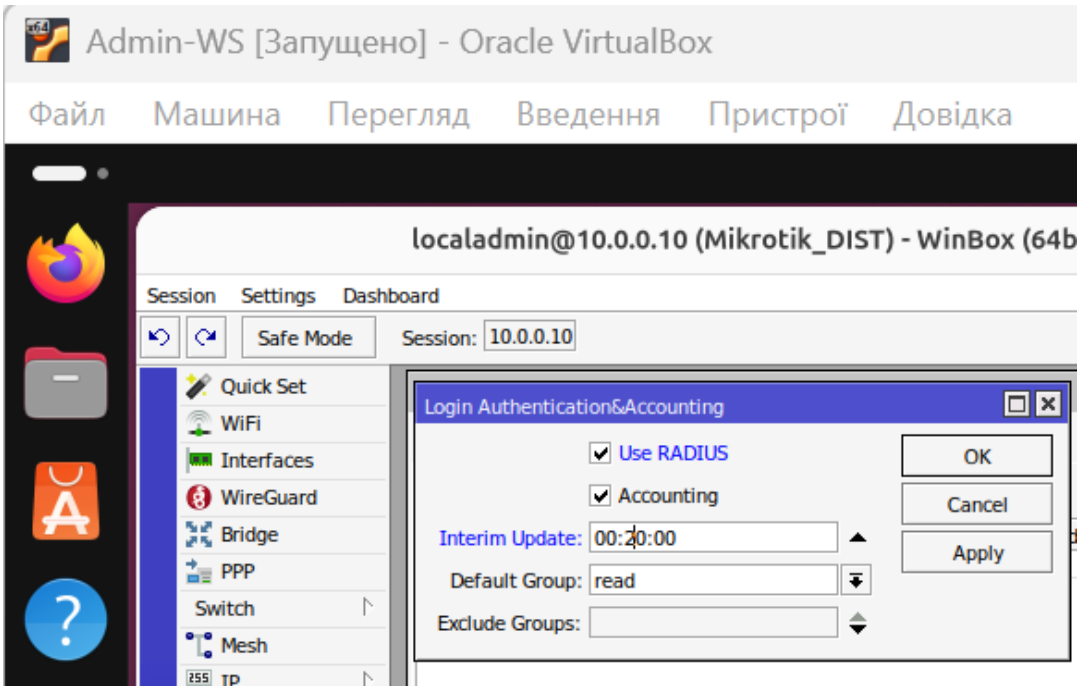


Рисунок 3.21 – Активація AAA та обліку

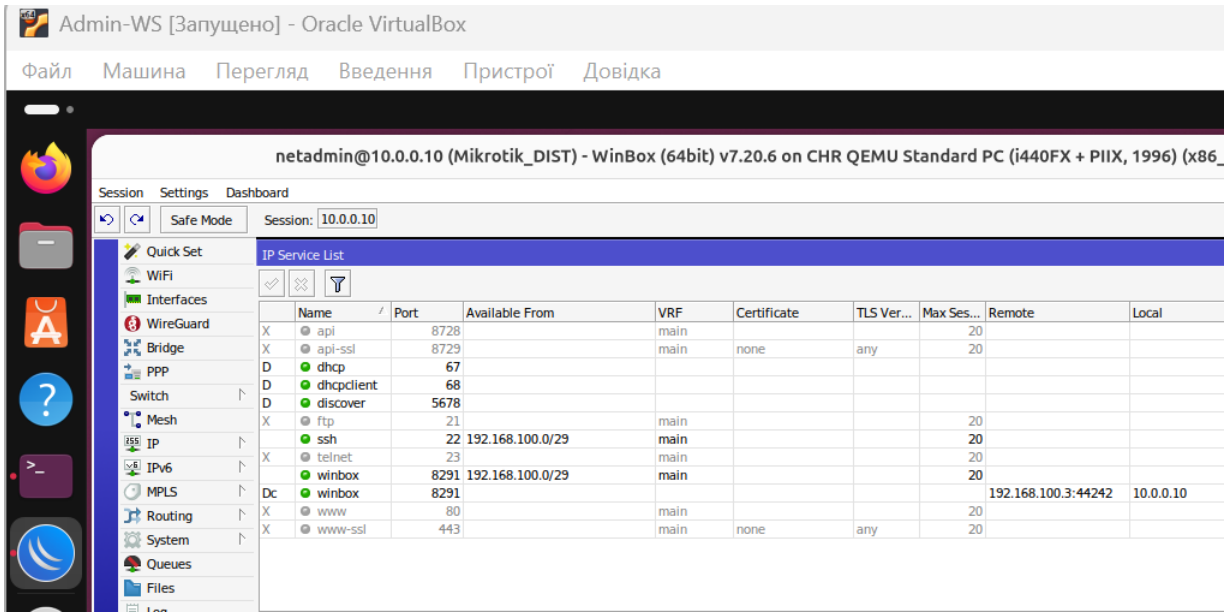


Рисунок 3.22 – Налаштування сервісів

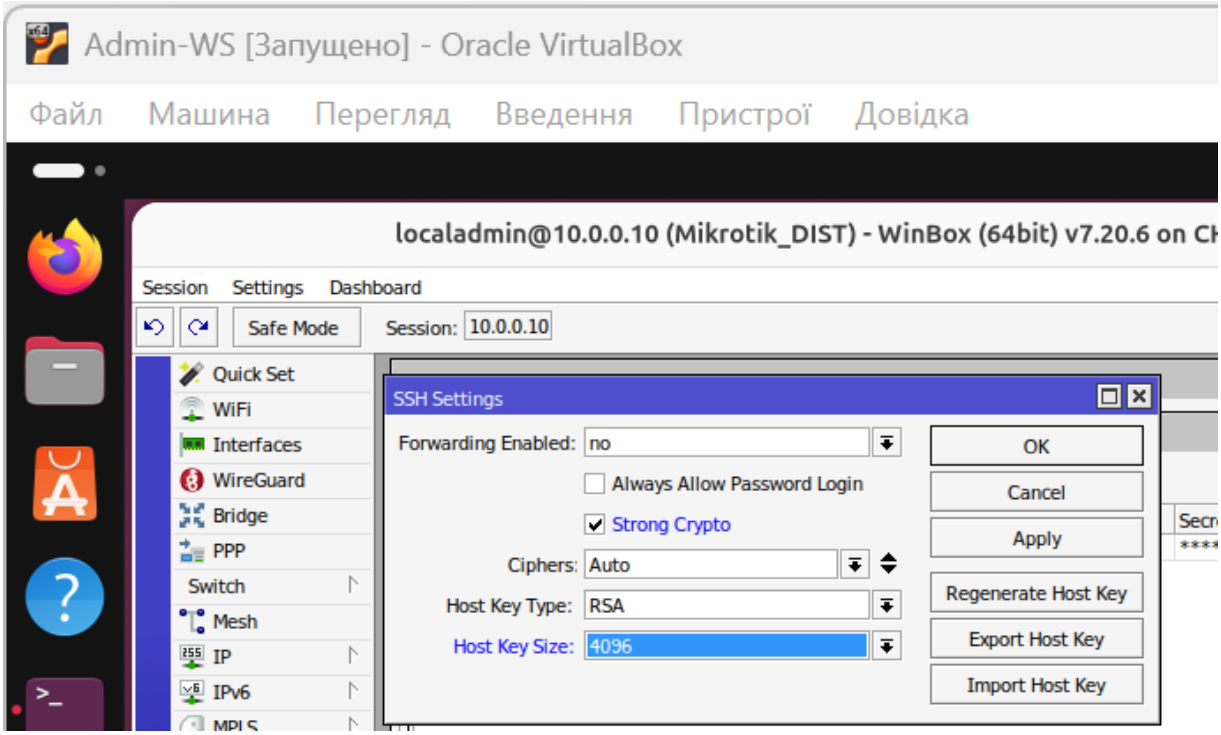


Рисунок 3.23 – Конфігурація параметрів безпеки SSH

Для централізації моніторингу та збору подій безпеки з усієї інфраструктури на робочій станції Admin-WS було розгорнуто та налаштовано окремий агент Filebeat. Для пристроїв Cisco активували модуль cisco.ios на UDP-порту 9003, а для збору журналів з обладнання MikroTik налаштували прийом повідомлень через UDP-порт 514. В блоці «output.elasticsearch» було визначено індекси та для логів з пристроїв MikroTik підключили конвеєр обробки «mikrotik-parsing-pipeline».

Для організації збору журналів FreeRADIUS створили окремий вхідний потік типу log, де вказали шлях до файлів обліку. Для коректного зведення багаторядкових записів в одну подію застосували регулярний вираз multiline.pattern, після чого налаштували перенаправлення цього масиву даних у сховище Elasticsearch під індексом radius-logs через виділений конвеєр обробки radius-parsing-pipeline.

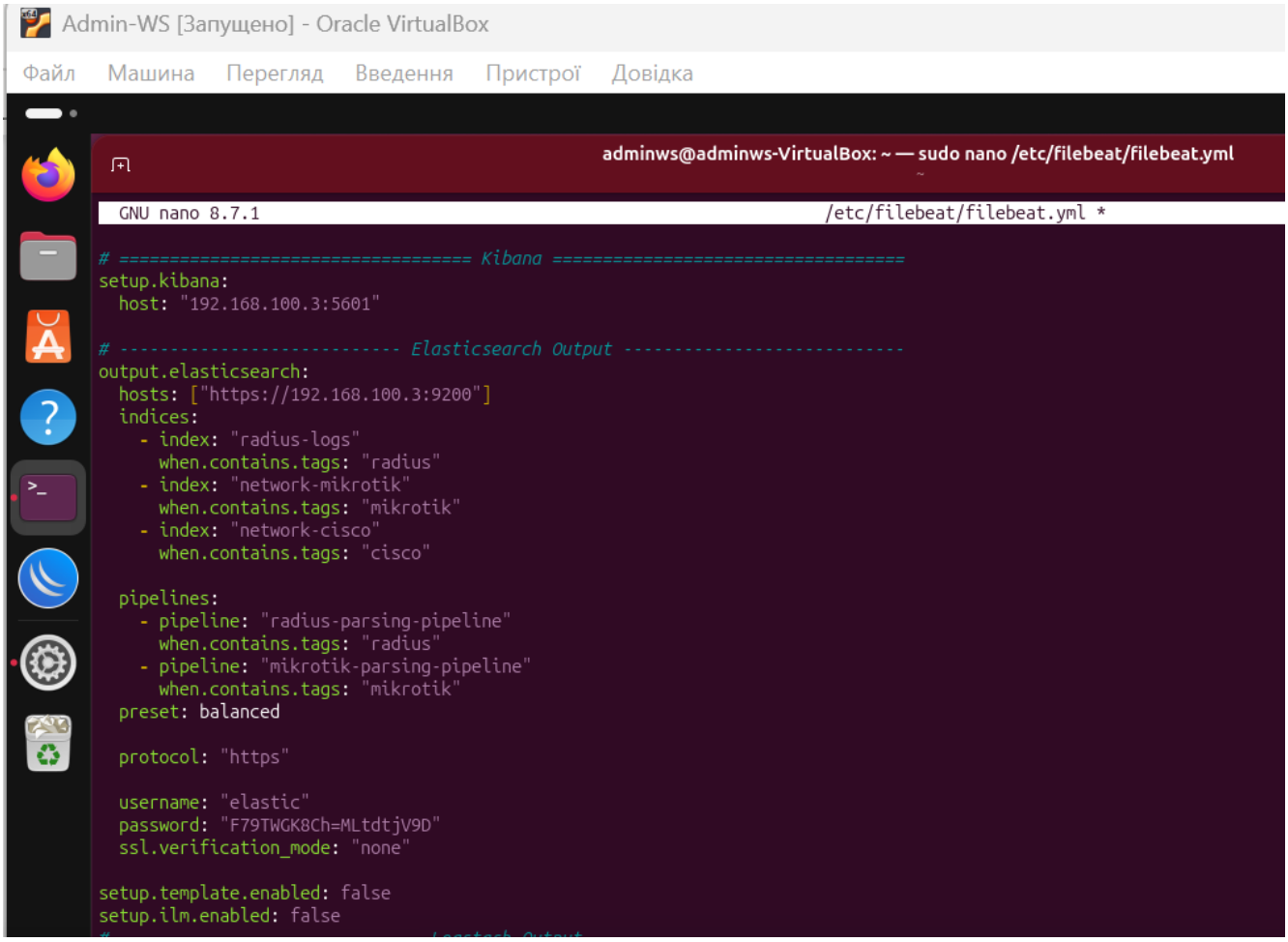


Рисунок 3.24 – Параметри підключення до Kibana та Elasticsearch

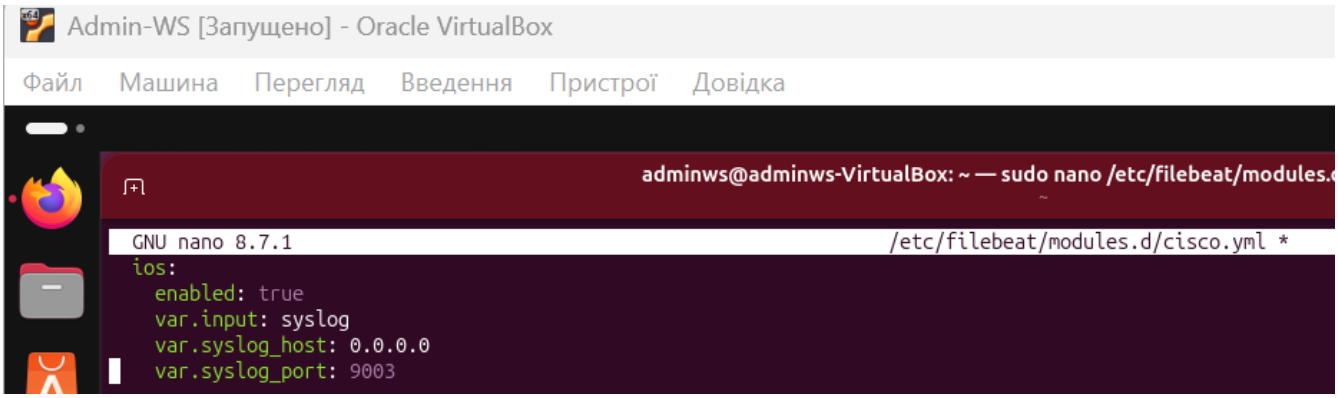


Рисунок 3.25 – Налаштування параметрів модуля cisco

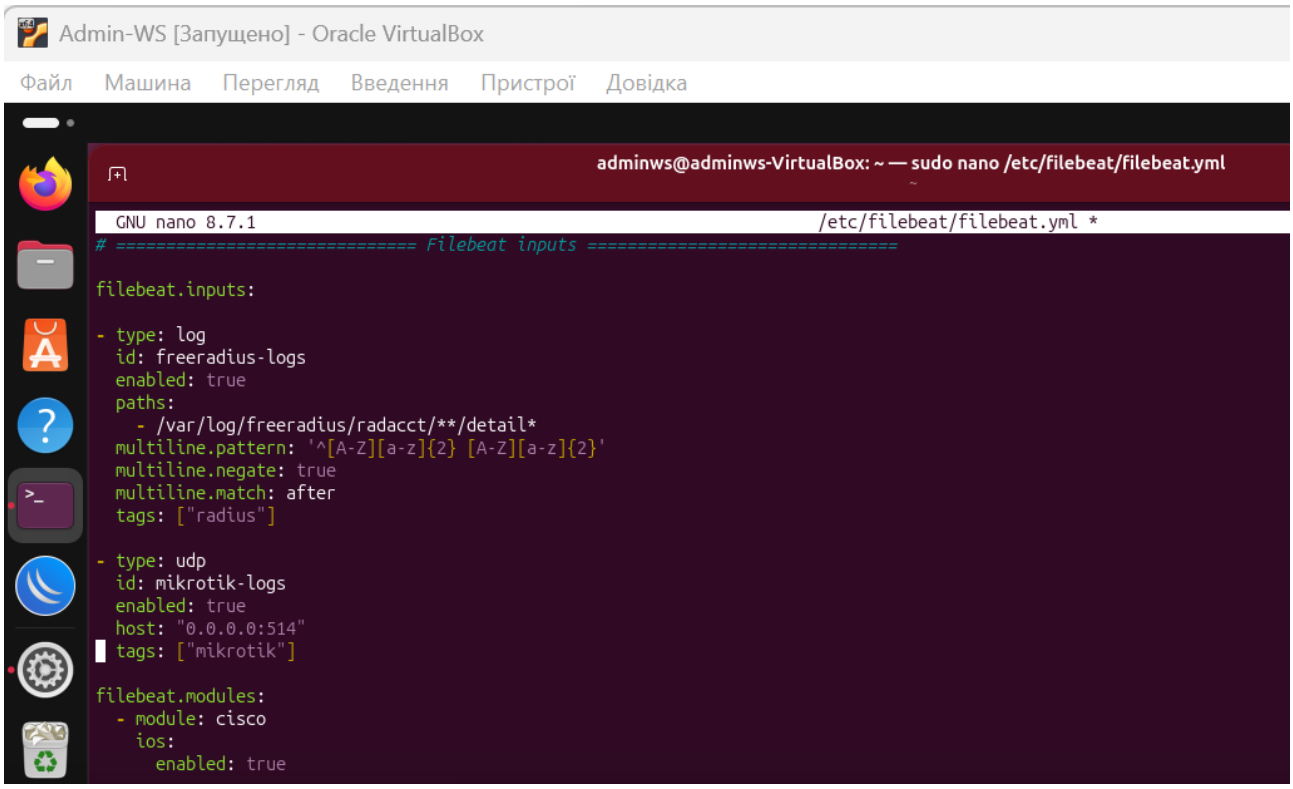


Рисунок 3.26 – Конфігурація вхідних потоків логів

На пристроях Cisco активували передачу подій інформаційного рівня, увімкнули мітки часу для синхронізації та додали ідентифікацію пристроїв за їхніми системними іменами. В свою чергу, на обладнанні MikroTik створили нову дію логування, після чого перенаправили на сервер моніторингу всі ключові категорії подій, включаючи детальний облік дій користувачів, критичні збої, системні помилки, попередження та інформаційні повідомлення.

