ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
КАФЕДРА КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ

КВАЛІФІКАЦІЙНА РОБОТА

на тему: «АВТОМАТИЗОВАНА СИСТЕМА ПІДКЛЮЧЕННЯ КОРИСТУВАЧІВ ДО ВНУТРІШНІХ РЕСУРСІВ ЗА ДОПОМОГОЮ OPENVPN»

на здобуття освітнього ступеня бакалавра
зі спеціальності 123 Комп'ютерна інженерія
(код, найменування спеціальності)
освітньо-професійної програми «Комп'ютерна інженерія»
(назва)

Кваліфікаційна робота містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело

_____ *Андрій КРАСЮК*
(підпис) *Ім'я, ПРІЗВИЩЕ здобувача*

Виконав: здобувач вищої освіти гр. КІД-41
Андрій КРАСЮК
Ім'я, ПРІЗВИЩЕ

Керівник: Ярослав ТОРОШАНКО
науковий ступінь, вчене звання Ім'я, ПРІЗВИЩЕ

Рецензент: _____
науковий ступінь, вчене звання Ім'я, ПРІЗВИЩЕ

Київ 2026

ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ

Навчально-науковий інститут Інформаційних технологій

Кафедра Комп'ютерної інженерії

Ступінь вищої освіти бакалавр

Спеціальність 123 Комп'ютерна інженерія

Освітньо-професійна програма «Комп'ютерна інженерія»

ЗАТВЕРДЖУЮ
Завідувач кафедри
Комп'ютерної інженерії
_____ Наталія ЛАЩЕВСЬКА
«_____» _____ 2026 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

Красюк Андрій Валерійович
(прізвище, ім'я, по батькові здобувача)

1. Тема кваліфікаційної роботи: «Автоматизована система підключення користувачів до внутрішніх ресурсів за допомогою OpenVPN»
керівник кваліфікаційної роботи: Ярослав ТОРОШАНКО, кандидат технічних наук, доцент, старший науковий співробітник ,
(Ім'я, ПРІЗВИЩЕ, науковий ступінь, вчене звання)
затверджені наказом Державного університету інформаційно-комунікаційних технологій від «20» лютого 2026р. №51
2. Строк подання кваліфікаційної роботи«14» травня 2026р.
3. Вихідні дані до кваліфікаційної роботи: науково-технічна література, стандарти та протоколи VPN, методи тунелювання і шифрування з механізмами автентифікації, клієнт-серверна архітектура, підходи до побудови захищених з'єднань, методи розгортання та адміністрування VPN-серверів, засоби автоматизації конфігурацій, а також аналіз продуктивності та ефективності систем у нестабільних умовах.
4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)
    1. Теоретичні основи організації захищеного доступу до внутрішніх мереж
    2. Проєктування та реалізація системи підключення користувачів до внутрішніх ресурсів
    3. Аналіз ефективності та перспективи розвитку системи
5. Перелік ілюстративного матеріалу: *презентація*
    1. Мета, об'єкт, предмет дослідження
    2. Актуальність роботи
    3. Архітектура запропонованого рішення
    4. Практична реалізація (сервер на базі вживаного обладнання)
    5. Особливості реалізованого рішення
    6. Алгоритми у реалізованому рішенні
    7. Інтеграція автоматизації у формуванні конфігурацій
    8. Практичне використання ресурсів через встановлений тунель

9. Навантаження під час використання ресурсів двома клієнтами
10. Аналіз продуктивності між клієнтом та сервером через iperf3
11. Методи оптимізації роботи реалізованого рішення
12. Економічна ефективність обраного рішення
13. Висновки
14. Публікації та апробація роботи

6. Дата видачі завдання «20» лютого 2026р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Збір даних	22.02-20.03.2026	Виконано
2	Обробка матеріалу	20.03-30.03.2026	Виконано
3	Розробка теоретичної частини	31.03-10.04.2026	Виконано
4	Проєктування та реалізація системи підключення користувачів до внутрішніх ресурсів	11.04-20.04.2026	Виконано
5	Висновки за результатами аналізу	21.04-26.04.2026	Виконано
6	Оформлення пояснювальної записки	26.04-28.04.2026	Виконано
7	Розробка презентації	29.04-01.05.2026	Виконано
8	Передзахист	08.05-11.05.2026	Виконано
9	Здача в деканат	08.05-14.05.2026	Виконано

Здобувач(ка) вищої освіти _____ Андрій КРАСЮК
(підпис) *(Ім'я, ПРІЗВИЩЕ)*

Керівник кваліфікаційної роботи _____ Ярослав ТОРОШАНКО
(підпис) *(Ім'я, ПРІЗВИЩЕ)*

РЕФЕРАТ

Текстова частина кваліфікаційної роботи на здобуття освітнього ступеня бакалавра: 98 стор., 48 рис., 6 табл., 32 джерел.

Мета роботи – оптимізація та спрощення процесу підключення користувачів до внутрішніх ресурсів за рахунок використання клієнт-серверної моделі та автоматизації управління доступом.

Об'єкт дослідження – процес організації захищеного віддаленого доступу до внутрішніх ресурсів у комп'ютерних мережах.

Предмет дослідження – засоби та технології побудови, налаштування й оптимізації VPN-систем на основі клієнт-серверної архітектури.

Короткий зміст роботи: У роботі розглянуто теоретичні основи організації захищеного віддаленого доступу, принципи функціонування VPN, механізми тунелювання, шифрування та автентифікації, а також архітектуру клієнт-серверних систем. Проаналізовано сучасні VPN-протоколи, зокрема OpenVPN, WireGuard та IPSec, і визначено їх переваги та обмеження з урахуванням вимог до безпеки, продуктивності та гнучкості.

Практична частина включає реалізацію автоматизованої системи підключення користувачів із використанням OpenVPN із розгортанням серверної частини на базі старого апаратного забезпечення — ноутбука Lenovo ThinkPad T480, що обґрунтовано його енергоефективністю, достатньою продуктивністю та автономністю. Виконано налаштування системи, автоматизацію створення клієнтських конфігурацій, впровадження багаторівневої автентифікації, а також проведено тестування продуктивності, стабільності та ефективності роботи в умовах обмежених ресурсів і нестабільних мереж.

КЛЮЧОВІ СЛОВА: VPN, OPENVPN, ВІДДАЛЕНИЙ ДОСТУП, КЛІЄНТ-СЕРВЕРНА АРХІТЕКТУРА, ШИФРУВАННЯ, АВТЕНТИФІКАЦІЯ, АВТОМАТИЗАЦІЯ, МЕРЕЖЕВА БЕЗПЕКА, THINKPAD T480, ОПТИМІЗАЦІЯ.

ЗМІСТ

ВСТУП..... 8
1. ТЕОРЕТИЧНІ ОСНОВИ ОРГАНІЗАЦІЇ ЗАХИЩЕНОГО ДОСТУПУ ДО ВНУТРІШНІХ МЕРЕЖ..... 9
1.1 Особливості доступу до внутрішніх ресурсів організацій 9
1.2 Принципи роботи технологій VPN..... 16
1.3 Архітектура клієнт-серверних систем віддаленого доступу 22
1.4 Методи захисту мережевих з'єднань 26
1.5 Огляд інструментів для організації захищеного доступу. 31
2. ПРОЄКТУВАННЯ ТА РЕАЛІЗАЦІЯ СИСТЕМИ ПІДКЛЮЧЕННЯ КОРИСТУВАЧІВ ДО ВНУТРІШНІХ РЕСУРСІВ..... 37
2.1 Архітектура системи підключення користувачів 37
2.2 Побудова системи віддаленого доступу 47
2.3 Алгоритм функціонування системи 62
2.4 Реалізація системи на основі існуючих інструментів 74
3. АНАЛІЗ ЕФЕКТИВНОСТІ ТА ПЕРСПЕКТИВИ РОЗВИТКУ СИСТЕМИ 89
3.1 Методи оптимізації безпеки та продуктивності системи 89
3.2 Порівняння запропонованого рішення з існуючими підходами 93
3.3 Вартість реалізації та перспективи розвитку системи 98
ВИСНОВКИ..... 105
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ..... 106
Додаток А. Фрагмент масового створення конфігурацій машин для OpenVPN Access Server…...……..108
Додаток Б. Фрагмент масового створення конфігурацій співробітників для OpenVPN Access Server……...109
Додаток В. ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ (Презентація)..... 111

ВСТУП

В умовах розвитку інформаційних технологій і зростання обсягів даних підвищується актуальність захищеного доступу до внутрішніх ресурсів, оскільки розширення мереж і кількості користувачів збільшує ризики несанкціонованого доступу та витоку інформації, при цьому віддалена взаємодія потребує використання захищених каналів передачі даних через публічні мережі із застосуванням механізмів контролю доступу та шифрування.

Основою такого доступу є технології віртуальних приватних мереж, які створюють ізольоване середовище зв’язку завдяки тунелюванню, шифруванню та автентифікації, що забезпечує конфіденційність даних, а аналіз сучасних VPN-рішень дозволяє оцінити їх безпеку, продуктивність і гнучкість для вибору оптимального підходу.

Організація віддаленого доступу базується на клієнт-серверній архітектурі, де взаємодія здійснюється через мережеві протоколи, а ефективність залежить від налаштування серверної частини та механізмів автентифікації, включаючи сертифікати, паролі та додаткові фактори, що формують багаторівневий захист доступу.

Реалізація передбачає побудову VPN на базі OpenVPN, розгортання сервера на фізичному обладнанні, налаштування доступу та автоматизацію клієнтських конфігурацій, впровадження багаторівневої автентифікації та перевірку роботи системи шляхом тестування підключень і аналізу її продуктивності та стабільності.

1. ТЕОРЕТИЧНІ ОСНОВИ ОРГАНІЗАЦІЇ ЗАХИЩЕНОГО ДОСТУПУ ДО ВНУТРІШНІХ МЕРЕЖ

1.1 Особливості доступу до внутрішніх ресурсів організацій

В умовах функціонування організацій практично вся діяльність прямо або опосередковано пов’язана з використанням інформаційних систем, сервісів та цифрової інфраструктури, у межах яких відбувається обробка, зберігання та передача даних, що забезпечують бізнес-процеси, управління, комунікацію та інші аспекти діяльності. Вся організаційна структура в інформаційному плані вибудовується навколо доступу до таких ресурсів, а отже одним із ключових завдань є правильна організація доступу користувачів до цих сервісів з урахуванням безпеки, продуктивності та зручності використання.

Аналіз сучасних моделей доступу до внутрішніх ресурсів підприємств та організацій. Внутрішні ресурси організацій зазвичай охоплюють широкий спектр сервісів, серед яких можуть бути:

- системи управління документами;
- бази даних;
- внутрішні веб-застосунки;
- бухгалтерські системи;
- файлові сховища;
- системи контролю версій програмного забезпечення.

Такі сервіси за своєю природою не призначені для відкритого доступу з мережі Інтернет, оскільки вони містять конфіденційну або критично важливу інформацію, а також часто розгорнуті на інфраструктурі, яка не розрахована на обслуговування великої кількості зовнішніх запитів. Відкриття подібних ресурсів у публічну мережу може призвести як до перевантаження обладнання, так і до суттєвих

ризиків інформаційної безпеки, пов’язаних із можливістю несанкціонованого доступу (див. рис. 1.1) [3].

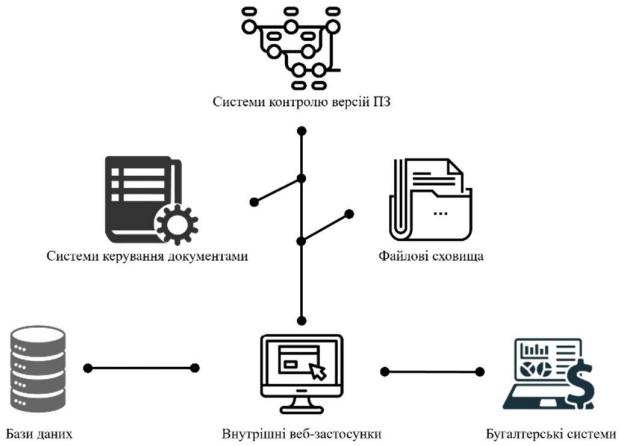


Рисунок 1.1 – Моделі доступу до внутрішніх ресурсів організацій

Традиційно доступ до внутрішніх ресурсів організації реалізовується через локальну мережу, у межах якої всі пристрої з’єднані між собою за допомогою мережевого обладнання, такого як комутатори та маршрутизатори, що забезпечують стабільну взаємодію між вузлами. У такій моделі користувач повинен фізично перебувати в межах корпоративної інфраструктури, наприклад в офісі або іншому робочому просторі, де забезпечується підключення до внутрішньої мережі. Доступ до ресурсів у цьому випадку здійснюється безпосередньо через локальну мережу, а захист забезпечується внутрішніми механізмами автентифікації та контролю доступу, що дозволяє обмежити доступ до інформації відповідно до ролей користувачів.

З розвитком технологій та переходом до гнучких форм організації праці, включаючи дистанційну та змішану роботу, виникла необхідність забезпечення доступу до внутрішніх ресурсів з віддалених місць. Значна частина працівників використовує домашні комп’ютери, ноутбуки або мобільні пристрої для взаємодії з корпоративними системами, що призводить до необхідності впровадження нових моделей доступу, які дозволяють поєднати зручність віддаленої роботи з високим рівнем безпеки.

Однією з таких моделей є надання доступу до внутрішніх сервісів через публічну мережу Інтернет із використанням веб-інтерфейсів, при якому організація відкриває доступ до певних ресурсів через доменне ім’я або публічну IP-адресу, однак такий підхід потребує впровадження додаткових механізмів захисту, оскільки пряме відкриття сервісів у мережу Інтернет суттєво підвищує ризик несанкціонованого доступу та атак з боку зловмисників.

Більш безпечним підходом є використання захищених каналів зв’язку, які дозволяють створити віртуальне підключення до внутрішньої мережі організації через публічну інфраструктуру Інтернету, при цьому передача даних здійснюється у зашифрованому вигляді, що унеможливлює їх перехоплення або модифікацію. У такій моделі користувач, перебуваючи поза межами офісу, отримує можливість працювати з внутрішніми ресурсами так, ніби його пристрій фізично підключений до локальної мережі, а самі ресурси залишаються ізольованими від прямого доступу ззовні, що значно підвищує рівень безпеки [3].

Проблеми безпечного віддаленого доступу користувачів до локальної інфраструктури. Надання віддаленого доступу до внутрішніх ресурсів організації супроводжується рядом технічних та безпекових викликів, які необхідно враховувати при проєктуванні та впровадженні відповідних систем. Основною проблемою є необхідність забезпечення захисту даних, що передаються через публічні мережі, оскільки мережа Інтернет за своєю природою є відкритим середовищем, у якому інформація може бути перехоплена або змінена під час передачі (див. рис. 1.2).

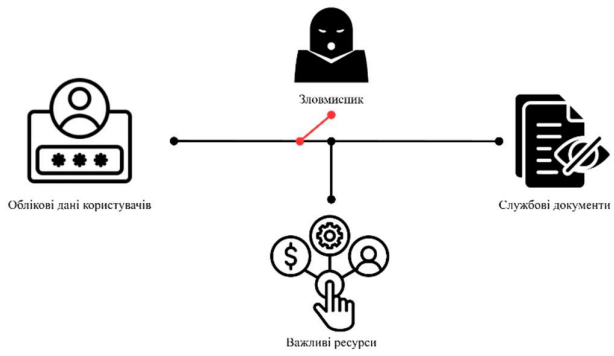


Рисунок 1.2 – Загрози безпеці при віддаленому доступі

Однією з найбільш поширених загроз є перехоплення мережевого трафіку, коли дані передаються через незахищені канали і можуть бути доступними для сторонніх осіб, у такому випадку зловмисники можуть отримати доступ до конфіденційної інформації, включаючи:

— облікові дані користувачів;

— службові документи;

— інші важливі ресурси.

Все це може призвести до витоку інформації та порушення роботи організації.

Ще одним важливим аспектом є забезпечення надійної автентифікації користувачів, оскільки перед наданням доступу до внутрішніх ресурсів необхідно точно встановити особу користувача, при цьому використання слабких або недостатньо захищених механізмів автентифікації значно підвищує ризик компрометації облікових записів і отримання несанкціонованого доступу до мережі [2, 3].

Окрім автентифікації важливим є контроль доступу до ресурсів, оскільки у великих організаціях різні категорії користувачів повинні мати різні рівні доступу до інформації, що вимагає впровадження механізмів розмежування прав доступу відповідно до ролей, посад або функціональних обов’язків. Неправильна реалізація таких механізмів може призвести до ситуацій, коли користувачі отримують доступ до ресурсів, які не входять до їхньої компетенції [2].

Додатковою складністю є різноманітність умов підключення користувачів, оскільки доступ може здійснюватися з домашніх мереж, мобільних пристроїв або через публічні точки доступу, де використовуються динамічні IP-адреси та відсутній контроль над мережевою інфраструктурою, що ускладнює процес ідентифікації користувачів і налаштування безпечного доступу.

У таких умовах виникає потреба у використанні спеціалізованих технологій, які дозволяють створити захищене середовище для передачі даних та забезпечити імітацію присутності користувача у внутрішній мережі незалежно від його фізичного розташування, що дозволяє обійти обмеження традиційних моделей доступу та підвищити рівень безпеки [2, 3].

Організація віддаленого доступу повинна враховувати всі зазначені загрози та обмеження, забезпечуючи баланс між безпекою, зручністю використання та продуктивністю системи, що є необхідною умовою для стабільного функціонування інформаційної інфраструктури в сучасних умовах розвитку цифрових технологій.

Причини використання захищених каналів зв'язку при роботі з внутрішніми сервісами. У інформаційних системах використання захищених каналів зв'язку зумовлене об'єктивною необхідністю забезпечення безпеки передавання даних між користувачами та внутрішніми ресурсами організацій, оскільки передача інформації через відкриті мережі без додаткового захисту робить її вразливою до перехоплення, аналізу або модифікації з боку сторонніх осіб. Передавання даних у незахищеному вигляді створює умови, за яких будь-який учасник мережевої взаємодії, що має доступ до каналів зв'язку, може здійснити перехоплення трафіку, а це у свою чергу призводить до ризику компрометації конфіденційної інформації, включаючи облікові дані користувачів, службову документацію або інші критично важливі ресурси (див. рис. 1.3) [2-4].

