Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії
(повна назва факультету)
Кафедра комп'ютерних наук
(повна назва кафедри)

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

бакалавр
(назва освітнього ступеня)

на тему: Аналіз концепції та практична реалізація програмно-визначеної мережі (SDN)

Виконав: студент IV курсу, групи СН-42
спеціальності 122 Комп’ютерні науки
(шифр і назва спеціальності)

_____ Ліщенюк А. С.
(підпис) (прізвище та ініціали)

Керівник _____ Гром'як Р.С.
(підпис) (прізвище та ініціали)

Нормоконтроль _____ Шимчук Г.В.
(підпис) (прізвище та ініціали)

Завідувач кафедри _____ Боднарчук І.О.
(підпис) (прізвище та ініціали)

Рецензент _____ Тимощук Д.І.
(підпис) (прізвище та ініціали)

Тернопіль
2025

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії
(повна назва факультету)

Кафедра комп'ютерних наук
(повна назва кафедри)

ЗАТВЕРДЖУЮ
Завідувач кафедри
_____ Боднарчук І.О.
(підпис) (прізвище та ініціали)
«___» _____ 2025 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

на здобуття освітнього ступеня Бакалавр
(назва освітнього ступеня)

за спеціальністю 122 Комп'ютерні науки
(шифр і назва спеціальності)

Студенту Ліщенюку Артуру Сергійовичу
(прізвище, ім'я, по батькові)

1. Тема роботи Аналіз концепції та практична реалізація програмно-визначеної мережі (SDN)

Керівник роботи Гром'як Роман Сильвестрович, к.т.н., доцент кафедри КН
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом ректора від «*07*» *травня* *2025* року № *4/7-444*

2. Термін подання студентом завершеної роботи 24 червня 2025р.

3. Вихідні дані до роботи Літературні та інтернет джерела інформації про технології SDN. Документація по MikroTik CHR, Windows Server 2022, ZeroTier, Hyper-V та Ubuntu Linux.

4. Зміст роботи (перелік питань, які потрібно розробити)
Вступ
1) Огляд технології програмно-визначених мереж
2) Побудова середовища тестування SDN-мережі на основі ZeroTier
3) Налаштування та тестування SDN-мережі на основі ZeroTier
4) Безпека життєдіяльності, основи охорони праці
Висновки

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, слайдів)
1. Титульна сторінка. 2. Актуальність дослідження. 3. Мета, Об'єкт, Предмет дослідження. 4. Завдання дослідження. 5. Схема під'єднання елементів системи. 6. Характеристика мікрокомп'ютера та мікроконтролера. 7. Характеристика давачів системи. 8. Характеристика давачів системи (продовження). 9. Вибір мови програмування та налаштування Arduino. 10. Налаштування MQTT та створення Telegram-bot на Raspberry Pi Zero 2 W. 11. Розробка корпусу для монтажу та вставлення системи. 12. Перевірка роботи системи. 13. Висновки. 14. Завершальний.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Безпека життєдіяльності, основи охорони праці	Окіпний Ігор Богданович, к.т.н. зав. кафедри МТ	12.06.2024	14.06.2024

7. Дата видачі завдання 29 січня 2024 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1.	Ознайомлення з завданням до кваліфікаційної роботи	30.01.2025	*Виконано*
2.	Підбір джерел по темі кваліфікаційної роботи	31.01.2025-03.02.2025	*Виконано*
3.	Опрацювання джерел по темі кваліфікаційної роботи	04.02.2025-06.02.2025	*Виконано*
4.	Виконання дослідження щодо технології SDN	07.02.2025-11.02.2025	*Виконано*
5.	Оформлення розділу «Огляд технології програмно-	03.06.2025-05.06.2025	*Виконано*
	визначених мереж»		
6.	Оформлення розділу «Побудова середовища	06.06.2025-08.06.2025	*Виконано*
	тестування SDN-мережі на основі ZeroTier»		
6.	Оформлення розділу «Налаштування та тестування	09.06.2025-11.06.2025	*Виконано*
	SDN-мережі на основі ZeroTier»		
7.	Виконання завдання до розділу «Безпека життєдіяльності»	12.06.2025-13.06.2025	*Виконано*
8.	Виконання завдання до підрозділу «Основи охорони	14.06.2025-15.06.2025	*Виконано*
	праці»		
9.	Оформлення кваліфікаційної роботи	16.06.2025-17.06.2025	*Виконано*
10.	Нормоконтроль	18.06.2025-19.06.2025	*Виконано*
11.	Перевірка на плагіат	20.06.2025	*Виконано*
12.	Попередній захист кваліфікаційної роботи	21.06.2025	*Виконано*
13.	Захист кваліфікаційної роботи	30.06.2025	

Студент _____ Ліщенюк А. С.
(підпис) (прізвище та ініціали)

Керівник роботи _____ Гром'як Р.С.
(підпис) (прізвище та ініціали)

АНОТАЦІЯ

Аналіз концепції та практична реалізація програмно-визначеної мережі (SDN) // Кваліфікаційна робота освітнього рівня «Бакалавр» // Ліщенюк Артур Сергійович // Тернопільський національний технічний університет імені Івана Пулюя, факультет комп'ютерно-інформаційних систем і програмної інженерії, кафедра комп'ютерних наук, група СН-42 // Тернопіль, 2025 // С.73, рис. – 32, табл. – 0, кресл. – 14, додат. – 2, бібліогр. – 35.

Ключові слова: SDN, ZeroTier, MikroTik CHR, overlay-мережа, віртуалізація, Hyper-V.

У кваліфікаційній роботі бакалавра досліджено концепцію програмно-визначених мереж і продемонстровано можливість побудови захищеної SDN на основі платформи ZeroTier у середовищі MikroTik CHR. У теоретичній частині проаналізовано еволюцію традиційних мережевих архітектур, обґрунтовано доцільність розділення площини керування та площини передачі й висвітлено механізми southbound- і northbound-API, що забезпечують централізоване програмне управління. Практичний етап включав розгортання середовища тестування в гіпервізорі Hyper-V із MikroTik CHR та контейнерним ZeroTier-агентом, сервером Windows Server 2022 RDS і клієнтською системою Ubuntu Linux. Створено приватну overlay-мережу, налаштовано ACL та статичні маршрути, здійснено криптографічну авторизацію вузлів і перевірено маршрутизацію трафіку до сегменту DMZ. Результати тестування показали стабільні затримки на рівні 1–2 мс без втрат пакетів і успішне встановлення RDP-сеансу, що підтвердило коректність тунелювання та готовність рішення до експлуатації. Робота доводить, що поєднання ZeroTier і MikroTik CHR забезпечує гнучке, масштабоване й економічно доцільне розгортання ізольованих мережевих середовищ.

ANNOTATION

Concept analysis and practical implementation of a software-defined network (SDN) // Qualification work of the educational level "Bachelor" // Artur Lishcheniuk // Ternopil Ivan Pulyu National Technical University, Computer and Information Systems and Software Engineering Faculty, Computer Sciences Department, group SN-42 // Ternopil, 2025 // P. 73, fig. - 32, tabl. - 0, drawings - 14, annexes. – 2, references - 35.

Keywords: SDN, ZeroTier, MikroTik CHR, overlay network, virtualisation, Hyper-V..

The bachelor's thesis explores the concept of software-defined networks and demonstrates the possibility of building a secure SDN based on the ZeroTier platform in the MikroTik CHR environment. The theoretical part analyses the evolution of traditional network architectures, justifies the feasibility of separating the control and transmission planes, and highlights the southbound and northbound API mechanisms that provide centralised software management. The practical stage included the deployment of a testing environment in the Hyper-V hypervisor with MikroTik CHR and a containerised ZeroTier agent, Windows Server 2022 RDS server, and Ubuntu Linux client system. A private overlay network was created, ACLs and static routes were configured, nodes were cryptographically authorised, and traffic routing to the DMZ segment was tested. The test results showed stable delays of 1-2 ms with no packet loss and successful RDP session establishment, which confirmed the correctness of tunnelling and the solution's readiness for operation. This work proves that the combination of ZeroTier and MikroTik CHR provides a flexible, scalable and cost-effective deployment of isolated network environments.

ПЕРЕЛІК СКОРОЧЕНЬ

SDN (англ. Software-Defined Networking) – Програмно-визначена мережа.

VPN (англ. Virtual Private Network) – Віртуальна приватна мережа.

CHR (англ. Cloud Hosted Router) – Хмарний маршрутизатор.

DMZ (англ. Demilitarized Zone) – Демілітаризована зона.

IoT (англ. Internet of Things) – Інтернет речей.

API (англ. Application Programming Interface) – Інтерфейс прикладного програмування.

IDS (англ. Intrusion Detection System) – Система виявлення вторгнень.

IPS (англ. Intrusion Prevention System) – Система запобігання вторгненням.

RDP (англ. Remote Desktop Protocol) - Протокол віддаленого робочого столу.

QoS (англ. Quality of Service) – Якість обслуговування.

NAT (англ. Network Address Translation) – Трансляція мережевих адрес.

ACL (англ. Access Control List) – Список керування доступом.

VL1 (англ. Virtual Layer 1) – Віртуальний рівень 1.

VL2 (англ. Virtual Layer 2) – Віртуальний рівень 2.

RDS (англ. Remote Desktop Services) – Служби віддаленого робочого столу.

UDP (англ. User Datagram Protocol) – Протокол датаграм користувача.

RDP (англ. Remote Desktop Protocol) – Протокол віддаленого робочого столу.

ЗМІСТ

ВСТУП

Актуальність теми. Мережеві інфраструктури стають дедалі складнішими, вимагаючи гнучкості, масштабованості та централізованого управління. Традиційні мережі, засновані на тісній інтеграції програмного забезпечення з апаратним забезпеченням, часто не відповідають вимогам динамічних та розподілених систем. У цьому контексті технологія SDN стає основним рішенням для забезпечення адаптивності, централізованого контролю та підвищеної безпеки. З-поміж багатьох реалізацій SDN, особливої уваги заслуговує платформа ZeroTier, яка поєднує в собі функції SDN, VPN та overlay-мережі, спрощуючи створення захищених та керованих віртуальних мереж. Практична реалізація SDN-рішень із використанням ZeroTier у середовищі MikroTik CHR є актуальною темою для дослідження, оскільки дозволяє інтегрувати інноваційні мережеві підходи у звичні інфраструктури підприємств.

Мета і задачі дослідження. Метою даної кваліфікаційної роботи є аналіз концепції SDN та реалізація прикладної SDN-мережі на основі платформи ZeroTier та MikroTik CHR.

Для досягнення цієї мети було поставлено наступні задачі:

– дослідити архітектуру, принципи роботи та класифікацію технологій SDN;

– проаналізувати особливості роботи та безпеки платформи ZeroTier як представника SDN-рішень;

– розробити лабораторне середовище з використанням MikroTik CHR, контейнера ZeroTier та клієнтських систем на базі Ubuntu Linux та Windows Server 2022;

– реалізувати захищений доступ до внутрішнього ресурсу в DMZ через SDN-мережу ZeroTier;

– провести тестування працездатності побудованої мережі.

Об’єкт дослідження. Процеси побудови та управління програмно-визначеними мережами у сучасних IT-інфраструктурах.

Предмет дослідження. Архітектура та практичні аспекти реалізації SDN-мережі з використанням ZeroTier у середовищі MikroTik CHR та клієнтській операційній системі Linux.

Практичне значення одержаних результатів. Результати дослідження можуть бути використані для впровадження захищених SDN-рішень у малих та середніх організаціях, з метою підвищення гнучкості управління мережею, спрощення розгортання віртуальних приватних мереж, а також забезпечення безпечного доступу до ресурсів, розміщених у DMZ-сегментах. Запропонована методика може бути застосована у навчальних лабораторіях та IT-підрозділах підприємств для швидкого створення ізольованих мережевих середовищ.

РОЗДІЛ 1. ОГЛЯД ТЕХНОЛОГІЇ ПРОГРАМНО-ВИЗНАЧЕНИХ МЕРЕЖ

1.1 Еволюція традиційних мережевих архітектур

Упродовж останніх десятиліть комп'ютерні мережі стали невід'ємною складовою функціонування організацій різних галузей [1]. З розвитком інформаційних технологій та збільшенням вимог до швидкості та безпеки з'явилася необхідність у принципово нових підходах до побудови та управління мережевими інфраструктурами. Перші архітектурні рішення у сфері мережевих технологій базувалися на жорстко структурованій та статичній моделі, яка мала низку суттєвих обмежень [2].

Традиційні мережеві архітектури сформувалися на основі підходу, при якому кожен пристрій в мережі - будь то маршрутизатор, комутатор або шлюз поєднував функції передачі даних та прийняття рішень щодо маршрутизації. Кожен пристрій мав локально збережену конфігурацію, виконував обробку трафіку на основі закладених правил і не мав глобального уявлення про всю мережу. Це означало, що програмне забезпечення для керування мережею було інтегроване в апаратне забезпечення, і зміна логіки роботи мережі вимагала або ручного втручання, або заміни обладнання. Такий підхід був прийнятним у часи, коли мережі були простими, статичними та включали обмежену кількість вузлів. Однак з плином часу ситуація змінилася. Кількість підключених пристроїв зросла в рази. З'явилися мобільні технології, хмарні сервіси, віртуалізація, IoT, що вимагають від мереж високої гнучкості, масштабованості та безперервного доступу. Класичні мережі, в яких управління розподілено між фізичними пристроями, почали втрачати ефективність. При кожній зміні в інфраструктурі (додавання нового вузла, впровадження політики доступу, балансування навантаження) адміністратори змушені були вручну змінювати конфігурації, часто без централізованого бачення та автоматизованого

контролю. Це підвищувало ризики помилок, знижувало надійність мережевих сервісів і збільшувало витрати на підтримку [3].

Однією з головних проблем класичних мереж є жорстка прив’язка програмного забезпечення до конкретного обладнання. Програмна логіка - це не окрема керуюча система, а частина прошивки пристрою, яку важко оновити або змінити без втручання в сам пристрій. Це призводить до відсутності гнучкості. Така структура унеможливлює швидке впровадження інновацій, створює залежність від виробника обладнання та гальмує автоматизацію.

Ще одна критична проблема - складність централізованого управління. У традиційній архітектурі кожен пристрій є незалежним елементом, з яким потрібно працювати окремо. Наприклад, щоб впровадити нову політику безпеки або змінити маршрут трафіку, адміністратор повинен підключитися до кожного відповідного пристрою, змінити конфігурацію вручну, перевірити синтаксис. У великих мережах це призводить до непрозорості управління, труднощів у виявленні причин несправностей і низької швидкості реагування на події.

Масштабованість також є слабким місцем класичної моделі. Зі збільшенням кількості підключених пристроїв та обсягів трафіку виникають нові виклики: мережа повинна вміти динамічно адаптувати маршрути, ефективно балансувати навантаження, забезпечувати ізоляцію трафіку для безпеки. У традиційних мережах реалізація таких функцій передбачає розгортання нових пристроїв, складну конфігурацію та ретельне тестування. Усе це потребує часу, ресурсів та досвіду персоналу.

Іще одним важливим недоліком є фрагментованість політик безпеки. Кожен пристрій виконує локальну фільтрацію трафіку, тому загальні політики важко синхронізувати між усіма вузлами. У випадку атаки або вторгнення, виявлення джерела загрози займає багато часу через відсутність централізованого журналювання і моніторингу. Це особливо критично в умовах сучасних кібератак, що вимагають оперативної реакції та постійного аналізу подій у режимі реального часу.

У зв’язку з цим усе більшого поширення набувають нові підходи до побудови мереж, які відокремлюють логіку керування від фізичної інфраструктури. Програмно-визначені мережі є відповіддю на обмеження традиційних моделей. Вони дозволяють централізовано керувати мережею через спеціалізоване програмне забезпечення, що забезпечує гнучкість, швидкість розгортання, автоматизацію процесів, спрощене масштабування та підвищену безпеку. Таким чином, перехід від класичних мереж до SDN є логічним кроком у напрямку модернізації IT-інфраструктури відповідно до викликів сучасного цифрового середовища [4].

1.2 Основні поняття та концепції SDN

Програмно-визначені мережі - це сучасний метод побудови та управління мережами, що докорінно змінює традиційний підхід до мережевої архітектури. Головна концепція SDN полягає у відокремленні площини керування від площини передачі даних [5]. У традиційних мережах ці функції поєднані в одному фізичному пристрої, наприклад, у маршрутизаторі чи комутаторі, який одночасно приймає рішення про маршрутизацію та виконує пересилання трафіку. У SDN ці функції поділяються: пристрої, що відповідають за пересилання пакетів, підпорядковуються централізованому контролеру, який приймає всі логічні рішення щодо маршрутизації, політик доступу, фільтрації та інших аспектів мережевого функціонування.

Основною особливістю SDN є логічна централізація управління мережею. Контролер отримує повну інформацію про поточний стан мережі, її топологію, трафік, активність користувачів, і на основі цих даних формує відповідні правила, які надсилаються до мережевих пристроїв [6]. Це дозволяє забезпечити глобальне бачення мережі в реальному часі та швидко реагувати на зміни умов або появу загроз. Усі мережеві налаштування в SDN визначаються програмно, через стандартизовані або відкриті інтерфейси, що спрощує автоматизацію та інтеграцію з іншими IT-сервісами [7].

Концепція SDN підтримує масштабованість конфігурації, що особливо важливо в умовах сучасних динамічних середовищ, таких як дата-центри, хмарні платформи чи корпоративні мережі з численними філіями. Завдяки програмному управлінню, мережа стає адаптивною та її поведінку можна змінювати в реальному часі без фізичного втручання. Наприклад, у разі виявлення перевантаження на певному вузлі, система може автоматично перенаправити трафік через альтернативні маршрути або обмежити доступ до деяких сегментів для певних категорій користувачів.

Ще одним ключовим поняттям SDN є відкритість і стандартизація. Інтерфейси взаємодії між контролером і мережевим обладнанням (так звані southbound API) та між контролером і мережевими додатками (northbound API) дозволяють створювати масштабовані, взаємозамінні та легко модернізовані рішення.

На рисунку 1.1 показано архітектуру SDN.

