МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний університет кораблебудування імені адмірала Макарова

Навчально-науковий інститут автоматики і електротехніки

Кафедра комп'ютерних технологій та інформаційної безпеки

ЗАТВЕРДЖУЮ
Завідувач кафедри комп'ютерних технологій та інформаційної безпеки, к.т.н., доцент
_____С.М. Нужний
«18» 06 2025 р.

Кваліфікаційна робота
на правах рукопису

КВАЛІФІКАЦІЙНА РОБОТА
ВІРТУАЛЬНА ПРИВАТНА МЕРЕЖА ДЛЯ ОБ'ЄДНАННЯ ПРИСТРОЇВ ЗА NAT

Ступінь вищої освіти: БАКАЛАВР
Галузь знань: 12 «Інформаційні технології»
Спеціальність: 125 «Кібербезпека та захист інформації»
Освітньо-професійна програма: Системи технічного захисту інформації, автоматизація її обробки

Виконав: студент 4 курсу групи 4387ст
Власов Дмитро Олександрович

Кваліфікаційна робота містить результати самостійного виконання навчального завдання. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело

Власов Д.О.

Керівник: старший викладач кафедри комп'ютерних технологій та інформаційної безпеки **Трибулькевич Сергій Леонідович** _____

Миколаїв – 2025

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний університет кораблебудування імені адмірала Макарова

Навчально-науковий інститут автоматики і електротехніки

ЗАТВЕРДЖУЮ
Гарант освітньої програми, к.т.н., доцент, завідувач кафедри КТІБ
_____ С.М. Нужний
«_17_» ____06____ 2025 р.

ЗАВДАННЯ

на кваліфікаційну роботу

Студент: Власов Дмитро Олександрович

Курс:4

Група: 4387ст

Ступінь вищої освіти: бакалавр

Галузь знань: 12 Інформаційні технології

Спеціальність: 125 «Кібербезпека та захист інформації»

Освітньо-професійна програма: Системи технічного захисту інформації, автоматизація її обробки

1. Тема роботи: «Віртуальна приватна мережа для об'єднання пристроїв за NAT»

затверджена наказом НУК від «_23__» _квітня_____2025 р. № УЧ- 343

2. Вихідні дані до роботи: транспортний протокол – TCP; шифрування – AES; VPN – OpenVPN, L2TP IPSec, SSTP.

Призначення та область застосування розроблюваного виробу; 2. Технічна характеристика; 3. Опис та обґрунтування вибраної конструкції; 4. Розрахунки, підтверджуючі працездатність і надійність конструкції.

4. Терміни виконання завдання:
термін видачі завдання: «_03_» _лютого_ 2025 р.
термін подання роботи: «18 » _червня_ 2025 р.

6. План-графік виконання кваліфікаційної роботи

№ п/п	Заходи	Дата		Готовність	Відмітка про виконання
		Навч. тиждень	Контрольна дата		
1.	Організаційні збори.	23	03.02.2025	Вибір теми, визначення мети, завдань, об'єкта та предмета дослідження	виконано
2.	Організаційні збори	25	20.02.2025	Попередній варіант оглядових розділів, підбір і опрацювання списку використаних джерел	виконано
3.	Рубіжний контроль	31	03.04.2025	Попередній варіант повної КР.	виконано
4.	Рубіжний контроль	33	20.04.2025	Попередній варіант презентації.	виконано
5.	ЄДКІ на рівень «бакалавр»	34	29.04.2025	Здавання ЄДКІ зі спеціальності на ступінь бакалавра	виконано
6.	Перевірка на плагіат	42	14.06.2025	1. Електронний варіант кваліфікаційної роботи; 2. Подання КР на перевірку на плагіат.	виконано
7.	Кафедральний захист	43	18.06.2025	1. Оформлена КР (в форматі pdf) (передається студентом на кафедрі для внутрішньої рецензії, висновок керівника). У разі отримання позитивної внутрішньої рецензії, КР подається на зовнішню рецензію. 2. Оформлена презентація (в форматі pdf).	виконано
8.	Подання документів на захист	44	24.06.2025	1. Подання щодо захисту КР 2. Зовнішня рецензія (окремо від КР). Резюме на КР. 3. Висновок кафедри про КР, допуск до захисту; 4. Електронний варіант та роздрукована презентація в кількості 5 примірників. 5. Електронний варіант КР на диску	виконано
9.	Захист КР	44	25.06.2025	1. Приєднання студента до засідання кваліфікаційної комісії (конференції) у встановлений час для проведення процедури дистанційного захисту 2. Процедура захисту КР.	виконано

ПРИМІТКА: Завдання додається до виконаної роботи та подається до атестаційної комісії.

Керівник
Старший викладач кафедри комп'ютерних технологій
та інформаційної безпеки _____ С.Л. Трибулькевич

Студент _____ Д.О. Власов

АНОТАЦІЯ

Власов Д.О. Віртуальна приватна мережа для об’єднання пристроїв за NAT

Кваліфікаційна робота на здобуття ступеня вищої освіти «бакалавр» за спеціальністю F5 «Кібербезпека та захист інформації» галузі знань F «Інформаційні технології». – Національний університет кораблебудування імені адмірала Макарова, Миколаїв, 2025.

Пояснювальна записка: 62 с., 39 рис., 7 посилань.

У бакалаврській кваліфікаційній роботі виконується розробка та реалізація віртуальної приватної мережі для об’єднання пристроїв за NAT.

Актуальність теми.

У сучасних умовах активного розвитку IoT-технологій та розподілених систем зростає потреба у надійній та безпечній взаємодії пристроїв, розташованих у різних локальних мережах за NAT. Традиційні методи доступу до таких пристроїв не забезпечують належного рівня безпеки та гнучкості. Створення захищеного каналу з використанням VPN-технологій дозволяє вирішити ці проблеми, забезпечуючи конфіденційність, цілісність і доступність даних. Особливо актуальним є використання хмарних рішень, таких як Oracle Cloud, для розгортання VPN-серверів як централізованих посередників.

Мета дипломного проекту.

Розробка безпечної віртуальної приватної мережі (VPN) для об’єднання пристроїв, які знаходяться за NAT, з використанням OpenVPN-сервера, розгорнутого в хмарній інфраструктурі Oracle Cloud, з метою забезпечення конфіденційного, захищеного та стабільного з’єднання між учасниками мережі.

Завдання дипломного проекту:

Аналіз існуючих рішень щодо об’єднання пристроїв за NAT.

Обґрунтування вибору OpenVPN як основного інструменту створення VPN-з'єднання.

Розгортання та налаштування VPN-сервера у хмарі Oracle Cloud.

Розробка та тестування клієнтських конфігурацій для різних типів пристроїв.

Забезпечення криптографічного захисту переданих даних відповідно до вимог інформаційної безпеки.

Оцінка ефективності, стійкості та продуктивності побудованої VPN-мережі.

Очікувані результати:

– Функціональна та захищена VPN-мережа, яка забезпечує надійний зв’язок між пристроями за NAT;

– Підвищення рівня захисту переданої інформації;

– Документовані інструкції з розгортання та експлуатації системи;

– Практичні рекомендації щодо використання хмарної інфраструктури для потреб кібербезпеки.

Ключові слова: VPN, OpenVPN, NAT traversal, кібербезпека, Oracle Cloud, хмарні технології, інформаційна безпека, тунелювання, криптографія, віддалений доступ.

SUMMARY

Vlasov D.O.

Virtual Private Network for Connecting Devices Behind NAT

Bachelor's Qualification Thesis for the degree of Bachelor of Higher Education in specialty F5 "Cybersecurity and Information Protection", field of knowledge F "Information Technology" – Admiral Makarov National University of Shipbuilding,

Mykolaiv, 2025.

Explanatory Note: 62 pages, 39 figures, 7 references.

This bachelor's thesis involves the design and implementation of a virtual private network for connecting devices located behind NAT.

Relevance of the topic

In the current context of rapid development of IoT technologies and distributed systems, the demand for secure and reliable interaction between devices located in different local networks behind NAT is increasing. Traditional access methods to such devices do not ensure an adequate level of security and flexibility. The creation of a secure communication channel using VPN technologies helps solve these issues by providing confidentiality, integrity, and availability of data. The use of cloud-based solutions, such as Oracle Cloud, for deploying VPN servers as centralized intermediaries is especially relevant.

Objective of the thesis project

To develop a secure virtual private network (VPN) for connecting devices behind NAT using an OpenVPN server deployed in Oracle Cloud infrastructure, in order to ensure a confidential, protected, and stable connection between network participants.

Tasks of the thesis project:

Analysis of existing solutions for connecting devices behind NAT;

Justification for selecting OpenVPN as the main tool for establishing a VPN connection;

Deployment and configuration of a VPN server in Oracle Cloud;

Development and testing of client configurations for various types of devices;

Implementation of cryptographic protection for transmitted data in accordance with information security requirements;

Evaluation of the effectiveness, resilience, and performance of the constructed VPN network.

Expected results:

A functional and secure VPN network that provides reliable communication between devices behind NAT;

An increased level of protection for transmitted data;

Documented instructions for system deployment and operation;

Practical recommendations for using cloud infrastructure for cybersecurity needs.

Keywords: VPN, OpenVPN, NAT traversal, cybersecurity, Oracle Cloud, cloud technologies, information security, tunneling, cryptography, remote access.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ СКОРОЧЕНЬ І ТЕРМІНІВ

OpenVPN — програмне забезпечення з відкритим кодом для реалізації VPN-з’єднання з використанням SSL/TLS шифрування.

NAT (Network Address Translation) — трансляція мережевих адрес, механізм, що дозволяє кільком пристроям використовувати одну зовнішню IP-адресу.

ISP (Internet Service Provider) — постачальник послуг Інтернету.

TCP/IP — стек мережевих протоколів, що використовується для передавання даних у комп’ютерних мережах.

UDP (User Datagram Protocol) — транспортний протокол, що забезпечує швидку, але ненадійну доставку даних.

TLS (Transport Layer Security) — криптографічний протокол для захищеної передачі даних у мережах.

PKI (Public Key Infrastructure) — інфраструктура відкритих ключів, яка забезпечує керування цифровими сертифікатами та шифруванням.

DHCP (Dynamic Host Configuration Protocol) — протокол динамічного призначення IP-адрес у мережі.

DNS (Domain Name System) — система доменних імен, що перетворює доменні імена на IP-адреси.

MikroTik — серія маршрутизаторів та програмне забезпечення для налаштування мережевих з’єднань.

CLI (Command Line Interface) — інтерфейс командного рядка для керування системою.

GUI (Graphical User Interface) — графічний інтерфейс користувача.

TLS-auth — механізм автентифікації TLS-з’єднань для запобігання DoS-атакам на VPN.

CIDR (Classless Inter-Domain Routing) — метод представлення IP-адрес та маршрутів з використанням префіксів.

RSA — криптографічний алгоритм із відкритим ключем, використовується для шифрування та підпису.

SSH (Secure Shell) — протокол для захищеного віддаленого доступу до комп’ютерів і серверів.

IP (Internet Protocol) — протокол мережевого рівня, який визначає формат адрес і правила маршрутизації пакетів.

IPv4 / IPv6 — четверта / шоста версія IP-протоколу.

Oracle Cloud — хмарна платформа, що надає обчислювальні ресурси та сервіси з публічним IP-доступом.

ICMP (Internet Control Message Protocol) — протокол, який використовується для передачі діагностичних повідомлень (наприклад, ping).

ПК – персональний комп‘ютер.

ПЗ – програмне забезпечення.

ВСТУП

У сучасному світі, що стрімко цифровізується, зростає потреба у створенні безпечних та гнучких мережевих інфраструктур, які забезпечують надійний обмін даними між пристроями, розташованими у різних географічних точках та підмережах. Особливу складність становить організація взаємодії між пристроями, що перебувають за NAT (Network Address Translation), де неможливо напряму ініціювати з'єднання ззовні без додаткових технічних засобів. Ця проблема є актуальною для систем моніторингу, телеметрії, відеоспостереження, «розумних» будинків та промислових IoT-рішень.

Одним із ефективних методів подолання обмежень NAT і забезпечення захищеної взаємодії є використання віртуальних приватних мереж (VPN). VPN дозволяє створити зашифрований тунель між пристроями, незалежно від їх фізичного розташування та конфігурації локальної мережі. У даному проекті обрана технологія OpenVPN — перевірене і широко підтримуване рішення з відкритим кодом, яке забезпечує високий рівень безпеки завдяки використанню сучасних криптографічних алгоритмів.

Щоб забезпечити постійну доступність VPN-сервера та гнучкість масштабування, у якості платформи для його розгортання використано хмарну інфраструктуру Oracle Cloud. Це дозволяє організувати централізовану точку доступу до VPN, мінімізувати витрати на апаратне забезпечення, а також реалізувати ефективні механізми адміністрування, моніторингу та резервного копіювання.

Даний дипломний проект присвячений розробці, реалізації та тестуванню віртуальної приватної мережі, яка дозволяє безпечно об'єднувати пристрої, розташовані за NAT, із використанням OpenVPN-сервера у хмарі Oracle. У проекті розглянуто актуальні загрози інформаційній безпеці, обґрунтовано архітектурні

рішення, реалізовано захищену інфраструктуру VPN, а також проведено оцінку ефективності запропонованого підходу.

1 АНАЛІЗ ОСТАННІХ ДОСЛІДЖЕНЬ І ПУБЛІКАЦІЙ

1.1 Огляд існуючих рішень для взаємодії пристроїв за NAT

У сучасних мережах проблема взаємодії між пристроями, які знаходяться в різних локальних мережах, значно ускладнюється при наявності NAT (Network Address Translation). У ситуації, коли обидва пристрої (наприклад, керуючий ПК і виконавчий пристрій) перебувають в одній локальній мережі, обмін даними відбувається без перешкод, оскільки IP-адреси належать до однієї підмережі. Однак, коли пристрої розташовані у різних мережах — особливо, якщо хоча б один із них знаходиться за NAT-пристроєм провайдера — виникає необхідність в маршрутизації трафіку або створенні спеціальних схем доступу.

Найпростішим способом налагодження зв’язку є проброс портів на маршрутизаторі, що дозволяє зовнішньому користувачу звертатися до внутрішнього пристрою через відкритий TCP/UDP порт. Однак цей метод не застосовний у випадку, коли немає доступу до налаштувань маршрутизатора, наприклад, у випадку NAT провайдера або в мобільних мережах. У таких умовах використання серверів-посередників є єдиним практичним вирішенням.

1.1.1 Використання хмарних серверів-посередників (пропрієтарні сервіси)

Багато сучасних пристроїв, особливо споживчого рівня, вирішують проблему NAT через підключення до хмарного серверу. Такий сервер-посередник приймає підключення від пристрою (зазвичай, ініційоване зсередини мережі, тобто NAT traversal), а керуючий пристрій (ПК, смартфон) також підключається до цього серверу — і вже через нього відбувається взаємодія.

Приклади:

– Sonoff, Tuya (розумний дім);

– Hikvision, Dahua (відеоспостереження);

– Xiaomi, TP-Link та інші виробники IoT-пристроїв.

Переваги:

– Просте підключення без потреби конфігурації мережі;

– Готові додатки для смартфонів і ПК;

– Автоматичне оновлення прошивки та доступу.

Недоліки:

1. Залежність від сторонніх серверів. При недоступності серверу (відмова, санкції, закриття компанії) пристрої втрачають функціональність.

2. Відсутність повного локального керування. Для доступу часто обов’язкове з’єднання з Інтернетом навіть у межах домашньої мережі.

3. Проблеми безпеки. Дані проходять через сторонні сервери (часто за кордоном), з ризиком перехоплення або несанкціонованого доступу.

4. Геополітичні ризики. Деякі компанії (наприклад, Dahua, Hikvision) потрапили під санкції через підозри у шпигунстві або порушенні прав людини.

5. Високі вимоги до підключення. Потрібен стабільний доступ до Інтернету.

6. Обмеження інтеграції. Закриті протоколи ускладнюють використання з власними системами (Home Assistant, Node-RED), часто необхідна перепрошивка.

7. Платні функції. Деякі сервіси стають доступними лише через підписку, що створює фінансову залежність.

1.1.2 Альтернативні рішення з відкритим кодом

1.1.2.1 OpenVPN / WireGuard (VPN-технології)

Ці технології дозволяють створити захищений тунель між пристроями, що обходить NAT через ініціацію з'єднання з обох сторін до хмарного сервера (VPN-сервер посередник).

Переваги:

– Повний контроль над даними;

– Високий рівень шифрування;

– Працює навіть при відсутності доступу до налаштувань маршрутизатора (через outbound з’єднання);

– Можливість об’єднання пристроїв у віртуальну мережу.

Недоліки:

– Потребує налаштування серверу у хмарі (наприклад, Oracle Cloud);

– Потребує знань з мережевої безпеки;

– Не всі пристрої мають підтримку OpenVPN/WireGuard.

1.1.2.2 ZeroTier, Tailscale (Overlay VPN)

Це більш автоматизовані VPN-рішення, які самостійно виконують NAT traversal, шифрування та маршрутизацію, часто з використанням технології WireGuard. Підходять для побудови приватних peer-to-peer мереж без складної конфігурації.

Переваги:

– Просте встановлення та налаштування;

– Працює з NAT без пробросу портів;

– Висока безпека та масштабованість.

Недоліки:

– Залежність від централізованих координаторів (можна мінімізувати у комерційних планах);

– Закритий код частини інфраструктури (у Tailscale);

– Може бути складніше для інтеграції в embedded-пристрої.

Отже, існуючі рішення для взаємодії між пристроями за NAT умовно поділяються на:

Прості, але обмежені пропрієтарні сервіси — зручні, проте залежні, небезпечні й не гнучкі;

Гнучкі та безпечні VPN-рішення — потребують налаштування, але забезпечують повний контроль, кастомізацію та незалежність.

Для користувачів, які цінують безпеку, конфіденційність, стабільність та незалежність, найдоцільнішим є створення власної інфраструктури з використанням OpenVPN у хмарі (наприклад, Oracle Cloud Free Tier). Це дозволяє поєднувати переваги хмарних серверів із повним контролем над трафіком, виключаючи сторонні ризики.

1.2 Реалізація хмарного доступу у системах відеоспостереження Dahua

Щоб підключити відеореєстратор Dahua до Інтернету, необхідно включити P2P в налаштуваннях відеореєстратора. Для цього потрібно зайти в головне меню реєстратора, натиснувши праву кнопку миші в інтерфейсі реєстратора і натиснувши «ГОЛОВНЕ МЕНЮ».

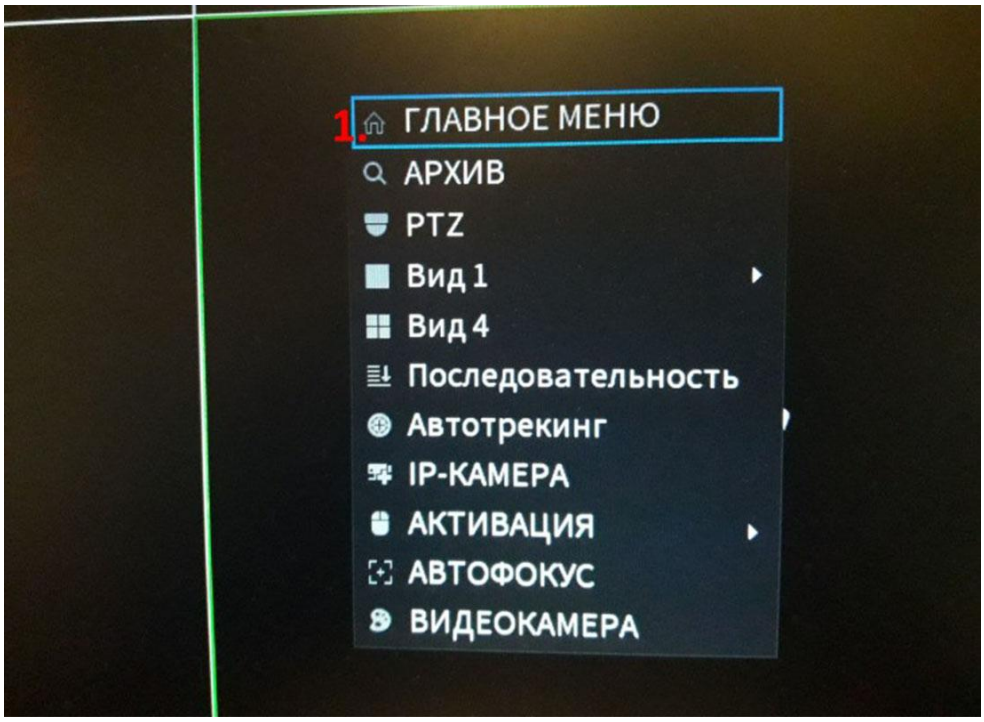

Рисунок 1.1 – Налаштування відереєєстратору Dahua

Відкриється Головне меню реєстратора, де потрібно натиснути кнопку «Мережа» (2).