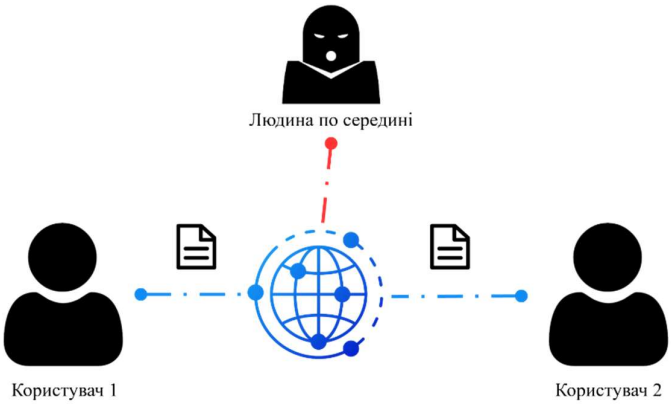


Рисунок 1.3 – Передача даних через публічні мережі без захисту

Використання захищених каналів зв'язку дозволяє усунути зазначені ризики завдяки застосуванню механізмів шифрування, які перетворюють передавані дані у форму, недоступну для розуміння без відповідних криптографічних ключів, у

результаті чого інформація стає доступною виключно для авторизованих сторін, що володіють засобами для її розшифрування. Такий підхід забезпечує конфіденційність передавання даних, оскільки навіть у випадку перехоплення трафіку зловмисник не зможе отримати корисну інформацію без доступу до ключів шифрування [4].

Поряд із конфіденційністю важливим аспектом є забезпечення цілісності інформації, оскільки під час передавання через публічні мережі існує ризик її навмисної або випадкової модифікації, що може призвести до викривлення даних або порушення роботи інформаційних систем. Використання захищених протоколів передавання даних дозволяє впровадити механізми перевірки цілісності, які забезпечують контроль незмінності інформації під час передавання, при цьому будь-які спроби втручання або модифікації даних можуть бути виявлені на стороні отримувача, що дозволяє своєчасно реагувати на потенційні загрози [3-4].

Ще одним важливим фактором є забезпечення автентифікації сторін, що беруть участь у встановленні з’єднання, оскільки перед початком обміну даними необхідно переконатися у справжності як клієнта, так і сервера, що дозволяє запобігти підключенню до підроблених ресурсів або несанкціонованому доступу до внутрішньої інфраструктури. У процесі встановлення захищеного з’єднання обидві сторони виконують взаємну перевірку, і лише у випадку успішного підтвердження автентичності відбувається подальший обмін даними, що значно знижує ризик атак типу «людина посередині» та інших форм втручання.

Окрім безпекових аспектів, використання захищених каналів зв’язку дозволяє оптимізувати організацію доступу до внутрішніх ресурсів, оскільки замість відкриття великої кількості окремих сервісів у мережу Інтернет організація може використовувати єдиний захищений канал, через який здійснюється доступ до всієї внутрішньої інфраструктури. Такий підхід спрощує адміністрування системи, зменшує кількість потенційних точок входу для зловмисників та підвищує загальний рівень керованості мережевої інфраструктури.

Роль VPN у забезпеченні безпечного віддаленого доступу. Розвиток технологій віддаленого доступу та зростання потреби у безпечній взаємодії з внутрішніми ресурсами організацій зумовили широке впровадження VPN-технологій як одного з найбільш ефективних засобів забезпечення захищеного доступу через публічні мережі. Virtual Private Network представляє собою технологію, яка дозволяє створити віртуальну приватну мережу поверх існуючої мережевої інфраструктури, зокрема мережі Інтернет, забезпечуючи безпечний обмін даними між клієнтськими пристроями та серверною частиною системи.

Основним принципом роботи VPN є створення логічного з'єднання між клієнтом і сервером у вигляді захищеного тунелю, через який передається весь мережевий трафік, при цьому дані, що передаються, проходять процес шифрування перед відправленням і розшифровуються після отримання, що забезпечує їх захист від перехоплення або несанкціонованої модифікації. Така модель дозволяє реалізувати ефект присутності користувача у внутрішній мережі організації незалежно від його фактичного місцезнаходження, оскільки після встановлення VPN-з'єднання клієнт отримує доступ до ресурсів так, ніби він підключений безпосередньо до локальної інфраструктури (див. рис. 1.4) [1].

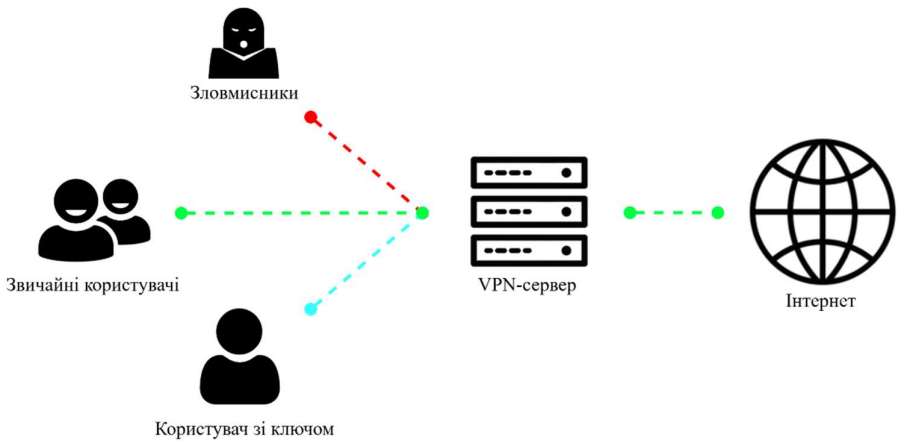


Рисунок 1.4 – Принцип роботи VPN-з'єднання

Застосування VPN-технологій у корпоративних мережах дозволяє уникнути необхідності відкриття внутрішніх сервісів у публічну мережу, що значно знижує

ризик несанкціонованого доступу та підвищує загальний рівень інформаційної безпеки. Внутрішні ресурси залишаються ізольованими від зовнішнього середовища, а доступ до них здійснюється виключно через захищений канал, що забезпечує контрольований і безпечний обмін даними [1, 4].

Одним із найбільш поширених рішень для реалізації VPN-інфраструктури є OpenVPN, який представляє собою програмне забезпечення з відкритим вихідним кодом і підтримує створення захищених тунелів між клієнтами та сервером із використанням сучасних криптографічних алгоритмів. Використання сертифікатів, ключів та протоколів шифрування дозволяє реалізувати надійну систему автентифікації та захисту даних, а гнучкі можливості конфігурації забезпечують адаптацію рішення до різних умов експлуатації.

Завдяки використанню VPN-технологій користувачі отримують можливість підключатися до внутрішніх ресурсів організації з будь-якого місця, де є доступ до мережі Інтернет, при цьому всі передавані дані залишаються захищеними, а доступ до ресурсів обмежується лише авторизованими користувачами. Такий підхід дозволяє поєднати зручність дистанційної роботи з високим рівнем безпеки, що є необхідною умовою функціонування сучасних інформаційних систем і робить VPN одним із базових елементів мережевої інфраструктури організацій [1].

1.2 Принципи роботи технологій VPN

У комп’ютерних мережах питання безпечного обміну даними між користувачами та інформаційними системами набуває особливої актуальності, оскільки значна частина взаємодії відбувається через публічні мережі, зокрема мережу Інтернет, яка за своєю природою є відкритим середовищем і не гарантує захисту переданої інформації. У таких умовах виникає необхідність використання технологій, що дозволяють забезпечити:

— конфіденційність;

— цілісність;

— автентичність даних.

Одним із найбільш поширених підходів до вирішення цієї задачі є застосування технології Virtual Private Network, яка забезпечує створення захищеного з'єднання між віддаленими пристроями поверх існуючої мережевої інфраструктури.

Загальна концепція Virtual Private Network та принципи тунелювання трафіку. VPN представляє собою технологію, яка дозволяє формувати логічно ізольовану мережу поверх публічної мережі, при цьому взаємодія між вузлами організовується таким чином, ніби вони знаходяться в межах єдиної локальної мережі, незважаючи на те, що фізично можуть бути розташовані в різних географічних точках і підключені через Інтернет. Основна ідея полягає у створенні віртуального середовища, у якому передача даних здійснюється через захищений канал, що дозволяє приховати інформацію від сторонніх учасників мережевої взаємодії [5].

Ключовим механізмом функціонування VPN є тунелювання, яке забезпечує передавання даних одного протоколу всередині іншого, тобто мережеві пакети внутрішньої мережі інкапсулюються у пакети транспортного або мережевого протоколу, що використовується для передачі через публічну інфраструктуру. У процесі передачі такі пакети проходять через мережу Інтернет, але залишаються захищеними від зовнішнього впливу, оскільки додатково застосовуються механізми шифрування, що перетворюють дані у недоступний для сторонніх вигляд. Після досягнення кінцевої точки тунелю відбувається зворотна операція, у ході якої дані розпаковуються та передаються до внутрішньої мережі призначення (див. рис. 1.5).

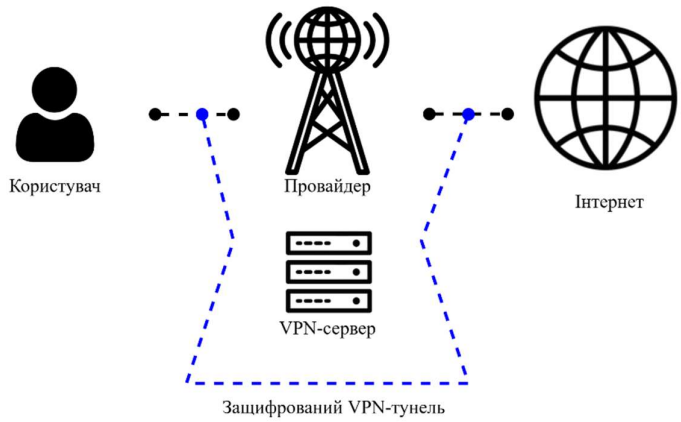


Рисунок 1.5 – Схема тунелювання мережевого трафіку

Уявлення про тунель як про віртуальний канал між двома вузлами дозволяє зрозуміти принцип роботи VPN, оскільки передача інформації здійснюється через публічну мережу, але при цьому вона залишається ізольованою від інших учасників мережі, що забезпечує необхідний рівень захисту. Використання шифрування в межах тунелю дозволяє гарантувати конфіденційність даних, а додаткові механізми перевірки забезпечують їх цілісність і автентичність [5].

VPN-з’єднання зазвичай встановлюється між клієнтським пристроєм і сервером, де клієнтом може виступати персональний комп’ютер, ноутбук або мобільний пристрій, а сервер розміщується у внутрішній мережевій інфраструктурі організації. Після встановлення з’єднання весь мережевий трафік між клієнтом і сервером передається через захищений тунель, що дозволяє контролювати доступ до ресурсів і забезпечувати захист інформації незалежно від фізичного розташування користувача.

У межах VPN можуть використовуватися різні типи з’єднань, серед яких:

— клієнт-сервер;

— мережа-мережа.

Найбільш поширеним є варіант клієнт-сервер, що забезпечує доступ окремих користувачів до корпоративної мережі, а також варіант мережа-мережа, який використовується для об’єднання декількох локальних мереж через Інтернет. Такий підхід дозволяє організувати безпечну взаємодію між віддаленими

підрозділами або офісами, використовуючи існуючу мережеву інфраструктуру без необхідності створення окремих фізичних каналів зв’язку [5].

Основні протоколи VPN та їх характеристики. Реалізація VPN-з’єднань здійснюється за допомогою спеціалізованих мережевих протоколів, які визначають:

— спосіб створення тунелю;

— методи шифрування;

— механізми автентифікації.

Існує кілька основних протоколів, кожен з яких має свої особливості, переваги та обмеження, що визначають сферу його застосування [6].

Одним із перших широко використовуваних протоколів є PPTP, який був розроблений для створення тунелів на основі протоколу PPP і відзначався простотою реалізації та високою швидкістю роботи, однак з часом було виявлено суттєві вразливості у механізмах шифрування, що призвело до зниження рівня довіри до цього протоколу, у результаті чого його використання у сучасних системах значно обмежене [6].

Іншим поширеним рішенням є L2TP, який забезпечує тунелювання на канальному рівні та часто використовується у поєднанні з IPsec, де L2TP відповідає за створення тунелю, а IPsec забезпечує шифрування та автентифікацію даних, такий підхід дозволяє досягти високого рівня безпеки, але водночас призводить до збільшення накладних витрат і навантаження на мережеві ресурси [6].

Протокол IPsec є одним із найбільш поширених рішень для забезпечення безпеки на мережевому рівні, оскільки він дозволяє захищати IP-пакети шляхом їх шифрування та перевірки цілісності, підтримується більшістю мережевого обладнання та широко використовується для організації як клієнт-серверних, так і міжмережевих VPN-з’єднань, однак складність налаштування може ускладнювати його використання у випадках, коли необхідно швидко розгорнути систему для великої кількості користувачів [6].

Окрім класичних протоколів існують програмні VPN-рішення, які використовують власні механізми тунелювання та працюють поверх стандартних

транспортних протоколів, таких як TCP або UDP, що дозволяє забезпечити гнучкість у налаштуванні та адаптацію до різних мережевих умов. Такі рішення можуть ефективно працювати навіть у середовищах із обмеженнями на рівні мережевої інфраструктури, наприклад при використанні NAT або фільтрації трафіку [5-6].

Механізми шифрування та автентифікації у VPN-мережах. Серед основних характеристик VPN-технологій, які визначають їхню ефективність у контексті захисту інформації, особливе значення має забезпечення безпеки даних, що передаються між вузлами мережі, оскільки передача інформації через публічні мережі супроводжується ризиками перехоплення, модифікації або несанкціонованого доступу. Для мінімізації таких ризиків застосовуються криптографічні механізми, що дозволяють гарантувати конфіденційність, цілісність та автентичність переданої інформації незалежно від характеру даних або середовища їх передачі [7].

Шифрування передбачає перетворення відкритого тексту у зашифрований вигляд, який неможливо інтерпретувати без відповідного криптографічного ключа, що забезпечує захист мережевого трафіку під час його передачі через публічні мережі та унеможливлює доступ до змісту даних сторонніми особами. У VPN-системах широко застосовуються алгоритми симетричного шифрування, серед яких поширеними є AES та Blowfish, що забезпечують високий рівень захисту при відносно невеликому обчислювальному навантаженні, при цьому симетричний підхід передбачає використання одного і того ж ключа як для шифрування, так і для розшифрування інформації (див. рис. 1.6) [6-7].

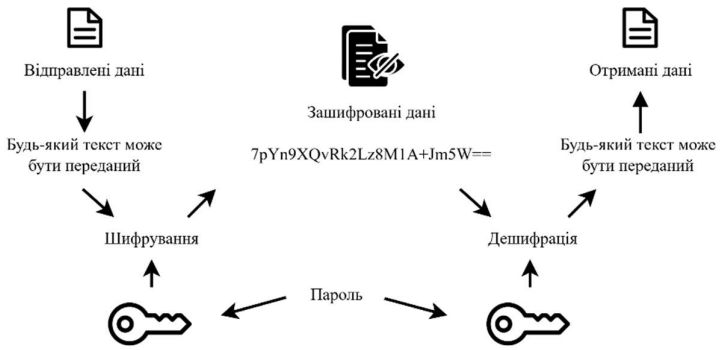


Рисунок 1.6 – Механізм шифрування даних у VPN

Окрім шифрування важливим елементом безпеки виступає автентифікація, яка дозволяє перевірити, чи має користувач або пристрій право доступу до мережі, а також підтвердити справжність сторін, що встановлюють з'єднання. У VPN-системах застосовуються різні методи автентифікації, включаючи використання паролів, криптографічних ключів та цифрових сертифікатів, при цьому особливо поширеним є використання інфраструктури відкритих ключів, яка забезпечує високий рівень довіри між учасниками взаємодії. У такій моделі кожен користувач отримує власний сертифікат, який використовується для підтвердження його ідентичності під час встановлення VPN-з'єднання, після чого сервер виконує перевірку сертифіката та приймає рішення щодо надання або відмови у доступі (див. рис. 1.7) [7].

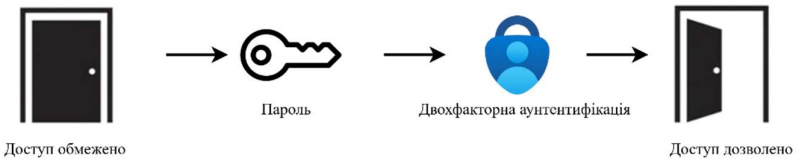


Рисунок 1.7 – Процес автентифікації користувача у VPN

Додатково застосовуються механізми контролю цілісності даних, які дозволяють визначити, чи були змінені пакети під час передачі, для чого використовуються криптографічні хеш-функції, що формують контрольні

значення для кожного блоку даних, при цьому будь-яка зміна інформації призводить до зміни контрольної суми, що дозволяє виявити факт втручання.

1.3 Архітектура клієнт-серверних систем віддаленого доступу

У комп’ютерних мережах значна частина інформаційних систем будується на основі клієнт-серверної архітектури, оскільки саме такий підхід дозволяє організувати ефективну взаємодію між користувачами та ресурсами мережі, забезпечуючи:

- централізоване управління;
- контроль доступу;
- раціональне використання обчислювальних ресурсів.

Подібна модель широко застосовується в організаціях і компаніях, де існує потреба у наданні доступу до внутрішньої інфраструктури як для локальних, так і для віддалених користувачів, зокрема в умовах дистанційної або гібридної роботи.

Поняття клієнт-серверної архітектури в комп’ютерних мережах. Клієнт-серверна модель передбачає розподіл функцій між двома основними компонентами системи, де клієнт виступає ініціатором запитів, а сервер забезпечує їх обробку та надання відповідних ресурсів або сервісів. У ролі клієнта може виступати:

- персональний комп’ютер;
- ноутбук;
- мобільний пристрій;
- програмне забезпечення.

У випадку програмного забезпечення – воно функціонує на цих пристроях і формує запити до серверної частини системи, основним призначенням клієнта є надсилання запитів і отримання результатів їх обробки. Сервер, у свою чергу, виступає центральним елементом інфраструктури, який виконує обробку запитів, забезпечує доступ до даних, керує ресурсами та реалізує необхідні мережеві

служби, включаючи роботу з базами даних, файловими сховищами або системами управління користувачами (див. рис. 1.8) [9].