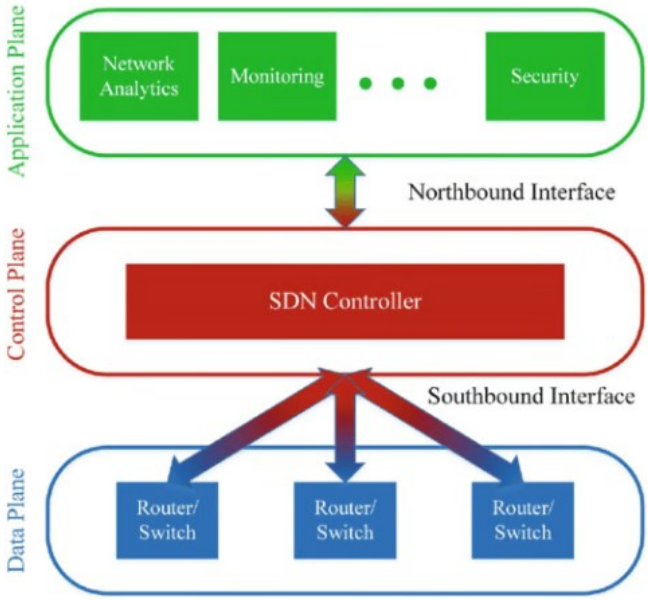

Рисунок 1.1 – Архітектура SDN

Площина передачі (Data Plane) є одним із ключових компонентів архітектури SDN та виконує основну функцію фізичної обробки і пересилання мережевого трафіку [8]. У традиційній мережевій архітектурі ця площина

поєднана з логікою керування безпосередньо в одному пристрої, що ускладнює централізоване адміністрування. У SDN підході площина передачі повністю відокремлена від площини керування, що дозволяє зосередити усю логіку управління мережею в централізованому контролері, а пристрої, розміщені у площині передачі, виконують лише інструкції, які надходять з контролера.

Фізично площина передачі складається з мережевих пристроїв нижнього рівня, таких як комутатори, маршрутизатори, точки доступу, або програмно-реалізованих компонентів, які працюють у віртуальних середовищах - наприклад, віртуальних комутаторів у гіпервізорах. Усі ці пристрої відповідають за прийом, обробку та перенаправлення мережевих пакетів згідно з таблицями маршрутизації чи фільтрації, які створює SDN-контролер.

На відміну від традиційних пристроїв, які автономно обчислюють маршрути та застосовують політики доступу, елементи площини передачі в SDN виконують лише ті дії, які були чітко задані. Вони не мають власної "інтелектуальної" логіки - їхнє завдання полягає у швидкому та ефективному виконанні вже сформульованих інструкцій. Наприклад, коли пакет даних надходить до комутатора, пристрій перевіряє його відповідно до таблиці потоків (flow table), яка була передана контролером, і приймає рішення: пропустити, змінити, пріоритезувати або відкинути пакет. Площина передачі тісно взаємодіє з контролером за допомогою southbound-протоколів, серед яких найпоширенішим є OpenFlow. Цей протокол дозволяє контролеру оновлювати таблиці потоків у пристроях, а також отримувати зворотну інформацію про стан трафіку. Така взаємодія забезпечує гнучкість та адаптивність, оскільки контролер може змінювати поведінку пристроїв площини передачі в реальному часі - наприклад, у відповідь на появу нових додатків, зміну політики безпеки або виявлення аномальної активності.

Ще однією важливою характеристикою площини передачі є її продуктивність і оптимізація. Оскільки пристрої цієї площини не витрачають ресурси на прийняття рішень, а лише виконують команди, вони можуть працювати максимально ефективно з точки зору швидкості обробки трафіку. У

віртуалізованих середовищах, де використовуються програмні комутатори, таких як Open vSwitch (OVS), площина передачі також може бути оптимізована для взаємодії з гіпервізорами та хмарними сервісами, що значно підвищує гнучкість мережевої інфраструктури. Хоча площина передачі не приймає автономних рішень, вона повинна мати здатність до базової обробки пакетів у разі втрати зв'язку з контролером. У деяких реалізаціях передбачено буферизацію пакетів, тимчасове збереження інструкцій або застосування локальних правил failover, щоб гарантувати безперервність обслуговування.

Таким чином, площина передачі у SDN виконує фундаментальну функцію доставки даних і є тим рівнем, на якому фізично реалізується логіка, сформована контролером. Її відокремлення від площини керування дозволяє мережі стати гнучкішою, більш автоматизованою і масштабованою, оскільки внесення змін до поведінки мережі більше не потребує втручання у фізичну інфраструктуру. У поєднанні з централізованим управлінням, це забезпечує новий рівень ефективності, безпеки та адаптивності сучасних мереж.

Площина керування (Control Plane) є центральним елементом архітектури SDN, відповідальним за прийняття логічних рішень щодо маршрутизації, комутації, управління трафіком, реалізації політик безпеки, балансування навантаження та інших аспектів функціонування мережі. На відміну від традиційних мережевих архітектур, де функції керування розподілені між усіма пристроями і реалізовані в їхньому вбудованому програмному забезпеченні, SDN винесла логіку керування в окремий централізований програмний компонент - SDN-контролер. Контролер - це програмний модуль, який виступає «мозком» мережі. Він має повну інформацію про мережеву топологію, стан каналів, активність користувачів, трафік, підключені пристрої та служби. На основі цієї інформації він приймає рішення про те, як повинен оброблятися трафік, і передає ці інструкції на пристрої у площині передачі за допомогою спеціалізованих протоколів, найпоширенішим серед яких є OpenFlow.

Площина керування функціонує на основі двосторонньої взаємодії. З одного боку, вона постійно отримує телеметричні дані з рівня передачі даних,

що дозволяє їй мати актуальну картину стану мережі в реальному часі. З іншого боку, вона приймає та виконує запити від прикладного рівня, який формує стратегічні політики і визначає, як має поводитися мережа за різних умов. Це реалізується за допомогою northbound API, що дозволяє додаткам надсилати вказівки до контролера у вигляді REST-запитів або через інші програмні інтерфейси. Контролер виконує функції централізованого менеджера потоків. Він формує так звані таблиці потоків (flow tables), де описано правила для обробки трафіку. Наприклад, пакет, що надходить на порт комутатора, повинен бути пересланий на інший порт, змінити заголовок, або бути відкинутий - відповідно до IP-адреси, VLAN-тегу, типу протоколу чи іншої інформації. Ці правила завантажуються у пристрої площини передачі, які самостійно виконують їх без необхідності додаткових запитів.

Ключова перевага площини керування полягає в можливості централізованого управління, що забезпечує єдину політику конфігурації для всієї мережі. Це дозволяє реалізовувати гнучкі сценарії де мережа може динамічно змінювати маршрути трафіку у разі перевантаження одного з каналів, або миттєво блокувати певний тип трафіку у відповідь на виявлення підозрілої активності. Крім того, централізоване управління значно спрощує автоматизацію. Через програмні інтерфейси адміністратор або система оркестрації може змінювати конфігурацію всієї мережі за кілька секунд.

Однак така централізація створює і певні виклики. Найбільш критичним є ризик єдиної точки відмови. Якщо SDN-контролер виходить з ладу або потрапляє під атаку, уся мережа може залишитися без керування. Для вирішення цієї проблеми в реальних реалізаціях застосовуються кластеризація контролерів, розподілені контролери, а також резервні канали зв’язку, які забезпечують високу доступність керування. Також важливим є питання безпеки контролера. Він має повний доступ до всіх компонентів мережі, його компрометація може мати катастрофічні наслідки. Тому критично важливо захищати канали зв’язку між контролером і пристроями, а також реалізовувати автентифікацію і шифрування. У практичних системах може

використовуватись один або кілька контролерів. Наприклад, у великих мережах часто впроваджують ієрархічну модель керування, де декілька локальних контролерів обслуговують певні домени, а глобальний контролер координує між ними політики на вищому рівні. Такий підхід дозволяє зберігати масштабованість без втрати керованості.

Завдяки можливості централізованого аналізу трафіку, гнучкому управлінню політиками та підтримці автоматизації, площина керування у SDN забезпечує основу для реалізації інтелектуальних, самокерованих і адаптивних мереж, які здатні відповідати на виклики сучасного цифрового середовища. Вона відкриває нові можливості для впровадження таких технологій, як мережевий моніторинг у реальному часі, контроль доступу на основі ролей, динамічне реагування на інциденти та інтеграція з хмарними оркестраторами.

Площина додатків (Application Plane) є найвищим рівнем у тришаровій архітектурі SDN і відповідає за визначення поведінки та політик роботи мережі. Цей рівень не взаємодіє з трафіком напряму, як це відбувається у площині передачі, і не відповідає за маршрутизацію чи обробку потоків, як це відбувається у площині керування. Натомість він реалізує бізнес-логіку, аналітику, автоматизацію, безпеку та інші мережеві сервіси, які визначають, як саме повинна функціонувати мережа відповідно до поставлених цілей. Основна роль площини додатків полягає у формуванні високорівневих правил та стратегій управління, які потім передаються до SDN-контролера. Ці правила можуть стосуватися контролю доступу, балансування навантаження, політик якості обслуговування, виявлення аномалій у трафіку, оркестрації хмарних ресурсів, моніторингу продуктивності, забезпечення відповідності стандартам безпеки тощо. Весь обсяг взаємодії між додатками і контролером здійснюється через northbound API - інтерфейс, що дозволяє додаткам взаємодіяти з контролером та отримувати необхідну інформацію про стан мережі.

Площина додатків складається з програмних рішень або модулів, які можуть бути, як розробленими спеціально під завдання організації, так і стандартними програмами, інтегрованими до SDN-контролера або зовнішніми.

До таких додатків можна віднести системи управління мережею (Network Management Systems), засоби реагування на інциденти, NGFW (фаєрволи нового покоління), IDS/IPS-системи [9-11], контролери доступу на основі ідентичностей користувачів, компоненти автоматизованого конфігураційного менеджменту, а також аналітичні платформи, що використовують дані трафіку для побудови звітів і прогнозів.

Завдяки площині додатків SDN-мережі можуть адаптуватися до змін у режимі реального часу. Наприклад, додаток моніторингу виявляє надмірне навантаження на певному сегменті мережі та надсилає запит до контролера з рекомендацією змінити маршрутизацію або застосувати нову політику QoS. Контролер приймає це рішення і розсилає оновлення до пристроїв у площині передачі. Або, у випадку виявлення підозрілої активності, система безпеки може динамічно оновити фільтри, миттєво блокуючи зловмисний трафік.

Перевагою такого підходу є можливість інтеграції мережевих процесів із загальною IT-інфраструктурою організації. Площина додатків дозволяє синхронізувати мережу з іншими сервісами. Наприклад з інструментами DevOps, хмарними оркестраторами, платформами контейнеризації (Kubernetes, Docker), системами ідентифікації (LDAP, Active Directory) та платформами моніторингу (Grafana, Zabbix). Завдяки цьому мережа перестає бути ізольованим елементом і перетворюється на гнучкий, керований ресурс, який динамічно підлаштовується під потреби користувачів і додатків. Ще однією важливою характеристикою площини додатків є автоматизоване прийняття рішень. У сучасних реалізаціях SDN активно використовуються засоби штучного інтелекту, машинного навчання та складної евристики для побудови самокерованих мереж (Self-Driving Networks). Додатки на цьому рівні можуть автоматично виявляти аномалії, оптимізувати маршрути, прогнозувати пік навантаження, регулювати пропускну здатність каналів залежно від критичності сервісів. Разом із тим, площина додатків створює вимоги до відкритості та стандартизації API, адже її ефективне функціонування можливе лише за наявності уніфікованого та передбачуваного способу взаємодії з

контролером. Розробники додатків повинні мати доступ до задокументованих API, мати змогу отримати дані про топологію, статус пристроїв, статистику трафіку тощо, а також надсилати вказівки, які будуть коректно оброблені контролером.

Площина додатків у SDN є тим рівнем, який втілює бізнес-логіку та операційні потреби в технічну конфігурацію мережі, перетворюючи складну інфраструктуру на керовану та адаптивну систему. Вона дозволяє забезпечити зв'язок між цілями підприємства, IT-сервісами та самою мережею, роблячи її не просто каналом для передачі даних, а інтелектуальним інструментом для підтримки цифрових процесів.

У архітектурі SDN інтерфейси API виконують важливу функцію взаємодії між рівнями: контролером, інфраструктурою та прикладними додатками. Вони поділяються на два основні типи: Southbound API (південний інтерфейс) та Northbound API (північний інтерфейс). Кожен з них має своє функціональне призначення, протоколи реалізації та вимоги до сумісності, без яких неможлива ефективна робота SDN-архітектури.

Southbound API забезпечує взаємодію між SDN-контролером та пристроями інфраструктурного рівня - комутаторами, маршрутизаторами, точками доступу, віртуальними мережевими елементами тощо [12]. Головне завдання цього інтерфейсу - це передача від контролера до пристроїв інструкцій з керування трафіком, які реалізуються у вигляді таблиць потоків, правил маршрутизації або фільтрації. У зворотному напрямку southbound API дозволяє пристроям передавати контролеру телеметричні дані про поточний стан, навантаження, статистику трафіку або події. Найбільш поширеним протоколом реалізації Southbound API є OpenFlow - стандарт, який дозволяє контролеру безпосередньо управляти таблицями потоків мережевих пристроїв. Завдяки OpenFlow контролер може детально визначати, як повинен оброблятися кожен мережевий пакет на основі його атрибутів: MAC/IP-адреси, порти, тип протоколу, VLAN-теги тощо. Протокол дозволяє контролеру створювати, оновлювати та видаляти правила, а також отримувати інформацію

про пакети, які не відповідають жодному з існуючих правил. Окрім OpenFlow, існують також інші southbound-протоколи, які використовуються в залежності від реалізації SDN-системи, зокрема NETCONF, OVSDB (Open vSwitch Database Management Protocol), BGP-LS, gRPC, RESTCONF, а також пропрієтарні рішення від виробників (наприклад, Cisco OpFlex). Кожен з них орієнтований на певний рівень деталізації та специфіку обладнання. Надійність і ефективність southbound API мають вирішальне значення для коректного функціонування SDN. Якщо інтерфейс не підтримує повну синхронізацію між контролером і пристроями, це може призвести до втрати керованості, неправильного пересилання трафіку або збоїв у безпеці. Тому в реальних реалізаціях часто використовують зворотні канали зв'язку, механізми повторної синхронізації та резервування.

Northbound API забезпечує комунікацію між SDN-контролером і додатками на рівні Application Plane [13]. Основне завдання northbound API - забезпечити можливість високорівневих додатків впливати на поведінку мережі, передаючи до контролера запити, політики та стратегічні рішення. Через northbound API прикладні сервіси можуть отримувати детальну інформацію про топологію мережі, статистику трафіку, стан пристроїв, а також можуть створювати або змінювати правила, керувати пріоритетами трафіку, ініціювати блокування підозрілих потоків, автоматизувати оркестрацію ресурсів тощо. На відміну від southbound API, який є більш стандартизованим, northbound API не має загальноприйнятого протоколу, тому зазвичай реалізується у вигляді RESTful API, GraphQL, gRPC або через специфічні SDK (software development kits), надані постачальниками SDN-рішень. Однією з переваг northbound API є гнучкість. Додатки можуть працювати незалежно від фізичної реалізації мережі, оскільки отримують абстрагований доступ до логіки керування. Це дозволяє створювати модифіковані сценарії, наприклад, автоматичне масштабування пропускної здатності під час пікових навантажень або динамічне розгортання мережевих сегментів відповідно до політик безпеки.

Водночас існує низка викликів, пов’язаних з реалізацією northbound API. По-перше, відсутність єдиного стандарту ускладнює сумісність між додатками та різними контролерами. По-друге, безпека інтерфейсу є критично важливою, адже помилкові або зловмисні запити з боку додатків можуть змінити поведінку всієї мережі. Тому сучасні рішення реалізовують автентифікацію, авторизацію, шифрування трафіку API-запитів і системи логування.

1.3 Висновок до першого розділу

У першому розділі було проведено огляд еволюції мережевих архітектур, що привів до появи концепції SDN, а також докладно проаналізовано їх ключові структурні компоненти, принципи функціонування та взаємодії.

Розглянуто, як традиційні мережі з жорсткою прив’язкою програмного забезпечення до апаратного обладнання, відсутністю централізованого управління та складністю масштабування стали стримуючим фактором в умовах стрімкої цифрової трансформації. Було показано, що потреба в гнучкості, автоматизації, безпеці та динамічному масштабуванні стала поштовхом до розвитку нових архітектур, у яких логіка управління винесена за межі пристроїв пересилання даних. У межах розділу детально охарактеризовано основні поняття SDN, зокрема ключову концепцію розділення площини керування та площини передачі, яка забезпечує логічну централізацію управління мережею. Було розкрито роль SDN-контролера як централізованого інтелектуальної ланки мережі, що приймає рішення на основі глобального бачення топології та стану трафіку. Проведено докладний аналіз трьох логічних площин SDN - передачі, керування та додатків. Площина передачі описана як набір пристроїв, що виконують команди контролера без прийняття самостійних рішень. Площина керування відповідає за логіку прийняття рішень і побудову політик, тоді як площина додатків втілює бізнес-логіку та високорівневе управління мережею. Окрему увагу приділено взаємодії між рівнями через API. Southbound API, який забезпечує керування

мережевими пристроями, та northbound API, що забезпечує зв’язок контролера з прикладними сервісами, системами аналітики, безпеки та автоматизації. Було підкреслено важливість відкритості інтерфейсів, їх безпеки та взаємної сумісності.

Узагальнюючи, можна зробити висновок, що SDN є перспективною архітектурною моделлю, яка дозволяє реалізовувати гнучкі, масштабовані та безпечні мережі, здатні динамічно адаптуватися до змін навантаження, бізнес-вимог та зовнішніх загроз. Отримані в цьому розділі знання створюють теоретичне підґрунтя для подальшого практичного дослідження SDN-рішень, зокрема на основі платформи ZeroTier, у наступних розділах роботи.

РОЗДІЛ 2. ПОБУДОВА СЕРЕДОВИЩА ТЕСТУВАННЯ SDN-МЕРЕЖІ НА ОСНОВІ ZEROTIER

2.1 Загальна характеристика ZeroTier

ZeroTier - це SDN рішення нового покоління, яке дозволяє будувати віртуальні приватні мережі будь-якої складності з мінімальними зусиллями [14]. Ця платформа поєднує можливості VPN, SD-WAN та SDN, забезпечуючи високий рівень безпеки, масштабованість, простоту розгортання та адаптацію до найрізноманітніших середовищ - від IoT до оборонної промисловості.