Налаштування маршрутизатора Mikrotik-DIST:

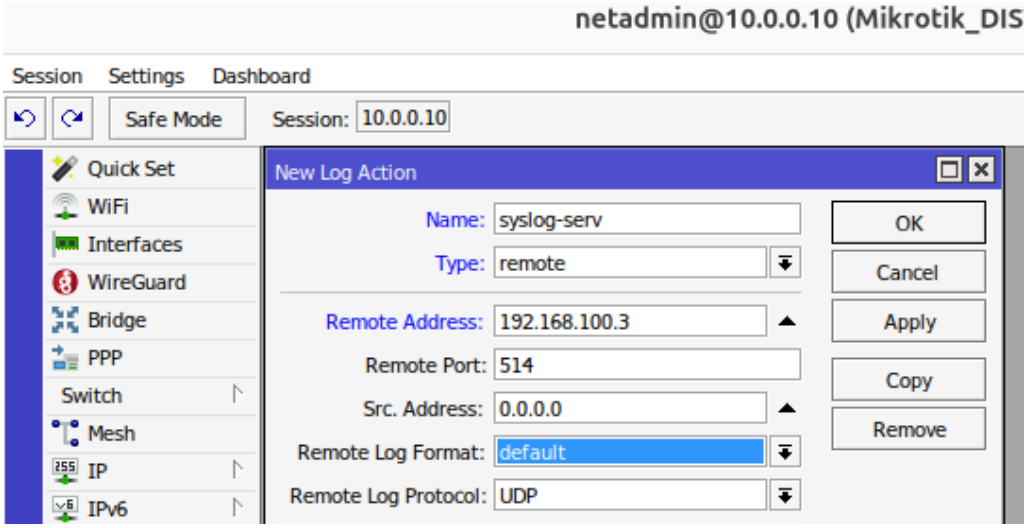


Рисунок 3.27 – Створення віддаленої дії відправки Syslog

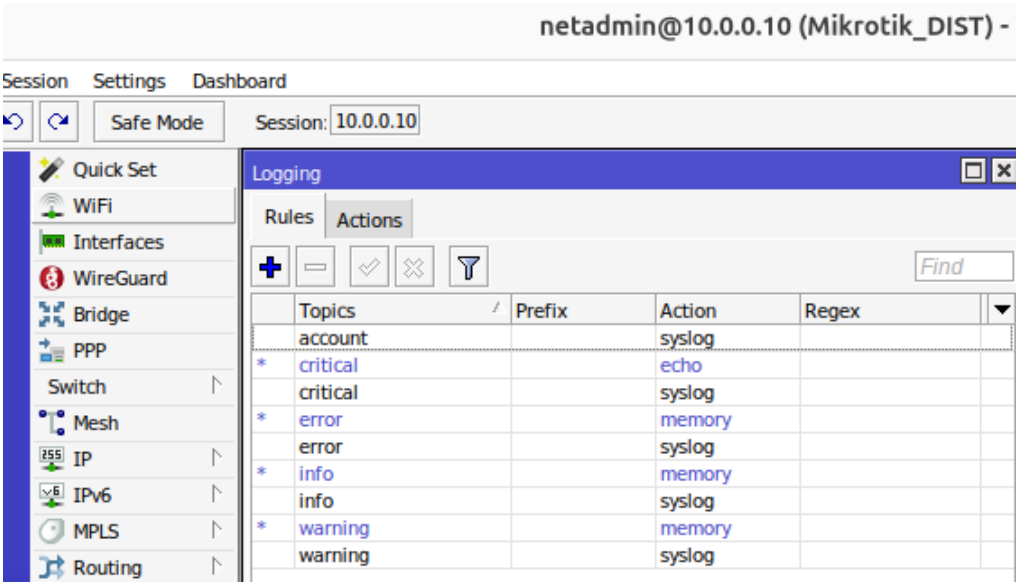


Рисунок 3.28 – Налаштування правил логування та категорій

Лістинг налаштувань маршрутизатора Cisco-CORE:

```
Cisco-CORE(config)#logging host 192.168.100.3 transport udp port 9003
Cisco-CORE(config)#logging trap informational
Cisco-CORE(config)#service timestamps log datetime msec
Cisco-CORE(config)#service timestamps debug datetime msec
Cisco-CORE(config)#logging origin-id hostname
Cisco-CORE(config)#exit
```

Для захисту канального рівня впроваджено комплекс заходів безпеки та контролю доступу. На абонентських інтерфейсах обох платформ активовано функцію BPDU Guard у поєднанні з технологією STP PortFast на Cisco для прискорення конвергенції та блокування портів у разі підключення стороннього комутаційного обладнання. Захист від несанкціонованих DHCP-серверів та атак типу ARP-spoofing реалізовано через механізми DHCP Snooping (із визначенням магістральних портів як довірених) та інструменти контролю адрес: на Cisco розгорнуто Dynamic ARP Inspection (DAI), а на Mikrotik — режим reply-only на клієнтських інтерфейсах у зв'язці з опцією Add ARP For Leases на DHCP-сервері. Додатково, на комутаторах Cisco безпеку кінцевих точок підсилено функцією Port Security з динамічним вивченням MAC-адрес і режимом блокування restrict, а всі незадіяні порти на пристроях переведено в стан shutdown. На комутаторах Mikrotik базову комутацію та роботу зазначених функцій безпеки організовано в межах створеного програмного мосту (Bridge), що об'єднує всі активні фізичні інтерфейси. Приклади налаштувань пристроїв наведено нижче, всі інші налаштовуються аналогічно:

Лістинг налаштувань комутатора Cisco-SW-LAN-B:

```
Cisco-SW-LAN-B(config)#interface range eth0/1-3
Cisco-SW-LAN-B(config-if-range)#switchport mode access
Cisco-SW-LAN-B(config-if-range)#switchport port-security
Cisco-SW-LAN-B(config-if-range)#switchport port-security maximum 2
Cisco-SW-LAN-B(config-if-range)#switchport port-security violation restrict
Cisco-SW-LAN-B(config-if-range)#switchport port-security mac-address sticky
Cisco-SW-LAN-B(config-if-range)#exit
Cisco-SW-LAN-B(config)#interface range eth1/0-3,eth2/0-3,eth3/0-3
Cisco-SW-LAN-B(config-if-range)#shutdown
Cisco-SW-LAN-B(config-if-range)#exit
Cisco-SW-LAN-B(config)#ip dhcp snooping
Cisco-SW-LAN-A(config)#ip dhcp snooping vlan 1
Cisco-SW-LAN-A(config)#ip arp inspection vlan 1
Cisco-SW-LAN-A(config)#no ip dhcp snooping information option
Cisco-SW-LAN-A(config)#interface range eth0/0-3
Cisco-SW-LAN-A(config-if-range)#ip dhcp snooping limit rate 256
Cisco-SW-LAN-A(config-if-range)#exit
Cisco-SW-LAN-A(config)#int eth0/0
Cisco-SW-LAN-A(config-if)#ip arp inspection trust
Cisco-SW-LAN-A(config-if)#ip dhcp snooping trust
Cisco-SW-LAN-A(config-if)#exit
Cisco-SW-LAN-A(config)#errdisable detect cause dhcp-rate-limit
Cisco-SW-LAN-A(config)#errdisable recovery cause dhcp-rate-limit
Cisco-SW-LAN-A(config)#errdisable recovery interval 120
Cisco-SW-LAN-B(config)#interface range eth0/1-3
Cisco-SW-LAN-B(config-if-range)#spanning-tree portfast
Cisco-SW-LAN-B(config-if-range)#spanning-tree bpduguard enable
Cisco-SW-LAN-B(config-if-range)#exit
```

Налаштування комутатора Mikrotik-SW-LAN-C:

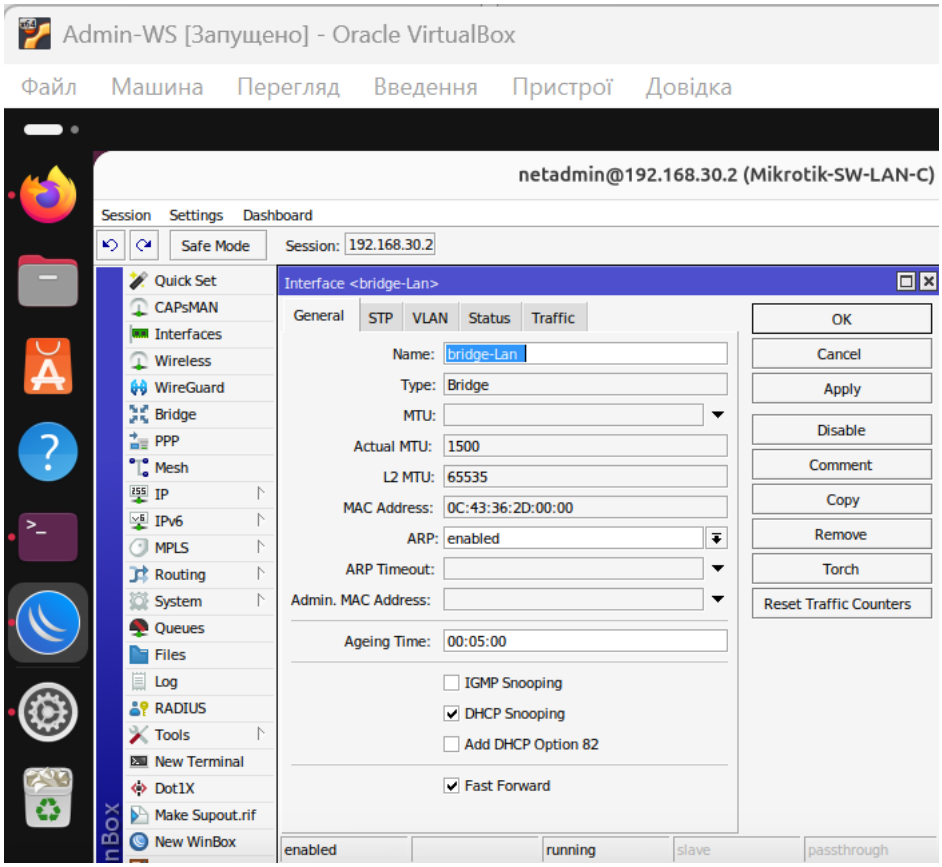


Рисунок 3.29 – Налаштування мосту

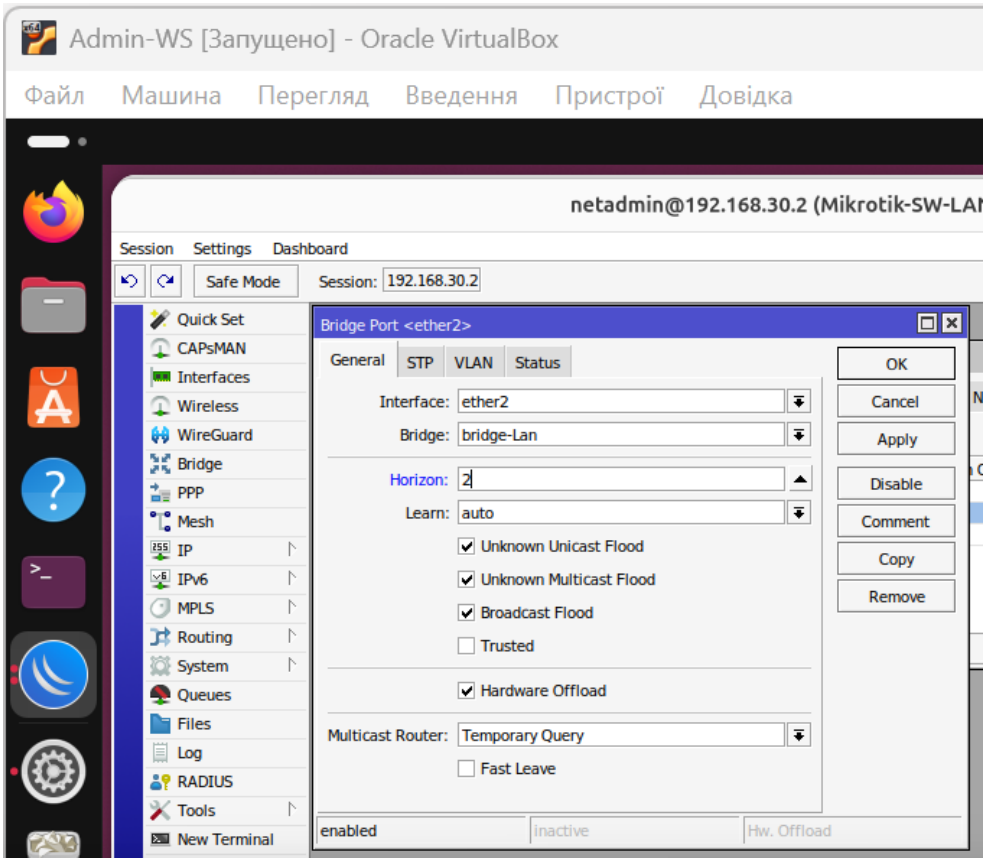


Рисунок 3.30 – Налаштування довіреного порту

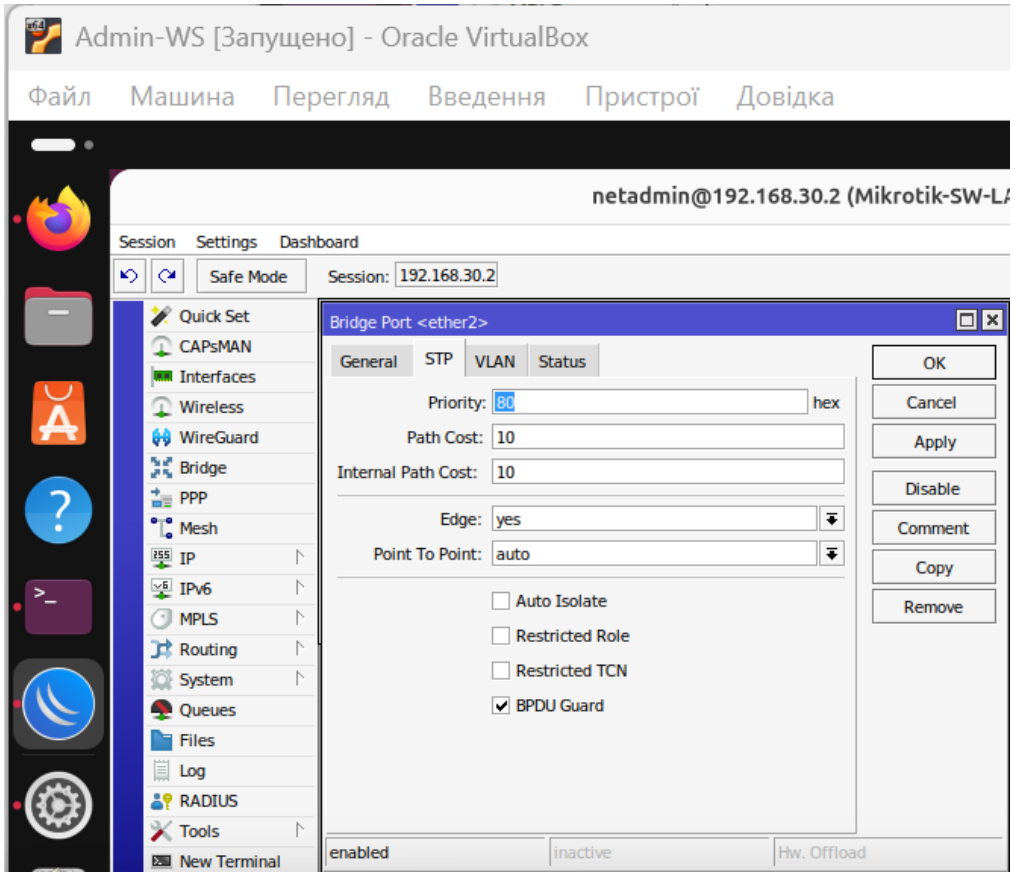


Рисунок 3.31 – Налаштування BPDU Guard на одному із інтефейсів

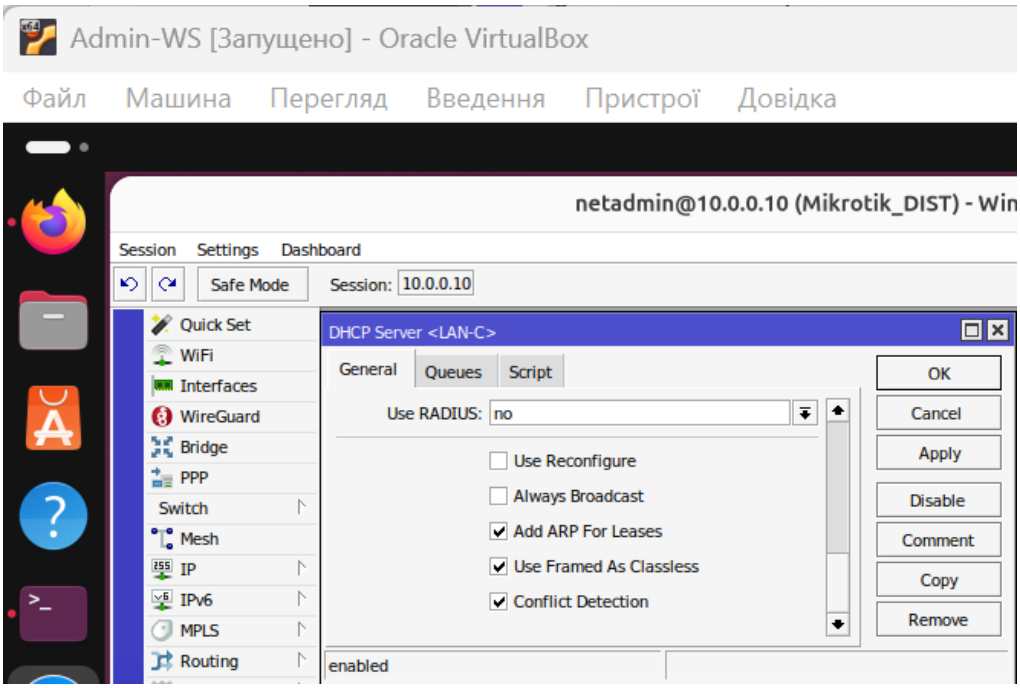


Рисунок 3.32 – Налаштування Add ARP For Leases на DHCP-сервері

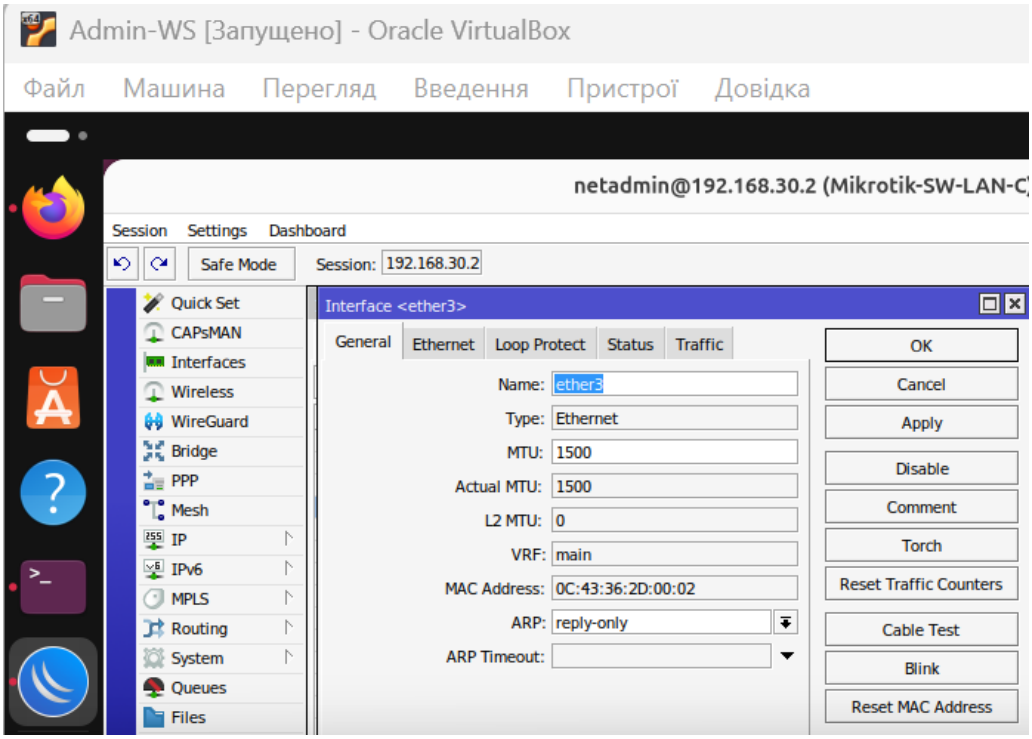


Рисунок 3.33 – Ввімкнення режиму reply-only на одному із інтефейсів

Для захисту серверної зони на інтерфейсі маршрутизатора Cisco-CORE застосовано список контролю доступу. Він блокує первинні підключення ззовні, але дозволяє серверам ініціювати вихідні TCP-сесії й отримувати зворотні пакети. Також дозволено лише вхідні UDP-відповіді від зовнішніх серверів DNS та NTP, а також ICMP-пакети типу echo-reply для діагностики. Вхідний технологічний трафік

обмежено виключно прийомом Syslog на UDP-порти 514 і 9003 та запитами автентифікації до RADIUS-сервера на порти 1812 і 1813, тоді як усі інші види активності примусово відкидаються.

Лістинг налаштувань:

```
Cisco-CORE(config)#ip access-list extended ACL-SERV
Cisco-CORE(config-ext-nacl)#permit tcp any 192.168.100.0 0.0.0.7 established
Cisco-CORE(config-ext-nacl)#permit udp any eq 53 192.168.100.0 0.0.0.7
Cisco-CORE(config-ext-nacl)#permit udp any eq 123 192.168.100.0 0.0.0.7
Cisco-CORE(config-ext-nacl)#permit udp any 192.168.100.0 0.0.0.7 eq 514
Cisco-CORE(config-ext-nacl)#permit udp any 192.168.100.0 0.0.0.7 eq 9003
Cisco-CORE(config-ext-nacl)#permit udp any 192.168.100.0 0.0.0.7 eq 1812
Cisco-CORE(config-ext-nacl)#permit udp any 192.168.100.0 0.0.0.7 eq 1813
Cisco-CORE(config-ext-nacl)#permit icmp any 192.168.100.0 0.0.0.7 echo-reply
Cisco-CORE(config-ext-nacl)#deny ip any any log
Cisco-CORE(config-ext-nacl)#exit
Cisco-CORE(config)#int gi0/3
Cisco-CORE(config-if)#ip access-group ACL-SERV out
Cisco-CORE(config-if)#exit
```

На периметрі мережі впроваджено правила фільтрації для ланцюжків input та forward. Захист маршрутизатора й внутрішньої мережі базується на відстеженні стану з'єднань: правила дозволяють легітимний зворотний трафік (established, related) та скидають пошкоджені пакети (invalid). З боку зовнішнього інтерфейсу повністю заблоковано будь-які вхідні запити до самого роутера, окрім ICMP-трафіку для діагностики, а в ланцюжку транзитного трафіку примусово блокуються всі нові вхідні з'єднання з інтернету, для яких відсутні правила прокидання портів.