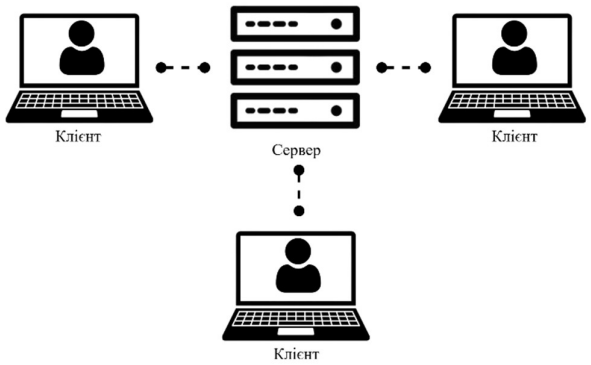


Рисунок 1.8 – Клієнт-серверна архітектура мережі

Однією з важливих характеристик клієнт-серверної архітектури є централізація управління, за якої основні ресурси та механізми контролю зосереджені на сервері, що дозволяє адміністраторам:

1. Здійснювати моніторинг роботи системи.
2. Впроваджувати політики безпеки.
3. Виконувати резервне копіювання даних.
4. Регулювати рівень доступу користувачів до різних ресурсів.

У такий підхід спрощується обслуговування системи та підвищується її надійність, оскільки всі критичні процеси контролюються з єдиного центру [9].

Ще однією суттєвою особливістю є масштабованість, яка дозволяє адаптувати систему до зростання кількості користувачів або збільшення навантаження шляхом розширення серверної інфраструктури або підвищення її продуктивності, що забезпечує гнучкість у розвитку інформаційної системи відповідно до потреб організації. Завдяки цьому клієнт-серверна модель застосовується у різних типах систем, включаючи веб-сервіси, системи управління базами даних, файлові сервери та рішення для віддаленого доступу, де необхідно забезпечити стабільну та безпечну взаємодію між користувачами та внутрішніми ресурсами.

У контексті віддаленого доступу така архітектура дозволяє організувати підключення користувачів до корпоративної мережі через спеціалізований сервер, який виконує функції:

— контролю доступу;

— автентифікації;

— захисту переданої інформації.

При цьому користувач отримує можливість працювати з внутрішніми ресурсами так, ніби він знаходиться безпосередньо в межах локальної мережі організації.

Організація взаємодії між клієнтом та сервером у системах віддаленого доступу. У системах віддаленого доступу взаємодія між клієнтом і сервером є основою функціонування всієї інфраструктури, оскільки саме вона забезпечує:

— передачу даних;

— доступ до ресурсів;

— виконання необхідних операцій.

Така взаємодія базується на принципі обміну запитами та відповідями, де клієнт ініціює встановлення з’єднання і надсилає запити, а сервер приймає їх, обробляє та повертає результати виконання [9-10].

Процес встановлення з’єднання починається з ініціалізації підключення клієнтським пристроєм, при цьому використовується спеціалізоване програмне забезпечення, яке формує запит на підключення до сервера, що містить необхідну інформацію, зокрема мережеву адресу сервера, параметри з’єднання та облікові дані користувача. Після отримання такого запиту сервер виконує перевірку автентичності, яка може включати перевірку логіна і пароля, використання цифрових сертифікатів або інших методів ідентифікації, передбачених політикою безпеки системи, у разі успішного проходження перевірки сервер дозволяє встановлення з’єднання та надає доступ до внутрішніх ресурсів [10].

Після встановлення з’єднання формується захищений канал передачі даних, у межах якого використовується шифрування для забезпечення конфіденційності інформації, що передається між клієнтом і сервером, у результаті всі дані проходять

через цей канал у захищеному вигляді, що знижує ризик їх перехоплення або модифікації.

У процесі роботи клієнт надсилає запити до сервера, які можуть стосуватися доступу до файлів, баз даних або інших ресурсів, сервер обробляє ці запити відповідно до наданих прав доступу та повертає результати, забезпечуючи таким чином повноцінну взаємодію користувача з інформаційною системою. Додатково сервер виконує функції контролю доступу, визначаючи перелік ресурсів, доступних для конкретного користувача, що дозволяє обмежувати доступ до конфіденційної інформації та забезпечувати дотримання встановлених політик безпеки.

Роль серверної інфраструктури у забезпеченні підключення користувачів. Серверна інфраструктура виступає базовою складовою системи віддаленого доступу, через яку реалізується:

— прийом підключень від клієнтів;

— обробка запитів;

— надання доступу до внутрішніх ресурсів організації.

Все це вже може бути розміщено у межах корпоративної мережі. Саме через сервери здійснюється взаємодія між користувачами та інформаційними системами, що дозволяє організувати централізовану модель управління доступом і контролю за використанням ресурсів [11].

Одним із основних завдань серверної інфраструктури є організація процесу автентифікації користувачів, під час якого:

1. Перевіряються облікові дані клієнтів.
2. Визначається наявність прав доступу до системи.

У випадку успішної перевірки стає можливим встановлення з’єднання з внутрішньою мережею, тоді як у разі невідповідності даних доступ блокується ще на етапі ініціалізації підключення, що дозволяє запобігти несанкціонованому доступу [11].

Подальшим важливим аспектом функціонування серверної інфраструктури є управління мережевими підключеннями, яке включає:

— відстеження активності користувачів;

— розподіл мережевих ресурсів;

— забезпечення стабільної роботи системи.

Все це навіть за умов значного навантаження, коли одночасно підключена велика кількість клієнтів. У межах цих процесів сервери виконують функції маршрутизації трафіку, спрямовуючи запити користувачів до відповідних ресурсів, таких як:

— сервери баз даних;

— файлові сховища;

— інші інформаційні системи.

Все те, що розташовано безпосередньо у внутрішній мережі.

Реалізація механізмів безпеки також покладається на серверну інфраструктуру, оскільки вона забезпечує:

— шифрування мережевого трафіку;

— контроль доступу до ресурсів;

— ведення журналів підключень;

— моніторинг активності користувачів.

Все що дозволяє виявляти підозрілі дії та оперативно реагувати на потенційні загрози. Такі механізми сприяють підтримці актуального рівня захисту інформації та дозволяють своєчасно запобігати інцидентам безпеки.

1.4 Методи захисту мережевих з’єднань

В умовах активного використання віддалених форм роботи питання захисту мережевих з’єднань набуває особливої актуальності, оскільки підключення до корпоративної інфраструктури через мережу Інтернет супроводжується низкою потенційних загроз, здатних призвести до:

— витоку конфіденційної інформації;

— несанкціонованого доступу до внутрішніх ресурсів;

— порушення стабільності функціонування інформаційних систем.

Передача даних через відкриті мережі передбачає проходження інформації через численні проміжні вузли, включаючи маршрутизатори та мережеві шлюзи, у разі відсутності належного захисту зловмисник може перехопити такі дані за допомогою спеціалізованих програмних засобів і отримати доступ до облікових даних, службових документів або іншої критично важливої інформації.

Основні загрози безпеці під час віддаленого доступу до мереж. Суттєву небезпеку становлять атаки типу «людина посередині», під час яких зловмисник перехоплює з'єднання між клієнтом і сервером та виступає у ролі проміжної ланки, отримуючи можливість не лише переглядати передані дані, але й змінювати їхній зміст, що може призвести до компрометації всієї системи та викривлення результатів обміну інформацією. Додатковим ризиком виступає недостатньо надійна система автентифікації, коли використання простих механізмів перевірки, наприклад лише логіна і пароля, створює умови для несанкціонованого доступу до внутрішньої мережі, після чого зловмисник може використовувати отримані привілеї для подальших атак або доступу до конфіденційних ресурсів (див. рис. 1.10) [13].



Рисунок 1.9 – Основні кіберзагрози у мережевих з'єднаннях

Окрему категорію загроз становить шкідливе програмне забезпечення, яке може бути присутнім на клієнтських пристроях і використовувати встановлене з'єднання для проникнення у внутрішню мережу організації, у результаті чого відбувається поширення загрози на інші системи та сервіси. Поряд із цим існують атаки, спрямовані на перевантаження мережевої інфраструктури, зокрема DDoS-

атаки, які можуть призвести до тимчасової недоступності сервісів або значного зниження їх продуктивності, що ускладнює доступ користувачів до необхідних ресурсів [13].

З урахуванням зазначених ризиків організація віддаленого доступу потребує комплексного підходу до захисту, що включає застосування криптографічних механізмів, впровадження надійної автентифікації, контроль доступу та постійний моніторинг мережевої активності, при цьому ефективність захисту значною мірою залежить від коректності налаштування та узгодженості всіх компонентів системи, оскільки навіть незначні помилки можуть створювати додаткові вектори для атак.

Методи шифрування та забезпечення конфіденційності переданих даних. Протидія основним загрозам мережевої безпеки передбачає використання криптографічних методів, які дозволяють забезпечити конфіденційність переданої інформації та унеможливити її використання сторонніми особами. Шифрування забезпечує перетворення даних у форму, що є недоступною для розуміння без наявності відповідного ключа, у результаті навіть у випадку перехоплення трафіку отримана інформація не може бути використана без процедури дешифрування [14].

У мережевих системах застосовуються два основні підходи до шифрування, серед яких:

— симетричне шифрування;

— асиметричне шифрування.

Симетричне шифрування передбачає використання одного ключа для шифрування та розшифрування даних, що забезпечує високу швидкість обробки та ефективність при передачі великих обсягів інформації, тому такий підхід широко використовується для захисту мережевого трафіку. До найбільш поширених алгоритмів належить AES, який забезпечує високий рівень криптографічної стійкості та активно застосовується у сучасних системах інформаційної безпеки, поряд із ним можуть використовуватися й інші алгоритми, зокрема Blowfish або ChaCha20, які також забезпечують надійний захист даних залежно від умов використання [14].

Асиметричне шифрування ґрунтується на використанні пари ключів, відкритого та закритого, при цьому відкритий ключ може бути доступним іншим учасникам взаємодії, тоді як закритий ключ зберігається у таємниці, дані, зашифровані за допомогою відкритого ключа, можуть бути розшифровані лише відповідним закритим ключем, що забезпечує високий рівень безпеки під час обміну інформацією та широко використовується для передачі ключів і організації захищених з'єднань між клієнтом і сервером [14].

Окрім забезпечення конфіденційності важливим аспектом є контроль цілісності даних, для чого застосовуються криптографічні хеш-функції, які формують унікальні контрольні значення для кожного повідомлення, у разі зміни переданих даних змінюється і хеш, що дозволяє виявити факт втручання. Поєднання симетричного та асиметричного шифрування разом із механізмами перевірки цілісності формує комплексний підхід до захисту інформації, який дозволяє створити надійний механізм безпеки в системах віддаленого доступу.

Застосування таких методів забезпечує належний рівень конфіденційності даних, що передаються між клієнтом і сервером, та ускладнює можливість їх перехоплення або використання зловмисниками, що є необхідною умовою для побудови безпечної мережевої інфраструктури в умовах використання публічних каналів зв'язку.

Механізми автентифікації та управління доступом. Забезпечення безпеки мережевих з'єднань не обмежується лише застосуванням шифрування, оскільки навіть при наявності надійних криптографічних алгоритмів існує ризик компрометації ключів або інших елементів захисту, у зв'язку з чим виникає необхідність впровадження механізмів автентифікації, що дозволяють перевірити особу користувача або пристрою, який намагається отримати доступ до системи, і тим самим забезпечити контроль доступу до внутрішніх ресурсів. Відсутність надійної автентифікації унеможливлює ефективне розмежування прав доступу навіть за умови використання шифрування, оскільки система не здатна визначити, чи має конкретний суб'єкт право на підключення до мережі (див. табл. 1.1).

Таблиця 1.1 – Методи автентифікації та їх особливості

Метод	Рівень безпеки	Зручність використання	Сфера застосування
Логін–пароль	Низький–середній	Висока	Базові системи, веб-сервіси
Одноразові паролі (OTP)	Середній–високий	Середня	Банківські системи, корпоративний доступ
Сертифікати (PKI)	Високий	Середня	VPN, корпоративні мережі
Двофакторна автентифікація (2FA)	Високий	Середня	Захищені інформаційні системи
Біометрична автентифікація	Високий	Висока	Мобільні пристрої, системи контролю доступу

Найпростішим і найбільш поширеним методом автентифікації залишається використання пари логін–пароль, яка виступає базовим рівнем захисту та реалізується у вигляді облікових даних, що зберігаються у системі у захищеному вигляді, зазвичай із застосуванням хешування, після чого під час введення користувачем відповідної інформації система здійснює перевірку її відповідності записам у базі даних. Водночас такий підхід має суттєві недоліки, пов’язані з можливістю підбору або викрадення паролів, використанням слабких комбінацій або повторним застосуванням однакових облікових даних, що значно знижує загальний рівень безпеки [15].

Підвищення надійності автентифікації досягається шляхом впровадження багатофакторних механізмів, у межах яких для підтвердження особи використовується декілька незалежних факторів, наприклад поєднання пароля з одноразовим кодом або криптографічним ключем, у результаті навіть у разі компрометації одного з факторів доступ до системи залишається захищеним, оскільки для входу необхідно підтвердити ще один рівень ідентифікації. Такий підхід значно зменшує ризик несанкціонованого доступу та підвищує стійкість системи до атак [13, 15].

Після успішного проходження автентифікації виникає потреба у визначенні рівня доступу користувача до ресурсів системи, що реалізується через механізми управління доступом, у межах яких кожному користувачу або групі користувачів

призначаються певні ролі та права, що визначають можливості взаємодії з інформаційними ресурсами. У результаті одні користувачі можуть отримувати доступ лише до обмеженого набору сервісів, тоді як адміністратори мають розширені привілеї, включаючи налаштування системи, управління обліковими записами та контроль за функціонуванням мережевої інфраструктури [15].

Додатковим елементом контролю виступає ведення журналів подій, які фіксують спроби входу до системи, активність користувачів і можливі помилки автентифікації, що дозволяє аналізувати поведінку у мережі та своєчасно виявляти підозрілі дії, пов’язані з потенційними загрозами безпеці.

1.5 Огляд інструментів для організації захищеного доступу

У комп’ютерних мережах, особливо в інфраструктурах організацій та компаній, організація захищеного віддаленого доступу до внутрішніх ресурсів стає необхідною умовою функціонування, оскільки використання публічних мереж для передачі службової інформації вимагає надійних механізмів її захисту. Реалізація такого доступу здійснюється за допомогою різноманітних VPN-рішень, які дозволяють створити захищену інфраструктуру обміну даними між клієнтами та корпоративною мережею, забезпечуючи конфіденційність, цілісність і контроль доступу до інформаційних ресурсів. Передача даних у цьому випадку відбувається через захищені тунелі, що дозволяє уникнути несанкціонованого доступу навіть при використанні відкритих каналів зв’язку.

Аналіз сучасних програмних засобів для створення VPN-інфраструктури. Створення VPN-інфраструктури може здійснюватися як на основі спеціалізованого мережевого обладнання, так і за допомогою програмних рішень, які встановлюються на сервери та клієнтські пристрої, при цьому програмні засоби залишаються більш доступними та гнучкими у використанні, оскільки не потребують значних фінансових витрат і можуть бути розгорнуті навіть у

середовищі віртуалізації або на звичайному серверному обладнанні. Такий підхід дозволяє швидко впровадити систему захищеного доступу навіть у невеликих організаціях або на початкових етапах розвитку інфраструктури.

Серед програмних рішень значне поширення отримали системи з відкритим вихідним кодом, які дозволяють створювати власну VPN-інфраструктуру без витрат на ліцензування, при цьому вони активно підтримуються спільнотами розробників, регулярно оновлюються та забезпечують широкий набір функціональних можливостей для налаштування мережевих з'єднань відповідно до конкретних потреб. Гнучкість конфігурації таких рішень дозволяє адаптувати їх до різних сценаріїв використання, включаючи як віддалений доступ користувачів, так і об'єднання декількох локальних мереж у єдину структуру.

Поряд із відкритими рішеннями існують комерційні VPN-системи, які розробляються виробниками мережевого обладнання та програмного забезпечення і часто інтегруються з корпоративними системами управління, забезпечуючи зручні інтерфейси адміністрування, розширені можливості моніторингу та централізований контроль доступу. Використання таких рішень передбачає додаткові фінансові витрати, пов'язані з ліцензуванням і підтримкою, що може бути обмежувальним фактором для деяких організацій, особливо на початкових етапах впровадження інфраструктури.

Сучасні VPN-рішення, незалежно від їх типу, підтримують широкий спектр функцій, включаючи використання криптографічних алгоритмів, механізмів автентифікації та управління доступом, а також можливість роботи у різних мережевих середовищах, при цьому важливою вимогою залишається підтримка мультиплатформеності, що дозволяє використовувати такі системи на різних операційних системах, включаючи серверні середовища на базі Linux та клієнтські пристрої під керуванням Windows або macOS. У підсумку вибір конкретного інструменту для організації захищеного доступу залежить від потреб організації, масштабу інфраструктури, доступних ресурсів і вимог до рівня безпеки.

Порівняння можливостей популярних VPN-рішень. Серед сучасних технологій, що застосовуються для організації захищеного мережевого доступу,

існує значна кількість рішень, кожне з яких має власні особливості, переваги та обмеження, що визначають доцільність їх використання у конкретних умовах (див. табл. 1.2).

Таблиця 1.2 – Характеристики VPN-протоколів (OpenVPN, WireGuard, IPsec)

Протокол	Тип шифрування	Продуктивність	Складність налаштування	Підтримка MFA
OpenVPN	TLS (AES-256)	Середня	Середня	Так
WireGuard	ChaCha20	Висока	Низька	Обмежена
IPsec	AES, 3DES	Висока	Висока	Так

Одним із найбільш поширених рішень є OpenVPN, яке належить до програмного забезпечення з відкритим вихідним кодом і широко використовується для створення захищених тунелів у корпоративних мережах, забезпечуючи високий рівень безпеки завдяки підтримці сучасних криптографічних алгоритмів і можливості гнучкого налаштування [8, 12, 16-17].

Іншою сучасною технологією є WireGuard, який відрізняється:

— спрощеною архітектурою;

— високою продуктивністю;

— використанням сучасних криптографічних механізмів.

Все це дозволяє досягати високої швидкості передачі даних і зменшувати затримки, проте обмежена гнучкість налаштування у деяких випадках може ускладнювати його використання у складних корпоративних середовищах, де необхідна детальна конфігурація політик доступу та маршрутизації [17].

Ще одним поширеним підходом є використання IPsec, який функціонує на рівні мережевого протоколу та підтримується великою кількістю мережевих пристроїв, включаючи маршрутизатори та міжмережеві екрани, що робить його зручним для інтеграції у вже існуючу інфраструктуру, водночас складність конфігурації та адміністрування може створювати додаткові труднощі, особливо у великих і розгалужених мережах [17].