Одна з ключових особливостей ZeroTier - це агент малого розміру, який легко інсталюється на практично будь-який пристрій: від серверів і настільних систем до вбудованих IoT-рішень і промислових маршрутизаторів. Агенти забезпечують пряме peer-to-peer з'єднання між вузлами, мінімізуючи затримки та усуваючи необхідність у централізованому маршрутизаційному вузлі. У разі неможливості встановлення прямого з'єднання система автоматично використовує ретрансляційні сервери, зберігаючи при цьому наскрізне шифрування. У сфері IoT та Edge Computing ZeroTier забезпечує безпечну, передачу даних з малими затримками між розподіленими пристроями, що дозволяє реалізовувати реальний обмін інформацією на краю мережі, без участі традиційної хмари. Платформа підтримується провідними виробниками промислового обладнання, зокрема Teltonika, InHand Networks і MikroTik, що робить її надійним компонентом сучасних індустріальних та віддалених інфраструктур.

Як SD-WAN-рішення, ZeroTier дозволяє швидко і без складної інфраструктури об'єднувати віддалені офіси, філії, будівельні майданчики, магазини чи кампуси в єдину логічну мережу. Відсутність прив'язки до конкретного виробника обладнання (vendor-agnostic design) знижує вартість

володіння й усуває необхідність укладання довгострокових контрактів із телеком-постачальниками.

У VPN-сценаріях платформа дозволяє забезпечити безпечний доступ працівників до корпоративних ресурсів без складного налаштування, що притаманне класичним VPN [15-16]. Унікальні криптографічні ідентифікатори пристроїв, простота розгортання і підтримка наскрізного шифрування дають змогу забезпечити захищене підключення без накладних витрат на продуктивність.

У транспортній галузі та автомобільній індустрії ZeroTier реалізується як рішення для зв'язку між транспортними засобами, працівниками, заводами та хмарними інфраструктурами. Завдяки можливості ізольованого керування потоками і логічного сегментування, виробники можуть реалізовувати гнучке масштабування без залучення дорогого обладнання.

ZeroTier довів свою ефективність як мережна платформа, здатна працювати у динамічних і нестабільних умовах. Система підтримує повну автономність і відповідає суворим стандартам безпеки, включаючи SOC 2 та FIPS-140. Для державного сектору ZeroTier надає можливість швидкого розгортання захищених мережевих з'єднань між розподіленими офісами, віддаленими командами та підрядниками. Платформа підтримує обмін даними для критично важливих операцій, при цьому знижуючи витрати на IT-інфраструктуру завдяки простоті налаштування та масштабування.

Завдяки протокольно та апаратно-незалежному підходу, ZeroTier легко впроваджується в середовища з жорсткими вимогами до безпеки та без затримок у продуктивності. Платформа також широко застосовується MSP (Managed Service Providers) і телеком-компаніями, які використовують її для створення SD-WAN-пропозицій, зниження залежності від традиційних VPN, а також для об'єднання клієнтських середовищ у глобальні логічні приватні мережі. Доступність ZeroTier через AWS Marketplace, а також підтримка VAD/VAR-каналів забезпечує її легке впровадження на ринку.

Однією з головних особливостей ZeroTier є її здатність віртуалізувати мережеві підключення між пристроями без потреби у зміні конфігурацій маршрутизаторів, NAT або брандмауерів. Це досягається завдяки поєднанню принципів peer-to-peer зв'язку, автоматичного NAT traversal і централізованого контролю доступу. Після встановлення програмного агента ZeroTier на пристрій, він може підключатися до логічної мережі, створеної на платформі, та отримувати IP-адресу відповідно до її конфігурації. При цьому забезпечується наскрізне шифрування між учасниками мережі, що робить її захищеною навіть при роботі через ненадійні канали зв'язку, як-от публічні Wi-Fi або мобільні мережі. ZeroTier дозволяє будувати гібридні мережі, в яких пристрої з різною операційною системою - Windows, Linux, macOS, Android, iOS, а також мережеві маршрутизатори або віртуальні машини - можуть взаємодіяти як частина єдиного логічного сегмента. Це досягається завдяки невеликому клієнтському агенту, який виконує функції мережевого адаптера, віртуального комутатора та маршрутизатора одночасно. Весь обмін керується через ZeroTier Network Controller - логічний вузол, який відповідає за аутентифікацію вузлів, розповсюдження конфігурації, оновлення маршрутів і контроль безпеки. З технічної точки зору ZeroTier поєднує концепції рівня 2 і рівня 3 моделі OSI. Це означає, що можна створити повноцінну віртуальну L2-мережу, яка поводитиметься так, ніби всі її учасники підключені до одного фізичного комутатора, або ж L3-мережу з маршрутизованим трафіком і доступом до зовнішніх ресурсів через шлюзи. Така гнучкість дозволяє інтегрувати ZeroTier як у невеликі домашні або офісні мережі, так і у великі корпоративні інфраструктури.

Особливу увагу в ZeroTier приділено масштабованості та мінімізації затримок. Завдяки розподіленій моделі маршрутизації, вузли зазвичай намагаються встановити пряме з'єднання один з одним, минаючи центральний сервер, що значно покращує продуктивність. Якщо прямий канал неможливий, використовується оптимізований ретранслятор (relay) з мінімальним впливом на швидкість. Важливою складовою ZeroTier є також контроль доступу на

основі ACL, який дозволяє адміністраторам керувати правами учасників мережі - наприклад, обмежити доступ до певних IP-адрес або портів, заблокувати або дозволити певні типи трафіку, реалізувати ізоляцію окремих підмереж.

2.2 Принципи роботи та модель мережі ZeroTier

ZeroTier реалізує гібридну модель віртуальної мережі, яка поєднує в собі властивості peer-to-peer архітектури, SDN і технологій overlay зв’язку. Основною ідеєю ZeroTier є створення логічної мережі, яка абстрагується від фізичної інфраструктури - вузли можуть перебувати в різних підмережах, за NAT або за брандмауером, однак при цьому сприймають одне одного як частини єдиного Ethernet-сегмента. Це дозволяє будувати приватні мережі поверх глобального Інтернету так, ніби всі пристрої фізично підключені до одного комутатора.

Мережа ZeroTier працює за допомогою агентів, що інсталюються на кожному з вузлів. Після запуску агент ініціалізує з'єднання з публічною мережею ZeroTier та отримує унікальний ідентифікатор - 40-бітний Node ID, який виступає унікальним криптографічним ключем. Агент реєструється в глобальній DHT (Distributed Hash Table) для того, щоб інші вузли могли його знайти. Підключення до конкретної логічної мережі відбувається за допомогою 16-значного Network ID, який є унікальним ідентифікатором мережі, створеної адміністратором через ZeroTier Central.

У момент підключення вузла до мережі контролер ZeroTier виконує автентифікацію, призначає внутрішню IP-адресу, синхронізує конфігураційні параметри (ACL, маршрути, дозволи) та повідомляє інші вузли про появу нового учасника. Далі вузли намагаються встановити пряме peer-to-peer з'єднання, використовуючи технології NAT traversal (STUN/TURN), щоб мінімізувати затримку і виключити проміжні ланки. Якщо прямий канал з технічних причин неможливий, трафік передається через ретрансляційні

сервери (relays), що підтримуються мережею ZeroTier, однак навіть у такому випадку зберігається наскрізне шифрування.

Архітектурно модель ZeroTier є повністю децентралізованою на рівні пересилання даних, але централізованою на рівні управління. Центральний контролер не бере участі в маршрутизації трафіку, але керує ключовими параметрами мережі: адресацією, доступом, оновленням політик. Він підтримує так звану "рівноправну модель взаємодії" (flat topology), де всі учасники є рівноправними і можуть напряму обмінюватися даними без маршрутизації через хаб.

Оскільки ZeroTier працює на 2-м (канальний рівень) та 3-м (мережевий рівень) моделі OSI, то це дозволяє як емуляцію Ethernet-мереж, так і IP-маршрутизацію. У віртуальній мережі можна створювати L2-сегменти з широкомовними адресами, ARP-повідомленнями та мультикастом, або ж обмежитися L3-мережею з маршрутизованими підмережами та шлюзами доступу до Інтернету чи інших мереж. Кожен вузол ZeroTier містить віртуальний мережевий інтерфейс, який інтегрується в стек ОС. Цей інтерфейс отримує IP-адресу з простору адрес, заданого адміністратором, і працює як звичайна мережева карта. Усі пакети, що передаються через цей інтерфейс, інкапсулюються у UDP та шифруються перед відправленням по фізичній мережі. Таким чином, жодні провайдери не можуть переглядати або змінювати вміст трафіку.

Безпека моделі базується на криптографічних ідентифікаторах та наскрізному шифруванні. Кожен вузол має унікальну криптографічну пару ключів (публічний та приватний), яка використовується для автентифікації та встановлення захищеного каналу. Платформа реалізує принцип zero trust: довіра до вузла формується не на основі його IP-адреси або місця розташування, а виключно на основі криптографічного ідентифікатора і дозволу на участь у конкретній мережі.

Таким чином, модель роботи ZeroTier дозволяє об'єднувати в єдину логічну мережу будь-які пристрої - незалежно від їхнього фізичного

розташування та політик маршрутизації. Це забезпечує виняткову гнучкість і дає змогу реалізовувати складні мережеві сценарії без зміни існуючої інфраструктури, зберігаючи при цьому високий рівень безпеки, продуктивності та масштабованості.

Протокол ZeroTier - це оригінальна реалізація мережевої віртуалізації, яка дозволяє будь-яким пристроям, серверам, віртуальним машинам або контейнерам взаємодіяти між собою так, ніби вони фізично підключені до одного комутатора в локальному дата-центрі [17]. Архітектура платформи спроєктована як глобальний віртуальний гіпервізор для мереж, що поєднує криптографічно захищену peer-to-peer мережу з Ethernet-емуляційним рівнем, подібним до VXLAN, і створює віртуальні з'єднання без необхідності конфігурувати IP-маршрутизацію чи порти вручну.

У фундаменті лежить двошарова модель: нижній рівень - це VL1, який функціонує як віртуальний аналог фізичних з'єднань (кабелів або бездротових каналів), створюючи захищене peer-to-peer з'єднання між вузлами. Він відповідає за ініціацію каналів зв'язку, шифрування, автентифікацію та проходження NAT. Верхній рівень - VL2, який реалізує емуляцію Ethernet і надає інтерфейси для операційних систем, дозволяючи сприймати мережу як звичайну локальну. Цей рівень підтримує функції мережевого сегментування, моніторингу та контролю доступу, подібно до корпоративних SDN-комутаторів [18].

ZeroTier використовує унікальну адресацію, засновану на криптографії. Кожен вузол має 40-бітну адресу, яка генерується на основі його відкритого ключа. Ця адреса слугує не тільки ідентифікатором пристрою, а й криптографічним токеном, що гарантує цілісність і захист від підробок. Віртуальні мережі ідентифікуються 64-бітними ID, що складаються з адреси контролера та унікального номера мережі. Вузол може бути підключений до кількох мереж одночасно, подібно до того, як один порт Ethernet може належати кільком VLAN. Ключовим елементом роботи ZeroTier є глобальна peer-to-peer транспортна мережа VL1, яка створює так званий "віртуальний

кабель”. У мережі використовується ієрархія на основі “планети” (глобального набору root-серверів, які підтримує ZeroTier Inc.) та “місяців” - приватних кореневих вузлів, які користувачі можуть запускати самостійно. Кожен вузол у ZeroTier стартує без прямого з'єднання з іншими, маючи лише вихід до root-серверів. Під час першої передачі пакета від одного вузла до іншого система автоматично ініціює механізм встановлення прямого каналу, використовуючи техніку, відому як транспортно-спровоковане створення з'єднання. Коли трафік передається через root, вузли отримують спеціальні повідомлення типу “rendezvous”, які допомагають їм знайти шляхи для прямої взаємодії.

Peer-to-peer трафік у ZeroTier шифрується наскрізно. Ключі для шифрування належать виключно пристроям-учасникам, а контролери та root-сервери не мають доступу до вмісту пакетів. Якщо прямий канал неможливо встановити, трафік маршрутизується через ретранслятори, що повільніше, але гарантує доступність. Протокол автоматично та постійно намагається оптимізувати з'єднання, навіть у разі встановлення зв’язку лише через ретрансляційні сервери спроби створити пряме з’єднання тривають у фоновому режимі.

Всі вузли автоматично створюють і зберігають ідентифікаційні пари ключів у вигляді identity.public і identity.secret. Ці ключі є основою безпеки, а самі адреси в ZeroTier - не просто номери, а результат криптографічної обчислювальної функції, що унеможливлює підробку без значних обчислювальних ресурсів. Керування логікою віртуальних мереж здійснюється через контролери - подібно до контролерів у SDN (наприклад, OpenFlow). Адміністратор може використовувати як хмарні контролери від ZeroTier, так і розгорнути приватний інстанс, керуючи JSON-конфігураціями вручну. Контролер не бере участі в маршрутизації, але відповідає за авторизацію, надання IP-адрес, конфігурацію ACL, маршрутів та визначення сегментів.

В основі філософії протоколу лежать ідеї "deperimeterization" (усунення жорстких мережевих меж) та Zero Trust, які передбачають, що безпека повинна забезпечуватись не розташуванням вузла в мережі, а його криптографічною

автентичністю та дозволами на доступ. Це дає змогу будувати масштабовані, ізольовані та гнучкі мережі без потреби в централізованій інфраструктурі або складному налаштуванні традиційних VPN і брандмауерів.

2.3 Безпека в ZeroTier

Криптографічна безпека ZeroTier - це один із найважливіших аспектів його архітектури, завдяки якому забезпечується цілісність, конфіденційність і автентичність даних, що передаються між вузлами в глобальній peer-to-peer мережі [19]. Протокол ZeroTier був спроєктований з урахуванням сучасних криптографічних стандартів і практик, рекомендованих професійною спільнотою фахівців з інформаційної безпеки. Усі пакети, що циркулюють мережею ZeroTier на рівні VL1 шифруються наскрізним методом. Це означає, що дані шифруються на відправнику і можуть бути розшифровані лише безпосередньо на приймачі, причому жоден проміжний вузол, включаючи root-сервери або ретранслятори, не має доступу до вмісту пакета. Усе це робить ZeroTier невидимим для будь-яких спостерігачів на рівні мережі - від провайдерів до потенційних зловмисників.

Для реалізації асиметричного шифрування та генерації ключів ZeroTier використовує Curve25519/Ed25519 це сучасна еліптична крива 256-бітної довжини, яка визнана як одна з найбезпечніших та найефективніших криптографічних конструкцій у своєму класі. Ця крива використовується для генерації публічних та приватних пар ключів, автентифікації вузлів, ініціалізації захищених сесій та ідентифікації пристроїв у мережі. Кожен пакет VL1 додатково шифрується симетричним потокових шифром Salsa20 (256-біт). Для перевірки цілісності та автентичності пакета застосовується алгоритм MAC (Message Authentication Code) - Poly1305. Реалізація заснована на моделі encrypt-then-MAC - спочатку виконується шифрування, потім обчислюється контрольна сума повідомлення. Така схема вважається найбільш безпечною з погляду криптографії. Вибрані алгоритми (Salsa20 + Poly1305) є частиною

бібліотеки NaCl (Networking and Cryptography Library) і рекомендовані до застосування в критичних системах через свою ефективність і надійність.

ZeroTier не використовує на рівні VL1 так звану forward secrecy - механізм, при якому кожна сесія генерує унікальний тимчасовий ключ, що знищує можливість дешифрування історичного трафіку навіть у разі компрометації ключа. Це рішення зумовлене прагненням до простоти, мінімального коду та забезпечення стабільності при масштабуванні, особливо у сценаріях із кластеризацією та аварійним відновленням. Разом з тим, команда розробників розглядає можливість впровадження forward secrecy у майбутніх версіях протоколу. На даному етапі користувачам, які потребують forward secrecy вже сьогодні, рекомендується використовувати додаткові криптографічні протоколи поверх ZeroTier, зокрема SSL/TLS або SSH. Використання таких протоколів забезпечує багаторівневий захист (defense-in-depth), оскільки навіть у малоймовірному випадку компрометації реалізації одного з рівнів шифрування, другий рівень забезпечує додатковий бар’єр безпеки. Практика подвійного шифрування не має істотного негативного впливу на продуктивність більшості пристроїв, особливо тих, які мають апаратне прискорення для криптографічних операцій.

Криптографічна модель ZeroTier не тільки надійно захищає дані, але й мінімізує ризик зовнішнього втручання або віддаленого аналізу трафіку. Оскільки адреси вузлів генеруються на основі публічних ключів, ідентифікація користувачів у системі не пов’язана з IP-адресами або фізичним розташуванням, що забезпечує анонімність і приватність у транспортному шарі. До того ж, механізм захисту від генерації колізій (адрес із однаковими ідентифікаторами) вимагає обчислювальних зусиль у десятки тисяч CPU-років, що робить атаки подібного роду не лише складними, а й економічно невиправданими.

У результаті, криптографічний стек ZeroTier є ретельно збалансованим між практичною безпекою, швидкодією та гнучкістю, забезпечуючи високий рівень захисту без надлишкового навантаження на систему. Його дизайн

відповідає сучасним вимогам до захищеного транспорту як у корпоративному, так і в персональному використанні, а можливість інтеграції із зовнішніми протоколами шифрування робить його придатним навіть для найвимогливіших сценаріїв.

2.4 Схема середовища тестування

З метою перевірки функціональності програмно-визначеної мережі на базі ZeroTier у сценаріях, що максимально наближені до реальних умов було побудовано середовище тестування [20].

Центральним елементом виступає віртуальний маршрутизатор MikroTik CHR [21], розгорнутий у середовищі гіпервізора Hyper-V [22]. На цей маршрутизатор інсталюється контейнерна версія клієнта ZeroTier, що дозволяє CHR інтегруватися до логічної віртуальної мережі ZeroTier. Це рішення забезпечує з’єднання між внутрішньою інфраструктурою тестового середовища та зовнішніми вузлами ZeroTier. Внутрішня мережа складається з віртуальної машини Windows Server 2022 з увімкненою службою RDS, яка виступає як цільовий вузол для тестування віддаленого доступу через тунель ZeroTier. Додатково до складу тестового середовища входить клієнтська операційна система на базі Ubuntu Linux, яка має встановлений ZeroTier-агент та підключається до тієї ж самої віртуальної мережі ZeroTier. Цей вузол використовується для моделювання підключення ззовні, з умовної публічної мережі (див. рисунок 2.1).

