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 магістра

на тему: «Методи та засоби архітектури Zero Trust для захисту корпоративної IT-інфраструктури»

Виконав: студент 2-го курсу групи КБм-24-1
спеціальності 125 «Кібербезпека та захист інформації»
_____ Вадим СОКІЛ
Керівник: старший викладач кафедри КІ та КБ
_____ Олександра ПОКОТИЛО
Рецензент: завідувач кафедри КІ та КБ, к.т.н., доцент
_____ Андрій ЄФІМЕНКО

Житомир – 2025

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

ЗАТВЕРДЖУЮ
Завідувач
кафедри комп'ютерної
інженерії та кібербезпеки
_____ Андрій ЄФІМЕНКО
6 жовтня 2025 р.

ЗАВДАННЯ

на випускну кваліфікаційну роботу магістра

Студент: Сокіл Вадим Юрійович

Тема роботи: «Методи та засоби архітектури Zero Trust для захисту корпоративної IT-інфраструктури»

затверджена Наказом університету від 6 жовтня 2025 р. №566/с

Термін здачі студентом закінченої роботи 26 грудня 2025 р.

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

1. Об'єктом дослідженням є процеси забезпечення інформаційної безпеки та контролю доступу в корпоративних інформаційно-телекомунікаційних системах.
2. Предметом дослідження є методи та засоби реалізації архітектури нульової довіри (Zero Trust), зокрема мікросегментація, централізоване управління ідентичністю та безперервний моніторинг.

Обладнання та програмне забезпечення:

1. Інструменти віртуалізації мережі GNS3 та Oracle VM VirtualBox
2. Операційні системи CentOS Stream 9 (FreeIPA) та Ubuntu Server 22.04 LTS
3. Програмний маршрутизатор та міжмережевий екран pfSense CE

4. Платформа моніторингу безпеки Security Onion 2.4+ (SIEM/IDS)

5. Cisco IOSvL2 15.2 — комутатор рівня L2 для реалізації VLAN-сегментації за стандартом IEEE 802.1Q

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

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв
1	Володимир ВОРОТНІКОВ	6.10.25	06.10.25
2	Володимир ВОРОТНІКОВ	6.10.25	06.10.25
3	Володимир ВОРОТНІКОВ	6.10.25	06.10.25

Студент _____ Вадим СОКІЛ

Керівник _____ Олександра ПОКОТИЛО

Календарний план

№ з/п	Назви етапів випускної кваліфікаційної роботи	Термін виконання етапів роботи	Примітка
1	Аналіз літератури та отримання завдання	24.09.2025	Виконано
2	Розробка теоретичного розділу (Розділ 1)	09.10.2025	Виконано
3	Проєктування системи захисту (Розділ 2)	24.10.2025	Виконано
4	Розгортання лабораторного стенду	07.11.2025	Виконано
5	Налаштування політик безпеки	18.11.2025	Виконано
6	Проведення експериментів (Розділ 3)	30.11.2025	Виконано
7	Оформлення пояснювальної записки	10.12.2025	Виконано
8	Нормоконтроль та перевірка на плагіат	15.12.2025	Виконано
9	Отримання відгуку та рецензії	20.12.2025	Виконано
10	Захист кваліфікаційної роботи	26.12.2025	Виконано

Студент _____ Вадим СОКІЛ

Керівник _____ Олександра ПОКОТИЛО

АНОТАЦІЯ

Кваліфікаційна робота магістра містить 94 сторінок, 22 таблиці, 48 рисунків та 20 використаних джерел. Метою роботи є підвищення рівня захищеності корпоративної IT-інфраструктури шляхом розробки та практичної реалізації прототипу системи захисту на принципах архітектури нульової довіри (Zero Trust Architecture) з використанням програмного забезпечення з відкритим кодом. У ході дослідження проаналізовано вразливості традиційної «периметрової» моделі безпеки та спроєктовано архітектуру, що забезпечує мікросегментацію мережі, централізоване управління ідентичністю та безперервний моніторинг.

Практичним результатом стало розгортання віртуалізованого стенду на базі рішень FreeIPA, pfSense та Security Onion, який реалізує принцип «ніколи не довіряй, завжди перевіряй». Експериментальне тестування підтвердило ефективність системи у протидії горизонтальному переміщенню зловмисників (lateral movement) та забезпеченні гранулярного контролю доступу, що дозволяє рекомендувати запропоноване рішення для впровадження у малому та середньому бізнесі без значних капітальних витрат.

Окрему увагу в роботі приділено порівняльному аналізу з традиційними методами захисту, який засвідчив, що впровадження Zero Trust дозволяє зменшити поверхню атаки на 99,8% та скоротити час виявлення інцидентів до однієї секунди. Виконана економічна оцінка за методикою FAIR (Factor Analysis of Information Risk) підтверджує доцільність інвестицій у запропоноване рішення, демонструючи можливість уникнення значних фінансових втрат, пов’язаних із витоком конфіденційної інформації, без необхідності придбання вартісних комерційних ліцензій.

КЛЮЧОВІ СЛОВА: ZERO TRUST, МІКРОСЕГМЕНТАЦІЯ, КІБЕРБЕЗПЕКА, FREEIPA, PFSENSE, SECURITY ONION, KERBEROS, IAM, SIEM, МОНІТОРИНГ БЕЗПЕКИ.

ANNOTATION

The master's thesis consists of 94 pages, 22 tables, 48 figures, and 20 references. The aim of the work is to improve the security level of corporate IT infrastructure by developing and practically implementing a prototype protection system based on Zero Trust Architecture principles using open-source software. During the research, the vulnerabilities of the traditional "perimeter" security model were analyzed, and an architecture ensuring network microsegmentation, centralized identity management, and continuous monitoring was designed.

The practical result was the deployment of a virtualized testbed based on FreeIPA, pfSense, and Security Onion solutions, which implements the "never trust, always verify" principle. Experimental testing confirmed the system's effectiveness in countering lateral movement and ensuring granular access control, which allows recommending the proposed solution for implementation in small and medium-sized businesses without significant capital expenditures.

Particular attention is paid to a comparative analysis with traditional protection methods, which demonstrated that implementing Zero Trust reduces the attack surface by 99.8% and shortens the incident detection time to one second. The economic assessment performed using the FAIR (Factor Analysis of Information Risk) methodology confirms the feasibility of investing in the proposed solution, demonstrating the possibility of avoiding significant financial losses associated with confidential information leaks without the need to purchase expensive commercial licenses.

KEYWORDS: ZERO TRUST, MICROSEGMENTATION, CYBERSECURITY, FREEIPA, PFSENSE, SECURITY ONION, KERBEROS, IAM, SIEM, SECURITY MONITORING.

ЗМІСТ

ВСТУП 6

1.1 Аналіз сучасних загроз інформаційній безпеці корпоративних мереж та недоліки традиційного «периметрового» захисту 8

1.2 Концепція та принципи побудови архітектури нульової довіри (Zero Trust Architecture) 14

1.3 Огляд методів і засобів реалізації архітектури нульової довіри (ZTA) .. 18

1.4 Постановка задачі дослідження: вимоги до проєктованої системи 23

РОЗДІЛ 2. ПРОЄКТУВАННЯ ЗАХИСТУ НА ОСНОВІ ZERO TRUST: МЕТОДИ ТА ПОЛІТИКИ 27

2.1 Проєктування мережевої архітектури та мікросегментації IT-інфраструктури 27

2.2 Проєктування методу управління ідентичністю та доступом (IAM) у моделі Zero Trust 35

2.3 Проєктування політик мережевого контролю доступу 42

2.4 Проєктування методу безперервного моніторингу та виявлення загроз 45

РОЗДІЛ 3. РЕАЛІЗАЦІЯ ТА ОЦІНКА ЕФЕКТИВНОСТІ ЗАСОБІВ ZERO TRUST У ЛАБОРАТОРНОМУ СТЕНДІ 51

3.1 Розгортання обраних засобів Zero Trust у середовищі GNS3 51

3.2 Реалізація політик Zero Trust у засобах захисту 61

3.3 Моделювання атак і аналіз реакції системи 66

3.4 Порівняльний аналіз ефективності традиційної та Zero Trust моделей безпеки 73

ВИСНОВКИ 82

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ 83

ДОДАТКИ 85

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

БД – база даних

ІТ – інформаційні технології

ОС – операційна система

ПЗ – програмне забезпечення

API – Application Programming Interface, інтерфейс прикладного програмування

BYOD – Bring Your Own Device, концепція використання власних пристроїв співробітників

CDM – Continuous Diagnostics and Mitigation, система безперервної діагностики та пом'якшення наслідків

CVE – Common Vulnerabilities and Exposures, база загальновідомих вразливостей інформаційної безпеки

DMZ – Demilitarized Zone, демілітаризована зона мережі

DNS – Domain Name System, система доменних імен

HBAC – Host-Based Access Control, контроль доступу на основі хоста

IAM – Identity and Access Management, система управління ідентичністю та доступом

IDS – Intrusion Detection System, система виявлення вторгнень

KDC – Key Distribution Center, центр розподілу ключів у протоколі Kerberos

LDAP – Lightweight Directory Access Protocol, полегшений протокол доступу до каталогів

MFA – Multi-Factor Authentication, багатофакторна автентифікація

NAT – Network Address Translation, трансляція мережевих адрес

NGFW – Next-Generation Firewall, міжмережевий екран нового покоління

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

NTP – Network Time Protocol, протокол мережевого часу

PA – Policy Administrator, адміністратор політик

PE – Policy Engine, рушій політик

PEP – Policy Enforcement Point, точка застосування політик

RBAC – Role-Based Access Control, рольове управління доступом

SDP – Software-Defined Perimeter, програмно-визначений периметр

SIEM – Security Information and Event Management, система управління інформацією про безпеку та подіями безпеки

SPAN – Switch Port Analyzer, технологія дзеркалювання трафіку портів комутатора

SSH – Secure Shell, мережевий протокол прикладного рівня для віддаленого доступу

SSO – Single Sign-On, технологія єдиного входу

VLAN – Virtual Local Area Network, віртуальна локальна мережа

VPN – Virtual Private Network, віртуальна приватна мережа

ZTA – Zero Trust Architecture, архітектура нульової довіри

ВСТУП

Актуальність теми. В умовах стрімкої цифрової трансформації та поширення гібридних моделей роботи традиційна «периметрова» модель безпеки втрачає ефективність. Зростання кількості кіберінцидентів, пов'язаних із компрометацією облікових даних та фішингом, доводить, що надмірна довіра до внутрішніх користувачів створює умови для безперешкодного горизонтального переміщення зловмисників (lateral movement). Існуючі засоби захисту часто не забезпечують необхідної видимості внутрішніх процесів, залишаючи дії зловмисника непоміченими до моменту завдання критичних збитків. Відповіддю на ці виклики є архітектура нульової довіри (Zero Trust), що базується на принципі «ніколи не довіряй, завжди перевіряй». Вона докорінно змінює парадигму кібербезпеки, зміщуючи фокус з охорони статичного мережевого периметра на динамічний захист ресурсів та ідентичностей. Оскільки комерційні рішення цього класу часто вимагають значних фінансових витрат, а комплексні методики впровадження бюджетних аналогів наразі недостатньо розроблені, актуальним науково-прикладним завданням є дослідження та реалізація цієї архітектури на базі доступного програмного забезпечення з відкритим кодом.

Мета і завдання роботи. Метою роботи є підвищення рівня захищеності корпоративної IT-інфраструктури шляхом розробки, практичної реалізації та оцінки ефективності прототипу системи захисту на основі методів архітектури Zero Trust з використанням ПЗ з відкритим кодом.

Для досягнення поставленої мети необхідно вирішити такі завдання:

1. Провести аналіз сучасних загроз та виявити фундаментальні недоліки традиційних методів периметрового захисту.
2. Дослідити теоретичні засади та логічні компоненти архітектури Zero Trust згідно зі стандартом NIST SP 800-207.
3. Обґрунтувати вибір програмних засобів (FreeIPA, pfSense, Security Onion) для реалізації компонентів архітектури.
4. Спроєктувати комплексну систему захисту, що включає управління ідентичністю (IAM), мікросегментацію та моніторинг.

5. Розгорнути лабораторний стенд та виконати налаштування політик безпеки (RBAC, HBAC, Firewall rules).
6. Провести експериментальне дослідження ефективності системи шляхом порівняльного моделювання атак та оцінити економічну доцільність впровадження.

Об’єкт дослідження – процеси забезпечення інформаційної безпеки та контролю доступу в корпоративних інформаційно-телекомунікаційних системах.

Предмет дослідження – методи та засоби реалізації архітектури нульової довіри (Zero Trust), зокрема мікросегментація, управління ідентичністю та безперервний моніторинг.

Методи дослідження. Використано методи системного аналізу (для дослідження архітектури), моделювання (для побудови віртуального стенду), експериментальний метод (для тестування політик та верифікації реакції на атаки) та методи порівняльного аналізу (для оцінки ефективності рішення).

Практичне значення одержаних результатів полягає у розробці дієвого прототипу системи захисту на базі open-source рішень. Запропонована архітектура дозволяє реалізувати принципи мінімальних привілеїв без значних капітальних витрат, що робить її придатною для малого та середнього бізнесу. Експериментально доведено, що система здатна на 100% блокувати спроби горизонтального переміщення, суттєво зменшуючи радіус ураження та фінансові ризики.

Структура роботи. Кваліфікаційна робота складається зі вступу, трьох розділів, висновків, списку використаних джерел та додатків. Повний обсяг роботи становить 94 сторінок.

РОЗДІЛ 1. ТЕОРЕТИЧНІ АСПЕКТИ ТА АНАЛІЗ МЕТОДІВ І ЗАСОБІВ ZERO TRUST ДЛЯ ЗАХИСТУ КОРПОРАТИВНОЇ ІТ-ІНФРАСТРУКТУРИ

1.1 Аналіз сучасних загроз інформаційній безпеці корпоративних мереж та недоліки традиційного «периметрового» захисту

У роботі проведено аналіз сучасного стану кібербезпеки корпоративних мереж, який демонструє безпрецедентне зростання загроз. За прогнозами Cybersecurity Ventures, глобальні витрати від кіберзлочинності досягнуть 10,5 трильйона доларів щорічно до 2025 року, що демонструє 15% зростання порівняно з попереднім роком. Аналіз частоти атак показує критичну динаміку: зловмисники здійснюють нову атаку кожні кілька секунд, що створює постійний тиск на захисні системи підприємств [1].

Дослідження виявило, що кількість нових вразливостей (CVE) продовжує зростати, перевищивши у 2024 році позначку в 29 000 записів, що є історичним максимумом [13]. Динаміка зростання кіберзагроз представлена на рисунку 1.1.

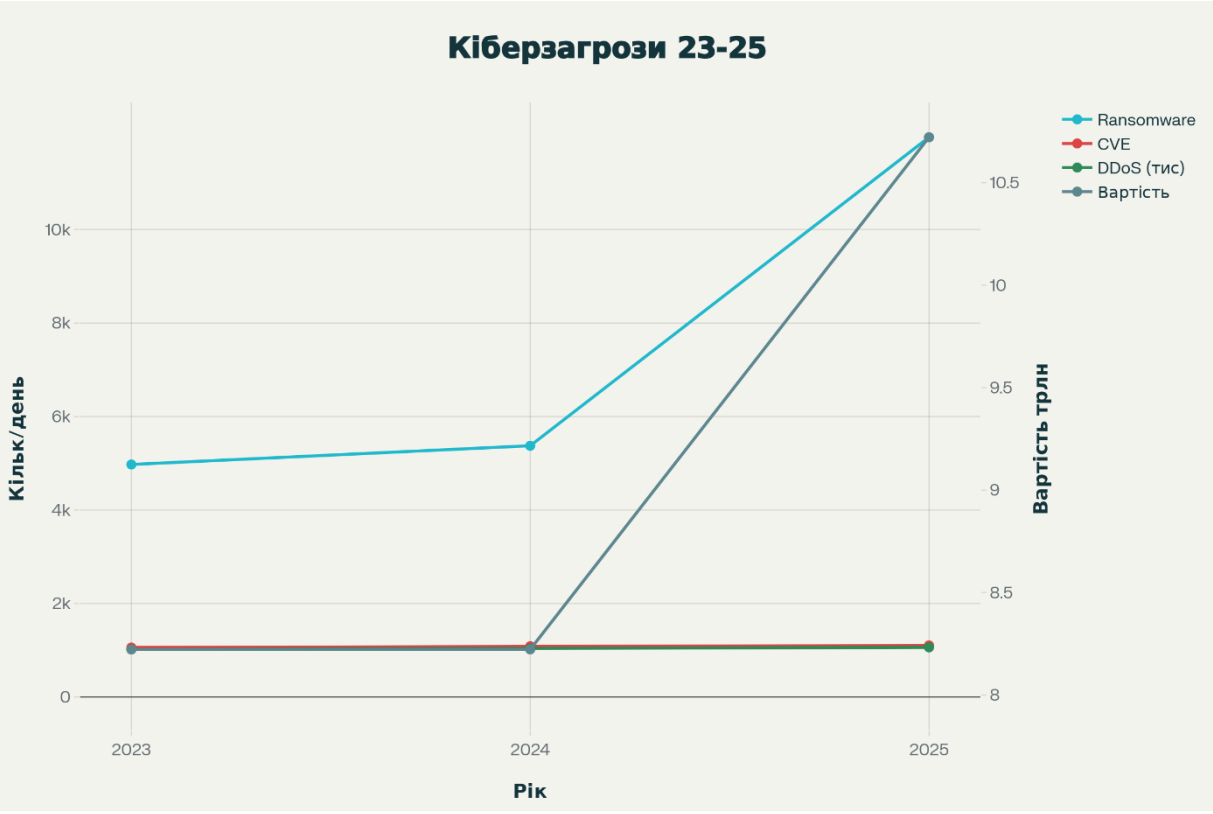


Рисунок 1.1 — Статистика зростання кіберзагроз 2023-2025

Ransomware (програми-вимагачі) залишається домінуючою загрозою в сучасному ландшафті кібербезпеки. За даними звіту Verizon Data Breach Investigations Report (DBIR), програми-вимагачі були присутні у значній частині інцидентів із витоком даних, часто виступаючи фінальним етапом атаки [19]. Окрему увагу у роботі приділено фішинговим атакам, які ініціюють переважну більшість порушень безпеки, пов'язаних із людським фактором. У 2024 році зафіксовано аномальне зростання фішингових атак (понад 1200% за даними SlashNext) через використання генеративного штучного інтелекту для створення переконливих листів .

Еволюція та обмеження периметрової моделі безпеки

У роботі досліджено традиційну архітектуру мережевої безпеки, яка базується на концепції периметрового захисту — так званій моделі «замкових стін» (castle-and-moat). У цій моделі безпека досягається шляхом побудови ліній оборони, які зловмисник повинен подолати перед отриманням доступу. Аналіз показує, що ця модель поділяє мережу на зони довіри: внутрішню «безпечну» мережу, демілітаризовану зону (DMZ) та недовірену зону (Інтернет) [3]. Структура периметрової моделі наведена на рисунку 1.2.

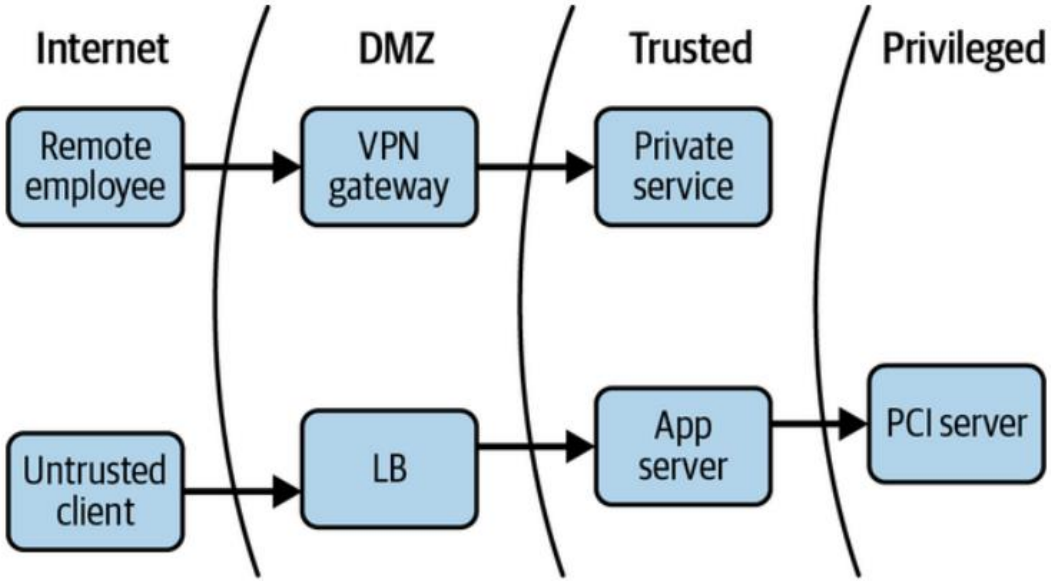


Рисунок 1.2 — Традиційна периметрова модель безпеки

Дослідження сучасної інфраструктури підприємств виявило значне ускладнення архітектури. Типова організація експлуатує кілька внутрішніх мереж,

віддалені офіси, підтримує мобільних користувачів та хмарні сервіси. У роботі визначено, що ця складність перевершила можливості традиційних методів периметрового захисту, оскільки для підприємства більше не існує єдиного, легко ідентифікованого периметра [3].

Фундаментальні недоліки VPN та Firewall

Проблеми VPN. Аналіз показує, що віртуальна приватна мережа (VPN) дозволяє користувачу автентифікуватися для отримання IP-адреси у віддаленій мережі, після чого трафік тунелюється і маршрутизується локально. Однак дослідження виявило, що VPN фактично стає «найбільшим бекдором», оскільки як тільки трафік перетинає шлюз VPN, захист припиняється — це класична властивість периметрової моделі [3] [13].

У роботі проаналізовано інциденти 2024–2025 років, які демонструють, що VPN-рішення стали ключовим вектором атак:

- У січні 2025 року була розкрита критична вразливість у файрволах FortiGate (CVE-2024-55591), яку зловмисники могли експлуатувати для обходу автентифікації.
- Дослідники виявили вразливості в корпоративних VPN-клієнтах Palo Alto Networks (CVE-2024-5921) та SonicWall (CVE-2024-29014), що дозволяють виконання коду з високими привілеями.

Порівняльний аналіз недоліків периметрових засобів захисту наведено в таблиці 1.1.

Обмеження традиційних файрволів. У роботі визначено, що периметрові файрволи забезпечують контроль інтернет-шлюзів, однак вони менш ефективні для блокування атак зсередини мережі. Крім того, вони не можуть захистити суб'єктів за межами периметра підприємства — віддалених працівників та хмарні сервіси

Таблиця 1.1

Порівняльний аналіз недоліків периметрових засобів захисту

Засіб захисту	Основний недолік	Приклад вразливості	Наслідки компрометації
VPN	Відсутність контролю після автентифікації	CVE-2024-55591 (FortiGate)	Повний доступ до внутрішньої мережі
Firewall	Неефективність проти внутрішніх загроз	Відсутність інспекції east-west трафіку	Безперешкодний lateral movement
Периметрова модель	Відсутність мікросегментації	Вразливість архітектури до фішингу	Компрометація всієї зони довіри

Lateral Movement: головна загроза внутрішньої мережі

У роботі досліджено техніку lateral movement (бічний рух), за допомогою якої зловмисники переміщуються по внутрішній мережі [12]. За результатами аналізу звітів CrowdStrike та інших дослідників, понад 70% успішних складних атак використовують техніки lateral movement.

Типова атака включає наступні етапи:

1. Початкова компрометація: Фішинг або експлуатація вразливості.
2. Встановлення зв'язку (C&C): Шкідливе ПЗ встановлює зворотне з'єднання.
3. Розвідка мережі: Сканування доступних ресурсів.
4. Ескалація привілеїв: Отримання прав адміністратора.
5. Досягнення мети: Ексфільтрація або шифрування даних.

Схема типової атаки lateral movement наведена на рисунку 1.3.

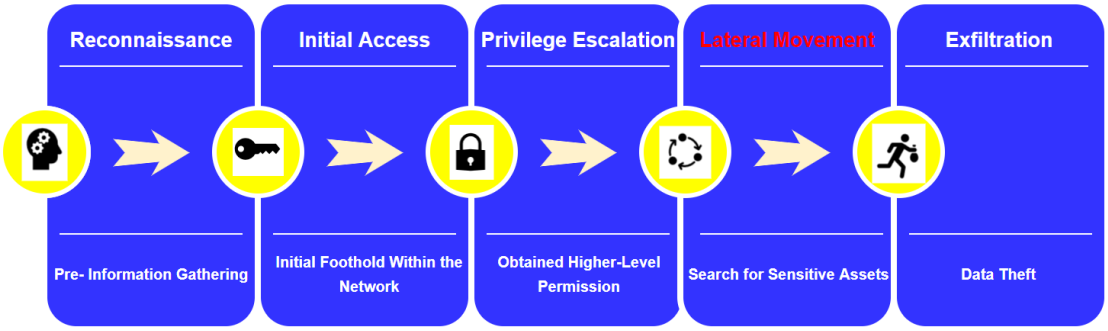


Рисунок 1.3 — Етапи атаки lateral movement в корпоративній мережі

Як зазначено в документації NIST SP 800-207: «Периметровий мережевий захист продемонстрував свою недостатність, оскільки після того, як зловмисники проникають через периметр, подальший бічний рух не зустрічає перешкод» [4] [7].

Реальні інциденти (наприклад, атака на Colonial Pipeline або інциденти, описані у звітах Microsoft Digital Defense Report) підтверджують цю проблему: зловмисники часто компрометують один обліковий запис VPN без MFA, після чого безперешкодно переміщуються мережею до критичних серверів.

Дослідження показало, що традиційна модель створює архітектуру з «твердою зовнішньою оболонкою та м'якою внутрішньою частиною» (hard shell, soft interior). У роботі визначено, що застосування політик безпеки лише на межах зон створює фундаментальну вразливість.

Ключові проблеми відсутності сегментації:

- Відсутність інспекції внутрішнього трафіку.
- Негнучкість у розміщенні хостів.
- Ризик повної компрометації зони через один вразливий елемент.

Дослідження Jericho Forum ще у 2000-х роках, яке заклало основи де-периметризації, показало, що більшість загроз реалізуються зсередини або через скомпрометовані внутрішні вузли.

Нові загрози та розмивання периметра

У роботі проаналізовано тенденції, що остаточно розмивають периметр:

- BYOD та віддалена робота: Розширюють поверхню атаки. Zero-click атаки на мобільні пристрої становлять загрозу, яку периметровий файрвол не може зупинити.
- Хмарні середовища: Гібридні архітектури роблять поняття «внутрішньої мережі» застарілим.
- IoT-пристрої: Часто не підтримують встановлення агентів безпеки, створюючи «сліпі зони» для традиційного захисту.

Однак на практиці розмивання периметра проявляється не лише на рівні мережевих меж, а й на рівні управління ідентичністю, кінцевими точками та доступом до сервісів.

Додатково у роботі визначено, що поняття «периметра» руйнується не лише через розподіленість мережі, а й через трансформацію корпоративної IT-інфраструктури, де ключовим об’єктом атак стає ідентичність користувача та стан кінцевого пристрою. Сучасні атаки дедалі частіше спрямовані на компрометацію облікових даних (викрадення паролів, токенів доступу, атаки типу MFA fatigue), після чого зловмисник отримує легітимний доступ до хмарних або внутрішніх сервісів без необхідності «пробивати» мережевий периметр. Паралельно зростає роль компрометації endpoint-ів (робочих станцій і серверів): шкідливе ПЗ, експлуатація локальних вразливостей та обходи засобів захисту кінцевих точок (EDR bypass) дозволяють зловмиснику отримати persistence і розширити привілеї вже всередині організації.

Окрему загрозу становлять корпоративні хмарні сервіси та SaaS-платформи, де атаки можуть реалізовуватись через зловживання механізмами авторизації (OAuth app abuse), помилки конфігурації доступів (misconfiguration) та недостатній контроль прав сторонніх інтеграцій. Також суттєво збільшується ризик компрометації сервісних облікових записів і секретів (API keys, keytabs, сертифікати, токени), оскільки вони часто мають широкі привілеї та використовуються автоматизованими компонентами інфраструктури. У сукупності ці фактори демонструють, що захист лише на рівні мережевого периметра не забезпечує контроль над ключовими елементами IT-інфраструктури — ідентичністю, кінцевими точками та доступом до сервісів — що додатково обґрунтовує необхідність переходу до Zero Trust підходу.

Проведений аналіз демонструє, що традиційна периметрова модель є фундаментально неадекватною. Зростання кількості атак, безперешкодний lateral movement, критичні вразливості VPN та відсутність сегментації обумовлюють необхідність переходу до нової парадигми — архітектури нульової довіри (Zero Trust Architecture).