Комерційні VPN-рішення пропонують розширені можливості управління, включаючи:

— централізоване адміністрування;

— інтеграцію з корпоративними системами автентифікації;

— моніторинг мережевої активності;

— гнучке налаштування політик безпеки.

Все це дозволяє підвищити ефективність управління інфраструктурою, однак їх використання пов’язане з додатковими витратами на ліцензування та технічну підтримку.

Порівняння зазначених рішень за такими критеріями, як рівень безпеки, продуктивність, складність налаштування, можливості масштабування та сумісність з різними операційними системами, дозволяє визначити найбільш доцільний варіант для конкретної організації, оскільки кожна технологія має власну сферу застосування та ефективність залежно від умов використання, структури мережі та вимог до захисту інформації.

Переваги використання OpenVPN у корпоративних мережах. Серед великої кількості сучасних VPN-рішень вибір OpenVPN обумовлюється поєднанням:

— високого рівня безпеки;

— гнучкості налаштування;

— використання у різних мережевих середовищах.

Все це робить його одним із найбільш поширених інструментів для організації захищеного доступу до корпоративних ресурсів. Популярність цього рішення пояснюється здатністю забезпечувати надійний захист переданих даних завдяки використанню сучасних криптографічних алгоритмів, серед яких широко застосовується AES, а також інших механізмів шифрування, що гарантують конфіденційність інформації під час її передачі між клієнтом і сервером. Додатковим фактором є використання бібліотеки OpenSSL, яка відповідає сучасним стандартам криптографічного захисту і забезпечує реалізацію перевірених механізмів безпеки [18].

Гнучкість конфігурації дозволяє адаптувати систему до конкретних вимог організації, оскільки доступна можливість налаштування параметрів шифрування, методів автентифікації та політик доступу, що забезпечує високий рівень контролю

над мережею та дозволяє реалізовувати різні сценарії використання залежно від структури інфраструктури. Такий підхід дає змогу створювати як прості конфігурації для невеликих систем, так і складні багаторівневі рішення для великих корпоративних мереж із розгалуженою структурою.

Важливою перевагою виступає мультиплатформеність, яка дозволяє використовувати OpenVPN у різних операційних середовищах, включаючи:

1. Серверні системи на базі Linux.
2. Клієнтські пристрої під керуванням Windows, macOS, Android і iOS.

Все це забезпечує універсальність застосування та спрощує інтеграцію у вже існуючу інфраструктуру. Така сумісність дозволяє організувати доступ до корпоративної мережі з різних типів пристроїв без необхідності суттєвої зміни архітектури системи [18].

Додатково варто враховувати можливість використання OpenVPN у різних умовах мережевого середовища, включаючи підключення через домашні мережі або мобільні канали зв'язку, що особливо актуально для віддалених працівників, які потребують стабільного і захищеного доступу до внутрішніх ресурсів незалежно від свого місцезнаходження. Така універсальність дозволяє забезпечити безперервність роботи та збереження продуктивності навіть за умов змінних мережевих параметрів [18].

Механізми централізованого управління доступом дозволяють контролювати підключення користувачів, створювати індивідуальні конфігурації, обмежувати доступ до окремих ресурсів і здійснювати моніторинг активності, що забезпечує додатковий рівень безпеки та дозволяє оперативно реагувати на потенційні загрози. Сукупність зазначених характеристик формує ефективну систему захищеного доступу, яка відповідає сучасним вимогам інформаційної безпеки та забезпечує стабільну роботу корпоративної мережі.

Обґрунтування вибору OpenVPN як основи для побудови системи. Вибір технології для організації захищеного доступу до внутрішніх ресурсів визначається низкою факторів, серед яких особливе значення мають:

— рівень безпеки;

— гнучкість налаштування;

— сумісність із різними платформами;

— економічна доцільність використання.

У цьому контексті OpenVPN розглядається як одне з найбільш оптимальних рішень, оскільки забезпечує високий рівень захисту завдяки використанню сучасних криптографічних алгоритмів і механізмів автентифікації, включаючи застосування цифрових сертифікатів, що дозволяє реалізувати надійну перевірку користувачів під час встановлення з'єднання [19].

Додатковою перевагою є відкритий вихідний код, який дозволяє використовувати систему без необхідності придбання ліцензій, що є важливим фактором для організацій із обмеженими ресурсами, а також забезпечує можливість незалежного аудиту безпеки та вдосконалення функціональності завдяки активній спільноті розробників. Такий підхід сприяє підвищенню надійності програмного забезпечення та дозволяє швидко реагувати на виявлені вразливості.

Гнучкість конфігурації забезпечує можливість реалізації різних сценаріїв використання, включаючи як організацію віддаленого доступу для окремих користувачів, так і об'єднання кількох мереж у єдину інфраструктуру, що робить систему універсальним інструментом для побудови захищених мережевих з'єднань. Підтримка інтеграції з існуючими мережевими системами дозволяє впроваджувати рішення без необхідності суттєвої модернізації інфраструктури, що спрощує процес розгортання та знижує витрати.

Сумісність із різними операційними системами та платформами дозволяє використовувати OpenVPN у гетерогенних середовищах, де одночасно застосовуються різні типи пристроїв і програмного забезпечення, що особливо важливо для сучасних організацій із розподіленою структурою [19].

2. ПРОЄКТУВАННЯ ТА РЕАЛІЗАЦІЯ СИСТЕМИ ПІДКЛЮЧЕННЯ КОРИСТУВАЧІВ ДО ВНУТРІШНІХ РЕСУРСІВ

2.1 Архітектура системи підключення користувачів

Формування цілісного уявлення про структуру, логіку побудови та принципи функціонування системи віддаленого доступу до внутрішніх ресурсів є необхідною умовою для коректного проєктування, особливо у випадках, коли система включає елементи автоматизації, оскільки зростання рівня автоматизації безпосередньо пов’язане зі зменшенням кількості помилок, підвищенням керованості та спрощенням адміністрування, а також забезпечує передбачуваність поведінки системи у різних сценаріях використання, у зв’язку з чим архітектура визначає не лише взаємозв’язки між окремими компонентами, але й впливає на рівень безпеки, продуктивності та масштабованості рішення в цілому, тому обраний підхід базується на класичній клієнт-серверній моделі із використанням VPN-технологій, що дозволяє реалізувати контрольований, захищений та централізований доступ користувачів до внутрішньої мережі з можливістю подальшої автоматизації процесів управління.

Концепція побудови системи на базі моделі Hub-and-Spoke. В основу архітектури покладено топологію типу Hub-and-Spoke, яка у своїй суті відповідає моделі «зірка», де всі з’єднання концентруються навколо центрального вузла, яким виступає VPN-шлюз, і саме через нього відбувається обробка усіх мережевих взаємодій, включаючи:

— автентифікацію користувачів;
— встановлення захищених тунелів;
— маршрутизацію трафіку;
— контроль доступу до внутрішніх ресурсів.

Тоді як кінцеві клієнти виступають у ролі периферійних вузлів, які підключаються до центрального сервера через зашифровані канали зв’язку, причому характерною особливістю даної топології є відсутність прямої взаємодії між клієнтами, оскільки весь обмін даними здійснюється виключно через центральний вузол, що дозволяє:

1. Централізувати політики доступу.
2. Спростити адміністрування.
3. Підвищити рівень безпеки.

Оскільки будь-які дії користувачів можуть бути проконтрольовані в одній точці, а також така модель добре підходить для невеликих і середніх інфраструктур, де необхідно забезпечити доступ до обмеженого набору ресурсів із чітко визначеними правилами, що у візуальному представленні може бути описано як центральний сервер, з’єднаний із кількома клієнтами через VPN-тунелі, які формуються у процесі автентифікації користувачів після отримання ними відповідних конфігураційних файлів формату .ovpn, що містять:

— параметри підключення;
— ключі;
— сертифікати.

Все що необхідні для встановлення захищеного з’єднання (див. рис. 2.1) [20].

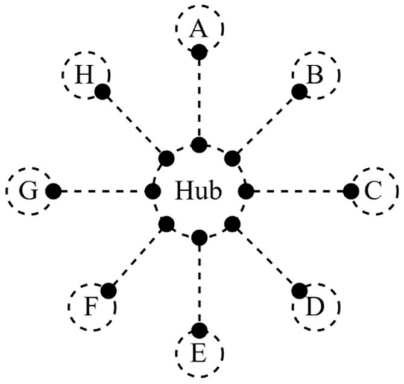


Рисунок 2.1 – Концепція мережі Hub-and-Spoke

Використання старого ноутбука як центрального VPN-шлюзу (Bare Metal сервер). Використання ноутбука Lenovo ThinkPad T480 у ролі центрального VPN-

шлюзу розглядається як інженерно обґрунтоване рішення, що поєднує доступність апаратної платформи з достатнім рівнем продуктивності для виконання серверних функцій, при цьому реалізація здійснюється відповідно до концепції Bare Metal, де програмне забезпечення розгортається безпосередньо на фізичному обладнанні без використання проміжних рівнів віртуалізації, що дозволяє уникнути додаткових накладних витрат на ресурси та забезпечити більш передбачувану поведінку системи під навантаженням, оскільки всі апаратні можливості використовуються напряму операційною системою та мережевими сервісами (див. рис. 2.2) [21].

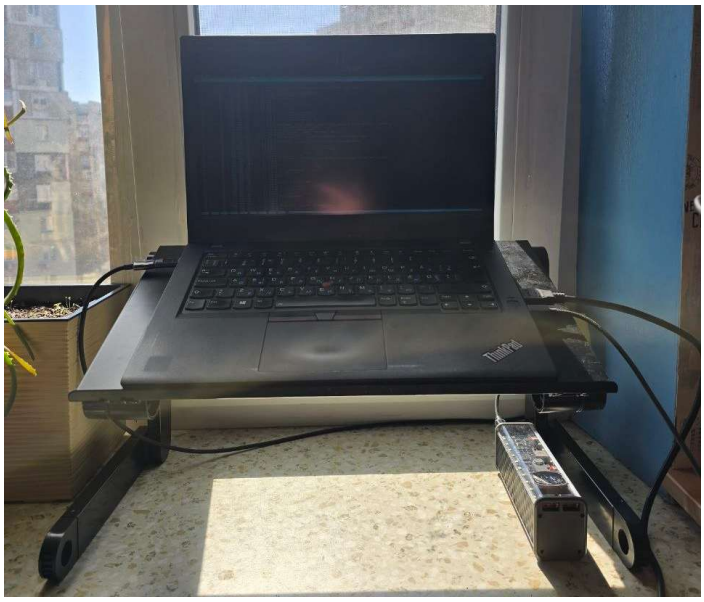


Рисунок 2.2 – Використання ноутбука як VPN-сервера

Обрана модель характеризується наявністю процесора Intel Core i5-8350U восьмого покоління, який має чотири фізичні ядра та підтримує вісім потоків обробки даних, що створює достатній рівень паралелізму для обслуговування кількох одночасних VPN-з’єднань, включаючи виконання криптографічних операцій шифрування та дешифрування трафіку, при цьому тепловий пакет на рівні 15 Вт визначає його як енергоефективне рішення, здатне працювати у тривалому безперервному режимі без значного тепловиділення та перевантаження системи охолодження, що є критично важливим для пристрою, який фактично виконує роль постійно увімкненого сервера.

Архітектурні можливості платформи доповнюються підтримкою сучасних інструкцій процесора, зокрема розширень для апаратного прискорення криптографії, що дозволяє частково компенсувати обмеження продуктивності у порівнянні з повноцінними серверними рішеннями, а можливість розширення оперативної пам’яті до значних обсягів створює потенціал для масштабування системи у випадку зростання кількості клієнтів або ускладнення сценаріїв використання, включаючи додаткові сервіси, які можуть бути розгорнуті на тому ж вузлі.

Найбільш вагомою особливістю, що визначила вибір саме цієї моделі, є реалізація технології Power Bridge, яка передбачає наявність двох незалежних акумуляторних модулів — внутрішнього з ємністю близько 24 Вт·год та зовнішнього, ємність якого може досягати 72 Вт·год залежно від конфігурації, така архітектура живлення дозволяє реалізувати механізм гарячої заміни зовнішнього акумулятора без вимкнення пристрою, що означає можливість фізичної заміни батареї під час роботи системи без переривання виконання серверних процесів та без розриву активних VPN-з’єднань, а отже забезпечується безперервність доступу користувачів до внутрішніх ресурсів навіть у випадках повної відсутності зовнішнього електроживлення.

З урахуванням цього підходу до організації живлення формується альтернативна модель забезпечення безперервної роботи, яка частково замінює класичні джерела безперебійного живлення, оскільки замість використання стаціонарних UPS із обмеженим часом автономності застосовується набір змінних акумуляторів, які можуть по черзі підключатися до системи, підтримуючи її функціонування протягом тривалого часу, така стратегія особливо актуальна в умовах нестабільного енергопостачання, де класичні рішення можуть бути економічно невиправданими або технічно обмеженими.

Енергоефективність обраної платформи у поєднанні з акумуляторною архітектурою дозволяє досягти тривалого часу автономної роботи, що виводить систему у категорію мобільних серверних рішень, здатних функціонувати незалежно від стаціонарної інфраструктури, а компактність форм-фактора

ноутбука разом із наявністю вбудованих засобів введення та виведення інформації спрощує обслуговування та розгортання, оскільки відпадає потреба у використанні додаткової периферії, такої як монітори або KVM-перемикачі.

У результаті формується підхід, за якого звичайний бізнес-ноутбук трансформується у повноцінний серверний вузол, що поєднує функції обробки мережевого трафіку, забезпечення захищеного доступу та зберігання даних із високим рівнем автономності, при цьому досягається баланс між продуктивністю, енергоефективністю та вартістю реалізації, що робить таке рішення доцільним для використання у малих та середніх інфраструктурах або у сценаріях, де важливо забезпечити швидке розгортання та мобільність системи без значних капітальних витрат на спеціалізоване серверне обладнання.

Організація віртуальної приватної мережі для доступу до внутрішніх ресурсів. Організація віртуальної приватної мережі для доступу до внутрішніх ресурсів формується як логічне продовження побудови захищеної інфраструктури віддаленого доступу, де основним завданням виступає створення ізольованого середовища, у межах якого забезпечується контрольований і безпечний обмін даними між клієнтами та серверними ресурсами, при цьому використання VPN-технології дозволяє реалізувати логічну мережу поверх публічної інфраструктури Інтернету, що фактично створює окремий захищений сегмент, незалежний від фізичної топології та зовнішніх мережевих умов, а така абстракція забезпечує не лише гнучкість, але й підвищений рівень безпеки за рахунок інкапсуляції та шифрування трафіку.

У процесі реалізації формується окремий адресний простір, який представлений у вигляді віртуальної підмережі, що функціонує паралельно до основної локальної мережі, до якої підключений VPN-сервер, причому ця підмережа використовує власний діапазон IP-адрес, що не перетинається із фізичною мережею, наприклад використання адресного простору типу "10.8.0.0/24" дозволяє логічно відокремити VPN-клієнтів від інших вузлів мережі, у результаті чого формується чітка сегментація, яка спрощує маршрутизацію та політики доступу, а також знижує ризик несанкціонованого доступу до внутрішніх ресурсів.

Під час підключення кожному клієнту динамічно або статично призначається унікальна внутрішня IP-адреса з цього віртуального діапазону, що дозволяє ідентифікувати кожен вузол у межах VPN-сегмента та забезпечити коректну маршрутизацію трафіку, при цьому на клієнтському пристрої створюється віртуальний мережевий інтерфейс, через який і відбувається передача всіх даних, спрямованих до внутрішніх ресурсів, а сам трафік інкапсулюється у VPN-пакети, шифрується та передається через публічну мережу до сервера, де відбувається його дешифрування та подальша обробка відповідно до налаштованих правил доступу.

Особливістю такої організації є те, що взаємодія між клієнтами не здійснюється безпосередньо, навіть незважаючи на те, що всі вони знаходяться в межах однієї віртуальної підмережі, оскільки обмін даними проходить через центральний вузол, який виконує функції маршрутизації та контролю доступу, а такий підхід відповідає концепції централізованого управління, де всі запити проходять перевірку та можуть бути обмежені або дозволені залежно від політик безпеки, що суттєво знижує ризик внутрішніх атак або неконтрольованого обміну даними між користувачами.

У межах реалізованої системи внутрішні ресурси розміщуються безпосередньо на VPN-сервері, що дозволяє максимально спростити архітектуру та забезпечити повний контроль над доступом до них, при цьому як приклад такого ресурсу використовується файловий сервіс на базі технології Samba, який виконує роль умовного корпоративного файлового сховища, доступ до якого можливий виключно після встановлення VPN-з’єднання, а це створює додатковий рівень ізоляції, оскільки навіть при наявності фізичного доступу до мережі, у якій розташований сервер, отримати доступ до ресурсу без проходження процедури автентифікації та встановлення захищеного тунелю неможливо (див. рис. 2.3).