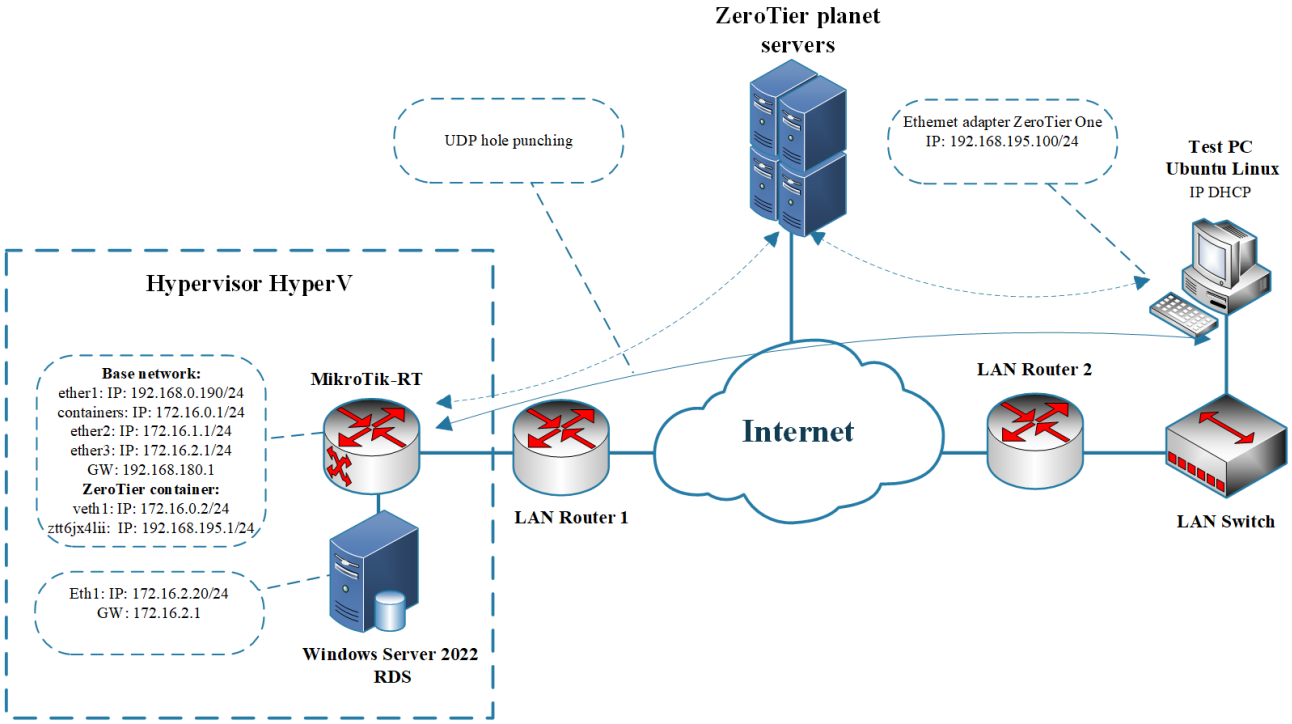

Рисунок 2.1 – Схема середовища тестування

MikroTik має кілька віртуальних інтерфейсів: ether1 підключено до мережі Інтернет з IP-адресою 192.168.0.190/24, ether2 і ether3 обслуговують локальні підмережі 172.16.1.0/24 і 172.16.2.0/24. Також окремо виділена мережа 172.16.0.0/24 для контейнерів, у якій розміщено контейнері ZeroTier з IP 172.16.0.2 та внутрішнім адаптером ZeroTierOne з IP-адресою 192.168.195.1/24 - це його віртуальна IP у логічній ZeroTier мережі.

Усі дані передаються через UDP, з можливістю використання механізму UDP hole punching для обходу NAT, забезпечуючи пряме з’єднання між вузлами. Контейнерний агент ZeroTier з'єднаний з публічною мережею ZeroTier Planet Servers, яка забезпечує контроль доступу, аутентифікацію та маршрутизацію логічних ідентифікаторів вузлів.

Тестова система Ubuntu Linux виступає в ролі віддаленого клієнта, підключеного до локального маршрутизатора (LAN Router 2) через звичайну комутовану локальну мережу. ZeroTier-клієнт інстальовано безпосередньо на операційній системі Ubuntu, де він створює віртуальний Ethernet-адаптер із IP-

адресою в просторі ZeroTier, зокрема 192.168.195.100/24. Завдяки цьому Ubuntu-клієнт має доступ до сервера у DMZ.

На сервері Windows Server 2022 RDS налаштовано інтерфейс з IP 172.16.2.20/24, що маршрутизується через MikroTik до контейнера ZeroTier. Уся мережева взаємодія між Ubuntu і Windows Server 2022 відбувається через тунель ZeroTier.

Таким чином, представлена схема середовища тестування є комплексною моделлю SDN-рішення, яке демонструє практичне застосування ZeroTier у ролі універсального інструмента для побудови захищеного мережевого доступу між географічно розділеними сегментами інфраструктури. Дана схема тестового середовища дозволяє комплексно перевірити усі функціональні компоненти ZeroTier у поєднанні з реальною серверною інфраструктурою, змодельованим зовнішнім доступом [23].

2.5 Елементи середовища тестування

2.5.1 Гіпервізор Hyper-V

Hyper-V - це вбудований в операційну систему Microsoft Windows гіпервізор типу 1, який дозволяє створювати, запускати та керувати віртуальними машинами та іншими ізольованими середовищами на фізичному сервері або робочій станції [24]. Його архітектура розроблена таким чином, щоб забезпечити максимально ефективне використання апаратних ресурсів, включаючи процесор, пам’ять, мережу та сховище, при цьому надаючи кожній віртуальній машині незалежне середовище з власною операційною системою.

Hyper-V функціонує на рівні ядра і отримує прямий доступ до фізичного обладнання, що дає змогу уникати надлишкових витрат ресурсів на проміжні програмні рівні. У системі керування Hyper-V особливе місце займає так званий management розділ - це основна інсталяція Windows, яка взаємодіє з користувачем і містить інструменти адміністрування. Всі інші інстанси, відомі

як guest віртуальні машини, виконуються в ізольованих середовищах, які не мають прямого доступу до фізичних пристроїв, а лише через віртуалізовані канали, що контролюються гіпервізором.

У контексті тестування мережевих рішень, таких як ZeroTier, Hyper-V забезпечує масштабовану платформу для розгортання маршрутизаторів, серверів, контейнерів або спеціалізованих пристроїв, які можуть взаємодіяти між собою як в межах одного вузла, так і через зовнішні мережеві підключення. У Hyper-V реалізовано кілька типів віртуальних комутаторів - зовнішній, внутрішній і приватний. Завдяки цьому можливо точно контролювати, які саме мережеві адаптери гостьових систем мають доступ до локальної мережі, Інтернету або ізольованого середовища, що особливо важливо для безпечного тестування DMZ-сценаріїв.

Завдяки підтримці знімків стану віртуальних машин, Hyper-V дозволяє швидко повертати систему до попереднього стану у випадку збоїв або потреби повторного тестування. Це дає змогу створювати відтворювані сценарії моделювання мережевої поведінки, зокрема перевірку працездатності тунелів ZeroTier у складних конфігураціях з декількома рівнями NAT, ізольованими маршрутами та DMZ-сегментами.

На рисунку 2.2 представлено інтерфейс Hyper-V Manager, який використовується для керування віртуальними машинами у середовищі Microsoft Hyper-V. В віртуальному середовищі розгорнуто дві віртуальні машини: MikroTik та Windows Server 2022 RDS. Обидві машини мають статус "Running", що свідчить про їхню активну роботу в межах віртуалізованого стенду.

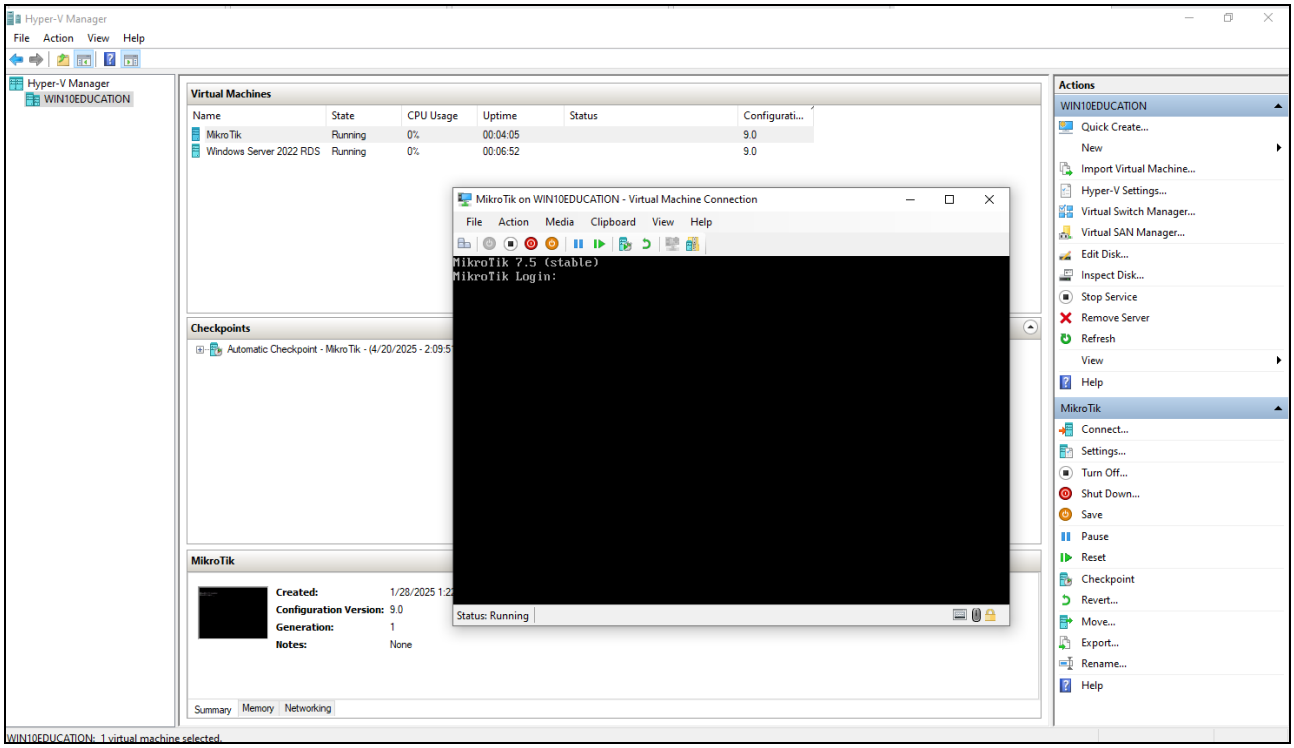

Рисунок 2.2 – Інтерфейс Hyper-V Manager

Hyper-V надає детальну інформацію про кожну віртуальну машину, включно з наявними знімками системи (Checkpoints).

Таким чином, Hyper-V виступає як високопродуктивна, надійна та безкоштовна основа для побудови віртуалізованого тестового середовища, де можна реалізовувати повноцінні архітектури програмно-визначених мереж, емулювати роботу хмарної або корпоративної інфраструктури та досліджувати взаємодію між компонентами ZeroTier і класичними мережевими пристроями у контрольованому середовищі.

2.5.2 Маршрутизатор MikroTik

Маршрутизатор MikroTik у контексті побудови програмно-визначеної мережі виконує функцію мережевого вузла, який поєднує фізичну інфраструктуру з логічним рівнем мережі ZeroTier [25]. У даній роботі використано віртуалізований варіант MikroTik у конфігурації CHR, який

працює як повноцінний маршрутизатор із підтримкою контейнеризації, VLAN, NAT, статичної та динамічної маршрутизації, тунелів та брандмауера [26].

MikroTik CHR використовує спеціальну версію RouterOS [27], яка адаптована для розгортання в середовищі гіпервізорів, таких як Hyper-V, VMware, KVM тощо. Завдяки цьому вона не потребує фізичного обладнання, але при цьому зберігає всі можливості фізичних моделей MikroTik. Після запуску в Hyper-V MikroTik CHR отримує кілька віртуальних інтерфейсів, кожен з яких підключено до певного сегмента мережі. Це дозволяє здійснювати ізоляцію трафіку, розділення зон доступу, маршрутизацію між підмережами та організацію DMZ.

У наведеному тестовому середовищі маршрутизатор виконує роль транзитного вузла між фізичною і логічною топологією, дозволяючи доступ до внутрішніх серверів з віддалених клієнтів, які підключені через ZeroTier. У ролі шлюзу MikroTik отримує пакети з віртуального адаптера контейнера ZeroTier і передає їх до підмережі, в якій розташований сервер Windows. У зворотному напрямку він здійснює маршрутизацію назад до логічного тунелю ZeroTier. Завдяки можливості створення статичних маршрутів і політик брандмауера, адміністратор має повний контроль над потоком трафіку, доступом до сервісів, а також над правилами NAT, що забезпечує безпечне з'єднання між сегментами.

На рисунку 2.3 показано вивід команди `ip/address/print detail`, яка відображає всі налаштовані IP-адреси інтерфейсів із розширеною інформацією в маршрутизаторі MikroTik CHR.

```
[admin@MikroTik-RT] > ip/address/print detail
Flags: X - disabled, I - invalid, D - dynamic
 0   ;;; For containers
     address=172.16.0.1/24 network=172.16.0.0 interface=containers
     actual-interface=containers

 1   ;;; WAN
     address=192.168.0.190/24 network=192.168.0.0 interface=ether1
     actual-interface=ether1

 2   ;;; LAN
     address=172.16.1.1/24 network=172.16.1.0 interface=ether2
     actual-interface=ether2

 3   ;;; Servers
     address=172.16.2.1/24 network=172.16.2.0 interface=ether3
     actual-interface=ether3
[admin@MikroTik-RT] >
```

Рисунок 2.3 – Вивід команди `ip/address/print detail` на маршрутизаторі MikroTik CHR

У конфігурації представлено чотири мережеві інтерфейси, кожен з яких прив’язаний до певного сегмента мережі. Кожен інтерфейс має свою IP-адресу, маску підмережі. Ця конфігурація демонструє чітке логічне розділення мереж на основі функціонального призначення: окремі сегменти для контейнерів, WAN, локальної мережі та серверної інфраструктури.

На рисунку 2.4 представлено вивід команди `ip/route/print` на маршрутизаторі MikroTik CHR.

```
[admin@MikroTik-RT] > ip/route/print
Flags: D - DYNAMIC; A - ACTIVE; c - CONNECT, s - STATIC
Columns: DST-ADDRESS, GATEWAY, DISTANCE
#      DST-ADDRESS       GATEWAY      DISTANCE
0  As  0.0.0.0/0         192.168.0.1         1
   DAc 172.16.0.0/24     containers          0
   DAc 172.16.1.0/24     ether2              0
   DAc 172.16.2.0/24     ether3              0
   DAc 192.168.0.0/24    ether1              0
1  As  192.168.195.0/24  172.16.0.2          1
[admin@MikroTik-RT] >
```

Рисунок 2.4 – Вивід команди `ip/route/print` на маршрутизаторі MikroTik CHR

На рисунку 2.5 представлено вивід конфігураційних параметрів інтерфейсів маршрутизатора MikroTik CHR.

```
[admin@MikroTik-RT] > interface/print
Flags: R - RUNNING; S - SLAVE
Columns: NAME, TYPE, ACTUAL-MTU, L2MTU, MAC-ADDRESS
#     NAME         TYPE       ACTUAL-MTU  L2MTU  MAC-ADDRESS
;;; WAN
0 R   ether1       ether            1500         00:0C:29:DE:1A:4C
;;; LAN
1 R   ether2       ether            1500         00:0C:29:DE:1A:56
;;; Servers
2 R   ether3       ether            1500         00:0C:29:DE:1A:60
3 R   containers   bridge           1500  65535  62:EB:EF:4D:76:DE
4 R   lo           loopback        65536         00:00:00:00:00:00
5 RS  veth1        veth             1500         62:EB:EF:4D:76:DE
[admin@MikroTik-RT] > interface/veth/print
Flags: X - disabled; R - running
 0   R name="veth1" address=172.16.0.2/24 gateway=172.16.0.1 gateway6=""
[admin@MikroTik-RT] >
```

Рисунок 2.5 – Вивід конфігураційних параметрів інтерфейсів на маршрутизаторі MikroTik CHR

Команда `interface/print` показує список всіх інтерфейсів з деталями про їхні типи, статус, MTU та MAC-адреси. Додатково виведено специфікацію віртуального Ethernet-інтерфейсу в контейнері ZeroTier через команду `interface/veth/print`.

Окремий інтерфейс containers має тип bridge, що вказує на його роль у створенні моста між контейнерами та основною маршрутизованою інфраструктурою. Через цей міст контейнер з ZeroTier може отримувати доступ до мережі маршрутизатора. Інтерфейс veth1 належить до категорії veth - це віртуальний Ethernet-інтерфейс, створений у межах контейнеризованого середовища MikroTik. Цей інтерфейс зв'язує контейнер з базовою bridge-мережею через інтерфейс containers, дозволяючи контейнеру працювати як повноцінний мережевий вузол. Veth1 має IP-адресу 172.16.0.2/24 і використовує шлюз 172.16.0.1, який належить MikroTik. Це дає змогу контейнеру, що працює всередині системи, зокрема агенту ZeroTier, направляти трафік до інших мережевих сегментів.

Завдяки такій структурі реалізовано мережеву ізоляцію ZeroTier-контейнера в окремому підмережному сегменті, з одночасним доступом до основної інфраструктури.

2.5.3 Операційна система Windows Server 2022

Windows Server 2022 - це серверна операційна система від корпорації Microsoft, яка є найновішим на момент дослідження представником сімейства Windows Server [28]. Однією з ключових особливостей Windows Server 2022 є її орієнтованість на гібридну хмарну інтеграцію, тобто здатність поєднувати локальну інфраструктуру з сервісами Microsoft Azure або іншими хмарними платформами. У контексті безпеки Windows Server 2022 запроваджує технології Secured-core server, які використовують апаратно-прискорену ізоляцію, безпечний завантажувач, контроль цілісності прошивки, верифікацію драйверів і захищену віртуалізацію. Операційна система інтегрується з Windows Defender, ATP (Advanced Threat Protection), контролем доступу на основі ролей (RBAC), а також підтримує оновлені криптографічні протоколи, включно з TLS 1.3 та AES-256 в Active Directory. Це робить її придатною для використання у критичних обчислювальних середовищах, включаючи фінансові установи, державні структури.