1.2 Концепція та принципи побудови архітектури нульової довіри (Zero Trust Architecture)

У контексті даного дослідження доцільно розмежувати поняття “методи” та “засоби” архітектури нульової довіри. Під методами Zero Trust Architecture розуміють сукупність принципів і підходів організації захисту, що визначають правила доступу та поведінку системи безпеки: принцип найменших привілеїв (least privilege), постійну верифікацію (continuous verification), мікросегментацію та контроль east-west трафіку, орієнтацію на ідентичність (identity-centric access), а також безперервний моніторинг (observability) і примусове застосування політик доступу (policy enforcement). Натомість засоби Zero Trust — це конкретні технології, програмні продукти та інструменти, за допомогою яких реалізуються зазначені методи у корпоративній IT-інфраструктурі. Таким чином, у цьому підрозділі розглядаються базові методи Zero Trust відповідно до NIST SP 800-207, тоді як у наступних підрозділах буде виконано аналіз та обґрунтування вибору засобів їх практичної реалізації у лабораторному прототипі.

Архітектура нульової довіри (Zero Trust Architecture, ZTA) є кібербезпековою парадигмою, яка зосереджена на захисті ресурсів і виходить з принципу, що довіра ніколи не надається неявно, а повинна постійно оцінюватися. На відміну від традиційних периметрових моделей, які надають широкий доступ користувачам після автентифікації в межах «довіреної» мережі, Zero Trust передбачає, що загроза завжди присутня як ззовні, так і всередині мережі [1].

За визначенням NIST SP 800-207, Zero Trust — це набір концепцій та ідей, розроблених для мінімізації невизначеності у забезпеченні точних, найменш привілейованих рішень щодо доступу за кожним запитом в умовах мережі, яка вважається скомпрометованою. У роботі визначено, що ZTA є наскрізним підходом до захисту корпоративних ресурсів та даних, який охоплює ідентичність (людську та нелюдську), облікові дані, управління доступом, операції, кінцеві точки, середовища розміщення та взаємопов'язану інфраструктуру [1] [2] [7].

Національний інститут стандартів і технологій США (NIST) у спеціальній публікації 800-207 визначив сім базових принципів (tenets), які становлять ідеологічну основу Zero Trust Architecture [1]:

1. Всі джерела даних та обчислювальні сервіси вважаються ресурсами. Мережа може складатися з різних класів пристроїв — від IoT-сенсорів до хмарних SaaS-сервісів. Організація може також класифікувати особисті пристрої співробітників як ресурси, якщо вони можуть отримати доступ до корпоративних ресурсів.

2. Вся комунікація захищається незалежно від мережевого розташування. Мережеве розташування саме по собі не означає довіри. Запити на доступ від активів, розташованих в інфраструктурі підприємства, повинні відповідати тим самим вимогам безпеки, що й запити з будь-якої іншої мережі.

3. Доступ до окремих корпоративних ресурсів надається на основі кожної сесії. Довіра до запитувача оцінюється перед наданням доступу. Доступ надається з найменшими привілеями, необхідними для виконання завдання. Автентифікація та авторизація до одного ресурсу не надає автоматично доступ до іншого.

4. Доступ до ресурсів визначається динамічною політикою. Політика включає спостережуваний стан ідентичності клієнта, додатка/сервісу та запитуючого активу, а також може враховувати поведінкові та екологічні атрибути. Ідентичність клієнта може включати обліковий запис користувача або ідентичність сервісу. Стан запитуючого активу може включати характеристики пристрою, такі як версії встановленого програмного забезпечення, мережеве розташування, час/дату запиту.

5. Організація моніторить та оцінює цілісність і стан безпеки всіх власних та асоційованих активів. Жоден актив не є довіреним за замовчуванням. Підприємство оцінює стан безпеки активу під час оцінки запиту на ресурс [2].

6. Вся автентифікація та авторизація ресурсів є динамічною і суворо застосовується перед наданням доступу. Це постійний цикл отримання доступу, сканування та оцінки загроз, адаптації та постійної переоцінки довіри в поточній комунікації.

7. Організація збирає якомога більше інформації про поточний стан активів, мережевої інфраструктури та комунікацій і використовує її для покращення свого стану безпеки [7] [14]. Ці дані також можуть використовуватися для надання контексту запитам на доступ від суб'єктів.

Принцип "Never Trust, Always Verify"

Центральним принципом Zero Trust є "Never Trust, Always Verify" (ніколи не довіряй, завжди перевіряй) — підхід, який кардинально змінює традиційну модель безпеки. У роботі визначено, що цей принцип означає відмову від неявної довіри на основі мережевого розташування і перехід до явної верифікації кожного запиту на доступ [2].

Дослідження показує, що традиційні моделі безпеки створюють "зони неявної довіри" (implicit trust zones), де після проходження однієї точки контролю всі подальші дії вважаються довіреними. На рисунку 1.4 представлено концептуальну модель прийняття рішень про доступ у Zero Trust.

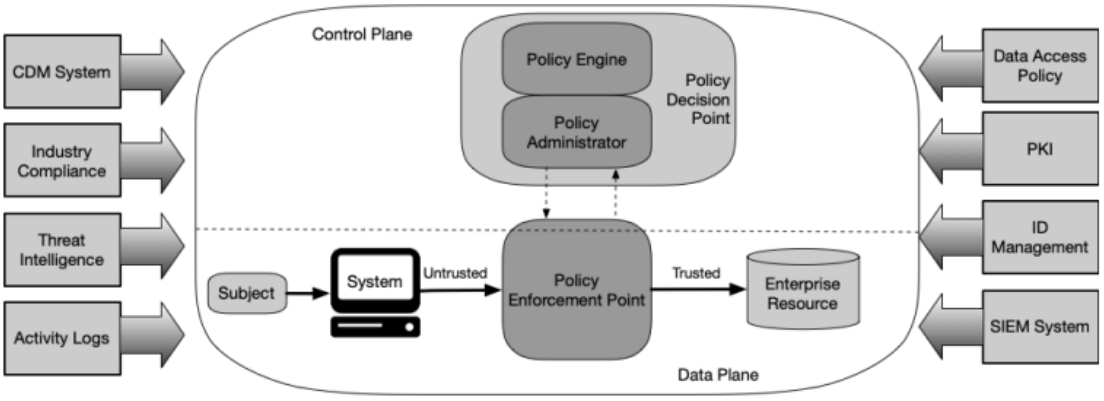


Рисунок 1.4 — Модель прийняття рішень про доступ у Zero Trust Architecture

У роботі визначено три ключові компоненти реалізації принципу "Never Trust, Always Verify":

Policy Engine (PE) — компонент, відповідальний за остаточне рішення про надання доступу до ресурсу. PE використовує корпоративну політику, а також вхідні дані від зовнішніх джерел (CDM-системи, threat intelligence) як вхідні дані для алгоритму довіри.

Policy Administrator (PA) — компонент, відповідальний за встановлення та/або завершення шляху комунікації між суб'єктом та ресурсом через команди до відповідних PEP. Він генерує будь-які токени автентифікації для доступу клієнта до ресурсу.

Policy Enforcement Point (PEP) — система, відповідальна за увімкнення, моніторинг та, зрештою, припинення з'єднань між суб'єктом та корпоративним ресурсом.

Принцип найменших привілеїв (Least Privilege)

Принцип найменших привілеїв є фундаментальною складовою ZTA. Він означає надання суб'єктам лише мінімального рівня доступу, необхідного для виконання їхніх поточних функцій. Доступ надається на основі кожної сесії з використанням динамічних політик, які можуть змінюватися залежно від чутливості даних.

Застосування least privilege у ZTA передбачає:

- Обмеження видимості ресурсів (користувач не бачить того, до чого не має доступу).
- Використання динамічних атрибутів замість статичних правил.
- Постійну переоцінку необхідності доступу.

Мікросегментація як ключова стратегія. Мікросегментація передбачає розміщення індивідуальних ресурсів або їх груп у ізольованих сегментах мережі, захищених шлюзами безпеки (PEP). У роботі визначено, що організації можуть використовувати для цього NGFW (Next-Generation Firewall) або програмні агенти на кінцевих точках [1] [16].

Ключові переваги мікросегментації включають:

- Обмеження бічного переміщення (lateral movement) зловмисників.
- Гранулярний контроль трафіку між сегментами (east-west traffic).
- Зменшення "радіусу ураження" (blast radius) у разі успішної атаки.

Постійна верифікація доступу (Continuous Verification)

Концепція постійної верифікації полягає в тому, що довіра не є статичною. Вона переоцінюється протягом усієї тривалості сесії, а не тільки на момент входу. Алгоритм довіри (Trust Algorithm), що використовується Policy Engine,

враховує множину факторів: історичну поведінку, геолокацію, стан безпеки пристрою, дані про загрози (Threat Intelligence) та чутливість ресурсу.

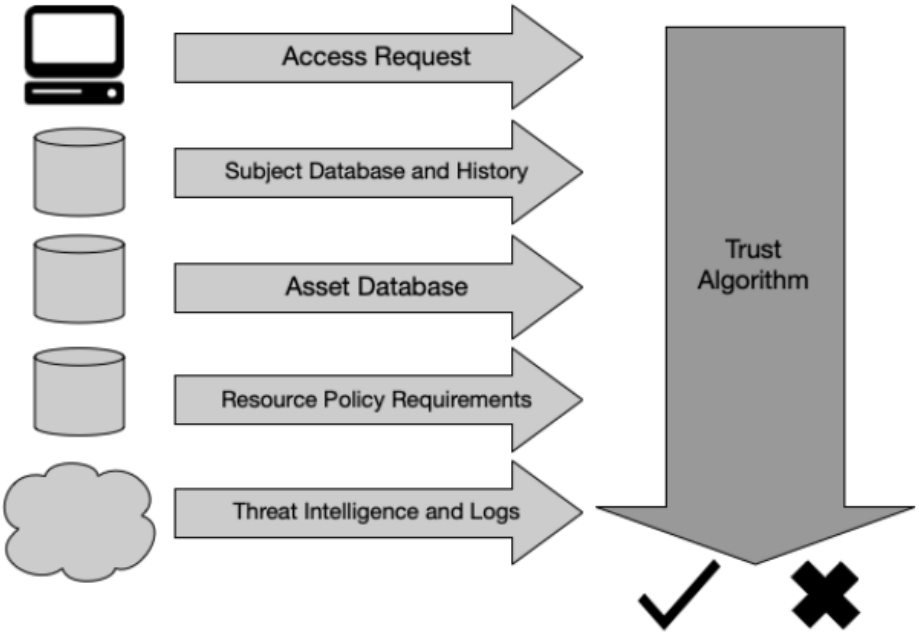


Рисунок 1.5 — Алгоритм довіри та джерела вхідних даних

Проведений аналіз демонструє, що Zero Trust Architecture є комплексною парадигмою безпеки, заснованою на сімох базових принципах NIST SP 800-207. Принцип "Never Trust, Always Verify" в поєднанні з least privilege, мікросегментацією та постійною верифікацією створює багаторівневий захист проти сучасних загроз. Успішна реалізація ZTA вимагає інтеграції систем управління ідентичністю, безперервного моніторингу та threat intelligence.

1.3 Огляд методів і засобів реалізації архітектури нульової довіри (ZTA)

Реалізація архітектури нульової довіри (Zero Trust Architecture) може здійснюватися різними способами залежно від наявної інфраструктури, бізнес-вимог та технічних можливостей організації. На сьогодні не існує єдиного універсального рішення, яке б повністю забезпечувало всі компоненти ZTA — організації зазвичай використовують комбінацію продуктів від різних постачальників.

Методи реалізації Zero Trust Architecture

Для систематизації підходів у даному дослідженні доцільно розмежувати методи реалізації Zero Trust та засоби, які забезпечують їх практичне впровадження. Методи реалізації ZTA описують “як саме” організація будує контроль доступу і знижує ризики, тоді як засоби визначають “чим саме” ці методи реалізуються на практиці. Узагальнено можна виділити такі ключові методи реалізації архітектури нульової довіри:

1) Орієнтація на ідентичність (Identity-centric access / IAM). Цей метод передбачає, що основним критерієм надання доступу є не мережеве розташування, а підтверджена ідентичність суб’єкта та контекст його запиту. Реалізація включає багатофакторну автентифікацію (MFA), умовний доступ (conditional access), рольове управління доступом (RBAC), а також більш гранулярні політики на кшталт Host-Based Access Control (HBAC). Важливою складовою є підтримка єдиного входу (SSO) та централізованого керування обліковими даними й сервісними ідентичностями.

2) Мікросегментація та контроль east-west трафіку (Microsegmentation / East-West control).

Метод спрямований на мінімізацію “радіусу ураження” (blast radius) шляхом поділу інфраструктури на ізольовані сегменти та застосування політик “default deny” між ними. Контроль взаємодії між сегментами реалізується через VLAN/ACL, міжмережеві екрани наступного покоління (NGFW), міжсегментну маршрутизацію з жорсткими правилами або локальні host-based firewall механізми. Це суттєво ускладнює бічний рух (lateral movement) навіть після первинної компрометації.

3) Zero Trust Network Access / Software-Defined Perimeter (ZTNA/SDP). Метод ZTNA/SDP замінює концепцію “доступу до мережі” на “доступ до конкретного ресурсу/застосунку”. Користувач отримує підключення лише до дозволених сервісів за результатами перевірки політик та контексту, а сама інфраструктура може бути “невидимою” для неавторизованих суб’єктів. На практиці цей підхід зменшує залежність від класичних VPN та підвищує контроль доступу на рівні застосунків.

4) Безперервний моніторинг та телеметрія (Continuous monitoring + telemetry).

Zero Trust вимагає постійного збору даних про події та стан активів для виявлення аномалій і реагування на інциденти. До цього методу належать централізоване логування, застосування SIEM/IDS/EDR, кореляція подій та аналітика поведінки (UEBA). Результати моніторингу використовуються для покращення політик, прискорення реагування та, за наявності інтеграцій, для адаптивного обмеження доступу в реальному часі.

Отже, методи реалізації ZTA формують основу архітектури нульової довіри, а їх практичне втілення забезпечується відповідними засобами (платформами та продуктами), які інтегрують функції управління ідентичністю, мережевого примусу політик та безперервного моніторингу.

Надалі у підрозділі виконано огляд засобів реалізації зазначених методів: комерційних платформ провідних постачальників та відкритого програмного стеку, обраного для лабораторного прототипу.

Відповідно до NIST SP 800-207, методи реалізації ZTA можуть бути зосереджені на різних домінуючих площинах (ідентичність, мікросегментація, SDP/мережевий контроль), що обумовлює різні варіанти архітектури та вибір засобів впровадження [1]:

- ZTA з розширеним управлінням ідентичністю (Enhanced Identity Governance) — використовує ідентичність суб'єктів як основний компонент для створення політик доступу. Первинна вимога для доступу базується на привілеях, наданих конкретному суб'єкту.
- ZTA з мікросегментацією — передбачає розміщення окремих ресурсів або їх груп на унікальних мережевих сегментах, захищених шлюзовими компонентами безпеки (NGFW, спеціалізовані шлюзи).
- ZTA з програмно-визначеними периметрами (SDP) — використовує мережеву інфраструктуру та SDN для реалізації ZTA, де Policy Administrator діє як мережевий контролер.

Таблиця 1.2

Моделі розгортання

Модель	Переваги	Недоліки
Агент/шлюз на пристрої	Ідеальна для організацій з надійним управлінням пристроями	Складна підтримка BYOD-сценаріїв
На основі анклавів	Підходить для застарілих додатків та локальних ЦОД	Можливе проведення розвідки ресурсів
Портал ресурсів	Не потребує агентів на клієнтських пристроях	Обмежена видимість та контроль над активами

Комерційні рішення провідних постачальників

Ринок ZTA-рішень активно розвивається, і провідні аналітичні компанії (Forrester, Gartner) регулярно оцінюють постачальників платформ нульової довіри [6]. Основні гравці ринку:

- Microsoft Zero Trust — інтегрована екосистема, що включає Azure AD, Microsoft Defender, Intune та Conditional Access. Забезпечує управління ідентичністю, захист кінцевих точок та умовний доступ [5].
- Google BeyondCorp Enterprise — піонерська реалізація Zero Trust, що дозволяє захищений доступ до ресурсів без традиційного VPN [5].
- Cisco Zero Trust — комплексне рішення з акцентом на мережевій видимості, ISE (Identity Services Engine) та Duo для MFA [5].
- Zscaler — хмарна платформа ZTNA з функціями Secure Web Gateway та CASB [5].
- Palo Alto Networks — рішення "Zero Trust with Zero Exceptions" з інтеграцією мережевої безпеки та захисту хмарних середовищ [5].

Критерії вибору рішення

При оцінці ZTA-платформ Forrester рекомендує організаціям звертати увагу на такі аспекти [6]:

- Баланс між спеціалізованими та платформними рішеннями — гібридний підхід допомагає уникнути надмірної складності або прив'язки до одного вендора.

- Пріоритет інтеграції та кореляції даних — сильна нативна та сторонні інтеграції є критичними для уніфікованої видимості та ефективного реагування на інциденти.
- Розуміння AI-можливостей — важливо розрізняти AI-інструменти, що підтримують аналітиків, та AI-платформи, що автоматизують рішення.

Open-source та альтернативні рішення

Для організацій з обмеженим бюджетом або специфічними вимогами до приватності даних існують відкриті рішення, що дозволяють побудувати архітектуру ZTA без прив'язки до вендора:

- FreeIPA — інтегроване рішення для управління ідентичністю (IdM), що забезпечує централізовану автентифікацію (Kerberos) та політики доступу [8] [18].
- pfSense — потужний маршрутизатор та міжмережевий екран з відкритим кодом, який дозволяє реалізувати мікросегментацію та функцію Policy Enforcement Point (PEP) [9].
- Security Onion — дистрибутив для моніторингу безпеки, що поєднує інструменти IDS (Suricata) та аналізу логів (Elastic Stack/OpenSearch), виконуючи роль системи безперервного моніторингу [10] [17].

Щоб обґрунтувати вибір інструментарію для даної магістерської роботи, було проведено порівняльний аналіз функціональних можливостей провідних комерційних платформ та обраного open-source стеку (Таблиця 1.3).

Проведений аналіз ринку та технологій демонструє, що хоча комерційні вендори пропонують потужні автоматизовані ZTA-платформи, вони створюють ефект "прив'язки до постачальника" (vendor lock-in) та вимагають значних фінансових вкладень.

Таблиця 1.3

Порівняльний аналіз комерційного підходу та обраного Open-Source рішення

Компонент архітектури	Комерційне рішення	Обране рішення для магістерської роботи	Переваги обраного рішення
Policy Administrator (Identity)	Cisco ISE / Azure AD	FreeIPA (LDAP + Kerberos)	Повний контроль над даними, відсутність ліцензійних платежів, підтримка HBAC
Policy Enforcement Point (Network)	Cisco Firepower / Palo Alto	pfSense (Packet Filter)	Гнучкість налаштування VLAN, можливість розгортання на будь-якому "залізі" або у віртуальному середовищі
Continuous Monitoring (Analytics)	Splunk / Google Chronicle	Security Onion	Відкритість алгоритмів детекції, можливість глибокого аналізу пакетів (PCAP) без додаткових витрат

Для цілей даного дослідження найбільш доцільним визначено використання інтегрованого open-source стеку (FreeIPA, pfSense, Security Onion). Цей набір інструментів дозволяє реалізувати всі логічні компоненти архітектури NIST SP 800-207, забезпечуючи при цьому прозорість, керованість та можливість детального моделювання процесів захисту, що буде реалізовано в наступних розділах роботи.

1.4 Постановка задачі дослідження: вимоги до проєктованої системи

На основі проведеного аналізу теоретичних засад архітектури нульової довіри та огляду існуючих підходів сформульовано задачу дослідження: дослідити методи та засоби архітектури Zero Trust для захисту корпоративної IT-інфраструктури та продемонструвати можливості їх практичного застосування шляхом розробки, розгортання і тестування лабораторного прототипу у віртуальному середовищі.

Метою дослідження є аналіз та обґрунтування методів Zero Trust (зокрема принципів “Never Trust, Always Verify”, least privilege, мікросегментації, безперервної верифікації та моніторингу), вибір засобів їх реалізації у корпоративній IT-інфраструктурі та експериментальна перевірка ефективності запропонованого підходу на стенді за допомогою моделювання типових атак і аналізу подій безпеки [5]:

- Спроектувати мережеву архітектуру на основі мікросегментації з використанням VLAN
- Реалізувати централізовану систему управління ідентичністю та доступом
- Впровадити безперервний моніторинг мережевого трафіку та виявлення загроз
- Забезпечити динамічне застосування політик доступу на основі контексту
- Провести тестування ефективності системи проти типових атак

Функціональні вимоги до системи:

Відповідно до логічних компонентів ZTA за стандартом NIST SP 800-207, проєктована система повинна реалізовувати три ключові функціональні блоки (Таблиця 1.4).

Таблиця 1.4

Реалізація логічних компонентів ZTA у проєктованій системі

Компонент NIST	**Функція**	**Технічна реалізація в проєкті**
Policy Engine (PE)	Прийняття рішень про надання або відмову в доступі на основі політик та ризиків	FreeIPA (HBAC правила) + Аналітика загроз
Policy Administrator (PA)	Генерація токенів/квитків та встановлення комунікаційних шляхів	FreeIPA (Kerberos KDC, LDAP)
Policy Enforcement Point (PEP)	Безпосередній контроль з'єднань, блокування/дозвіл трафіку	pfSense (Firewall Rules, VLAN tagging)

Система повинна підтримувати такі функціональні можливості:

- Автентифікація та авторизація: Підтримка протоколів Kerberos та LDAP для уніфікованого управління доступом.
- Мікросегментація мережі: Розділення на логічні зони (Management, Users, Servers, DMZ) з жорстким контролем трафіку між сегментами.
- Безперервний моніторинг: Збір та аналіз журналів подій у режимі реального часу з використанням SIEM.
- Принцип найменших привілеїв: Надання лише мінімально необхідних прав для виконання завдань (Default Deny).

Для реалізації прототипу у віртуальному середовищі GNS3 визначено вимоги до обчислювальних ресурсів, необхідні для розгортання повнофункціонального стенду (Таблиця 1.5).

Таблиця 1.5

Розрахункові вимоги до ресурсів для лабораторного стенду

Компонент	**Роль у системі**	**vCPU**	**RAM**	**Storage**
Security Onion	SIEM / IDS / Моніторинг	4-8	16-24 GB	200 GB
pfSense	Шлюз безпеки (PEP) / Маршрутизатор	2	2 GB	20 GB
FreeIPA Server	Контролер ідентифікації (IdP)	2	2 GB	20 GB
Web Server (Ubuntu)	Цільовий ресурс (DMZ)	1	1 GB	20 GB
Database Server (Ubuntu)	Внутрішній ресурс (Protected Zone)	1	2 GB	25 GB
Клієнти (Ubuntu/Kali)	Робоча станція / Пентест	2	4 GB	40 GB
ЗАГАЛОМ	Повний стенд	12-16	27-35 GB	~325 GB

Для оптимізації роботи на лабораторному обладнанні допускається використання полегшених версій ПЗ або зменшення виділених ресурсів без втрати критичної функціональності.

Технічні обмеження проєктованої системи:

- Використання виключно програмного забезпечення з відкритим кодом (Open Source) для забезпечення доступності та відтворюваності результатів.

- Підтримка стандартних протоколів (802.1Q, Syslog) для сумісності з типовою корпоративною інфраструктурою.
- Можливість розгортання у віртуалізованому середовищі (Hypervisor-agnostic).

Висновок до першого розділу

У першому розділі проведено аналіз сучасного стану захисту корпоративних мереж, який виявив критичні недоліки традиційної моделі «захисту периметра». Встановлено, що в умовах гібридної інфраструктури, хмарних обчислень та віддаленої роботи периметровий підхід не здатен протидіяти актуальним загрозам, таким як фішинг та програми-вимагачі. Зокрема, вразливості VPN та відсутність контролю внутрішнього трафіку створюють умови для безперешкодного горизонтального переміщення зловмисників (lateral movement).

Дослідження концепції Zero Trust Architecture (NIST SP 800-207) підтвердило її ефективність як сучасної парадигми безпеки. Визначено, що впровадження принципів «ніколи не довіряй, завжди перевіряй», найменших привілеїв та безперервної верифікації дозволяє нівелювати слабкі місця традиційних підходів. Основними стратегіями захисту обрано мікросегментацію мережі та динамічне управління доступом.

За результатами порівняльного аналізу для реалізації прототипу обґрунтовано вибір стеку технологій з відкритим кодом: FreeIPA (управління ідентичністю), pfSense (шлюз безпеки) та Security Onion (моніторинг). Такий підхід дозволяє побудувати повнофункціональну архітектуру Zero Trust без фінансових витрат на пропрієтарні рішення. Сформульовані функціональні та технічні вимоги створюють підґрунтя для практичного проєктування системи у наступних розділах роботи.

РОЗДІЛ 2. ПРОЄКТУВАННЯ ЗАХИСТУ НА ОСНОВІ ZERO TRUST: МЕТОДИ ТА ПОЛІТИКИ

2.1 Проєктування мережевої архітектури та мікросегментації IT-інфраструктури

Для реалізації архітектури нульової довіри (Zero Trust Architecture) було спроєктовано лабораторну топологію корпоративної IT-інфраструктури, яка відображає типові потреби організації середнього масштабу. Проєктування виконано з урахуванням того, що в межах нової теми роботи мікросегментація є методом Zero Trust, який забезпечує зменшення “радіусу ураження” (blast radius) та ускладнює lateral movement, а засобами реалізації цього методу виступають технології VLAN 802.1Q, міжсегментна маршрутизація та політики контролю трафіку на міжмережевому екрані (pfSense) [16]. Відповідно, у підрозділі 2.1 описано саме проєктування топології, VLAN/IP-плану, а також базових інфраструктурних сервісів (DNS/NTP), які є фундаментом для подальшого впровадження політик доступу та системи спостережуваності.

Топологія побудована на принципах microsegmentation та default-deny: кожен сегмент ізольований за замовчуванням, а міжсегментна комунікація дозволяється лише у випадках, коли вона явно необхідна для бізнес-функцій (автентифікація, доступ до веб-ресурсу, backend-з’єднання тощо).

Архітектура складається з наступних логічних компонентів:

• Policy Engine (PE) / Policy Administrator (PA) — FreeIPA Server (централізоване керування ідентичністю та політиками доступу)

• Policy Enforcement Point (PEP) — pfSense Firewall (контроль міжсегментного трафіку та примусове застосування правил)

• Continuous Monitoring / Telemetry — Security Onion (збір, кореляція та аналіз подій)

• Protected Resources — Web Server, Database Server

• Client Endpoints — робочі станції користувачів

Загальна топологія інфраструктури представлена на рисунку 2.1.

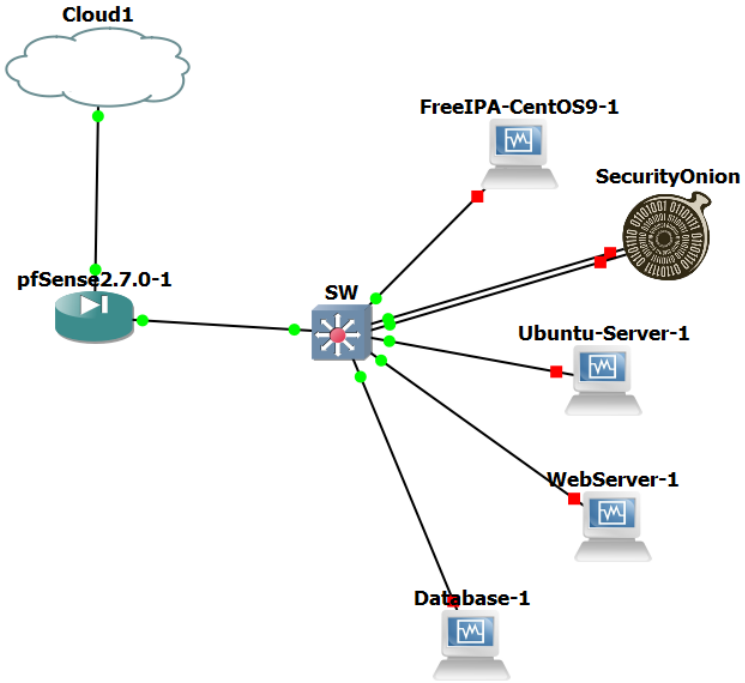


Рисунок 2.1 — Загальна топологія лабораторної мережі Zero Trust Architecture

Відповідно до принципів мікросегментації NIST SP 800-207, мережа розділена на чотири логічні сегменти з використанням технології VLAN 802.1Q. Кожен сегмент виконує специфічну роль у загальній архітектурі безпеки та має власну підмережу з незалежним адресним простором.