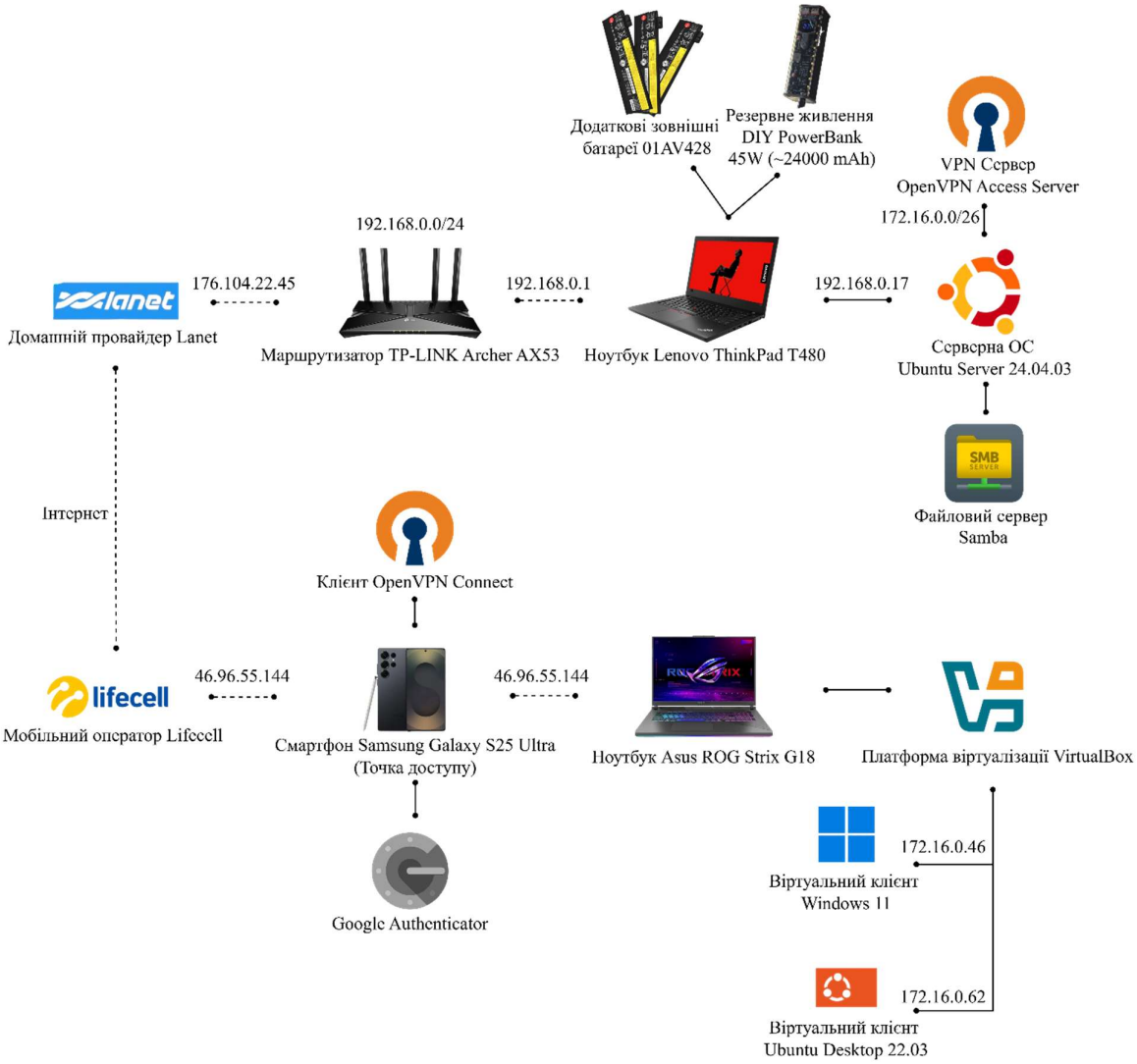


Рисунок 2.3 –Архітектура підключення користувачів до внутрішніх ресурсів

Використання файлового сервісу як тестового середовища дозволяє не лише перевірити працездатність VPN-з’єднання, але й змоделювати реальні сценарії використання системи, пов’язані з передачею даних, роботою з документами та доступом до спільних ресурсів, а процес копіювання файлів через захищений тунель наочно демонструє ефективність інкапсуляції та шифрування трафіку, при цьому одночасно можна оцінити вплив VPN на швидкість передачі даних та загальну продуктивність системи.

Обмеження доступу до ресурсу виключно через VPN-з’єднання підсилює модель безпеки, у межах якої внутрішня мережа фактично ізольована від

зовнішнього світу, а всі спроби доступу до неї без використання захищеного тунелю блокуються, що дозволяє реалізувати принцип "zero trust" у спрощеному вигляді, де кожне підключення проходить перевірку незалежно від його походження, у результаті чого досягається високий рівень захищеності даних та стабільності роботи системи навіть у умовах використання публічних або потенційно небезпечних мереж.

Взаємодія між клієнтами та VPN-сервером. Взаємодія між клієнтами та VPN-сервером у межах розробленої архітектури розглядається як окремий функціональний рівень, що забезпечує практичну перевірку працездатності всієї системи віддаленого доступу, при цьому для коректного моделювання реальних умов експлуатації було сформовано ізольований клієнтський стенд, який дозволяє відтворити різні сценарії підключення користувачів із різними операційними системами та мережевими умовами, а основою такого стенда стало використання середовища віртуалізації, розгорнутого на окремому пристрої, що функціонує незалежно від фізичного VPN-сервера, завдяки чому досягається чітке розмежування між серверною та клієнтською частинами інфраструктури.

Для реалізації клієнтського середовища використано гіпервізор VirtualBox, який дозволяє створювати ізольовані віртуальні машини з різними операційними системами, зокрема було розгорнуто конфігурації на базі Windows 11 та Ubuntu Desktop, що забезпечує перевірку кросплатформеності рішення та демонструє універсальність підходу до організації VPN-доступу, при цьому обмеження кількості одночасних підключень визначається використанням серверного рішення OpenVPN Access Server, безкоштовна версія якого підтримує лише два активних клієнти, а така особливість обумовила вибір саме двох віртуальних машин для тестування, чого достатньо для демонстрації основних принципів роботи системи без необхідності використання додаткових ліцензій або переходу на альтернативні рішення.

Наявність двох різних клієнтських операційних систем дозволяє перевірити поведінку VPN-з’єднання в різних середовищах, включаючи обробку мережевих налаштувань, імпорт конфігураційних файлів та стабільність встановлення тунелю,

при цьому додатково передбачається використання смартфону Android як точки доступу, що імітує сценарій доступу з використанням мобільних мереж. Такий підхід дозволяє оцінити роботу системи в умовах змінних мережевих характеристик, зокрема при використанні мобільного інтернету з підвищеною затримкою та нестабільною пропускною здатністю.

На всіх клієнтських пристроях використовується застосунок OpenVPN Connect, який забезпечує повний цикл взаємодії з VPN-сервером, включаючи:

1. імпорт конфігураційних файлів формату ".ovpn";
2. ініціалізацію з’єднання;
3. обробку процесу автентифікації;
4. підтримку захищеного тунелю.

Сам процес взаємодії реалізується як послідовність дій, де після запуску клієнтського застосунку відбувається встановлення з’єднання із сервером за заданою адресою, проходження процедури автентифікації та отримання внутрішньої IP-адреси з віртуального адресного простору, після чого відкривається доступ до внутрішніх ресурсів, доступних у межах VPN-сегмента, включаючи файловий сервер (див. рис. 2.4).

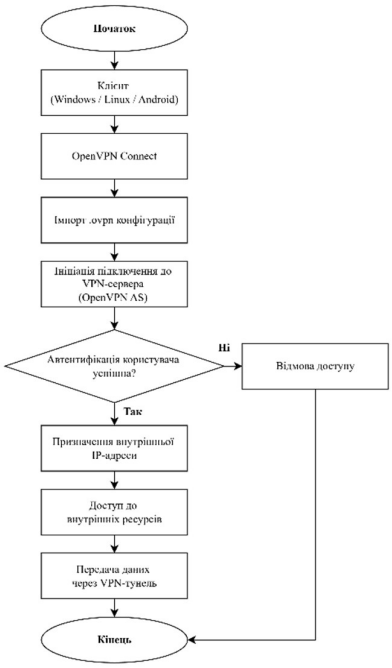


Рисунок 2.4 – Взаємодія клієнтів із сервером

Така модель взаємодії дозволяє забезпечити чітке відокремлення клієнтського середовища від серверного, оскільки всі запити проходять через захищений тунель і обробляються центральним вузлом, що створює умови для централізованого контролю доступу та моніторингу активності користувачів, а також дозволяє моделювати різні сценарії навантаження, оцінюючи поведінку системи при одночасному підключенні декількох клієнтів та виконанні операцій передачі даних.

Роль OpenVPN як основного компонента архітектури системи. Центральним програмним компонентом архітектури виступає OpenVPN, який забезпечує реалізацію всіх базових функцій VPN-інфраструктури, включаючи:

1. створення захищених тунелів між клієнтами та сервером;
2. шифрування переданих даних;
3. управління мережевими параметрами.

При цьому важливою особливістю є підтримка механізмів автентифікації на основі сертифікатів, що дозволяє ідентифікувати клієнтів та забезпечити контроль доступу до ресурсів, а додаткове впровадження багатофакторної автентифікації значно підвищує рівень захищеності системи.

У рамках реалізації використовується підхід, за якого доступ користувача базується не лише на наявності конфігураційного файлу, але й на додатковій перевірці через одноразові коди, що генеруються застосунками типу Google Authenticator, у результаті чого навіть у випадку компрометації сертифіката або конфігураційного файлу стороння особа не зможе отримати доступ до системи без фізичного доступу до пристрою користувача, на якому збережено другий фактор автентифікації.

Функціональні можливості платформи доповнюються підтримкою інструментів автоматизації, зокрема використанням утиліти командного рядка "sacli", яка дозволяє:

1. виконувати налаштування серверу;
2. створювати користувачів;
3. генерувати конфігураційні файли без використання веб-інтерфейсу.

Все це значно знижує ймовірність помилок, пов’язаних із ручним введенням даних, і спрощує масштабування системи, оскільки більшість операцій може бути виконана автоматизовано через скрипти, інтегровані у загальний процес адміністрування.

У результаті поєднання віртуалізованого клієнтського середовища, централізованого VPN-сервера та засобів автоматизації формується цілісна система, яка дозволяє не лише перевірити працездатність запропонованої архітектури, але й продемонструвати її придатність для використання у реальних умовах, забезпечуючи стабільний, захищений та контрольований доступ до внутрішніх ресурсів незалежно від платформи клієнта або умов мережевого підключення.

2.2 Побудова системи віддаленого доступу

Розгортання системи віддаленого доступу неможливо розглядати як абстрактну концепцію без практичної реалізації, оскільки лише послідовне проходження всіх етапів — від підготовки фізичного обладнання до створення повноцінної VPN-інфраструктури — дозволяє сформувати цілісне уявлення про її функціонування, стабільність та ефективність, тому процес побудови передбачає не лише інсталяцію програмного забезпечення, але й комплексне налаштування мережевої взаємодії, механізмів безпеки та засобів автоматизації, які у сукупності формують працездатну систему доступу до внутрішніх ресурсів через захищені канали зв’язку, а додаткове розгортання файлового сервісу на сервері дозволяє перевірити реальні сценарії використання, що наближені до умов корпоративного середовища.

Встановлення серверної операційної системи на фізичне обладнання. Першим логічним етапом такої побудови стає підготовка серверного середовища, яка реалізується шляхом встановлення операційної системи безпосередньо на

фізичне обладнання, причому обраний підхід відповідає концепції використання Bare Metal, що виключає накладні витрати на віртуалізацію та дозволяє максимально ефективно використовувати доступні апаратні ресурси, у якості базової системи застосовується Ubuntu Server 24.04.03 LTS, яка характеризується довгостроковою підтримкою, стабільністю та широкими можливостями адміністрування, що робить її оптимальним вибором для побудови серверної інфраструктури, підготовка до встановлення включає створення завантажувального USB-накопичувача об’ємом 32 ГБ, для чого використовується утиліта Ventoy, яка значно спрощує процес роботи з інсталяційними образами, оскільки дозволяє розміщувати декілька ISO-файлів на одному носії без необхідності повторного форматування, що суттєво економить час і спрощує процес оновлення або заміни образів операційних систем. Після створення завантажувального носія здійснюється перехід до етапу безпосереднього встановлення системи, який починається з налаштування порядку завантаження у BIOS або через Boot-меню ноутбука, де обирається запуск із USB-накопичувача, після чого відкривається інтерфейс Ventoy із переліком доступних образів, серед яких обирається Ubuntu Server 24.04.03 LTS, далі запускається стандартний процес інсталяції, під час якого обирається варіант негайного встановлення без попереднього тестування системи, що дозволяє скоротити час розгортання, при цьому ключовим рішенням стає вибір мінімальної конфігурації без графічного інтерфейсу, оскільки відсутність GUI зменшує споживання оперативної пам’яті та навантаження на центральний процесор, а також знижує кількість потенційних вразливостей, пов’язаних із графічними компонентами системи, що у свою чергу підвищує загальний рівень безпеки серверного середовища (див. рис. 2.5).

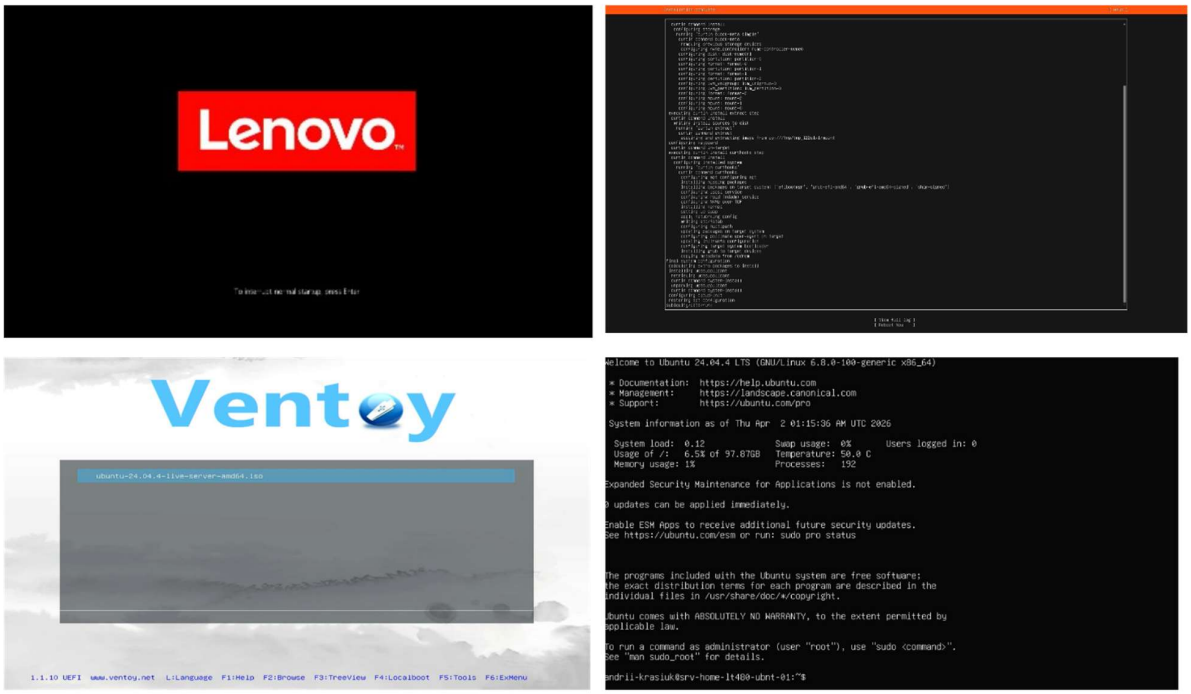


Рисунок 2.5 – Встановлення серверної ОС Ubuntu Server 24.04.03 LTS

У процесі інсталяції додатково виконуються базові налаштування, які формують фундамент для подальшої роботи системи, зокрема створюється адміністративний обліковий запис, що використовується для керування сервером, налаштовується служба SSH для забезпечення віддаленого доступу, що дозволяє адмініструвати систему без фізичної присутності біля пристрою, а також виконується конфігурація дискової підсистеми, яка забезпечує коректне зберігання даних і стабільну роботу файлової системи, у результаті формується оптимізоване серверне середовище, яке поєднує у собі мінімальне використання ресурсів, високу стабільність та готовність до подальшого розгортання мережевих сервісів.

Налаштування мережевих інтерфейсів сервера. Налаштування мережевих інтерфейсів сервера є одним із найбільш критичних етапів побудови всієї системи віддаленого доступу, оскільки без коректної мережевої конфігурації подальше розгортання VPN-інфраструктури стає неможливим або працює нестабільно, при цьому після встановлення операційної системи відсутність налаштованого доступу до мережі Інтернет є очікуваним станом, що пояснюється необхідністю ручного визначення параметрів мережевої взаємодії відповідно до обраної архітектури,

особливо у випадку використання мобільного серверного рішення, де важливим стає не лише стабільність з’єднання, але й гнучкість у зміні середовища підключення.

Особливістю апаратної платформи є наявність двох мережевих інтерфейсів — провідного (LAN) та безпровідного (Wi-Fi), причому з технічної точки зору провідне підключення забезпечує нижчі затримки, менший рівень джитера та вищу стабільність передачі даних, однак у контексті реалізації мобільного VPN-шлюзу пріоритет надається саме безпровідному інтерфейсу, оскільки такий підхід дозволяє використовувати сервер як переносний вузол, який може бути швидко переміщений між різними локаціями та підключений до доступної мережі без необхідності фізичної інфраструктури у вигляді кабелів, що у поєднанні з автономним живленням формує повноцінну мобільну платформу, здатну функціонувати незалежно від конкретного місця розгортання.

Після визначення типу підключення наступним кроком стає налаштування мережевих параметрів у середовищі операційної системи, де використовується утиліта Netplan, яка відповідає за декларативне описання конфігурації мережевих інтерфейсів, у стандартній інсталяції Ubuntu Server відповідний конфігураційний файл може бути відсутній або мати базовий вигляд, тому виникає необхідність створення нового файлу у директорії "/etc/netplan" із назвою на зразок "00-netplan-config.yaml", після чого через текстовий редактор вноситься опис параметрів мережі, що включає визначення версії конфігурації, вибір конкретного безпровідного інтерфейсу, який попередньо ідентифікується за допомогою системних утиліт, наприклад "ip addr" або "ifconfig", де відображається його ім’я, наприклад "wlp3s0" [23].

У конфігураційному файлі задається відмова від використання DHCP, що означає перехід до статичної адресації, оскільки для серверних систем важливо мати незмінну IP-адресу, яка буде використовуватися клієнтами для підключення, тому вручну прописується адреса, наприклад "192.168.0.17" із маскою підмережі "/24", додатково визначається шлюз за замовчуванням, який зазвичай відповідає адресі маршрутизатора у локальній мережі, наприклад "192.168.0.1", а також

налаштовуються DNS-сервери, серед яких можуть використовуватися публічні адреси, такі як "8.8.8.8" та "1.1.1.1", що забезпечують стабільну роботу системи доменних імен, у випадку безпровідного підключення окремо вказуються параметри доступу до Wi-Fi мережі, включаючи SSID та пароль, що дозволяє серверу автоматично підключатися до заданої мережі після запуску (див. рис. 2.6).



Рисунок 2.6 – Налаштування мережевих інтерфейсів через власний config

Після завершення редагування конфігураційного файлу застосування змін виконується за допомогою команди "netplan apply", у результаті чого система ініціалізує нові параметри мережі, а перевірка коректності здійснюється через перегляд інформації про інтерфейси та тестування доступу до зовнішніх ресурсів, наприклад за допомогою команди "ping", що дозволяє впевнитися у наявності підключення до Інтернету та правильності маршрутизації [23].