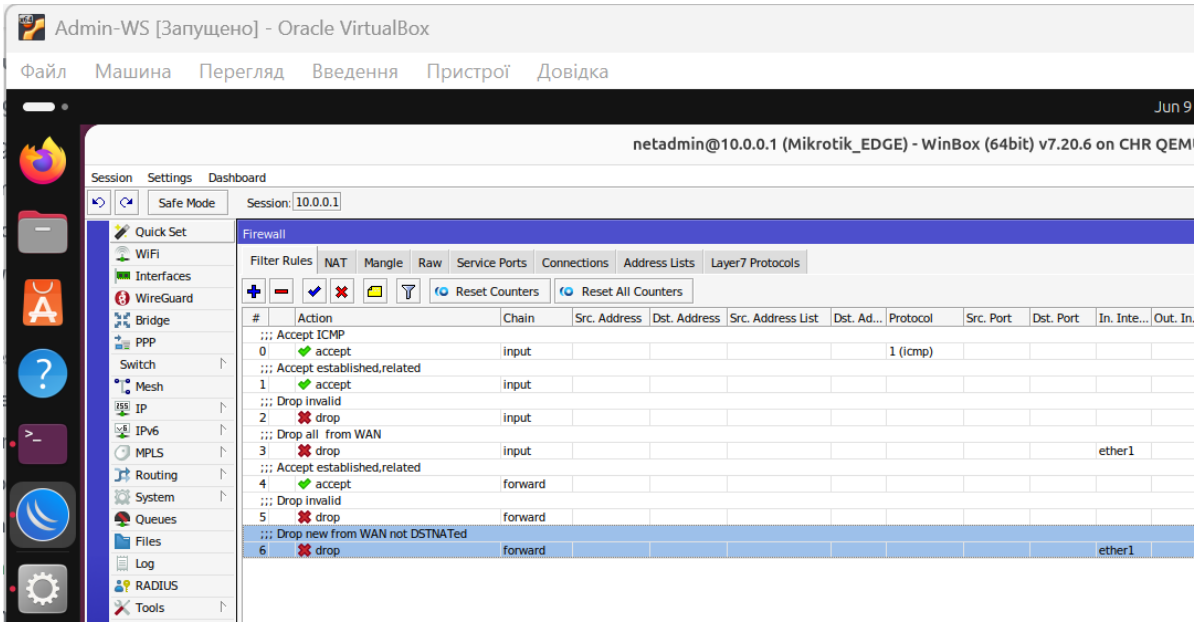


Рисунок 3.34 – Налаштовані правила

У вкладці NAT налаштовано два правила трансляції адрес. Перше правило виконує прокидання 80-го порту (TCP) зі зовнішнього інтерфейсу на внутрішній веб-сервер (розгорнутий для тестування засобів безпеки), а друге — стандартне маскування вихідного трафіку для доступу локальної мережі до інтернету.

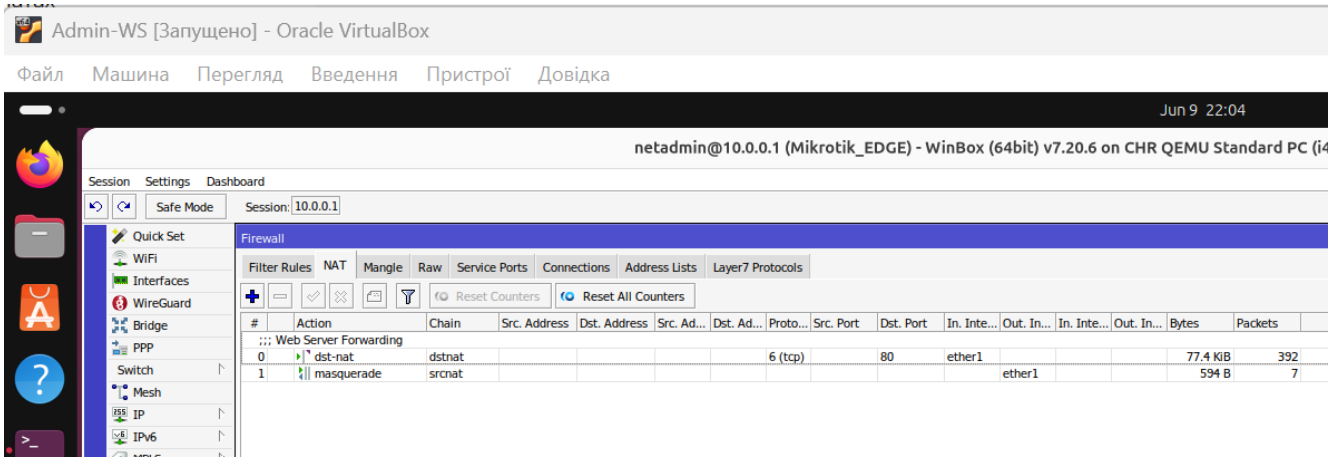


Рисунок 3.35 – Налаштування NAT

3.2 Перевірка роботи та тестування безпеки мережі

Для перевірки правильності налаштування мережі було протестовано зв'язок між усіма пристроями. Перевірка таблиць маршрутизації та протоколу OSPF на обладнанні Cisco та MikroTik показала, що пристрої успішно бачать один одного і працюють без втрати даних чи затримок.

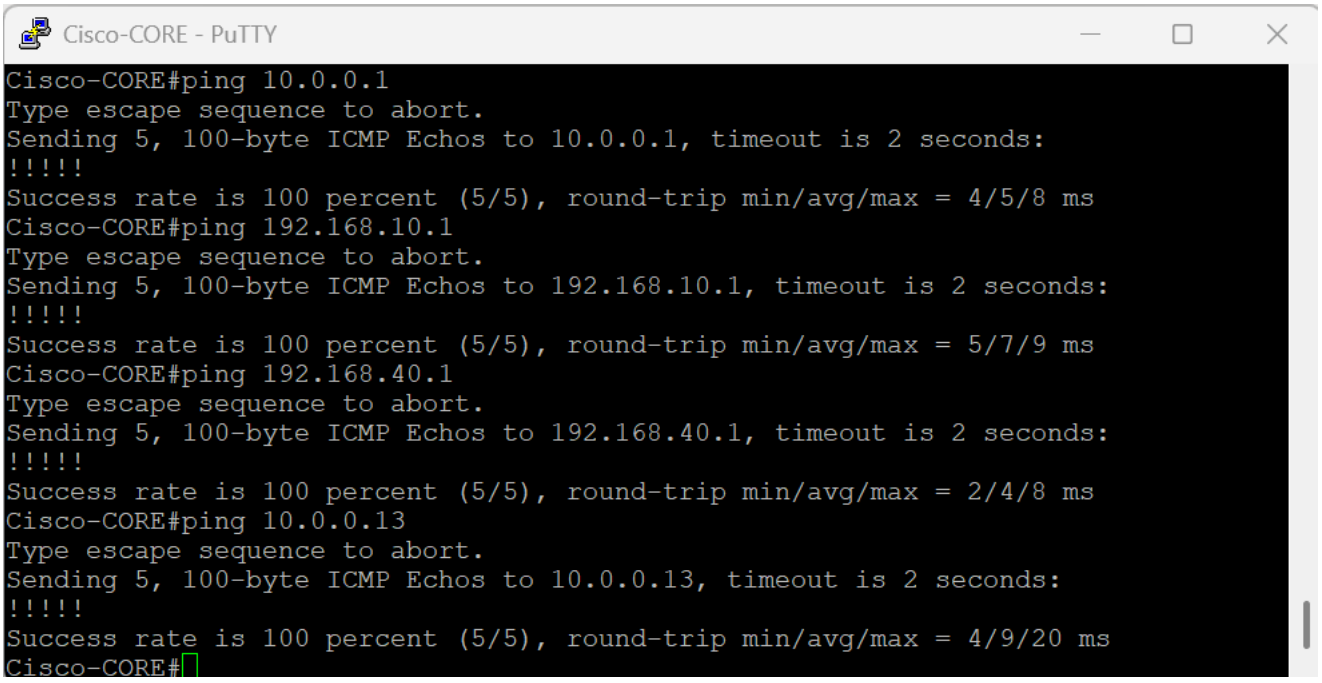


Рисунок 3.36 – Перевірка зв’язку

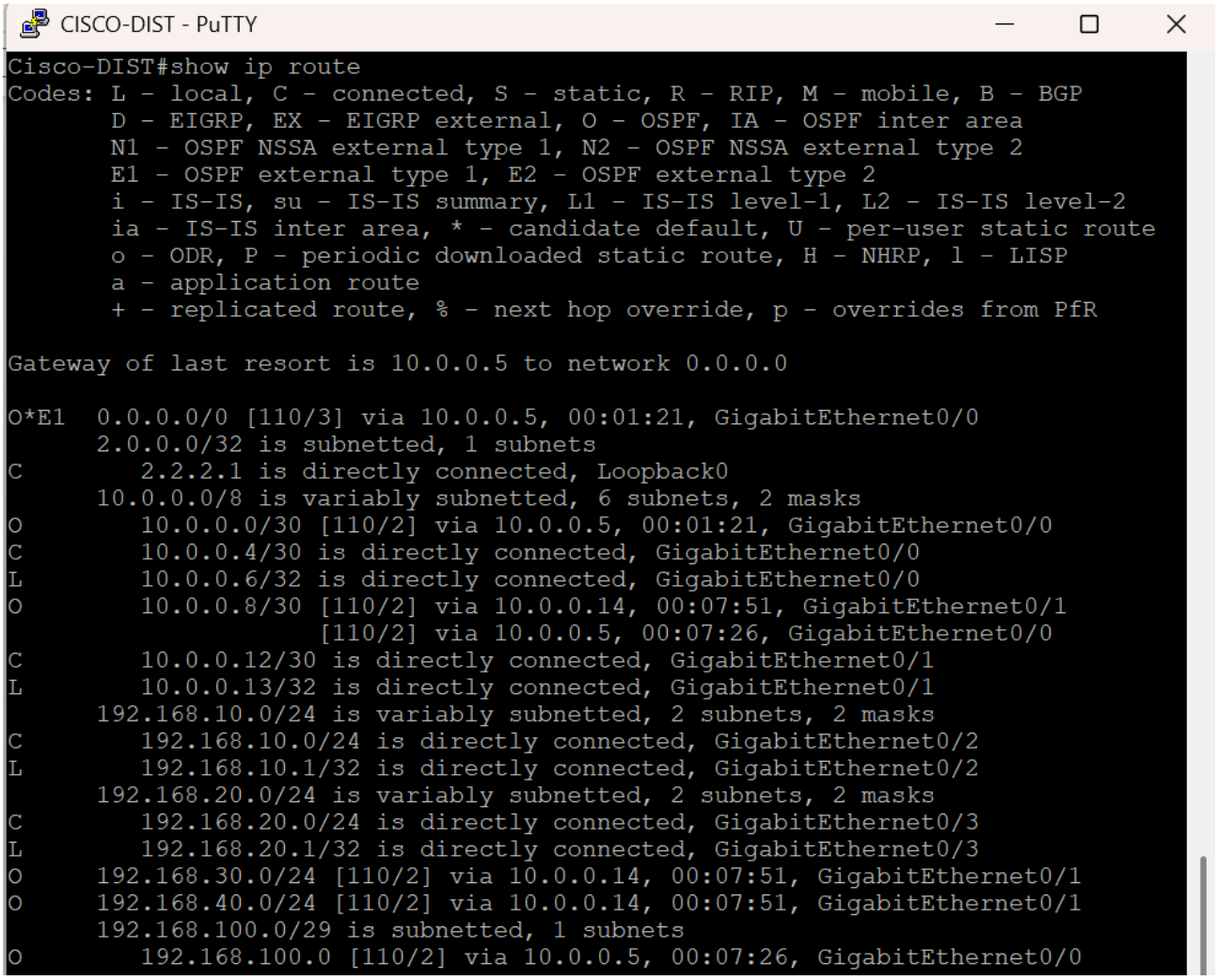


Рисунок 3.37 – Перевірка таблиці маршрутизації на пристрої Cisco-DIST

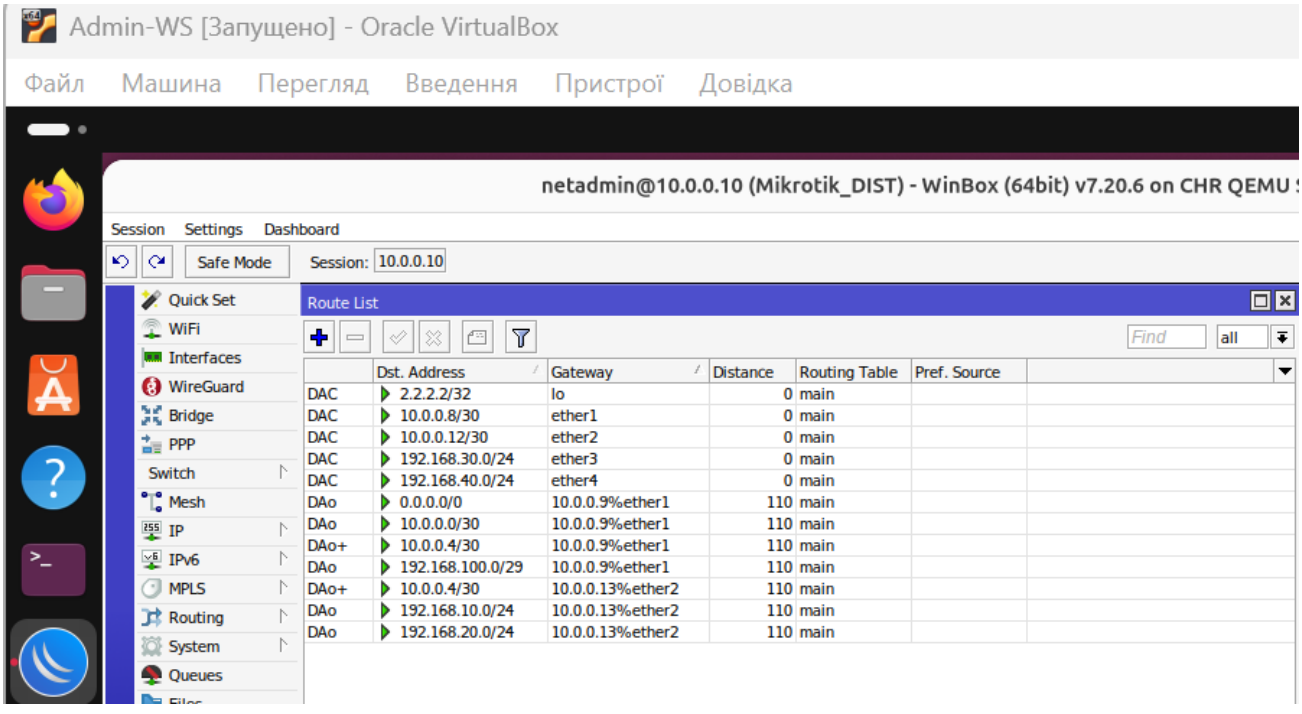


Рисунок 3.38 – Перевірка таблиці маршрутизації на пристрої Mikrotik-DIST

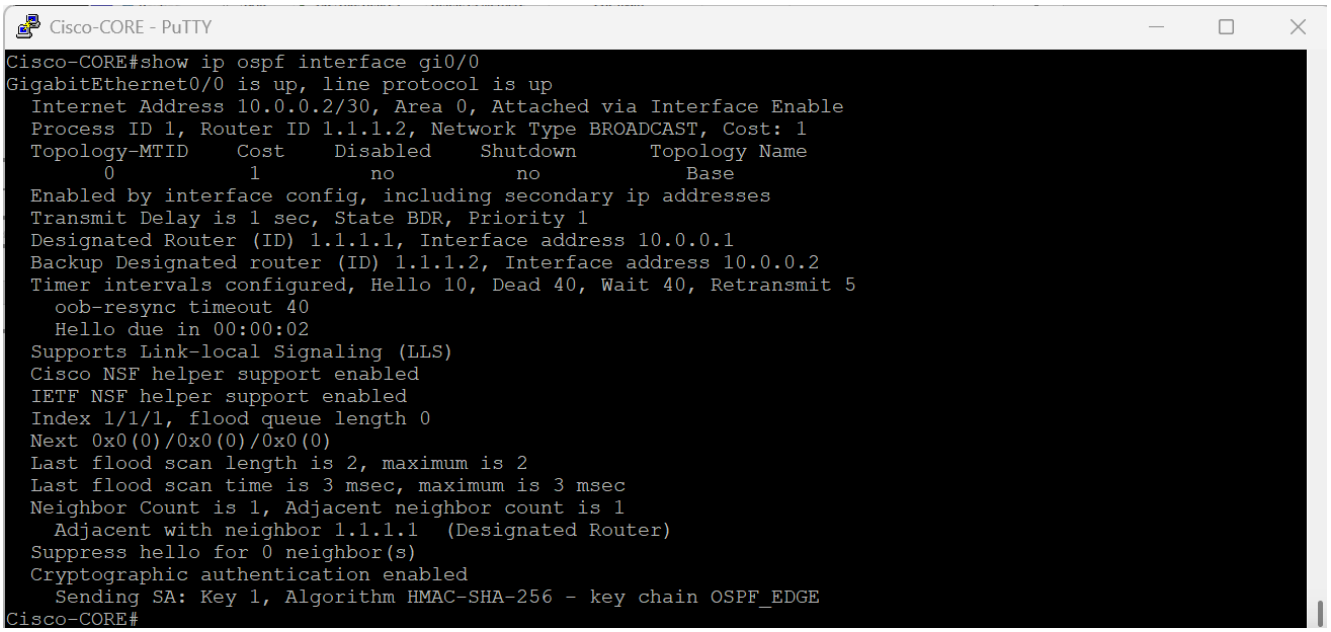


Рисунок 3.39 – Перевірка налаштування автентифікації OSPF на Cisco-CORE

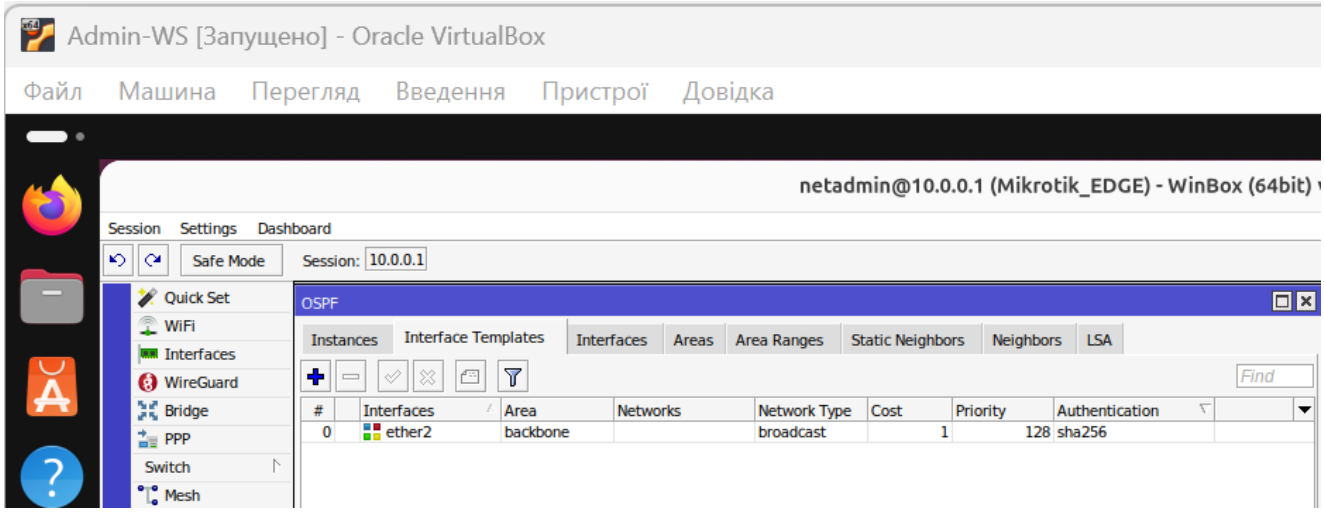


Рисунок 3.40 – Перевірка налаштування автентифікації OSPF на Mikrotik_EDGE

Тестування служби DHCP на кінцевих пристроях підтвердило автоматичне та коректне отримання ними IP-адрес. Перевірка таблиці прив'язок підтвердили коректний розподіл адрес та відсутність конфліктів у мережі.

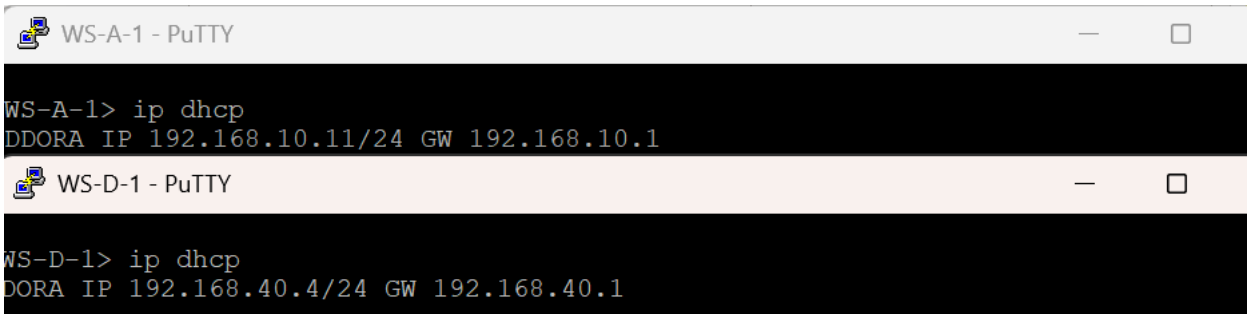


Рисунок 3.41 – Отримання адреси на кінцевих пристроях

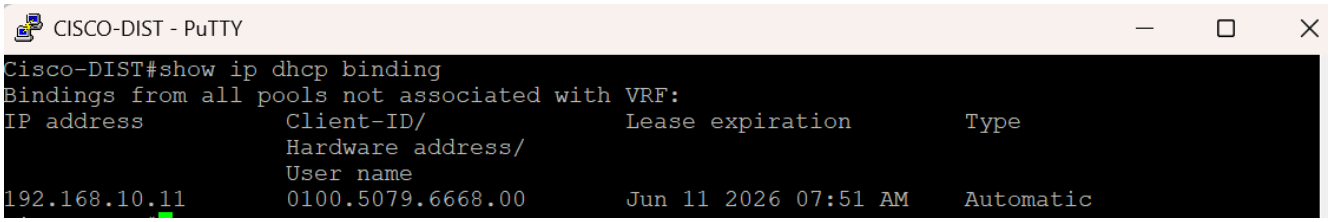


Рисунок 3.42 – Таблиця прив’язок на пристрої Cisco-DIST

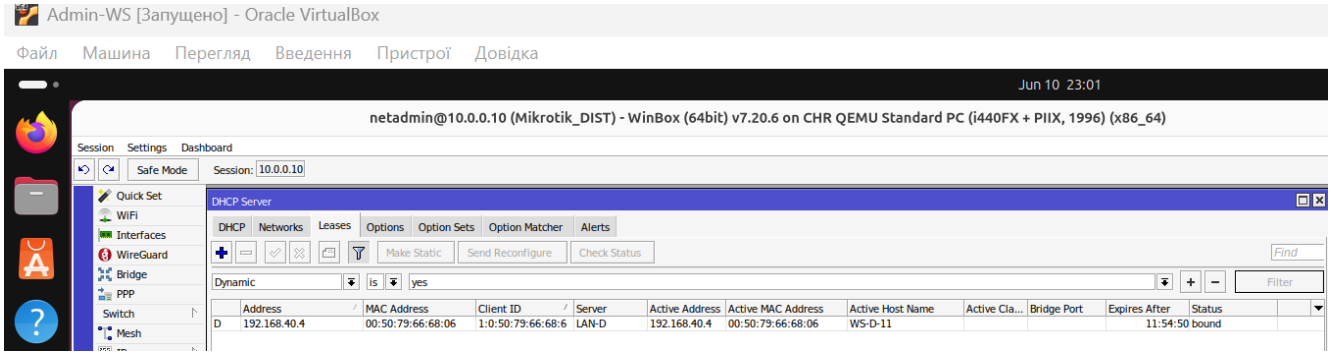


Рисунок 3.43 – Таблиця прив’язок на пристрої Mikrotik-DIST

Перевірка системи централізованої автентифікації підтвердила успішний захист доступу до мережного обладнання. У логах сервера зафіксовано точний час сесій та успішні спроби авторизації.