Windows Server 2022 підтримує як класичну модель розгортання у вигляді Desktop Experience з графічним середовищем, так і Server Core, яка позбавлена GUI У лабораторному середовищі Windows Server 2022 використовується як цільовий вузол, до якого організовано доступ через віртуальну мережу ZeroTier. Його розгортання відбувається у віртуальному середовищі Hyper-V. У конфігурації включено службу RDS [29], яка надає можливість безпечного віддаленого підключення до робочого столу через тунель ZeroTier, без потреби відкривати порти в зовнішній мережі. У контексті дослідження це дозволяє перевірити, наскільки безпечно та стабільно працює доступ з віддаленого

клієнта на Ubuntu до служб RDS, і наскільки легко інтегрується традиційна інфраструктура Windows у новітні моделі overlay-мереж.

На рисунку 2.6 представлено мережеві налаштування Windows Server 2022.

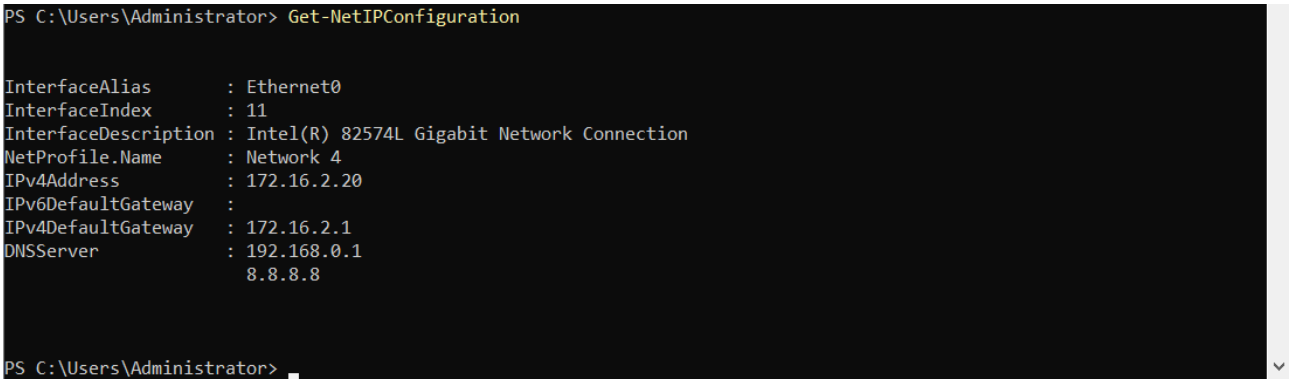

Рисунок 2.6 – Мережеві налаштування Windows Server 2022

На рисунку 2.7 показано консоль PowerShell у середовищі Windows Server 2022, в якій здійснюється поетапна перевірка налаштувань, пов’язаних зі службами RDS та правами користувача artur на підключення через RDP.

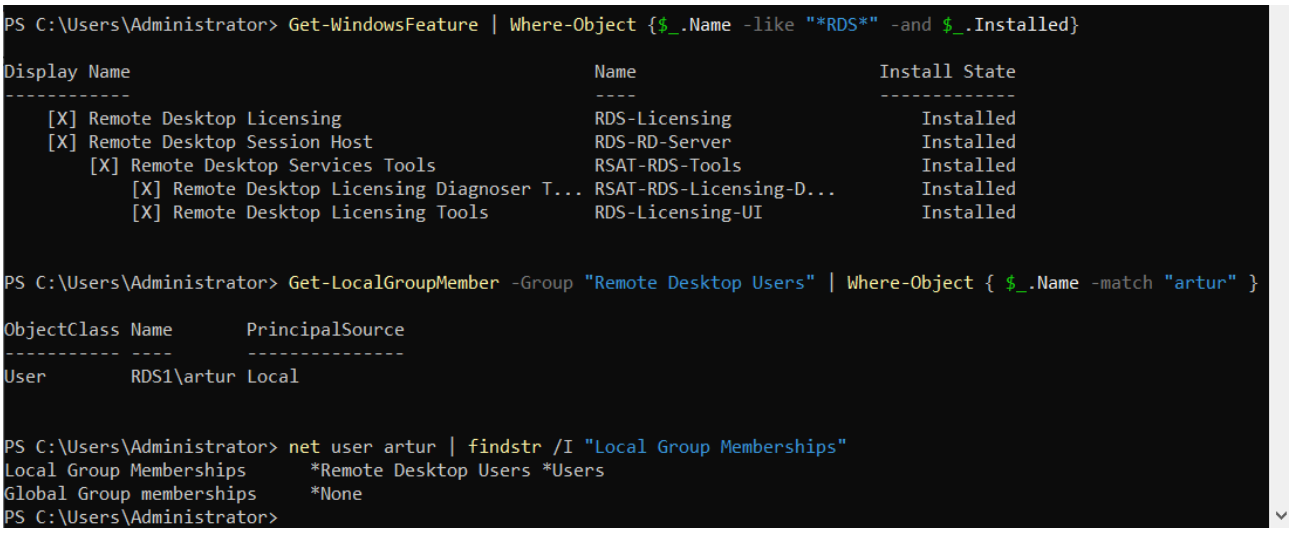

Рисунок 2.7 – Перевірка налаштувань пов’язаних зі службами RDS Windows Server 2022

Усі три блоки команд демонструють узгоджені результати: RDS служби встановлено, користувач існує, і він має повні дозволи для підключення через віддалений робочий стіл. Це означає, що зі сторони серверної операційної системи всі умови для доступу виконано, і подальше тестування підключення з боку клієнта Ubuntu через ZeroTier може проводитися без необхідності додаткових змін на сервері.

2.5.4 Операційна система Ubuntu Linux

Ubuntu Linux - це сучасна операційна система, яка орієнтована на простоту використання, стабільність і підтримку широкого спектру апаратного забезпечення [30]. Вона є одним з найпопулярніших дистрибутивів Linux як серед початківців, так і серед професіоналів у сфері інформаційних технологій, системного адміністрування, розробки та досліджень. Ubuntu підтримується компанією Canonical і випускається у двох основних гілках: стандартній (із регулярними оновленнями кожні півроку) та довготривалій підтримці LTS (Long-Term Support), яка оновлюється раз на два роки і отримує оновлення безпеки та підтримку протягом щонайменше п’яти років. Це робить Ubuntu привабливою для серверних і критичних середовищ, де стабільність є пріоритетом. Ubuntu поширюється як у вигляді десктопної версії з графічним середовищем GNOME, так і в серверній редакції без графічного інтерфейсу, яка оптимізована для продуктивної роботи в терміналі.

На рисунку 2.8 представлено мережеві налаштування Ubuntu Linux.

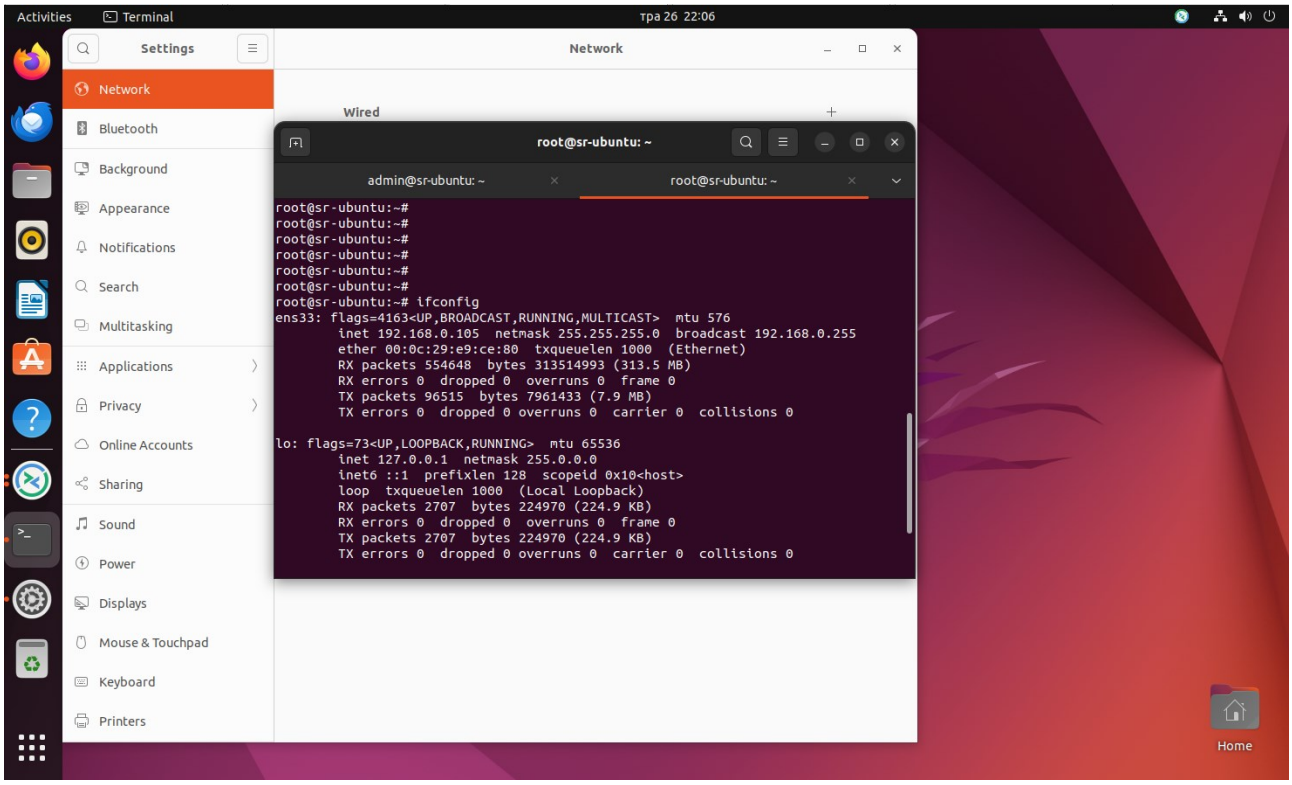

Рисунок 2.8 – Мережеві налаштування Ubuntu Linux

У контексті лабораторного середовища Ubuntu Linux виконує роль клієнтської системи, яка підключається до віртуальної SDN-мережі ZeroTier. Завдяки легкій інсталяції та повній підтримці ZeroTier-клієнта, ця операційна система дозволяє створити повноцінний вузол логічної overlay-мережі, що забезпечує захищене peer-to-peer з’єднання з сервером Windows Server 2022. Ubuntu також є потужним середовищем для тестування мережевих з'єднань, VPN, тунелювання, сканування портів, аналізу трафіку та розробки скриптів автоматизації. У графічному середовищі GNOME доступні програми для віддалених з'єднань через RDP, зокрема Remmina [31], яка дозволяє протестувати RDP-підключення до Windows Server через тунель ZeroTier.

У лабораторному середовищі Ubuntu Linux виступає елементом тестування - як клієнт, що ініціює з’єднання через ZeroTier до DMZ-сегмента, вона дозволяє емулювати сценарій віддаленого доступу до корпоративної інфраструктури. Це дозволяє перевірити наскільки ефективно реалізовано механізм тунелювання, наскільки безпечно та стабільно працює доступ до

служб Windows Server та чи витримує система сучасні вимоги до розмежування доступу в overlay-мережах.

2.6 Висновок до другого розділу

В другому розділі було здійснено докладний аналіз технології ZeroTier та побудова тестового середовища з використанням віртуалізованих інфраструктур і реальних прикладних сценаріїв доступу. Розглянуто загальні характеристики ZeroTier як гнучкого, масштабованого та безпечного рішення, яке поєднує в собі переваги VPN, SD-WAN та SDN підходів. Особливу увагу приділено опису архітектурних принципів роботи ZeroTier, моделі peer-to-peer з наскрізним шифруванням, криптографічним ідентифікаторам та моделі розподіленого з'єднання з централізованим керуванням.

Було детально описано елементи тестового середовища: маршрутизатор MikroTik CHR з контейнером ZeroTier, сервер Windows Server 2022 RDS, клієнтську систему Ubuntu Linux як віддалений вузол, що ініціює з’єднання через логічну мережу ZeroTier, а також гіпервізор Hyper-V, який виконує роль платформи віртуалізації.

Таким чином, у межах другого розділу було не лише описано теоретичні основи побудови SDN-мереж із використанням ZeroTier, а й створено практичне середовище для моделювання сучасного сценарію безпечного віддаленого доступу до внутрішньої серверної інфраструктури.

РОЗДІЛ 3. НАЛАШТУВАННЯ ТА ТЕСТУВАННЯ SDN-МЕРЕЖІ НА ОСНОВІ ZEROTIER

3.1 Створення мережі ZeroTier

Створення мережі в ZeroTier починається з формування логічного простору, який дозволить вузлам, розміщеним у різних фізичних або географічних середовищах, взаємодіяти як частини єдиного віртуального Ethernet-сегмента. Увесь процес базується на філософії "мережа як код", де адміністратор керує поведінкою, адресацією та доступом через централізований контролер - веб-інтерфейс ZeroTier Central [32]. У більшості випадків, особливо в невеликих або дослідницьких середовищах, адміністратори використовують хмарну платформу ZeroTier Central, яка є безкоштовною для невеликої кількості вузлів.

На сайті my.zerotier.com можна створювати нові віртуальні мережі. При створенні мережі системою автоматично генерується унікальний 16-значний Network ID, який слугує ідентифікатором логічного сегмента і використовується клієнтами для приєднання (див. рисунок 3.1).

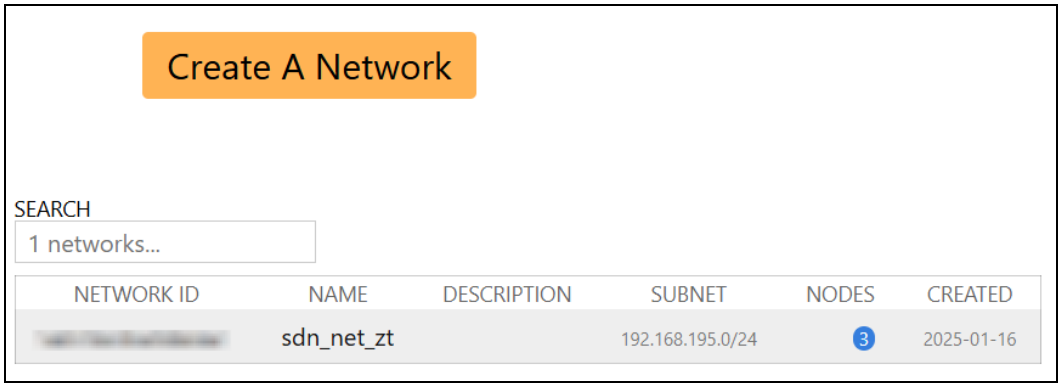

Рисунок 3.1 – Вебінтерфейс ZeroTier Central

NETWORK ID є унікальним 16-символьним ідентифікатором мережі, який слугує ключем для приєднання нових пристроїв до цієї логічної

структури. NAME вказує на ім’я мережі. Ім’я використовується лише для ідентифікації в інтерфейсі адміністратора і не впливає на функціонування мережі. У полі SUBNET зазначена IP-адресна підмережа - 192.168.195.0/24, яка буде використана для призначення адрес вузлам, що приєднуються до цієї логічної мережі. Вузли, які входять до мережі, отримують унікальні IP-адреси з цієї підмережі. Ці адреси дозволяють вузлам взаємодіяти один з одним у рамках цієї віртуальної приватної мережі, незалежно від їх реального фізичного розташування чи зовнішніх IP. Колонка NODES показує кількість пристроїв, які наразі підключені до цієї мережі.

На рисунку 3.2 представлено вебінтерфейс ZeroTier Central, а саме панель налаштувань для окремої віртуальної мережі з іменем sdn_net_zt.

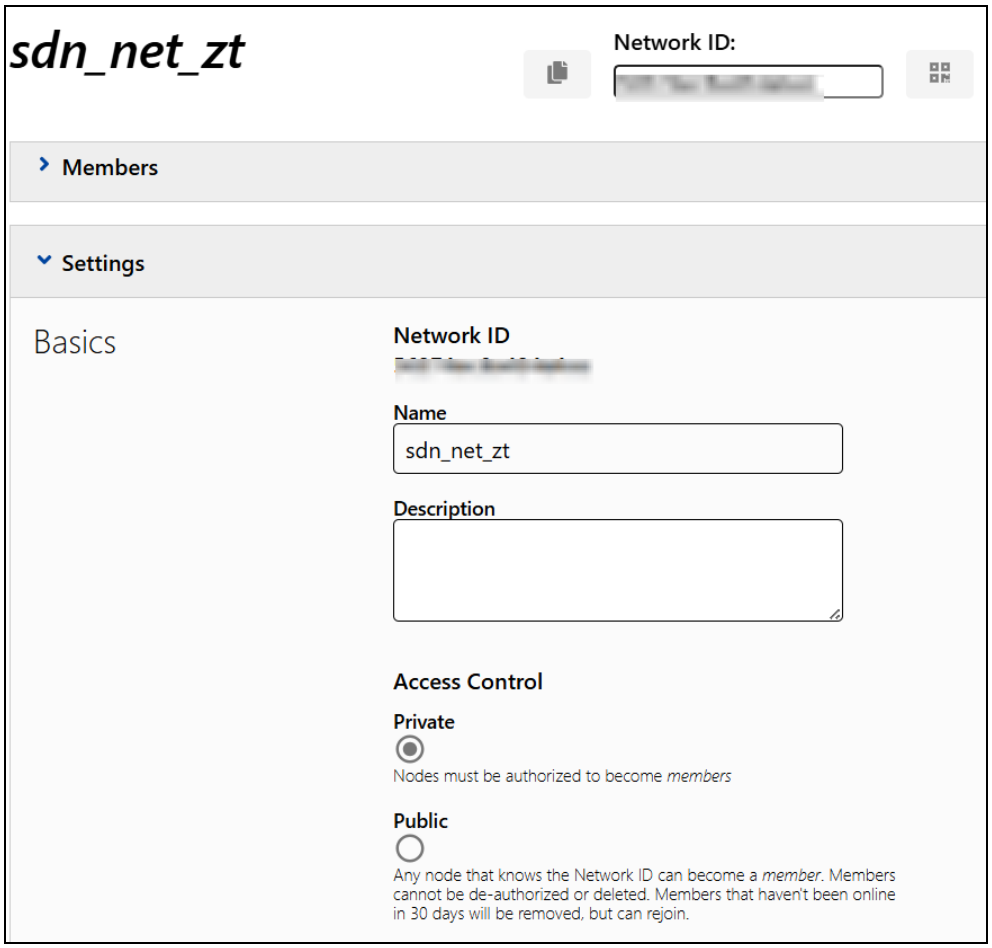

Рисунок 3.2 – Панель налаштувань мережі sdn_net_zt в ZeroTier Central

Параметр Private означає, що всі нові вузли, які спробують приєднатися до цієї мережі за допомогою її Network ID, повинні бути авторизовані вручну адміністратором. Це гарантує, що доступ отримують лише перевірені учасники, чим забезпечується високий рівень безпеки та контроль над усіма вузлами в логічній мережі. Другий доступний варіант Public дозволяє будь-якому пристрою, який знає Network ID, автоматично стати членом мережі без потреби у схваленні.