Паралельно із налаштуванням сервера виконується конфігурація маршрутизатора, до якого підключається пристрій, оскільки навіть при встановленні статичної адреси на самому сервері необхідно забезпечити її

резервування на рівні DHCP-сервера маршрутизатора, щоб уникнути конфліктів адрес у мережі, для цього через веб-інтерфейс пристрою, наприклад маршрутизатора TP-Link Archer AX53, виконується прив'язка MAC-адреси мережевого інтерфейсу сервера до обраної IP-адреси, у результаті чого навіть при зміні умов підключення або перезавантаженні пристрою адреса залишається сталою, що є критично важливим для роботи VPN-сервісу (див. рис. 2.7).

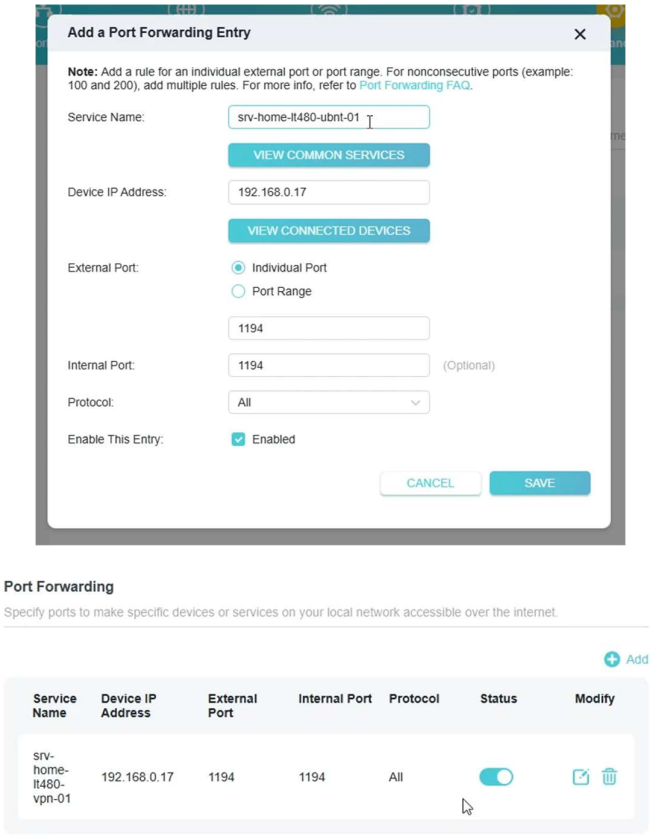


Рисунок 2.7 – Резервування IP-адреси сервера на маршрутизаторі TP-Link

Додатково виконується ще один важливий системний параметр, який безпосередньо впливає на здатність сервера обробляти мережевий трафік між різними сегментами мережі, у конфігураційному файлі "/etc/sysctl.conf" розкоментовується рядок "net.ipv4.ip_forward=1", що активує пересилання пакетів на рівні ядра Linux, після застосування змін сервер отримує можливість функціонувати як маршрутизатор, тобто передавати пакети між VPN-інтерфейсом та локальною мережею, без чого клієнти, навіть після успішного підключення до

VPN, не змогли б отримати доступ до внутрішніх ресурсів, оскільки трафік не був би маршрутизований далі за межі тунелю [23].

У сукупності виконані дії формують повноцінне мережеве середовище, у якому сервер має стабільну статичну адресу, коректно взаємодіє з маршрутизатором, підтримує доступ до Інтернету та здатний обробляти і перенаправляти трафік між різними мережевими сегментами, що створює необхідну основу для подальшого розгортання VPN-сервера та забезпечує стабільну роботу всієї системи віддаленого доступу навіть у умовах змінного мережевого середовища.

Розгортання та конфігурація OpenVPN Access Server. Розгортання та конфігурація VPN-сервера є логічним продовженням побудови всієї системи віддаленого доступу, оскільки після підготовки апаратної платформи та базового мережевого середовища виникає необхідність у впровадженні програмного компонента, який забезпечує встановлення захищених з’єднань між клієнтами та внутрішньою мережею, у даній архітектурі таким компонентом обрано OpenVPN Access Server, що поєднує у собі гнучкість класичної реалізації OpenVPN із наявністю зручного веб-інтерфейсу для адміністрування, а також підтримку командного керування через консольні утиліти, що дозволяє реалізувати автоматизацію процесів керування користувачами та конфігураціями (див. табл. 2.1) [22].

Таблиця 2.1 – Основні параметри конфігурації OpenVPN-сервера

Параметр	Значення	Призначення
port	1194	Порт для встановлення VPN-з’єднання
proto	udp	Протокол передачі даних
dev	tun	Тип віртуального мережевого інтерфейсу
server	172.16.0.0/26'	Внутрішня підмережа VPN
ca	ca.crt	Сертифікат центру сертифікації
cert	server.crt	Сертифікат сервера
key	server.key	Приватний ключ сервера
cipher	AES-256-GCM	Алгоритм шифрування
auth	SHA256	Алгоритм автентифікації
keepalive	10 120	Контроль активності клієнтів

Встановлення програмного забезпечення здійснюється безпосередньо на підготовлену систему Ubuntu Server 24.04.03 LTS, використовуючи офіційні репозиторії розробника, що гарантує отримання актуальної та стабільної версії продукту, попередньо виконується оновлення пакетної бази та встановлення необхідних допоміжних компонентів, таких як "ca-certificates", "wget", "gnupg" та "net-tools", після чого додається ключ репозиторію OpenVPN та відповідний запис до списку джерел пакетів, наступним кроком виконується інсталяція пакета openvpn-as, у результаті чого система автоматично розгортає серверну частину, створює необхідні служби та конфігураційні файли, а також генерує початкові параметри доступу до адміністративного інтерфейсу (див. рис. 2.8).

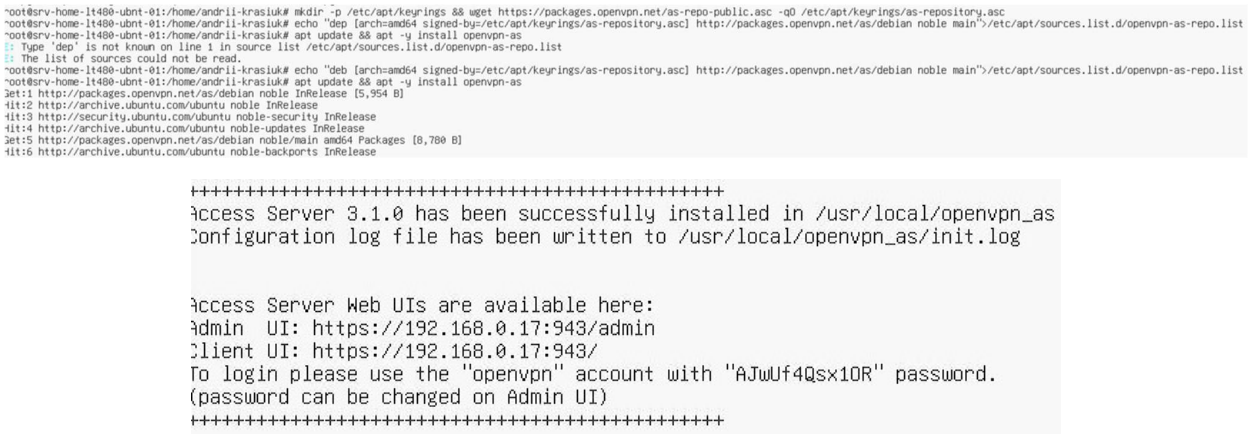


Рисунок 2.8 – Процес розгортання OpenVPN Access Server

Після завершення інсталяції виводиться службове повідомлення, яке містить інформацію про:

— шлях встановлення;

— статус конфігурації;

— адреси доступу до веб-інтерфейсів.

Адміністративна панель доступна через захищений HTTPS-запит за портом 943, а клієнтський інтерфейс — за окремою URL-адресою, для первинного входу використовується стандартний обліковий запис із автоматично згенерованим паролем, який надалі змінюється з міркувань безпеки, підключення до веб-

інтерфейсу дозволяє виконати базове налаштування сервера без необхідності редагування конфігураційних файлів вручну, що значно спрощує процес початкового розгортання (див. рис. 2.9) [22].

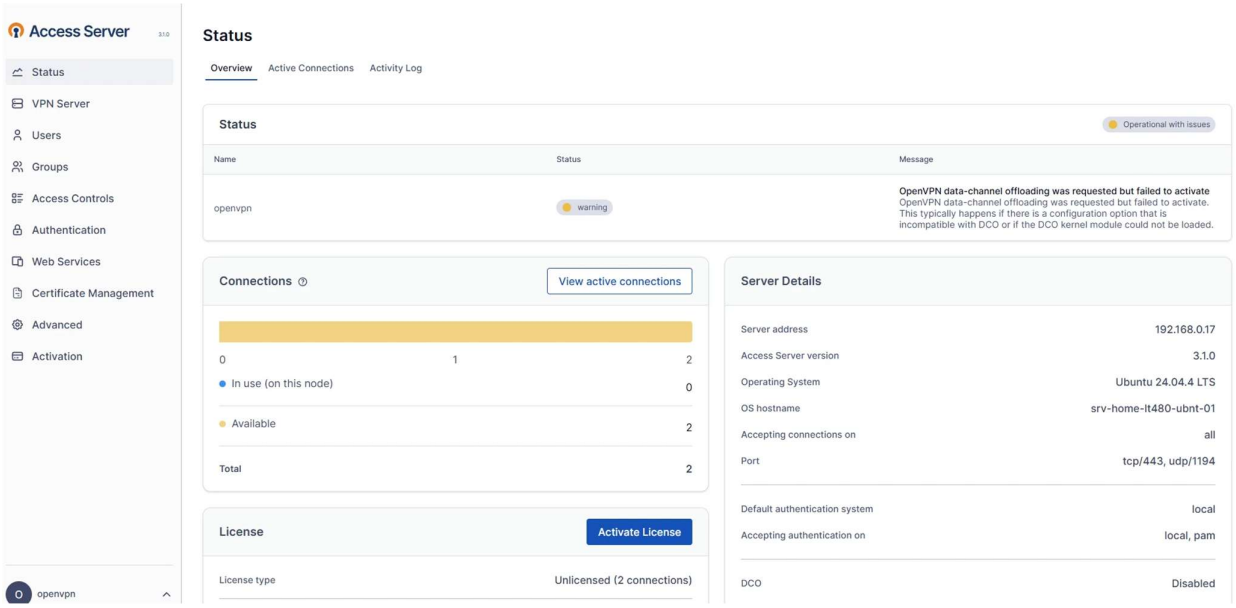


Рисунок 2.9 – Інтерфейс OpenVPN Access Server

Конфігурація сервера передбачає визначення ключових параметрів функціонування VPN-інфраструктури, зокрема задається публічна IP-адреса або доменне ім’я, яке буде використовуватися клієнтами для встановлення з’єднання, у випадку використання статичної IP-адреси від інтернет-провайдера забезпечується стабільність доступу незалежно від зміни мережевого середовища, також визначається транспортний протокол для передачі даних, де основним обирається UDP завдяки меншому рівню накладних витрат та кращій продуктивності, тоді як TCP може використовуватися як резервний варіант у мережах із жорсткими обмеженнями (див. рис. 2.10).

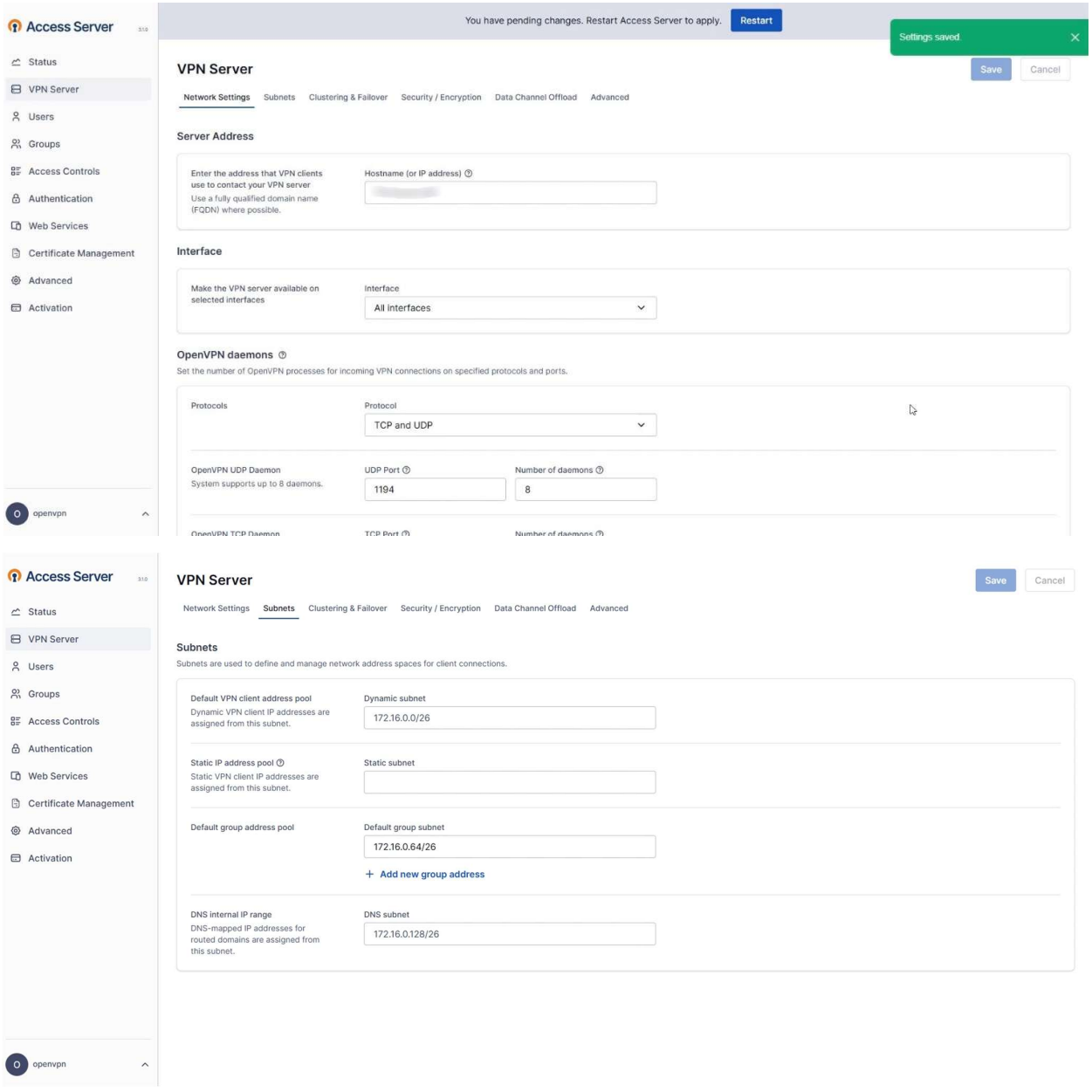


Рисунок 2.10 – Конфігурація VPN-сервера

Окрему увагу приділено налаштуванню внутрішнього адресного простору VPN-мережі, створюється ізольована підмережа з власним діапазоном IP-адрес, яка не перетинається з локальною мережею, що дозволяє уникнути конфліктів маршрутизації, визначаються окремі пулі адрес для динамічного призначення клієнтам, службових груп та DNS-запитів, у результаті після застосування конфігурації сервер автоматично створює віртуальні мережеві інтерфейси, які

використовуються для обробки VPN-трафіку, їх наявність підтверджується через системні утиліти перегляду мережевих інтерфейсів [22].

Важливим етапом є інтеграція VPN-сервера з мережею доступу до Інтернету, оскільки фізично сервер розміщений за маршрутизатором, необхідно забезпечити можливість зовнішніх підключень, для цього на мережевому обладнанні виконується налаштування перенаправлення портів, зокрема UDP-порт 1194 спрямовується на внутрішню IP-адресу сервера, у результаті всі запити ззовні, що надходять на цей порт, автоматично передаються до VPN-служби, що працює на ноутбуці.

Питання безпеки при відкритті порту вирішується завдяки використанню механізму TLS-Auth, який реалізує принцип відхилення всіх неавторизованих пакетів ще до етапу встановлення з’єднання, сервер реагує лише на запити, які містять правильний криптографічний підпис, унаслідок чого будь-які сторонні сканування або спроби взаємодії ігноруються без відповіді, відкритий порт фактично виглядає як недоступний для зовнішнього середовища, що значно знижує ймовірність атак типу перебору або виявлення сервісу [22].

Після завершення всіх налаштувань виконується перезапуск служби для застосування змін, система переходить у робочий стан, що підтверджується як через веб-інтерфейс, так і через перевірку статусу служби в операційній системі, додатково перевіряється коректність створених мережевих інтерфейсів та доступність сервера ззовні, у результаті формується повноцінний VPN-шлюз, готовий до обробки підключень клієнтів.

Формування конфігураційних файлів для підключення клієнтів. Заключний етап побудови системи віддаленого доступу пов’язаний із формуванням клієнтських конфігурацій, оскільки саме на цьому рівні відбувається перехід від підготовленої серверної інфраструктури до фактичного надання доступу кінцевим користувачам і пристроям, при цьому класичний підхід із ручним створенням конфігурацій через веб-інтерфейс адміністратора втрачає ефективність у випадках, коли кількість користувачів зростає, а також підвищується ризик допущення помилок через людський фактор, тому у межах реалізації обрано автоматизований

підхід, який базується на використанні консольної утиліти "sacli", що входить до складу OpenVPN Access Server і дозволяє напряму взаємодіяти з внутрішньою конфігураційною базою сервера.

У рамках цього підходу реалізовано два окремих Bash-скрипти, кожен із яких орієнтований на різні сценарії використання VPN-інфраструктури, перший сценарій стосується автоматизованого підключення машин та серверів у режимі "Machine-to-Machine", де не передбачається участь користувача під час встановлення з’єднання, а другий — організації безпечного доступу для співробітників із використанням багатофакторної автентифікації, що відповідає сучасним вимогам до захисту корпоративних ресурсів.