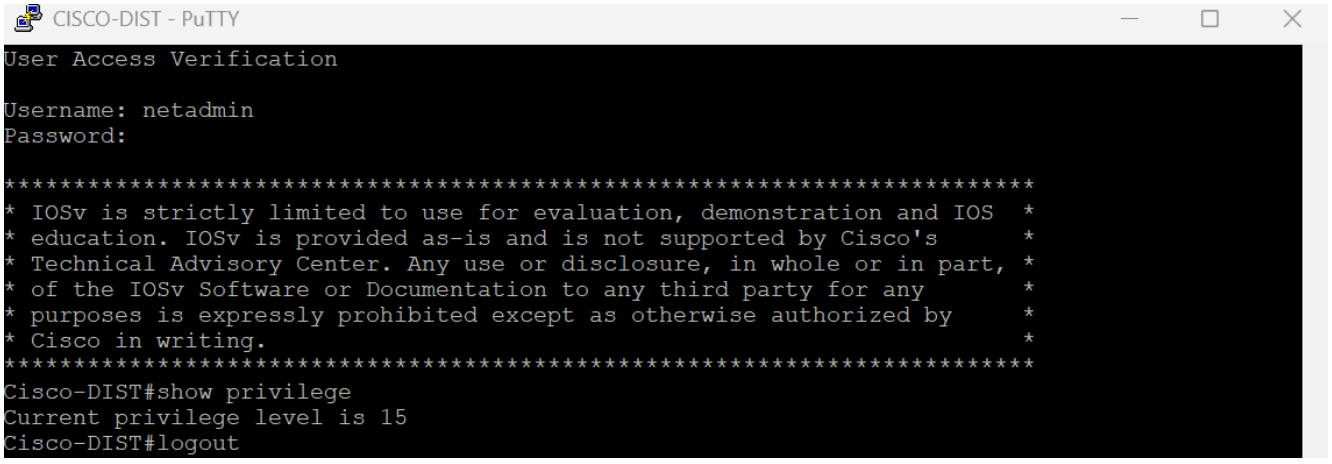


Рисунок 3.44 – Вхід під користувачем netadmin

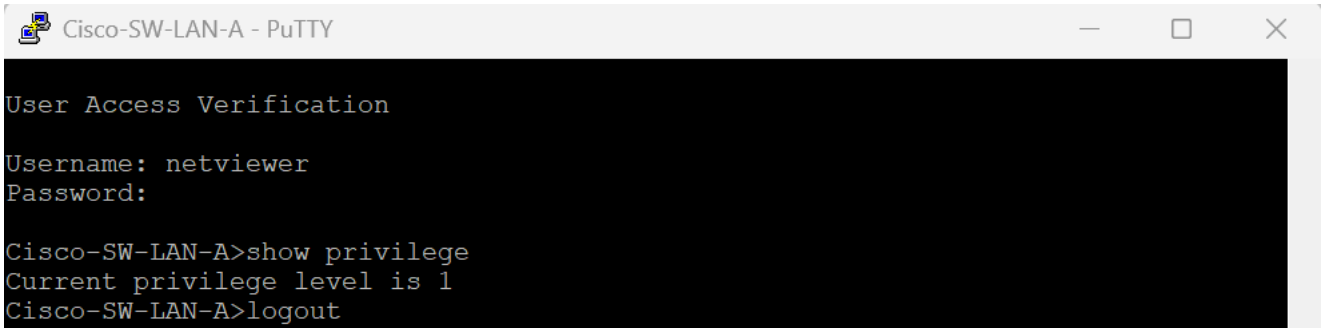


Рисунок 3.45 – Вхід під користувачем netviewer

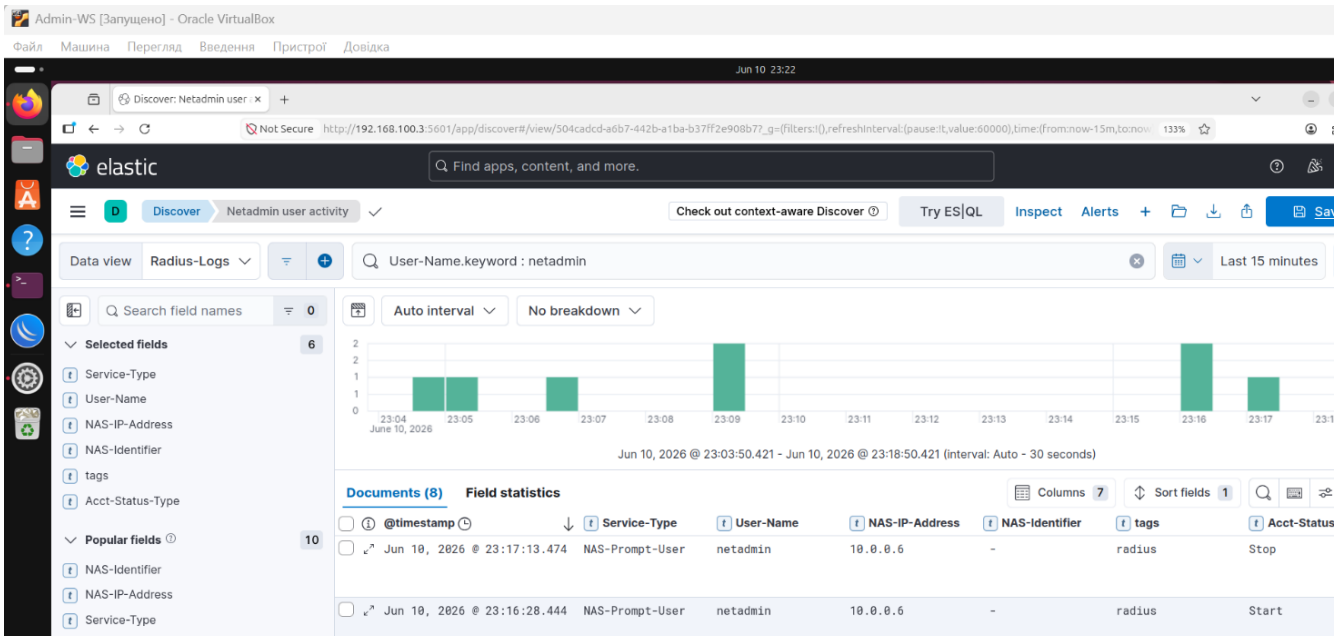


Рисунок 3.46 – Логування активності користувача netadmin

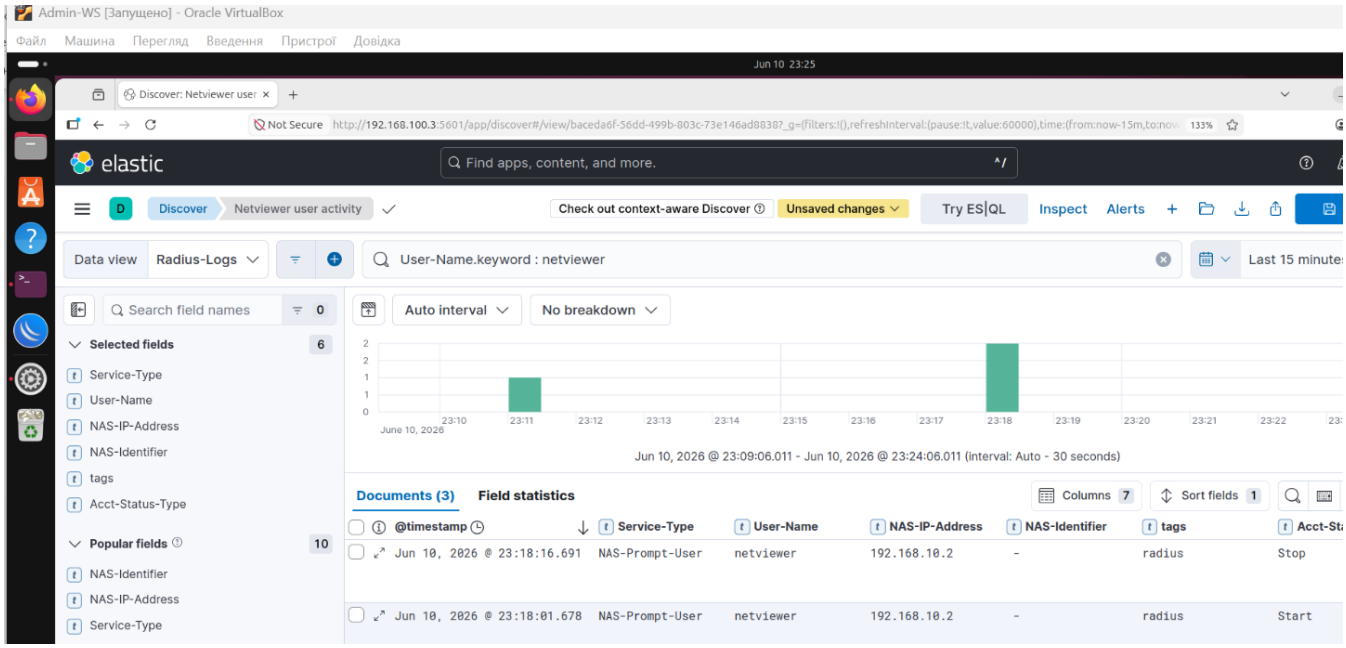


Рисунок 3.47 – Логування активності користувача netviewer

Перевірка списку контролю доступу на маршрутизаторі Cisco-CORE підтвердила точну роботу правил безпеки. Системний трафік автентифікації і повідомлення Syslog безперешкодно проходять через фільтр. Подальше тестування також продемонструвало коректну роботу заданої конфігурації.

З робочого місця адміністратора вдається перевірити зв'язок із мережними пристроями. Водночас спроба надіслати запити тестування зв'язку до серверної мережі завершилася невдачею.

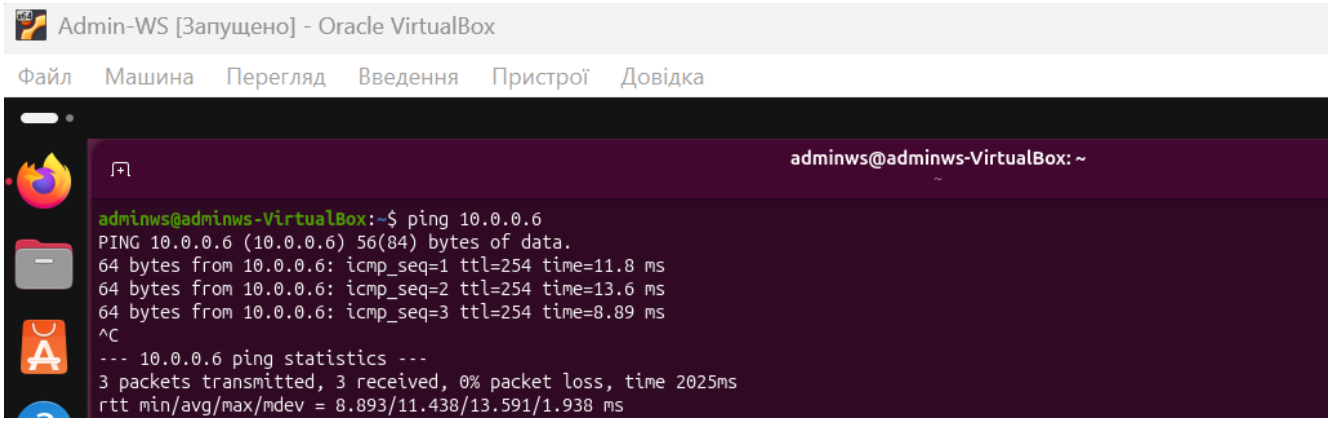


Рисунок 3.48 – Пінг пристроїв з серверної мережі

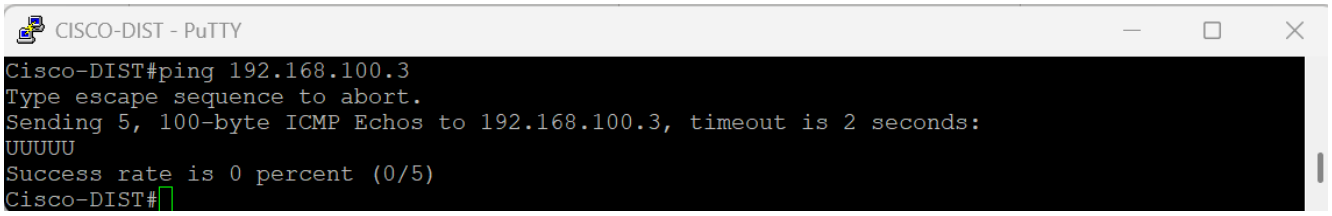


Рисунок 3.49 – Спроба пінгу до серверної мережі

Блокування несанкціонованого доступу також підтверджено під час спроби встановити SSH-сесію до комутатора серверної зони, яка відхиляється системою. При цьому SSH-підключення з робочої станції адміністратора проходить успішно.

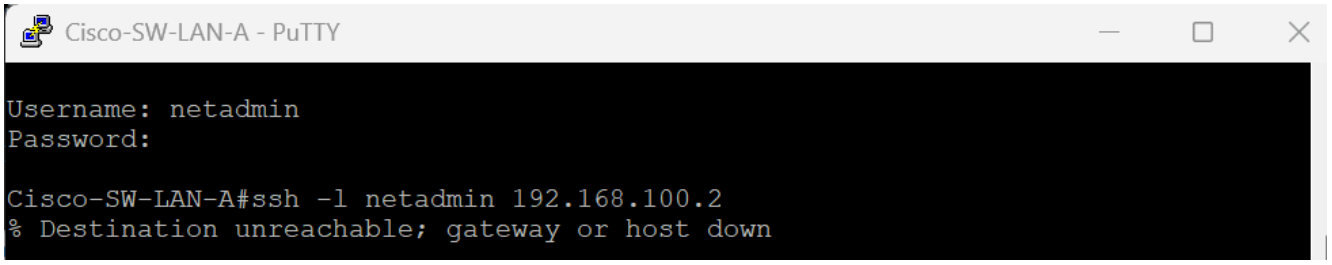


Рисунок 3.50 – Спроба підключення по SSH до комутатора серверної зони

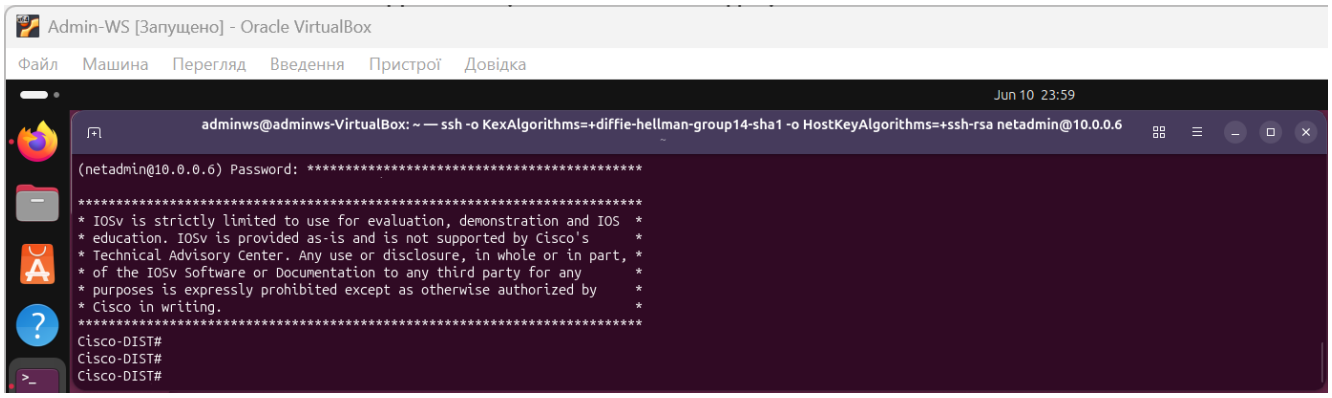


Рисунок 3.51 – Підключення по SSH до одного із мережних пристроїв

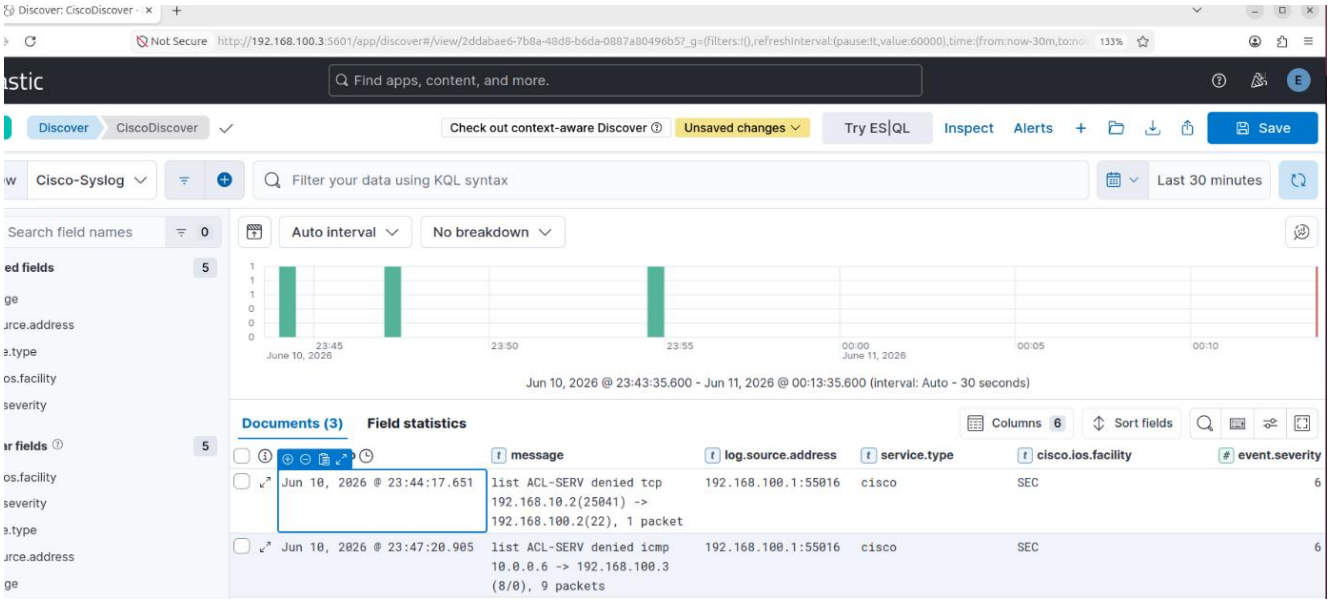


Рисунок 3.52 – Події блокування трафіку у веб-інтерфейсі Kibana

Для оцінки загальної ефективності розгорнутої системи безпеки було проведено тестування інфраструктури методами імітації кібератак.

На першому етапі здійснено спробу просканувати мережу ззовні. За допомогою утиліти Nmap запускається сканування для пошуку відкритих портів та доступних сервісів.