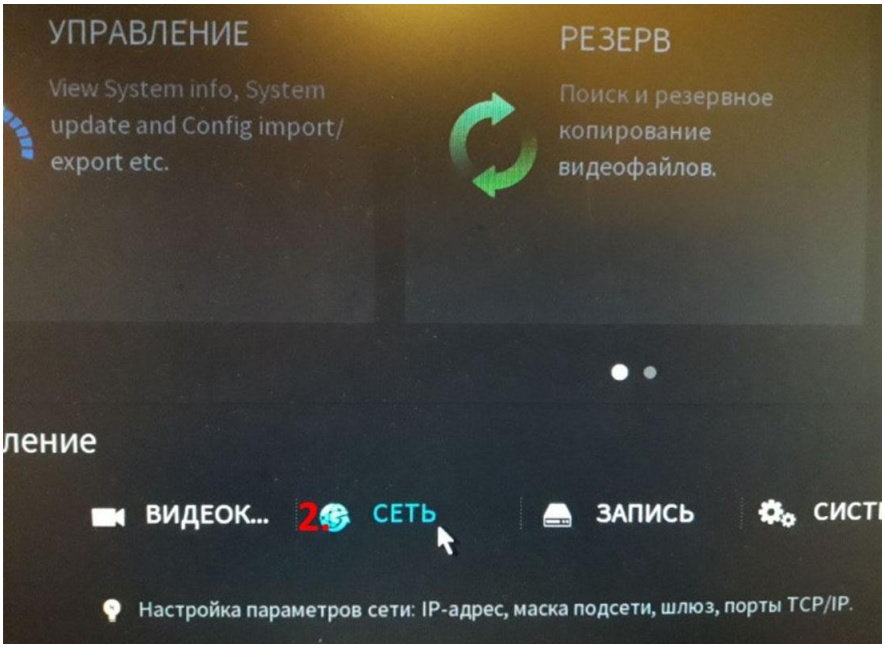

Рисунок 1.2 – Налаштування відереєєстратору Dahua

Після цього відкриється меню конфігурування мережі. Тут потрібно перейти на вкладку «P2P» (3) і ввімкнути цю функцію, натиснувши на прапорець «Увімк.» (4). А потім натиснути кнопку «Застосувати» (5).

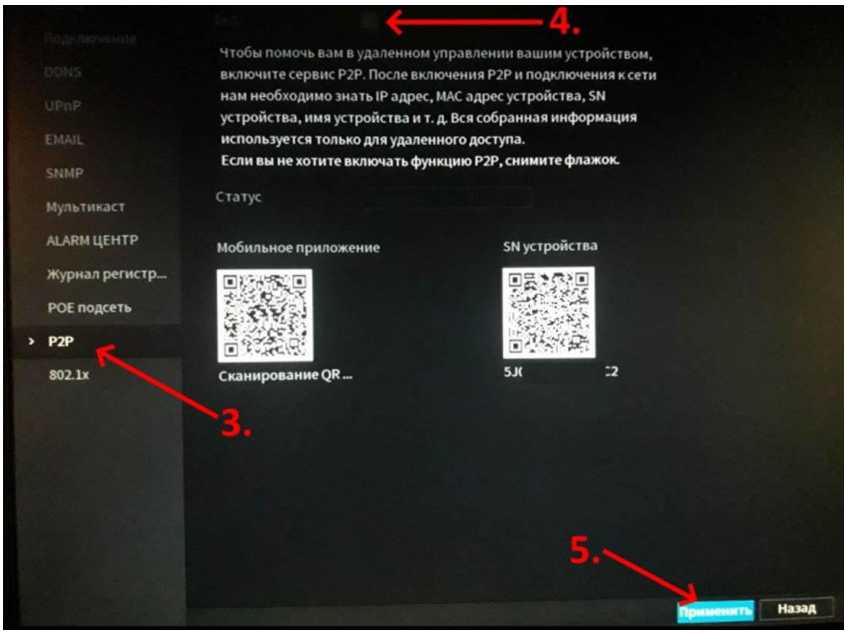

Рисунок 1.3 – Налаштування відереєєстратору Dahua

Після цього прапорець змінить колір із сірого на синій і реєстратор буде програмно підключеним до Інтернету. Щоб перевірити чи це так, необхідно перейти на іншу вкладку, наприклад на вкладку «Підключення», а після цього відразу ж перейти назад на вкладку «P2P». Якщо все зроблено правильно - "Статус" (6) зміниться на "Онлайн".

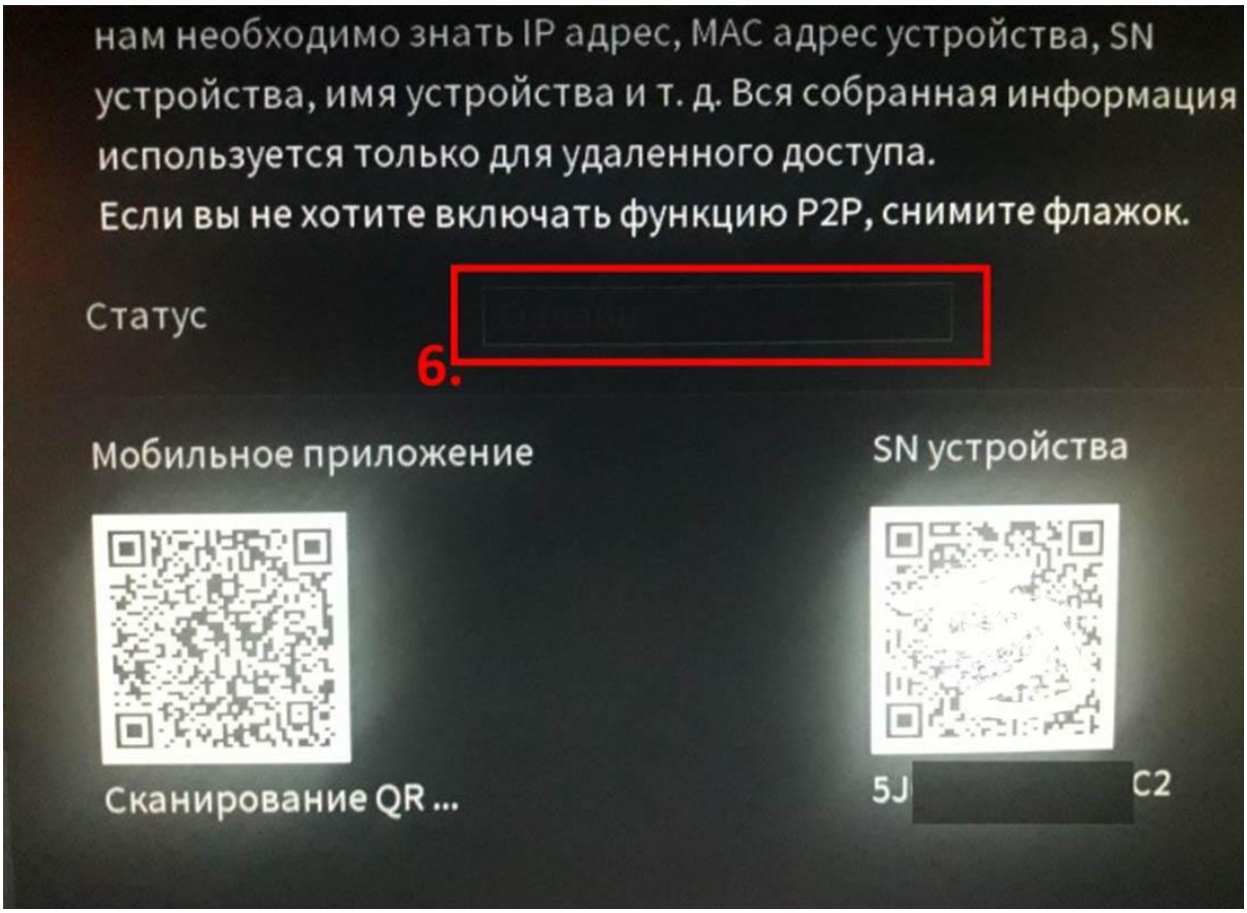

Рисунок 1.4 – Налаштування відереєєстратору Dahua

1.3 Реалізація хмарного доступу у системах відеоспостереження Dahua

Як і у реєстраторах Dahua, необхідно увімкнути режим роботи P2P на пристрої, незалежно від того, який пристрій налаштовується.

Це робиться через головне меню пристрою, "Конфігурація системи" - "Мережа" - "Платформа доступу".

Необхідно поставити галочку "Увімкнено".

Далі треба придумати код перевірки та прописати його у вказаному полі, після чого потрібно погодитися з користувальницькою угодою.

Реєстратор має бути підключений до мережі та підключитися до сервера, а у рядку «Статус» з'явиться напис: «У мережі». Якщо показує, що обладнання неактивне, необхідно з'єднатися з DHCP та підключення до Інтернету.

Далі необхідно зареєструватись на сервісі Hik-connect, перейти на офіційний сайт програми: https://www.hik-connect.com/register, де потрібно пройти просту процедуру створення облікового запису. Під час реєстрації потрібно правильно вказати країну, а також підтвердити електронну пошту та телефон. Провести процедуру реєстрації ви зможете абсолютно з будь-якого пристрою.

Після завершення процедури реєстрації можна додати пристрій до облікового запису, який потрібно підключити. З цією метою треба перейтив розділ «Керування пристроєм» - «Додати», після чого ввести серійний номер пристроїв, який вказано на коробці. Якщо додавати пристрої за допомогою смартфона, можна відсканувати QR-код, який розташований в меню реєстратора.

Далі треба вказати код перевірки, який раніше було вигадано. Якщо все нормально функціонує і кожне, і пристроїв у мережі, у рядку «Стан пристрою» буде статус Online.

Щоб розпочати віддалений перегляд того, що відбувається на об'єкті за допомогою камер, необхідно завантажити програму на телефон IVMS-4500 або на ПК IVMS-4200. Якщо відеоспостереження буде проводитися за допомогою телефону, після монтажу програми потрібно пройти процедуру авторизації, вказавши дані, які були вказані під час реєстрації.

При відеоспостереженні онлайн за допомогою комп'ютера, потрібно перейти на вкладку «Управління пристрій» - «Додати новий пристрій», а в новому

вікні поставити позначку біля Hik-connect Device (P2P Device). Зліва з'явиться Hik-connect Device (P2P Device), вибрати країну та ввести дані, які вказували під час реєстрації.

У тому випадку, якщо кожен етап був проведений правильно, у програмі з'являться всі пристрої, які додані в акаунті.

1.4 Реалізація хмарного доступу у системах розумного будинку eWeLink

Багато користувачів вважають, що смарт-пристрої Wi-Fi підключаються до серверів eWeLink безпосередньо. Однак це не так. Насправді смарт-пристрої миттєво проходять через маршрутизатор, модем та DNS-сервер для підключення до серверів eWeLink. Тільки тоді можна керувати пристроями дистанційно. Повне з'єднання складається з трьох фаз.

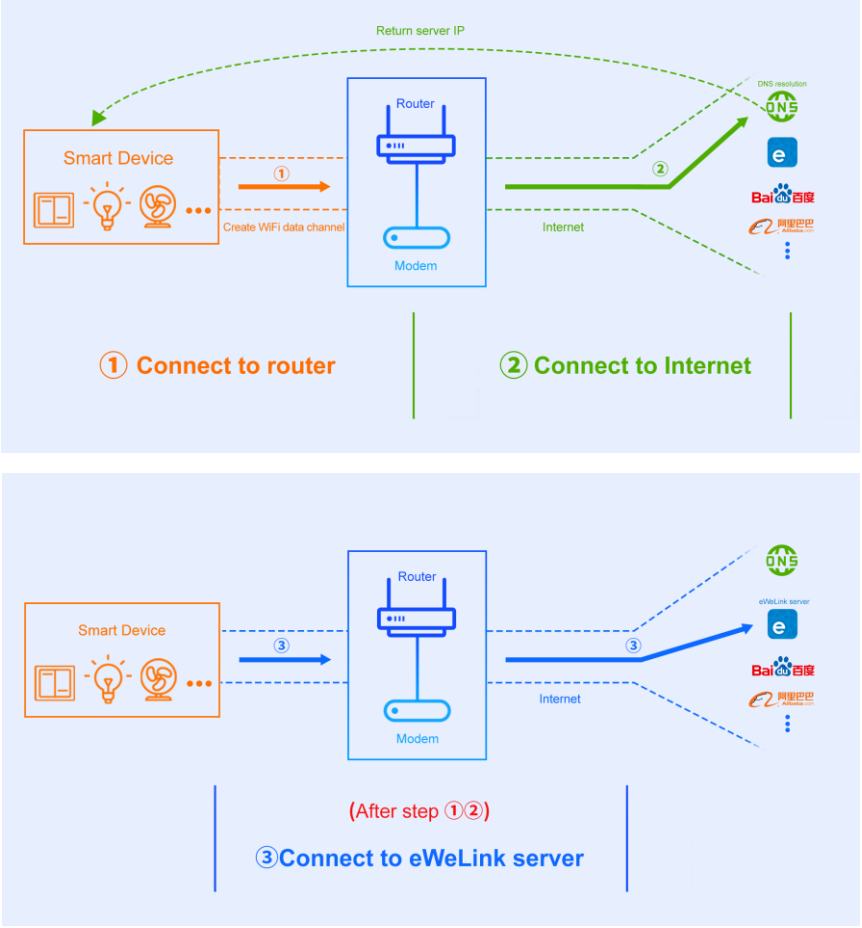

Рисунок 1.4 – Підключення до серверів eWeLink

1.4.1 Фаза 1. Підключення до маршрутизатора

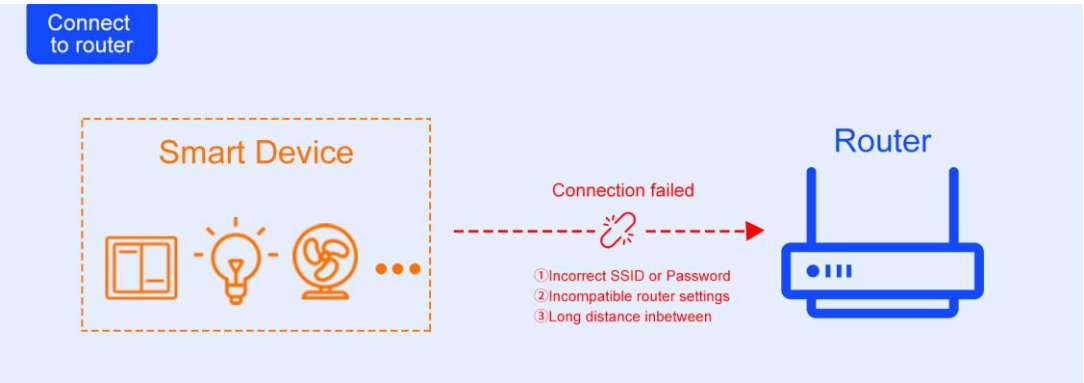

Рисунок 1.5 – Підключення пристрою eWeLink до маршрутизатору

Домашній маршрутизатор – це перший місток між смарт-пристроєм та зовнішнім зв'язком. Тому нам слід виконати правильні кроки, щоб підключити смарт-пристрої до маршрутизатора та створити канал для передачі даних Wi-Fi.

Якщо смарт-пристрій не може підключитися до маршрутизатора необхідно виконати наступні кроки:

1. Перевірити правильність SSID та пароля маршрутизатора.

SSID та пароль – це номер кімнати та ключ вашого маршрутизатора. Якщо будь-який з них неправильний, пристрій взагалі не підключиться до маршрутизатора.

2. Перевірити налаштування маршрутизатора.

Якщо ваш SSID та пароль Wi-Fi правильні, але пристрій залишається офлайн або сполучення все ще не вдається, слід перевірити налаштування маршрутизатора та подивитися, чи не завадили вони підключенню смарт-пристрою. З міркувань безпеки багато людей увімкнули деякі налаштування маршрутизатора, такі як фіксована IP-адреса, білий список тощо.

Щоб уникнути збою підключення, переконайтеся, що налаштування вашого маршрутизатора такі.

Таблиця 1.1 – Рекомендовані налаштування

Router Settings	Recommended Status
DHCP	Enabled
White list	Disabled
Two-way authentication	Disabled
AP Isolation	Disabled
MAC address filter	Disabled

3. Переконатися, що смарт-пристрій знаходиться в межах досяжності маршрутизатора.

Якщо смарт-пристрій знаходиться занадто далеко від маршрутизатора, наприклад, біля кількох стін або кімнат між ними, слабкий сигнал Wi-Fi може призвести до збою з'єднання. Щоб виправити це, необхідно розмістістити смарт-пристрій там, де сигнал сильніший.

1.4.2 Фаза 2. Підключення до Інтернету.

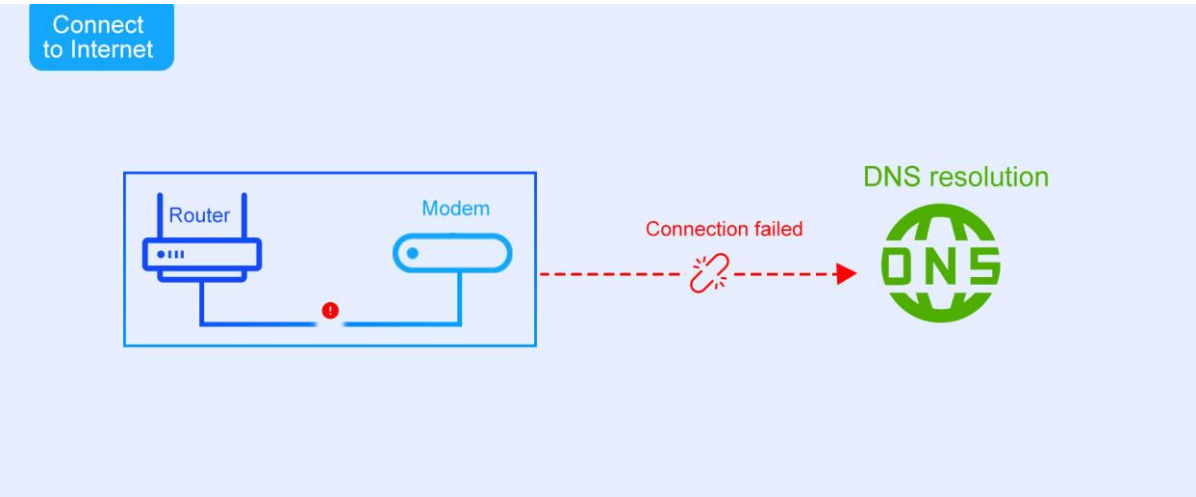

Рисунок 1.6 – Підключення пристрою eWeLink до Інтернету

На цьому етапі дані смарт-пристрою пройдуть через маршрутизатор та підключаться до Інтернету через модем, намагаючись отримати IP-адресу сервера eWeLink за допомогою DNS-дозволу. Щоб вирішити помилки або збої на цьому етапі, необхідно виконати ці кроки з усунення несправностей:

1. Переконатися, що пристрій підключено до маршрутизатора, а модем працює.