Таблиця 2.1

Мережева сегментація за VLAN

VLAN ID	Назва сегмента	Призначення	Мережа	Шлюз
10	Management	Інфраструктурні сервіси (FreeIPA, Security Onion)	192.168.10.0/24	192.168.10.1
20	Users	Робочі станції користувачів	192.168.20.0/24	192.168.20.1
30	Servers	Внутрішні додатки (Database)	192.168.30.0/24	192.168.30.1
40	DMZ	Публічно доступні сервіси (Web Server)	192.168.40.0/24	192.168.40.1

Кожен VLAN налаштовується на pfSense як окремий інтерфейс з власними правилами файрвола. За замовчуванням весь трафік між VLAN заборонений (implicit deny), дозволені лише явно визначені з'єднання згідно з політиками доступу [4] [7] [16].

Детальний опис мережевих сегментів

VLAN 10: Management Zone (Зона управління)

У цьому сегменті розміщуються критичні компоненти Zero Trust Architecture, що забезпечують центральне управління ідентичністю, синхронізацію часу та безперервний моніторинг безпекових подій.

FreeIPA Server (192.168.10.10)

- Функція: Policy Engine / Policy Administrator (IdP домену)
- Операційна система: CentOS 9
- Ключові сервіси:
  - Kerberos KDC (88/TCP, 88/UDP)
  - LDAP Directory (389/TCP), (за потреби LDAPS 636/TCP)
  - DNS (53/UDP, 53/TCP)
  - NTP (123/UDP) — як внутрішнє джерело часу для домену
- Роль у ZTA: централізована автентифікація та авторизація (зокрема через HBAC), облік ідентичностей, керування доступом.
- Ресурси: 2 vCPU, 4 GB RAM, 20 GB Storage

Security Onion (192.168.10.20)

- Функція: SIEM / IDS / Network Monitoring
- Компоненти: Suricata (IDS), Zeek (аналізатор протоколів), Elasticsearch, Kibana
- Роль: Безперервний моніторинг, виявлення аномалій, форензика
- Інтерфейси: enp0s3 (management), enp0s8 (monitoring у promiscuous mode)
- Ресурси: 4-8 vCPU, 16-24 GB RAM, 200 GB Storage

pfSense Management Interface (192.168.10.1)

- Функція: Policy Enforcement Point
- Роль: Веб-інтерфейс управління файрволом, шлюз для Management VLAN, fallback NTP-сервер

Доступ до Management Zone обмежується адміністративними сценаріями (керування FreeIPA, Security Onion, pfSense). Це відповідає практикам Zero Trust,

де керуючі компоненти мають бути відокремлені в окремому сегменті, а доступ до них контролюється максимально жорстко.

VLAN 20: Users Zone (Зона користувачів)

Сегмент призначений для робочих станцій кінцевих користувачів, які проходять автентифікацію через FreeIPA та отримують доступ до корпоративних ресурсів відповідно до своїх ролей.

Ubuntu Client (192.168.20.100)

- Функція: Типова робоча станція користувача
- Інтеграція: Приєднана до домену FreeIPA через SSSD
- Автентифікація: Kerberos tickets, LDAP-автентифікація
- Синхронізація часу: Через NTP з FreeIPA Server
- Доступ: HTTP/HTTPS до Web Server (VLAN 40), Kerberos/LDAP до FreeIPA (VLAN 10)
- ОС: Ubuntu 22.04 LTS
- Ресурси: 2 vCPU, 2-4 GB RAM, 30 GB Storage

Політика доступу Users Zone:

Користувачі можуть звертатися до веб-додатків у DMZ та отримувати автентифікацію через FreeIPA. Прямий доступ до внутрішніх серверів (VLAN 30) заборонений згідно з принципом least privilege.

VLAN 30: Servers Zone (Захищена зона)

Найбільш критичний сегмент, що містить внутрішні сервіси з чутливими даними. Доступ до цієї зони суворо обмежений та логується.

Database Server (192.168.30.10)

- ОС: Ubuntu Server 22.04 LTS
- Сервіси: PostgreSQL 15 (порт 5432)
- База даних: webapp_db для веб-додатку
- Доступ: Лише з Web Server (VLAN 40) на порт 5432
- Автентифікація: Інтеграція з FreeIPA через SSSD
- Синхронізація часу: Через NTP з FreeIPA Server
- Ресурси: 2 vCPU, 2 GB RAM, 25 GB Storage

Політика безпеки Servers Zone:

Прямий доступ користувачів до Database Server категорично заборонений. З'єднання можливе лише через Web Server як проксі-рівень додатку. Події доступу до сервера баз даних, системні журнали ОС та мережеві блокування/дозволи фіксуються та передаються до Security Onion через syslog; додатково журнали СУБД (PostgreSQL) використовуються для аналізу прикладних подій доступу.

VLAN 40: DMZ Zone (Демілітаризована зона)

Сегмент для розміщення публічно доступних сервісів, що обслуговують запити користувачів та зовнішніх клієнтів.

Web Server (192.168.40.10)

- ОС: Ubuntu Server 22.04 LTS
- Сервіси: Apache2 з mod_auth_gssapi, PHP 8.1
- Автентифікація: Kerberos SSO через інтеграцію з FreeIPA
- Доступ: HTTP/HTTPS з VLAN 20 (Users), підключення до Database (VLAN 30), автентифікація через FreeIPA (VLAN 10)
- Синхронізація часу: Через NTP з FreeIPA Server
- Ресурси: 2 vCPU, 2 GB RAM, 20 GB Storage

Service Principal:

Для веб-сервера створено service principal HTTP/web01.zerotrust.local у FreeIPA, що дозволяє реалізувати Single Sign-On (SSO) для користувачів без повторного введення облікових даних [8] [11].

Для реалізації принципу least privilege визначено матрицю дозволених з'єднань між сегментами. Усі з'єднання, не вказані явно, блокуються на рівні pfSense з логуванням спроб несанкціонованого доступу.

Зазначені потоки реалізуються на рівні pfSense як набір правил між VLAN-інтерфейсами. При цьому усі блокування доцільно логувати для демонстрації принципів Zero Trust у практичній частині та для формування доказової бази в розділі тестування.

Таблиця 2.2

Матриця контролю доступу між VLAN

Джерело (Source)	Призначення (Destination)	Протокол/Порт	Дозвіл	Обґрунтування
Users (VLAN 20)	FreeIPA (VLAN 10)	TCP/UDP 88 (Kerberos), TCP 389 (LDAP), UDP 123 (NTP)	Дозволено	Автентифікація користувачів та синхронізація часу
Users (VLAN 20)	FreeIPA (VLAN 10)	UDP/TCP 53 (DNS)	Дозволено	DNS-резолюція домену та зовнішніх імен через forwarders
Users (VLAN 20)	Web Server (VLAN 40)	TCP 80, TCP 443	Дозволено	Доступ до веб-додатку
Users (VLAN 20)	Database (VLAN 30)	ANY	Заборонено	Порушення least privilege (захист БД)
Web Server (VLAN 40)	Database (VLAN 30)	TCP 5432	Дозволено	Backend connectivity (Web → DB)
Web Server (VLAN 40)	FreeIPA (VLAN 10)	TCP/UDP 88, TCP 389, UDP 123	Дозволено	Kerberos/LDAP/NTP для прикладної інтеграції
Web Server (VLAN 40)	FreeIPA (VLAN 10)	UDP/TCP 53	Дозволено	DNS-резолюція доменних імен сервісів
Database (VLAN 30)	FreeIPA (VLAN 10)	TCP/UDP 88, TCP 389, UDP 123	Дозволено	Автентифікація/довідник/NTP
Database (VLAN 30)	FreeIPA (VLAN 10)	UDP/TCP 53	Дозволено	DNS-резолюція домену
Усі сегменти	Security Onion (VLAN 10)	UDP 514 (Syslog)	Дозволено	Централізований збір логів та подій

Повна схема IP-адресації та призначення інтерфейсів наведена у таблиці 2.3.

Таблиця 2.3

Детальна IP-адресація компонентів системи

Пристрій	Інтерфейс	IP-адреса	Маска	VLAN	Шлюз	DNS	NTP
pfSense	em0 (WAN)	DHCP або 192.168.1.100	/24	–	192.168.1.1	–	pool.ntp.org / time.cloudflare.com
pfSense	em1 (trunk/LAN base)	192.168.10.1	/24	10	–	–	–
pfSense	em1.20	192.168.20.1	/24	20	–	–	–
pfSense	em1.30	192.168.30.1	/24	30	–	–	–
pfSense	em1.40	192.168.40.1	/24	40	–	–	–
FreeIPA	eth0	192.168.10.10	/24	10	192.168.10.1	127.0.0.1	time.cloudflare.com, pool.ntp.org
Security Onion	enp0s3 (mgmt)	192.168.10.20	/24	10	192.168.10.1	192.168.10.10	192.168.10.10
Security Onion	enp0s8 (monitor)	без IP	–	–	–	–	–
Ubuntu Client	eth0	192.168.20.100	/24	20	192.168.20.1	192.168.10.10	192.168.10.10
Web Server	eth0	192.168.40.10	/24	40	192.168.40.1	192.168.10.10	192.168.10.10
Database Server	eth0	192.168.30.10	/24	30	192.168.30.1	192.168.10.10	192.168.10.10

У даній архітектурі FreeIPA виконує роль внутрішнього DNS для домену zerotrust.local. Усі клієнти та сервери налаштовані на використання 192.168.10.10 як primary DNS. Для резолюції зовнішніх доменів FreeIPA використовує DNS forwarders (наприклад, 1.1.1.1 та 8.8.8.8), що дозволяє зберегти єдину точку контролю DNS у домені та зменшити конфігураційну складність для клієнтів.

Синхронізація часу є критично важливою для коректної роботи Kerberos, оскільки протокол чутливий до розбіжностей часу між клієнтом і сервером; типовий допустимий зсув становить близько 5 хвилин. У спроєктованій інфраструктурі реалізовано ієрархічну модель синхронізації часу: FreeIPA синхронізується із зовнішніми NTP-джерелами та виступає авторитетним внутрішнім NTP-сервером для всіх вузлів домену, тоді як pfSense може використовуватись як резервне джерело часу.

Таблиця 2.4

Ієрархія NTP-синхронізації в мережі

Вузол	**Stratum**	**NTP-джерело**	**Роль**
time.cloudflare.com	1–2	еталонні джерела	Зовнішнє еталонне джерело часу
pool.ntp.org	1–2	еталонні джерела	Альтернативне еталонне джерело
FreeIPA (192.168.10.10)	3	time.cloudflare.com, pool.ntp.org	Авторитетний NTP для домену
pfSense (WAN)	3	pool.ntp.org / time.cloudflare.com	Резервне джерело часу (за потреби)
Ubuntu Client / Web / DB	4	FreeIPA (192.168.10.10)	NTP-клієнти домену

Така модель відповідає практикам побудови доменних інфраструктур (зокрема AD/FreeIPA), де контролер домену виступає як єдине джерело часу для членів домену, що підвищує стабільність Kerberos-автентифікації та зменшує ризики помилок авторизації.

Топологію розгорнуто у віртуальному середовищі GNS3 з використанням QEMU-віртуалізації. Архітектура включає:

- Центральний Ethernet Switch з підтримкою 802.1Q VLAN tagging.
- pfSense підключений до комутатора через trunk-порт em1 з передачею VLAN 10/20/30/40.
- Кожен хост підключений до комутатора як access-порт у відповідному VLAN.

- Security Onion має два інтерфейси: керуючий у VLAN 10 та моніторинговий інтерфейс у promiscuous mode, який отримує mirrored/SPAN трафік для аналізу без впливу на продуктивність.

NAT та доступ до Інтернету

- WAN-інтерфейс pfSense (em0) підключено до NAT Cloud у GNS3 для доступу до зовнішніх ресурсів.
- pfSense виконує Source NAT (SNAT) для вихідних з'єднань з внутрішніх сегментів.
- Внутрішні хости використовують pfSense як default gateway відповідно до VLAN.

Усі VM використовують virtio-мережеві адаптери для оптимізації продуктивності. Для Security Onion наявність окремого моніторингового інтерфейсу є обов'язковою умовою коректного пасивного аналізу трафіку.

У підрозділі 2.1 спроєктовано топологію корпоративної IT-інфраструктури як фундамент для впровадження Zero Trust Architecture. На рівні методів Zero Trust реалізовано підхід мікросегментації та default deny, що зменшує blast radius та обмежує lateral movement. На рівні засобів визначено VLAN-сегментацію 802.1Q, міжсегментний контроль трафіку через pfSense (PEP), централізовані DNS/NTP сервіси на FreeIPA для коректної доменної інтеграції та основу спостережуваності через Security Onion.

Створене середовище готове до наступного етапу — проєктування та впровадження політик доступу на рівні ідентичностей і прикладної інтеграції (FreeIPA, Kerberos/SSO), а також деталізації правил примусового виконання політик (pfSense) та системи моніторингу (Security Onion).

2.2 Проєктування методу управління ідентичністю та доступом (IAM) у моделі Zero Trust

Управління ідентичністю та доступом (Identity and Access Management, IAM) є одним із базових методів реалізації архітектури нульової довіри, оскільки безпосередньо забезпечує принцип "verify explicitly" — явну перевірку кожного

запиту на доступ на основі ідентичності суб'єкта та контексту. На відміну від традиційної периметрової моделі, де довіра часто визначається мережевою локацією ("inside = trusted"), підхід Zero Trust передбачає, що автентифікація та авторизація виконуються для кожної сесії та кожного ресурсу, а доступ надається з дотриманням принципів least privilege та default deny.

У межах даного дослідження важливо розмежувати:

- метод IAM у Zero Trust — сукупність принципів і підходів (централізація ідентичності, RBAC, контекстна авторизація, аудит),
- засоби реалізації методу IAM — технології/продукти, які ці підходи реалізують (у прототипі: FreeIPA як IdM/IdP, Kerberos/LDAP/SSSD, правила RBAC/HBAC, сервісні ідентичності та облік подій).

Проєктування IAM-підсистеми для лабораторного прототипу базується на трьох взаємопов'язаних методах:

1. Централізація ідентичності (Single Source of Truth).

2. RBAC (Role-Based Access Control) як метод рольового розподілу доступів.

3. HBAC (Host-Based Access Control) як метод контекстно-залежної авторизації для реалізації least privilege і default deny.

Методологічне обґрунтування централізованого IAM

У традиційних корпоративних мережах поширеною є ситуація, коли облікові записи створюються локально на серверах (наприклад, /etc/passwd, /etc/shadow). Це призводить до password sprawl, неузгодженості політик доступу, складності відкликання прав та фрагментованого аудиту. Для Zero Trust такий підхід є недостатнім, оскільки ускладнює контроль ідентичності та централізоване застосування політик для всієї IT-інфраструктури.

У логіці NIST SP 800-207 рішення про доступ приймаються через компоненти Policy Engine (PE) та Policy Administrator (PA), а примусове виконання забезпечує Policy Enforcement Point (PEP). У лабораторному прототипі роль засобу централізованого IAM виконує FreeIPA, який забезпечує:

- єдине сховище ідентичностей (каталог LDAP),
- сильну автентифікацію (Kerberos),

- політики доступу (RBAC/HBAC),
- сервісні ідентичності (service principals),
- аудит і журнали подій (як база для SIEM).

FreeIPA (Free Identity, Policy, Audit) — інтегрована IAM-система для Linux-середовищ, що поєднує каталог, автентифікацію, політики доступу та інфраструктурні сервіси. Вибір FreeIPA як основного засобу IAM у роботі обумовлений відкритим кодом, підтримкою Kerberos/LDAP та можливістю реалізації granular політик доступу.

Таблиця 2.5

Компоненти FreeIPA та їх роль у реалізації IAM/Zero Trust

Компонент	Базова технологія	Функціональна роль	Зв'язок із принципами ZTA
389 Directory Server	LDAP v3	Централізоване сховище users/groups/hosts/policies	Single Source of Truth
MIT Kerberos 5 KDC	Kerberos	Ticket-based автентифікація, mutual authentication	Verify explicitly
Dogtag CA	X.509 PKI	Сертифікати для TLS/mTLS (host/service identity)	Підсилення довіри до сервісів
BIND DNS	DNS (+SRV)	Authoritative DNS домену, service discovery	Infrastructure integrity
SSSD (клієнт)	NSS/PAM	Інтеграція з ОС, кешування, застосування HBAC на вузлі	Host-based enforcement

Сертифікати (Dogtag CA) застосовуються для TLS/mTLS, тоді як keytab-файли використовуються Kerberos-сервісами для service principals.

У FreeIPA LDAP використовується як централізоване сховище атрибутів ідентичності (користувачі, групи, хости, політики), тоді як Kerberos забезпечує механізм автентифікації з квитками (tickets). Типовий цикл: створення користувача формує LDAP-запис та Kerberos principal; додавання хоста/сервісу формує host/service principal; правила авторизації (HBAC) зберігаються централізовано і застосовуються на вузлах через SSSD [11].

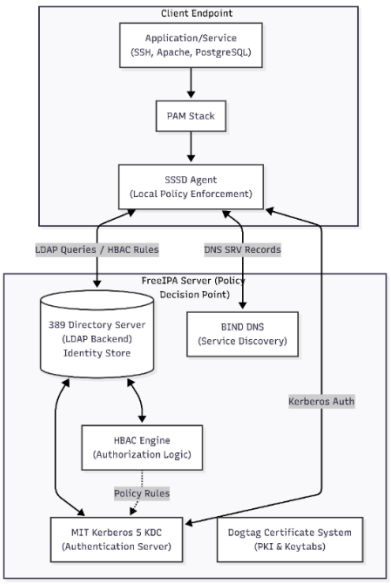


Рисунок 2.2 — Архітектура FreeIPA як Policy Decision Point у Zero Trust

RBAC (Role-Based Access Control): метод рольового розподілу доступу

RBAC — метод управління доступом, за якого права визначаються через ролі, що відображають функціональні обов'язки. RBAC зменшує складність адміністрування, підтримує separation of duties та забезпечує least privilege через мінімально необхідні набори прав для кожної ролі.

Формальна модель RBAC складається з таких множин:

- U (Users) — множина користувачів: {admin, jdoe, asmith, bwilson, cmartinez}
- R (Roles) — множина ролей: {admins, sysadmins, employees, webdevs, dbadmins}
- P (Permissions) — множина дозволів: {ssh_access, sudo_rights, db_query, web_deploy}
- S (Sessions) — активні сесії користувачів

Відношення:

- UA (User Assignment): U → R (користувач належить до ролі)
- PA (Permission Assignment): P → R (дозвіл призначено ролі)

Авторизаційне рішення:

$$\text{access}(u, p) = \exists r \in R : (u, r) \in UA \wedge (p, r) \in PA$$

У FreeIPA RBAC реалізується через LDAP-групи (ролі) та membership (зв'язок user→group). Ці зв'язки далі використовуються при оцінці HBAC-правил [8] [18].

Таблиця 2.6

Проєктування RBAC-моделі для прототипу

Роль (група)	**Бізнес-функція**	**Типові доступи**	**Обґрунтування least privilege**
admins (break-glass)	аварійний доступ	широкий доступ до всіх вузлів/сервісів	мінімум акаунтів, контрольований сценарій
sysadmins	супровід інфраструктури	SSH/sudo на інфраструктурні вузли	потрібен для maintenance/patching
employees	бізнес-користувачі	доступ до workstation + вебсервіс	без доступу до серверів
webdevs	розробники	SSH/sudo лише на web-tier	без доступу до db-tier
dbadmins	адміністратори БД	SSH/sudo на db-tier	ізоляція критичних даних

RBAC відповідає на питання "хто" (роль/група), тоді як HBAC додає контекст "де" (цільовий хост) і "що" (сервіс/тип доступу). Це дозволяє реалізувати Zero Trust-логіку: навіть після успішної автентифікації доступ надається лише тоді, коли запит відповідає явному правилу.

Формально рішення HBAC можна подати як функцію контексту:

$$\text{HBAC\_decision} = f(\text{user,source\_host, target\_host,service})$$

У лабораторному прототипі enforcement HBAC виконується на цільових хостах через SSSD, що інтегрується з PAM (наприклад, для sshd, login, sudo).

Нижче наведено узагальнений алгоритм, який пояснює логіку прийняття рішення:

1. На цільовому хості сервіс (наприклад, sshd) ініціює PAM-перевірку.

2. PAM викликає модуль pam_sss.so, який передає контекст до SSSD: user, target_host, service.

3. SSSD отримує/оновлює список увімкнених HBAC-правил із FreeIPA через LDAP (з кешуванням).

4. Для кожного правила rule із множини enabled_rules виконується перевірка відповідності:

- користувач або його група ∈ дозволених у правилі,
- target_host ∈ дозволених у правилі (або hostgroup),
- service ∈ дозволених у правилі,
- правило увімкнене.

5. Якщо знайдено хоча б одне правило, що збігається → ALLOW.

6. Якщо жодне правило не збіглося → DENY (default deny).

7. Рішення (ALLOW/DENY) логується локально та може передаватися в SIEM.

Таблиця 2.7

Матриця HBAC-правил у прототипі

Rule	Users/Groups (WHO)	Hosts (WHERE)	Services (WHAT)	Суть
admin_full_access	admins, sysadmins	ключові вузли	sshd, login, sudo	адміністрування/аварійний доступ
employee_workstation_access	employees	workstation	login, sudo(обмежено)	бізнес-користувачі без серверів
webdev_web_access	webdevs	web01	sshd, sudo	доступ лише до web-tier
dbadmin_db_access	dbadmins	db01	sshd, sudo	доступ лише до db-tier
allow_all	ALL	ALL	ALL	має бути вимкнено

Правило allow_all за замовчуванням руйнує Zero Trust-логіку, тому має бути вимкнене.

Сценарії доступу:

Сценарій 1 — Доступ дозволено (webdevs - web01 через SSH)

Учасники: користувач bwilson (група webdevs), ресурс web01. Мета: SSH-доступ для задач web-tier.

Фази взаємодії:

1. Service discovery: клієнт виконує DNS-резолюцію web01 (порт 53/UDP/TCP).

2. Kerberos TGS: клієнт звертається до KDC (порт 88/TCP/UDP) та отримує service ticket для host/web01.

3. SSH/GSSAPI authentication: sshd на web01 перевіряє ticket через keytab (порт 22/TCP).

4. HBAC authorization: SSSD на web01 застосовує HBAC-правила (отримання правил через LDAP 389/TCP з кешуванням).

5. Audit/logging: факт доступу та результат HBAC логуються та можуть передаватися в систему моніторингу (наприклад, syslog 514/UDP).

Доступ дозволено, оскільки правило webdev_web_access явно покриває (group=webdevs, host=web01, service=sshd).

Сценарій 2 — Доступ заборонено (webdevs - db01)

Учасники: bwilson (webdevs), ресурс db01.
Мета: спроба SSH-доступу до db-tier (має бути заборонено).

Логіка:

1. Kerberos автентифікація може бути виконана (service ticket для host/db01 може бути видано).

2. На етапі HBAC авторизації SSSD перевіряє правила.

3. Жодне правило не дозволяє webdevs доступ до db01 через sshd.

4. allow_all вимкнено → спрацьовує default deny.

5. Подія “DENY” логується та може корелюватися SIEM як порушення політики доступу.

Доступ заборонено — демонстрація separation of duties та least privilege enforcement.

У підрозділі спроєктовано метод IAM у моделі Zero Trust та визначено засоби його реалізації у лабораторному прототипі. Централізація ідентичності на базі FreeIPA забезпечує єдине джерело істини для користувачів, груп, хостів і політик; RBAC формалізує розподіл прав за ролями, а HBAC реалізує контекстно-залежну авторизацію та принцип default deny за рахунок явних правил доступу. Запропоновані сценарії демонструють розділення автентифікації (Kerberos) і авторизації (HBAC) та підтверджують виконання принципів least privilege і separation of duties.

2.3 Проєктування політик мережевого контролю доступу

Policy Enforcement Point (PEP) у моделі Zero Trust забезпечує примусове виконання політик доступу на мережевому рівні (L3/L4). На відміну від IAM-підсистеми (розділ 2.2), яка приймає рішення про доступ на рівні ідентичностей і контексту (RBAC/HBAC), PEP обмежує мережеві потоки між сегментами, реалізуючи принципи default deny, least privilege та мікросегментації.

У лабораторному прототипі PEP реалізовано на базі міжмережевого екрана pfSense, який виконує:

- Traffic filtering: дозволяє лише необхідні міжсегментні з'єднання;
- Stateful inspection: контролює стани сесій TCP/UDP та автоматично обробляє "return traffic" для дозволених з'єднань;
- Logging/telemetry: фіксує дозволені/заборонені спроби доступу та передає події у систему моніторингу.

pfSense є програмним міжмережевим екраном/маршрутизатором, який підтримує VLAN 802.1Q, міжсегментну маршрутизацію, stateful inspection та централізоване логування. Для лабораторного середовища цей вибір є доцільним через відтворюваність конфігурації, достатню функціональність і наявність WebGUI для документованого налаштування правил.

Таблиця 2.8

Порівняльна оцінка засобів реалізації PEP у прототипі

Критерій	**Комерційні NGFW (Palo Alto / Fortinet)**	**Linux iptables / nftables**	**pfSense (pf firewall)**
Підтримка VLAN / міжсегментної маршрутизації	Так	Так	Так
Stateful inspection (контроль стану сесій)	Так	Так	Так
Логування / експорт подій (syslog)	Так	Так, але потребує ручної конфігурації	Так (вбудовано)

Продовження таблиці 2.8

Критерій	**Комерційні NGFW (Palo Alto / Fortinet)**	**Linux iptables / nftables**	**pfSense (pf firewall)**
Складність налаштування	Середня (vendor-специфіка)	Висока (ручні правила/скрипти)	Низька–середня (WebGUI + можливість CLI)
Відтворюваність у лабораторії	Обмежена (вартість/ліцензії/апаратна залежність)	Висока	Висока (VM/стандартне обладнання)

Ключова ідея взаємодії рівнів:

- IAM-рівень визначає хто може звертатися і за яких умов (ідентичності/ролі/контекст);
- PEP-рівень визначає чи може існувати мережевий шлях між сегментами та сервісами.

Це формує defense in depth: навіть якщо користувач автентифікований, мережеві обмеження зменшують площу атаки та стримують lateral movement.

Принцип Default Deny та мікросегментація

Політика default deny означає, що будь-який міжсегментний трафік заборонено за замовчуванням, доки він не дозволений явним правилом.

Мікросегментація у прототипі реалізується через поділ мережі на VLAN, між якими трафік контролюється pfSense (PEP). Це дозволяє локалізувати компрометацію сегмента та зменшити “blast radius”.

Правила pfSense формуються за принципом мінімально необхідних залежностей між сервісами. У межах прототипу критично необхідні лише такі потоки:

1. Users → IAM/DNS/NTP (для доменної інтеграції та Kerberos)
2. Users → Web (доступ до застосунку)
3. Web → DB (backend-залежність)
4. Усі сегменти → SIEM/syslog (централізований аудит)
5. Internet → Web (лише публічні порти, якщо потрібен інбаунд)

Усі інші потоки блокуються як порушення least privilege.

Таблиця 2.9

Ключові дозволені потоки між VLAN

Джерело → Призначення	Протокол/порт	Дія	Призначення
Users (VLAN20) → FreeIPA (VLAN10)	88/TCP(UDP), 389/TCP, 123/UDP	Allow	Kerberos/LDAP/NTP
Users (VLAN20) → pfSense (DNS)	53/UDP(TCP)	Allow	DNS резолюція
Users (VLAN20) → Web (VLAN40)	80/TCP, 443/TCP	Allow	Доступ до веб-додатку
Web (VLAN40) → DB (VLAN30)	5432/TCP	Allow	Backend-з'єднання
Web (VLAN40) → FreeIPA (VLAN10)	88/TCP, 389/TCP, 123/UDP, 53/UDP	Allow	SSO/довідник/час/DNS
Усі VLAN → Security Onion (VLAN10)	514/UDP	Allow	Централізація логів
Users (VLAN20) → Servers (VLAN30)	Any	Block+Log	Заборона прямого доступу до DB-tier
DMZ (VLAN40) → Users (VLAN20)	Any	Block+Log	Заборона "reverse" підключень
Servers (VLAN30) → Users/DMZ	Any	Block+Log	Обмеження lateral movement з backend
Internet (WAN) → Web (VLAN40)	80/443 (NAT)	Allow+Log	Контрольований зовнішній доступ
Internet (WAN) → Internal (VLAN10/20/30)	Any	Block+Log	Заборона прямого інбаунду

Таким чином, найбільш принципові правила для демонстрації Zero Trust на PEP-рівні:

- USERS → SERVERS = BLOCK (зріз lateral movement до критичних даних),
- DMZ ↛ USERS (немає зворотних ініційованих з'єднань),
- SERVERS ↛ ANY (мінімізація вихідних потоків backend-сегмента),
- default deny наприкінці кожного інтерфейсу.

Stateful inspection як механізм контролю сесій

pfSense використовує stateful firewall, тобто зберігає стан дозволених з'єднань у state table. Це забезпечує:

- автоматичний дозвіл return traffic для встановлених дозволених сесій;
- блокування небажаних вхідних пакетів, які не належать жодному дозволеному стану;
- кращу керованість і прозорість (можна переглядати активні сесії в реальному часі).