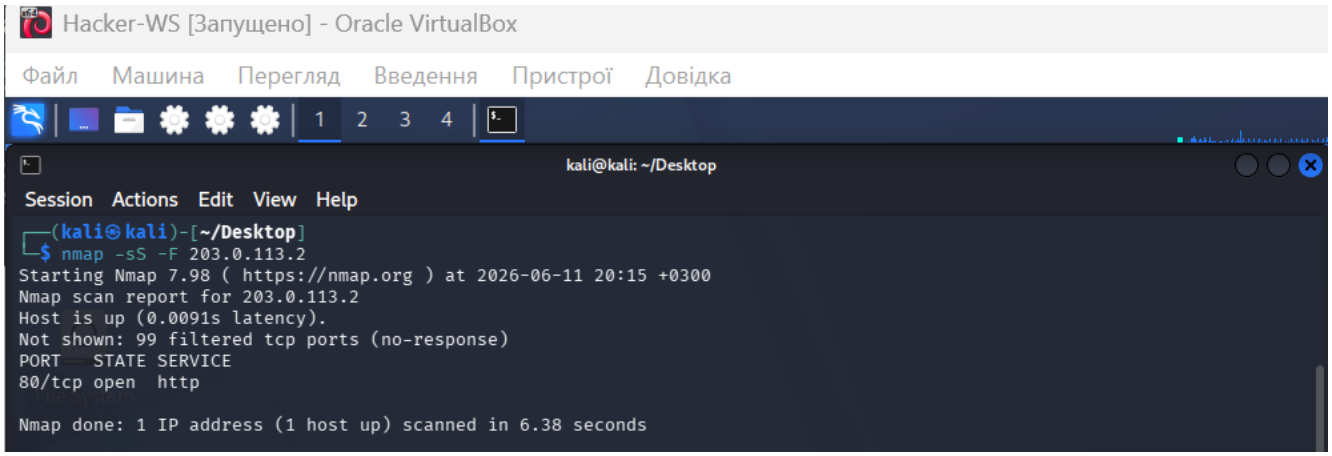


Рисунок 3.53 – Результат сканування портів

Результат сканування показує, що утиліті вдалося зафіксувати лише один відкритий сервіс — вебсервер на 80-му порту (HTTP), який і стане ціллю для наступних етапів тестування. Факт блокування решти розвідувального трафіку фіксується в системі моніторингу Kibana.

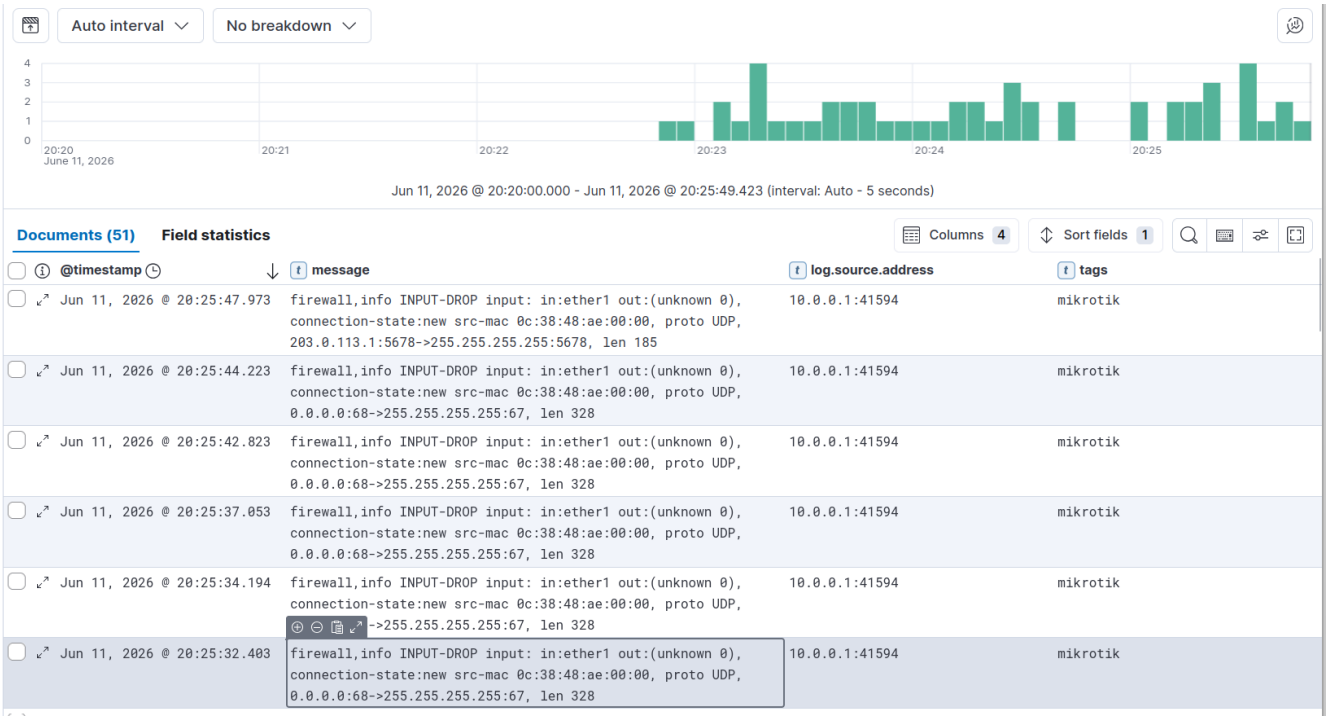


Рисунок 3.54 –Блокування трафіку фаєрволом MikroTik

У внутрішній мережі було розгорнуто тестовий веб-сервер. Як програмне забезпечення було обрано Nginx. Конфігурація мережного екрана дозволяє безперешкодно отримувати доступ до цього ресурсу із зовнішньої мережі.

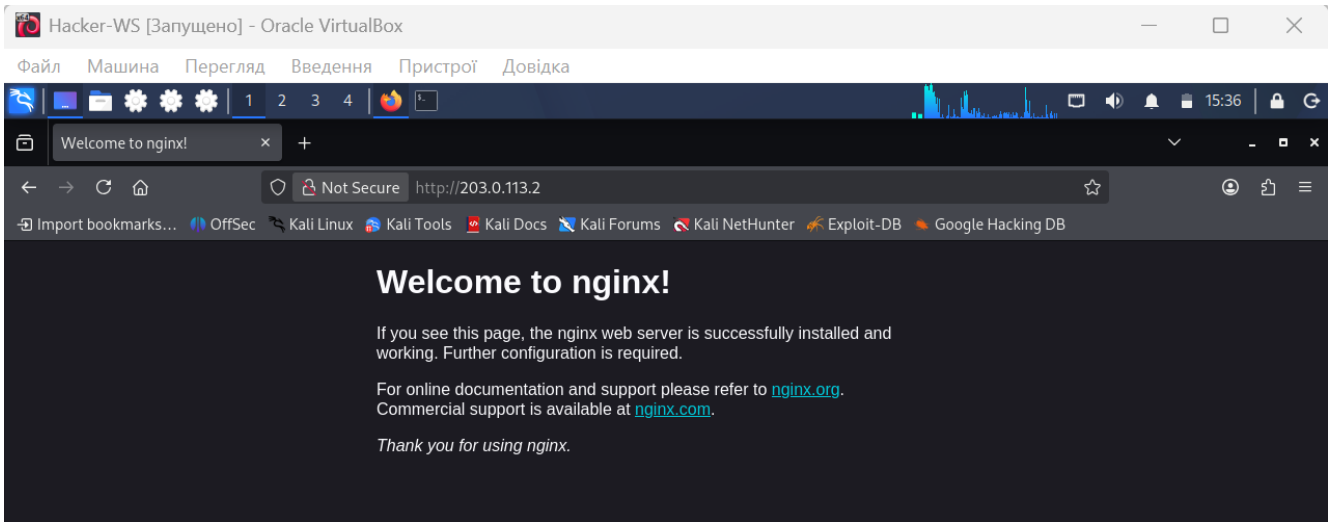


Рисунок 3.55 – Доступ до ресурсу

Після виявлення відкритого порту було змодельовано атаку типу HTTP SYN Flood, спрямовану на відмову вебсервера в обслуговуванні. За допомогою утиліти hping3 із зовнішньої машини було запущено шторм запитів на встановлення TCP-з'єднань. Усі факти атаки були успішно зафіксовані у системі моніторингу Kibana.

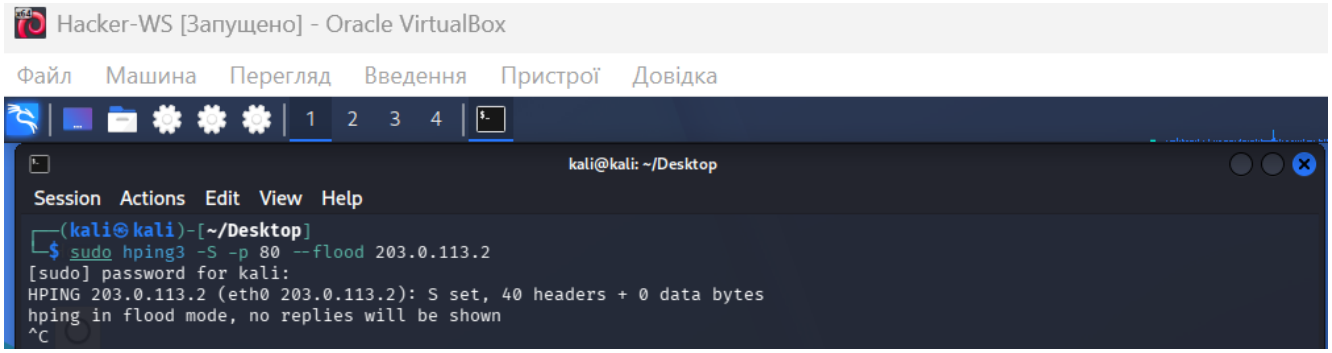


Рисунок 3.56 – Проведення атаки HTTP SYN Flood

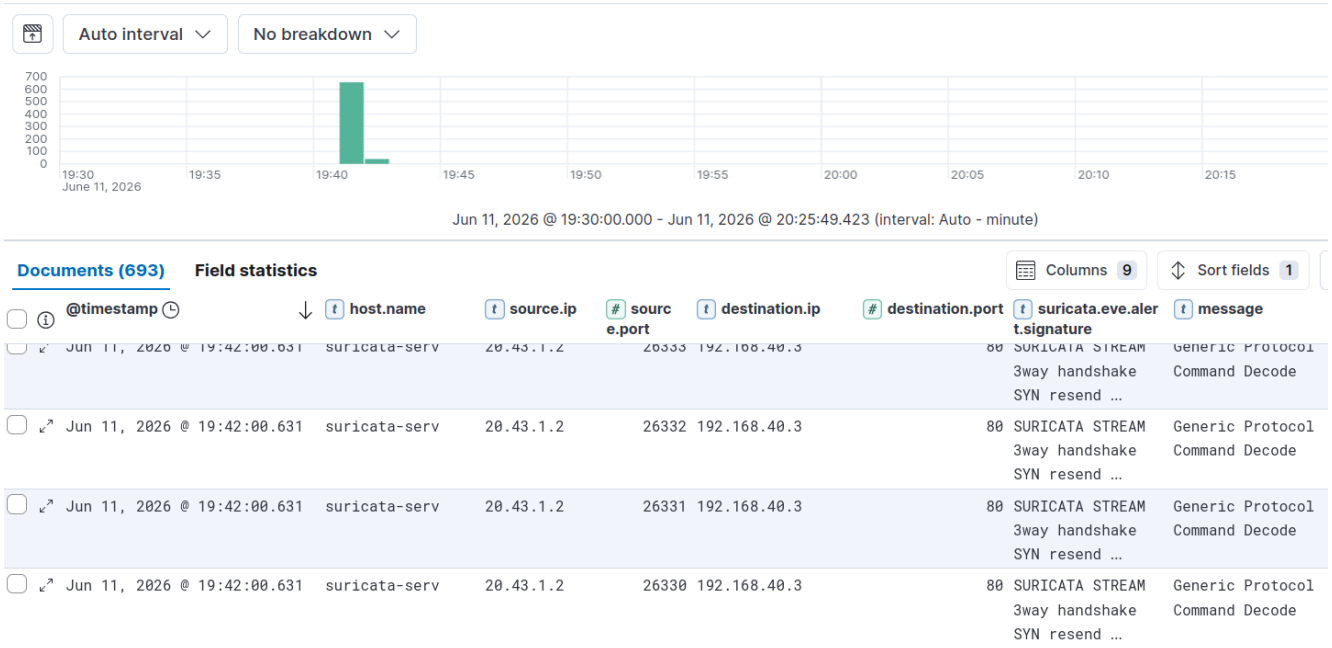


Рисунок 3.57 – Вивід подій Suricata в інтерфейсі Kibana

На наступному етапі було перевірено стійкість системи до атак прикладного рівня. За допомогою утиліти «curl» було надіслано HTTP-запит до вебсервера Nginx із конструкцією UNION SELECT. Цей тест моделював спробу впровадження SQL-ін'єкції для обходу автентифікації та несанкціонованого вилучення конфіденційних даних із бази даних.

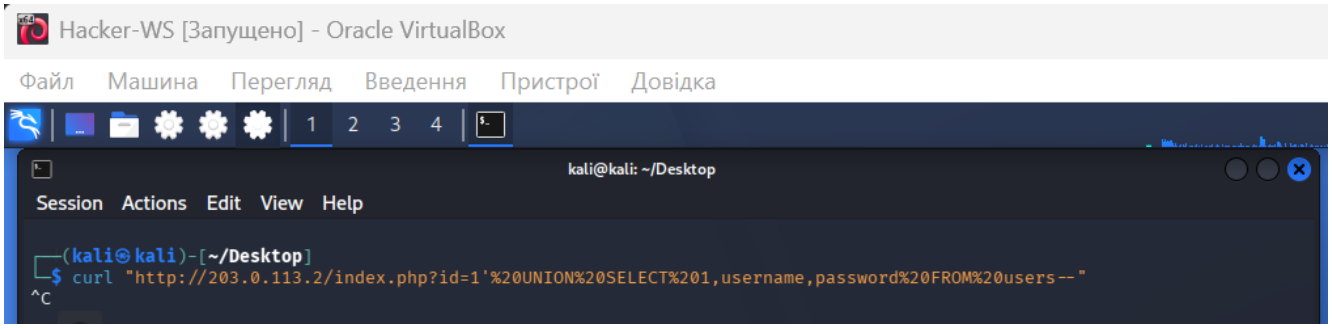


Рисунок 3.58 – SQL-ін'єкція за допомогою утиліти curl

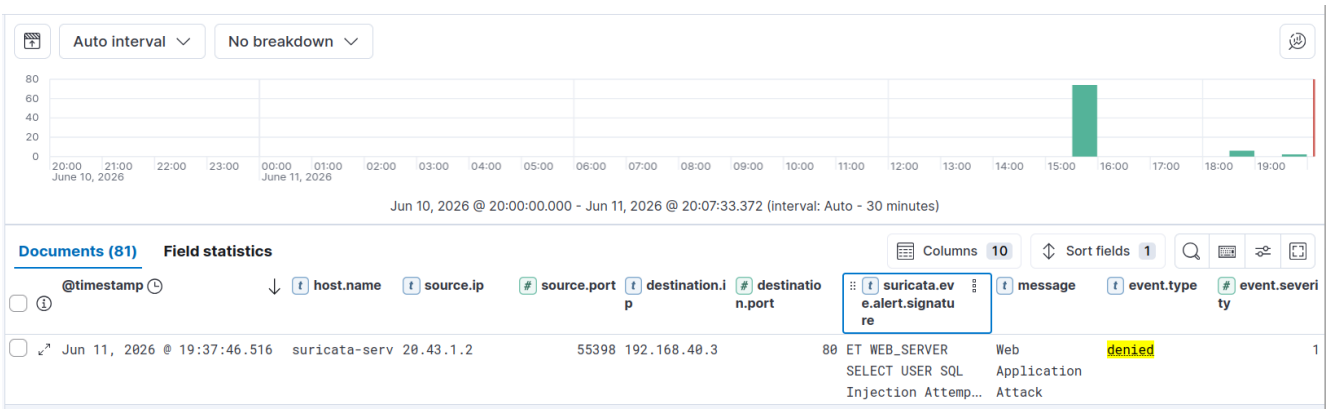


Рисунок 3.59 – Результат виявлення та блокування атаки системою Suricata

Використовуючи утиліту Nikto було виконано автоматизований пошук вразливостей веб-додатка з метою перевірки реакції системи захисту на роботу спеціалізованих сканерів. Усі запити та спроби пошуку були успішно заблоковані системою.

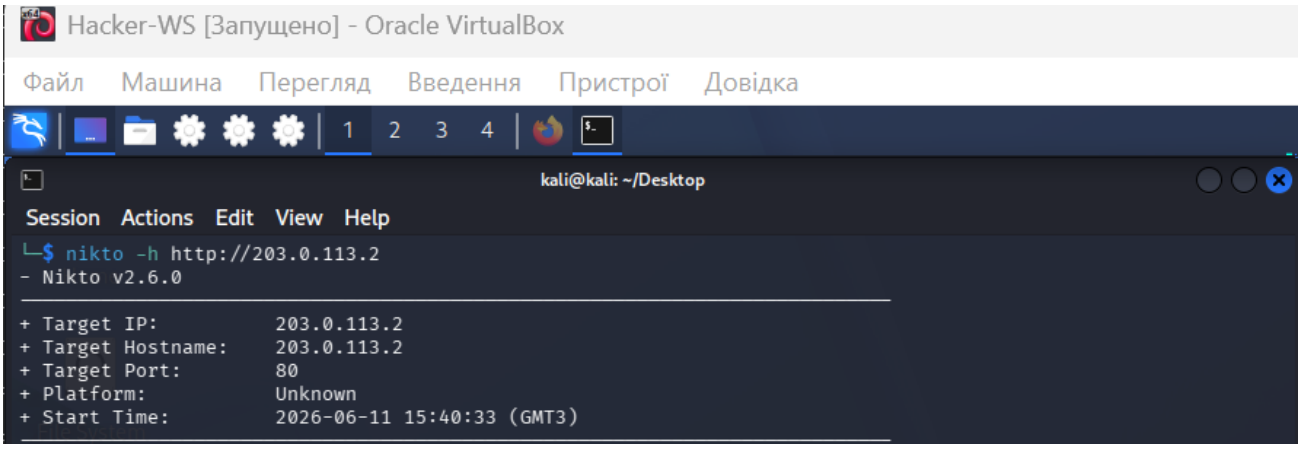


Рисунок 3.60 – Запуск сканування веб-сервера за допомогою утиліти Nikto

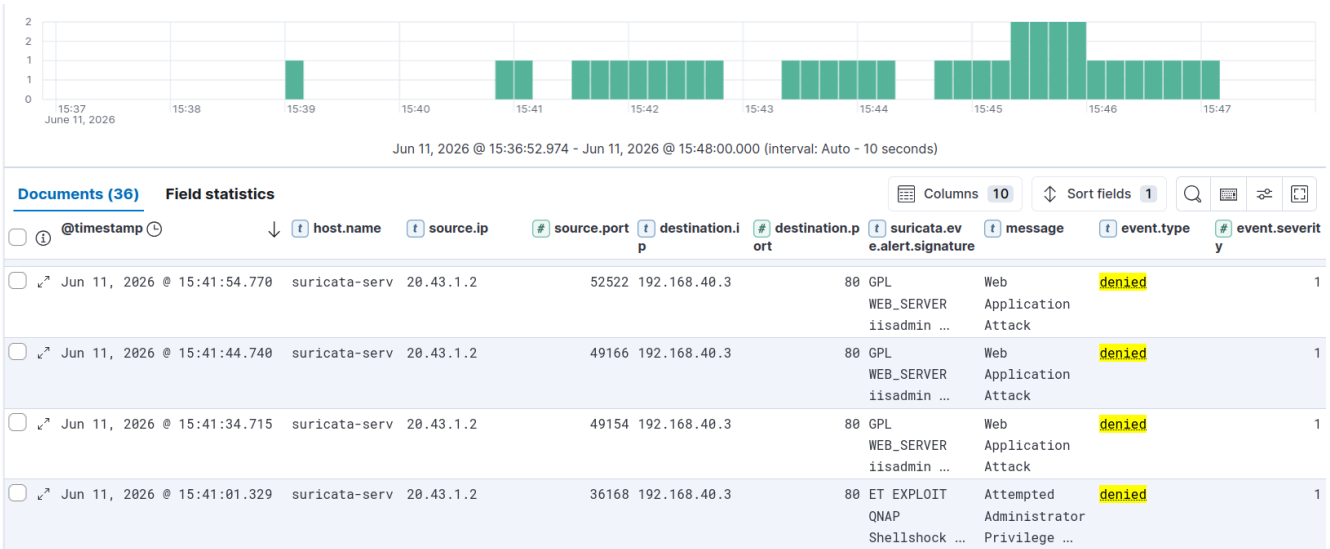


Рисунок 3.61 – Блокування запитів сканера Nikto

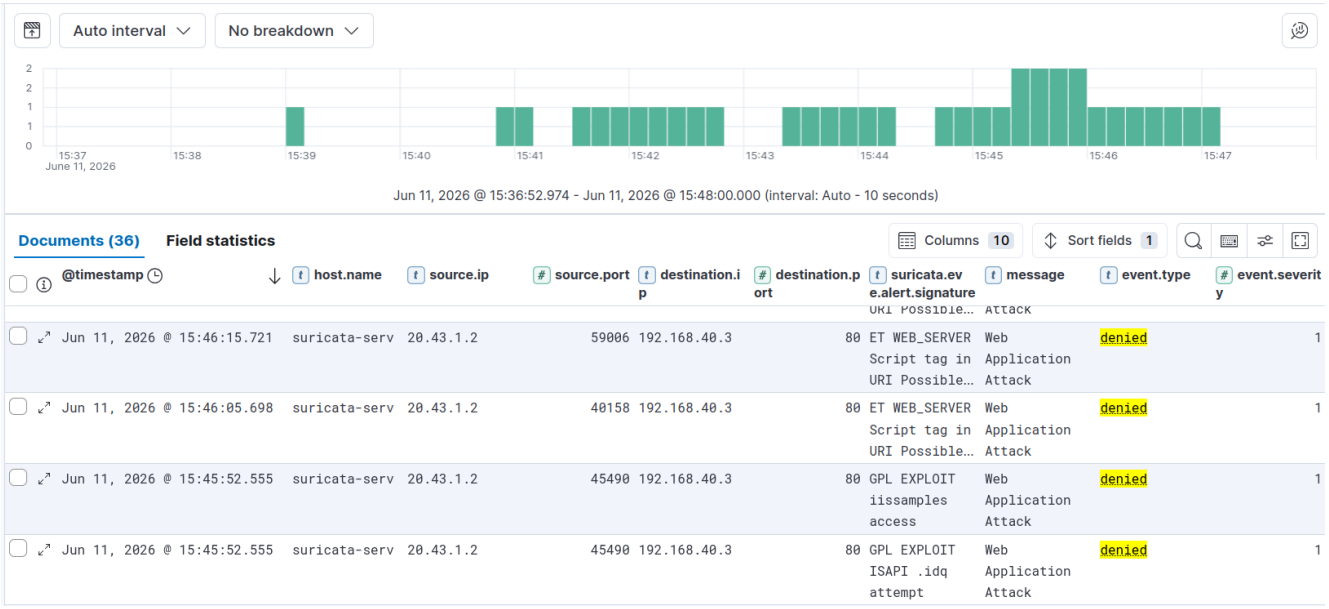


Рисунок 3.62 – Блокування запитів сканера Nikto

В одній із кінцевих підмереж легітимну робочу станцію було замінено на станцію зловмисника, з якої було змодельовано атаку типу "пакетний шторм". Автоматизовані механізми безпеки комутатора доступу миттєво зреагували на аномальне навантаження: активована політика обмеження швидкості зафіксувала критичне перевищення ліміту трафіку на вхідному порті. Комутатор миттєво зреагував і заблокував цей порт.