Перевірити індикатори живлення, приймача, надсилання, онлайн/інтернет/кабель/синхронізація/сигнал на вашому модемі, щоб побачити, чи вони працюють нормально. Крім того, також треба перевірити наявність нещільного з'єднання в проводах між маршрутизатором і модемом.

2. Переконатися, що інтернет-з'єднання доступне.

Якщо модем підключено та працює, вам слід перевірити, чи інтернет доступний і стабільний. Ви можете зробити це, підключивши свій смартфон виключно до Wi-Fi та спробувавши переглянути будь-який веб-сайт на своєму телефоні.

3. 3. Помилка розв'язання DNS

Якщо виникає помилка розв'язання DNS, ваш смарт-пристрій не зможе отримати IP-адреси серверів eWeLink, що призведе до помилки підключення до сервера. Щоб виправити це, зверніться до свого інтернет-провайдера, щоб перевірити, чи є помилка DNS-сервера або чи адреси серверів eWeLink були заборонені.

1.4.3 Фаза 3. Підключення до сервера

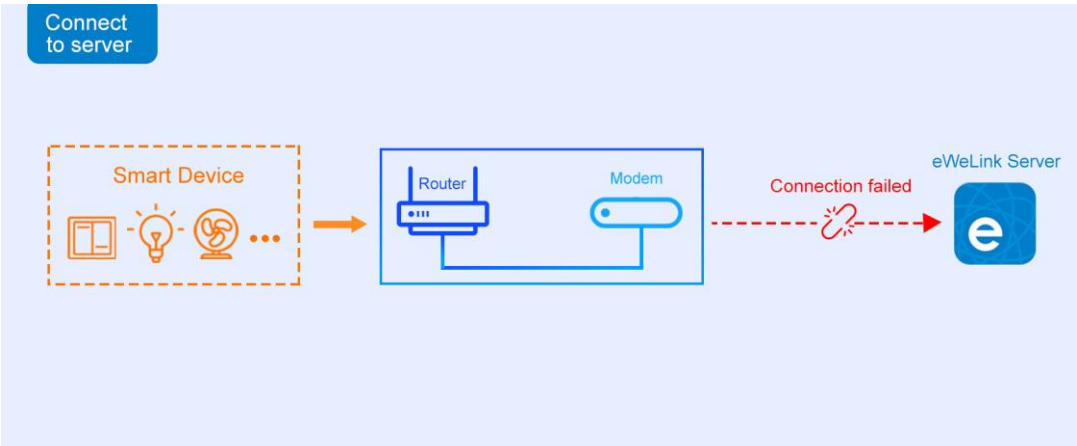

Рисунок 1.7 – Підключення пристрою eWeLink до серверу

Після отримання IP-адрес сервера eWeLink, смарт-пристрій Wi-Fi спробує встановити з'єднання з сервером eWeLink. Після цього користувач зможе контролювати та керувати пристроєм.

Аварії, які можуть траплятися.

1. Сервер eWeLink заборонив доступ

У рідкісних випадках сервер eWeLink відхилятиме запити на підключення від деяких смарт-пристроїв, наприклад, коли пристрій не зареєстровано належним чином або ідентифікатор пристрою не відповідає базам даних. Виправлення полягають у повторному додаванні пристрою або зверненні до продавця чи служби післяпродажного обслуговування для отримання додаткової допомоги.

2. Сервер не працює

Коли раптово велика кількість пристроїв у певному регіоні перестає працювати одночасно або подібна помилка підключення трапляється з багатьма користувачами, сервер може бути непрацюючим з неочікуваних причин, таких як землетрус або пошкодження оптоволоконного кабелю.

Зазвичай звичайним користувачам не потрібно перевіряти підключення, оскільки смарт-пристрої реагують на наші команди за мілісекунди.

1.5 Реалізація хмарного доступу у системах розумного будинку Tuya smart

Людство прагне підвищувати комфорт свого життя. І найкращим вибором для цього будуть технології розумного будинку.

Тепер необов'язково встановлювати такі складні системи, оскільки є хмарна система управління пристроями розумного будинку Tuya Smart, якої цілком достатньо для штатного функціоналу та вирішення більшості побутових завдань, наприклад, керувати світлодіодним освітленням.

Розглянемо докладніше особливості системи розумного будинку Tuya Smart, можливості, переваги, недоліки та тонкощі інтеграції.

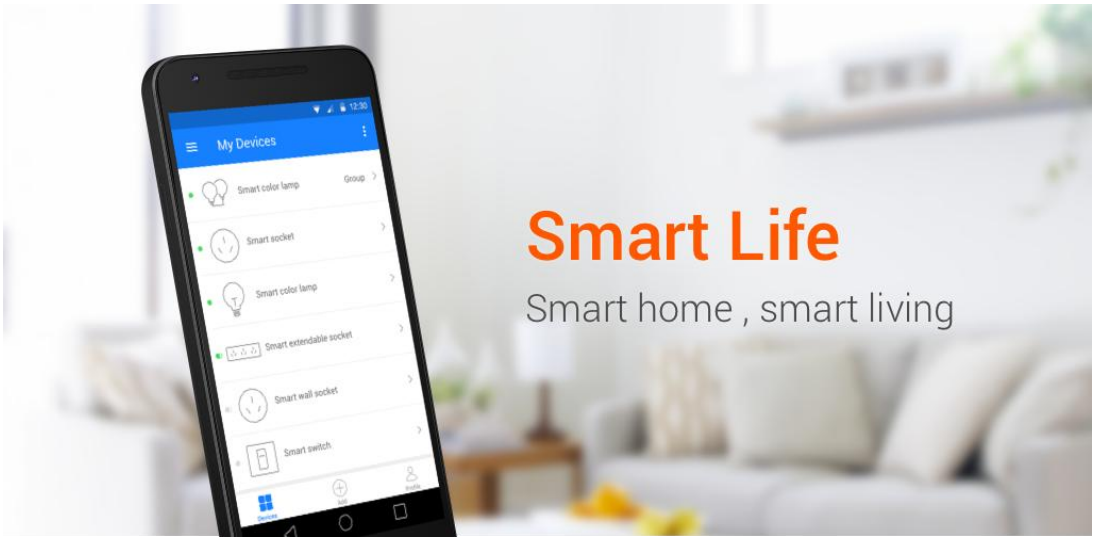

Рисунок 1.8 – Система розумного будинку Tuya Smart

З'явилася система ще 2014 року. Розробкою зайнялися колишні співробітники Alibaba. Штаб-квартира компанії знаходиться в Індії, Китаї, Японії, США та Німеччині, тому дана система має високий попит по всьому світу. Слід зазначити, що це система займається розвитком багатьох напрямів IoT. Але основний напрям – це програмне забезпечення для екосистеми розумного будинку.

Унікальність системи Tuya Smart полягає в тому, що вона поєднує в собі не тільки кінцеві пристрої користувача, але також і розробників, які можуть ви-

користовувати API, за допомогою нього можна розробляти особисті додатки та пристрої для роботи з обладнанням.

Тобто це платформа, завдяки якій у виробників є можливість проводити розробку додатків, тестувати та оновлювати Smart-обладнання. Це сприяло тому, що є тисячі різних пристроїв від різних виробників, які можуть бути об'єднані в єдину систему.

На відміну від більшості інших систем, це по-справжньому народний бренд – велика кількість бюджетного та середнього класу. Серед каталогу обладнання можна знайти абсолютно все – контролери для світлодіодного освітлення, базові датчики, специфічні напівпромислові пристрої та інше. І важливим є те, що є продукція як від маловідомих виробників, так і іменитих брендів, тому кожен може вибрати відповідний варіант під власний бюджет.

Необхідно відзначити, що компанія Tuya Smart співпрацює навіть із всесвітньо відомими брендами, наприклад, Schneider Electric, PHILIPS та іншими. Це означає, що є продукція преміальної якості. Постійно розширення списку пристроїв сприяє тому, що вже довгий час розумний будинок від Tuya Smart – це найкраще рішення для домашньої системи.

Важливо, що все підключене обладнання працює стабільно і безвідмовно при підключенні Wi-Fi, Bluetooth або ZigBee – все залежить від конкретного пристрою. Користувач має можливість через спеціалізований додаток об'єднати в єдину систему сотні гаджетів для зручного керування.

Деякі можливості розумного будинку Tuya Smart

Незважаючи на всю простоту, можливості Tuya Smart практично безмежні. Розглянемо деякі з них з використанням різного обладнання:

– Розетки. Користувач може керувати подовжувачами, розетками, вимикачами світла та іншими подібними приладами.

– Освітлення. За допомогою програми можна керувати стельовим вентилятором, гірляндою, світлодіодною стрічкою, неоном, магнітною трековою системою тощо.

– Велика побутова техніка. Запуск, програмування та інше керування пральною машиною, холодильником, посудомийною машиною тощо.

– Кухонне обладнання. За допомогою обладнання можна керувати тостером, машиною для кави, блендером та іншими кухонними приладами.

– Системи безпеки. Цілодобове керування та налаштування сигналізацією, сейфом, різними датчиками на приміщенні, що охороняється.

– Здоров'я. Розумне керування ліжком, вагами, тонометром та іншими приладами, що призначаються для здоров'я.

– Відеоспостереження. Тонка настройка та керування камерами відеоспостереження для забезпечення цілодобового контролю над об'єктом.

Крім цього, із системою розумного будинку Tuya Smart можна керувати телевізором, шлюзом, щитовими приладами, агротехнікою та іншим обладнанням. Залежно від власних переваг можна встановлювати різні сценарії автоматизації. Єдина вимога – наявність на пристрої позначки Powered by Tuya.

Ключові переваги застосування системи Tuya Smart

– Широкий вибір пристроїв. Представлені тисячі різних позицій пристроїв, завдяки чому можна знайти відповідне обладнання абсолютно під різні завдання. При цьому перелік пристроїв стабільно поповнюється.

– Просте підключення. Немає необхідності мати якісь особливі знання у налаштуванні та установці – все максимально просто та інтуїтивно зрозуміло. Не потрібно жодних дротів.

– Універсальність. Всі пристрої з чіпом Tuya підходять для фірмового застосування, а також для деяких його аналогів.

– Працює через стандартні технології. Немає необхідності в якомусь спеціалізованому шлюзі для взаємодії – більшість працює через Wi-Fi, а деякі пристрої працюють через ZigBee, GPRS та Bluetooth.

– Повна свобода доступу. Є можливість підключення до пристроїв із будь-якої точки світу. А також програма відправляє push-повідомлення на телефон. Швидкість відгуку досить висока, навіть якщо йдеться про бюджетне обладнання.

– Широкий вибір виробників. Велика кількість виробників, у тому числі іменитих брендів, пропонують продукцію, що підходить для підключення за технологією Tuya Smart.

– Може працювати з іншими системами розумного будинку. Технологія Tuya Smart функціонує без проблем з іншими системами розумного будинку, тому ви можете створити масштабну систему автоматизації.

– •олосове керування. Є можливість підключення до сервісів голосового управління, наприклад Google Home і Amazon Alexa.

– Різні сценарії. Всередині програми максимально просто і зручно створювати сценарії та алгоритми автоматизації – сценаріїв просто безмежна кількість.

– Сумісність. Абсолютно всі пристрої, які мають логотип Powered by Tuya, ідеально сумісні між собою.

– Доступність. Вартість обладнання досить низька, єдине, що потрібно заплатити за сховище для системи відеоспостереження. В іншому все безкоштовно.

Недоліки:

– Працює на базі хмарного сервісу Tuya Cloud, що підтримує зв'язок між усіма пристроями. І якщо немає інтернету, тоді сценарії автоматизації не функціонуватимуть.

– Пристрої виготовляються різними виробниками, у тому числі маловідомими. Це означає, що є велика кількість обладнання низької якості, яке може зіпсувати враження про систему. Потрібно бути уважним під час виборів.

– Локалізована продукція іноді надто складна для українського споживача. Навіть якщо щось і перекладено, то сам переклад бажає кращого.

– Незручно, що не передбачено web-інтерфейс або версія клієнта для комп'ютера, що спростило б роботу з обладнанням.

Якщо оцінювати кількість переваг, то недоліки зовсім незначні. Все це компенсується доступною вартістю, якщо порівнювати з іншими системами розумного будинку.

Виробник подбав про те, щоб платформа не тільки підтримувала сотні різних продуктів, але також була максимально простою в плані встановлення.

1. Завантажити мобільний додаток Tuya Smart або Smart Life, пройти процедуру реєстрації, увімкнути на пристрої Wi-Fi та Bluetooth.

2. Датчик, освітлювальне обладнання, камеру відеоспостереження або інший пристрій потрібно перевести в режим сполучення, після чого в програмі знайти девайс, що додається - програма автоматично його розпізнає, після чого надати підказки про те, як його додати.

3. Коли пристрій був доданий, перед користувачем буде відкрито широкий вибір налаштувань, функцій, сценаріїв та автоматизацій, а також голосове керування.

Усього цих 3 кроків достатньо, щоб підключити обладнання. Необхідно відзначити, що для Wi-Fi пристроїв цілком достатньо наявності роутера.

Очевидно, що Tuya Smart – це найкращий розумний будинок для користувачів, які не хочуть створювати масштабні системи автоматизації. Це рішення цілком достатньо для вирішення більшості побутових завдань, при цьому доступна вартість робить систему доступною абсолютно для кожної людини.

1.6 Перспективи розвитку віртуальних приватних мереж для об’єднання пристроїв за NAT

У сучасному цифровому середовищі дедалі більше пристроїв працюють у закритих або ізольованих локальних мережах (LAN), без прямого доступу до глобального Інтернету. Водночас виникає потреба у безпечному, постійному та стабільному обміні даними між такими пристроями, які знаходяться за NAT (Network Address Translation). Створення віртуальних приватних мереж (VPN) є одним із найефективніших способів вирішення цієї задачі, і має високі перспективи подальшого розвитку.

1.6.1 Інтеграція з IoT-пристроями

Інтернет речей (IoT) активно розвивається, і більшість таких пристроїв працює за NAT. VPN дозволяє:

- об'єднувати розподілені IoT-пристрої в єдину захищену мережу;
- керувати ними централізовано;
- реалізовувати безпечне оновлення прошивок і обробку даних.

У майбутньому очікується поява легковагових VPN-клієнтів спеціально для IoT, що можуть працювати навіть на пристроях із обмеженими ресурсами.

1.6.2 Хмарна інфраструктура як посередник (Cloud VPN Broker)

Використання таких платформ, як Oracle Cloud, AWS, Azure для хостингу VPN-серверів – популярна тенденція:

- не потребує фізичного сервера;
- дозволяє масштабувати підключення;
- підтримує автоматизацію (наприклад, автооновлення сертифікатів, моніторинг стану каналів).

Подальший розвиток піде у бік автоматизованих VPN-інфраструктур, які самостійно створюють тунелі між пристроями залежно від політик доступу.

1.6.3 Заміна класичного VPN новими протоколами

OpenVPN і IPSec — це перевірені рішення, але сучасні виклики стимулюють перехід до:

– WireGuard — швидкий, безпечний, легкий у налаштуванні протокол із відкритим кодом;

– ZeroTier або Tailscale — реалізації P2P VPN із NAT traversal, що дозволяють об’єднувати пристрої без ручного налаштування.

Це відкриває можливості для масового використання VPN навіть у побутових рішеннях (розумний будинок, мобільні застосунки, геймерські мережі).

1.6.4 📌 4. Розвиток NAT traversal технологій

Наразі VPN — не єдиний спосіб обійти NAT. Перспективними є також:

– STUN/TURN/ICE — техніки з WebRTC, що дозволяють встановлювати прямі P2P-з’єднання;

– Hole punching — динамічне пробивання NAT через відкриті вихідні з’єднання;

– Smart relays — проміжні сервери, що автоматично оптимізують маршрути.

Інтеграція цих методів у VPN-протоколи дозволить досягти меншої затримки та кращої адаптації до різних типів NAT (symmetric, restricted).

1.6.5 Автоматизація та управління на базі політик

У майбутньому очікується розвиток інтелектуальних VPN-систем, які будуть:

– самі налаштовувати маршрути між вузлами;

– виявляти загрози в тунелях;

– контролювати рівень шифрування залежно від контенту;

– підтримувати RBAC (Role-Based Access Control) на рівні VPN-інфраструктури.

1.6.6 Інтеграція з системами кібербезпеки

VPN усе частіше стає частиною комплексної системи захисту, а не лише каналом передачі. Майбутні VPN-рішення:

– вбудовуватимуть IDS/IPS (наприклад, Suricata) для фільтрації трафіку в тунелі;

– взаємодіятимуть із SIEM-платформами для журналювання;

– використовуватимуть двофакторну автентифікацію, сертифікати, TPM-чіпи.

Віртуальні приватні мережі для об’єднання пристроїв за NAT залишаються актуальним і стратегічно важливим напрямом, зокрема в галузі кібербезпеки, розподілених систем, розумних будинків, відеоспостереження та IT-інфраструктури. Подальший розвиток VPN-інструментів буде спрямований на:

– підвищення гнучкості та простоти налаштування;

– адаптацію до нових протоколів;

– тісну інтеграцію з іншими засобами безпеки та моніторингу.

2 СТРУКТУРА СИСТЕМИ

Система, яка базується на власних сервісах, може уникнути недоліків, притаманних іншим підходам. Її узагальнена мережева структура, що дозволяє досягти цього, представлена на рис 2.1.

Головним елементом цієї системи є VPN-сервер, який розміщується або на власному маршрутизаторі, або в хмарному середовищі, і має публічну IP-адресу. Це ключовий елемент для забезпечення зв'язку.

Усі пристрої, які знаходяться за технологією NAT (наприклад, у домашній мережі чи в польових умовах), ініціюють VPN-з'єднання з цим хмарним сервером. Центральний сервер управління, який може бути розташований будь-де в Інтернеті, також приєднується до того ж VPN-сервера.

Завдяки цьому VPN-з'єднанню, всі ці пристрої об'єднуються у віртуальну приватну мережу. Це дозволяє їм безпосередньо обмінюватися даними та здійснювати керування, повністю обходячи обмеження, які накладає NAT.

Для перевірки функціональності цієї системи була розгорнута віртуальна машина в хмарі Oracle. Варто зазначити, що такий сервер легко реалізувати за допомогою образів CHR (Cloud Hosted Router) від Mikrotik. Однак програма Oracle Cloud Free Tier не підтримує використання власних образів.