Для реалізації принципу continuous monitoring у прототипі увімкнено логування ключових подій на pfSense (насамперед block-правил) із передачею на Security Onion через syslog (UDP/514). Це дозволяє:

- фіксувати порушення політик (наприклад, Users → DB);
- корелювати мережеві події з подіями автентифікації та журналами хостів;
- формувати доказову базу для розділу тестування/експериментів.

У підрозділі спроєктовано політики мережевого контролю доступу на рівні PEP із використанням pfSense як засобу реалізації. Реалізовано підхід default deny та мікросегментацію VLAN, що обмежує міжсегментні потоки лише до бізнес-необхідних залежностей (Users→IAM/Web, Web→DB, централізовані логи). Усі критичні порушення (зокрема Users→Servers) блокуються та логуються, що забезпечує як стримування lateral movement, так і основу для подальшого аналізу ефективності системи в експериментальній частині роботи.

2.4 Проєктування методу безперервного моніторингу та виявлення загроз

Безперервний моніторинг та виявлення загроз (Continuous Diagnostics and Mitigation, CDM) є третім критичним компонентом Zero Trust Architecture поряд із підсистемою управління ідентичністю (IAM, підрозділ 2.2) та мережевим контролем доступу (PEP, підрозділ 2.3). Якщо IAM і PEP реалізують переважно превентивні механізми (автентифікація, авторизація, фільтрація трафіку), то CDM забезпечує детективні та реагувальні можливості, необхідні для принципу

assume breach — припущення, що компрометація можлива навіть за наявності коректно налаштованих політик.

У межах лабораторного прототипу CDM-підсистема проєктується як метод забезпечення спостережуваності (observability) [14] і включає:

- early detection — оперативне виявлення порушень політик і аномалій (цільове вікно для критичних подій: <10 секунд), на відміну від традиційних підходів, де виявлення може вимірюватися десятками днів (орієнтовно 58 днів);
- forensics — накопичення артефактів (журнали, мережеві метадані, PCAP) для відтворення подій інциденту;
- compliance & audit — формування доказової бази дотримання політик доступу (default deny, least privilege) та контрольних заходів.

Таким чином, CDM виконує роль "третього шару" захисту: якщо IAM/PEP не зупинили загрозу на етапі доступу, система моніторингу забезпечує виявлення, кореляцію подій та основу для розслідування.

У лабораторному стенді роль централізованого засобу CDM виконує Security Onion — платформа для мережевого моніторингу та управління подіями безпеки, що інтегрує IDS/NSM-компоненти та інструменти централізації логів. Її використання узгоджується з метою побудови відтворюваного прототипу з enterprise-подібною функціональністю: аналіз мережевого трафіку, збір журналів із вузлів, кореляція та візуалізація інцидентів.

Таблиця 2.10

Порівняльний аналіз SIEM/моніторингових підходів для CDM

Критерій	**Commercial SIEM (Splunk/QRadar)**	**Open-source стек (ELK вручну)**	**Security Onion**
Вартість	висока (ліцензування)	низька, але потребує інтеграції	низька (інтегрований стек)
Network IDS	потребує окремих інтеграцій	ручне налаштування	вбудовано (Suricata)
Network analytics	залежить від модулів	ручне налаштування (Zeek тощо)	вбудовано (Zeek)

Продовження таблиці 2.10

Критерій	**Commercial SIEM (Splunk/QRadar)**	**Open-source стек (ELK вручну)**	**Security Onion**
Full packet capture	часто окремий продукт	ручна реалізація	підтримується в стеку
Лог-агрегація/пошук	так	так	так
Візуалізація	так	так (після налаштування)	так (готові дашборди)
Складність старту	середня	висока	нижча (єдина платформа)

Security Onion у контексті NIST ZTA (узгодження ролей):

1. Verify explicitly: агрегація подій автентифікації/авторизації (FreeIPA Kerberos/HBAC) у централізованому сховищі.

2. Use least privilege: виявлення та кореляція порушень доступу (HBAC DENY + firewall block).

3. Assume breach: детекція аномалій і атак у мережевому трафіку (Suricata IDS + Zeek).

4. Inspect and log traffic: отримання копії міжсегментного трафіку (SPAN) та збереження артефактів для аналізу.

5. Continuous monitoring: дашборди та правила сповіщення для інцидентів рівня ≥ HIGH.

Архітектура CDM

Для формалізації CDM-підсистеми у прототипі використовується багаторівневий підхід: від захоплення трафіку і збору логів до їх нормалізації, зберігання та представлення аналітику.

У такій моделі Security Onion виконує роль центрального вузла збору телеметрії, а ефективність виявлення визначається не лише окремими правилами IDS, але й можливістю кореляції між джерелами (IAM, firewall, хости, сервіси).

Таблиця 2.11

5-layer архітектура CDM у прототипі

Layer	**Компонент**	**Функція в системі**	**Призначення для ZTA**
5. Visualization	Kibana	дашборди, пошук, аналітика, інциденти	continuous monitoring
4. Storage	Elasticsearch	індексація та зберігання подій	швидкий пошук і ретеншн
3. ETL / Ingest	Log ingestion (syslog/агенти)	парсинг, нормалізація, enrichment	уніфікована модель подій
2. Detection	Suricata + Zeek	IDS (сигнатури) + поведінкова аналітика	відомі/невідомі загрози
1. Capture	SPAN + PCAP	видимість East–West трафіку	forensics та виявлення lateral movement

Оскільки прототип реалізує мікросегментацію (підрозділ 2.1) і контроль потоків між VLAN (підрозділ 2.3), критично важливою є видимість міжсегментного трафіку. Для цього у проєкті застосовується метод port mirroring (SPAN) — пасивне копіювання трафіку з trunk-порту комутатора на моніторинговий інтерфейс Security Onion.

Таблиця 2.12

Порівняння підходів до network visibility

Метод	**Видимість**	**Вплив на трафік**	**Вартість**	**Коментар для прототипу**
SPAN (port mirroring)	висока (копія трафіку)	0%	низька	оптимально для лабораторного середовища
TAP	дуже висока	0%	середня/висока	доцільно для високих швидкостей
NetFlow/IPFIX	лише метадані	мінімальний	низька	недостатньо для IDS (немає payload)

Методологічна цінність CDM у Zero Trust полягає у кореляції: окремий сигнал може бути шумом, тоді як комбінація мережевих та прикладних подій формує надійний індикатор інциденту. У прототипі реалізовано централізацію подій від ключових компонентів: pfSense (PEP), FreeIPA (PDP/IAM), серверів застосунку та бази даних, робочих станцій, а також мережевої телеметрії (Suricata/Zeek).

Для забезпечення повноти виявлення загроз у прототипі використовується комбінований підхід:

- Suricata (signature-based IDS) — виявлення відомих шаблонів атак і політик-порушень на основі правил;
- Zeek (network security monitoring) — формування мережевих метаданих і виявлення аномалій поведінки через аналіз протоколів.

Для практичного застосування CDM необхідно визначити критичність подій і очікуваний час реакції. У прототипі використовується градація severity, де найвищий пріоритет мають сценарії lateral movement, reverse shell та спроби обходу політик доступу.

Автоматизація сповіщення для подій severity ≥ HIGH реалізується правилами alerting у SIEM (без деталізації конфігурацій у розділі 2); приклади правил та їх застосування наведено в практичній частині.

CDM у запропонованій архітектурі не зводиться до “окремого IDS”, а реалізує метод multi-source correlation: мережеві блокування на pfSense, відмови HBAC на вузлах, журнали веб/БД та метадані Zeek/Suricata розглядаються як єдиний ланцюг подій. Це дозволяє:

- підтвердити порушення політик least privilege (наприклад, Users→DB) одночасно на мережевому та прикладному рівні;
- локалізувати інцидент у межах конкретного VLAN, зменшуючи blast radius;
- забезпечити відтворюваність експериментів у розділі тестування за рахунок збережених артефактів (логи та PCAP).

У підрозділі спроєктовано підсистему Continuous Diagnostics and Mitigation (CDM) як метод реалізації спостережуваності в моделі Zero Trust.

Висновок до другого розділу

У другому розділі здійснено комплексне проєктування системи захисту корпоративної IT-інфраструктури на основі архітектури Zero Trust, що інтегрує методи мікросегментації, управління ідентичністю та безперервного моніторингу. Розроблено мережеву топологію з використанням VLAN-сегментації для ізоляції критичних активів та зменшення радіусу потенційного ураження, а також спроєктовано модель управління доступом на базі FreeIPA, яка забезпечує перехід до явної верифікації через механізми суворої автентифікації Kerberos та контекстних правил HBAC.

Для контролю потоків даних визначено політики шлюзу безпеки pfSense, що реалізують принцип найменших привілеїв та блокування нелегітимного трафіку за моделлю Default Deny, тоді як підсистема CDM на базі Security Onion забезпечує централізований збір телеметрії та виявлення аномалій у реальному часі згідно з принципом припущення компрометації. Запропонована архітектура поєднує превентивні та детективні механізми в єдиний ешелонований контур, а обґрунтований вибір програмного стеку з відкритим кодом дозволяє створити економічно ефективне рішення, яке є готовим до практичного розгортання та експериментального тестування у наступному розділі.

РОЗДІЛ 3. РЕАЛІЗАЦІЯ ТА ОЦІНКА ЕФЕКТИВНОСТІ ЗАСОБІВ ZERO TRUST У ЛАБОРАТОРНОМУ СТЕНДІ

3.1 Розгортання обраних засобів Zero Trust у середовищі GNS3

Для практичної реалізації прототипу архітектури Zero Trust на основі компонентів відкритого програмного забезпечення обрано платформу віртуалізації мережі GNS3 (Graphical Network Simulator-3) версії 2.2.49. Вибір даного середовища обумовлений можливістю інтеграції мережевого обладнання Cisco IOS з віртуальними машинами на базі QEMU/KVM, що дозволяє створити гетерогенну інфраструктуру, максимально наближену до реальних умов експлуатації корпоративних мереж.

Відповідно до архітектури NIST SP 800-207, розроблена топологія включає основні функціональні компоненти Zero Trust: Policy Engine (PE), Policy Administrator (PA), Policy Enforcement Point (PEP) та систему Continuous Diagnostics and Mitigation (CDM). Реалізація даних компонентів здійснена з використанням наступних програмних рішень: FreeIPA Server (PE/PA), pfSense (PEP), та Security Onion (CDM).

Розгортання лабораторного стенду Zero Trust розпочинається з підготовки програмного середовища GNS3 та необхідних образів віртуальних машин. Для забезпечення достатньої продуктивності симуляції встановлено наступні системні вимоги до хост-системи: процесор з підтримкою апаратної віртуалізації (Intel VT-x або AMD-V), не менше 16 ГБ оперативної пам'яті, та дисковий простір обсягом 200 ГБ.

Встановлення GNS3 виконано з офіційного репозиторію з наступними компонентами:

- GNS3 GUI (графічний інтерфейс)
- GNS3 Server (backend для виконання віртуальних машин)
- QEMU/KVM (гіпервізор для емуляції)
- Wireshark (для захоплення та аналізу трафіку)

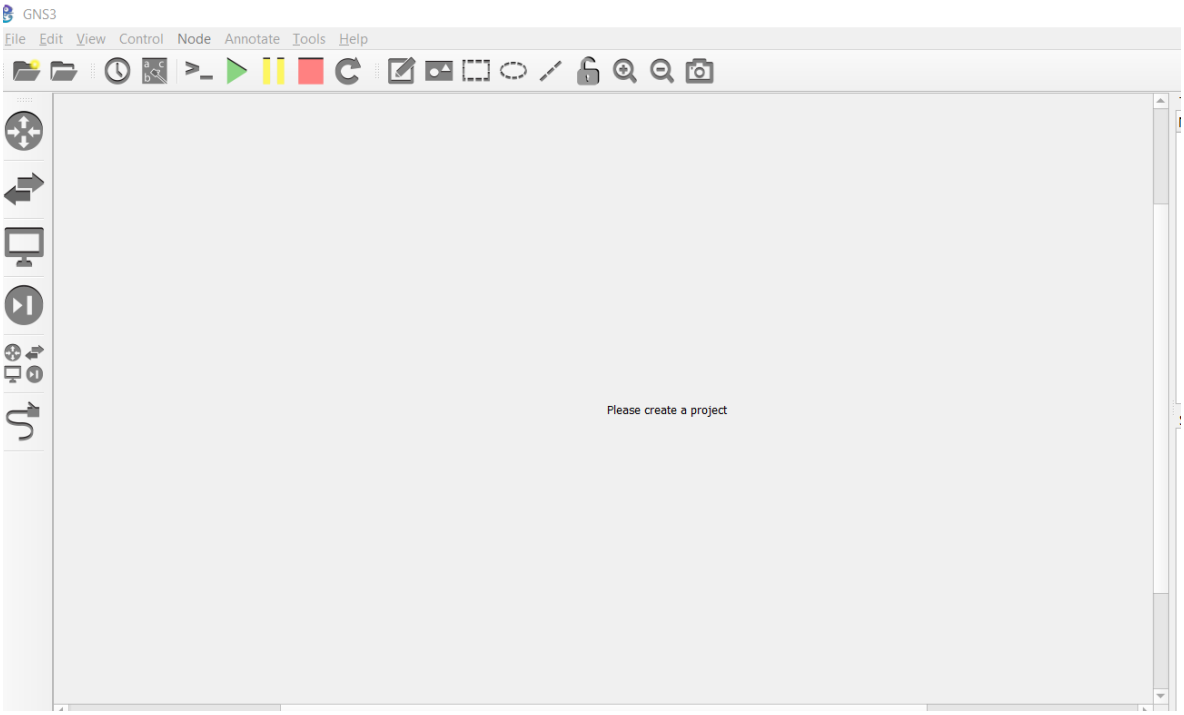


Рисунок 3.1 – Інтерфейс середовища віртуалізації GNS3

Розгортання компонентів інфраструктури виконано за комбінованою методологією:

1. Мережеві appliances (pfSense, IOSvL2) – імпортовано як готові шаблони GNS3 з офіційного репозиторію GNS3 Marketplace

2. Серверні компоненти (FreeIPA, Security Onion) – попередньо розгорнуто у VirtualBox для швидкого налаштування, потім конвертовано у формат QCOW2 та імпортовано до GNS3

3. Клієнтські та тестові вузли (Ubuntu) – аналогічно розгорнуто через VirtualBox.

Даний підхід дозволяє використати переваги апаратного прискорення VirtualBox для початкового встановлення складних систем (FreeIPA, Security Onion), уникаючи повільної емуляції QEMU під час інтерактивного налаштування, а потім інтегрувати готові образи у топологію GNS3 для комплексного тестування [15] [20].

У проекті GNS3 створено нову топологію "ZeroTrust-Lab", яка реалізує архітектуру NIST SP 800-207 з чітким розділенням функціональних компонентів. Топологія включає наступні елементи:

- Комутатор Cisco IOSvL2 15.2 – емуляція Layer 2 switch з підтримкою 802.1Q VLAN та SPAN моніторингу
- pfSense 2.7.2 – міжмережевий екран та маршрутизатор (Policy Enforcement Point)
- FreeIPA 4.11 на CentOS 9 – сервер автентифікації та авторизації (Policy Engine/Administrator)
- Security Onion 2.4.50 – платформа моніторингу безпеки (Continuous Diagnostics & Mitigation)
- Клієнтські та серверні вузли Ubuntu 22.04 LTS – для тестування та валідації

Архітектура мережі побудована на принципі мікросегментації з використанням технології VLAN 802.1Q для логічного розділення функціональних зон. Розроблена схема сегментації включає чотири основні VLAN (таблиця 3.1).

Таблиця 3.1

План адресації та VLAN сегментації мережі

VLAN ID	Призначення	Мережа	Шлюз	DNS	Кількість хостів
10	Management	192.168.10.0/24	192.168.10.1	192.168.10.10	FreeIPA, Security Onion, pfSense
20	Users	192.168.20.0/24	192.168.20.1	192.168.10.10	Робочі станції
30	Servers	192.168.30.0/24	192.168.30.1	192.168.10.10	Сервери PostgreSQL
40	DMZ	192.168.40.0/24	192.168.40.1	192.168.10.10	Веб-сервери Apache/Nginx

Топологічна схема з'єднань побудована таким чином, що весь міжсегментний трафік обов'язково проходить через pfSense для застосування політик контролю доступу, а Security Onion отримує копію всього трафіку через SPAN порт для моніторингу.

Комутатор IOSvL2 налаштовано з trunk-портом для передачі трафіку всіх VLAN до pfSense:

interface GigabitEthernet0/0

description TRUNK to pfSense

switchport trunk encapsulation dot1q

switchport mode trunk

switchport trunk allowed vlan 10,20,30,40

no shutdown

```
SW-ZeroTrust#show vlan brief

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Gi1/2, Gi1/3, Gi2/0, Gi2/1
                                                Gi2/2, Gi2/3, Gi3/0, Gi3/1
                                                Gi3/2, Gi3/3
10   MGMT_VLAN                        active    Gi0/1, Gi1/1
20   USERS_VLAN                       active    Gi0/2
30   SERVERS_VLAN                     active    Gi1/0
40   DMZ_VLAN                         active    Gi0/3
999  UNUSED                           active
1002 fddi-default                     act/unsup
1003 token-ring-default               act/unsup
1004 fddinet-default                  act/unsup
1005 trnet-default                    act/unsup
SW-ZeroTrust#
```

Рисунок 3.2 – Конфігурація VLAN на комутаторі Cisco IOSvL2

Базова конфігурація мережевих інтерфейсів pfSense включає головний інтерфейс LAN (em1) та чотири VLAN-інтерфейси (таблиця 3.2).

Таблиця 3.2

Конфігурація інтерфейсів pfSense

VLAN ID	Призначення	Мережа	Шлюз (pfSense)	DNS
10	Management	192.168.10.0/24	192.168.10.1	192.168.10.10
20	Users	192.168.20.0/24	192.168.20.1	192.168.10.10
30	Servers	192.168.30.0/24	192.168.30.1	192.168.10.10
40	DMZ	192.168.40.0/24	192.168.40.1	192.168.10.10

Налаштування VLAN виконано через веб-інтерфейс pfSense (https://192.168.10.1) за маршрутом Interfaces - Assignments - VLANs з вказанням Parent Interface (em1) та відповідних VLAN Tag.

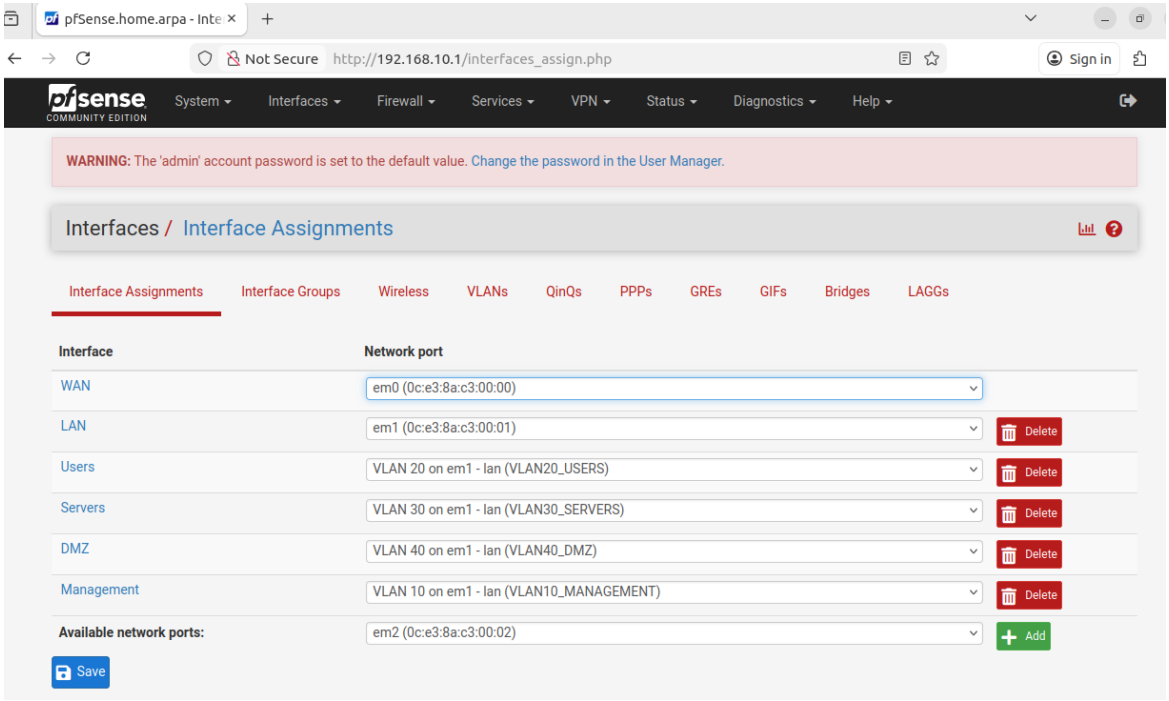


Рисунок 3.3 – Налаштування VLAN-інтерфейсів у pfSense

На етапі первинного налаштування залишено default allow any для перевірки зв'язку, який згодом змінено на default deny.

Віртуальна машина FreeIPA на базі CentOS9 імпортована до GNS3 з наступними параметрами: 4096 МБ RAM, 2 vCPU. Інтерфейс eth0 підключено до VLAN 10 через порт IOSvL2 Gi0/1 в режимі access.

Встановлення FreeIPA сервера виконано з інтегрованими DNS та NTP сервісами:

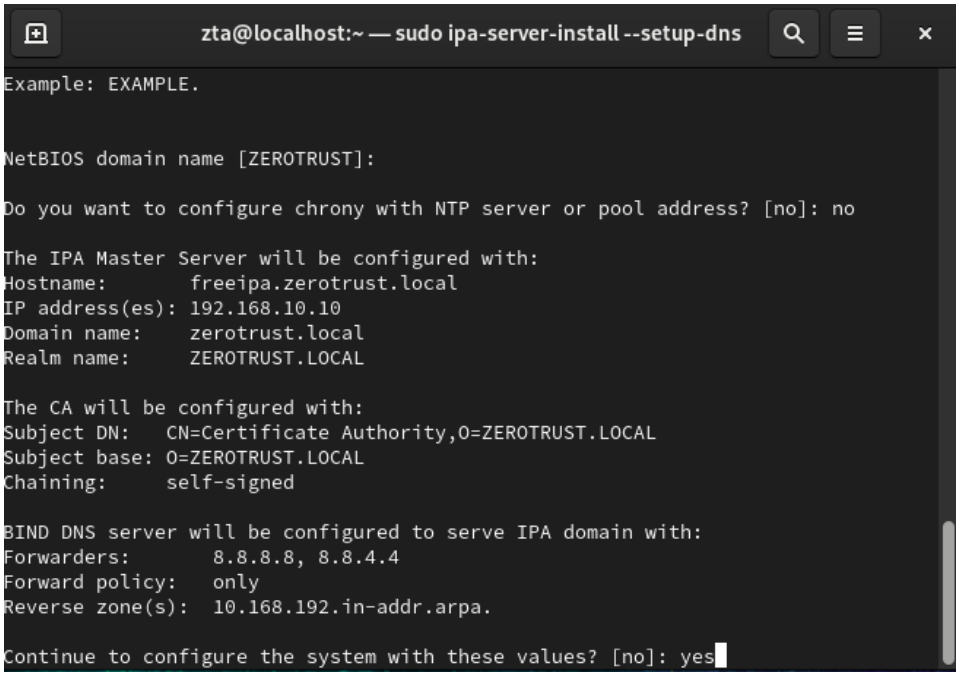


Рисунок 3.4 – Процес встановлення FreeIPA сервера

Після завершення встановлення перевірено доступність веб-інтерфейсу FreeIPA за адресою https://freeipa.zerotrust.local та виконано вхід з обліковими даними адміністратора.

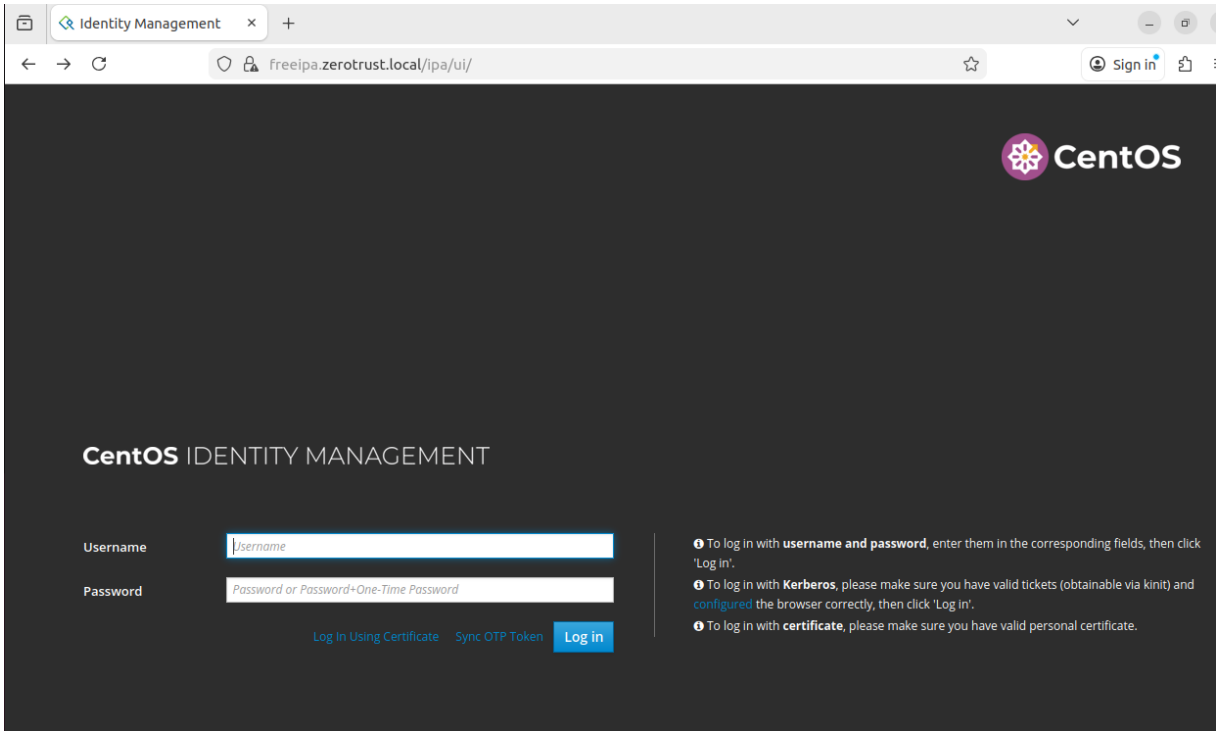


Рисунок 3.5 – Веб-інтерфейс FreeIPA після успішного розгортання

Для інтеграції клієнтських систем та серверів з FreeIPA domain на кожному вузлі встановлено клієнт ipa-client:

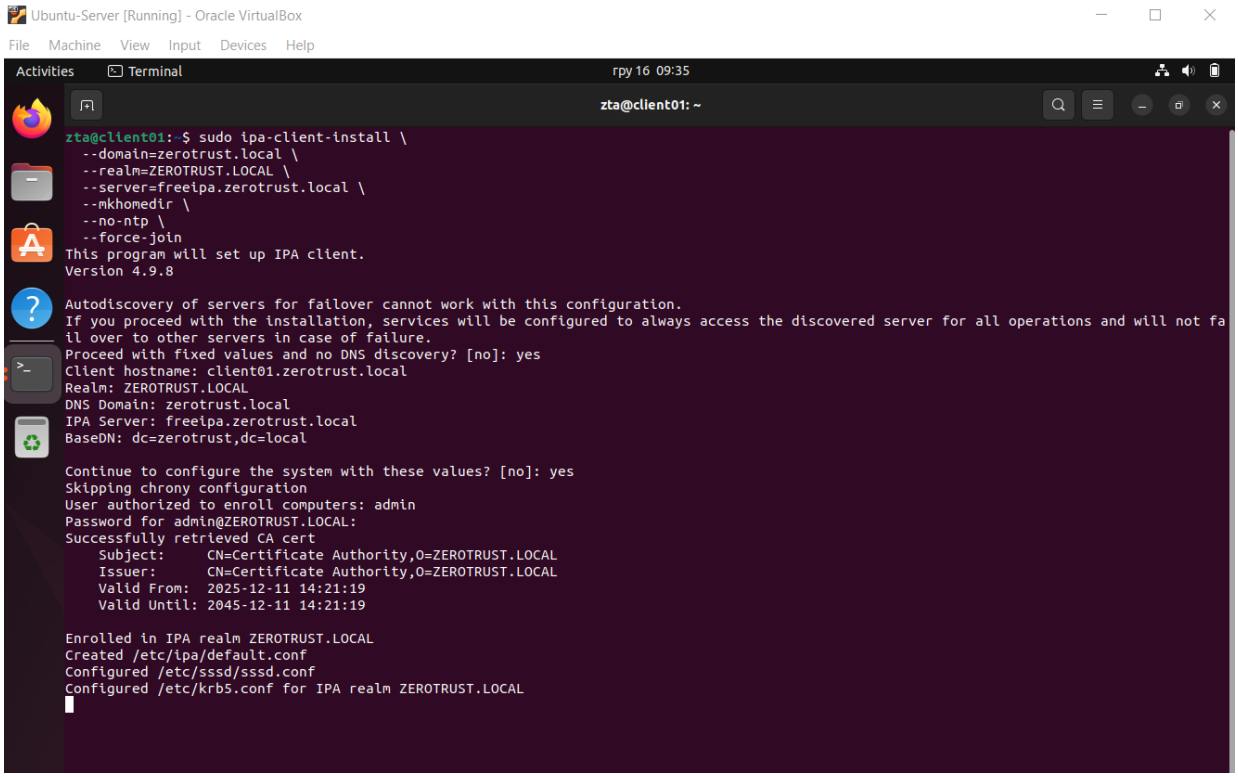


Рисунок 3.6 – Процес встановлення FreeIPA клієнта

Після успішного приєднання до домену користувачі можуть виконувати автентифікацію через протокол Kerberos командою kinit:

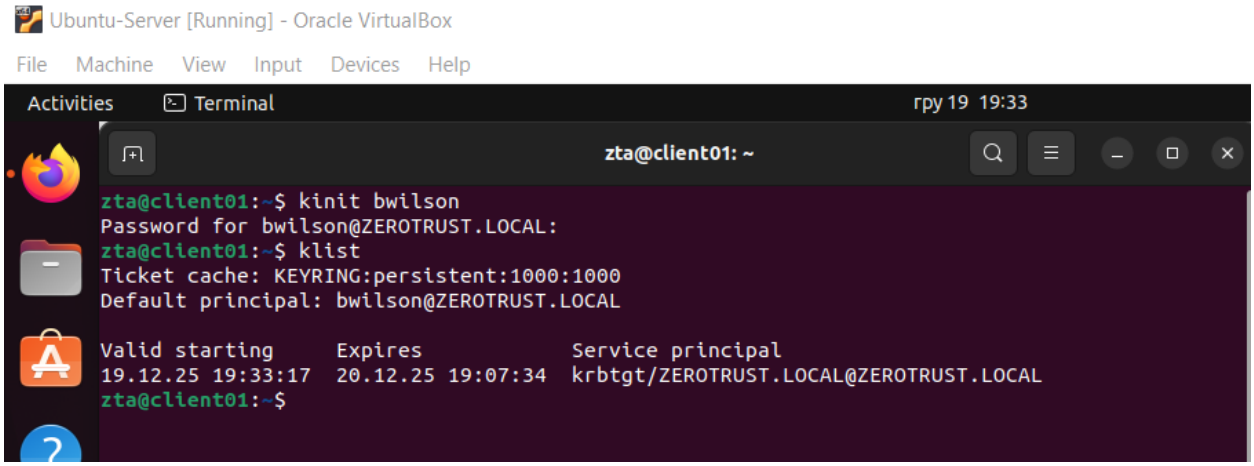


Рисунок 3.7 – Перевірка автентифікації

Отриманий Ticket-Granting Ticket (TGT) має термін дії 24 години за замовчуванням, що відповідає принципу Zero Trust щодо короткотривалих сесій доступу. Після закінчення терміну дії TGT необхідно повторно автентифікуватись, що знижує ризик компрометації облікових даних при крадіжці сесії.