На рисунку 3.4 показано секцію Managed Routes у веб-інтерфейсі ZeroTier Central, яка відповідає за налаштування маршрутизації трафіку всередині віртуальної мережі.

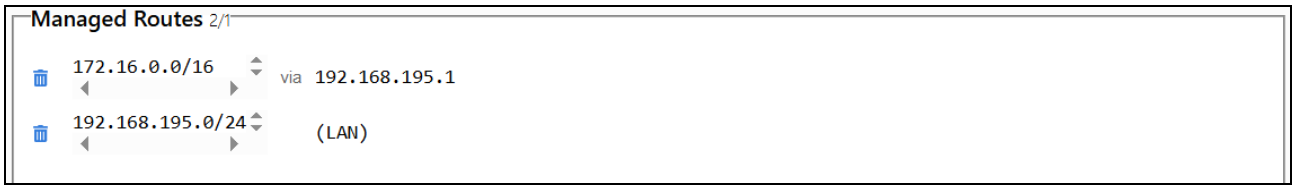

Рисунок 3.3 – Секція Managed Routes в ZeroTier Central

Цей розділ дозволяє адміністраторам вказати, які IP-підмережі будуть досяжні через які вузли мережі, тобто через які ZeroTier-агенти повинен передаватися трафік до певних адресних діапазонів.

У першому рядку налаштовано маршрут до підмережі 172.16.0.0/16 через вузол з IP-адресою 192.168.195.1. Це означає, що весь трафік, призначений для будь-якої адреси в межах цієї підмережі буде передаватися через пристрій ZeroTier, який має віртуальну адресу 192.168.195.1. У тестовому середовищи ця адреса належить агенту ZeroTier, встановленому в контейнері всередині MikroTik CHR, який виконує роль шлюзу до внутрішньої фізичної або віртуалізованої інфраструктури. Таким чином, маршрути дозволяють іншим учасникам ZeroTier-мережі взаємодіяти з серверами чи іншими вузлами в мережі, які фізично розташовані в сегменті 172.16.x.x.

Другий маршрут стосується підмережі 192.168.195.0/24, яка є основною мережею самої ZeroTier. Усі учасники, яким призначено IP-адреси з цього

діапазону зможуть взаємодіяти один з одним безпосередньо, без необхідності в специфічній маршрутизації через інші пристрої.

3.2 Контейнер ZeroTier в маршрутизаторі MikroTik

Контейнер ZeroTier у маршрутизаторі MikroTik CHR є спеціалізованим середовищем, у якому виконується ізольований агент ZeroTier, що забезпечує інтеграцію віртуального маршрутизатора в логічну overlay-мережу ZeroTier [33]. У цій конфігурації контейнер створено на базі функціоналу RouterOS, який у версіях з підтримкою containerization дозволяє розгортати окремі програмні компоненти безпосередньо в операційному середовищі маршрутизатора, подібно до звичних Docker-контейнерів [33]. На рисунку 3.4 представлено вивід команди `ip add` у контейнері ZeroTier.

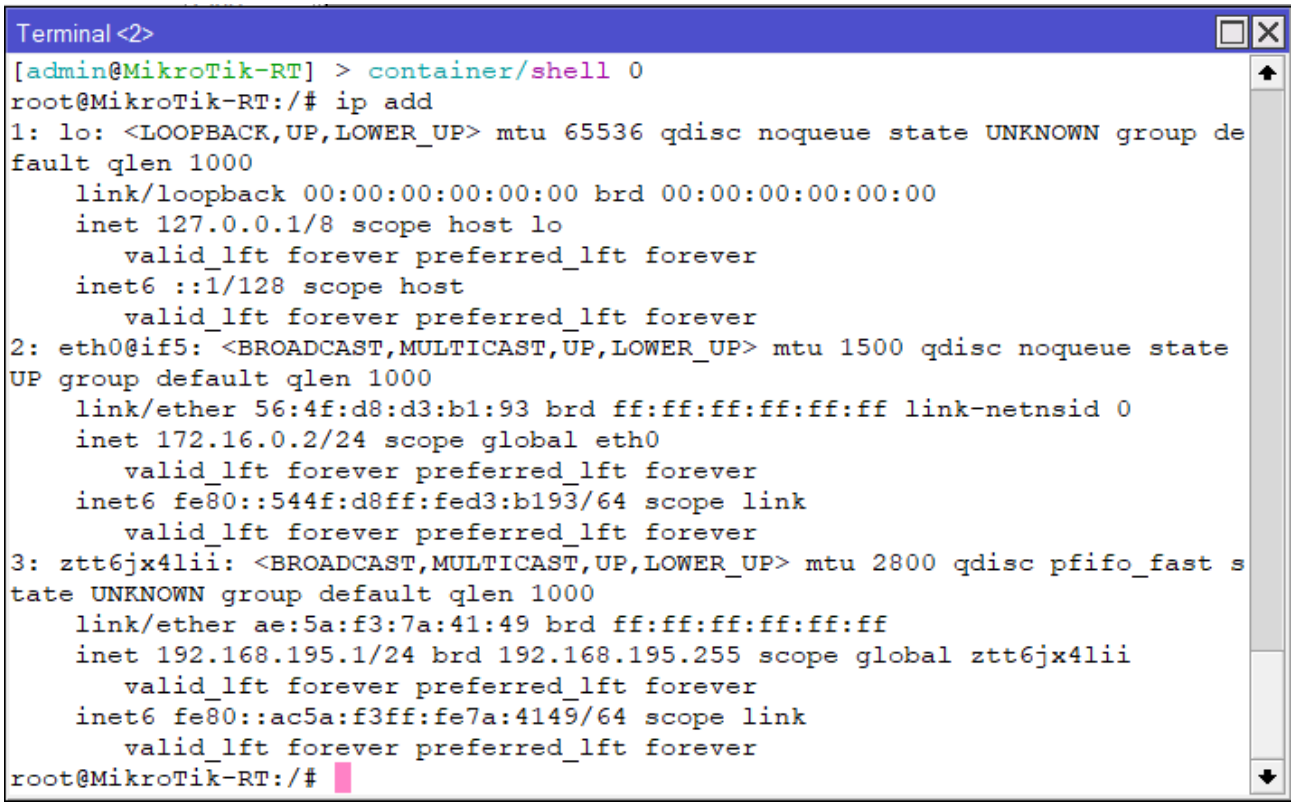

Рисунок 3.4 – Вивід команди `ip add` у контейнері ZeroTier

Такий підхід дозволяє обійти обмеження класичних маршрутизаторів, які не мають вбудованого ZeroTier-клієнта, і водночас інтегрувати їх у глобальну SDN-мережу без зовнішніх проксі чи додаткових пристроїв.

Інтерфейс eth0@if5 є основним мережевим інтерфейсом контейнера, який з'єднано з bridge-інтерфейсом MikroTik. Йому призначено адресу 172.16.0.2/24 - це IP з підмережі, зарезервованої на маршрутизаторі MikroTik для контейнерів. Саме через цей інтерфейс контейнер має доступ до внутрішніх сегментів мережі. Інтерфейс ztt6jx4lii - це віртуальний адаптер ZeroTier One, створений після підключення контейнера до логічної overlay-мережі ZeroTier. Цей інтерфейс отримав IP-адресу 192.168.195.1/24, яка належить до віртуального сегменту ZeroTier і використовується для тунельного трафіку між вузлами ZeroTier. Цей інтерфейс фактично виступає шлюзом для всієї логічної overlay-мережі. Через нього відбувається зашифрований peer-to-peer обмін даними з іншими членами мережі ZeroTier.

На рисунку 3.5 зображено результати виконання кількох команд, надають діагностичну інформацію про стан агента та підключення до інфраструктури ZeroTier.

```
root@MikroTik-RT:/# zerotier-cli status
200 info bec7de8803 1.14.2 ONLINE
root@MikroTik-RT:/# zerotier-cli listpeers
200 listpeers <ztaddr> <path> <latency> <version> <role>
200 listpeers 56374ac9a4 35.209.155.175/48503;270;2546 141 1.14.2 LEAF
200 listpeers 778cde7190 103.195.103.66/9993;270;105648 154 - PLANET
200 listpeers cafe04eba9 84.17.53.155/9993;270;105768 34 - PLANET
200 listpeers cafe80ed74 185.152.67.145/9993;270;105607 195 - PLANET
200 listpeers cafefd6717 79.127.159.187/9993;270;105521 278 - PLANET
root@MikroTik-RT:/# zerotier-cli listnetworks
200 listnetworks <nwid> <name> <mac> <status> <type> <dev> <ZT assigned ips>
200 listnetworks [redacted] sdn_net_zt [redacted] OK PRIVATE ztt6
jx4lii 192.168.195.1/24
root@MikroTik-RT:/#
```

Рисунок 3.5 – Вивід команд діагностичної інформації у контейнері ZeroTier

Перша команда `zerotier-cli status` виводить загальний стан агента ZeroTier. Вона показує унікальний ідентифікатор вузла, версію встановленого

клієнта (1.14.2) та статус ONLINE, що означає, що агент успішно підключений до мережі ZeroTier і може приймати та надсилати трафік. Команда `zerotier-cli listpeers` виводить список активних peer-з'єднань між локальним вузлом і іншими нодами у глобальній peer-to-peer інфраструктурі. Кожен запис містить ID віддаленого вузла (Ztaddr), IP-адресу та порт, через який встановлено зв'язок (Path), затримку у мілісекундах (Latency), версію клієнта (Version) та роль вузла в інфраструктурі (Role). Зокрема, серед ролей вказано PLANET, що позначає глобальні root-сервери ZeroTier. Наявність підключення до кількох PLANET-вузлів свідчить про нормальну роботу агенту та можливість встановлення з'єднання з іншими peer-ами. Наступна команда `zerotier-cli listnetworks` відображає інформацію про логічну мережу, до якої приєднано цей вузол. У колонці <name> вказано назву мережі sdn_net_zt, яка відповідає раніше створеній та налаштованій мережі в ZeroTier Central. MAC-адреса виводиться як ідентифікатор пристрою, статус показано як OK, що свідчить про успішну авторизацію вузла у приватній мережі. У колонці assigned ips видно, що вузол отримав IP-адресу 192.168.195.1/24 на інтерфейсі ztt6jx4lii`, що відповідає мережевому інтерфейсу ZeroTier.

Цей вивід підтверджує, що контейнер ZeroTier успішно запущено, він авторизований у приватній мережі, має активне з'єднання з глобальними вузлами, отримав IP-адресу з внутрішнього простору ZeroTier, і готовий до маршрутизації трафіку через логічну overlay-мережу.

3.3 Налаштування Ubuntu Linux

Налаштування ZeroTier-клієнта в Ubuntu Linux починається зі встановлення офіційного клієнта ZeroTier. Для дистрибутивів, що базуються на Debian і RPM ZeroTier пропонує універсальний скрипт встановлення, який автоматично додає відповідне репозиторій та встановлює необхідні пакети. Завдяки цьому інсталяція є простою, швидкою та не потребує ручної конфігурації джерел або залежностей. Найзручнішим способом інсталяції

ZeroTier One є використання однорядкової команди через curl, яка завантажує офіційний інсталяційний скрипт та передає його у bash з правами адміністратора `curl -s https://install.zerotier.com | sudo bash`.

Цей метод передбачає довіру до SSL-сертифікатів офіційного сайту ZeroTier і підходить для більшості користувачів, яким потрібна швидка та безпечна інсталяція без глибокого втручання. Після виконання команди скрипт:

– автоматично визначає тип операційної системи та її версію;

– додає відповідний репозиторій до системи;

– імпортує GPG-ключ;

– оновлює кеш пакетного менеджера;

– встановлює останню версію ZeroTier One як системну службу;

Після завершення інсталяції служба запускається автоматично (див. рисунок 3.6).

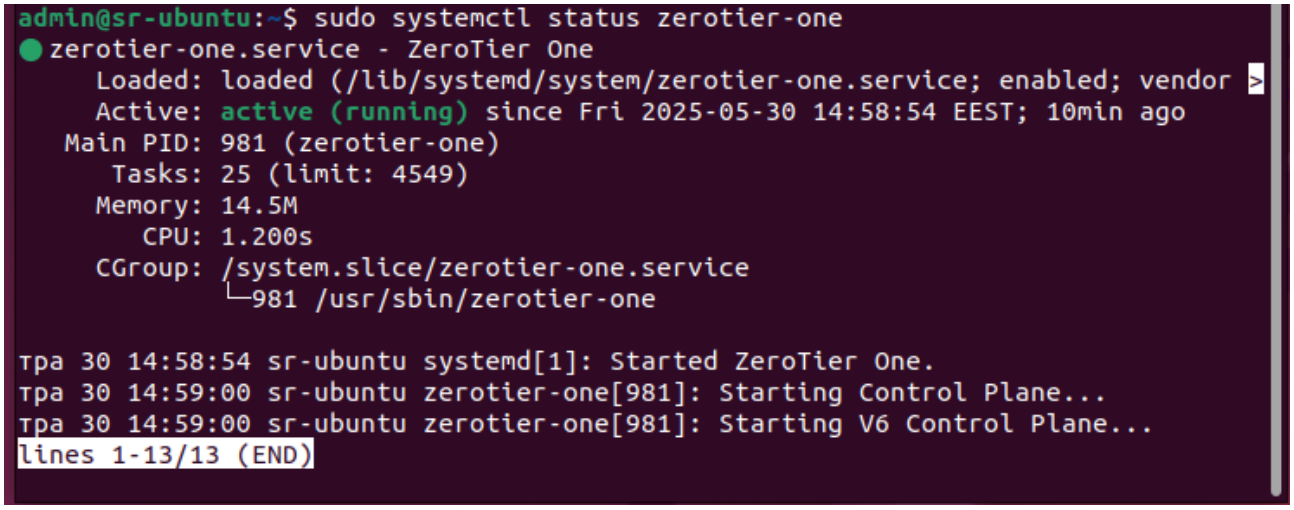

Рисунок 3.6 – Служба ZeroTier в Ubuntu Linux

На рисунку 3.7 представлено вивід консолі, у якій здійснено перевірку стану підключення ZeroTier та конфігурацію відповідного віртуального інтерфейсу.

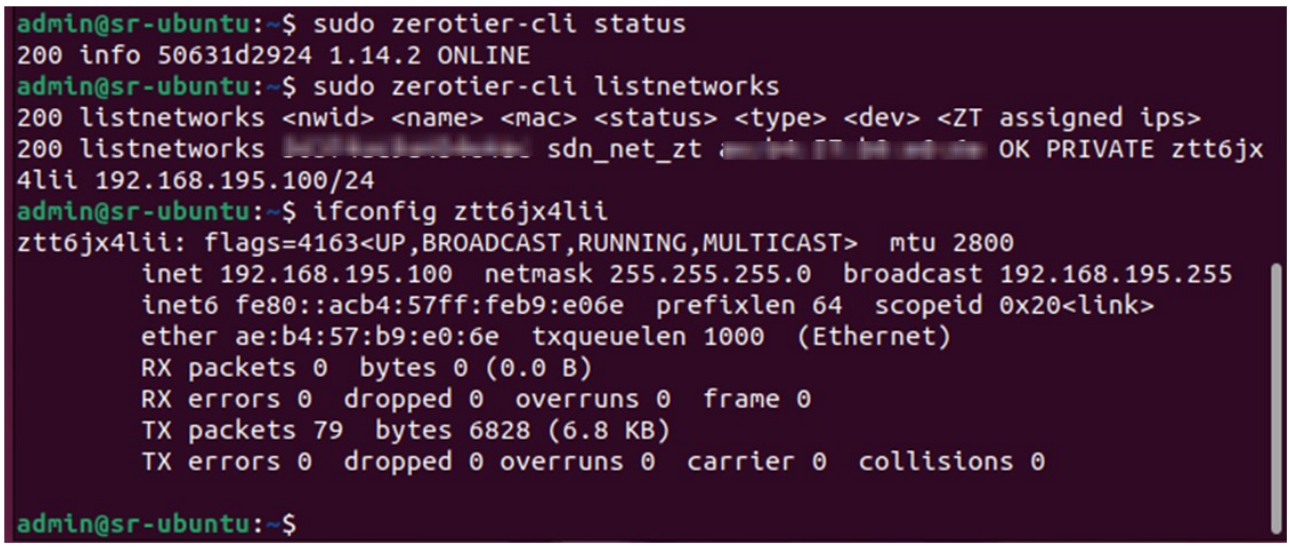

Рисунок 3.7 – Перевірка стану підключення ZeroTier ZeroTier в Ubuntu Linux

Після запуску команди `sudo zerotier-cli status` відображається статус ONLINE, що означає успішне підключення агента ZeroTier до глобальної peer-to-peer мережі та взаємодію з root-серверами ZeroTier. Далі виконується команда `sudo zerotier-cli listnetworks`, яка показує, що вузол успішно приєднаний до мережі з ідентифікатором з назвою sdn_net_zt. Показано MAC-адресу віртуального інтерфейсу, статус OK, тип мережі PRIVATE, інтерфейс ztt6jx4lii і присвоєна IP-адреса 192.168.195.100/24, що підтверджує отримання адреси з простору логічної мережі ZeroTier. Команда `ifconfig ztt6jx4lii` надає повну інформацію про створений ZeroTier-інтерфейс у системі. Інтерфейс має статус UP та RUNNING, що підтверджує його активність. Зазначена мережева конфігурація містить IP-адресу 192.168.195.100 з маскою підмережі 255.255.255.0.

На рисунку 3.8 показано таблицю маршрутизації.