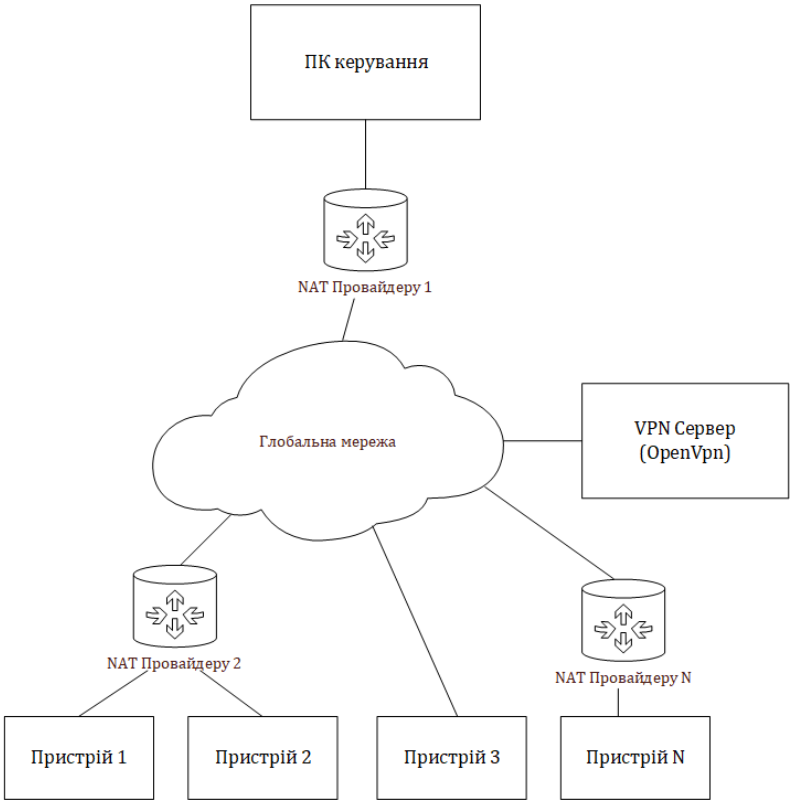

Рисунок 2.1 – Узагальнена схема мережі

Для перевірки роботи такої системи у хмарі Oracle обрано віртуальну машину під керуванням операційної системи Linux Debian (рис. 2.2). Система має IP адресу, що належить діапазону приватних мереж, але доступ до сервісів віртуальної машини може бути наданий через відкриття необхідних портів у налаштуваннях безпеки віртуальної хмарної мережі. Подібна система може бути реалізована і на повністю власному сервері, що знаходиться за маршрутизатором з білою IP адресою. Це з одного боку підвищує безпеку через незалежність від роботи обладнання сторонніх організацій, але в умовах нестабільності роботи як електричних, так і телекомунікаційних мереж в Україні може призводити до нестабільності роботи.

До додаткових переваг використання хмарного сервісу можна віднести маскування реальної IP адреси при виході до мережі Інтернет за рахунок того, що весь трафік може перенаправлятися через тунель (рис. 3).

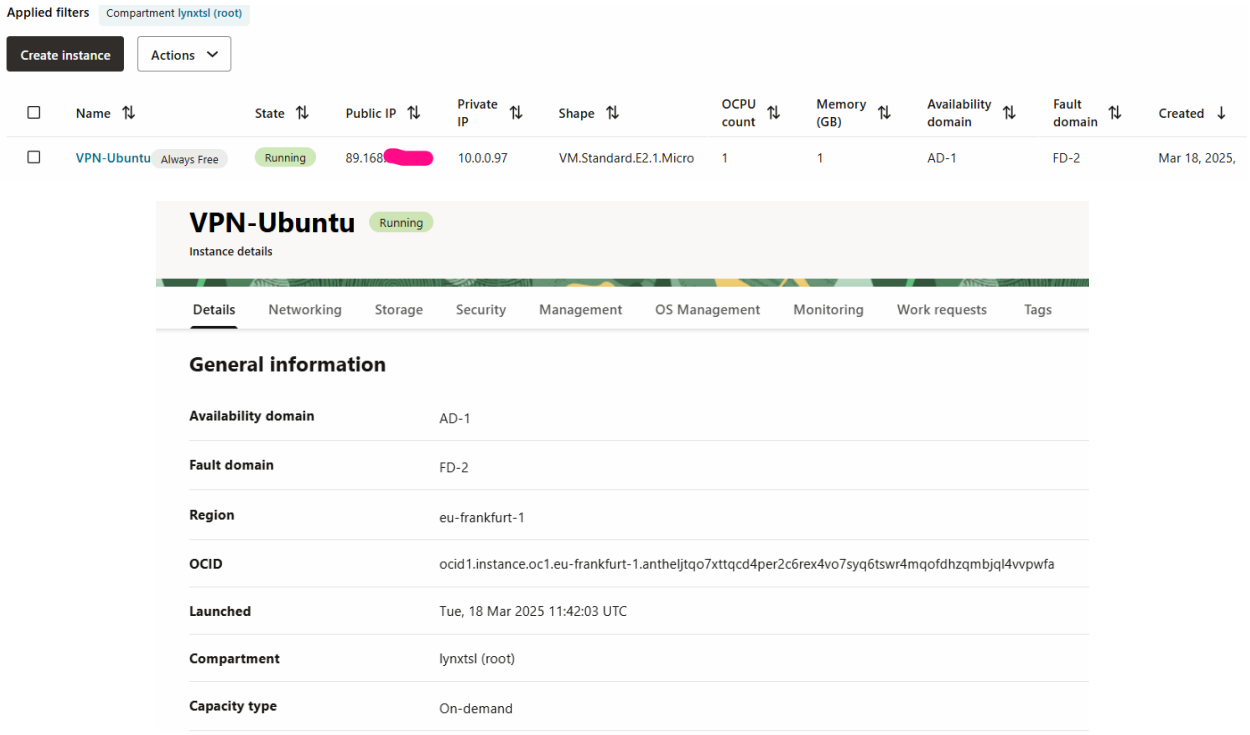

Рисунок 2.2 – Віртуальна машина з VPN сервером у хмарі

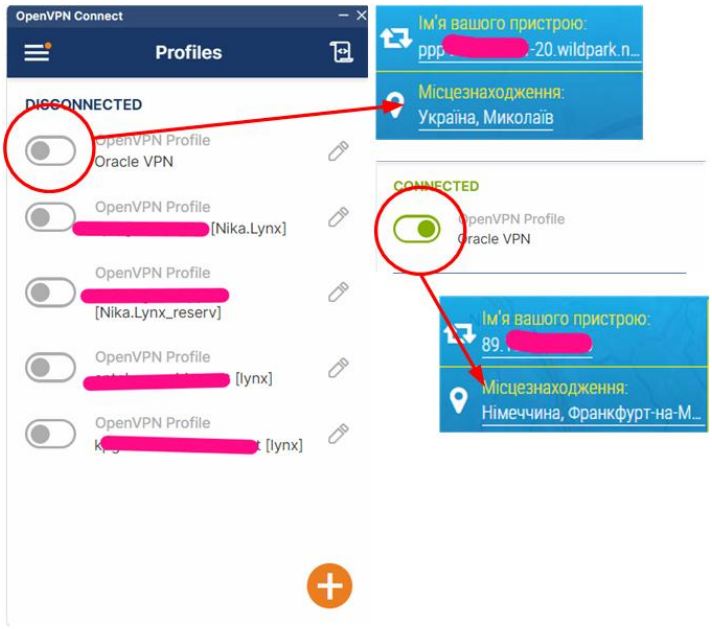

Рисунок 2.3 – Перенаправлення трафіку через VPN тунель

Прикладом використання подібної системи може бути система телекерування роботом (рис. 2.4).

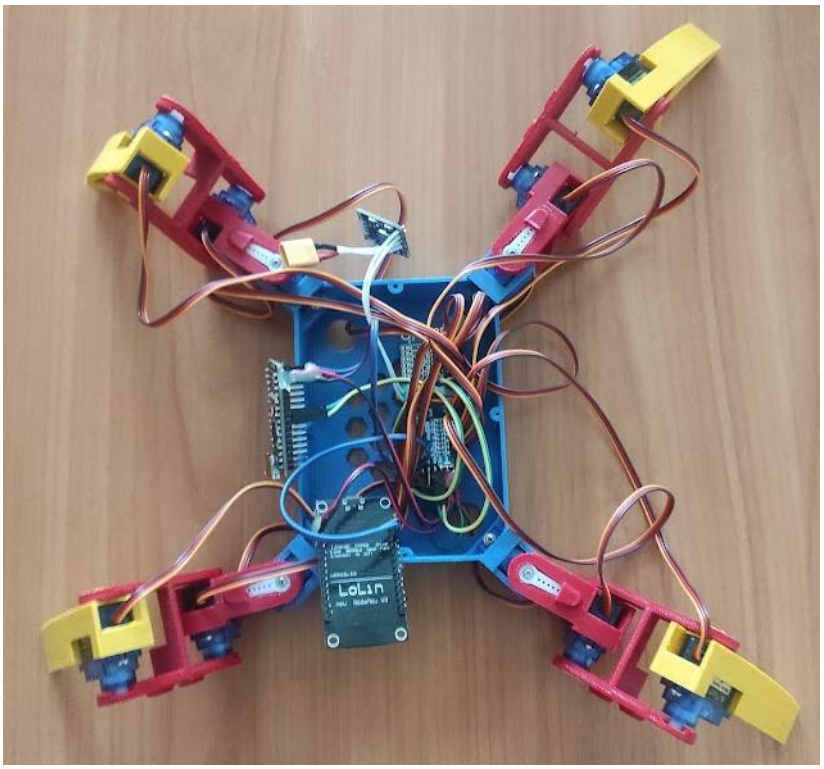

Рисунок 2.4 – Робот, як приклад пристрою системи телекерування

Система керування рухом даного роботу побудована на базі мікроконтролеру Atmega328. За зв’язок відповідає модуль ESP32, який обмінюється даними з контролером руху через інтерфейс UART. Даний модуль зв’язку передбачає лише бездротовий інтерфейс WiFi обміну даними з комп’ютером керування, тому дальність роботи буде обмежена розмірами локальної мережі. Для збільшення дальності можливо використання аналогічного модуля, але із вбудованим GSM модемом рис.2. 5. Але OpenVPN використовує повноцінний SSL/TLS-стек, шифрування на основі OpenSSL (або mbedTLS), управління сертифікатами, TCP/UDP-обробку — це ресурсоємно. ESP32 має обмежену оперативну пам’ять (до ~520 КБ RAM) та обчислювальні можливості, чого недостатньо для стабільної роботи OpenVPN. Наразі немає офіційної або спільнотної реалізації OpenVPN-клієнта для ESP32. Навіть якщо зібрати мінімальну версію з сильно обрізаними можливостями — вона не буде стабільно працювати, особливо з використанням сертифікатів, TLS-шифрування, перевірки ключів.

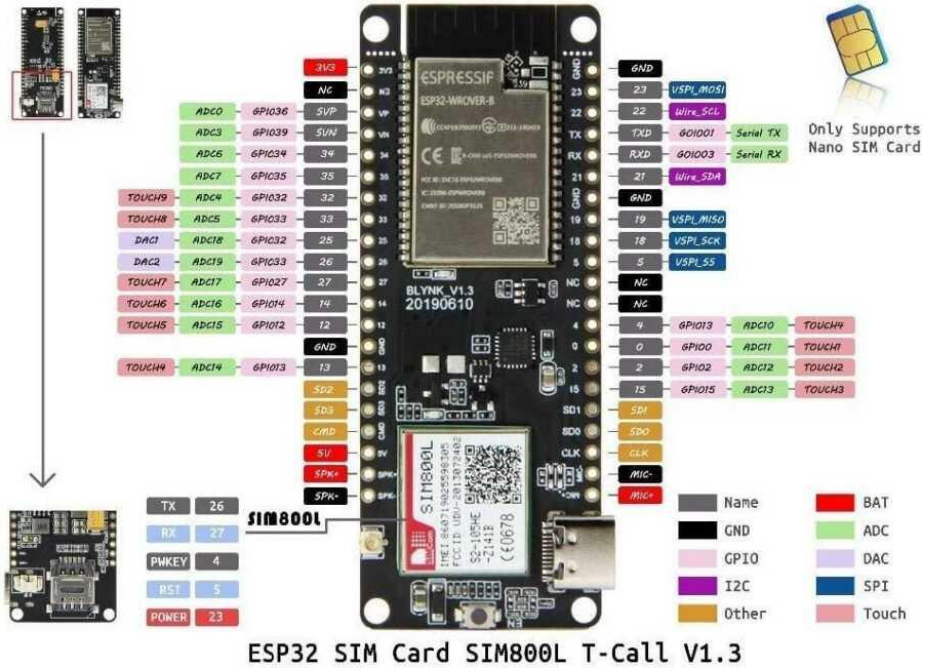

Рисунок 2.5 – Модуль зв’язку ESP32 з GSM модемом

Тому для повноцінної роботи через різноманітні канали зв’язку доцільно використовувати маршрутизатор, наприклад Mikrotik чи OpenWRT, або міні ПК типу Raspberry Pi на які буде покладено функції клієнта OpenVPN та задачі маршрутизації трафіку.

3 НАЛАГОДЖЕННЯ

3.1 Налаштування віртуальної машини

Встановлення віртуальної машини Debian на Oracle Cloud Free Tier вимагає декількох кроків, оскільки Oracle зазвичай надає власні оптимізовані образи, а для Debian потрібно використовувати "власний образ" (Custom Image). Ось основні етапи:

1. Створення облікового запису Oracle Cloud Free Tier

Реєстрація. Необхідно перейдіть на сайт Oracle Cloud і натиснути "Sign Up" або "Try for Free".

Введення даних. Заповнити необхідні дані (ім'я, електронна пошта, країна).

Підтвердження пошти. Підтвердити свою електронну пошту.

Пароль та регіон. Ввести пароль і обрати "Home Region" (домашній регіон). Це важливо, оскільки Free Tier ресурси доступні в конкретних регіонах, і їх не можна змінити пізніше.

Кредитна картка. Ввести дані кредитної картки. Плату за Free Tier ресурси не буде стягнуто, це лише для верифікації особи. Переконайтеся, що ви залишаєтеся в межах "Always Free" лімітів.

2. Завантаження образу Debian для хмарних середовищ

Знайти образ. Перейти на офіційний сайт Debian, де доступні хмарні образи (Cloud Images). Знайти образи, призначені для віртуальних машин (зазвичай у форматі .qcow2). Потрібен "Generic cloud" образ, можливо, для архітектури ARM64 (якщо планується використовувати Ampere A1 Compute Instance, який є частиною Free Tier).

3. Створення екземпляра віртуальної машини (VM Instance)

Перехід до "Instances": У консолі Oracle Cloud користувач переходить до розділу "Compute" -> "Instances".

Створення екземпляра: Натискається кнопка "Create Instance".

Назва: Віртуальній машині надається назва.

Розміщення: Обирається "Availability Domain" та "Fault Domain".

Змінити образ: Натискається "Change Image". Переходять на вкладку "Custom Images" та обирається імпортований образ Debian.

Форма (Shape): Вибирається відповідна форма, яка відповідає лімітам "Always Free" (наприклад, VM.Standard.A1.Flex для ARM, якщо імпортовано ARM-образ Debian). Необхідно переконатися, що ліміти не перевищуються (наприклад, 4 vCPUs та 24 GB RAM для A1.Flex).

4. Мережа: Налаштування мережі залишаються за замовчуванням або налаштовуються відповідно до потреб користувача (Virtual Cloud Network - VCN, Subnet).

Ключ SSH: Обов'язково додається публічний ключ SSH. Це єдиний спосіб підключитися до ВМ. Користувач може згенерувати пару ключів (приватний та публічний) за допомогою ssh-keygen на своєму комп'ютері.

Том завантаження (Boot Volume): Розмір залишається за замовчуванням або змінюється в межах Free Tier (до 200 ГБ).

Створення: Натискається "Create".

5. Налаштування мережевих правил (Security List / Firewall)

Відкриття портів: За замовчуванням більшість портів будуть закриті для вхідних з'єднань. Користувачу необхідно відкрити порти, які він планує використовувати (наприклад, SSH (порт 22), HTTP (порт 80), HTTPS (порт 443)).

Переходять до "Virtual Cloud Networks" (VCN), пов'язаної з ВМ.

Вибирається "Subnet", до якої належить ВМ.

Натискається на "Default Security List" або створюється власний.

Додаються "Ingress Rules" (правила вхідного трафіку) для потрібних портів, вказуючи "Source CIDR" як 0.0.0.0/0 (якщо потрібен доступ з будь-якого IP, але для виробничих систем рекомендується обмежити IP-адресами).

6. Підключення до віртуальної машини Debian через SSH

Отримання публічної IP-адреси: У консолі Oracle Cloud користувач знаходить щойно створений екземпляр ВМ і копіює його публічну IP-адресу.

Підключення: Використовується SSH-клієнт на комп'ютері користувача:

Bash

ssh -i /шлях/до/вашого/приватного/ключа.pem debian@_IP_АДРЕСА

(Зазвичай ім'я користувача для хмарних образів Debian - debian).

7. Оновлення системи та встановлення необхідних пакетів

Після успішного підключення:

Bash

sudo apt update

sudo apt upgrade

Після оновлення пакетів встановлюється OpenVPN сервер. (рис. 3.1)

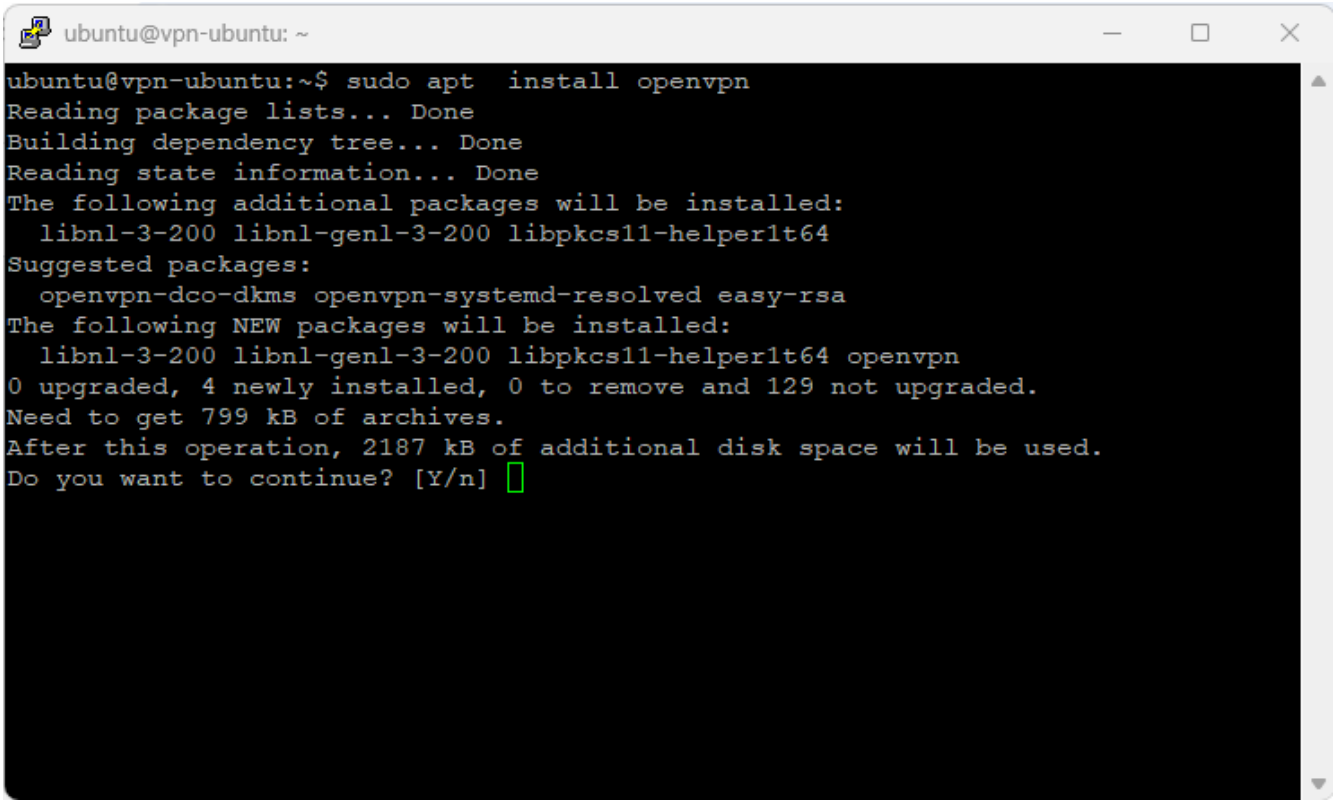

Рисунок 3.1 – Встанвленнння OpenVPN

Після успішного встановлення (рис. 3.2), встановлюється пакет easy-rsa (рис. 3.3)