Компонент Continuous Diagnostics & Mitigation (CDM) згідно з архітектурою NIST ZTA реалізовано на основі платформи Security Onion версії 2.4.50, яка інтегрує інструменти мережевого моніторингу Suricata IDS/IPS (версія 7.0), аналізу протоколів Zeek (Bro) версії 6.0, та SIEM-системи на базі Elasticsearch 8.x / Kibana 8.x. Security Onion розгорнуто в режимі EVAL (Evaluation standalone deployment) з наступними параметрами:

- Hostname: so01.zerotrust.local
- Management IP: 192.168.10.20/24 (інтерфейс enp0s3, VLAN 10)
- Monitoring interface: enp0s8 (без IP-адреси, режим promiscuous)
- Installation Type: EVAL (всі компоненти на одному сервері)
- Ресурси: 8 ГБ RAM (мінімум для EVAL mode), 8 vCPU, 200 ГБ дисковий простір

Особливістю конфігурації Security Onion є використання dual-homed архітектури з двома мережевими інтерфейсами різного призначення:

- enp0s3 (management) – адміністративний інтерфейс з IP-адресою для доступу через SSH та веб-інтерфейс Kibana/Security Onion Console (SOC). Через цей

інтерфейс здійснюється управління системою та перегляд аналітичних дашбордів.

- enp0s8 (monitoring) – спеціалізований інтерфейс для пасивного перехоплення мережевого трафіку в режимі promiscuous mode без призначення IP-адреси. Інтерфейс налаштовано лише на прийом пакетів (RX) без передачі (TX), що унеможливлює його використання для атак з компрометованої Security Onion системи.

Інтерфейс enp0s3 підключено до IOSvL2 Gi1/1 (access VLAN 10), інтерфейс enp0s8 підключено до Gi1/2 (SPAN destination port). Конфігурація SPAN на комутаторі:

monitor session 1 source interface Gi0/0 both

monitor session 1 destination interface Gi1/2

```
SW-ZeroTrust#show monitor session 1
Session 1
---------
Type                   : Local Session
Source Ports           :
    Both               : Gi0/0
Destination Ports      : Gi1/2
    Encapsulation      : Native
```

Рисунок 3.8 – Конфігурація SPAN моніторингу на комутаторі Cisco IOSvL2

Первинне налаштування Security Onion виконано через утиліту so-setup:

- Installation Type: EVAL (standalone)
- Hostname: so01.zerotrust.local
- Management Interface: enp0s3 (192.168.10.20/24)
- Monitor Interface: enp0s8 (promiscuous)
- Home Networks: 192.168.0.0/16
- Web Interface: Enabled
- Admin Username: admin
- DNS Server: 192.168.10.10 (FreeIPA)
- NTP Server: 192.168.10.10 (FreeIPA)

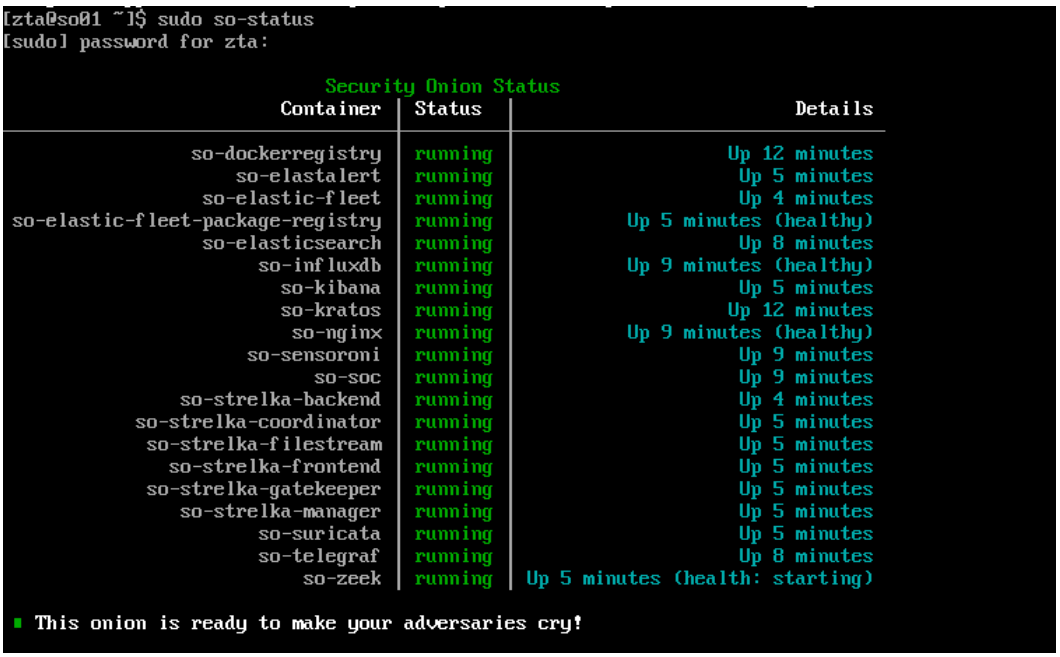


Рисунок 3.9 – Статус сервісів Security Onion

Після завершення встановлення перевірено доступність веб-інтерфейсу Security Onion Console за адресою https://192.168.10.20.

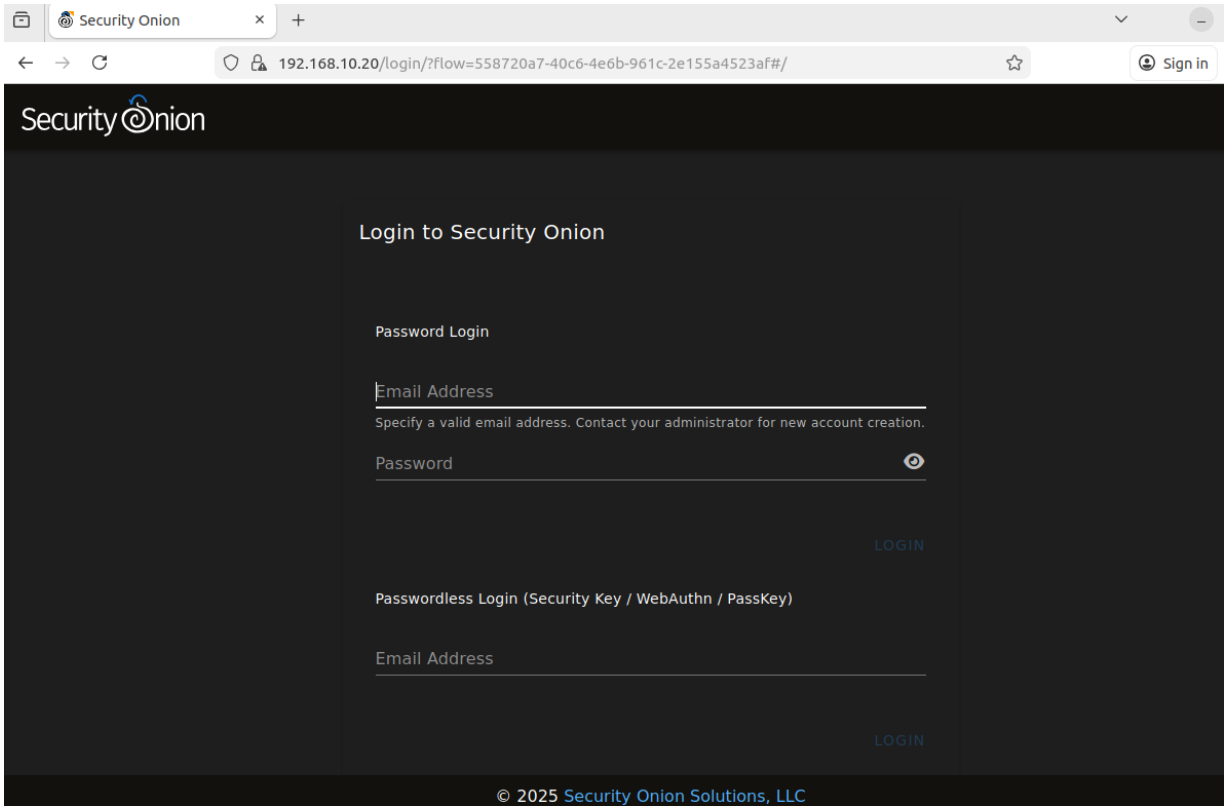


Рисунок 3.10 – Веб-інтерфейс Security Onion Console

Для тестування функціональності архітектури Zero Trust розгорнуто наступні додаткові вузли на базі Ubuntu 22.04 LTS.

Клієнтська станція (client01):

Сервер бази даних (db01):

Веб-сервер (web01):

Усі вузли налаштовано з статичними IP-адресами відповідно до плану адресації (таблиця 3.1) та вказано FreeIPA (192.168.10.10) як DNS-сервер у конфігурації мережі.

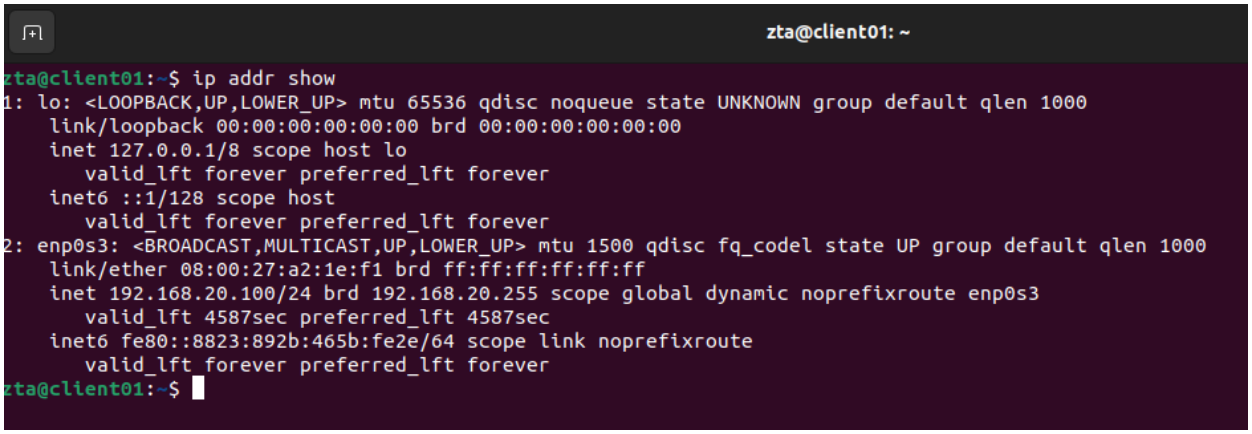


Рисунок 3.11 – Мережева конфігурація клієнтської станції

Після розгортання всіх компонентів виконано перевірку базової мережевої зв'язності для підтвердження коректності топології.

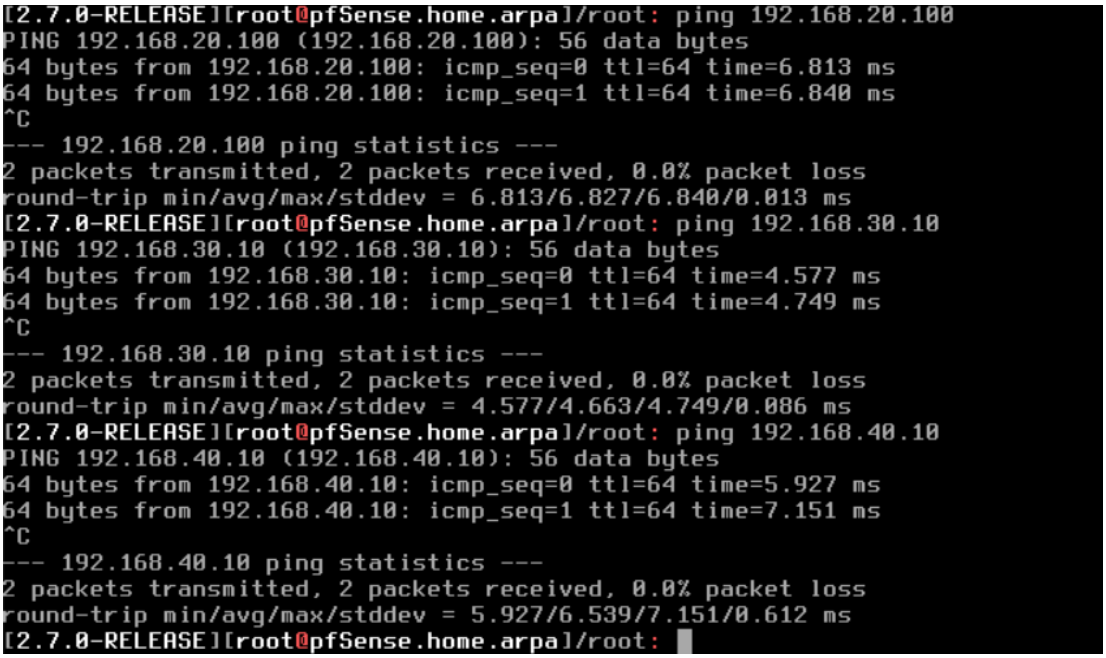


Рисунок 3.12 – Перевірка зв’язку у мережі

Успішний обмін ICMP-пакетами підтверджує, що маршрутизація між VLAN працює коректно через pfSense.

Перевірка роботи DNS через FreeIPA:

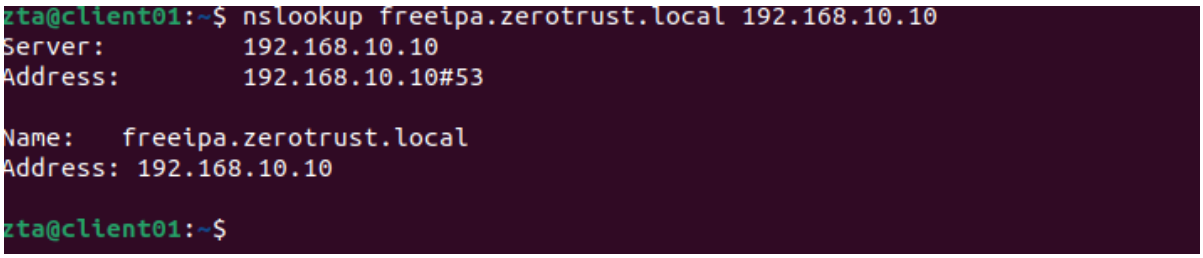


Рисунок 3.12 – Перевірка роботи DNS

Перевірка моніторингу Security Onion:

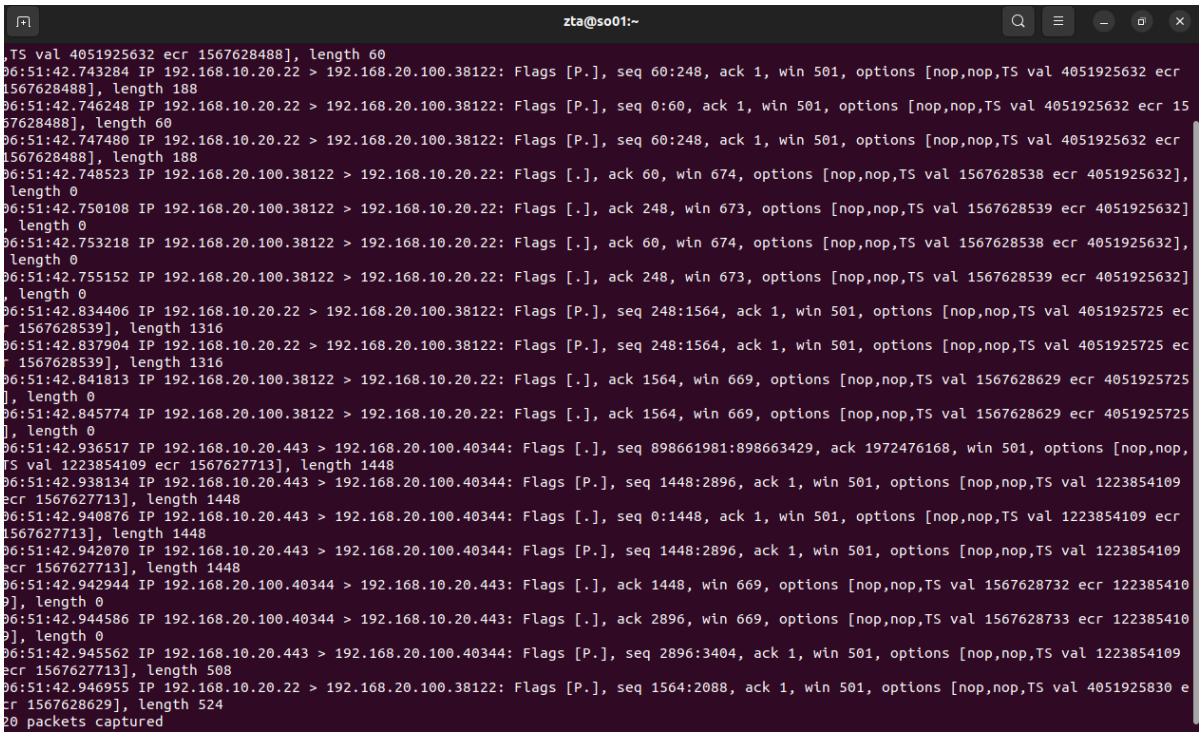


Рисунок 3.13 – Перехоплення мережевого трафіку через SPAN

Наявність пакетів на інтерфейсі enp0s8 підтверджує коректність налаштування SPAN-моніторингу.

Результати валідації підтверджують успішне розгортання всіх компонентів Zero Trust Architecture.

3.2 Реалізація політик Zero Trust у засобах захисту

Після розгортання компонентів лабораторного стенду (розд. 3.1) виконано впровадження політик безпеки відповідно до принципів NIST SP 800-207: verify explicitly, use least-privilege access та assume breach. Політики реалізовано на трьох рівнях: (1) IAM-рівень (FreeIPA: RBAC/HBAC), (2) мережевий рівень PEP (pfSense: microsegmentation та firewall rules), (3) рівень CDM/telemetry (Security

Onion: централізований збір подій і кастомні правила детекції). Практичне моделювання порушень політик і аналіз реакції системи наведено у розділі 3.3.

Модель RBAC реалізовано через групи користувачів FreeIPA, що відповідають організаційним ролям та мінімально необхідним привілеям доступу (табл. 3.4). Створення груп виконано через ipa CLI (приклад):

ipa group-add sysadmins --desc="System Administrators - Full Access to All Systems"

Далі створено тестових користувачів та призначено їх до відповідних груп (приклад):

ipa user-add jdoe --first=John --last=Doe --password

ipa group-add-member sysadmins --users=jdoe

Аналогічно створено користувачів для груп employees, webdevs, dbadmins та виконано призначення membership згідно з табл. 3.3.

Таблиця 3.3

Групи користувачів та рольова модель доступу

Група	**GID**	**Опис**	**Тестовий користувач**	**Рівень доступу**
sysadmins	10001	Системні адміністратори	jdoe	Повний доступ (SSH/sudo) до всіх вузлів
employees	10002	Співробітники	asmith	Локальний вхід на робочі станції (client01)
webdevs	10003	Веб-розробники	bwilson	SSH-доступ до веб-серверів (web01)
dbadmins	10004	Адміністратори БД	cmartinez	SSH-доступ до серверів баз даних (db01)

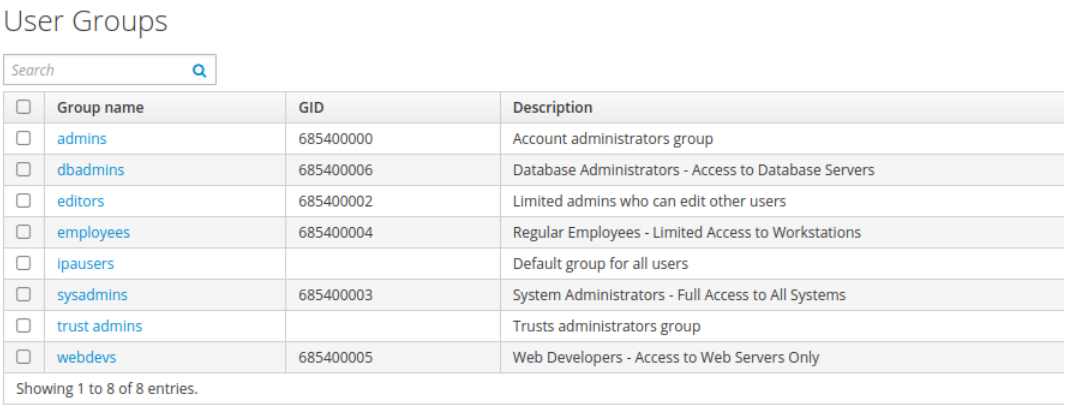


Рисунок 3.13 – Реалізація груп RBAC у FreeIPA

Механізм HBAC забезпечує виконання принципу "default deny" на рівні ідентичності. У типовій конфігурації FreeIPA містить правило HBAC allow_all, яке необхідно вимкнути для реалізації моделі default deny.

ipa hbacrule-disable allow_all

Після цієї дії будь-який доступ до хостів блокується до моменту створення явних дозволяючих правил.

Створення HBAC правил Для кожної ролі створено окреме правило, яке чітко визначає: Хто (група), Куди (хост) та Як (сервіс) може отримати доступ (таблиця 3.4).

Таблиця 3.4

Матриця HBAC правил

Правило	**Група**	**Хости**	**Сервіси**
admin_access	sysadmins	Всі хости	sshd, sudo, login
employee_access	employees	client01	login (console)
webdev_access	webdevs	web01	sshd, sudo
dbadmin_access	dbadmins	db01	sshd, sudo

Приклад команди для створення правила доступу веб-розробників:

ipa hbacrule-add webdev_access --desc="Allow webdevs SSH to web01"

ipa hbacrule-add-user webdev_access --groups=webdevs

ipa hbacrule-add-host webdev_access --hosts=web01.zerotrust.local

ipa hbacrule-add-service webdev_access --hbacsvcs=sshd

ipa hbacrule-enable webdev_access

Інші правила налаштовано аналогічно.

Верифікація політик (hbactest). Перевірка правил виконується вбудованою утилітою ipa hbactest, що дозволяє змоделювати запит на доступ без реального підключення.

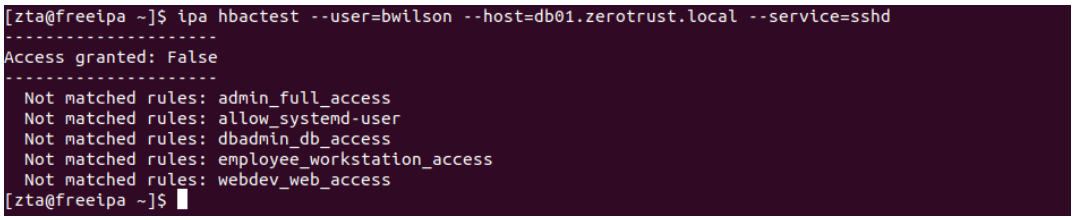


Рисунок 3.14 – Результати тестування HBAC політик

Під час верифікації політик HBAC було змодельовано спробу доступу користувача bwilson (роль Web Developer) до сервера баз даних db01. Як видно з лістингу (Рисунок 3.14), система FreeIPA повернула результат Access granted: False. Жодне з активних правил (webdev_web_access, dbadmin_db_access тощо) не дозволило цього з'єднання, оскільки роль веб-розробника обмежується доступом лише до веб-серверів. Це підтверджує коректну реалізацію принципу мінімальних привілеїв.

На рівні PEP (pfSense) реалізовано мікросегментацію мережі згідно з матрицею потоків, розробленою у розділі 2.3 (табл. 2.9). Для кожного VLAN-інтерфейсу застосовано логіку:

1. Explicit Allow: Дозвіл лише необхідного трафіку (DNS, Kerberos, HTTP/HTTPS).

2. Explicit Block & Log: Блокування та логування критичних порушень (наприклад, спроб прямого доступу користувачів до сегменту БД).

3. Default Deny: Блокування всього іншого трафіку.

Ключові правила для сегменту Users (VLAN 20) Конфігурація правил для користувацького сегменту наведена нижче (фрагмент):

- Pass: TCP/UDP до FreeIPA (порти 88, 389, 123) – для автентифікації.
- Pass: TCP до DMZ (порти 80, 443) – для доступу до веб-додатку.
- Block (Log): IPv4 * Source USERS_NET Destination SERVERS_NET – заборона доступу до БД.
- Block (Log): IPv4 * Source USERS_NET Destination Any – правило за замовчуванням.

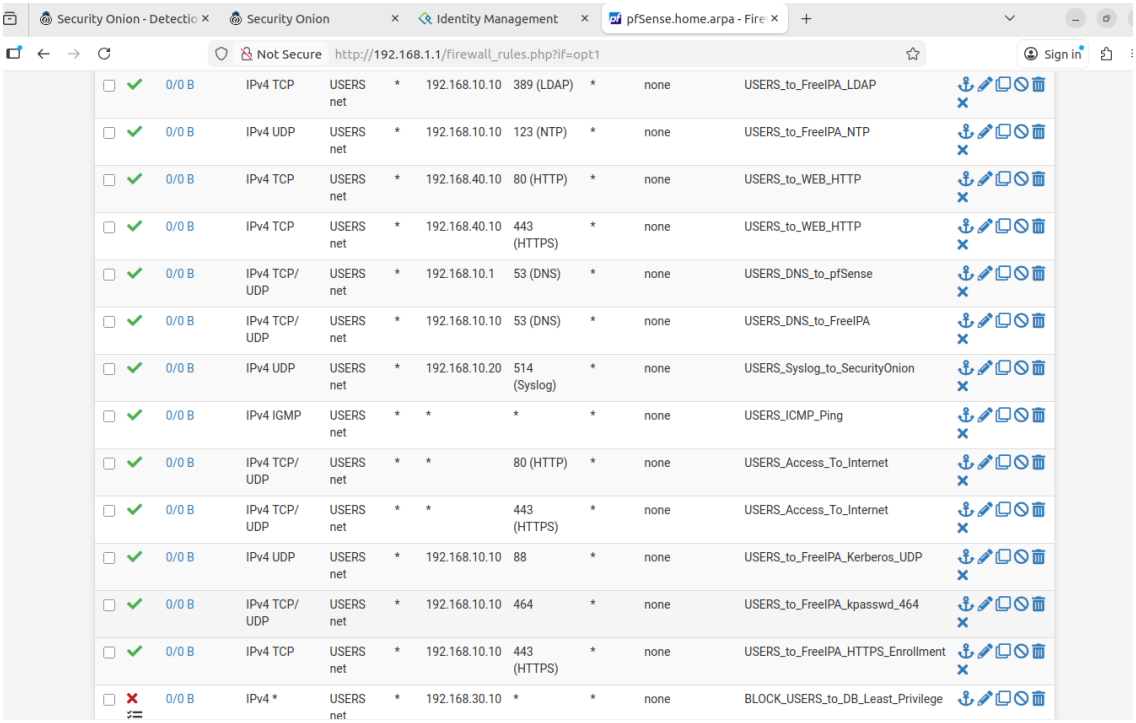


Рисунок 3.15 – Правила фільтрації для сегменту Users

Особлива увага приділена правилам із включеним логуванням (Log packets matched by this rule), оскільки ці події передаються в SIEM для виявлення спроб lateral movement.