Обробка сценарію для машин передбачає використання вхідного текстового файлу, який містить список ідентифікаторів пристроїв, кожен рядок якого інтерпретується як окремий клієнт, під час виконання скрипта ці значення очищуються від зайвих символів, нормалізуються шляхом заміни пробілів на символи підкреслення та перевіряються на наявність порожніх записів, після чого для кожного елемента створюється відповідний обліковий запис у системі OpenVPN Access Server, налаштовується тип підключення "user_connect" та активується параметр "autologin", який дозволяє виконувати підключення без введення пароля, далі відбувається генерація конфігураційного файлу у форматі ".ovpn", що одразу зберігається у визначену директорію, завершальним етапом є оновлення конфігурації сервера у пам’яті, що забезпечує застосування змін без необхідності ручного втручання (див. рис. 2.11). (Додаток А)

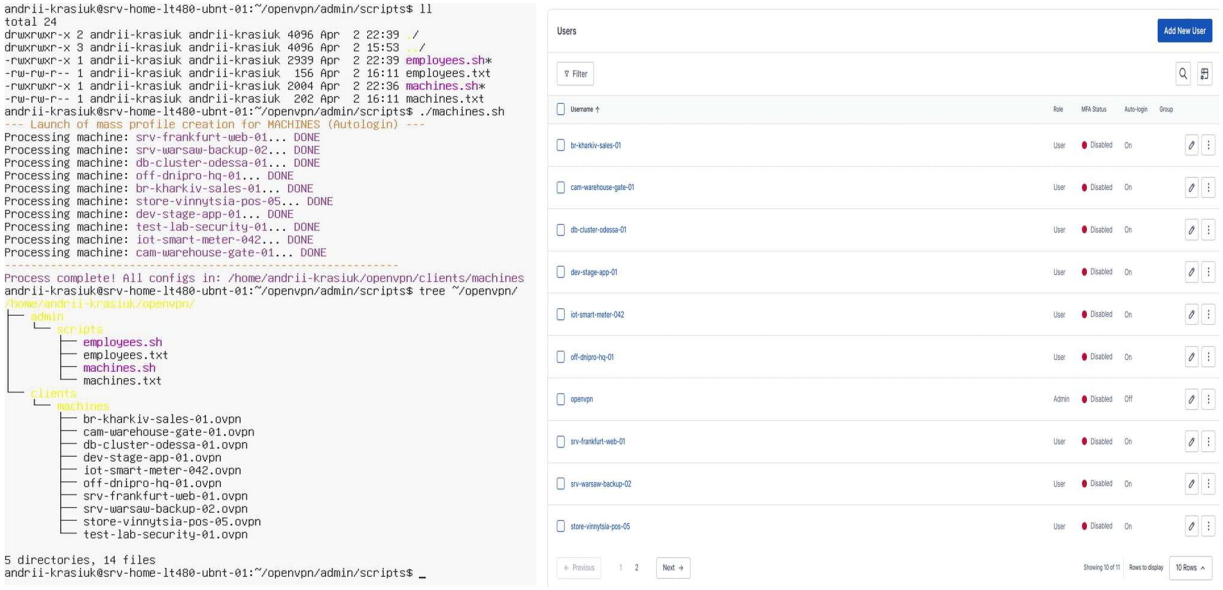


Рисунок 2.11 – Генерація машинних конфігурацій

Такий підхід дозволяє швидко створювати десятки або навіть сотні конфігурацій для серверів, IoT-пристроїв або інших автоматизованих систем, де відсутня необхідність у додатковому рівні автентифікації, при цьому значно спрощується інтеграція між різними компонентами інфраструктури, а сам процес розгортання стає повторюваним та стандартизованим, що особливо важливо у великих системах.

Паралельно реалізовано другий сценарій, орієнтований на співробітників, де вимоги до безпеки значно вищі, обробка також базується на текстовому файлі, який містить список користувачів, зазвичай у форматі ім’я та прізвище, кожен запис проходить нормалізацію та використовується для створення облікового запису у системі, після чого автоматично встановлюється тимчасовий пароль за замовчуванням, який має бути змінений під час першого входу, додатково користувач прив’язується до групи, що дозволяє централізовано керувати політиками доступу (див. рис. 2.12).

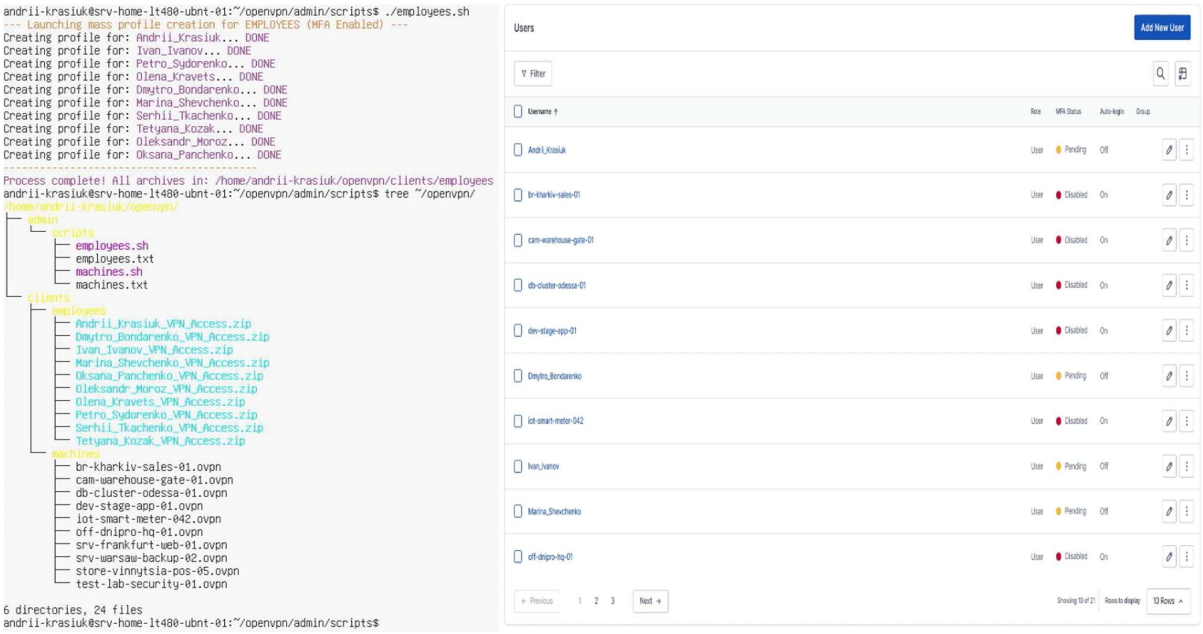


Рисунок 2.12 – Генерація клієнтських конфігурацій

Особливістю цього сценарію є автоматичне налаштування багатофакторної автентифікації, де для кожного користувача генерується унікальний секретний ключ, який використовується у додатках типу Google Authenticator, далі активується обов’язкове використання другого фактору, у результаті під час підключення необхідно не лише мати конфігураційний файл і пароль, але й одноразовий код, який генерується на мобільному пристрої користувача, що значно підвищує рівень захищеності системи навіть у випадку компрометації основних облікових даних.

У процесі виконання скрипта формується не лише конфігураційний файл ".ovpn", але й додатковий текстовий файл із детальною інформацією для користувача, включаючи логін, тимчасовий пароль та секретний ключ для налаштування MFA, ці файли автоматично об’єднуються в архів, який передається користувачу як готовий пакет доступу, після чого тимчасові файли видаляються, що дозволяє уникнути накопичення зайвих даних на сервері (див. рис. 2.13).

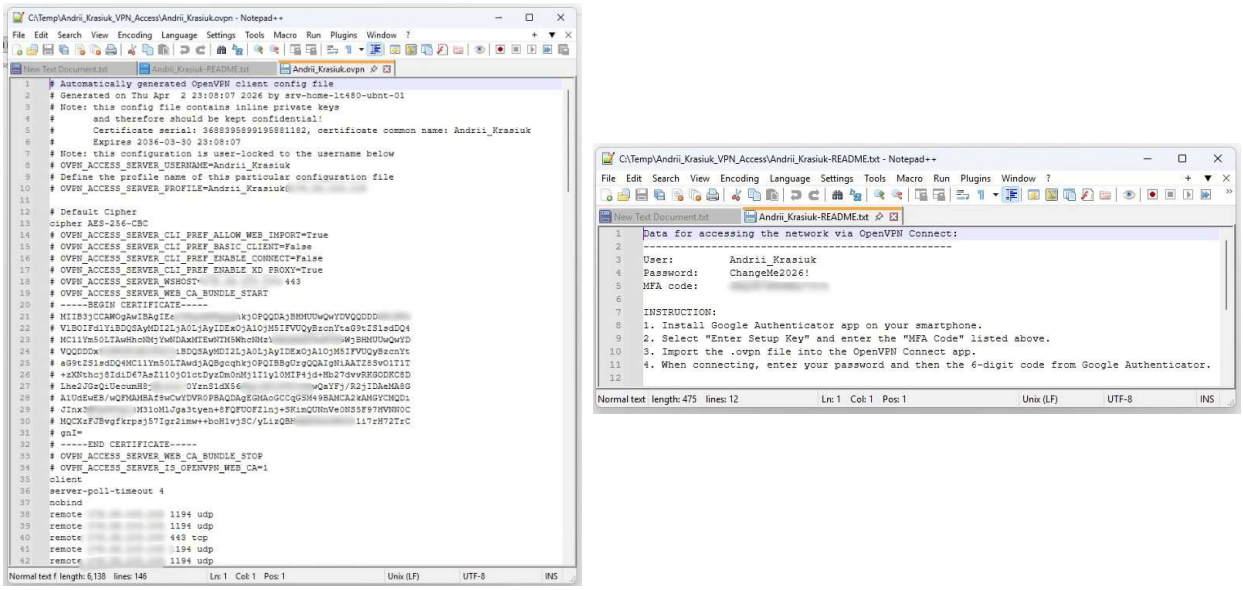


Рисунок 2.13 – Формування унікальних файлів зі детальною інформацією для співробітника

Результатом виконання обох сценаріїв стає чітко структурована директорія, у якій окремо зберігаються конфігурації для машин та співробітників, що спрощує адміністрування та подальшу підтримку системи, така організація дозволяє швидко знаходити необхідні файли, контролювати доступи та виконувати їх відкликання у разі потреби, оскільки всі конфігурації прив’язані до конкретних облікових записів у системі (див. рис. 2.14). (Додаток Б)



Рисунок 2.14 – Робоча структура батьківської директорії після ініціації bash-скриптів

Застосування автоматизації у процесі створення клієнтських конфігурацій дозволяє значно скоротити час розгортання інфраструктури, мінімізувати кількість помилок, пов’язаних із ручним введенням параметрів, а також забезпечити єдині стандарти для всіх користувачів і пристроїв, у результаті система набуває властивостей масштабованості та гнучкості, що дозволяє адаптувати її до різних умов експлуатації та вимог безпеки, поєднання двох підходів — для машин та для людей — формує універсальну модель доступу, яка охоплює як автоматизовані сервіси, так і інтерактивну роботу користувачів, забезпечуючи при цьому високий рівень контролю та захисту внутрішніх ресурсів.

2.3 Алгоритм функціонування системи

Розгляд алгоритму функціонування системи віддаленого доступу дозволяє перейти від практичної реалізації до глибшого розуміння внутрішніх процесів, що відбуваються під час роботи VPN-інфраструктури, оскільки окрім фізичного розгортання компонентів та їх базового налаштування важливо усвідомлювати логіку встановлення з’єднання, механізми ідентифікації та автентифікації, принципи автоматизованого створення конфігурацій і особливості передачі даних через захищений канал, такий підхід дозволяє сформувати цілісне уявлення про систему не лише як про набір компонентів, а як про узгоджену послідовність алгоритмічних дій, що забезпечують її функціонування.

Алгоритм підключення користувача до внутрішньої мережі через VPN. Алгоритм підключення користувача до внутрішньої мережі через VPN починається з клієнтського пристрою, на якому встановлено програмне забезпечення для підключення, наприклад OpenVPN Connect, та імпортовано конфігураційний файл формату .ovpn, який містить усі необхідні параметри для встановлення з’єднання, включаючи:

— публічну IP-адресу або доменне ім’я сервера;

— номер порту;
— транспортний протокол;
— криптографічні налаштування;
— сертифікати;
— ключі.

Після запуску клієнтського застосунку ініціюється запит на встановлення з’єднання, який формується відповідно до заданих параметрів і передається у мережу Інтернет.

Мережеві пакети, що генеруються клієнтом, проходять через проміжну інфраструктуру, включаючи мережу провайдера та глобальну мережу, після чого досягають маршрутизатора, до якого підключено VPN-сервер, у даному випадку запит потрапляє на зовнішній інтерфейс маршрутизатора та обробляється правилом перенаправлення портів, яке спрямовує трафік з визначеного порту на внутрішню IP-адресу сервера, у результаті пакет передається до локальної мережі та досягає фізичного пристрою, на якому розгорнуто VPN-службу.

На стороні сервера запит приймається службою OpenVPN Access Server, де перед обробкою виконується перевірка коректності пакета з використанням механізму TLS-Auth, що дозволяє відфільтрувати будь-які неавторизовані або підроблені запити ще до етапу встановлення з’єднання, у випадку відповідності пакета очікуваним параметрам ініціюється процес встановлення сеансу, який передбачає обмін службовими повідомленнями між клієнтом і сервером.

Після проходження початкового етапу встановлення з’єднання клієнт отримує внутрішню IP-адресу з пулу, визначеного у конфігурації VPN-сервера, наприклад із діапазону "172.16.0.0/26", що дозволяє інтегрувати клієнтський пристрій у віртуальну мережу, створену поверх фізичної інфраструктури, у результаті клієнт стає логічним учасником цієї мережі та отримує можливість взаємодіяти з ресурсами, доступними через VPN-шлюз.

Подальша передача трафіку здійснюється через встановлений тунель, при цьому всі пакети, що генеруються клієнтським пристроєм і спрямовані до внутрішніх ресурсів, інкапсулюються у VPN-пакети та передаються до сервера, де

відбувається їх розпакування та маршрутизація до відповідних сервісів у локальній мережі, зворотний трафік проходить аналогічний шлях у зворотному напрямку, що забезпечує прозору взаємодію між клієнтом і внутрішніми ресурсами незалежно від їх фізичного розташування (див. рис. 2.15).

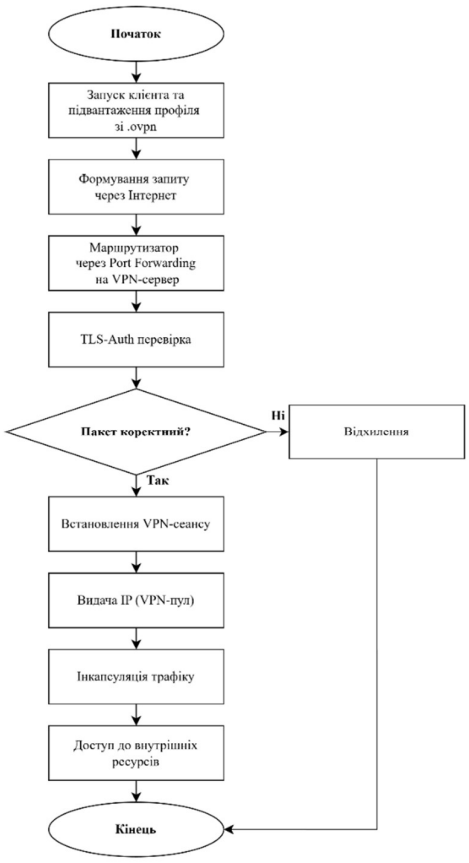


Рисунок 2.15 – Алгоритм підключення користувача
до внутрішньої мережі через VPN

Така організація роботи дозволяє досягти ефекту присутності клієнта у внутрішній мережі, навіть якщо фактичне підключення здійснюється через зовнішні мережі, наприклад мобільний Інтернет або іншу віддалену інфраструктуру, при цьому весь трафік проходить через VPN-сервер, який виконує функцію єдиного шлюзу, забезпечуючи контроль доступу, фільтрацію та маршрутизацію, що формує централізовану модель управління мережею.

Процес встановлення захищеного TLS-з'єднання. У межах аналізу функціонування системи віддаленого доступу особливу увагу доцільно приділити саме процесу встановлення захищеного TLS-з'єднання між клієнтом та сервером, оскільки саме цей механізм забезпечує конфіденційність, цілісність та автентичність переданих даних у межах VPN-інфраструктури, а також формує базу для подальшого функціонування всього захищеного каналу зв'язку, при цьому сам процес встановлення з'єднання реалізується у вигляді послідовності взаємопов'язаних кроків, які прийнято називати "TLS-handshake" і які відбуваються між клієнтським програмним забезпеченням та серверною частиною OpenVPN у момент ініціалізації з'єднання [24].

На початковому етапі клієнт ініціює встановлення з'єднання шляхом надсилання повідомлення типу "ClientHello", у якому міститься інформація про підтримувані версії протоколу TLS, перелік криптографічних алгоритмів (cipher suites), що можуть бути використані для шифрування, а також випадкове число, яке генерується для подальшого формування сесійного ключа, при цьому така передача параметрів дозволяє узгодити між сторонами найбільш оптимальні та безпечні умови подальшої взаємодії [24].

Після отримання цього запиту сервер формує відповідь "ServerHello", у якій обирає конкретну версію протоколу TLS із запропонованого списку, визначає криптографічний алгоритм, що буде використовуватись у сесії, а також генерує власне випадкове число, яке разом із числом клієнта надалі використовується у процесі генерації ключів шифрування, і таким чином відбувається узгодження параметрів захищеного каналу ще до початку передачі будь-яких чутливих даних [24].

На наступному етапі сервер передає клієнту свій цифровий сертифікат, який містить публічний ключ сервера та підписаний довіреним центром сертифікації, і цей сертифікат виступає основою для підтвердження автентичності сервера, оскільки клієнт має можливість перевірити його дійсність за допомогою вбудованих механізмів довіри, що включають перевірку підпису центру

сертифікації, відповідність доменного імені або IP-адреси, а також контроль строку дії сертифіката.

Після отримання сертифіката клієнт виконує процедуру його перевірки, у межах якої здійснюється аналіз ланцюжка довіри, перевіряється, чи не був сертифікат відкликаний, а також чи відповідає він очікуваним параметрам підключення, і лише у випадку успішного проходження всіх перевірок продовжується процес встановлення з'єднання, що дозволяє виключити можливість атак типу «людина посередині» та забезпечити високий рівень довіри до сервера.

Далі відбувається генерація сесійного ключа, при якій клієнт формує секретне значення, що передається серверу у зашифрованому вигляді з використанням публічного ключа сервера, отриманого із сертифіката, і завдяки цьому навіть у випадку перехоплення трафіку сторонніми особами неможливо отримати доступ до цього ключа без наявності приватного ключа сервера, що гарантує захищеність процесу узгодження параметрів шифрування.