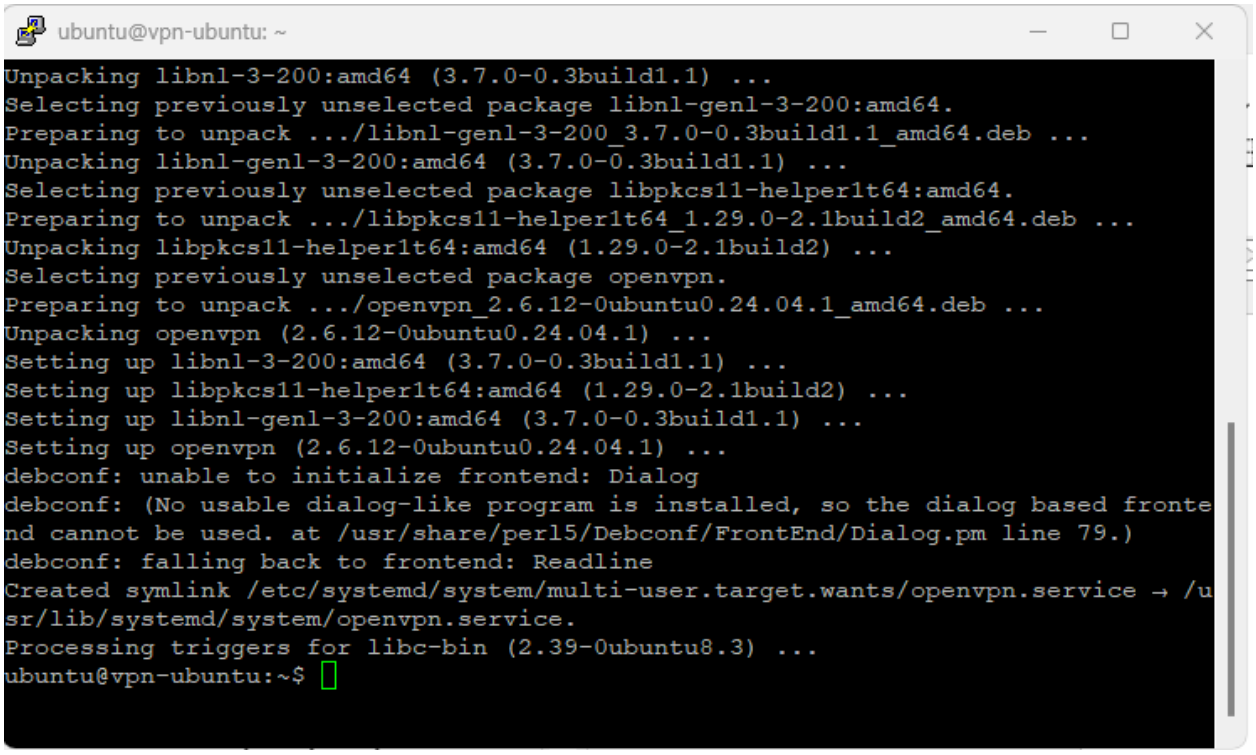

Рисунок 3.2 – Встанвленння OpenVPN

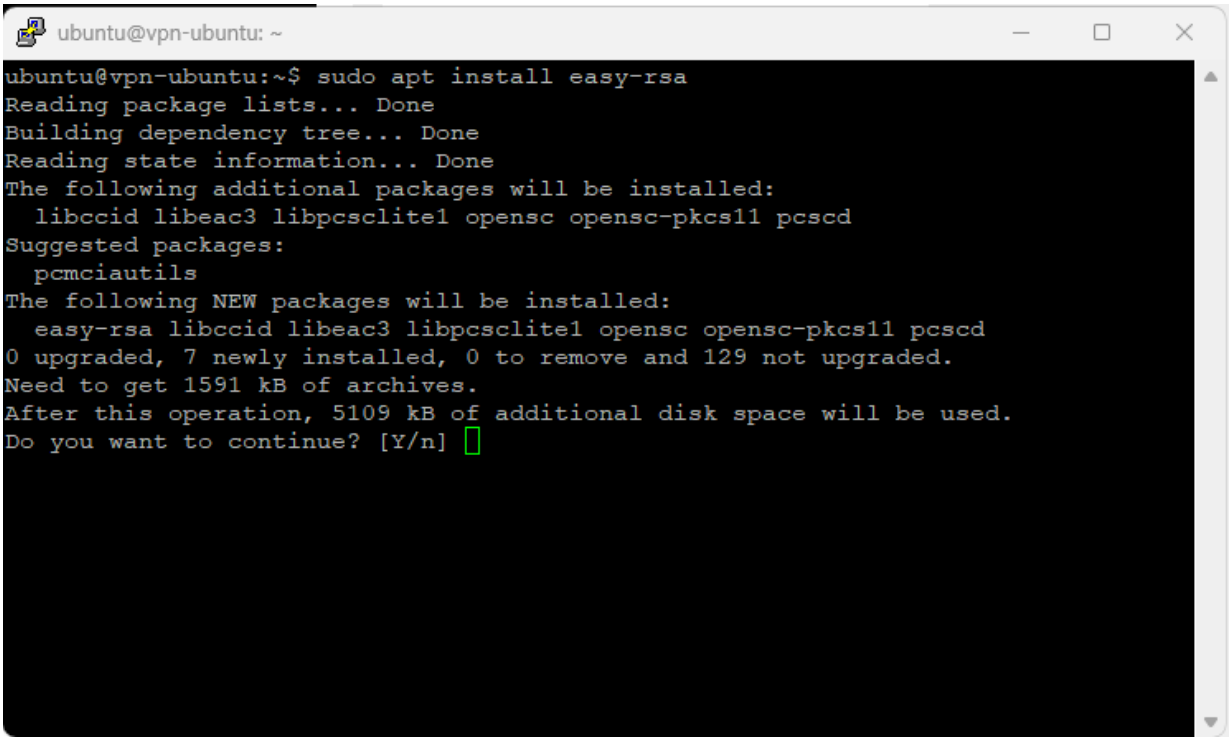

Рисунок 3.3 – Встанвленння easy-rsa

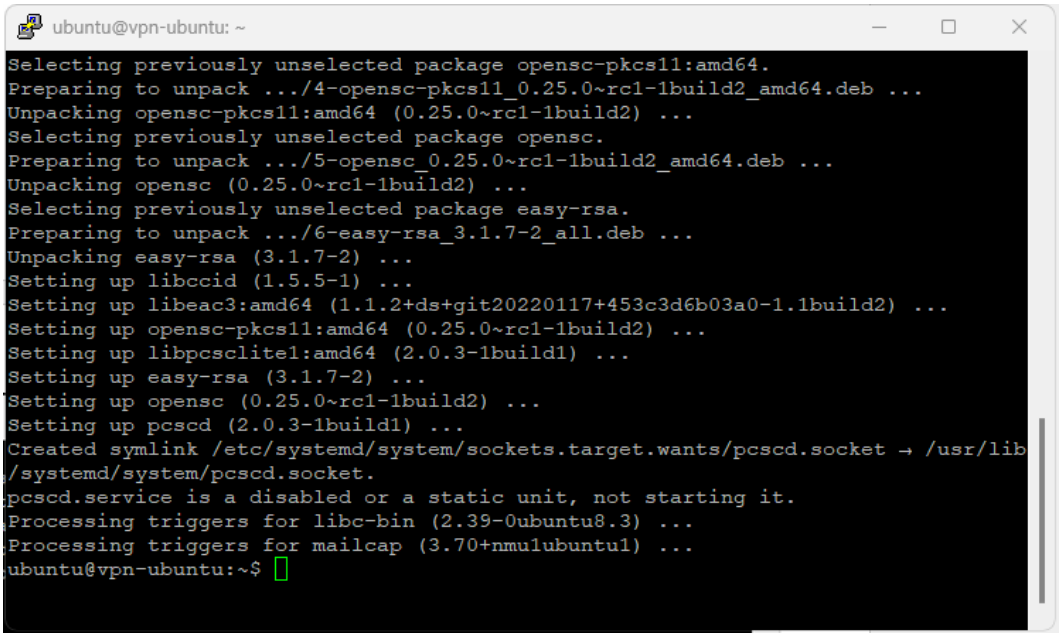

Рисунок 3.4 – Встанвленння easy-rsa

Після успішного встановлення (рис. 3.4), необхідно заповнити файл vars, та можна продовжити створення каталогу PKI. Для цього запустити скрипт easyrsa з параметром init-pki (рис. 3.5):

./easyrsa init-pki

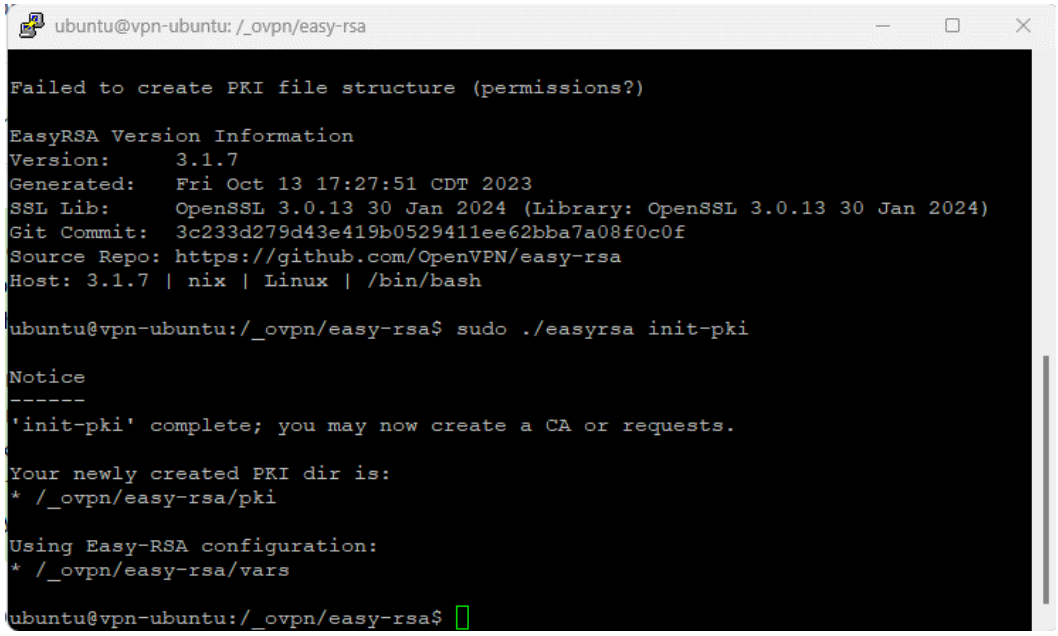

Рисунок 3.5 – Налаштування easy-rsa

Далі створюється сертифікат CA (рис. 3.6):

sudo ./easyrsa build-ca nopass

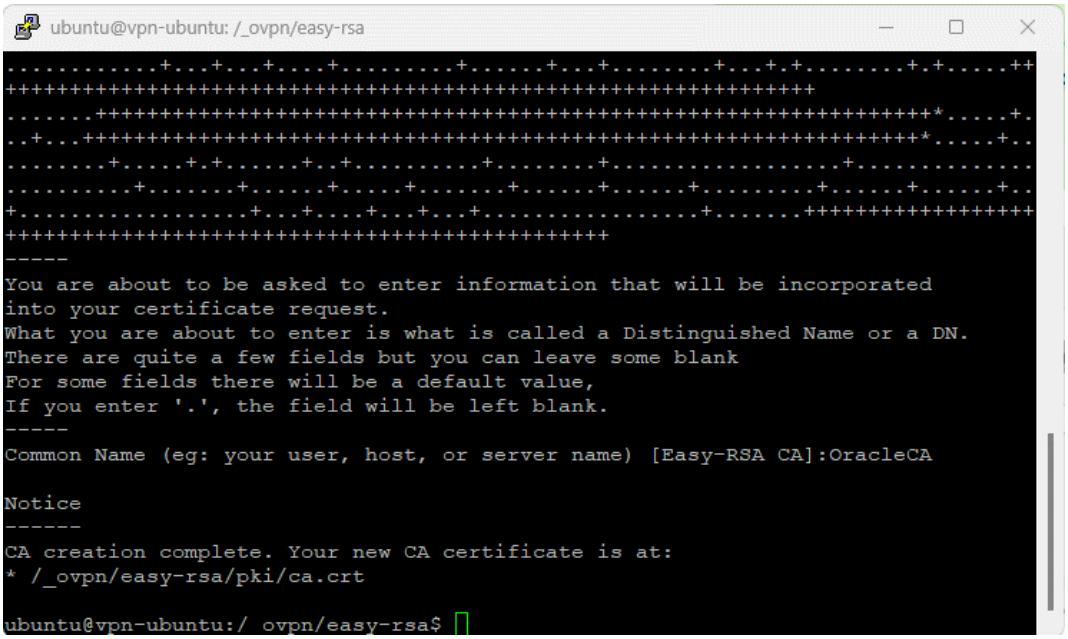

Рисунок 3.6 – Створення CA сертифікату

Далі створюється сертифікат серверу (рис. 3.7):

Sudo ./easyrsa gen-req server nopass

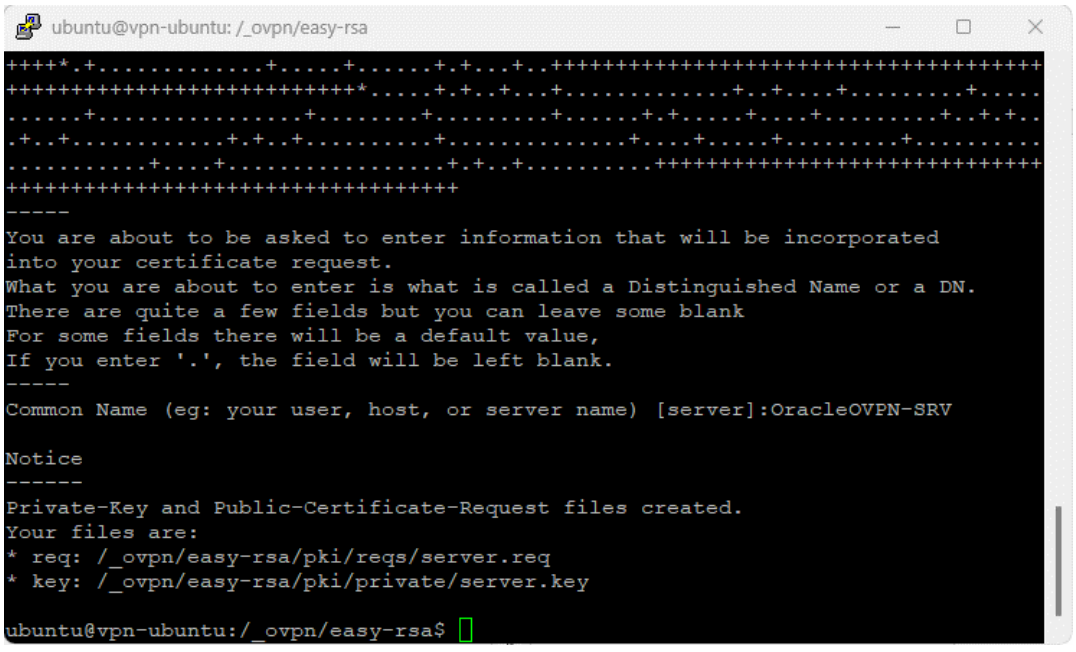

Рисунок 3.7 – Створення сертифікату серверу

Щоб згенерувати попередньо наданий ключ tls-crypt, виконуєтьс наступна команду на сервері OpenVPN у каталозі ~/easy-rsa (рис. 3.8):

cd ~/easy-rsa

openvpn --genkey --secret ta.key

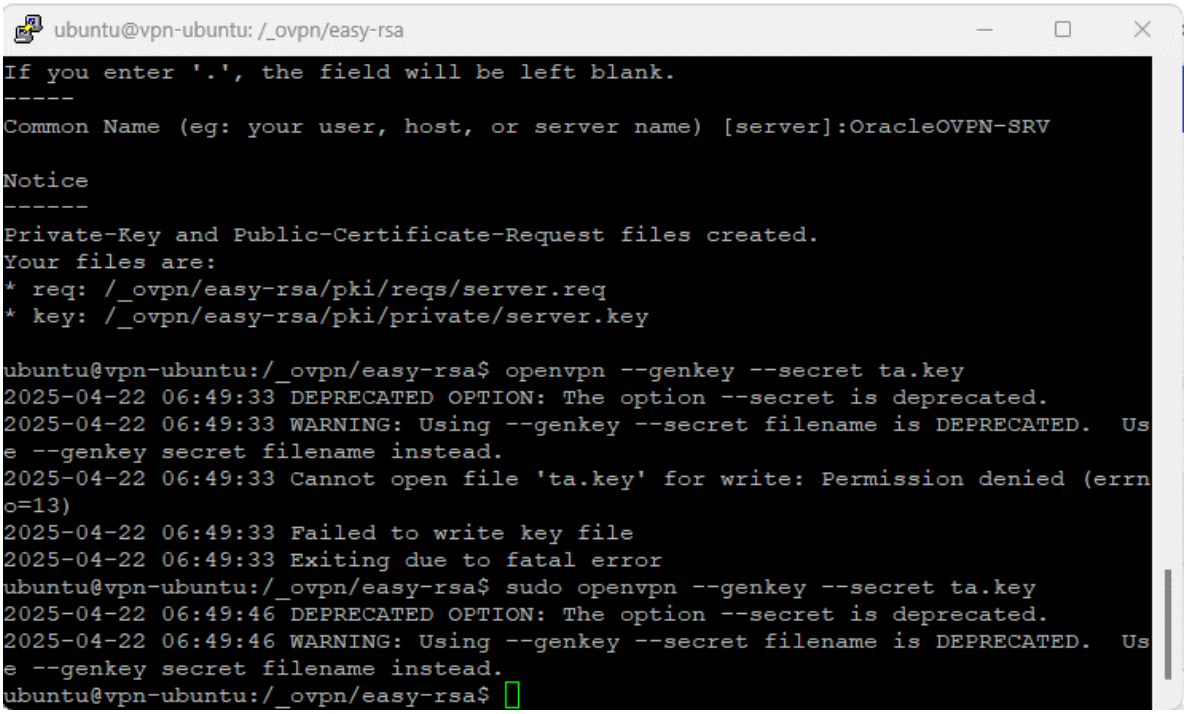

Рисунок 3.8 – Створення ключа tls-crypt

Далі генерується сертифікат клієнта (рис. 3.9):