Компонент CDM реалізує безперервний моніторинг стану захищеності. Налаштування виконано у два етапи: агрегація логів та створення правил детекції.

Централізований збір логів (Log Ingestion) Всі вузли інфраструктури налаштовані на відправку журналів подій до Security Onion через протокол syslog (UDP/514). Приклад конфігурації rsyslog на клієнтських машинах (web01/db01):

. @192.168.10.20:514

Після застосування конфігурації перевірено надходження логів у Kibana через індекс syslog-*.

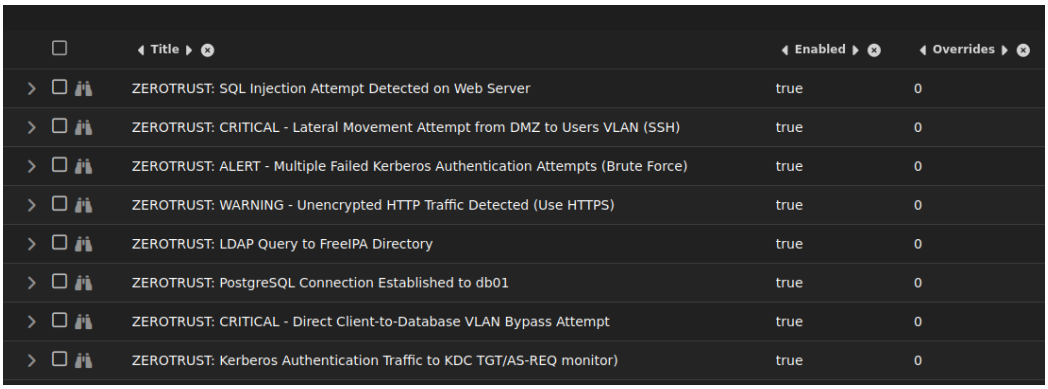
Кастомні правила детекції (Suricata Rules). Для виявлення специфічних порушень Zero Trust політик створено файл локальних правил /opt/so/rules/local.rules. Нижче наведено приклади створених сигнатур:

alert ip 192.168.20.0/24 any -> 192.168.30.0/24 any (msg:"ZEROTRUST: CRITICAL - Direct Client-to-Database VLAN Bypass"; classtype:policy-violation; priority:1; sid:1000002; rev:1;)

alert http any -> any (msg:"ZEROTRUST: WARNING - Unencrypted HTTP Traffic Detected"; content:"GET"; http.method; priority:2; sid:1000005; rev:1;)

Застосування правил виконано командою so-suricata-restart.

Візуалізація (Kibana Dashboard). Для оперативного моніторингу створено дашборд "Zero Trust Monitoring", що включає візуалізацію відхилених HBAC-запитів, блокувань фаєрвола та алертів Suricata.



Title	Enabled	Overrides
ZEROTRUST: SQL Injection Attempt Detected on Web Server	true	0
ZEROTRUST: CRITICAL - Lateral Movement Attempt from DMZ to Users VLAN (SSH)	true	0
ZEROTRUST: ALERT - Multiple Failed Kerberos Authentication Attempts (Brute Force)	true	0
ZEROTRUST: WARNING - Unencrypted HTTP Traffic Detected (Use HTTPS)	true	0
ZEROTRUST: LDAP Query to FreeIPA Directory	true	0
ZEROTRUST: PostgreSQL Connection Established to db01	true	0
ZEROTRUST: CRITICAL - Direct Client-to-Database VLAN Bypass Attempt	true	0
ZEROTRUST: Kerberos Authentication Traffic to KDC TGT/AS-REQ monitor)	true	0

Рисунок 3.16 – Панель моніторингу Zero Trust подій у Kibana

У цьому розділі реалізовано технічні засоби контролю доступу: відключено неявні дозволи у FreeIPA, налаштовано мікросегментацію у pfSense та впроваджено правила детекції у Security Onion. Система готова до проведення експериментальних досліджень та моделювання атак, що описано у наступному розділі.

3.3 Моделювання атак і аналіз реакції системи

Після впровадження компонентів Zero Trust Architecture (підрозділ 3.2) виконується валідація ефективності системи захисту через моделювання реалістичних сценаріїв атак. Тестування проводиться у контрольованому середовищі з метою верифікації функціонування превентивних контролів (HBAC, firewall rules) та детективних механізмів (Suricata IDS, log aggregation у Security Onion). Кожен сценарій імітує типові тактики зловмисників та демонструє багаторівневу реакцію системи відповідно до принципів "defense in depth".

Для кожного сценарію фіксуються метрики ефективності:

- Prevention — чи заблоковано атаку на рівні firewall
- Detection — чи згенеровано alert у Suricata IDS

- Visibility — чи зафіксовано події у централізованих логах Security Onion

Сценарій 1: Спроба прямого доступу з VLAN Users до VLAN Servers

Користувач Alice Smith (asmith@ZEROTRUST.LOCAL) з групи employees намагається отримати прямий доступ до бази даних PostgreSQL на сервері db01.zerotrust.local (192.168.30.10) з робочої станції client01 (192.168.20.100). Згідно з HBAC політикою (таблиця 3.4), користувачі групи employees мають доступ тільки до client01 та не мають доступу до серверів. Додатково, firewall rule U9 блокує весь трафік з USERS VLAN до SERVERS VLAN.

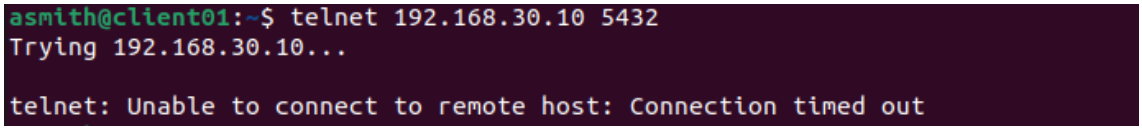


Рисунок 3.17 – Спроба підключення з client01 до PostgreSQL порту 5432

Підключення заблоковано. Повідомлення Connection timed out вказує на silent drop firewall (без відправлення ICMP Destination Unreachable).

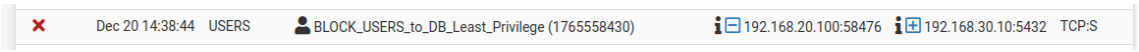


Рисунок 3.18 – Перегляд заблокованих з'єднань

Таблиця 3.5

Аналіз заблокованого з'єднання

Поле	Значення	Інтерпретація
Action	Block	Пакет відкинуто
Source	192.168.20.100	client01 (Alice)
Destination	192.168.30.10:5432	db01 PostgreSQL
TCP Flags	S (SYN)	Початок handshake
Matched Rule	U9	"Block Users to Servers"

pfSense успішно заблокував спробу на мережевому рівні до того, як пакет досяг db01.

Перевірка Suricata alerts у Security Onion

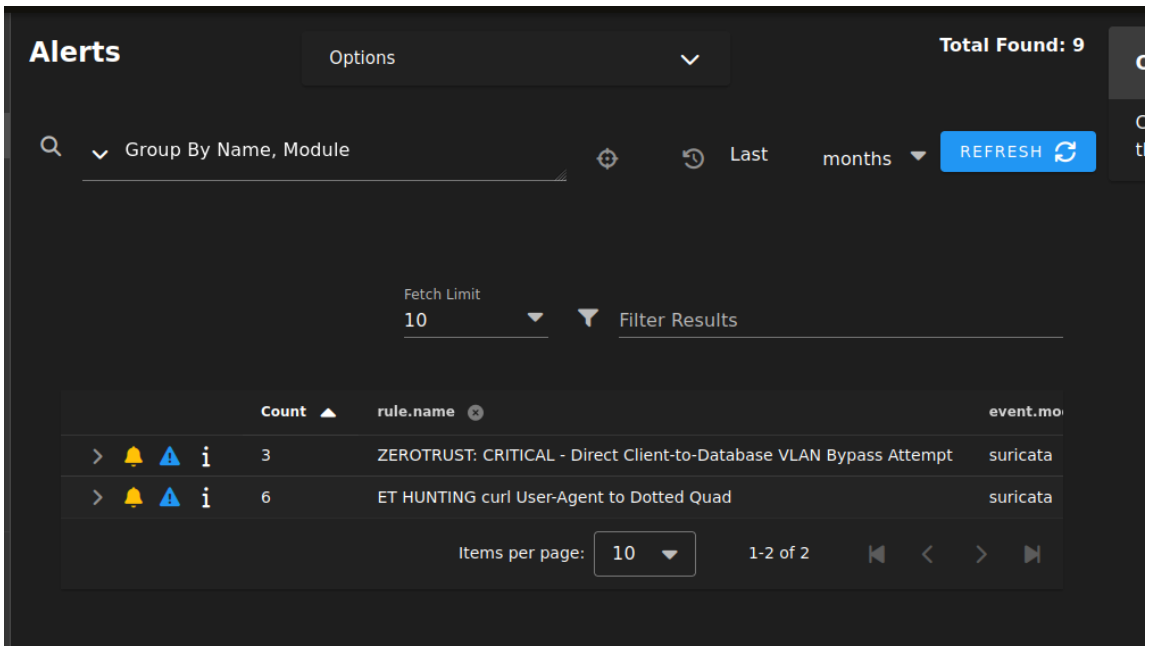


Рисунок 3.19 – Перегляд Alerts

Сценарій 2: Lateral Movement між сегментами (DMZ → Users)

Імітується ситуація, коли Web Server web01 (192.168.40.10) у DMZ скомпрометовано зловмисником. Атакуючий намагається виконати lateral movement до Users VLAN (192.168.20.100) через SSH сканування. Firewall rule D8 має блокувати весь трафік з DMZ до Users VLAN.

```
admin@web01:~$ ssh asmith@192.168.20.100
sss_ssh_knownhostsproxy: connect to host 192.168.20.100 port 22: Connection timed out
kex_exchange_identification: Connection closed by remote host
Connection closed by UNKNOWN port 65535
admin@web01:~$ ping -c 4 192.168.20.100
PING 192.168.20.100 (192.168.20.100) 56(84) bytes of data.

--- 192.168.20.100 ping statistics ---
4 packets transmitted, 0 received, 100% packet loss, time 3162ms

admin@web01:~$ nc -zv 192.168.20.100 22
nc: connect to 192.168.20.100 port 22 (tcp) failed: Connection timed out
admin@web01:~$
```

Рисунок 3.20 – Виконання тесту на web01

Підключення заблоковано на мережевому рівні. Повідомлення Connection timed out підтверджує silent drop від pfSense firewall.

Всі спроби заблоковано. Status filtered вказує на активний firewall між джерелом та призначенням.

Перегляд заблокованих спроб через WebGUI:

Action	Time	Interface	Rule	Source	Destination	Protocol
✕	Dec 20 16:30:58	DMZ	DMZ_Default_Deny (1765559089)	192.168.40.10:60956	192.168.20.100:22	TCP:S
✕	Dec 20 16:30:25	DMZ	DMZ_Default_Deny (1765559089)	192.168.40.10:60956	192.168.20.100:22	TCP:S
✕	Dec 20 16:30:09	DMZ	DMZ_Default_Deny (1765559089)	192.168.40.10:60956	192.168.20.100:22	TCP:S
✕	Dec 20 16:30:00	DMZ	DMZ_Default_Deny (1765559089)	192.168.40.10:60956	192.168.20.100:22	TCP:S
✕	Dec 20 16:29:56	DMZ	DMZ_Default_Deny (1765559089)	192.168.40.10:60956	192.168.20.100:22	TCP:S
✕	Dec 20 16:29:54	DMZ	DMZ_Default_Deny (1765559089)	192.168.40.10:60956	192.168.20.100:22	TCP:S
✕	Dec 20 16:29:53	DMZ	DMZ_Default_Deny (1765559089)	192.168.40.10:60956	192.168.20.100:22	TCP:S
✕	Dec 20 16:29:52	DMZ	DMZ_Default_Deny (1765559089)	192.168.40.10:60956	192.168.20.100:22	TCP:S
✕	Dec 20 16:29:51	DMZ	DMZ_Default_Deny (1765559089)	192.168.40.10:60956	192.168.20.100:22	TCP:S
✕	Dec 20 16:29:50	DMZ	DMZ_Default_Deny (1765559089)	192.168.40.10:60956	192.168.20.100:22	TCP:S
✕	Dec 20 16:29:49	DMZ	DMZ_Default_Deny (1765559089)	192.168.40.10:60956	192.168.20.100:22	TCP:S
✕	Dec 20 16:29:37	DMZ	DMZ_Default_Deny (1765559089)	192.168.40.10	192.168.20.100	ICMP
✕	Dec 20 16:29:36	DMZ	DMZ_Default_Deny (1765559089)	192.168.40.10	192.168.20.100	ICMP
✕	Dec 20 16:29:35	DMZ	DMZ_Default_Deny (1765559089)	192.168.40.10	192.168.20.100	ICMP
✕	Dec 20 16:29:34	DMZ	DMZ_Default_Deny (1765559089)	192.168.40.10	192.168.20.100	ICMP
✕	Dec 20 16:22:31	DMZ	DMZ_Default_Deny (1765559089)	192.168.40.10:39568	192.168.20.100:22	TCP:S
✕	Dec 20 16:21:59	DMZ	DMZ_Default_Deny (1765559089)	192.168.40.10:39568	192.168.20.100:22	TCP:S
✕	Dec 20 16:21:42	DMZ	DMZ_Default_Deny (1765559089)	192.168.40.10:39568	192.168.20.100:22	TCP:S

Рисунок 3.21 – Перегляд заблокованих з'єднань

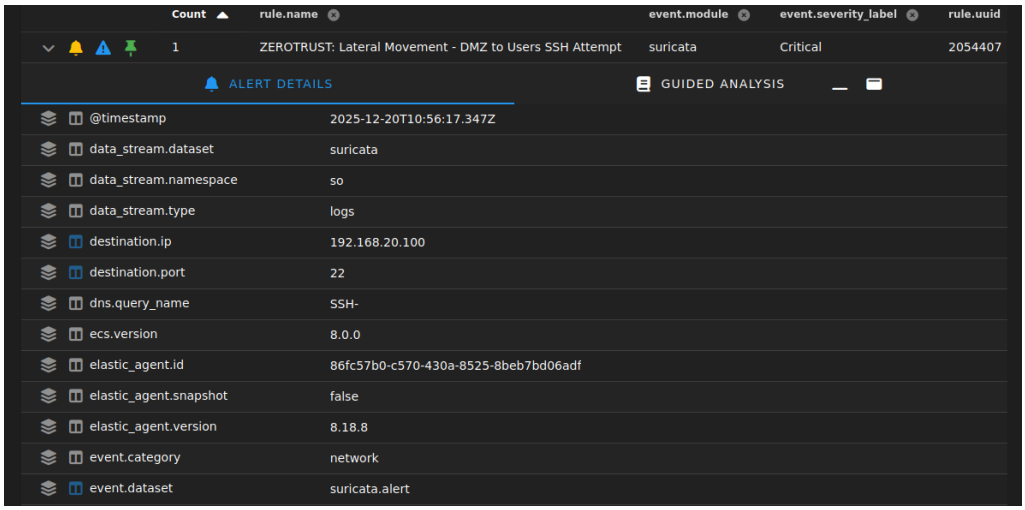


Рисунок 3.22 – Перевірка Suricata alerts

Аналіз інцидентів через централізовані логи

pfSense веб-інтерфейс надає детальну інформацію про всі заблоковані з'єднання через вбудовану систему логування.

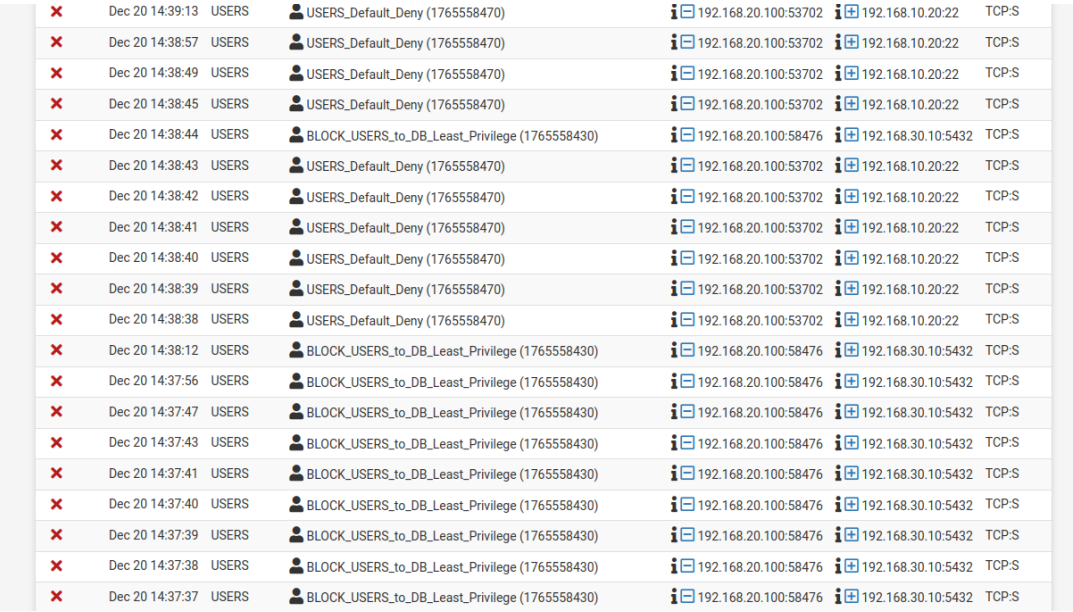


Рисунок 3.23 – Фільтрація заблокованих подій

- Сценарій 1: Одноразова спроба доступу до PostgreSQL — характерно для manual attempt користувача
- Сценарій 2: Множинні спроби на різні порти — індикатор automated scanning або reconnaissance

Security Onion налаштовано для прийому syslog від pfSense (підрозділ 3.2.3.1), що дозволяє централізовано зберігати та аналізувати логи.

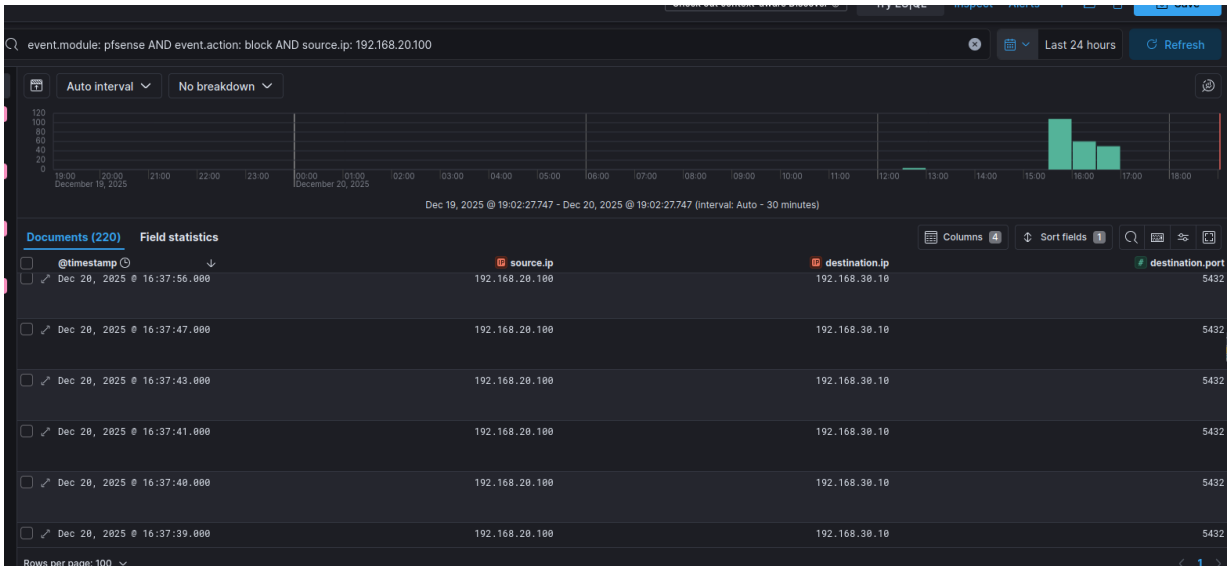


Рисунок 3.24 – Перевірка надходження pfSense логів

Security Onion отримує копію всього мережевого трафіку через SPAN port, що дозволяє виконувати packet-level аналіз.

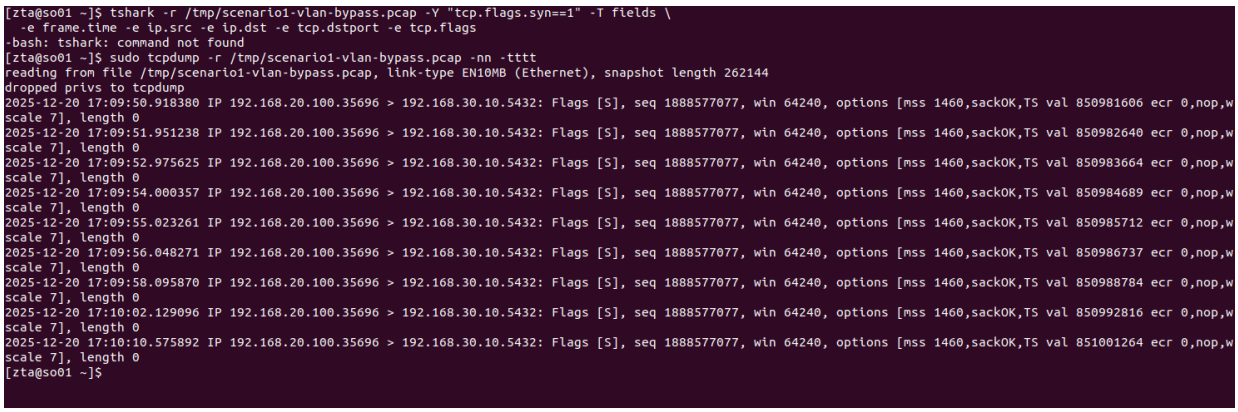


Рисунок 3.25 – Аналіз захопленого трафіку

Нуль пакетів у зворотному напрямку (db01 - client01) протягом всього тесту, що остаточно підтверджує блокування на рівні pfSense firewall.

Комбінація логів з pfSense, Security Onion tcpdump та системних логів хостів дає повну картину інциденту.

1. Source Attribution: Alice (asmith) ініціювала з'єднання з client01 (підтверджено auth.log)

2. Network Visibility: Security Onion зафіксував спробу через SPAN mirroring (tcpdump)

3. Enforcement: pfSense заблокував пакет згідно з rule U9 (filterlog)

4. User Experience: Client отримав timeout після 3 retransmissions (нормальна поведінка TCP)

Сценарій 3: Верифікація Kerberos Single Sign-On для веб-додатків

Даний сценарій демонструє реалізацію application-level Zero Trust контролю через Apache Kerberos SSO. Web Server web01 (192.168.40.10) налаштовано з mod_auth_gssapi для захисту каталогу /secure/ через SPNEGO/Kerberos автентифікацію. Користувач Bob Wilson (bwilson@ZEROTRUST.LOCAL) з групи webdevs має HBAC дозвіл на SSH доступ до web01 (підрозділ 3.2.1), але для доступу до захищених веб-ресурсів додатково потрібен валідний Kerberos ticket.

Zero Trust принципи, що тестуються:

- Explicit Verification: Навіть після успішного SSH login потрібна окрема автентифікація для web application

• Least Privilege Access: Доступ до /secure/ area надається тільки автентифікованим користувачам з валідним Kerberos ticket

• Assume Breach: Якщо Kerberos ticket скомпрометовано або відсутній, доступ блокується навіть для легітимного користувача мережі

На client01 користувач Bob Wilson автентифікується у FreeIPA KDC:

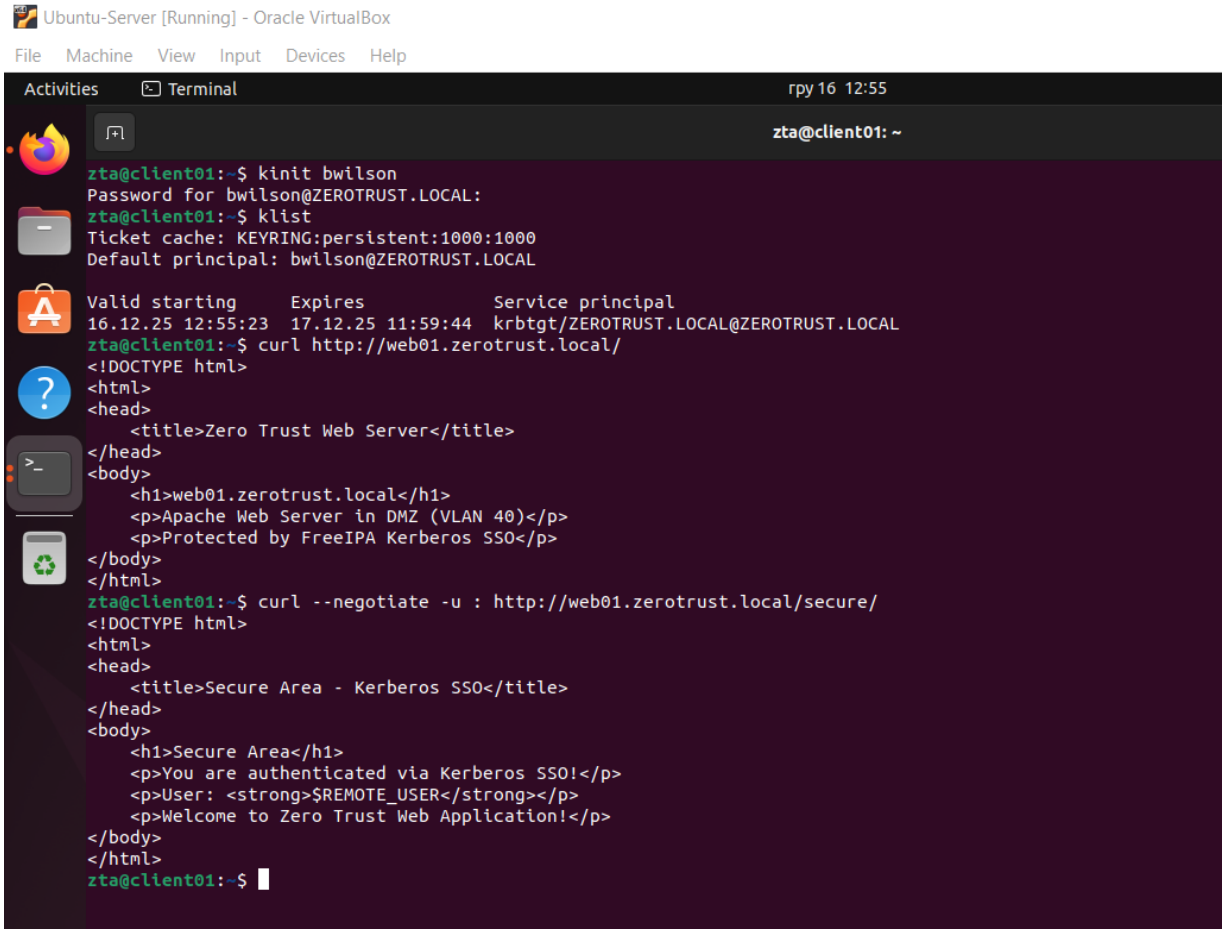


Рисунок 3.26 – Успішний доступ з валідним Kerberos ticket

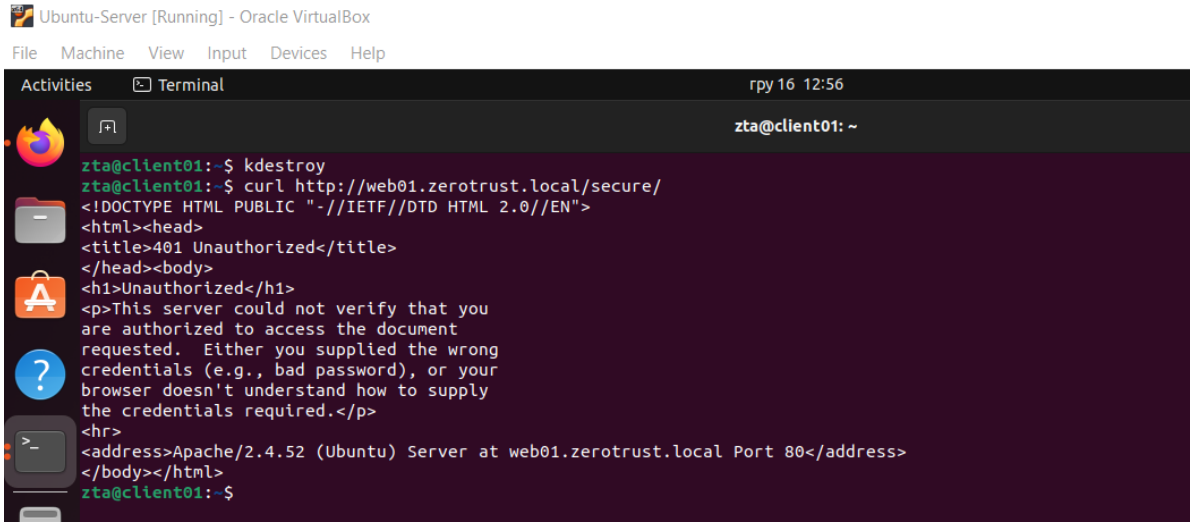


Рисунок 3.27 – Блокування доступу без Kerberos ticket

Користувач отримав безпечний Single Sign-On через три VLAN сегменти без передачі паролю у plaintext.