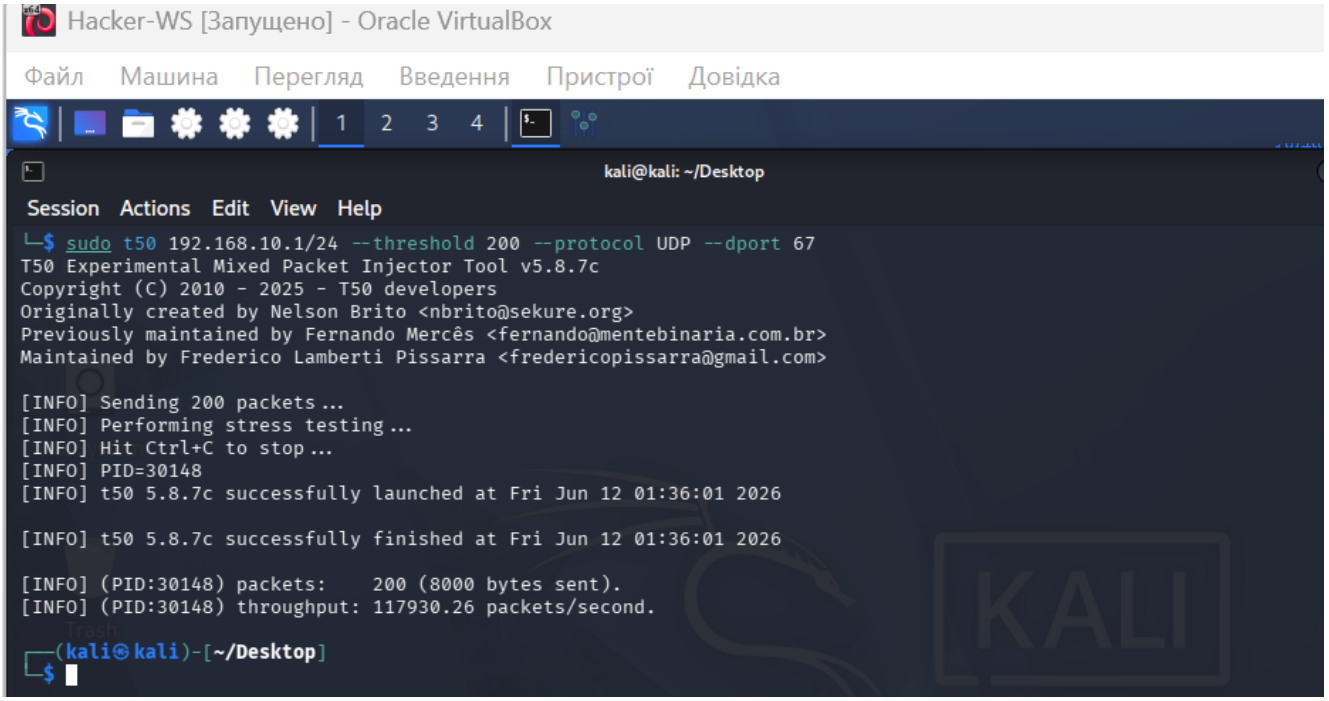


Рисунок 3.63 – Запуск атаки

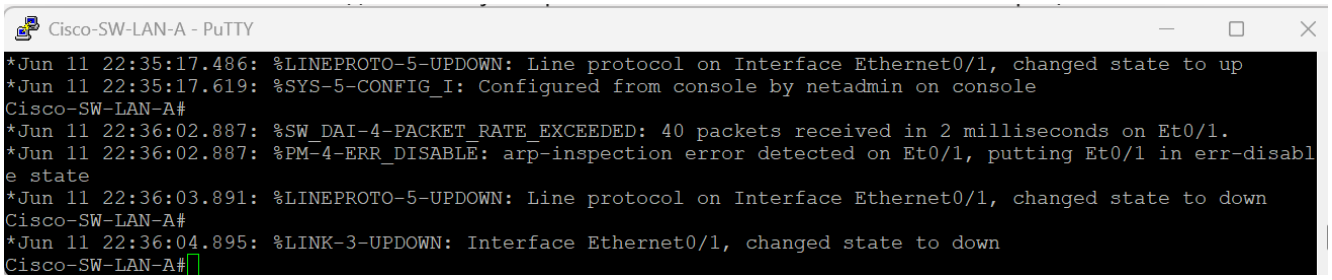


Рисунок 3.64 – Реакція комутатора на атаку



Рисунок 3.65 – Логування подій у Kibana

Було змодельовано атаку з метою примусової зміни таблиць маршрутизації протоколу OSPF. Через спеціалізовану утиліту vRIN було запущено анонсування нелегітимних маршрутів у мережу. Попри це, таблиця маршрутизації на обладнанні не зазнала жодних змін. Також спрацювали криптографічні фільтри перевірки заголовків OSPF: лічильник Authentication зафіксував 62 спроби порушення безпеки (пакети з невірним або відсутнім паролем/хешем).

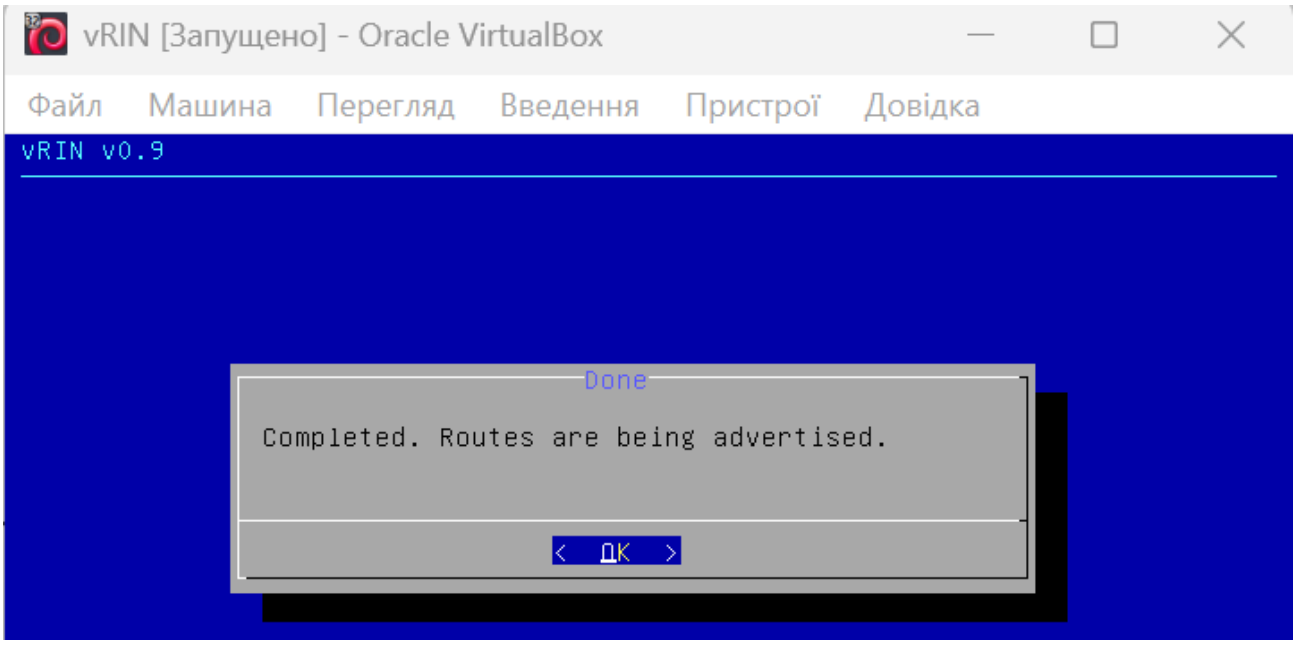


Рисунок 3.66 – Запуск атаки Route Injection

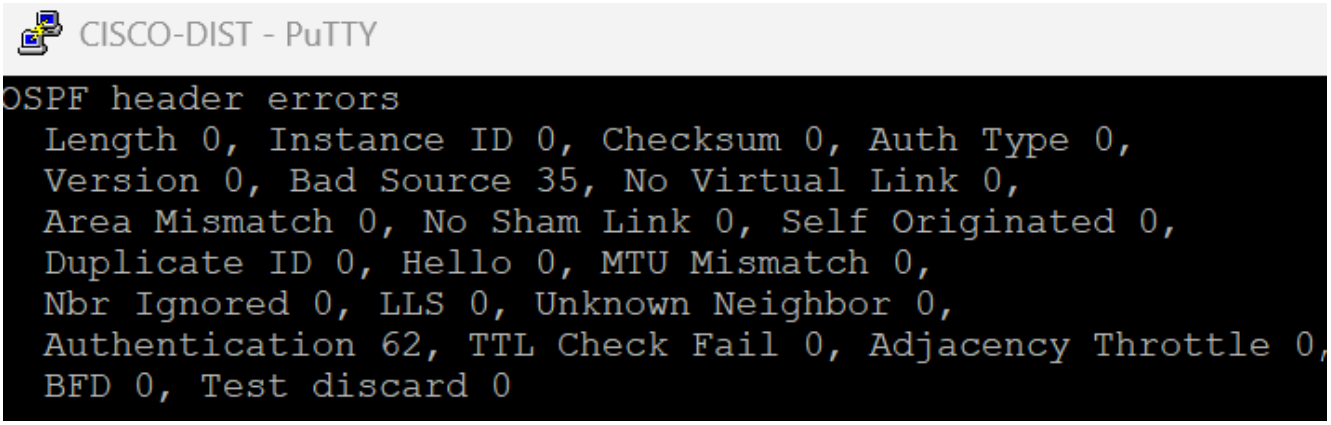


Рисунок 3.67 – Статистика помилок заголовків OSPF на маршрутизаторі

Висновок до третього розділу

У третьому розділі кваліфікаційної роботи було успішно реалізовано розгортання та комплексну перевірку безпеки мережної інфраструктури.

Було налаштовано IP-адресацію, динамічний протокол OSPF та сервіси автоматичного розподілу адрес. Для побудови ешелонованого захисту впроваджено списки контролю доступу, захист від підміни ідентифікаторів пристроїв на рівні комутації, а також сервер FreeRADIUS для централізованої автентифікації. Моніторинг та активний захист мережних процесів забезпечено інтеграцією системи виявлення та запобігання вторгненням Suricata, а також розгортанням підсистеми централізованого логування подій безпеки.

Етап тестування підтвердив стійкість інфраструктури до різних векторів загроз. Шляхом імітації аномальної активності перевірено захищеність мережі від розвідки та атак на внутрішні сервіси, механізми захисту продемонстрували вчасне виявлення й блокування порушень. Додатково підтверджено відмовостійкість протоколу OSPF в умовах штучного навантаження та збоїв.

ВИСНОВКИ

Поставлену мету кваліфікаційної роботи успішно реалізовано – спроєктовано та реалізовано захищену мережну інфраструктуру на базі обладнання Cisco та MikroTik, яка забезпечує автоматизоване виявлення та нейтралізацію мережних загроз шляхом інтеграції системи виявлення та запобігання вторгненням.

На початковому етапі проаналізовано методи побудови захищених комп'ютерних мереж, принципи забезпечення інформаційної безпеки та сучасні підходи до виявлення мережних атак. У результаті порівняльного аналізу програмних рішень було обґрунтовано доцільність використання системи Suricata, яка відзначається високою продуктивністю та підтримкою багатопотокової обробки даних у реальному часі.

Відповідно до поставлених завдань розроблено топологію комп'ютерної мережі та схему IP-адресації, що забезпечують логічну структурованість, керованість і можливість подальшого масштабування мережі. Виконано проєктування архітектури безпеки із застосуванням багаторівневого підходу до захисту інформаційних ресурсів.

У роботі впроваджено та налаштовано систему Suricata для моніторингу мережного трафіку в режимі реального часу, виявлення ознак атак та автоматичного реагування на потенційні загрози. Для централізованого збору, аналізу та візуалізації подій безпеки реалізовано систему моніторингу на базі Elastic Stack. Додатково налаштовано сервер FreeRADIUS для централізованої автентифікації користувачів і адміністраторів мережного обладнання, а також впроваджено механізми журналювання подій та засоби мережного захисту.

Проведене тестування підтвердило працездатність розробленого рішення та ефективність впроваджених засобів захисту. За результатами перевірки встановлено здатність мережної інфраструктури виявляти та блокувати підозрілу активність, забезпечувати централізований контроль подій безпеки та підтримувати стабільну роботу мережних сервісів.

Практична цінність отриманих результатів полягає у створенні працездатного шаблону безпечної мережної архітектури, який успішно поєднує

комерційні рішення та відкрите програмне забезпечення. Напрямки майбутнього розвитку роботи можуть передбачати розширення функціональності системи безпеки шляхом впровадження елементів машинного навчання для аналізу мережних загроз, автоматизації реагування на інциденти, а також інтеграції з хмарними сервісами моніторингу та управління безпекою.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

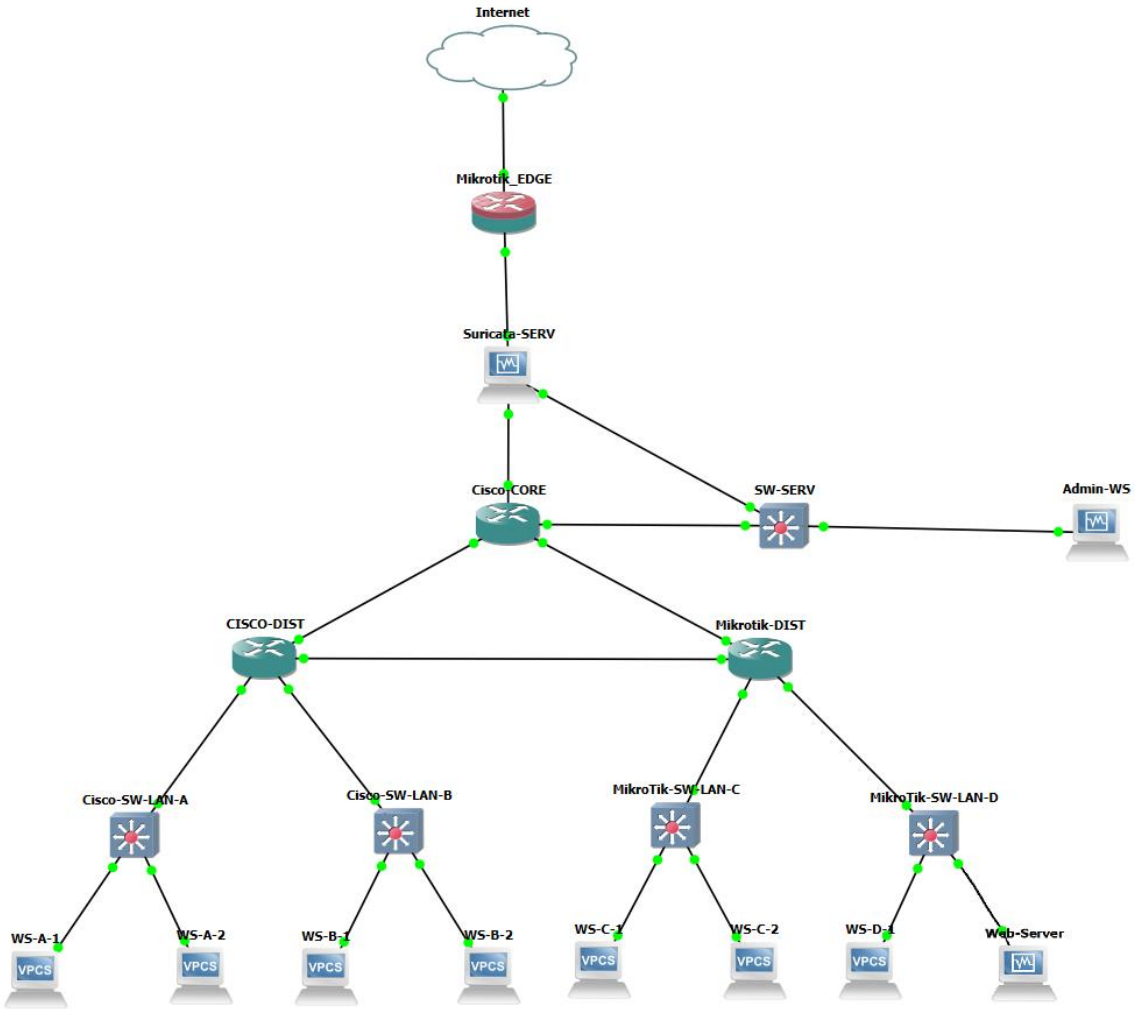
1. Tanenbaum E., Wetherall D. Computer Networks. 5-те вид. Pearson, 2010. URL: https://networking.harshkapadia.me/files/books/computer-networks-tanenbaum-5th-edition.pdf
2. Teare D., Paquet C. Campus Network Design Fundamentals. Cisco Press, 2006.
3. Goodrich M. T., Tamassia R. Introduction to Computer Security. Pearson, 2011. URL: https://www.pearson.de/media/muster/toc/toc_9781292037912.pdf
4. Kurose J. F., Ross K. W. Computer Networking: A Top-Down Approach. 7-ме вид. Pearson, 2017.
5. Cisco. AI Infrastructure, Secure Networking, and Software Solutions. URL: https://www.cisco.com/ (дата звернення: 17.05.2026).
6. Cisco. Secure Networking Overview. URL: https://www.cisco.com/site/us/en/solutions/transform-infrastructure/secure-networking-overview.html (дата звернення: 17.05.2026).
7. Cisco. Network Security Solutions. URL: https://www.cisco.com/site/us/en/products/security/network-security/index.html (дата звернення: 17.05.2026).
8. Mikrotik. About. URL: https://mikrotik.com/aboutus (дата звернення: 18.05.2026).
9. RouterOS Manual. URL: https://manual.mikrotik.com (дата звернення: 18.05.2026).
10. Intrusion Detection System (IDS) - GeeksforGeeks. URL: https://www.geeksforgeeks.org/ethical-hacking/intrusion-prevention-system-ips/ (дата звернення: 20.05.2026).
11. Intrusion Detection System (IDS) - GeeksforGeeks. URL: https://www.geeksforgeeks.org/ethical-hacking/intrusion-detection-system-ids/ (дата звернення: 20.05.2026).
12. Snort - Network Intrusion Detection & Prevention System. URL: https://www.snort.org/ (дата звернення: 21.05.2026).