/easyrsa build-client-full <CLIENT_NAME> nopass

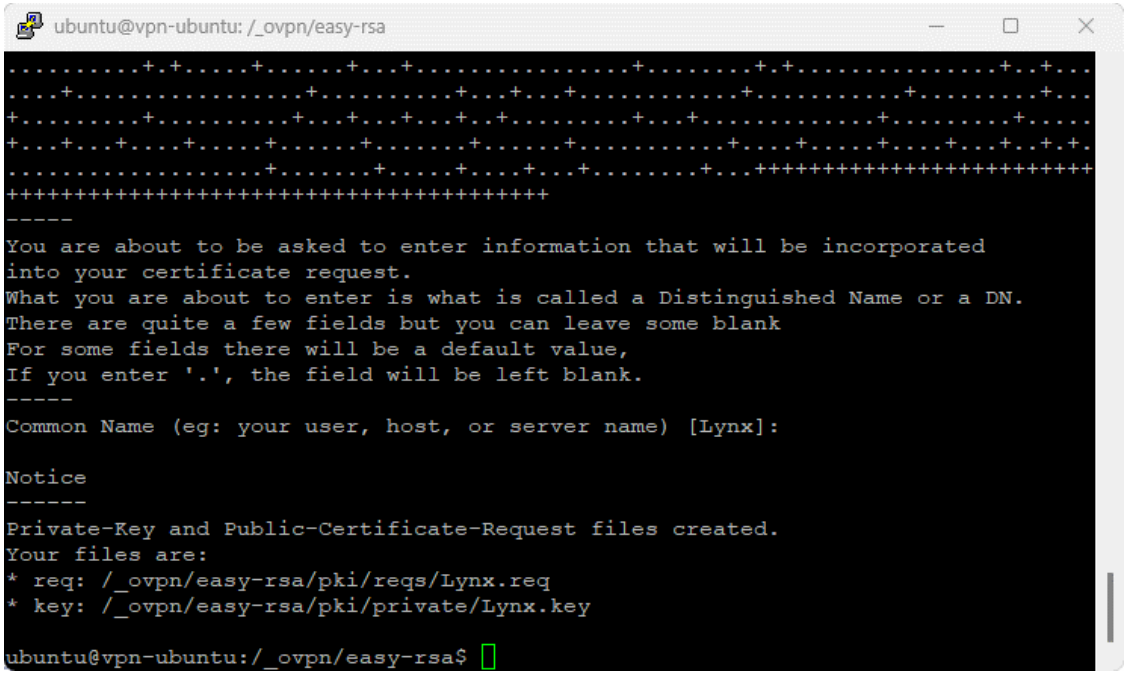

Рисунок 3.9 – Створення сертифікату клієнта

Після створення всіх необхідних ключів і сертифікатів, створюється файл конфігурації серверу (рис. 3.10):

Nano /etc/openvpn/server.conf

Також необхідно налаштувати брандмауер за допомогою iptables:

ubuntu@vpn-ubuntu:/_ovpn/easy-rsa$ sudo iptables -A INPUT -p udp -m udp --dport 1194 -j ACCEPT

ubuntu@vpn-ubuntu:/_ovpn/easy-rsa$ sudo iptables -t nat -A POSTROUTING -o eth0 -s 10.8.0.0/24 -j MASQUERADE

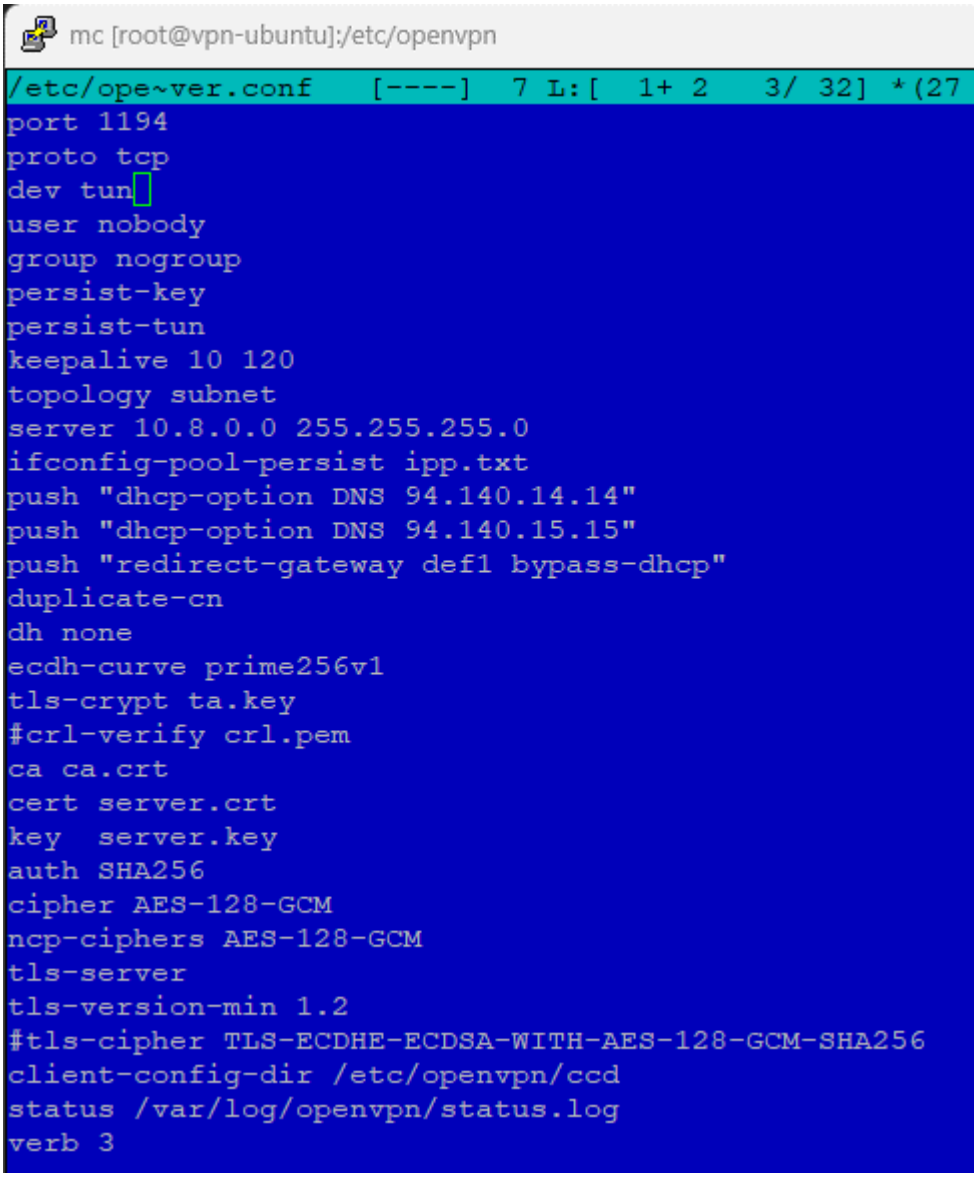

Рисунок 3.10 – Файл конфігурації серверу

3.2 Налаштування маршрутизатору

Для підключення маршрутизатора MikroTik до OpenVPN-сервера, розгорнутого у хмарній інфраструктурі (наприклад, Oracle Cloud), необхідно виконати кілька послідовних кроків:

1. Вимоги до системи

– OpenVPN-сервер із білим IP (наприклад, Oracle Cloud Free Tier).

– MikroTik RouterOS версії 6.45+ (OpenVPN працює лише з TCP, підтримує тільки AES-128-CBC і SHA1).

– Сертифікати OpenVPN-серверу (сертифікат CA, користувача, ключ користувача).

– Налаштований порт (зазвичай TCP 1194) відкритий у фаєрволі та cloud-сек'юріті Oracle.

2. Отримання необхідних файлів для MikroTik

З сервера необхідно забрати:

– ca.crt — сертифікат центру сертифікації;

– client.crt — сертифікат клієнта (маршрутизатора);

– client.key — приватний ключ клієнта.

Сертифікати необхідно (клієнтські сертифікат та ключ (client.crt, client.key та за бажанням кореневої (ca.crt)) на роутер:

3. Заходимо на Mikrotik за допомогою утиліти winbox, переходимо в розділ Files і копіюємо туди сертифікати: client.crt, client.key та ca.crt (рис. 3.11).

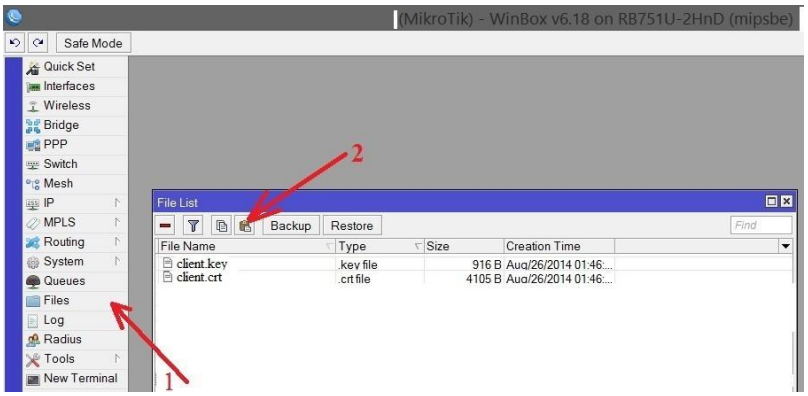

Рисунок 3.11 – Добавання сертифікатів на маршрутизатор

4. Після цього здійснюється імпорт сертифікатів в розділі System – Certificates, вибирається по черзі сертифікати зі списку (client.crt->client.key(-> ca.crt)) і натискається кнопку Import (рис. 3.12).

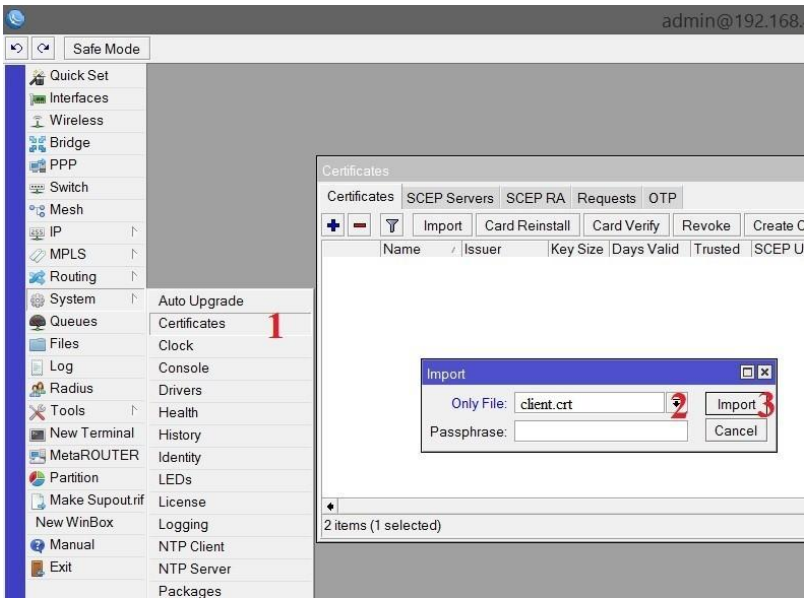

Рисунок 3.12 – Імпорт сертифікатів

5. Отримуємо наступну структуру (рис. 3.13):

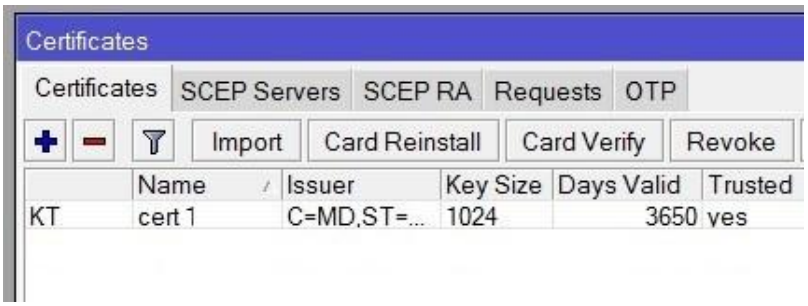

Рисунок 3.13 – Імпорт сертифікатів

6. Саме з'єднання OpenVPN налаштовується в меню PPP->Натискається кнопку «+» -> у списку обирається OVPN Client

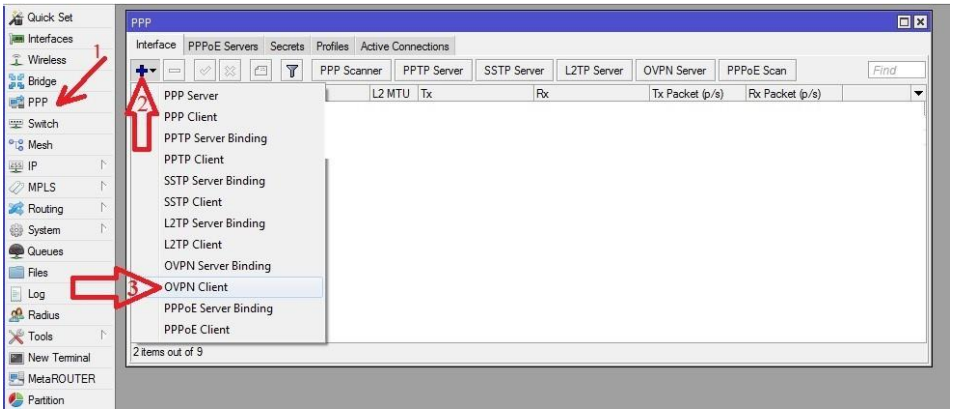

Рисунок 3.14 – Налаштування з’єднання OpenVPN

7. На вкладці Dial Out вказується адреса сервера, логін/пароль, порт, клієнтський сертифікат та тип шифрування.

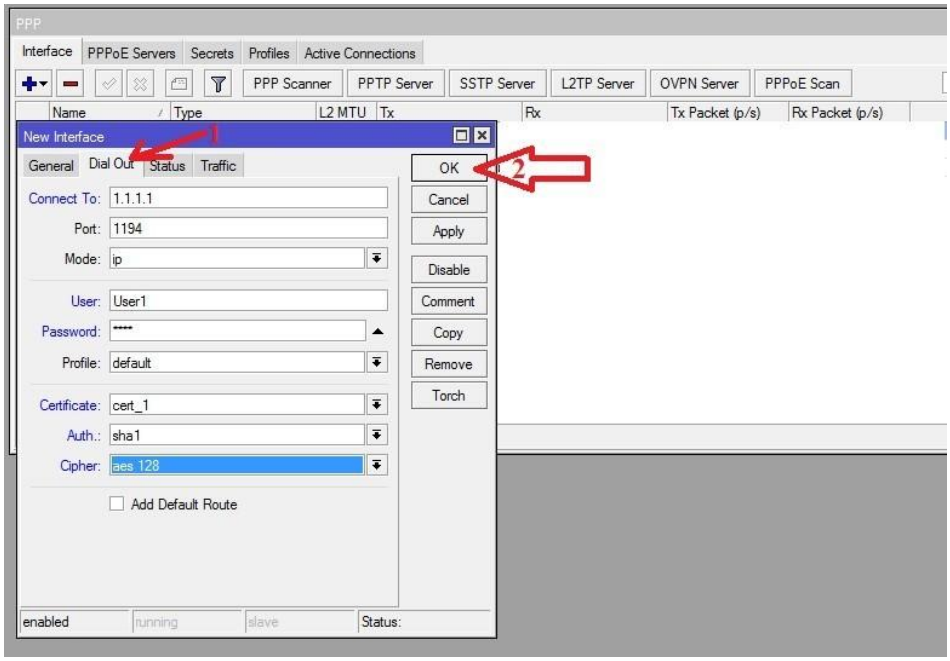

Рисунок 3.15 – Налаштування з’єднання OpenVPN

8. Після цього можна відкрити термінал і перевірити, чи піднялося з'єднання:

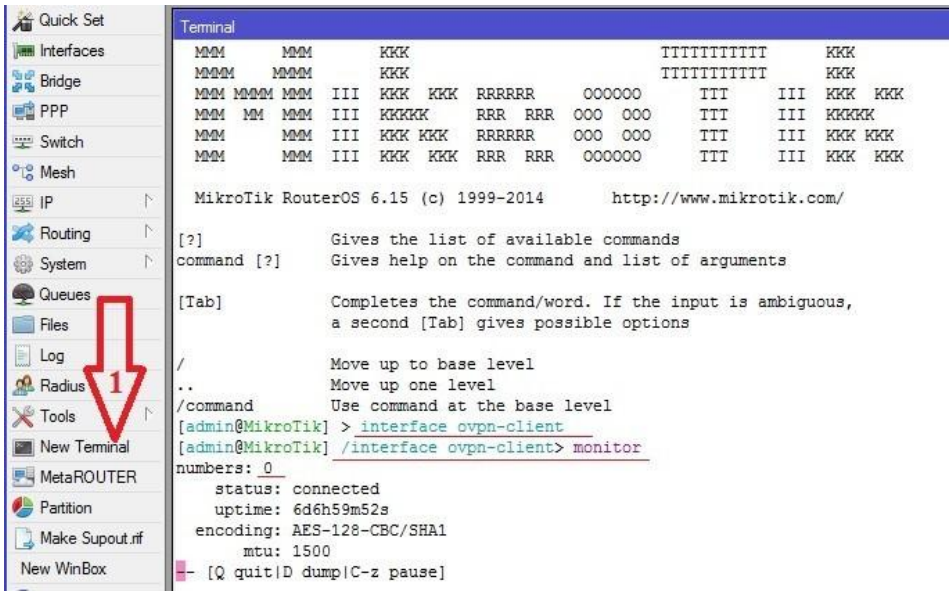

Рисунок 3.15 – Перевірка з’єднання OpenVPN

4 ВИПРОБУВАННЯ

Для перевірки працездатності на персональний комп’ютер було встановлене програмне забезпечення OpenVPN Connect (рис.4.1) та імпортовано профіль *.ovpn.

OpenVPN Connect — це офіційний клієнтський додаток, розроблений OpenVPN Inc., який надає можливість користувачам підключатися до VPN-серверів OpenVPN. Він забезпечує безпечне та зашифроване з'єднання, дозволяючи обходити географічні обмеження та захищати приватність в інтернеті.

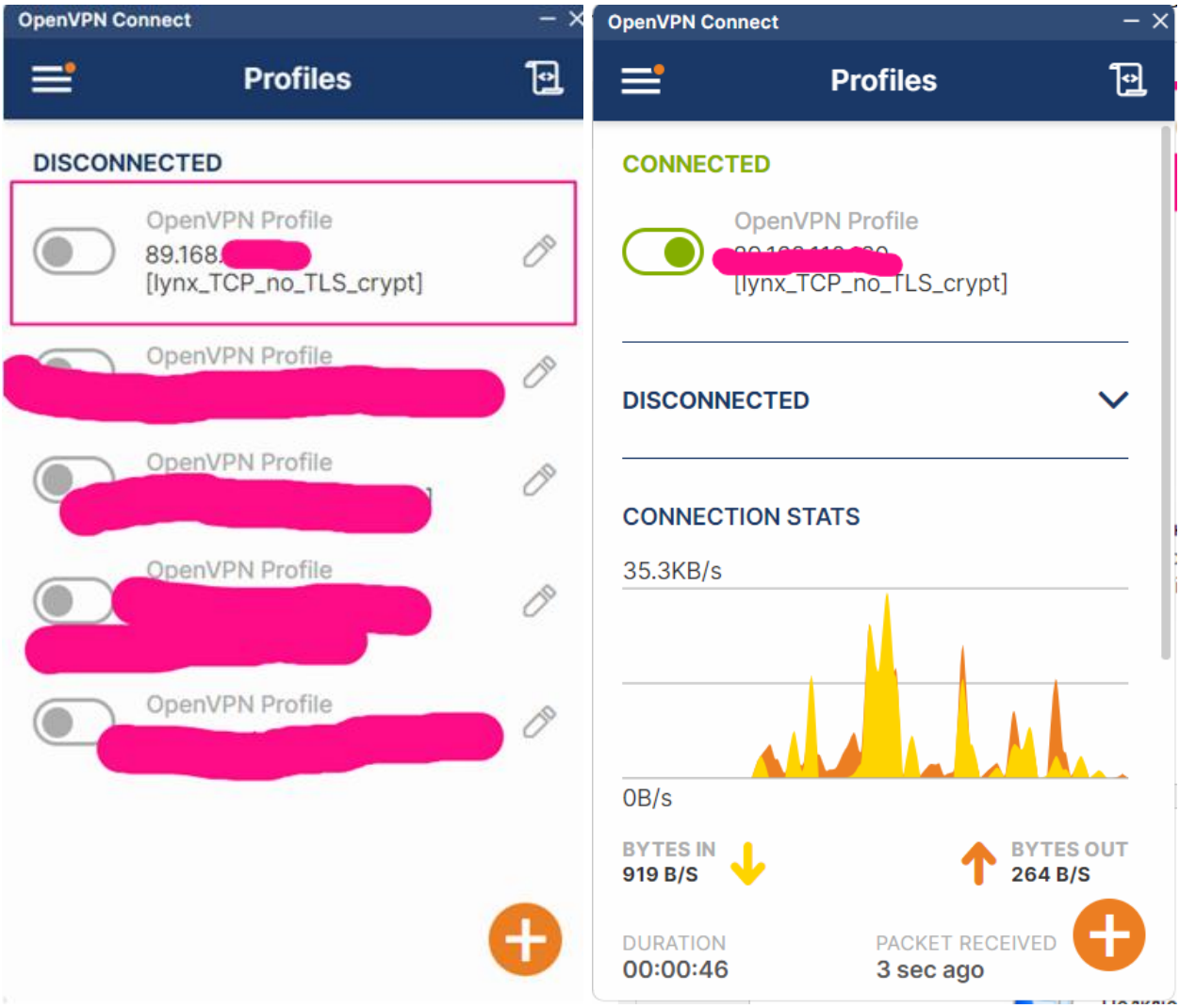

Рисунок 4.1 – Налаштування OpenVPN Connect

Після підключення TAP адаптер, драйвер якого встановлюється разом із OpenVPN Connect, переходить у стан «Підключено».