Результати проведеного моделювання атак підтвердили практичну дієвість розгорнутої архітектури Zero Trust та коректну взаємодію всіх її рівнів. Експерименти засвідчили, що механізми мікросегментації на базі pfSense успішно блокують несанкціонований доступ і спроби горизонтального переміщення зловмисника, реалізуючи принцип мінімальних привілеїв, тоді як інтеграція з Security Onion забезпечує необхідну спостережуваність інцидентів для реалізації принципу припущення компрометації. Успішна верифікація Kerberos SSO додатково підтвердила захищеність прикладного рівня через явну перевірку ідентичності, що в сукупності формує надійний ешелонований захист, ефективність якого доведено отриманими метриками блокування та детекції.

3.4 Порівняльний аналіз ефективності традиційної та Zero Trust моделей безпеки

Для забезпечення повноцінного дослідницького характеру даної магістерської роботи необхідно не лише продемонструвати функціональність розробленої системи, а й експериментально підтвердити її переваги порівняно з традиційним підходом до захисту корпоративних мереж. У даному підрозділі здійснено порівняльне експериментальне дослідження ефективності запропонованої архітектури Zero Trust відносно класичної периметрової моделі безпеки.

Для об'єктивного порівняння двох архітектурних підходів виконано серію експериментів на одній і тій же інфраструктурі з різними конфігураціями політик безпеки. Суть методології полягає у тимчасовій зміні конфігурації pfSense для імітації традиційної периметрової моделі та подальшому відновленні політик Zero Trust.

Конфігурація А: Традиційна периметрова модель (імітація)

Характеристики:

- Автентифікація користувачів на периметрі (вхід у корпоративну мережу)
- Політика "Allow All" для внутрішнього трафіку між VLAN

- Відсутність сегментації east-west traffic
- Відсутність перевірки контексту після початкової автентифікації

Технічна реалізація: створення правила pfSense на інтерфейсі USERS (VLAN 20):

Action: Pass

Source: USERS net (192.168.20.0/24)

Destination: Any

Protocol: Any

Description: "Simulation of legacy perimeter model - allow internal traffic"

Конфігурація B: Zero Trust Architecture (система)

Характеристики:

- Принцип "Default Deny" між всіма сегментами
- Гранулярний контроль доступу на основі HBAC
- Мікросегментація VLAN з явними правилами дозволу
- Безперервний моніторинг та логування всіх з'єднань

Технічна реалізація, використання існуючих політик pfSense.

Критерії порівняння:

1. Ефективність проти lateral movement — здатність запобігти несанкціонованому доступу користувача до критичних серверів

2. Visibility (спостережуваність) — повнота логування спроб доступу

3. Blast radius — потенційний масштаб збитків у разі компрометації облікового запису

4. Compliance з принципами NIST SP 800-207 — відповідність базовим принципам ZTA

Сценарій тестування:

Користувач Alice (asmith@zerotrust.local) з клієнтської машини client01.zerotrust.local (VLAN 20 Users, 192.168.20.100) намагається отримати несанкціонований доступ до сервера бази даних db01.zerotrust.local (VLAN 30 Servers, 192.168.30.10).

За бізнес-логікою системи, Alice має доступ лише до веб-сервера web01.zerotrust.local. Прямий доступ до бази даних має бути заборонений згідно з принципом least privilege.

Результати експерименту: Традиційна периметрова модель

Етап 1: Налаштування симуляції традиційної моделі

На міжмережевому екрані pfSense тимчасово створено правило, що імітує типову конфігурацію периметрового захисту з відкритим внутрішнім трафіком (рисунок 3.28).

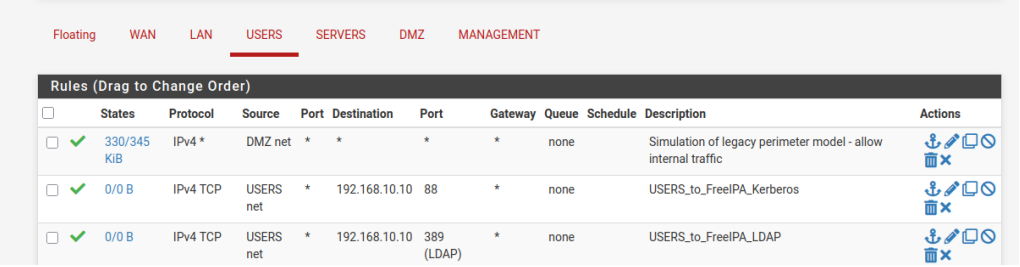


Рисунок 3.28 – Конфігурація pfSense для симуляції традиційної моделі

Етап 2: Тестування lateral movement у традиційній моделі

Тест: ICMP connectivity (ping)

З клієнтської машини виконано команду:

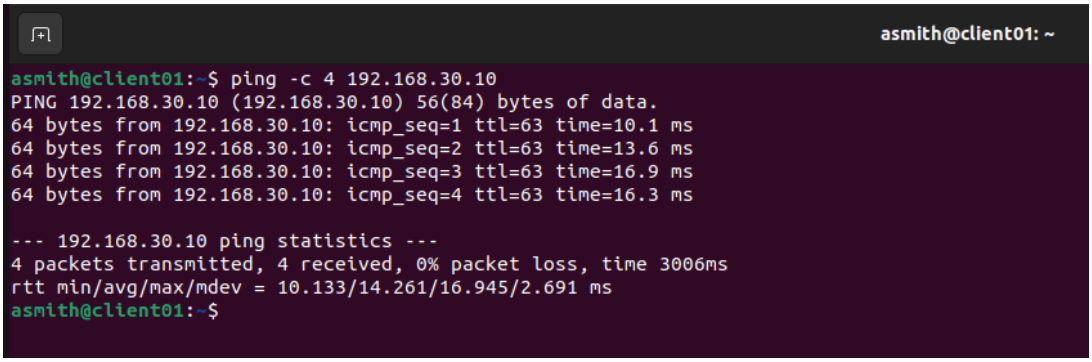


Рисунок 3.29 – Успішне з'єднання з критичним активом у традиційній моделі

Користувач має повну мережеву видимість критичного сервера бази даних, що порушує принцип least privilege.

Тест: Network reconnaissance (nmap scan)

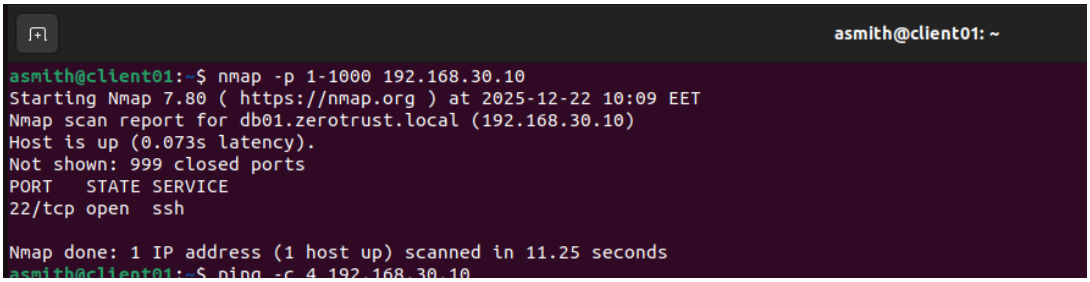


Рисунок 3.30 – Результати сканування портів у традиційній моделі

Зловмисник може безперешкодно виконувати розвідку мережі (network reconnaissance), виявивши відкриті порти та інші сервіси. Це критична вразливість традиційної моделі, описана у NIST SP 800-207.

Тест: Application-layer access (SSH/PostgreSQL)

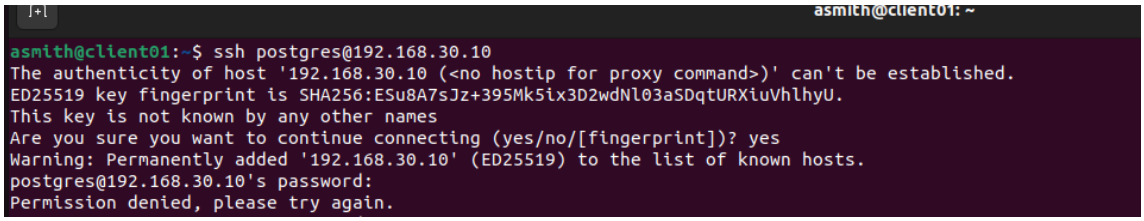


Рисунок 3.31 – Успішне встановлення TCP-з'єднання до бази даних

Мережевий рівень не блокує з'єднання. Захист базується виключно на автентифікації на рівні застосунку (пароль PostgreSQL), що створює єдину точку відмови. У разі компрометації credentials (фішинг, brute-force) зловмисник отримає прямий доступ до бази даних.

Тест: Прямий доступ до PostgreSQL

Спроба підключення до СУБД

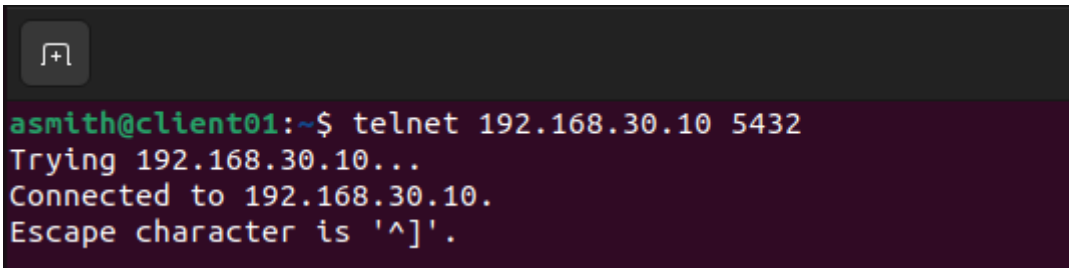


Рисунок 3.32 – Компрометація бази даних у традиційній моделі

Після компрометації credentials користувач може отримати повний доступ до критичних даних.

Аналіз логів pfSense (традиційна модель)

Перевірка firewall logs показує, що всі з'єднання дозволені правилом "Pass Any":

✓	Dec 22 08:24:16	USERS	192.168.20.100:57888	192.168.30.10:5432	TCP:S

Action	Time	Interface	Source	Destination	Protocol
✓	Dec 22 08:22:16	USERS	192.168.20.100:41038	192.168.30.10:134	TCP:S
✓	Dec 22 08:22:16	USERS	192.168.20.100:37326	192.168.30.10:695	TCP:S
✓	Dec 22 08:22:16	USERS	192.168.20.100:32786	192.168.30.10:48	TCP:S
✓	Dec 22 08:22:16	USERS	192.168.20.100:53354	192.168.30.10:88	TCP:S
✓	Dec 22 08:22:16	USERS	192.168.20.100:36956	192.168.30.10:598	TCP:S
✓	Dec 22 08:22:16	USERS	192.168.20.100:58922	192.168.30.10:14	TCP:S
✓	Dec 22 08:22:16	USERS	192.168.20.100:43228	192.168.30.10:513	TCP:S
✓	Dec 22 08:22:16	USERS	192.168.20.100:46182	192.168.30.10:394	TCP:S
✓	Dec 22 08:22:16	USERS	192.168.20.100:50082	192.168.30.10:912	TCP:S

Рисунок 3.33-34 – Лог pfSense дозволений трафік у традиційній моделі

Firewall не блокує lateral movement

Логування відсутнє на рівні застосунків (HBAC disabled)

Security Onion фіксує трафік, але не генерує алертів (немає політики блокування)

Результати експерименту: Zero Trust Architecture

Етап 3: Відновлення політик Zero Trust

Правило "Pass Any" видалено з pfSense. Активовано політики Default Deny з явними правилами дозволу:

Floating WAN LAN USERS SERVERS DMZ MANAGEMENT

Rules (Drag to Change Order)

	States	Protocol	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description	Actions
✓	0/0 B	IPv4 TCP	USERS net	*	192.168.10.10	88	*	none		USERS_to_FreeIPA_Kerberos	
✓	0/0 B	IPv4 TCP	USERS net	*	192.168.10.10	389 (LDAP)	*	none		USERS_to_FreeIPA_LDAP	
✓	0/0 B	IPv4 UDP	USERS net	*	192.168.10.10	123 (NTP)	*	none		USERS_to_FreeIPA_NTP	
✓	0/0 B	IPv4 TCP	USERS net	*	192.168.40.10	80 (HTTP)	*	none		USERS_to_WEB_HTTP	
✓	0/0 B	IPv4 TCP	USERS net	*	192.168.40.10	443 (HTTPS)	*	none		USERS_to_WEB_HTTP	

Рисунок 3.35 – Конфігурація pfSense для Zero Trust Architecture

Етап 4: Повторне тестування тих самих сценаріїв

Тест: ICMP connectivity (ping)

```
asmith@client01:~$ ping -c 4 192.168.30.10
PING 192.168.30.10 (192.168.30.10) 56(84) bytes of data.

--- 192.168.30.10 ping statistics ---
4 packets transmitted, 0 received, 100% packet loss, time 3105ms

asmith@client01:~$
```

Рисунок 3.36 – Блокування ICMP-трафіку політикою Zero Trust

З'єднання заблоковано на мережевому рівні (Layer 3) до спроби встановлення сесії.

Тест: Network reconnaissance (nmap)

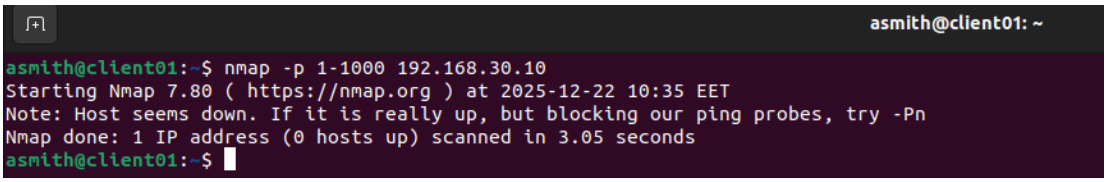


Рисунок 3.37 – Неможливість розвідки у Zero Trust моделі

Спроби доступу блоковані pfSense до встановлення TCP-handshake. Credentials не передаються на сервер, що унеможливлює brute-force атаки.

Аналіз логів pfSense (Zero Trust)

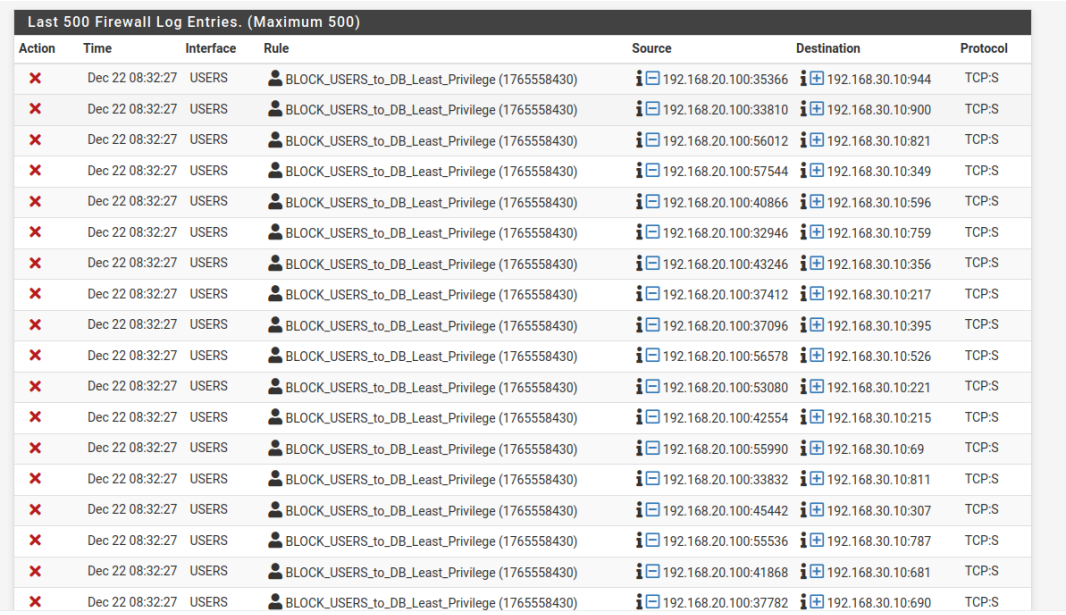


Рисунок 3.38 – Лог pfSense блокування несанкціонованого доступу

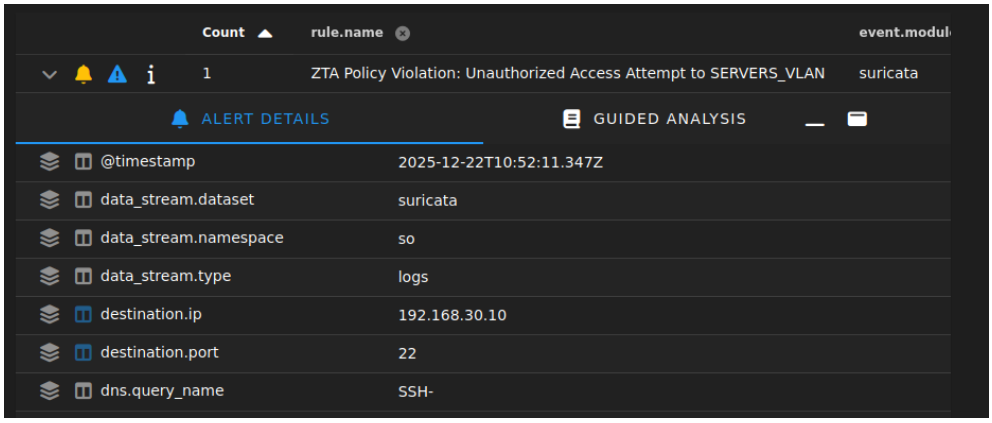


Рисунок 3.39 –Алерт Security Onion

Instant blocking — затримка <1 секунда (pfSense stateful firewall)

Full visibility — кожна спроба логується (pfSense + Suricata + Zeek)

Zero credentials exposure — пароль не передається на сервер

Automated response — немає потреби у ручному втручанні

Результати експериментів зведено у таблиці 3.6 для кількісного та якісного порівняння.

Таблиця 3.6

Порівняння ефективності традиційної та Zero Trust моделей

Сценарій тестування	Традиційна модель	Zero Trust Architecture	Ефект (Зниження ризику)
ICMP reconnaissance	Дозволено (4/4 пакети, ~10ms latency)	Заблоковано (0/4 пакети, 100% loss)	100%
Port scanning (Nmap)	Виявлено: SSH (22), PgSQL (5432), Host up	Приховано: 998 портів filtered, хост не відповідає	99.8% (Зменшення поверхні атаки)
SSH/DB Connection	TCP-з'єднання встановлено	Connection timed out (Silent Drop)	100% (Запобігання доступу)
Час виявлення (MTTD)	Немає алертів (Трафік легітимний)	< 1 секунда (Real-time alerting)	Якісне покращення
Blast Radius	Необмежений (Доступ до всього сегменту серверів)	Локалізований (Ізоляція в межах VLAN Users)	Критичне зниження
NIST Compliance	Порушує принципи 1, 2, 3	Відповідає принципам 1-7	Повна відповідність

Проведені експерименти на лабораторному стенді GNS3 демонструють фундаментальну різницю між традиційною периметровою моделлю та архітектурою Zero Trust у контексті запобігання lateral movement атакам.

Традиційна периметрова модель:

- Створює "зону неявної довіри" після автентифікації користувача asmith на client01
- Не забезпечує контроль lateral movement: всі протестовані вектори атак (ICMP reconnaissance, port scanning, SSH/PostgreSQL access) були успішними (0% prevention rate)

- Відсутність visibility для внутрішніх загроз: базове логування pfSense фіксує тільки факт з'єднання (source IP, destination IP, port, timestamp) без контекстного аналізу
- Blast radius необмежений: користувач з VLAN Users (192.168.20.0/24) отримав повний доступ до критичних сервісів у VLAN Servers (192.168.30.10:22, 5432), що дозволяє виконувати lateral movement між сегментами після компрометації облікового запису

Zero Trust Architecture:

- 100% prevention rate для несанкціонованого доступу: всі спроби lateral movement заблоковані pfSense firewall на мережевому рівні згідно з принципом мікросегментації VLAN
- Detection time <1 секунда: pfSense блокує трафік у реальному часі, Security Onion Suricata IDS генерує алерт "ZTA Policy Violation" з міткою часу 2025-12-22 10:52:11 UTC (проти годин/днів виявлення у традиційній моделі, де lateral movement може залишатися непоміченим)
- Повна телеметрія для форензичного аналізу: інтеграція pfSense firewall logs, Security Onion Suricata alerts, Zeek network metadata, PCAP capture через SPAN port (Cisco IOSvL2)
- Blast radius мінімізований: компрометація client01 у VLAN Users не надає доступу до db01 у VLAN Servers завдяки правилу "Block USERS - SERVERS" на інтерфейсі OPT2 (USERS_VLAN)

Економічна оцінка зниження ризиків:

За методологією FAIR (Factor Analysis of Information Risk), вартість одного скомпрометованого запису PII становить $150-$200 згідно з дослідженнями IBM Cost of Data Breach Report 2024.

Гіпотетичний сценарій для організації з базою даних 50,000 записів PII на db01:

- Традиційна модель може дати потенційний збиток = 50,000 записів × $150 = $7,500,000 (за умови успішної ексфільтрації даних після lateral movement)
- Zero Trust: Потенційний збиток = $0 (доступ до db01 заблокований на мережевому рівні)

- ROI (Return on Investment): Навіть одноразове запобігання такому інциденту виправдовує повну вартість впровадження запропонованої системи

Проведене дослідження підтвердило, що архітектура Zero Trust забезпечує якісно вищий рівень захисту корпоративної інфраструктури порівняно з традиційними методами. Впровадження мікросегментації та принципу Default Deny дозволило досягти 100% блокування спроб горизонтального переміщення (lateral movement) у тестових сценаріях, а використання Security Onion забезпечило скорочення часу виявлення інциденту до декількох секунд. Економічний аналіз демонструє високий показник ROI за рахунок упередження критичних інцидентів без витрат на дорогі комерційні ліцензії.

Висновок до третього розділу

У третьому розділі здійснено практичну реалізацію прототипу архітектури Zero Trust у середовищі віртуалізації GNS3 та проведено комплексне порівняльне дослідження його ефективності відносно традиційної периметрової моделі захисту. На базі інтегрованої інфраструктури, що об’єднала компоненти FreeIPA, pfSense та Security Onion, успішно імплементовано ключові принципи NIST SP 800-207: явну верифікацію ідентичності (RBAC/HBAC), мікросегментацію мережі та політику заборони за замовчуванням (Default Deny).

Результати порівняльних експериментів підтвердили фундаментальні переваги розробленої системи: на відміну від традиційної моделі, де атаки типу lateral movement були успішними, архітектура Zero Trust забезпечила 100% блокування несанкціонованого доступу до критичних активів на мережевому рівні. Система моніторингу продемонструвала здатність виявляти інциденти в режимі реального часу (MTTD < 1 с), забезпечуючи повну спостережуваність інфраструктури. Проведена економічна оцінка за методикою FAIR засвідчила високу ефективність запропонованого рішення, яке дозволяє мінімізувати потенційні фінансові збитки від кіберінцидентів без необхідності витрат на дорогі комерційні ліцензії.

ВИСНОВКИ

У кваліфікаційній роботі вирішено актуальне науково-прикладне завдання підвищення рівня захищеності корпоративної IT-інфраструктури шляхом розробки та впровадження прототипу системи захисту на основі архітектури нульової довіри (Zero Trust). В ході виконання роботи проведено аналіз сучасного ландшафту кіберзагроз, який підтвердив критичну вразливість традиційної «периметрової» моделі безпеки та довів, що відсутність інспекції внутрішнього трафіку і надмірна довіра до авторизованих користувачів створюють умови для безперешкодного горизонтального переміщення зловмисників. Це обґрунтувало доцільність переходу до парадигми Zero Trust, що базується на принципах «ніколи не довіряй, завжди перевіряй», найменших привілеїв та припущення компрометації.

На основі стандарту NIST SP 800-207 спроєктовано комплексну архітектуру захисту, що включає мережеву топологію з використанням мікросегментації для зменшення радіусу ураження, рольову та контекстну моделі доступу, а також політики фільтрації трафіку за принципом заборони за замовчуванням. Для практичної реалізації було обґрунтовано та розгорнуто програмний стек на базі технологій з відкритим кодом, де FreeIPA забезпечив централізоване управління ідентичністю та сильну автентифікацію, pfSense реалізував жорсткий контроль міжсегментного трафіку, а Security Onion гарантував повну спостережуваність мережі та виявлення інцидентів.