13. Suricata: Home. URL: https://suricata.io/ (дата звернення: 21.05.2026).
14. The Zeek Network Security Monitor. URL: https://zeek.org/ (дата звернення: 21.05.2026).
15. Suricata User Guide — Suricata 8.0.3 documentation. URL: https://docs.suricata.io/en/suricata-8.0.3/ (дата звернення: 22.05.2026).
16. Elastic. Filebeat: Lightweight Log Analysis & Elasticsearch. URL: https://www.elastic.co/beats/filebeat (дата звернення: 22.05.2026).
17. Elastic. Elasticsearch: The Official Distributed Search & Analytics Engine. URL: https://www.elastic.co/elasticsearch (дата звернення: 22.05.2026).
18. Elastic. Kibana: Visualize, explore, and manage data in Elasticsearch. URL: https://www.elastic.co/kibana (дата звернення: 22.05.2026).
19. The FreeRADIUS project - Documentation. URL: https://www.freeradius.org/documentation/freeradius-server/3.2.9/ (дата звернення: 23.05.2026).
20. GNS3 Documentation. URL: https://docs.gns3.com/docs/ (дата звернення: 23.05.2026).
21. Configuration Guides - Cisco IOS 15.0S. URL: https://www.cisco.com/c/en/us/support/ios-nx-os-software/ios-15-0s/products-installation-and-configuration-guides-list.html (дата звернення: 29.05.2026).
22. Elasticsearch ingest pipelines. URL: https://www.elastic.co/docs/manage-data/ingest/transform-enrich/ingest-pipelines (дата звернення: 01.06.2026).

ДОДАТКИ

Додаток А

Схема побудованої мережі



Додаток Б

Таблиця Б.1

Технічні характеристики маршрутизатора Cisco ISR 4331

Параметр	Значення
Тип пристрою	Маршрутизатор
Архітектура	Multi-core x86
CPU	2 ядра (вбудований мережний процесор)
Продуктивність	до 100 Mbps/до ~300 Mbps
WAN інтерфейси	2 × RJ-45 Gigabit Ethernet
LAN інтерфейси	комбіновані WAN/LAN порти
Слоти розширення	2 × NIM (Network Interface Module), 1 × SM
USB порти	2 × USB 2.0
Console порт	RJ-45 + USB console
RAM	4 GB (до 16 GB)
Flash	4 GB
Протоколи маршрутизації	OSPF, EIGRP, BGP, RIP v2
VPN	IPsec, SSL VPN, DMVPN, FlexVPN
Firewall	Zone-Based Firewall (ZBFW)
NAT	Static, Dynamic, PAT
QoS	Advanced QoS (LLQ, CBWFQ, shaping, policing)
Максимум NAT сесій	>100 000
Операційна система	Cisco IOS XE
Живлення	AC internal PSU

Таблиця Б.2

Технічні характеристики комутатора Cisco WS-C2960+24LC-S

Параметр	Значення
Тип пристрою	Комутатор рівня доступу (Layer 2)
Архітектура	ASIC-based switching
Порти доступу	24 × 10/100 Mbps Fast Ethernet
Uplink порти	2 × 10/100/1000 Mbps
PoE	IEEE 802.3af (обмежений бюджет)
Switching capacity	16 Gbps
Forwarding rate	~10.1 Mpps
MAC-таблиця	8000 адрес
VLAN	до 255 VLAN
VLAN типи	Access, Trunk (802.1Q)
Spanning Tree	STP, RSTP, PVST+
Port Security	MAC filtering, limit MAC per port
DHCP Snooping	підтримується
Dynamic ARP Inspection	підтримується
QoS	802.1p, CoS, traffic shaping
Link Aggregation	EtherChannel (LACP/PAgP)
Console доступ	RJ-45
Операційна система	Cisco IOS

Таблиця Б.3

Технічні характеристики маршрутизатора MikroTik CCR2004-16G-2S+

Параметр	Значення
Тип пристрою	Маршрутизатор
CPU	Annapurna Labs Alpine v2
Архітектура CPU	ARM 64-bit
Кількість ядер	4
Частота CPU	1.7 GHz
RAM	4 GB DDR4
Storage	NAND 128 MB
Порти Ethernet	16 × 10/100/1000 Mbps
SFP+ порти	2 × 10G
Switching capability	L3 routing (hardware accelerated)
Продуктивність	до 3.4 Mpps
Routing throughput	до ~7.5 Gbps (залежить від конфігурації)
VLAN	802.1Q
Routing протоколи	OSPF, BGP, RIP, MPLS
Firewall	Stateful firewall
NAT	Source NAT, Destination NAT
VPN	IPsec, L2TP, WireGuard, GRE
Queue/QoS	Simple queues, queue tree
Операційна система	RouterOS v7
Охолодження	активне (вентилятори)

Таблиця Б.4

Технічні характеристики комутатора MikroTik CRS328-24P-4S+RM

Параметр	Значення
Тип пристрою	Layer 2/3 комутатор
Архітектура	Switch-chip + RouterOS
Порти Ethernet	24 × 10/100/1000 Mbps
PoE підтримка	802.3af/at + Passive PoE
PoE бюджет	до 450 W
Uplink порти	4 × 10G SFP+
Switching capacity	64 Gbps
Forwarding rate	до 95 Mpps
MAC-таблиця	16 000 записів
VLAN	802.1Q VLAN filtering
Spanning Tree	STP / RSTP / MSTP
IGMP Snooping	підтримується
Port isolation	підтримується
Jumbo frames	до 10218 bytes
CPU	Dual-core 800 MHz
RAM	512 MB
Storage	16 MB flash
Операційна система	RouterOS / SwitchOS
Cooling	активне охолодження

Додаток В

Таблиця В.1

Адресація мережних пристроїв

Мережа / Пристрій	Інтерфейс/Мережний Адаптер/Шлюз	IP-Адреса	Маска	Префікс
Підмережа Internet – Mikrotik_EDGE	-	203.0.113.0	255.255.255.252	/30
Internet	Ether1	203.0.113.1	255.255.255.252	/30
Mikrotik_EDGE	Ether1	203.0.113.2	255.255.255.252	/30
Підмережа Mikrotik_EDGE – Cisco-CORE	-	10.0.0.1	255.255.255.252	/30
Mikrotik_EDGE	Ether2	10.0.0.1	255.255.255.252	/30
Cisco-CORE	Gi0/0	10.0.0.2	255.255.255.252	/30
Підмережа Cisco-CORE – Cisco-DIST	-	10.0.0.4	255.255.255.252	/30
Cisco-CORE	Gi0/1	10.0.0.5	255.255.255.252	/30
Cisco-DIST	Gi0/0	10.0.0.6	255.255.255.252	/30
Підмережа Cisco-CORE – Mikrotik-DIST	-	10.0.0.8	255.255.255.252	/30
Cisco-CORE	Gi0/2	10.0.0.9	255.255.255.252	/30
Mikrotik-DIST	Ethe1	10.0.0.10	255.255.255.252	/30
Підмережа Cisco-DIST – Mikrotik-DIST	-	10.0.0.12	255.255.255.252	/30
Cisco-DIST	Gi0/2	10.0.0.13	255.255.255.252	/30
Mikrotik-DIST	Ether2	10.0.0.14	255.255.255.252	/30
Підмережа A	-	192.168.10.0	255.255.255.0	/24
Cisco-CORE	Gi0/3	192.168.10.1	255.255.255.0	/24
Cisco-SW-LAN-A	Vlan 1	192.168.10.2	255.255.255.0	/24
	Шлюз за замовчуванням	192.168.10.1		
Підмережа B	-	192.168.20.0	255.255.255.0	/24
Cisco-DIST	Gi0/4	192.168.20.1	255.255.255.0	/24
Cisco-SW-LAN-B	Vlan 1	192.168.20.2	255.255.255.0	/24
	Шлюз за замовчуванням	192.168.20.1		
Підмережа C	-	192.168.30.0	255.255.255.0	/24
Mikrotik-DIST	Ether3	192.168.30.1	255.255.255.0	/24
Mikrotik-SW-LAN-C	bridge-Lan	192.168.30.2	255.255.255.0	/24
	Шлюз за замовчуванням	192.168.30.1		
Підмережа D	-	192.168.40.0	255.255.255.0	/24
Mikrotik-DIST	Ether4	192.168.40.1	255.255.255.0	/24
Mikrotik-SW-LAN-D	bridge-Lan	192.168.40.2	255.255.255.0	/24
	Шлюз за замовчуванням	192.168.40.1		
Web-Server	DHCP(Static Lease)	192.168.40.3	255.255.255.0	/24
	Шлюз за замовчуванням	192.168.40.1		

Підмережа SERV	-	192.168.100.0	255.255.255.248	/29
Cisco-CORE	Gi0/3	192.168.100.1	255.255.255.248	/29
SW-LAN-SERV	Vlan 1	192.168.100.2	255.255.255.248	/29
	Шлюз за замовчуванням	192.168.100.1		
Admin-SERV	Мережний адаптер	192.168.100.3	255.255.255.248	/29
	Шлюз за замовчуванням	192.168.100.1		
Suricata-SERV	Мережний адаптер	192.168.100.4	255.255.255.248	/29
	Шлюз за замовчуванням	192.168.100.1		

Додаток Г

Конфігураційні файли мережних пристроїв

Mikrotik_EDGE:

```
# 2026-06-12 23:34:11 by RouterOS 7.20.6
# system id = ONScjkpA6/A
/interface ethernet
set [ find default-name=ether1 ] comment=LINK-TO-INTERNET \
    disable-running-check=no
set [ find default-name=ether2 ] comment=LINK-TO-IPS disable-running-check=no
/port
set 0 name=serial0
/routing ospf instance
add disabled=no name=ospf-main originate-default=if-installed router-id=\
    1.1.1.1
/routing ospf area
add disabled=no instance=ospf-main name=backbone
/system logging action
add name=syslog remote=192.168.100.3 target=remote
/ip address
add address=10.0.0.1/30 interface=ether2 network=10.0.0.0
add address=1.1.1.1 interface=lo network=1.1.1.1
add address=203.0.113.2/30 interface=ether1 network=203.0.113.0
/ip dhcp-client
add interface=ether1
/ip firewall filter
add action=accept chain=input comment="Accept ICMP" limit=5,10:packet \
    protocol=icmp
add action=accept chain=input comment="Accept established,related" \
    connection-state=established,related
add action=drop chain=input comment="Drop invalid" connection-state=invalid
add action=drop chain=input comment="Drop all  from WAN" in-interface=ether1 \
    log=yes log-prefix=INPUT-DROP
add action=accept chain=forward comment="Accept established,related" \
    connection-state=established,related
add action=drop chain=forward comment="Drop invalid" connection-state=invalid
add action=drop chain=forward comment="Drop new from WAN not DSTNATed" \
    connection-nat-state=!dstnat connection-state=new in-interface=ether1 \
    log=yes log-prefix="WAN-DROP: "
/ip firewall nat
add action=dst-nat chain=dstnat comment="Web Server Forwarding" dst-port=80 \
    in-interface=ether1 protocol=tcp to-addresses=192.168.40.3 to-ports=80
add action=masquerade chain=srcnat out-interface=ether1
/ip route
add dst-address=0.0.0.0/0 gateway=203.0.113.1
/ip service
set ftp disabled=yes
set ssh address=192.168.100.0/29
set telnet disabled=yes
set www disabled=yes
set winbox address=192.168.100.0/29
set api disabled=yes
set api-ssl disabled=yes
/ip ssh
set host-key-size=4096 strong-crypto=yes
/radius
add address=192.168.100.3 service=login timeout=300ms
/routing ospf interface-template
add area=backbone auth=sha256 auth-id=1 disabled=no interfaces=ether2
/system identity
set name=Mikrotik_EDGE
/system logging
add action=syslog topics=info
add action=syslog topics=error
add action=syslog topics=warning
add action=syslog topics=critical
add action=syslog topics=account
/tool bandwidth-server
set enabled=no
/tool mac-server mac-winbox
set allowed-interface-list=none
```

```
/user aaa
set interim-update=20m use-radius=yes
```

Cisco-CORE:

```
Using 5338 out of 262144 bytes
!
! Last configuration change at 14:49:20 UTC Mon Jun 8 2026 by netadmin
!
version 15.6
service timestamps debug datetime msec
service timestamps log datetime msec
service password-encryption
!
hostname Cisco-CORE
!
boot-start-marker
boot-end-marker
!
!
enable secret 5 $1$BwaX$urk4hg8bR6I1yYjN005Qs1
!
aaa new-model
!
!
aaa group server radius RadiusGroup
 server name Radius-Server
!
aaa authentication login default group RadiusGroup local
aaa authorization console
aaa authorization exec default group RadiusGroup local
aaa accounting exec default start-stop group RadiusGroup
!
!
aaa session-id common
ethernet lmi ce
!
!
mmi polling-interval 60
no mmi auto-configure
no mmi pvc
mmi snmp-timeout 180
!
!
!
no ip icmp rate-limit unreachable
!
!
!
no ip domain lookup
ip domain name bilous.com
ip cef
login block-for 60 attempts 5 within 30
no ipv6 cef
!
multilink bundle-name authenticated
!
!
!
key chain OSPF_EDGE
 key 1
  key-string 7 045F2A0D000A144703341C1634
  cryptographic-algorithm hmac-sha-256
key chain OSPF_CISCO_DIST
 key 1
  key-string 7 05052358254F6F5B15344F0804
  cryptographic-algorithm hmac-sha-256
key chain OSPF_MIKROTIK_DIST
 key 1
  key-string 7 1511321F17057D303A142F38
  cryptographic-algorithm hmac-sha-256
!
!
username localadmin privilege 15 secret 5 $1$lgIJ$LqL/HgA//mVnM4713QQDr.
!
redundancy
!
no cdp log mismatch duplex
!
ip tcp synwait-time 5
!
!
!
interface Loopback0
 ip address 1.1.1.2 255.255.255.255
!
interface GigabitEthernet0/0
 description LINK-TO-IPS
 ip address 10.0.0.2 255.255.255.252
 ip ospf authentication key-chain OSPF_EDGE
 ip ospf 1 area 0
 duplex auto
 speed auto
 media-type rj45
!
interface GigabitEthernet0/1
 description LINK-TO-CISCO-DIST
 ip address 10.0.0.5 255.255.255.252
```

```
 ip ospf authentication key-chain OSPF_CISCO_DIST
 ip ospf 1 area 0
 duplex auto
 speed auto
 media-type rj45
!
interface GigabitEthernet0/2
 description LINK-TO-MIKROTIK-DIST
 ip address 10.0.0.9 255.255.255.252
 ip ospf authentication key-chain OSPF_MIKROTIK_DIST
 ip ospf 1 area 0
 duplex auto
 speed auto
 media-type rj45
!
interface GigabitEthernet0/3
 description LINK-TO-SERV
 ip address 192.168.100.1 255.255.255.248
 ip access-group ACL-SERV out
 ip ospf 1 area 0
 duplex auto
 speed auto
 media-type rj45
!
router ospf 1
 router-id 1.1.1.2
 passive-interface GigabitEthernet0/3
!
ip forward-protocol nd
!
!
no ip http server
no ip http secure-server
ip ssh time-out 30
ip ssh authentication-retries 5
ip ssh version 2
!
ip access-list standard MGMT-ACCESS
 permit 192.168.100.0 0.0.0.7
 deny   any
!
ip access-list extended ACL-SERV
 permit tcp any 192.168.100.0 0.0.0.7 established
 permit udp any eq domain 192.168.100.0 0.0.0.7
 permit udp any eq ntp 192.168.100.0 0.0.0.7
 permit udp any 192.168.100.0 0.0.0.7 eq syslog
 permit udp any 192.168.100.0 0.0.0.7 eq 9003
 permit udp any 192.168.100.0 0.0.0.7 eq 1812
 permit udp any 192.168.100.0 0.0.0.7 eq 1813
 permit icmp any 192.168.100.0 0.0.0.7 echo-reply
 deny   ip any log
!
logging origin-id hostname
logging host 192.168.100.3 transport udp port 9003
!
!
!
!
radius server Radius-Server
 address ipv4 192.168.100.3 auth-port 1812 acct-port 1813
 key 7 15363E38117B3C13781F67701D
!
!
line con 0
 exec-timeout 0 0
 privilege level 15
 logging synchronous
line aux 0
 exec-timeout 0 0
 privilege level 15
 logging synchronous
line vty 0 4
 access-class MGMT-ACCESS in
 transport input ssh
!
no scheduler allocate
!
End
```

Cisco-DIST:

```
Using 5215 out of 262144 bytes
!
! Last configuration change at 18:05:53 UTC Mon Jun 1 2026 by netadmin
!
version 15.6
service timestamps debug datetime msec
service timestamps log datetime msec
service password-encryption
!
hostname Cisco-DIST
!
boot-start-marker
boot-end-marker
!
!
```

```
enable secret 5 $1$M9Jt$JiUJMH6cKV33ImH1QuWwm/
!
aaa new-model
!
!
aaa group server radius RadiusGroup
 server name Radius-Server
!
aaa authentication login default group RadiusGroup local
aaa authorization console
aaa authorization exec default group RadiusGroup local
aaa accounting exec default start-stop group RadiusGroup
!
!
!
!
!
aaa session-id common
ethernet lmi ce
!
!
!
mmi polling-interval 60
no mmi auto-configure
no mmi pvc
mmi snmp-timeout 180
!
!
!
no ip icmp rate-limit unreachable
!
!
!
ip dhcp excluded-address 192.168.10.1 192.168.10.10
ip dhcp excluded-address 192.168.10.171 192.168.10.254
ip dhcp excluded-address 192.168.20.1 192.168.20.10
ip dhcp excluded-address 192.168.20.171 192.168.20.254
!
ip dhcp pool LAN-A
 network 192.168.10.0 255.255.255.0
 default-router 192.168.10.1
 lease 0 12
!
ip dhcp pool LAN-B
 network 192.168.20.0 255.255.255.0
 default-router 192.168.20.1
 lease 0 12
!
!
!
no ip domain lookup
ip domain name bilous.com
ip cef
login block-for 60 attempts 5 within 30
no ipv6 cef
!
multilink bundle-name authenticated
!
!
!
key chain OSPF_CORE
 key 1
  key-string 7 000A365100582A540310145406
  cryptographic-algorithm hmac-sha-256
key chain OSPF_MIKROTIK_DIST
 key 1
  key-string 7 045E21531814155D033C530616
  cryptographic-algorithm hmac-sha-256
!
!
username localadmin privilege 15 secret 5 $1$cGAF$mpqHZOLGoBUJ3OUU4eEJe1
!
redundancy
!
no cdp log mismatch duplex
!
ip tcp synwait-time 5
!
!
!
interface Loopback0
 ip address 2.2.2.1 255.255.255.255
!
interface GigabitEthernet0/0
 description LINK-TO-CISCO-CORE
 ip address 10.0.0.6 255.255.255.252
 ip ospf authentication key-chain OSPF_CORE
 ip ospf 1 area 0
 duplex auto
 speed auto
 media-type rj45
!
interface GigabitEthernet0/1
 description LINK-TO-MIKROTIK-DIST
 ip address 10.0.0.13 255.255.255.252
```

```
 ip ospf authentication key-chain OSPF_MIKROTIK_DIST
 ip ospf 1 area 0
 duplex auto
 speed auto
 media-type rj45
!
interface GigabitEthernet0/2
 description LINK-TO-LAN-A
 ip address 192.168.10.1 255.255.255.0
 ip ospf 1 area 0
 duplex auto
 speed auto
 media-type rj45
!
interface GigabitEthernet0/3
 description LINK-TO-LAN-B
 ip address 192.168.20.1 255.255.255.0
 ip ospf 1 area 0
 duplex auto
 speed auto
 media-type rj45
!
router ospf 1
 router-id 2.2.2.1
 passive-interface GigabitEthernet0/2
 passive-interface GigabitEthernet0/3
!
ip forward-protocol nd
!
!
no ip http server
no ip http secure-server
ip ssh time-out 30
ip ssh authentication-retries 5
ip ssh version 2
!
ip access-list standard MGMT-ACCESS
 permit 192.168.100.0 0.0.0.7
 deny   any
!
!
logging origin-id hostname
logging host 192.168.100.3 transport udp port 9003
!
!
!
radius-server retransmit 1
radius-server timeout 2
!
radius server Radius-Server
 address ipv4 192.168.100.3 auth-port 1812 acct-port 1813
 key 7 09687B3D0C5400255B20567825
!
!
line con 0
 exec-timeout 0 0
 privilege level 15
 logging synchronous
line aux 0
 exec-timeout 0 0
 privilege level 15
 logging synchronous
line vty 0 4
 access-class MGMT-ACCESS in
 transport input ssh
!
no scheduler allocate
!
End
```