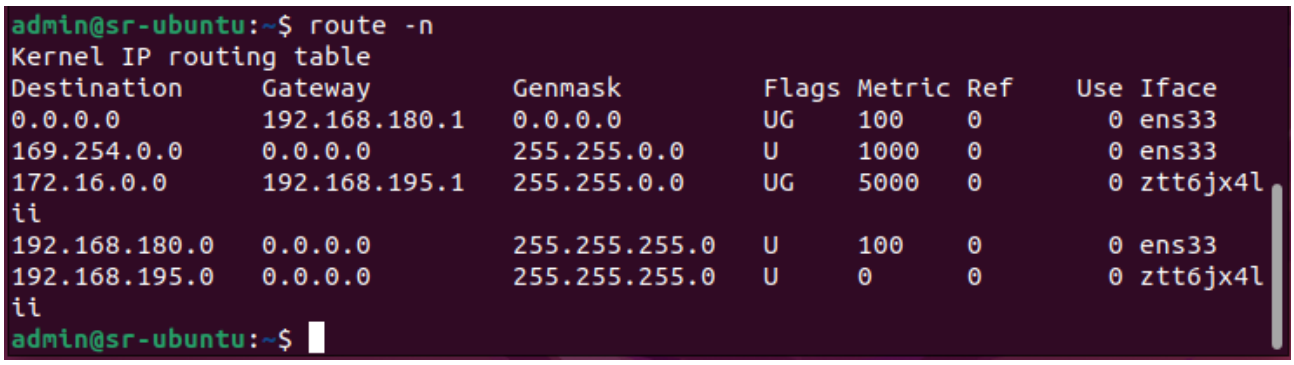

Рисунок 3.8 – Таблиця маршрутизації в Ubuntu Linux

Маршрут до підмережі 172.16.0.0/24 вказано через шлюз 192.168.195.1, який є IP-адресом ZeroTier-агента в контейнері MikroTik.

Згідно маршрутизації система Ubuntu Linux налаштована одночасно для доступу до локальної мережі через фізичний інтерфейс та до логічної SDN-мережі ZeroTier через віртуальний адаптер. Завдяки цьому вона може взаємодіяти як з традиційною інфраструктурою, так і з пристроями, які знаходяться в логічній ZeroTier-мережі.

3.4 Тестування мережі ZeroTier

Тестування мережі ZeroTier проводиться з метою перевірки працездатності з'єднання між вузлами, що належать до однієї логічної overlay-мережі, а також для підтвердження коректності маршрутизації, доступності сервісів і цілісності тунелю.

На рисунку 3.9 представлено вебінтерфейс ZeroTier Central, де відображається список учасників логічної мережі з назвою sdn_net_zt.

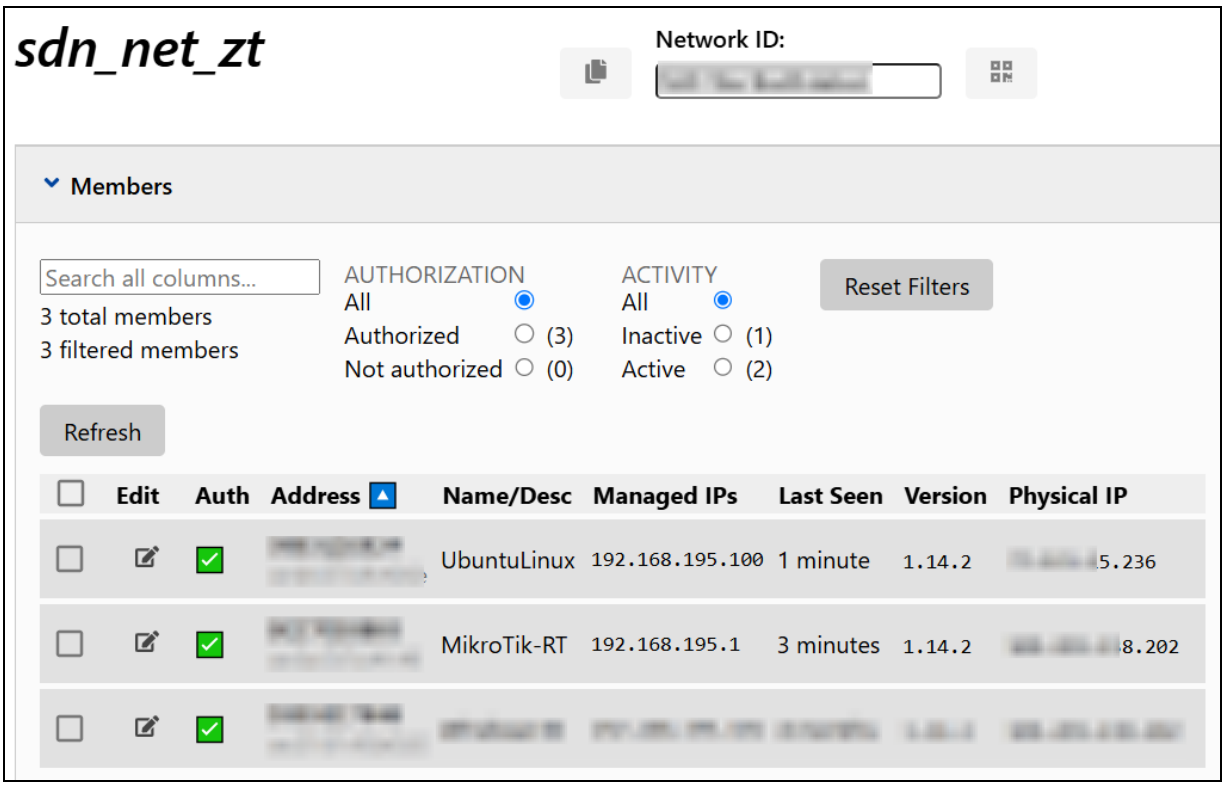

Рисунок 3.9 – Список учасників логічної мережі з назвою sdn_net_zt

Для кожного пристрою зазначено кілька параметрів: його криптографічна адреса, яка виступає унікальним ідентифікатором, логічне ім’я або опис, IP-адреса у просторі ZeroTier, а також інформація про останній момент активності. Поле Physical IP показує публічну IP-адресу, з якої пристрій підключається до ZeroTier-мережі. Візуальне представлення статусу вузлів дозволяє швидко оцінити, чи всі елементи інфраструктури присутні в мережі, чи мають вони призначені адреси та чи спостерігається якась активність.

На рисунку 3.10 представлено вікно термінала з інтерфейсом утиліти mtr (My Traceroute), яка виконує одночасне трасування маршруту і вимірювання параметрів затримки між джерелом і призначенням.

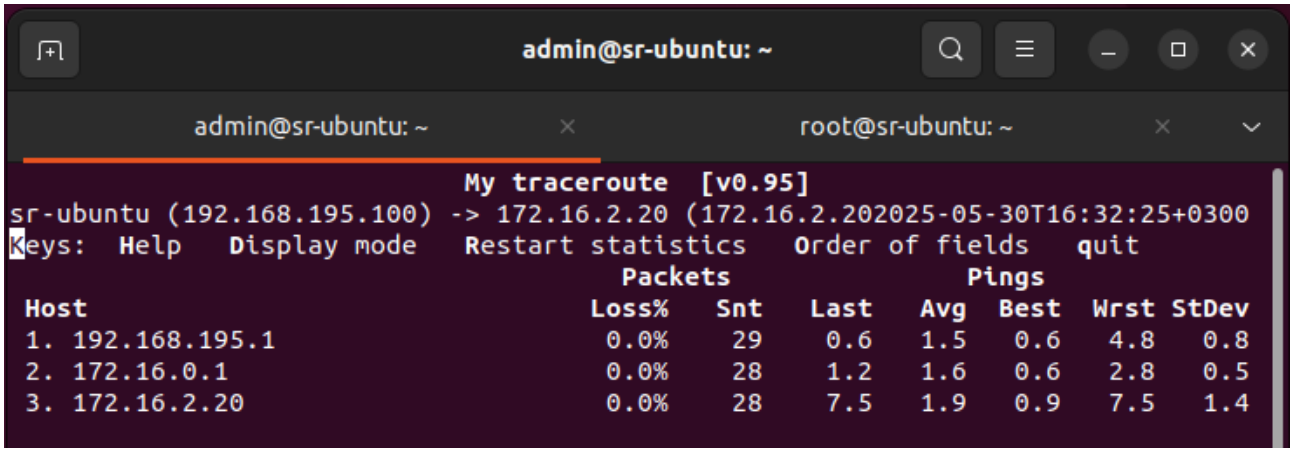

Рисунок 3.10 – Трасування маршруту до IP-адреси 172.16.2.20

У першому рядку показано IP-адресу першого вузла маршруту - 192.168.195.1. Це адреса ZeroTier-інтерфейсу маршрутизатора MikroTik, що виступає шлюзом між віртуальною мережею ZeroTier і локальними сегментами. Значення затримки в колонках Last, Avg, Best, Wrst та StdDev вказують на стабільну відповідь з мінімальними відхиленнями (середнє значення 1.5 мс). Другим стрибком у маршруті є 172.16.0.1 - IP-адреса bridge-інтерфейсу маршрутизатора MikroTik для контейнерного середовища, через який контейнер ZeroTier передає трафік до внутрішньої інфраструктури. Третій вузол - це сервер Windows Server 2022 з IP-адресом 172.16.2.20 до якого йде перевірка доступності. Час відгуку до цієї адреси складає близько 1.9 мс у середньому, що свідчить про коректну маршрутизацію через тунель ZeroTier, відсутність втрат пакетів і добру якість з’єднання. Усі три вузли у маршруті відповідають без втрат (0.0% Loss), що підтверджує стабільність і надійність логічної overlay-мережі. Затримки перебувають у межах мілісекунд, що є хорошим показником для peer-to-peer тунелів.

Перевірка доступу з Ubuntu Linux до Windows Server 2022 через протокол віддаленого робочого столу RDP демонструє практичну реалізацію безпечного з'єднання між географічно ізольованими вузлами без потреби у відкритті портів на фізичних маршрутизаторах чи використанні класичних VPN. Підключення до RD-сервісу здійснюється за допомогою графічної утиліти Remmina, яка підтримує параметри автентифікації, вибору роздільної златності та інших

параметрів. Завдяки попередньо встановленим маршрутами в конфігурації мережі ZeroTier, трафік з Ubuntu передається у зашифрованому вигляді безпосередньо до MikroTik, потім через внутрішню інфраструктуру до Windows Server 2022, де увімкнено службу Remote Desktop Services.

ПІсля успішного з’єднанні відкривається повноцінна сесія робочого столу, яка функціонує так, ніби обидва пристрої перебувають у локальній мережі (див. рисунок 3.11).

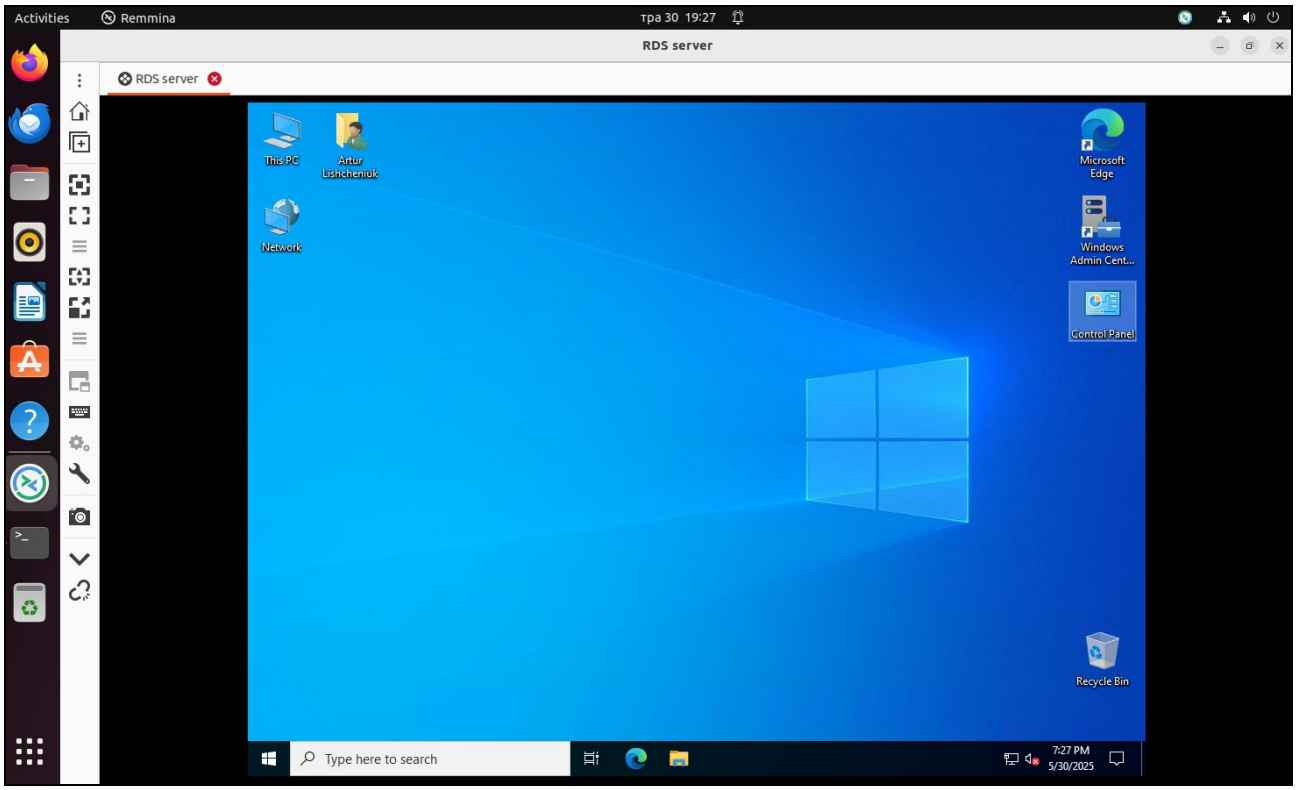

Рисунок 3.11 – Робочий стіл Windows Server 2022

Візуальна відповідь, взаємодія з інтерфейсом Windows, передача буфера обміну, файловий доступ і контроль над сесією відбуваються без помітних затримок, що свідчить про стабільність тунелю ZeroTier. Така перевірка є останнім кроком для підтвердження коректності конфігурації всієї SDN-мережі.

3.5 Висновок до третього розділу

В третьому розділі було розглянуто процедуру створення, налаштування та тестування SDN-мережі на основі ZeroTier у тестовому середовищі.

Початковим етапом було створення логічної overlay-мережі через хмарну платформу ZeroTier Central, де був сформований унікальний Network ID та IP-підмережа, що стала основою для адресації всіх учасників. Завдяки використанню приватного режиму доступу була забезпечена централізована авторизація кожного підключеного вузла, що підвищує рівень безпеки мережі. Наступним етапом було розгортання контейнера ZeroTier безпосередньо в маршрутизаторі MikroTik CHR. Це дозволило інтегрувати маршрутизатор у логічну мережу. У контейнері агент ZeroTier отримав IP-адресу в просторі логічної мережі та виконував функцію шлюзу до внутрішніх підмереж. Через інтерфейси eth0 і ztt6jx4lii забезпечувалося повноцінна маршрутизація між фізичними сегментами та віртуальним тунелем. Було описано підключення клієнта Ubuntu Linux до мережі ZeroTier. Після приєднання до мережі клієнт отримав IP-адресу та маршрут до внутрішньої інфраструктури через шлюз MikroTik. Аналіз інтерфейсів, таблиці маршрутизації та стану служби підтвердив правильну конфігурацію. Фінальний етап був присвячений тестуванню мережевої взаємодії. Засобами ZeroTier Central було підтверджено наявність усіх вузлів в активному стані. За допомогою утиліти mtr було здійснено трасування до внутрішньої IP-адреси, що продемонструвало стабільність з’єднання, мінімальні затримки та відсутність втрат пакетів. Успішне підключення до Windows Server 2022 через RDP з Ubuntu Linux стало практичним підтвердженням функціональності тунелю, правильності маршрутизації та логіки доступу.

Результати тестування підтверджують, що архітектура SDN-мережі з використанням ZeroTier може бути ефективно використана як у дослідницьких, так і в корпоративних цілях для реалізації ізольованих та керованих мережевих середовищ.

РОЗДІЛ 4. БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ

4.1 Фактори ризику і можливі порушення здоров’я користувачів комп’ютерної мережі

Велика кількість людей проводить багато часу за комп’ютером, що збільшує ризики та можливі негативні наслідки для здоров’я. Довге сидіння за комп’ютером призводить до різних проблем із здоров’ям. Найпоширенішими причин погіршення стану здоров’я є порушення постави та біль у спині [34]. Неправильна постава, яка зумовлена внаслідок погано організованого робочого місця або неправильно підібраного комп’ютерного стільця, це може спричинити хронічні болі у спині, шиї та плечах. Також поширена проблема із зап’ястям та кистями, наприклад синдром зап’ястного каналу, який виникає через повторюванні рухи, наприклад використання миші чи клавіатури [34]. Ще одним серйозним ризиком, пов’язаним із тривалим використанням комп’ютера, є проблеми зі здоров’ям очей. Проведення надто тривалого часу перед екраном може призвести до таких симптомів, як погіршенням зору або напруження очей, яке характеризується сухістю, свербінням, подразненням і втомою очей. Робота за комп'ютером часто пов'язана з високим рівнем стресу, особливо в умовах коли терміни стислі, а вимоги високі. Це може призвести до вигоряння, депресії та інших психологічних проблем. Крім того, ізоляція, яка часто супроводжує роботу вдома або в невеликих кабінетах, може вплинути на соціальне та емоційне самопочуття.

Також важливо розглянути вплив на якість сну та загальне фізичне здоров'я. Багато користувачів комп'ютерів скаржаться на порушення сну, що може бути пов'язано з надмірним впливом на світлові екрани перед сном. Це світло пригнічує вироблення мелатоніну, гормону сну, заважаючи людям засинати та підтримувати здоровий цикл сну. Використання спеціальних програм або окулярів для блокування синього світла є важливим для тих, хто регулярно працює з комп'ютерами ввечері. Фізична неактивність, яка часто

супроводжує довготривалу роботу за комп'ютером, також є фактором ризику для ряду хронічних захворювань, серцево-судинні захворювання та діабет другого типу. Недостатня фізична активність призводить до уповільнення метаболізму, що може спричинити набір ваги і зниження загальної фізичної форми. Це підвищує ризик розвитку серцевих захворювань.

Залежність від Інтернету та соціальних медіа є ще однією проблемою, з якою стикаються користувачі комп'ютерних мереж. Ця залежність може призвести до зниження продуктивності, соціальної ізоляції та погіршення психічного здоров'я. Важливо встановлювати межі та регулярно проводити час без використання електронних пристроїв.

Для зменшення цих ризиків існують різні стратегії. По-перше, важливо правильно організувати робоче місце. Ергономічні крісла, належно розташовані монітори та клавіатури, підставка для ніг, регулярні перерви для руху та розтяжки можуть значно знизити ризик мускул скелетних проблем. Крім того, використання спеціальних окулярів або застосування програм, що знижують синє світло від екранів, можуть допомогти зменшити втому очей.

Що стосується психологічного здоров'я, важливо забезпечити баланс між роботою та особистим життям, включаючи регулярні перерви, хобі, соціальну взаємодію та вправи на свіжому повітрі. Також корисною може бути практика медитації для зниження рівня стресу.