Проведене порівняльне дослідження підтвердило ефективність розробленої системи: моделювання сценаріїв атак продемонструвало, що впроваджені механізми забезпечують 100% блокування спроб горизонтального переміщення та несанкціонованого доступу, а система моніторингу скорочує час виявлення інцидентів до декількох секунд. Економічна оцінка засвідчила високу рентабельність рішення за рахунок упередження фінансових збитків, а практичне значення полягає у створенні масштабованого прототипу, що забезпечує високий рівень безпеки для підприємств малого та середнього бізнесу без витрат на дорогі ліцензії.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Rose S., Borchert O., Mitchell S., Connelly S. Zero Trust Architecture. NIST Special Publication 800-207. Gaithersburg : National Institute of Standards and Technology, 2020. 59 p. URL: https://nvlpubs.nist.gov/nistpubs/specialpublications/NIST.SP.800-207.pdf
2. What Is Zero Trust Architecture? Why Every Business Needs A Zero Trust Strategy in 2025 [Електронний ресурс] // Keypasco. URL: https://www.keypasco.com/en/what-is-zero-trust-architecture-why-every-business-needs-a-zero-trust-strategy-in-2025/
3. NIST SP 800-207 [Електронний ресурс] // ISMS.online. URL: https://www.isms.online/nist/nist-sp-800-207/
4. Guide on Implementing Zero Trust in Your Organisation [Електронний ресурс] // ISMS.online. URL: https://www.isms.online/zero-trust/guide-on-implementing-zero-trust-in-your-organisation/
5. Gilman E., Barth D. Zero Trust Networks: Building Secure Systems in Untrusted Networks. Sebastopol : O'Reilly Media, 2017. 240 p.
6. Announcing The Forrester Wave™: Zero Trust Platforms, Q3 2025 [Електронний ресурс] // Forrester Blogs. URL: https://www.forrester.com/blogs/announcing-the-forrester-wave-zero-trust-platforms-q3-2025-choosing-a-platform-solution-for-your-zero-trust-journey/
7. Zero Trust Maturity Model. Version 2.0 [Електронний ресурс] // Cybersecurity and Infrastructure Security Agency (CISA). 2023. URL: https://www.cisa.gov/zero-trust-maturity-model
8. FreeIPA Documentation [Електронний ресурс] // FreeIPA. URL: https://www.freeipa.org/page/Documentation.html
9. pfSense® software Configuration Recipes [Електронний ресурс] // Netgate Documentation. URL: https://docs.netgate.com/pfsense/
10. Security Onion Documentation [Електронний ресурс] // Security Onion. URL: https://docs.securityonion.net

11. Neuman C. et al. The Kerberos Network Authentication Service (V5). RFC 4120. 2005. URL: https://datatracker.ietf.org/doc/html/rfc4120

12. Lateral Movement (TA0008) [Електронний ресурс] // MITRE ATT&CK. URL: https://attack.mitre.org/tactics/TA0008/

13. National Vulnerability Database (NVD) [Електронний ресурс] // NIST. URL: https://nvd.nist.gov/

14. SIEM Log Management Best Practices [Електронний ресурс] // Picus Security. URL: https://www.picussecurity.com/how-to-enhance-siem-log-management

15. GNS3 Documentation [Електронний ресурс] // GNS3. URL: https://docs.gns3.com

16. IEEE Standard for Local and Metropolitan Area Networks — Bridges and Bridged Networks. IEEE Std 802.1Q-2018. New York : IEEE, 2018. URL: https://standards.ieee.org/ieee/802.1Q/6844/

17. Suricata Rules [Електронний ресурс] // Suricata User Guide. URL: https://docs.suricata.io/en/latest/rules/intro.html

18. Sermersheim J. Lightweight Directory Access Protocol (LDAP): The Protocol. RFC 4511. 2006. URL: https://datatracker.ietf.org/doc/html/rfc4511

19. 2024 Data Breach Investigations Report [Електронний ресурс] // Verizon. URL: https://www.verizon.com/business/resources/reports/dbir/

20. Oracle VM VirtualBox User Guide [Електронний ресурс] // Oracle. URL: https://www.virtualbox.org/manual/

ДОДАТКИ

```
SW-ZeroTrust#show run
version 15.2
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
service compress-config
hostname SW-ZeroTrust
interface GigabitEthernet0/0
 description Trunk to pfSense em1
 switchport trunk allowed vlan 10,20,30,40
 switchport trunk encapsulation dot1q
 switchport mode trunk
 media-type rj45
 negotiation auto
!
interface GigabitEthernet0/1
 description FreeIPA Server
 switchport access vlan 10
 switchport mode access
 media-type rj45
 negotiation auto
 spanning-tree portfast edge
!
interface GigabitEthernet0/2
 description Ubuntu Client
 switchport access vlan 20
 switchport mode access
 media-type rj45
 negotiation auto
 spanning-tree portfast edge
!
interface GigabitEthernet0/3
 description Web Server
 switchport access vlan 40
 switchport mode access
 media-type rj45
 negotiation auto
 spanning-tree portfast edge
!
interface GigabitEthernet1/0
 description Database Server
 switchport access vlan 30
 switchport mode access
 media-type rj45
 negotiation auto
 spanning-tree portfast edge
!
interface GigabitEthernet1/1
 description Security Onion ens33 (Management)
 switchport access vlan 10
 switchport mode access
```

```
 media-type rj45
 negotiation auto
 spanning-tree portfast edge
!
interface GigabitEthernet1/2
 switchport access vlan 999
 switchport mode access
 media-type rj45
 negotiation auto
control-plane
monitor session 1 source interface Gi0/0
monitor session 1 destination interface Gi1/2
end
SW-ZeroTrust#
```

Rules (Drag to Change Order)

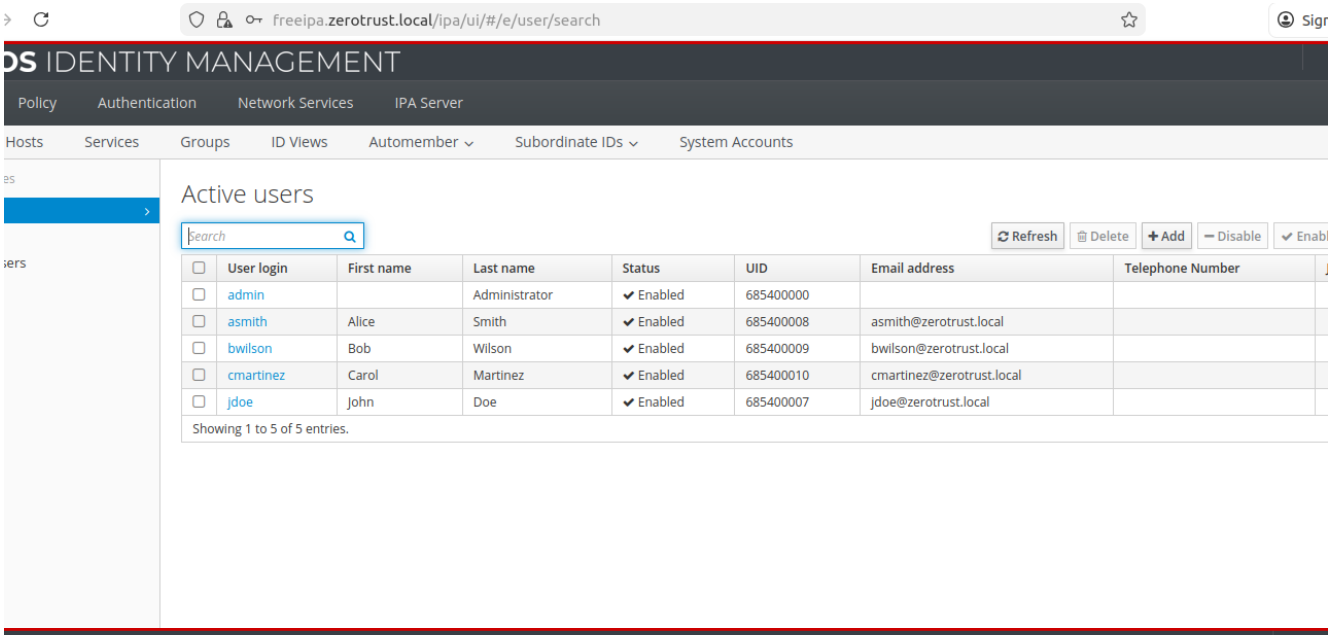
	States	Protocol	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description	Actions
✓	0/0 B	IPv4 TCP/UDP	192.168.30.10	*	192.168.10.10	88	*	none		DB_to_FreeIPA_Kerberos	
✓	0/0 B	IPv4 TCP	192.168.30.10	*	192.168.10.10	389 (LDAP)	*	none		DB_to_FreeIPA_LDAP	
✓	0/0 B	IPv4 UDP	192.168.30.10	*	192.168.10.10	123 (NTP)	*	none		DB_to_FreeIPA_NTP	
✓	0/0 B	IPv4 TCP/UDP	192.168.30.10	*	192.168.10.1	53 (DNS)	*	none		DB_DNS_to_pfSense	
✓	0/0 B	IPv4 TCP/UDP	192.168.30.10	*	192.168.10.10	53 (DNS)	*	none		SERVERS_to_FreeIPA_DNS	
✓	0/0 B	IPv4 UDP	192.168.30.10	*	192.168.10.20	514 (Syslog)	*	none		DB_Syslog_to_SecurityOnion	
✓	0/0 B	IPv4 TCP/UDP	SERVERS net	*	192.168.10.10	464	*	none		SERVERS_to_FreeIPA_kpasswd_464	
✓	0/0 B	IPv4 TCP	SERVERS net	*	192.168.10.10	443 (HTTPS)	*	none		SERVERS_to_FreeIPA_HTTPS	
✗	0/0 B	IPv4 *	SERVERS net	*	192.168.20.0/24	*	*	none		BLOCK_SERVERS_to_USERS_Lateral_Movement	
✗	0/0 B	IPv4 *	SERVERS net	*	*	*	*	none		SERVERS_Default_Deny	

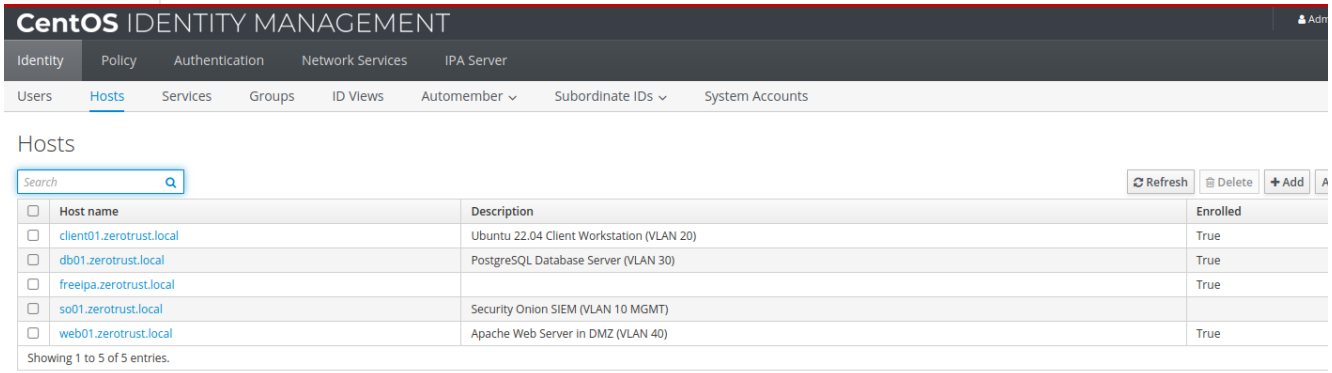
Floating WAN LAN USERS SERVERS DMZ MANAGEMENT

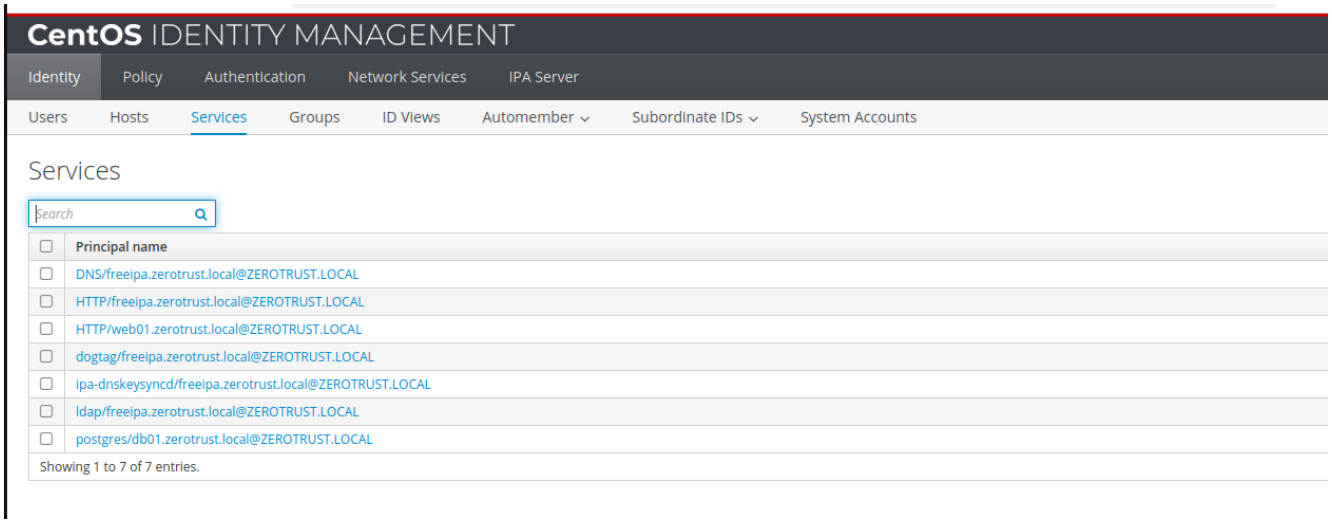
Rules (Drag to Change Order)

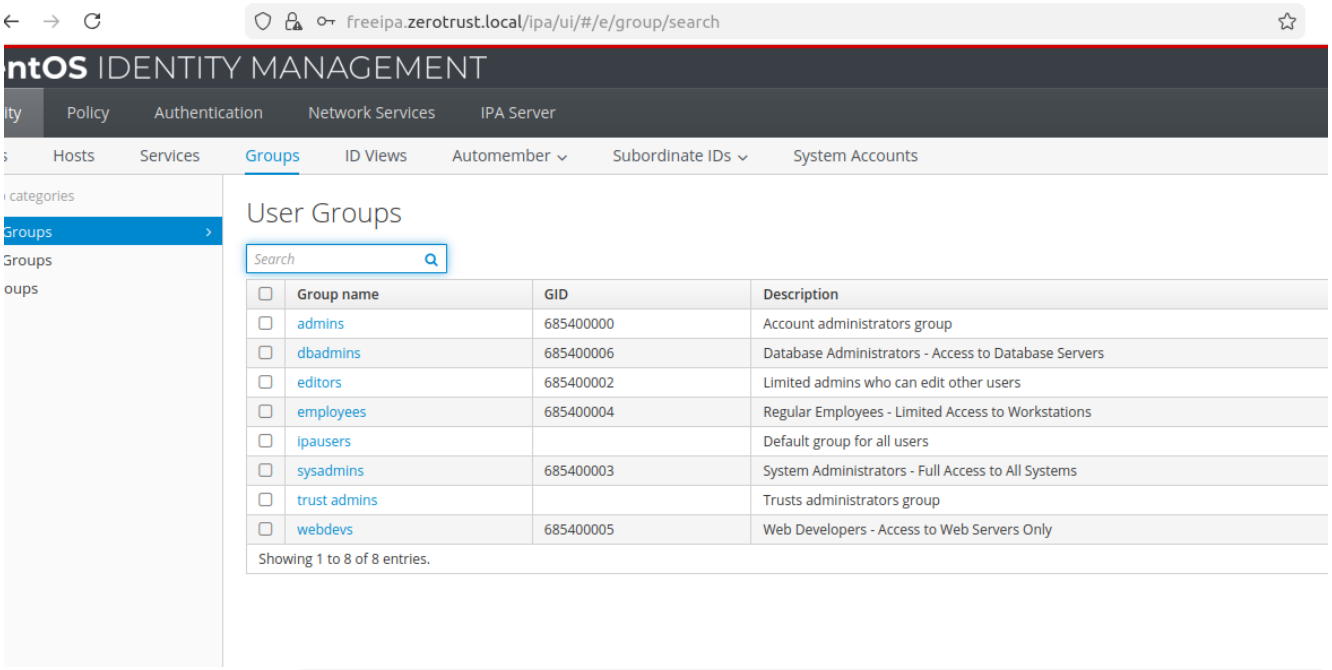
	States	Protocol	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description	Actions
✓	0/0 B	IPv4 TCP	192.168.40.10	*	192.168.30.10	5432	*	none		WEB_to_DB_PostgreSQL	
✓	0/0 B	IPv4 TCP/UDP	192.168.40.10	*	192.168.10.10	88	*	none		WEB_to_FreeIPA_Kerberos_SSO	
✓	0/0 B	IPv4 TCP	192.168.40.10	*	192.168.10.10	389 (LDAP)	*	none		WEB_to_FreeIPA_LDAP	
✓	0/0 B	IPv4 UDP	192.168.40.10	*	192.168.10.10	123 (NTP)	*	none		WEB_to_FreeIPA_NTP	
✓	0/0 B	IPv4 TCP/UDP	192.168.40.10	*	192.168.10.1	53 (DNS)	*	none		WEB_DNS_to_pfSense	
✓	0/0 B	IPv4 TCP/UDP	192.168.40.10	*	192.168.10.10	53 (DNS)	*	none		WEB_DNS_to_pfSense	
✓	0/0 B	IPv4 UDP	192.168.40.10	*	192.168.10.20	514 (Syslog)	*	none		WEB_Syslog_to_SecurityOnion	
✓	0/0 B	IPv4 TCP	192.168.40.10	*	*	80 (HTTP)	*	none		WEB_Internet_Updates_HTT	
✓	0/0 B	IPv4 TCP	192.168.40.10	*	*	443 (HTTPS)	*	none		WEB_Internet_Updates_HTTPS	
✓	0/0 B	IPv4 TCP/UDP	192.168.40.10	*	192.168.10.10	464	*	none		WEB_to_FreeIPA_kpasswd_464	
✗	0/0 B	IPv4 *	DMZ net	*	*	*	*	none		DMZ_Default_Deny	

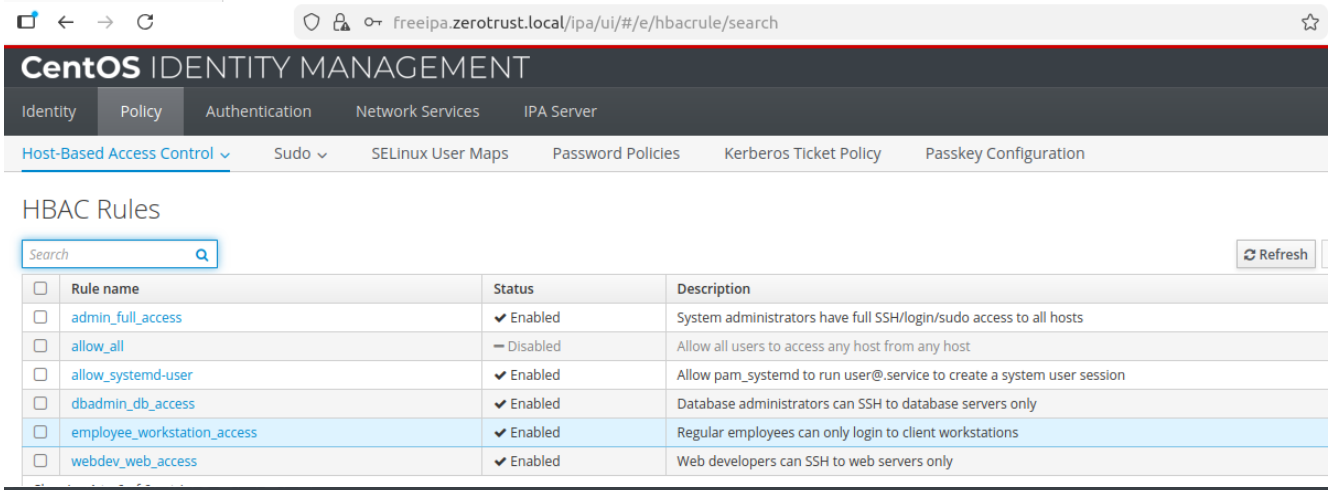
✓	0/0 B	IPv4 ICMP any	MANAGEMENT net	*	DMZ net	*	*	none	Admin access: MGMT → DMZ ping	
✓	0/0 B	IPv4 TCP/UDP	MANAGEMENT net	*	DMZ net	22 (SSH)	*	none	Admin access: MGMT → DMZ SSH	
✓	0/0 B	IPv4 TCP	MANAGEMENT net	*	MANAGEMENT address	80 (HTTP)	*	none	Anti-Lockout - WebGUI access	
✓	0/0 B	IPv4 UDP	192.168.10.10	*	*	123 (NTP)	*	none	FreeIPA_NTP_Sync_External	
✓	0/0 B	IPv4 TCP/UDP	192.168.10.10	*	*	53 (DNS)	*	none	FreeIPA_DNS_External	
✓	0/0 B	IPv4 TCP	192.168.10.20	*	*	80 (HTTP)	*	none	SecurityOnion_Updates_HTTP	
✓	0/0 B	IPv4 TCP	192.168.10.20	*	*	443 (HTTPS)	*	none	SecurityOnion_Updates_HTTPS	
✓	0/0 B	IPv4 UDP	192.168.10.20	*	*	123 (NTP)	*	none	SecurityOnion_NTP_Sync	
✗	0/0 B	IPv4 TCP	DMZ net	*	*	*	*	none	MGMT_Default_Deny	
✓	0/0 B	IPv4 ICMP any	MANAGEMENT net	*	MANAGEMENT address	*	*	none	Allow_Ping_to_Gateway	
✓	0/0 B	IPv4 TCP	USERS net	*	192.168.10.10	88	*	none	USERS_to_FreeIPA_Kerberos	
✓	0/0 B	IPv4 TCP	USERS net	*	192.168.10.10	389 (LDAP)	*	none	USERS_to_FreeIPA_LDAP	
✓	0/0 B	IPv4 UDP	USERS net	*	192.168.10.10	123 (NTP)	*	none	USERS_to_FreeIPA_NTP	
✓	0/0 B	IPv4 TCP	USERS net	*	192.168.40.10	80 (HTTP)	*	none	USERS_to_WEB_HTTP	
✓	0/0 B	IPv4 TCP	USERS net	*	192.168.40.10	443 (HTTPS)	*	none	USERS_to_WEB_HTTP	
✓	0/0 B	IPv4 TCP/UDP	USERS net	*	192.168.10.1	53 (DNS)	*	none	USERS_DNS_to_pfSense	
✓	0/0 B	IPv4 TCP/UDP	USERS net	*	192.168.10.10	53 (DNS)	*	none	USERS_DNS_to_FreeIPA	
✓	0/0 B	IPv4 UDP	USERS net	*	192.168.10.20	514 (Syslog)	*	none	USERS_Syslog_to_SecurityOnion	
✓	0/0 B	IPv4 IGMP	USERS net	*	*	*	*	none	USERS_ICMP_Ping	
✓	0/0 B	IPv4 TCP/UDP	USERS net	*	*	80 (HTTP)	*	none	USERS_Access_To_Internet	
✓	0/0 B	IPv4 TCP/UDP	USERS net	*	*	443 (HTTPS)	*	none	USERS_Access_To_Internet	
✓	0/0 B	IPv4 UDP	USERS net	*	192.168.10.10	88	*	none	USERS_to_FreeIPA_Kerberos_UDP	
✓	0/0 B	IPv4 TCP/UDP	USERS net	*	192.168.10.10	464	*	none	USERS_to_FreeIPA_kpasswd_464	
✓	0/0 B	IPv4 TCP	USERS net	*	192.168.10.10	443 (HTTPS)	*	none	USERS_to_FreeIPA_HTTPS_Enrollment	
✗	0/0 B	IPv4 *	USERS net	*	192.168.30.10	*	*	none	BLOCK_USERS_to_DB_Least_Privilege	
✗	0/0 B	IPv4 *	USERS net	*	*	*	*	none	USERS_Default_Deny	

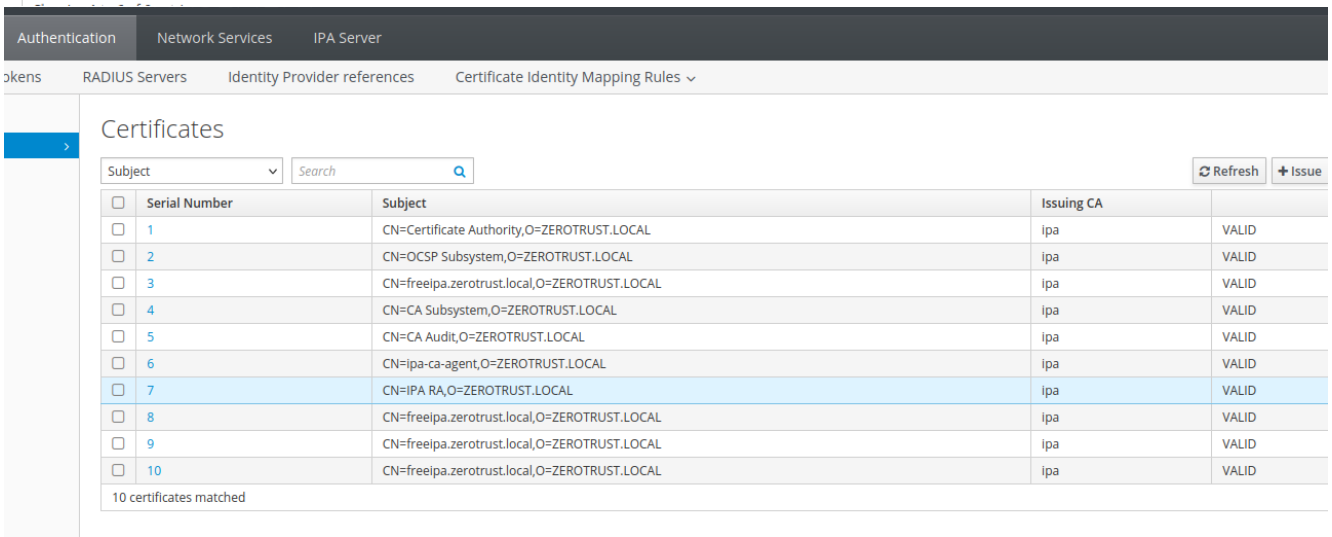

freeipa.zerotrust.local/ipa/ui/#/e/user/search
OS IDENTITY MANAGEMENT
Policy
Authentication
Network Services
IPA Server
Hosts
Services
Groups
ID Views
Automember
Subordinate IDs
System Accounts
Active users
Search
Refresh
Delete
Add
Disable
User login
First name
Last name
Status
UID
Email address
Telephone Number
admin
Administrator
Enabled
685400000
asmith
Alice
Smith
685400008
asmith@zerotrust.local
bwilson
Bob
Wilson
685400009
bwilson@zerotrust.local
cmartinez
Carol
Martinez
685400010
cmartinez@zerotrust.local
jdoe
John
Doe
685400007
jdoe@zerotrust.local
Showing 1 to 5 of 5 entries.


CentOS IDENTITY MANAGEMENT
Identity
Policy
Authentication
Network Services
IPA Server
Users
Hosts
Services
Groups
ID Views
Automember
Subordinate IDs
System Accounts
Hosts
Search
Refresh
Delete
Add
Host name
Description
Enrolled
client01.zerotrust.local
Ubuntu 22.04 Client Workstation (VLAN 20)
True
db01.zerotrust.local
PostgreSQL Database Server (VLAN 30)
True
freeipa.zerotrust.local
True
so01.zerotrust.local
Security Onion SIEM (VLAN 10 MGMT)
web01.zerotrust.local
Apache Web Server in DMZ (VLAN 40)
True
Showing 1 to 5 of 5 entries.


CentOS IDENTITY MANAGEMENT
Identity
Policy
Authentication
Network Services
IPA Server
Users
Hosts
Services
Groups
ID Views
Automember
Subordinate IDs
System Accounts
Services
Search
Principal name
DNS/freeipa.zerotrust.local@ZEROTRUST.LOCAL
HTTP/freeipa.zerotrust.local@ZEROTRUST.LOCAL
HTTP/web01.zerotrust.local@ZEROTRUST.LOCAL
dogtag/freeipa.zerotrust.local@ZEROTRUST.LOCAL
ipa-dnskeysyncd/freeipa.zerotrust.local@ZEROTRUST.LOCAL
ldap/freeipa.zerotrust.local@ZEROTRUST.LOCAL
postgres/db01.zerotrust.local@ZEROTRUST.LOCAL
Showing 1 to 7 of 7 entries.


freeipa.zerotrust.local/ipa/ui/#/e/group/search
ntOS IDENTITY MANAGEMENT
Policy
Authentication
Network Services
IPA Server
Hosts
Services
Groups
ID Views
Automember
Subordinate IDs
System Accounts
User Groups
Search
Group name
GID
Description
admins
685400000
Account administrators group
dbadmins
685400006
Database Administrators - Access to Database Servers
editors
685400002
Limited admins who can edit other users
employees
685400004
Regular Employees - Limited Access to Workstations
ipausers
Default group for all users
sysadmins
685400003
System Administrators - Full Access to All Systems
trust admins
Trusts administrators group
webdevs
685400005
Web Developers - Access to Web Servers Only
Showing 1 to 8 of 8 entries.


freeipa.zerotrust.local/ipa/ui/#/e/hbacrule/search
CentOS IDENTITY MANAGEMENT
Identity
Policy
Authentication
Network Services
IPA Server
Host-Based Access Control
Sudo
SELinux User Maps
Password Policies
Kerberos Ticket Policy
Passkey Configuration
HBAC Rules
Search
Refresh
Rule name
Status
Description
admin_full_access
Enabled
System administrators have full SSH/login/sudo access to all hosts
allow_all
Disabled
Allow all users to access any host from any host
allow_systemd-user
Enabled
Allow pam_systemd to run user@.service to create a system user session
dbadmin_db_access
Enabled
Database administrators can SSH to database servers only
employee_workstation_access
Enabled
Regular employees can only login to client workstations
webdev_web_access
Enabled
Web developers can SSH to web servers only


Authentication
Network Services
IPA Server
RADIUS Servers
Identity Provider references
Certificate Identity Mapping Rules
Certificates
Subject
Search
Refresh
Issue
Serial Number
Subject
Issuing CA
1
CN=Certificate Authority,O=ZEROTRUST.LOCAL
ipa
VALID
2
CN=OCSP Subsystem,O=ZEROTRUST.LOCAL
ipa
VALID
3
CN=freeipa.zerotrust.local,O=ZEROTRUST.LOCAL
ipa
VALID
4
CN=CA Subsystem,O=ZEROTRUST.LOCAL
ipa
VALID
5
CN=CA Audit,O=ZEROTRUST.LOCAL
ipa
VALID
6
CN=ipa-ca-agent,O=ZEROTRUST.LOCAL
ipa
VALID
7
CN=IPA RA,O=ZEROTRUST.LOCAL
ipa
VALID
8
CN=freeipa.zerotrust.local,O=ZEROTRUST.LOCAL
ipa
VALID
9
CN=freeipa.zerotrust.local,O=ZEROTRUST.LOCAL
ipa
VALID
10
CN=freeipa.zerotrust.local,O=ZEROTRUST.LOCAL
ipa
VALID
10 certificates matched