Mikrotik-DIST:

```
# 2026-06-13 00:04:49 by RouterOS 7.20.6
# system id = Hfn86TDc45L
#
/interface ethernet
set [ find default-name=ether1 ] comment=LINK-TO-CISCO-CORE \
    disable-running-check=no
set [ find default-name=ether2 ] comment=LINK-TO-CISCO-DIST \
    disable-running-check=no
set [ find default-name=ether3 ] comment=LINK-TO-LAN-C disable-running-check=\
    no
set [ find default-name=ether4 ] comment=LINK-TO-LAN-D disable-running-check=\
    no
/ip pool
add name=dhcp_pool_lan_c ranges=192.168.30.2-192.168.30.170
add name=dhcp_pool_lan_d ranges=192.168.40.2-192.168.40.170
/ip dhcp-server
add add-arp=yes address-pool=dhcp_pool_lan_c interface=ether3 lease-time=12h \
    name=LAN-C
```

```
add add-arp=yes address-pool=dhcp_pool_lan_d interface=ether4 lease-time=12h \
   name=LAN-D
/port
set 0 name=serial0
/routing ospf instance
add disabled=no name=ospf-main router-id=2.2.2.2
/routing ospf area
add disabled=no instance=ospf-main name=backbone
/system logging action
add name=syslog remote=192.168.100.3 target=remote
/ip address
add address=10.0.0.10/30 interface=ether1 network=10.0.0.8
add address=10.0.0.14/30 interface=ether2 network=10.0.0.12
add address=192.168.40.1/24 interface=ether4 network=192.168.40.0
add address=192.168.30.1/24 interface=ether3 network=192.168.30.0
add address=2.2.2.2 interface=lo network=2.2.2.2
/ip dhcp-client
add interface=ether1
/ip dhcp-server lease
add address=192.168.30.2 mac-address=0C:43:36:2D:00:00 server=LAN-C
add address=192.168.40.2 mac-address=0C:17:78:CD:00:00 server=LAN-D
add address=192.168.40.3 mac-address=08:00:27:D4:51:04 server=LAN-D
/ip dhcp-server network
add address=192.168.30.0/24 dns-none=yes gateway=192.168.30.1
add address=192.168.40.0/24 dns-none=yes gateway=192.168.40.1
/ip service
set ftp disabled=yes
set ssh address=192.168.100.0/29
set telnet disabled=yes
set www disabled=yes
set winbox address=192.168.100.0/29
set api disabled=yes
set api-ssl disabled=yes
/ip ssh
set host-key-size=4096 strong-crypto=yes
/radius
add address=192.168.100.3 service=login timeout=300ms
/routing ospf interface-template
add area=backbone auth=sha256 auth-id=1 disabled=no interfaces=ether1
add area=backbone auth=sha256 auth-id=1 disabled=no interfaces=ether2
add area=backbone disabled=no interfaces=ether3 passive
add area=backbone disabled=no interfaces=ether4 passive
/system identity
set name=Mikrotik_DIST
/system logging
add action=syslog topics=info
add action=syslog topics=error
add action=syslog topics=warning
add action=syslog topics=account
add action=syslog topics=critical
/tool bandwidth-server
set enabled=no
/tool mac-server mac-winbox
set allowed-interface-list=none
/user aaa
set interim-update=20m use-radius=yes
```

SW-SERV:

```
Using 2934 out of 8192 bytes
!
! Last configuration change at 18:06:18 UTC Mon Jun 1 2026 by netadmin
!
version 15.2
service timestamps debug datetime msec
service timestamps log datetime msec
service password-encryption
service compress-config
!
hostname SW-SERV
!
boot-start-marker
boot-end-marker
!
!
logging discriminator EXCESS severity drops 6 msg-body drops EXCESSCOLL
logging buffered 50000
logging console discriminator EXCESS
enable secret 5 $1$/3dH$ouUlz7G/x2vWptr7sxbuz/
!
```

```
username localadmin privilege 15 secret 5 $1$rj6.$MA4z6f1WUSmztTdG9kNUx/
aaa new-model
!
!
aaa group server radius RadiusGroup
 server name Radius-Server
!
aaa authentication login default group RadiusGroup local
aaa authorization console
aaa authorization exec default group RadiusGroup local
aaa accounting exec default start-stop group RadiusGroup
!
!
!
aaa session-id common
!
!
!
no ip routing
no ip icmp rate-limit unreachable
!
!
!
no ip domain-lookup
ip domain-name bilous.com
no ip cef
login block-for 60 attempts 5 within 30
no ipv6 cef
!
!
spanning-tree mode pvst
spanning-tree extend system-id
!
vlan internal allocation policy ascending
!
ip tcp synwait-time 5
ip ssh time-out 30
ip ssh authentication-retries 5
ip ssh version 2
!
!
!
!
interface Ethernet0/0
!
interface Ethernet0/1
 switchport mode access
 switchport port-security violation restrict
 switchport port-security mac-address sticky
 switchport port-security mac-address sticky 0800.276f.a12e
 switchport port-security
 spanning-tree portfast
 spanning-tree bpduguard enable
!
interface Ethernet0/2
 switchport mode access
 switchport port-security violation restrict
 switchport port-security mac-address sticky
 switchport port-security mac-address sticky 0800.270a.d57d
 switchport port-security
 spanning-tree portfast
 spanning-tree bpduguard enable
!
interface Ethernet0/3
 shutdown
!
interface Ethernet1/0
 shutdown
!
interface Ethernet1/1
 shutdown
!
interface Ethernet1/2
 shutdown
!
interface Ethernet1/3
 shutdown
!
interface Ethernet2/0
 shutdown
!
interface Ethernet2/1
 shutdown
!
interface Ethernet2/2
 shutdown
!
interface Ethernet2/3
 shutdown
!
interface Ethernet3/0
 shutdown
!
interface Ethernet3/1
 shutdown
!
interface Ethernet3/2
 shutdown
```

```
!
interface Ethernet3/3
 shutdown
!
interface Vlan1
 ip address 192.168.100.2 255.255.255.248
 no ip route-cache
!
ip forward-protocol nd
!
no ip http server
no ip http secure-server
!
!
ip access-list standard MGMT-ACCESS
 permit 192.168.100.0 0.0.0.7
 deny   any
!
!
logging origin-id hostname
logging host 192.168.100.3 transport udp port 9003
!
!
!
!
radius server Radius-Server
 address ipv4 192.168.100.3 auth-port 1812 acct-port 1813
 key 7 052F333B341D593E4929454005
!
!
control-plane
!
!
line con 0
 exec-timeout 0 0
 privilege level 15
 logging synchronous
line aux 0
 exec-timeout 0 0
 privilege level 15
 logging synchronous
line vty 0 4
 access-class MGMT-ACCESS in
 transport input ssh
!
!
End
```

Cisco-SW-LAN-A:

```
Using 3593 out of 8192 bytes
!
! Last configuration change at 18:14:21 UTC Mon Jun 1 2026 by netadmin
!
version 15.2
service timestamps debug datetime msec
service timestamps log datetime msec
service password-encryption
service compress-config
!
hostname Cisco-SW-LAN-A
!
boot-start-marker
boot-end-marker
!
!
logging discriminator EXCESS severity drops 6 msg-body drops EXCESSCOLL
logging buffered 50000
logging console discriminator EXCESS
enable secret 5 $1$7dUu$okPVJE96yyJLsEo/11lr7/
!
username localadmin privilege 15 secret 5 $1$RST3$PjYgye84eqWpYjmT2e/fL1
aaa new-model
!
!
aaa group server radius RadiusGroup
 server name Radius-Server
!
aaa authentication login default group RadiusGroup local
aaa authorization console
aaa authorization exec default group RadiusGroup local
aaa accounting exec default start-stop group RadiusGroup
!
!
!
aaa session-id common
!
!
no ip routing
ip arp inspection vlan 1
no ip icmp rate-limit unreachable
!
!
!
ip dhcp snooping vlan 1
no ip dhcp snooping information option
```

```
ip dhcp snooping
no ip domain-lookup
ip domain-name bilous.com
no ip cef
login block-for 60 attempts 5 within 30
no ipv6 cef
!
!
errdisable recovery cause dhcp-rate-limit
errdisable recovery interval 120
spanning-tree mode pvst
spanning-tree extend system-id
!
vlan internal allocation policy ascending
!
ip tcp synwait-time 5
ip ssh time-out 30
ip ssh authentication-retries 5
ip ssh version 2
!
!
!
!
interface Ethernet0/0
 ip arp inspection trust
 ip dhcp snooping limit rate 256
 ip dhcp snooping trust
!
interface Ethernet0/1
 switchport mode access
 switchport port-security maximum 2
 switchport port-security violation restrict
 switchport port-security mac-address sticky
 switchport port-security mac-address sticky
0050.7966.6800
 switchport port-security
 spanning-tree portfast
 spanning-tree bpduguard enable
 ip dhcp snooping limit rate 256
!
interface Ethernet0/2
 switchport mode access
 switchport port-security maximum 2
 switchport port-security violation restrict
 switchport port-security mac-address sticky
 switchport port-security mac-address sticky
0050.7966.6801
 switchport port-security
 spanning-tree portfast
 spanning-tree bpduguard enable
 ip dhcp snooping limit rate 256
!
interface Ethernet0/3
 switchport mode access
 switchport port-security maximum 2
 switchport port-security violation restrict
 switchport port-security mac-address sticky
 switchport port-security
 spanning-tree portfast
 spanning-tree bpduguard enable
 ip dhcp snooping limit rate 256
!
interface Ethernet1/0
 shutdown
!
interface Ethernet1/1
 shutdown
!
interface Ethernet1/2
 shutdown
!
interface Ethernet1/3
 shutdown
!
interface Ethernet2/0
 shutdown
!
interface Ethernet2/1
 shutdown
!
interface Ethernet2/2
 shutdown
!
interface Ethernet2/3
 shutdown
!
interface Ethernet3/0
 shutdown
!
interface Ethernet3/1
 shutdown
!
interface Ethernet3/2
 shutdown
!
interface Ethernet3/3
 shutdown
!
interface Vlan1
 ip address 192.168.10.2 255.255.255.0
 no ip route-cache
!
ip forward-protocol nd
!
```

```
no ip http server
no ip http secure-server
!
!
ip access-list standard MGMT-ACCESS
 permit 192.168.100.0 0.0.0.7
 deny   any
!
!
logging origin-id hostname
logging host 192.168.100.3 transport udp port 9003
!
!
!
!
radius server Radius-Server
 address ipv4 192.168.100.3 auth-port 1812 acct-port 1813
 key 7 1433273F19553D1C740461672C
!
!
control-plane
!
!
line con 0
 exec-timeout 0 0
 privilege level 15
 logging synchronous
line aux 0
 exec-timeout 0 0
 privilege level 15
 logging synchronous
line vty 0 4
 access-class MGMT-ACCESS in
 transport input ssh
!
!
End
```

Cisco-SW-LAN-B:

```
Using 3593 out of 8192 bytes
!
! Last configuration change at 18:17:30 UTC Mon Jun 1 2026 by netadmin
!
version 15.2
service timestamps debug datetime msec
service timestamps log datetime msec
service password-encryption
service compress-config
!
hostname Cisco-SW-LAN-B
!
boot-start-marker
boot-end-marker
!
!
logging discriminator EXCESS severity drops 6 msg-body drops EXCESSCOLL
logging buffered 50000
logging console discriminator EXCESS
enable secret 5 $1$Gyru$ljWJzyuPDi9XwZOp0moT.0
!
username localadmin privilege 15 secret 5 $1$oNnd$DwzPGujQ6byAVw68J7vQZ/
aaa new-model
!
!
aaa group server radius RadiusGroup
 server name Radius-Server
!
aaa authentication login default group RadiusGroup local
aaa authorization console
aaa authorization exec default group RadiusGroup local
aaa accounting exec default start-stop group RadiusGroup
!
!
!
aaa session-id common
!
!
no ip routing
ip arp inspection vlan 1
no ip icmp rate-limit unreachable
!
!
ip dhcp snooping vlan 1
no ip dhcp snooping information option
ip dhcp snooping
no ip domain-lookup
ip domain-name bilous.com
no ip cef
login block-for 60 attempts 5 within 30
no ipv6 cef
!
!
errdisable recovery cause dhcp-rate-limit
errdisable recovery interval 120
spanning-tree mode pvst
```

```
spanning-tree extend system-id
!
vlan internal allocation policy ascending
!
ip tcp synwait-time 5
ip ssh time-out 30
ip ssh authentication-retries 5
ip ssh version 2
!
!
!
interface Ethernet0/0
 ip arp inspection trust
 ip dhcp snooping limit rate 256
 ip dhcp snooping trust
!
interface Ethernet0/1
 switchport mode access
 switchport port-security maximum 2
 switchport port-security violation restrict
 switchport port-security mac-address sticky
 switchport port-security mac-address sticky 0050.7966.6802
 switchport port-security
 spanning-tree portfast
 spanning-tree bpduguard enable
 ip dhcp snooping limit rate 256
!
interface Ethernet0/2
 switchport mode access
 switchport port-security maximum 2
 switchport port-security violation restrict
 switchport port-security mac-address sticky
 switchport port-security mac-address sticky 0050.7966.6803
 switchport port-security
 spanning-tree portfast
 spanning-tree bpduguard enable
 ip dhcp snooping limit rate 256
!
interface Ethernet0/3
 switchport mode access
 switchport port-security maximum 2
 switchport port-security violation restrict
 switchport port-security mac-address sticky
 switchport port-security
 spanning-tree portfast
 spanning-tree bpduguard enable
 ip dhcp snooping limit rate 256
!
interface Ethernet1/0
 shutdown
!
interface Ethernet1/1
 shutdown
!
interface Ethernet1/2
 shutdown
!
interface Ethernet1/3
 shutdown
!
interface Ethernet2/0
 shutdown
!
interface Ethernet2/1
 shutdown
!
interface Ethernet2/2
 shutdown
!
interface Ethernet2/3
 shutdown
!
interface Ethernet3/0
 shutdown
!
interface Ethernet3/1
 shutdown
!
interface Ethernet3/2
 shutdown
!
interface Ethernet3/3
 shutdown
!
interface Vlan1
 ip address 192.168.20.2 255.255.255.0
 no ip route-cache
!
ip forward-protocol nd
!
no ip http server
no ip http secure-server
!
!
ip access-list standard MGMT-ACCESS
 permit 192.168.100.0 0.0.0.7
 deny   any
!
!
logging origin-id hostname
logging host 192.168.100.3 transport udp port 9003
```

```
!
!
radius server Radius-Server
 address ipv4 192.168.100.3 auth-port 1812 acct-port 1813
 key 7 1433273F19553D1C740461672C
!
!
control-plane
!
!
line con 0
 exec-timeout 0 0
 privilege level 15
 logging synchronous
line aux 0
 exec-timeout 0 0
 privilege level 15
 logging synchronous
line vty 0 4
 access-class MGMT-ACCESS in
 transport input ssh
!
!
End
```

Mikrotik-SW-LAN-C:

```
# jun/13/2026 00:22:21 by RouterOS 7.8
# software id =
#
/interface bridge
add dhcp-snooping=yes name=bridge-Lan
/interface ethernet
set [ find default-name=ether1 ] disable-running-check=no
set [ find default-name=ether2 ] arp=reply-only disable-running-check=no
set [ find default-name=ether3 ] arp=reply-only disable-running-check=no
set [ find default-name=ether4 ] disable-running-check=no disabled=yes
set [ find default-name=ether5 ] disable-running-check=no disabled=yes
set [ find default-name=ether6 ] disable-running-check=no disabled=yes
set [ find default-name=ether7 ] disable-running-check=no disabled=yes
set [ find default-name=ether8 ] disable-running-check=no disabled=yes
set [ find default-name=ether9 ] disable-running-check=no disabled=yes
set [ find default-name=ether10 ] disable-running-check=no disabled=yes
set [ find default-name=ether11 ] disable-running-check=no disabled=yes
set [ find default-name=ether12 ] disable-running-check=no disabled=yes
set [ find default-name=ether13 ] disable-running-check=no disabled=yes
set [ find default-name=ether14 ] disable-running-check=no disabled=yes
set [ find default-name=ether15 ] disable-running-check=no disabled=yes
set [ find default-name=ether16 ] disable-running-check=no disabled=yes
set [ find default-name=ether17 ] disable-running-check=no disabled=yes
set [ find default-name=ether18 ] disable-running-check=no disabled=yes
set [ find default-name=ether19 ] disable-running-check=no disabled=yes
set [ find default-name=ether20 ] disable-running-check=no disabled=yes
set [ find default-name=ether21 ] disable-running-check=no disabled=yes
set [ find default-name=ether22 ] disable-running-check=no disabled=yes
set [ find default-name=ether23 ] disable-running-check=no disabled=yes
set [ find default-name=ether24 ] disable-running-check=no disabled=yes
set [ find default-name=ether25 ] disable-running-check=no disabled=yes
set [ find default-name=ether26 ] disable-running-check=no disabled=yes
set [ find default-name=ether27 ] disable-running-check=no disabled=yes
set [ find default-name=ether28 ] disable-running-check=no disabled=yes
/disk
set slot1 slot=slot1
set slot2 slot=slot2
set slot3 slot=slot3
set slot4 slot=slot4
set slot5 slot=slot5
set slot6 slot=slot6
set slot7 slot=slot7
set slot8 slot=slot8
set slot9 slot=slot9
set slot10 slot=slot10
set slot11 slot=slot11
/interface wireless security-profiles
```

```
set [ find default=yes ] supplicant-identity=MikroTik
/port
set 0 name=serial0
/system logging action
add name=syslog remote=192.168.100.3 target=remote
/interface bridge port
add bpdu-guard=yes bridge=bridge-Lan edge=yes horizon=2 interface=ether2
add bpdu-guard=yes bridge=bridge-Lan edge=yes horizon=2 interface=ether3
add bridge=bridge-Lan interface=ether1 trusted=yes
/ip dhcp-client
add interface=bridge-Lan
/ip service
set telnet disabled=yes
set ftp disabled=yes
set www disabled=yes
set ssh address=192.168.100.0/29
set api disabled=yes
set winbox address=192.168.100.0/29
set api-ssl disabled=yes
/ip ssh
set host-key-size=4096 strong-crypto=yes
/radius
add address=192.168.100.3 service=login
/system identity
set name=Mikrotik-SW-LAN-C
/system logging
add action=syslog topics=critical
add action=syslog topics=info
add action=syslog topics=error
add action=syslog topics=warning
add action=syslog topics=account
/tool bandwidth-server
set enabled=no
/tool mac-server mac-winbox
set allowed-interface-list=none
/user aaa
set interim-update=20m use-radius=yes
```

Mikrotik-SW-LAN-D:

```
# jun/13/2026 00:23:58 by RouterOS 7.8
# software id =
#
/interface bridge
add name=bridge-Lan
/interface ethernet
set [ find default-name=ether1 ] disable-running-check=no
set [ find default-name=ether2 ] disable-running-check=no
set [ find default-name=ether3 ] disable-running-check=no
set [ find default-name=ether4 ] disable-running-check=no
set [ find default-name=ether5 ] disable-running-check=no
set [ find default-name=ether6 ] disable-running-check=no
set [ find default-name=ether7 ] disable-running-check=no
set [ find default-name=ether8 ] disable-running-check=no
set [ find default-name=ether9 ] disable-running-check=no
set [ find default-name=ether10 ] disable-running-check=no
set [ find default-name=ether11 ] disable-running-check=no
set [ find default-name=ether12 ] disable-running-check=no
set [ find default-name=ether13 ] disable-running-check=no
set [ find default-name=ether14 ] disable-running-check=no
set [ find default-name=ether15 ] disable-running-check=no
set [ find default-name=ether16 ] disable-running-check=no
set [ find default-name=ether17 ] disable-running-check=no
set [ find default-name=ether18 ] disable-running-check=no
set [ find default-name=ether19 ] disable-running-check=no
set [ find default-name=ether20 ] disable-running-check=no
set [ find default-name=ether21 ] disable-running-check=no
set [ find default-name=ether22 ] disable-running-check=no
set [ find default-name=ether23 ] disable-running-check=no
set [ find default-name=ether24 ] disable-running-check=no
set [ find default-name=ether25 ] disable-running-check=no
set [ find default-name=ether26 ] disable-running-check=no
set [ find default-name=ether27 ] disable-running-check=no
```

```
set [ find default-name=ether28 ] disable-running-check=no
/disk
set slot1 slot=slot1
set slot2 slot=slot2
set slot3 slot=slot3
set slot4 slot=slot4
set slot5 slot=slot5
set slot6 slot=slot6
set slot7 slot=slot7
set slot8 slot=slot8
set slot9 slot=slot9
set slot10 slot=slot10
set slot11 slot=slot11
set slot12 slot=slot12
/interface wireless security-profiles
set [ find default=yes ] supplicant-identity=MikroTik
/port
set 0 name=serial0
/system logging action
add name=syslog remote=192.168.100.3 target=remote
/interface bridge port
add bridge=bridge-Lan interface=ether1
add bridge=bridge-Lan interface=ether2
add bridge=bridge-Lan interface=ether3
/ip dhcp-client
add interface=bridge-Lan
/ip service
set telnet disabled=yes
set ftp disabled=yes
set www disabled=yes
set ssh address=192.168.100.0/29
set api disabled=yes
set winbox address=192.168.100.0/29
set api-ssl disabled=yes
/ip ssh
set host-key-size=4096 strong-crypto=yes
/radius
add address=192.168.100.3 service=login
/system identity
set name=Mikrotik-SW-LAN-D
/system logging
add action=syslog topics=info
add action=syslog topics=critical
add action=syslog topics=error
add action=syslog topics=account
add action=syslog topics=warning
/tool bandwidth-server
set enabled=no
/tool mac-server mac-winbox
set allowed-interface-list=none
/user aaa
set interim-update=20m use-radius=yes
```

Додаток Д

Лістинг «radius-parsing-pipeline»

```
{
 "description": "Radius Detail",
 "processors": [
  {
   "dissect": {
    "field": "message",
    "pattern": "%{log_timestamp}\n\t%{kv_payload}",
    "ignore_failure": true
   }
  },
  {
   "gsub": {
    "field": "kv_payload",
    "pattern": "[\\n\\t]",
    "replacement": "|"
   }
  },
  {
   "kv": {
    "field": "kv_payload",
    "field_split": "\\|\\|",
    "value_split": " = ",
    "trim_key": " ",
    "trim_value": "  \"",
    "ignore_failure": true
   }
  },
  {
   "remove": {
    "field": ["message", "kv_payload"],
    "ignore_missing": true
   }
  }
 ]
}
```