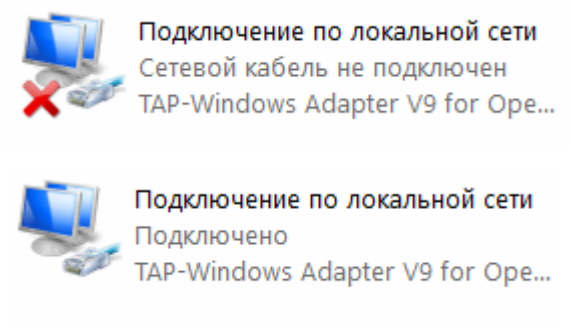

Рисунок 4.2 – TAP адаптер OpevVPN

Якщо перевірити властивості адаптеру, то можна дізнатися IP адресу, отриману від серверу. Вона повинна попадати у діапазон, який вказано у опції server на рис. 3.10.

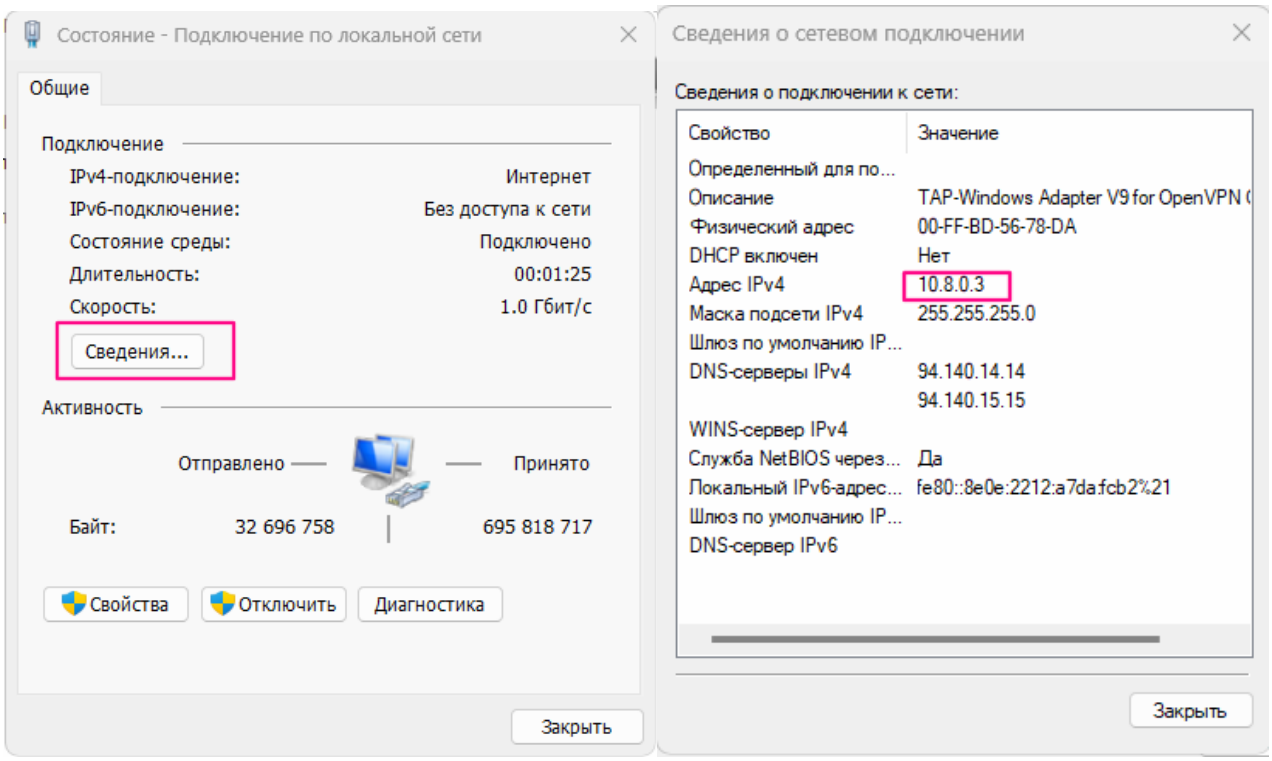

Рисунок 4.3 – TAP адаптер OpenVPN

Так як, маршрутизатор та ПК тепер знаходяться у одній мережі, то можна підключитися до налаштувань маршрутизатору через IP адресу, що видана OpenVPN сервером (рис. 4.4).

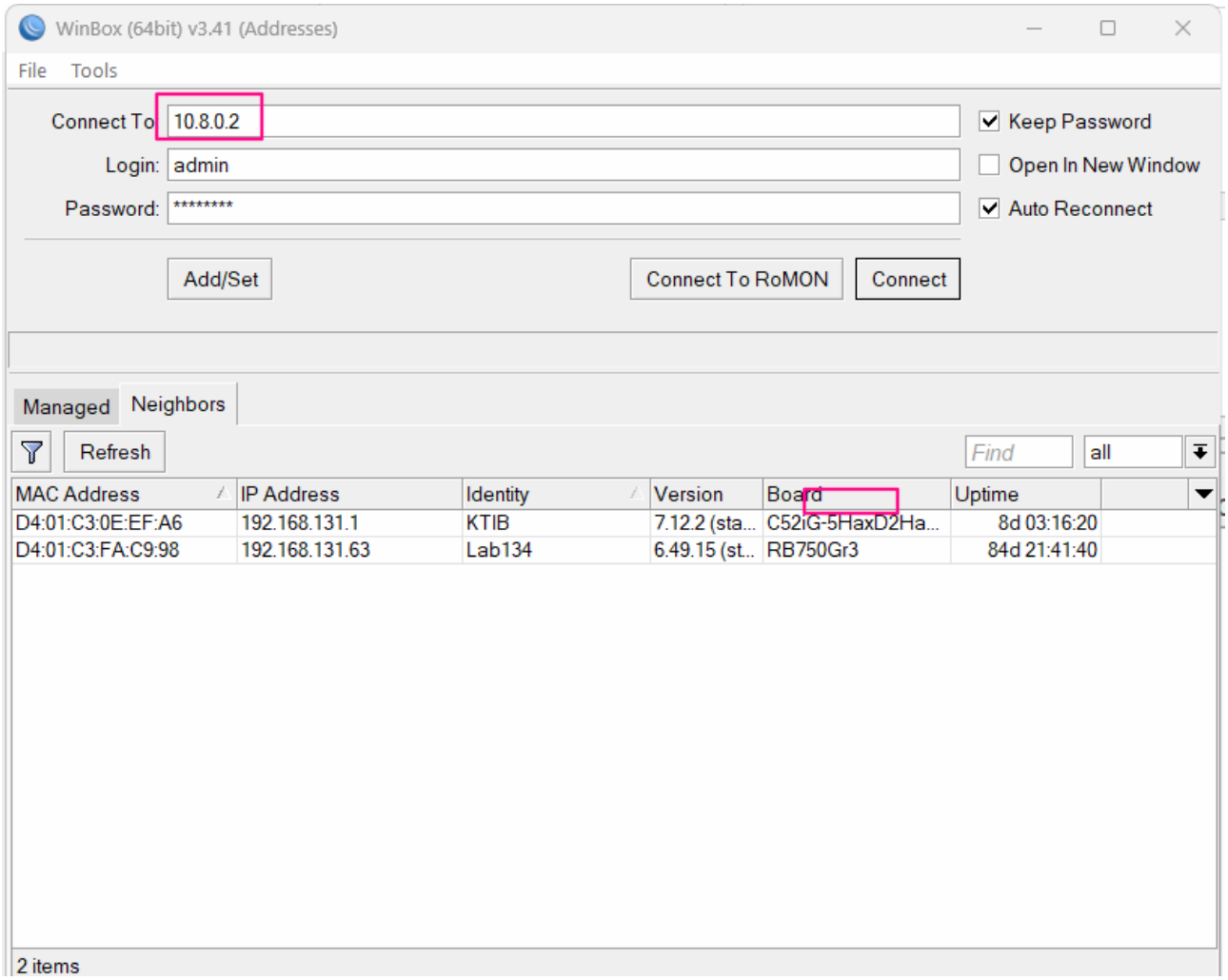

Рисунок 4.4 – Підключення до налаштувань маршрутизатору

На вкладці PPP (рис. 4.5) видно, що інтерфейс opvpn знаходится у стані «Running».

На вкладці IP – addresses можна перевірити IP адресу, що призначена для інтерфейсу ovpn, вона також повинна попадати у діапазон, який вказано у опції server на рис. 3.10.

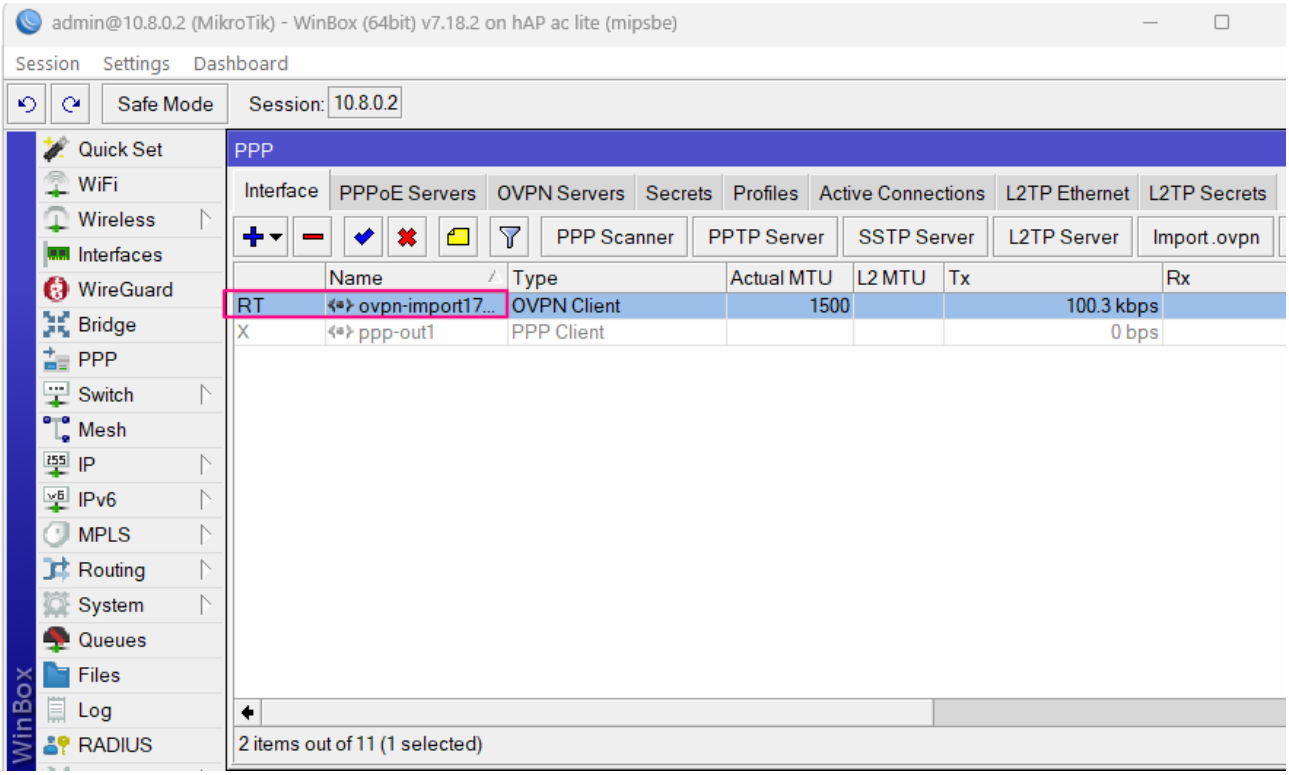

Рисунок 4.5 – PPP інтерфейси

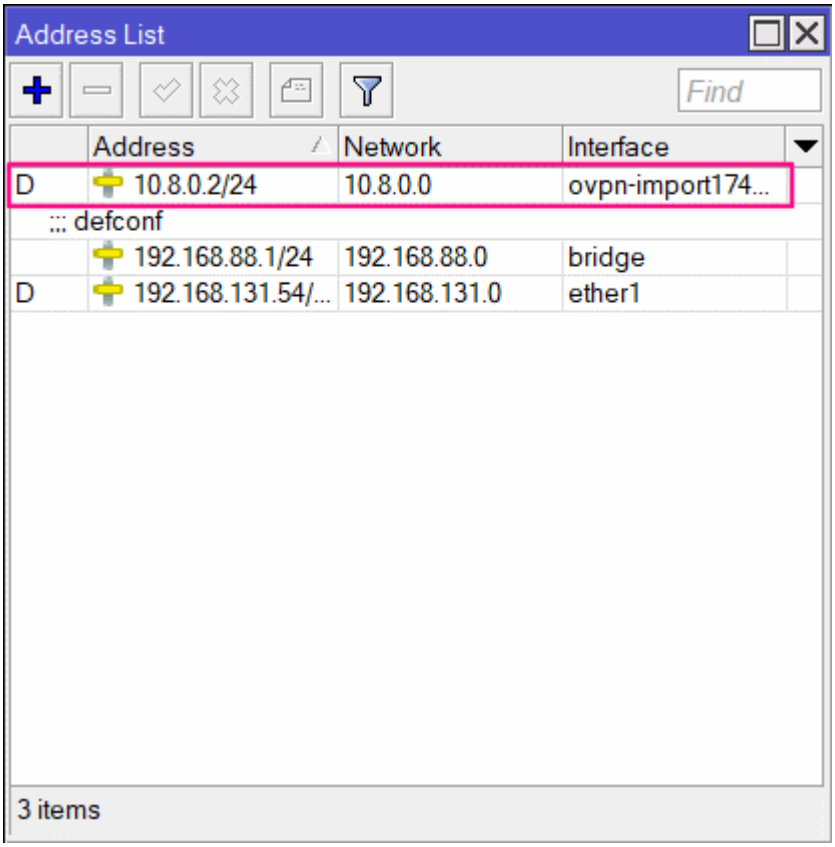

Рисунок 4.6 – IP адреси

Налаштування профілю та сертифікатів показано на рис. 4.7 – 4.8.

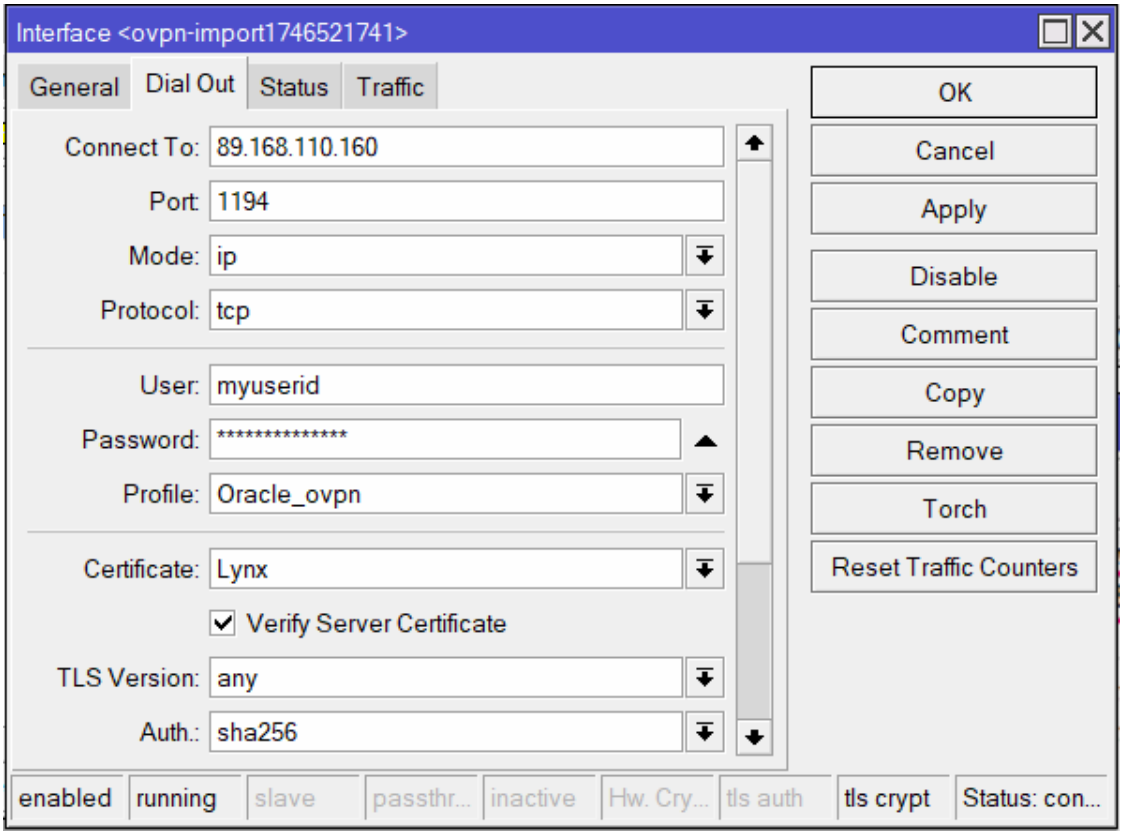

Рисунок 4.7 – Профіль ovpn

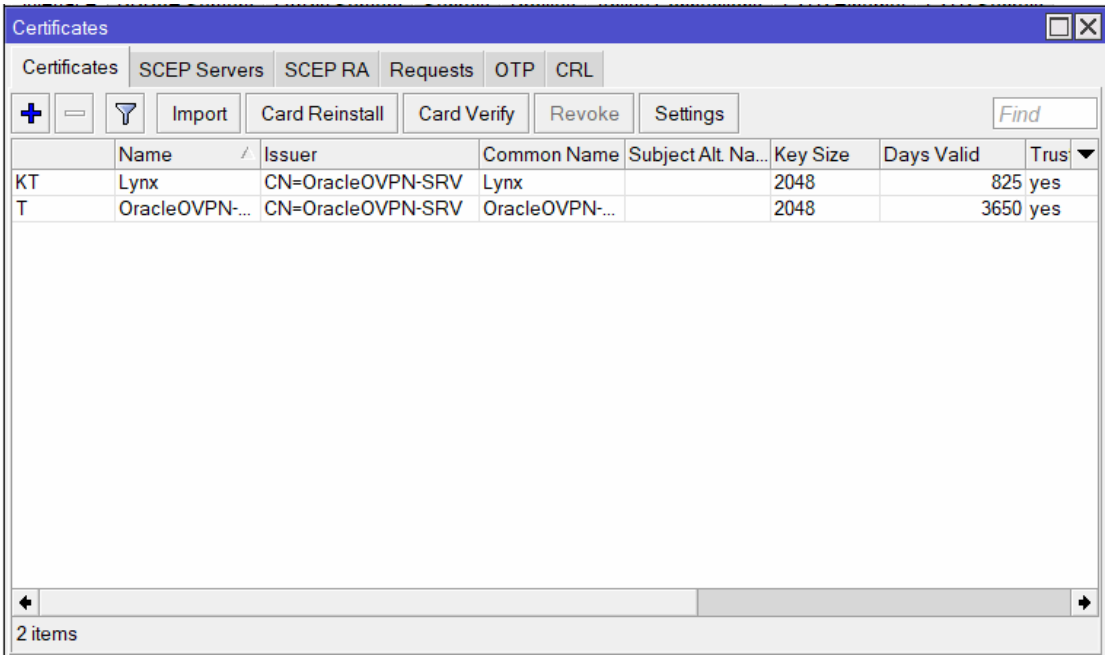

Рисунок 4.8 – Сертифікати

Оскільки маршрутизатор оснащений USB портом, то підключення пристрою із послідовним портом для дистанційного керування не викликає складностей (рис. 4.9 – 4.10).