4.2 Оцінка стійкості роботи промислового підприємства до дії світлового випромінювання ядерного вибуху

Основну небезпеку для наземних об'єктів, зокрема і телекомунікаційних, становлять ударна хвиля, світлове (теплове) випромінювання, вторинні вражаючі фактори і радіоактивне зараження місцевості. Проте іноді доводиться враховувати і вплив проникаючої радіації та електромагнітного імпульсу.

Критеріями оцінки фізичної стійкості об'єкта прийняті: при впливі ударної хвилі - надлишкові тиски, при яких елементи виробничого комплексу

не руйнуються або одержують такі ушкодження чи руйнування (слабкі і середні), при яких вони можуть бути відновлені в короткі терміни; при впливі світлового випромінювання - максимальні значення світлових імпульсів, при яких не відбувається загоряння матеріалів, сировини, устаткування, будинків і споруд; при впливі вторинних факторів - надлишкові тиски, при яких руйнування і пошкодження не призводять до аварій, пожеж, вибухів, затоплень, небезпечного зараження місцевості й атмосфери, тобто не призводять до ураження людей і виходу з ладу засобів виробництва [35].

Оцінка стійкості об'єкта включає визначення: видів вражаючих факторів, вплив яких можливий на об'єкт, та їх параметрів; впливу ударної хвилі на елементи об'єкта; можливості виникнення пожеж; впливу вторинних вражаючих факторів.

Світлове випромінювання - це електромагнітне випромінювання, основним джерелом якого є світна область вибуху (вогненна куля), що складається з розпечених продуктів вибуху і повітря. Температура в ній сягає від 6 тисяч градусів за С. Тривалість світіння залежить від потужності ядерного заряду: при вибуху малого калібру - 1-2 сек., середнього - 2-4 сек, крупного та надкрупного - 10 і більше сек. На світлове випромінювання припадає приблизно 30 % всієї енергії ядерного вибуху. Воно складається з ультрафіолетових, інфрачервоних і видимих променів. Основна кількість енергії світлового випромінювання (85%) виділяється в перші секунди з моменту вибуху. Кількість енергії світлового випромінювання, яке падає на 1 см 2 поверхні, перпендикулярної напрямку його поширення, за весь час світіння, називається світловим імпульсом. Його величина вимірюється в джоулях на квадратний метр (Дж/м2).

Уражуюча дія світлового випромінювання вимірюється, головним чином, величиною світлового імпульсу і часом дії. Чим більша величина світлового імпульсу, що випромінюється за менший час, тим сильніший уражуючий ефект, який пропорційний поглинутій кількості енергії. Остання перетворюється в тепло і здатна викликати опіки та приводити до спалахування різних предметів.

Ураження людини можливе, як в результаті безпосередньої дії світлового випромінювання на шкірні покриви - світлові (первинні) опіки, так і в результаті спалахування одягу і навколишніх предметів - опосередковані (вторинні) опіки. Можливість виникнення пожеж встановлюють за займистістю матеріалів від світлового імпульсу ядерного вибуху, руйнування печей, газопроводів, пошкодження електромережі, які можуть виникнути при аваріях, землетрусах, бурях та ін.

При оцінці стійкості об'єкта проти світлового випромінювання ядерного вибуху необхідно визначити максимальне значення світлового імпульсу яке може бути на об'єкті. Світловий імпульс можна розрахувати за температурою загорання або нагрівання матеріалів і виробів. Для оцінки стійкості об'єкта проти світлового випромінювання необхідні такі вихідні дані: характеристика будівель і споруд, характер виробництва, які горючі матеріали застосовуються у виробництві; вид готової продукції та місце її зберігання.

Оцінку стійкості електроенергетичного об'єкта до світлового випромінювання ядерного вибуху доцільно проводити у такій послідовності:

- визначити мінімальну відстань до можливого центру вибуху Rx, км;
- визначити максимальне значення світлового імпульсу I св.max, кДж/м2;
- визначити ступінь вогнестійкості будинку цеху;
- виявити в конструкціях будівлі елементи, виготовлені із горючих матеріалів та визначити їх характеристики (наприклад: для дверей та віконних рам – дерев’яних, пофарбованих в темний колір I св.max =250, кДж/м2);
- визначити границю стійкості об’єкту до світлового випромінювання за мінімальним світловим імпульсом, що викликає спалахування в будівлі I св.lim. кДж/м2;
- здійснити порівняння та дати оцінку стійкості об’єкту: при I св.lim < I св.max об’єкт не стійкий до світлового випромінювання, при I св.lim > I св.max - стійкий;

- визначити ступінь руйнування будівлі від ударної хвилі при максимальному очікуваному надлишковому тиску;
- визначити зону пожеж, в якій може опинитися об’єкт.

На основі отриманих даних робиться відповідний висновок щодо стійкості електроенергетичного об'єкта до світлового випромінювання ядерного вибуху:

- чи викликає складну пожежну ситуацію на об’єкті при ядерному вибуху заданої потужності очікуваний максимальний світловий імпульс (кДж/м2) та надлишковий тиск ударної хвилі (кПа);
- чи опиниться об’єкт в зоні суцільних пожеж;
- чи об’єкт стійкий до світлового випромінювання за границею стійкості;
- що із обладнання, конструктивних елементів будівлі складає пожежну небезпеку для об’єкту;
- робитися висновок про доцільність підвищення границі стійкості об’єкту.

Підвищити границю стійкості можливо шляхом виконання наступних заходів: замінити крівлю об’єкту на азбестоцементові; замінити дерев’яні віконні рами на металеві; набити на двері сталлю по азбестовій прокладці; провести на об’єкті профілактичні протипожежні заходи.

4.3 Висновок до четвертого розділу

В четвертому розділі було проаналізовано аспекти безпеки життєдіяльності, пов’язані з використанням комп’ютерних мереж, а також оцінено потенційну стійкість об’єктів до світлового випромінювання ядерного вибуху. Перший підрозділ розкрив ризики для здоров’я користувачів комп’ютерів, зумовлені тривалим перебуванням за екраном. Було висвітлено проблеми опорно-рухового апарату, зору, психоемоційного стану та сну. Зокрема, акцент зроблено на наслідках неправильно організованого робочого

місця та малорухомого способу життя. Встановлено, що ці фактори можуть спричинити хронічні захворювання, порушення сну, підвищений рівень стресу, соціальну ізоляцію, а також психоемоційне виснаження. Водночас запропоновано низку ефективних профілактичних заходів, спрямованих на покращення умов праці, ергономіку, гігієну зору та психофізіологічний баланс, що дозволяє знизити вплив шкідливих чинників. У другому підрозділі було розглянуто оцінку стійкості інфраструктурних об'єктів до дії світлового випромінювання при ядерному вибуху. Детально описано природу світлового імпульсу, його характеристики, тривалість, інтенсивність і механізм ураження. Основна увага приділена методиці визначення потенційної загрози для об’єкта. Зроблено акцент на критеріях вразливості та необхідності підвищення межі стійкості шляхом технічної модернізації конструкцій і впровадження протипожежних заходів.

ВИСНОВКИ

Під час виконання кваліфікаційної роботи бакалавра було досліджено концепцію програмно-визначених мереж і на практиці показано побудову повноцінної SDN-інфраструктури за допомогою платформи ZeroTier у поєднанні з віртуальним маршрутизатором MikroTik CHR. Опрацьований теоретичний і практичний матеріал дав змогу переконливо показати, що сучасні мережі здатні виходити за межі традиційної парадигми, де логіка керування жорстко прив’язана до апаратного забезпечення, і переходити до моделей з централізованим, програмним управлінням, яке забезпечує більшу гнучкість, кращу видимість та гетерогенну інтеграцію.

У першому розділі було висвітлено історичний розвиток мережевих архітектур: від статичних, ручних конфігурацій до появи ідей SDN, що розділяють площину передачі й площину керування. На цьому тлі детально розібрано механізми, завдяки яким SDN усуває обмеження класичного підходу, а саме логічна централізація спрощує впровадження нових політик без перепрограмування кожного комутатора окремо, стандартизовані southbound-інтерфейси спрощують інтеграцію обладнання різних виробників, а northbound-API відкриває дорогу для автоматизації та зв’язку мережі з додатками в реальному часі. Цей теоретичний фундамент сформував рамку, у межах якої надалі оцінювалася життєздатність конкретного рішення на базі ZeroTier.

Другий розділ був присвячений аналізові ZeroTier - платформи, що поєднує принципи SDN із можливостями VPN та SD-WAN і водночас залишає користувачеві мінімальний поріг входження. Ретельно описано, як ZeroTier реалізує двошарову модель VL1/VL2, чому використання Curve25519/Ed25519 та Salsa20/Poly1305 забезпечує криптостійкість і конфіденційність, у чому полягає перевага криптографічних ідентифікаторів над класичною IP-адресацією. На основі цих положень було створено лабораторне середовище в Hyper-V, де віртуальний MikroTik із контейнерним агентом ZeroTier став мостом між фізичними й overlay-мережами, а Windows Server 2022 та Ubuntu

Linux відіграли ролі, відповідно, захищеного ресурсу в DMZ і клієнта, що під’єднується ззовні. Докладно описано параметри кожного інтерфейсу, адресні простори, маршрути та правила брандмауера, що робить тестове середовище відтворюваним для сторонніх дослідників і адміністраторів.

У третьому розділі вся збудована система була приведена в дію. У ZeroTier Central створено приватну мережу, вручну авторизовано вузли, визначено ACL і керовані маршрути. За допомогою утиліт mtr виміряно затримки, які становили приблизно 1–2 мс без втрат пакетів, що засвідчує ефективність peer-to-peer маршрутизації та відсутність потреби у централізованих шлюзах. Кульмінацією стала успішна RDP-сесія до Windows Server 2022: віддалений робочий стіл функціонував плавно, що в реальному середовищі означає можливість безпечної експлуатації сервісів, котрі традиційно виносилися у DMZ.

Узагальнюючи отримані результати, можна стверджувати, що зв’язка ZeroTier + MikroTik CHR не просто замінює класичні VPN, а пропонує якісно інший рівень керованості. Мережа стає “програмним об’єктом”, який легко масштабувати скриптами або запитами REST-API, а адміністратор отримує єдину точку зору на топологію, політики та телеметрію. Виконана робота продемонструвала конкретний шлях впровадження SDN мереж без значних витрат на обладнання. Отримані знання та практичні напрацювання можуть бути використані для побудови безпечного віддаленого доступу, обєднання офісних сегментів або швидкого розгортання тестових середовищ, де досліджуються сучасні кіберзагрози й методи їхнього виявлення. У підсумку проєкт показав, що SDN на базі ZeroTier і MikroTik CHR є зрілим, масштабованим і економічно доцільним рішенням.

ПЕРЕЛІК ДЖЕРЕЛ

1. А.Г. Микитишин, М.М. Митник, П.Д. Стухляк, В.В. Пасічник Комп'ютерні мережі. Книга 1. [навчальний посібник] - Львів, "Магнолія 2006", 2013. – 256 с.

2. А.Г. Микитишин, М.М. Митник, П.Д. Стухляк, В.В. Пасічник Комп'ютерні мережі. Книга 2. [навчальний посібник] (Лист МОНУ №1/11-11650 від 16.07.12р.) - Львів, "Магнолія 2006", 2014. – 312 с.

3. GeeksforGeeks. (2020, October 31). Evolution of Network Architecture - GeeksforGeeks. https://www.geeksforgeeks.org/evolution-of-network-architecture/

4. What is Software-Defined Networking (SDN)? (n.d.). VMware by Broadcom - Cloud Computing for the Enterprise. https://www.vmware.com/topics/software-defined-networking

5. GeeksforGeeks. (2019, July 2). What is software defined networking (SDN)? - geeksforgeeks. https://www.geeksforgeeks.org/software-defined-networking/

6. Software-Defined networking (SDN) definition. (n.d.). Cisco. https://www.cisco.com/c/en/us/solutions/software-defined-networking/overview.html

7. Rosencrance, L., English, J., & Burke, J. (2022, May 11). What is software-defined networking (SDN)? Definition from techtarget.com. Search Networking. https://www.techtarget.com/searchnetworking/definition/software-defined-networking-SDN

8. Software-Defined Networking (SDN) Explained in 5 Minutes or Less. (n.d.). Geekflare. https://geekflare.com/cloud/software-defined-networking/

9. Tymoshchuk, V., Mykhailovskyi, O., Dolinskyi, A., Orlovska, A., & Tymoshchuk, D. (2024). OPTIMISING IPS RULES FOR EFFECTIVE DETECTION OF MULTI-VECTOR DDOS ATTACKS. Матеріали конференцій МЦНД, (22.11. 2024; Біла Церква, Україна), 295-300.

10. Tymoshchuk, V., Vantsa, V., Karnaukhov, A., Orlovska, A., & Tymoshchuk, D. (2024). COMPARATIVE ANALYSIS OF INTRUSION

DETECTION APPROACHES BASED ON SIGNATURES AND ANOMALIES. Матеріали конференцій МЦНД, (29.11. 2024; Житомир, Україна), 328-332.

11. Tymoshchuk, V., Pakhoda, V., Dolinskyi, A., Karnaukhov, A., & Tymoshchuk, D. (2024). MODELLING CYBER THREATS AND EVALUATING THE PERFORMANCE OF INTRUSION DETECTION SYSTEMS. Grail of Science, (46), 636–641. https://doi.org/10.36074/grail-of-science.29.11.2024.081

12. Awati, R., & Wright, G. (2025, April 2). What is a northbound interface/southbound interface? | Definition from TechTarget. WhatIs. https://www.techtarget.com/whatis/definition/northbound-interface-southbound-interface

13. SDN North-bound and South-bound APIs and Interfaces. (2020, January 12). ComputerNetworkingNotes. https://www.computernetworkingnotes.com/ccna-study-guide/sdn-north-bound-and-south-bound-apis-and-interfaces.html

14. How ZeroTier Works | ZeroTier Documentation. (n.d.). Getting Started with ZeroTier | ZeroTier Documentation. https://docs.zerotier.com/zerotier/

15. Tymoshchuk, V., Karnaukhov, A., & Tymoshchuk, D. (2024). USING VPN TECHNOLOGY TO CREATE SECURE CORPORATE NETWORKS. Collection of scientific papers «ΛΌГОΣ», (June 21, 2024; Seoul, South Korea), 166-170.

16. Karnaukhov, A., Tymoshchuk, V., Orlovska, A., & Tymoshchuk, D. (2024). USE OF AUTHENTICATED AES-GCM ENCRYPTION IN VPN. Матеріали конференцій МЦНД, (14.06. 2024; Суми, Україна), 191-193.

17. The protocol | zerotier documentation. (n.d.). Getting Started with ZeroTier | ZeroTier Documentation. https://docs.zerotier.com/protocol

18. Rules engine. (n.d.). ZeroTier Knowledge Base. https://zerotier.crisp.help/en/article/rules-engine-kradl4/

19. Security. (n.d.). ZeroTier Knowledge Base. https://zerotier.crisp.help/en/article/security-m9wi8s/

20. ТИМОЩУК, Д., & ЯЦКІВ, В. (2024). USING HYPERVISORS TO CREATE A CYBER POLYGON. MEASURING AND COMPUTING DEVICES IN TECHNOLOGICAL PROCESSES, (3), 52-56.

21. Тимощук, В., Долінський, А., & Тимощук, Д. (2024). СИСТЕМА ЗМЕНШЕННЯ ВПЛИВУ DOS-АТАК НА ОСНОВІ MIKROTIK. Матеріали конференцій МЦНД, (17.05. 2024; Ужгород, Україна), 198-200

22. Тимощук, В., Долінський, А., & Тимощук, Д. (2024). ЗАСТОСУВАННЯ ГІПЕРВІЗОРІВ ПЕРШОГО ТИПУ ДЛЯ СТВОРЕННЯ ЗАХИЩЕНОЇ ІТ-ІНФРАСТРУКТУРИ. Матеріали конференцій МЦНД, (24.05. 2024; Запоріжжя, Україна), 145-146. https://doi.org/10.62731/mcnd-24.05.2024.001

23. ТИМОЩУК, Д., ЯЦКІВ, В., ТИМОЩУК, В., & ЯЦКІВ, Н. (2024). INTERACTIVE CYBERSECURITY TRAINING SYSTEM BASED ON SIMULATION ENVIRONMENTS. MEASURING AND COMPUTING DEVICES IN TECHNOLOGICAL PROCESSES, (4), 215-220.

24. Hyper-V technology overview. (n.d.). Microsoft Learn: Build skills that open doors in your career. https://learn.microsoft.com/en-us/windows-server/virtualization/hyper-v/hyper-v-overview?pivots=windows

25. User Manuals - User manuals - MikroTik Documentation. (n.d.). MikroTik Routers and Wireless - Support. https://help.mikrotik.com/docs/spaces/UM/pages/8978698/User+Manuals

26. Cloud Hosted Router, CHR - RouterOS - MikroTik Documentation. (n.d.). MikroTik Routers and Wireless - Support. https://help.mikrotik.com/docs/spaces/ROS/pages/18350234/Cloud+Hosted+Router+CHR

27. RouterOS - RouterOS - MikroTik Documentation. (n.d.). MikroTik Routers and Wireless - Support. https://help.mikrotik.com/docs/spaces/ROS/pages/328059/RouterOS

28. Windows Server documentation. (n.d.). Microsoft Learn: Build skills that open doors in your career. https://learn.microsoft.com/en-us/windows-server/

29. Remote Desktop Services overview in Windows Server. (n.d.-b). Microsoft Learn: Build skills that open doors in your career. https://learn.microsoft.com/en-us/windows-server/remote/remote-desktop-services/remote-desktop-services-overview

30. Official Ubuntu Documentation. (n.d.). Official Ubuntu Documentation. https://help.ubuntu.com/

31. Remmina remote desktop client. (n.d.). Remmina. https://remmina.org/

32. ZeroTier Central. (n.d.). ZeroTier Central. https://my.zerotier.com/

33. Container - RouterOS - MikroTik Documentation. (n.d.). MikroTik Routers and Wireless - Support. https://help.mikrotik.com/docs/spaces/ROS/pages/84901929/Container

34. Шкідливий вплив персонального комп'ютера на здоров'я людини. URL: https://ir.lib.vntu.edu.ua/bitstream/handle/123456789/21370/5363.pdf

35. Стручок В.С. Техноекологія та цивільна безпека. Частина «Цивільна безпека». Навчальний посібник. Тернопіль: ТНТУ. 2022. 150 с.