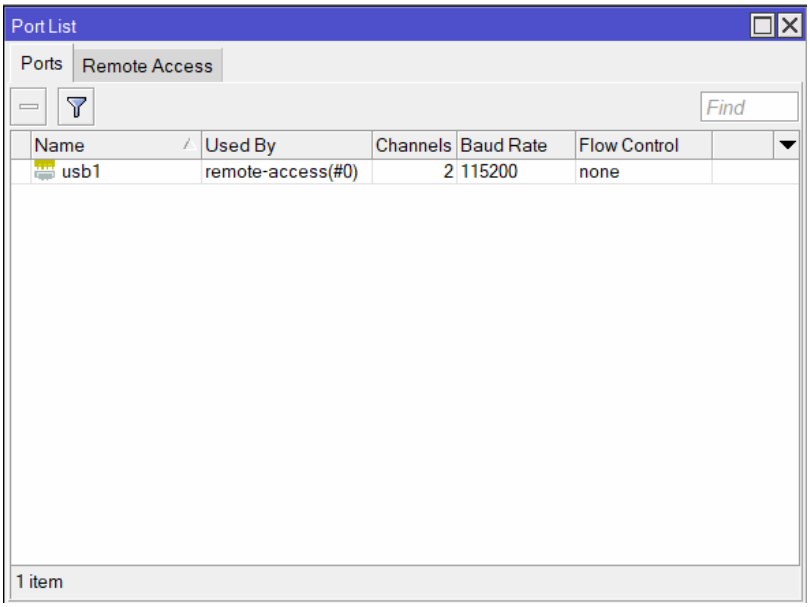

Рисунок 4.9 – Налаштування USB порту

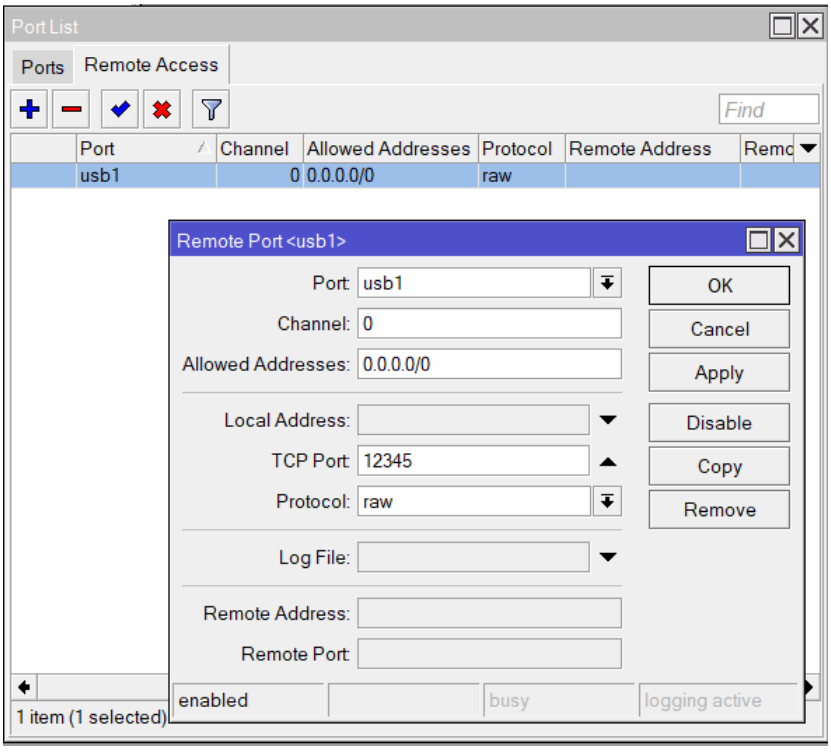

Рисунок 4.10 – Перетворювач Serial – TCP

Якщо програмне забезпечення для відповідного пристрою має можливість підключення через TCP або UDP протокол, то інших налаштувань на стороні ПК не потрібно. Якщо ПЗ працює лише через послідовні порти, то на стороні ПК потрібно встановити ще додаткове ПЗ, що буде перенаправляти дані, які отримано через TCP або UDP протокол у послідовний порт. У такому випадку обмін даними здійснюється за допомогою спеціалізованих драйверів віртуальних послідовних портів.

Досвід тестування подібного програмного забезпечення показав, що найбільш функціональним рішенням є Null-modem emulator. Це драйвер віртуального послідовного порту для Windows, який працює в режимі ядра (kernel-mode driver), є проектом з відкритим вихідним кодом і поширюється під ліцензією GPL.

Null-modem emulator дозволяє створювати необмежену кількість пар віртуальних COM-портів. Кожна така пара складається з двох COM-портів, віртуально з'єднаних між собою каналом даних. Вихід одного COM-порту в парі логічно підключається до входу іншого, і навпаки. Це означає, що будь-яка програма, розроблена для роботи з фізичним COM-портом, може бути пов'язана з іншим додатком, що також використовує COM-порт, і вони зможуть безперешкодно обмінюватися даними.

Єдиним виявленим недоліком цього програмного забезпечення є відсутність цифрового підпису драйверів. Це може ускладнити процес встановлення на сучасних операційних системах Windows, які вимагають підписаних драйверів для забезпечення безпеки та стабільності системи.

Для перевірки стабільності роботи системи з використанням послідовних інтерфейсів було зібрано спеціальний тестовий стенд. На стороні ПК за зв'язок відповідають три пари послідовних портів, об'єднаних єдиним концентратором (як показано на рис. 4.11):

– Пара COM3-TCP відповідає за зв'язок безпосередньо з обладнанням. У цьому налаштуванні IP-адреса перетворювача Ethernet – RS232 є `172.24.10.12`.

– Пара COM5-Software призначена для опитування та передачі параметрів за допомогою існуючого програмного забезпечення. Це дозволяє інтегрувати старе ПЗ без модифікацій.

– Пара COM7-Terminal Software використовується для діагностики та налагодження з'єднань за допомогою будь-якого термінального програмного забезпечення, наприклад, популярної програми PuTTY.

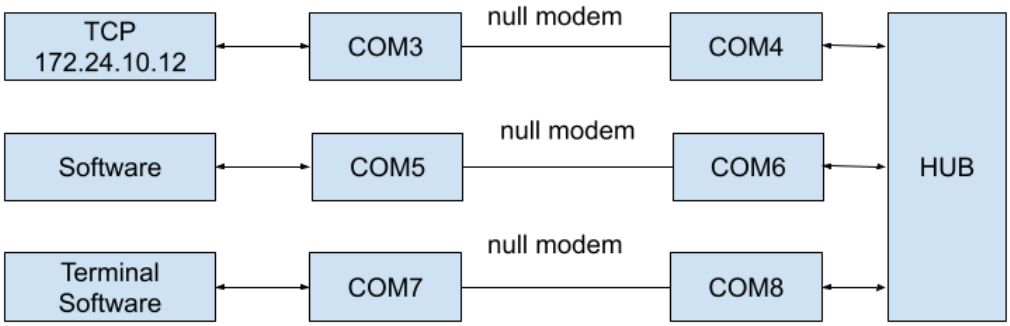

Рисунок 4.11 – Схема послідовних портів

ВИСНОВКИ

У ході виконання дипломного проєкту було розроблено, реалізовано та протестовано систему об’єднання розподілених пристроїв у захищене віртуальне середовище за допомогою технології VPN. Актуальність проєкту визначається зростаючою потребою у безпечній та надійній взаємодії між пристроями, що знаходяться за мережевими екранами та NAT, особливо в умовах відсутності прямого доступу до конфігурації маршрутизаторів користувача.

Як сервер посередник використано OpenVPN, розгорнутий у хмарній інфраструктурі Oracle Cloud, що дало змогу забезпечити постійну доступність, масштабованість та незалежність від сторонніх пропрієтарних сервісів. У результаті виконано такі основні завдання:

– проведено аналіз існуючих рішень для взаємодії пристроїв за NAT;

– налаштовано сервер OpenVPN з відкритими ключами та обмеженим доступом;

– реалізовано підключення клієнтів (у тому числі маршрутизаторів MikroTik) до VPN-тунелю;

– протестовано стабільність з’єднання, маршрутизацію та можливості для захищеної взаємодії між пристроями;

– розглянуто ризики, пов’язані з безпекою даних, та впроваджено базові захисні механізми.

Отримані результати підтвердили працездатність запропонованої архітектури. Побудована система дозволяє організувати безпечний зв’язок між пристроями у різних мережах без потреби у статичних IP-адресах або пробросі портів, що є надзвичайно важливим у багатьох реальних сценаріях — від моніторингу IoT-пристроїв до дистанційного адміністрування вузлів.

ПЕРЕЛІК ПОСИЛАНЬ

1. Postel J. User Datagram Protocol. RFC 768 [Електронний ресурс]. – IETF, 1980. – Режим доступу: https://datatracker.ietf.org/doc/html/rfc768 – Назва з екрана.

2. Kent S., Atkinson R. IP Encapsulating Security Payload (ESP). RFC 4303 [Електронний ресурс]. – IETF, 2005. – Режим доступу: https://datatracker.ietf.org/doc/html/rfc4303 – Назва з екрана.

3. OpenVPN. Official OpenVPN Documentation [Електронний ресурс]. – Режим доступу: https://openvpn.net/community-resources/reference-manual-for-openvpn-2-5 – Назва з екрана.

4. Oracle Cloud Infrastructure. Getting Started with Always Free Services [Електронний ресурс]. – Режим доступу: https://docs.oracle.com/en-us/iaas/Content/FreeTier/overview.htm – Назва з екрана.

5. MikroTik. MikroTik RouterOS Documentation [Електронний ресурс]. – Режим доступу: https://help.mikrotik.com/docs/ – Назва з екрана.

6. NAT Traversal Techniques [Електронний ресурс] // STUN, TURN and ICE overview – Режим доступу: https://webrtc.org/faq/#nat-traversal – Назва з екрана.

7. RFC 1918 – Address Allocation for Private Internets [Електронний ресурс]. – IETF, 1996. – Режим доступу: https://datatracker.ietf.org/doc/html/rfc1918 – Назва з екрана.

ДОДАТОК

Конфігурація маршрутризатора

```
# 2025-06-26 15:12:06 by RouterOS 7.18.2
# software id = WRLK-1EKM
#
# model = RB952Ui-5ac2nD
# serial number = CC3E0E17499E
/interface bridge
add admin-mac=2C:C8:1B:DC:46:0E auto-mac=no comment=defconf name=bridge \
    port-cost-mode=short
/interface ethernet
set [ find default-name=ether1 ] mtu=1400
/interface list
add comment=defconf name=WAN
add comment=defconf name=LAN
/interface wireless security-profiles
set [ find default=yes ] supplicant-identity=MikroTik
add authentication-types=wpa2-psk eap-methods="" mode=dynamic-keys name=Lynx \
    supplicant-identity=MikroTik
add authentication-types=wpa-psk,wpa2-psk eap-methods="" mode=dynamic-keys \
    name=KTIB-129 supplicant-identity=""
/interface wireless
set [ find default-name=wlan1 ] band=2ghz-b/g/n channel-width=20/40mhz-XX \
    country=ukraine distance=indoors frequency=auto installation=indoor \
    security-profile=KTIB-129 ssid=KTIB-Lab129 wireless-protocol=802.11
set [ find default-name=wlan2 ] band=5ghz-a/n/ac channel-width=\
    20/40/80mhz-XXXX country=ukraine distance=indoors frequency=auto mode=\
    ap-bridge security-profile=Lynx ssid=Lynx wireless-protocol=802.11 \
    wps-mode=disabled
/ip pool
add name=default-dhcp ranges=192.168.88.10-192.168.88.254
/ip dhcp-server
add address-pool=default-dhcp interface=bridge lease-time=10m name=defconf
/port
set 0 baud-rate=115200
/interface ppp-client
add apn=internet name=ppp-out1 port=usb1
/ppp profile
add change-tcp-mss=yes interface-list=LAN name=Oracle_ovpn
/interface ovpn-client
add auth=sha256 certificate=Lynx cipher=aes128-gcm connect-to=89.168.110.160 \
    mac-address=FE:CB:32:B3:45:D0 name=ovpn-import1746521741 profile=\
    Oracle_ovpn route-nopull=yes user=myuserid verify-server-certificate=yes
/interface bridge port
add bridge=bridge comment=defconf interface=ether2 internal-path-cost=10 \
    path-cost=10
add bridge=bridge comment=defconf interface=ether3 internal-path-cost=10 \
    path-cost=10
add bridge=bridge comment=defconf interface=ether4 internal-path-cost=10 \
    path-cost=10
add bridge=bridge comment=defconf interface=ether5 internal-path-cost=10 \
    path-cost=10
add bridge=bridge comment=defconf interface=wlan2 internal-path-cost=10 \
```

```
   path-cost=10
/ip firewall connection tracking
set udp-timeout=10s
/ip neighbor discovery-settings
set discover-interface-list=LAN
/interface list member
add comment=defconf interface=bridge list=LAN
add comment=defconf interface=ether1 list=WAN
add interface=wlan1 list=WAN
/interface ovpn-server server
add mac-address=FE:E1:A7:91:1B:3D name=ovpn-server1
/ip address
add address=192.168.88.1/24 comment=defconf interface=bridge network=\
   192.168.88.0
/ip dhcp-client
add comment=defconf interface=ether1
# Interface not active
add comment=defconf interface=wlan1
/ip dhcp-server lease
add address=192.168.88.250 client-id=1:e0:cc:f8:f7:ab:5a mac-address=\
   E0:CC:F8:F7:AB:5A server=defconf
/ip dhcp-server network
add address=192.168.88.0/24 comment=defconf dns-server=192.168.88.1 gateway=\
   192.168.88.1
/ip dns
set allow-remote-requests=yes
/ip dns static
add address=192.168.88.1 comment=defconf name=router.lan type=A
/ip firewall filter
add action=accept chain=input comment=\
   "defconf: accept established,related,untracked" connection-state=\
   established,related,untracked
add action=drop chain=input comment="defconf: drop invalid" connection-state=\
   invalid
add action=accept chain=input comment="defconf: accept ICMP" protocol=icmp
add action=accept chain=input comment=\
   "defconf: accept to local loopback (for CAPsMAN)" dst-address=127.0.0.1
add action=drop chain=input comment="defconf: drop all not coming from LAN" \
   in-interface-list=!LAN
add action=accept chain=forward comment="defconf: accept in ipsec policy" \
   ipsec-policy=in,ipsec
add action=accept chain=forward comment="defconf: accept out ipsec policy" \
   ipsec-policy=out,ipsec
add action=fasttrack-connection chain=forward comment="defconf: fasttrack" \
   connection-state=established,related hw-offload=yes
add action=accept chain=forward comment=\
   "defconf: accept established,related, untracked" connection-state=\
   established,related,untracked
add action=drop chain=forward comment="defconf: drop invalid" \
   connection-state=invalid
add action=drop chain=forward comment=\
   "defconf: drop all from WAN not DSTNATed" connection-nat-state=!dstnat \
   connection-state=new in-interface-list=WAN
/ip firewall mangle
add action=change-mss chain=postrouting new-mss=clamp-to-pmtu out-interface=\
   all-ppp protocol=tcp tcp-flags=syn
```

```
/ip firewall nat
add action=masquerade chain=srcnat comment="defconf: masquerade" \
    ipsec-policy=out,none out-interface-list=WAN
/ipv6 firewall address-list
add address=::/128 comment="defconf: unspecified address" list=bad_ipv6
add address=::1/128 comment="defconf: lo" list=bad_ipv6
add address=fec0::/10 comment="defconf: site-local" list=bad_ipv6
add address=::ffff:0.0.0.0/96 comment="defconf: ipv4-mapped" list=bad_ipv6
add address=::/96 comment="defconf: ipv4 compat" list=bad_ipv6
add address=100::/64 comment="defconf: discard only " list=bad_ipv6
add address=2001:db8::/32 comment="defconf: documentation" list=bad_ipv6
add address=2001:10::/28 comment="defconf: ORCHID" list=bad_ipv6
add address=3ffe::/16 comment="defconf: 6bone" list=bad_ipv6
/ipv6 firewall filter
add action=accept chain=input comment=\
    "defconf: accept established,related,untracked" connection-state=\
    established,related,untracked
add action=drop chain=input comment="defconf: drop invalid" connection-state=\
    invalid
add action=accept chain=input comment="defconf: accept ICMPv6" protocol=\
    icmpv6
add action=accept chain=input comment="defconf: accept UDP traceroute" port=\
    33434-33534 protocol=udp
add action=accept chain=input comment=\
    "defconf: accept DHCPv6-Client prefix delegation." dst-port=546 protocol=\
    udp src-address=fe80::/10
add action=accept chain=input comment="defconf: accept IKE" dst-port=500,4500 \
    protocol=udp
add action=accept chain=input comment="defconf: accept ipsec AH" protocol=\
    ipsec-ah
add action=accept chain=input comment="defconf: accept ipsec ESP" protocol=\
    ipsec-esp
add action=accept chain=input comment=\
    "defconf: accept all that matches ipsec policy" ipsec-policy=in,ipsec
add action=drop chain=input comment=\
    "defconf: drop everything else not coming from LAN" in-interface-list=\
    !LAN
add action=accept chain=forward comment=\
    "defconf: accept established,related,untracked" connection-state=\
    established,related,untracked
add action=drop chain=forward comment="defconf: drop invalid" \
    connection-state=invalid
add action=drop chain=forward comment=\
    "defconf: drop packets with bad src ipv6" src-address-list=bad_ipv6
add action=drop chain=forward comment=\
    "defconf: drop packets with bad dst ipv6" dst-address-list=bad_ipv6
add action=drop chain=forward comment="defconf: rfc4890 drop hop-limit=1" \
    hop-limit=equal:1 protocol=icmpv6
add action=accept chain=forward comment="defconf: accept ICMPv6" protocol=\
    icmpv6
add action=accept chain=forward comment="defconf: accept HIP" protocol=139
add action=accept chain=forward comment="defconf: accept IKE" dst-port=\
    500,4500 protocol=udp
add action=accept chain=forward comment="defconf: accept ipsec AH" protocol=\
    ipsec-ah
add action=accept chain=forward comment="defconf: accept ipsec ESP" protocol=\
```

```
   ipsec-esp
add action=accept chain=forward comment=\
   "defconf: accept all that matches ipsec policy" ipsec-policy=in,ipsec
add action=drop chain=forward comment=\
   "defconf: drop everything else not coming from LAN" in-interface-list=\
   !LAN
/port remote-access
add port=usb1 protocol=raw tcp-port=12345
/system clock
set time-zone-name=Europe/Kiev
/system note
set show-at-login=no
/system ntp client
set enabled=yes
/tool mac-server
set allowed-interface-list=LAN
/tool mac-server mac-winbox
set allowed-interface-list=LAN
```