Після завершення обміну ключовими параметрами обидві сторони незалежно обчислюють однаковий сесійний ключ, який надалі використовується для симетричного шифрування всього переданого трафіку, при цьому зазвичай застосовуються сучасні алгоритми, такі як AES-256-GCM, що забезпечують високий рівень криптографічного захисту та ефективність обробки даних навіть при значному навантаженні [24].

У результаті виконання всіх зазначених етапів формується повноцінний захищений TLS-канал, який використовується як основа для створення VPN-тунелю, через який передається весь подальший трафік між клієнтом та сервером, і такий підхід дозволяє гарантувати, що всі дані, які передаються у межах системи, залишаються недоступними для сторонніх осіб та не можуть бути змінені або підроблені під час передачі (див. рис. 2.16).

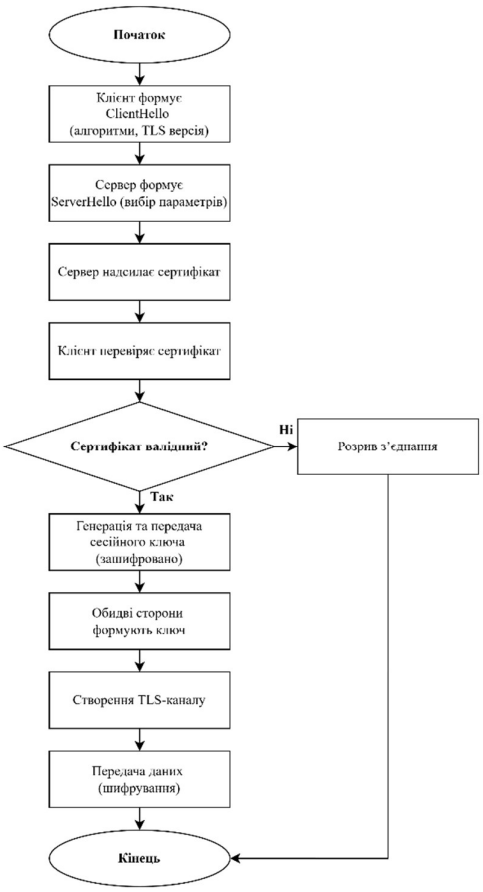


Рисунок 2.16 – Процес встановлення захищеного TLS-з'єднання

Важливою особливістю реалізації є використання тимчасових сесійних ключів, що змінюються для кожного нового з'єднання або навіть у межах однієї сесії, і така властивість відповідає концепції Perfect Forward Secrecy, згідно з якою навіть у випадку компрометації довгострокових ключів неможливо розшифрувати попередні сеанси зв'язку, оскільки кожен із них використовує унікальні параметри шифрування, що суттєво підвищує загальний рівень безпеки системи [24].

Алгоритм автентифікації користувача на основі сертифікатів і ключів. У межах побудованої системи віддаленого доступу механізм автентифікації організовано з урахуванням різних сценаріїв використання, що передбачає розділення підходів залежно від типу клієнта, оскільки автоматизовані вузли інфраструктури та кінцеві користувачі мають принципово різні вимоги до безпеки, зручності та способу взаємодії із VPN-сервером, у зв'язку з чим реалізовано два окремих алгоритмічних сценарії, що базуються на використанні сертифікатів і

криптографічних ключів, але відрізняються додатковими етапами перевірки та рівнем захисту.

У випадку машинного підключення, яке застосовується для серверів, службових систем або інших автоматизованих компонентів інфраструктури, процес автентифікації побудований за максимально спрощеною, але при цьому криптографічно захищеною схемою, де основним елементом виступає унікальний сертифікат, згенерований для конкретного вузла під час автоматизованого створення конфігурацій, і на початковому етапі клієнт ініціює підключення до VPN-сервера, передаючи власний сертифікат у межах встановлення TLS-з'єднання, після чого сервер виконує перевірку підпису цього сертифіката через внутрішню інфраструктуру центру сертифікації, що використовується у системі, а також перевіряє його дійсність, цілісність та відповідність політикам доступу.

У разі успішного проходження перевірки сервер автоматично дозволяє встановлення з'єднання без додаткових етапів введення облікових даних, що обумовлено використанням режиму автологіну, при якому конфігураційний файл вже містить усі необхідні параметри для ідентифікації клієнта, і така модель забезпечує безперервну роботу сервісів без необхідності втручання людини, що особливо актуально для серверів баз даних, систем резервного копіювання або інших інфраструктурних компонентів, де затримки або необхідність ручного введення пароля є неприпустимими.

Водночас спрощення процесу автентифікації у цьому випадку компенсується обмеженням сфери застосування, оскільки автологін-конфігурації використовуються виключно у контрольованому середовищі, де доступ до самих конфігураційних файлів суворо обмежений, а їх компрометація розглядається як критичний інцидент безпеки, що потребує негайного відкликання сертифіката та перевипуску нових облікових даних.

У сценарії автентифікації кінцевих користувачів, тобто співробітників, реалізовано значно більш складний та багаторівневий алгоритм, який поєднує у собі декілька факторів перевірки, що дозволяє суттєво підвищити рівень захисту доступу до внутрішніх ресурсів, і початкові етапи цього процесу збігаються із

машинним підходом, оскільки також передбачають перевірку сертифіката клієнта у межах TLS-з'єднання, що дозволяє підтвердити автентичність пристрою або конфігураційного файлу.

Після успішної перевірки сертифіката виконується наступний етап, який передбачає введення логіна та пароля користувача, що дозволяє ідентифікувати конкретну особу, яка здійснює підключення, і таким чином навіть у випадку отримання доступу до конфігураційного файлу сторонньою особою неможливо завершити процес автентифікації без знання відповідних облікових даних.

Подальший етап включає перевірку одноразового коду, що генерується у мобільному застосунку автентифікації, наприклад у системах типу Google Authenticator, і цей код формується на основі секретного ключа, який попередньо був згенерований та прив'язаний до облікового запису користувача, при цьому кожен код має обмежений час дії, зазвичай близько 30 секунд, що унеможливлює його повторне використання або перехоплення з подальшим застосуванням.

Послідовне проходження трьох рівнів перевірки, що включають:

— сертифікат;

— пароль;

— одноразовий код.

Все це формує повноцінну багатофакторну модель автентифікації, яка відповідає сучасним вимогам кібербезпеки та значно знижує ризики несанкціонованого доступу навіть у випадку компрометації одного з факторів, і така модель наближена до концепції Zero Trust, у межах якої кожна спроба доступу розглядається як потенційно небезпечна та потребує підтвердження (див. рис. 2.17).

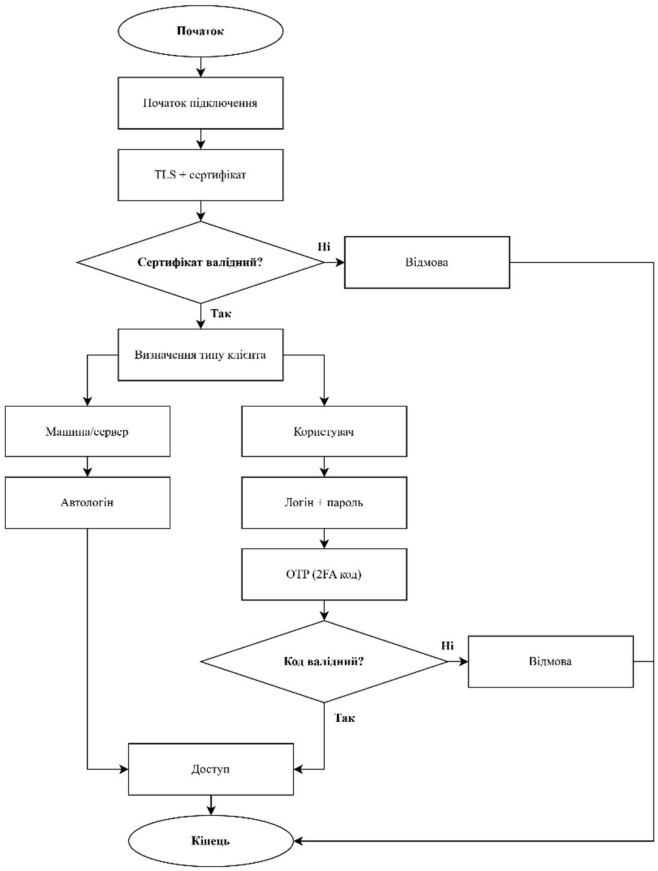


Рисунок 2.17 – Алгоритм автентифікації користувача
на основі сертифікатів, ключів та другого фактора

Алгоритм автоматизованого створення клієнтських конфігурацій за допомогою bash-скриптів. У межах реалізованої системи віддаленого доступу значна увага приділяється автоматизації процесу створення клієнтських конфігурацій, оскільки ручне формування профілів підключення при зростанні кількості користувачів або вузлів інфраструктури призводить до суттєвих витрат часу, підвищує ймовірність помилок та ускладнює централізоване управління доступом, у зв’язку з чим впроваджено алгоритм, що базується на використанні bash-скриптів, які взаємодіють із серверною частиною OpenVPN Access Server через командний інтерфейс керування, забезпечуючи повністю автоматизований цикл створення, налаштування та збереження конфігураційних файлів.

На початковому етапі роботи скрипта відбувається зчитування вхідних даних із підготовленого текстового файлу, що має формат ".txt" і містить перелік

ідентифікаторів користувачів або назв пристроїв, при цьому кожен запис розміщується з нового рядка, що дозволяє обробляти великі списки без додаткової структуризації, а сама логіка побудована таким чином, що вхідні файли можуть бути легко сформовані автоматично з інших інформаційних систем, наприклад кадрових або інвентаризаційних баз.

Після зчитування даних виконується етап нормалізації імен, у межах якого відбувається очищення рядків від зайвих символів, видалення пробілів, заміна їх на підкреслення, а також приведення ідентифікаторів до єдиного формату, що відповідає вимогам системи, і такий підхід дозволяє уникнути помилок під час створення облікових записів та забезпечує уніфікованість імен у межах всієї інфраструктури [25].

На наступному етапі здійснюється створення облікового запису у системі OpenVPN Access Server шляхом виклику відповідних команд утиліти "sacli", яка забезпечує взаємодію із внутрішньою конфігураційною базою сервера, і в процесі виконання цього кроку для кожного запису формується окремий користувач або вузол із відповідними параметрами доступу, що зберігаються у внутрішній базі даних, яка реалізована на основі SQLite.

Подальша логіка передбачає встановлення параметрів доступу, які відрізняються залежно від типу клієнта, і для автоматизованих систем застосовується режим автологіну, що дозволяє виконувати підключення без введення пароля, тоді як для користувачів встановлюється пароль доступу та активується механізм багатофакторної автентифікації, у межах якого генерується унікальний секретний ключ, що використовується для налаштування одноразових кодів у мобільному застосунку автентифікації, і цей ключ надалі передається користувачу разом із іншими обліковими даними.

Після завершення налаштування параметрів виконується запис усіх змін до внутрішньої бази даних сервера, що забезпечує їх збереження та подальше використання під час встановлення з’єднань, а також дозволяє централізовано керувати доступом, змінювати політики безпеки або відкликати сертифікати у разі необхідності.

На наступному етапі алгоритму здійснюється експорт конфігураційного файлу у форматі ".ovpn", який містить усі необхідні параметри для підключення, включаючи сертифікати, криптографічні ключі, адресу сервера, порт та інші мережеві налаштування, і цей файл є основним елементом для встановлення VPN-з’єднання на стороні клієнта.

Подальша обробка результатів залежить від типу створюваного профілю, оскільки для машин або серверів формується лише один конфігураційний файл, який одразу може бути використаний для автоматичного підключення, тоді як для користувачів додатково генерується текстовий файл із обліковими даними, що містить логін, тимчасовий пароль та секретний ключ для налаштування багатофакторної автентифікації, а також коротку інструкцію щодо використання цих даних у клієнтському програмному забезпеченні.

На завершальному етапі для користувацьких профілів обидва файли об’єднуються в архів формату ".zip", що спрощує їх передачу та використання, оскільки вся необхідна інформація знаходиться в одному пакеті, тоді як для машинних профілів зберігається лише окремий файл конфігурації без додаткового пакування, і всі згенеровані файли розміщуються у структурованих каталогах, що відповідають типу клієнтів, забезпечуючи зручність подальшого адміністрування та розповсюдження (див. рис. 2.18) [25].

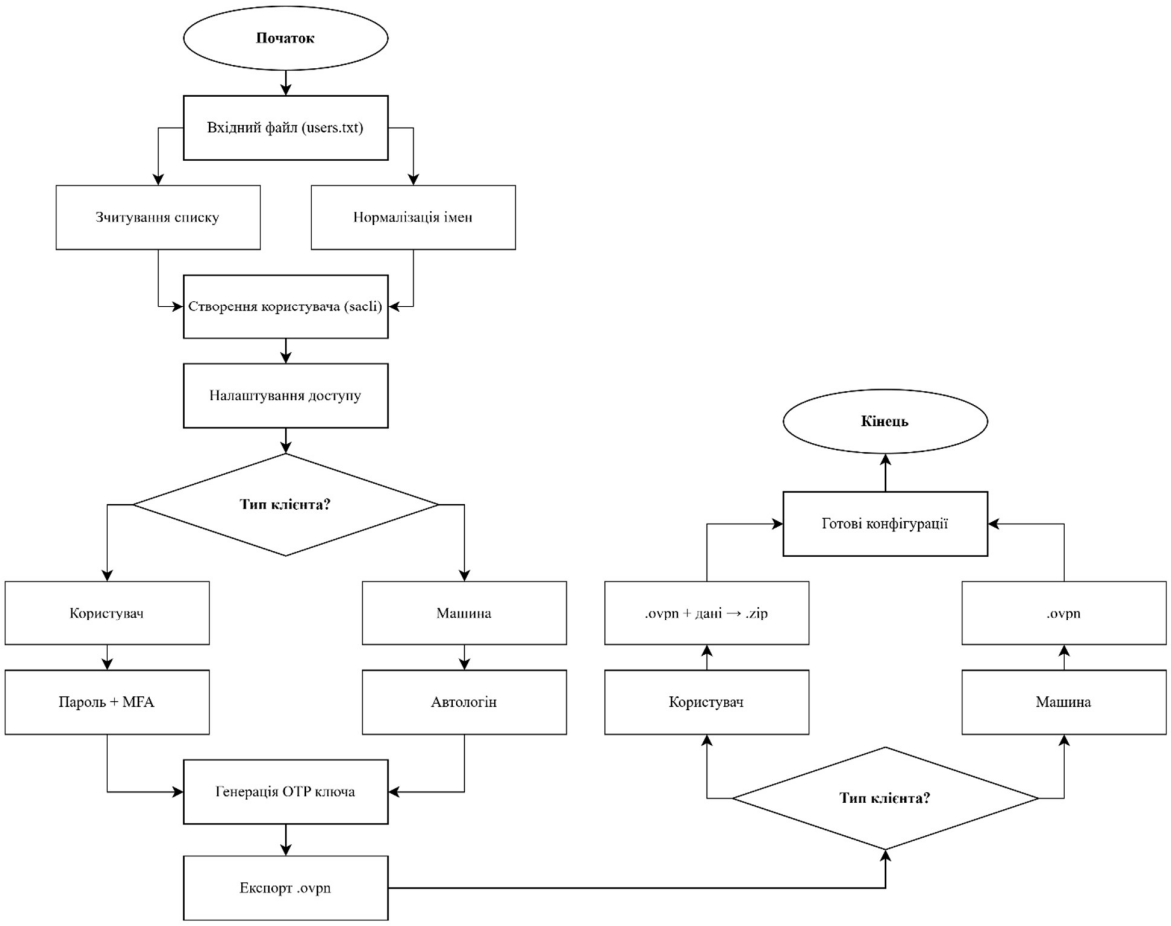


Рисунок 2.18 – Алгоритм автоматизованого створення клієнтських конфігурацій за допомогою bash-скриптів

У результаті реалізації такого алгоритму досягається повна автоматизація процесу створення клієнтських конфігурацій, що дозволяє формувати десятки або навіть сотні профілів без ручного втручання, зменшує навантаження на адміністратора, мінімізує ризик помилок та забезпечує уніфікований підхід до управління доступом у межах всієї VPN-інфраструктури, при цьому система залишається гнучкою та масштабованою, що дозволяє легко адаптувати її до змін у структурі організації або кількості користувачів і пристроїв.

2.4 Реалізація системи на основі існуючих інструментів

Побудова системи віддаленого доступу на основі існуючих інструментів передбачає не лише теоретичне проєктування архітектури, а й її практичну реалізацію з урахуванням реальних умов експлуатації, де інтеграція апаратних та програмних компонентів поєднується з перевіркою працездатності всієї інфраструктури у середовищі, максимально наближеному до реального використання, при цьому розгортання виконується відповідно до концепції bare metal, що означає встановлення серверного програмного забезпечення безпосередньо на фізичне обладнання без використання проміжних віртуалізованих середовищ на рівні сервера, що дозволяє досягти максимальної ефективності використання ресурсів та мінімізації накладних витрат.

Обґрунтування вибору апаратної платформи сервера. Вибір апаратної платформи у такій системі визначає не лише рівень продуктивності, а й стабільність, енергоефективність та здатність до автономної роботи, що особливо актуально в умовах нестабільного електроживлення, тому у якості серверного рішення використано ноутбук бізнес-класу Lenovo ThinkPad T480, який поєднує достатній рівень обчислювальної потужності з високою мобільністю та гнучкістю експлуатації, а також дозволяє реалізувати серверну інфраструктуру без необхідності придбання спеціалізованого серверного обладнання (див. рис. 2.19).



Рисунок 2.19 – Апаратна платформа сервера: Lenovo ThinkPad T480

Особливістю цієї моделі є архітектура живлення на основі технології Power Bridge, що передбачає використання двох акумуляторів — внутрішнього та зовнішнього, причому зовнішній може замінюватися без вимкнення пристрою, що забезпечує можливість гарячої заміни та безперервної роботи системи навіть під час фізичної заміни батареї, а така конфігурація фактично виконує функцію джерела безперебійного живлення, дозволяючи підтримувати роботу сервера протягом тривалого часу без підключення до електромережі, при цьому сумарна ємність акумуляторів забезпечує автономність до десятків годин із можливістю подальшого розширення шляхом використання додаткових змінних батарей (див. рис. 2.20).

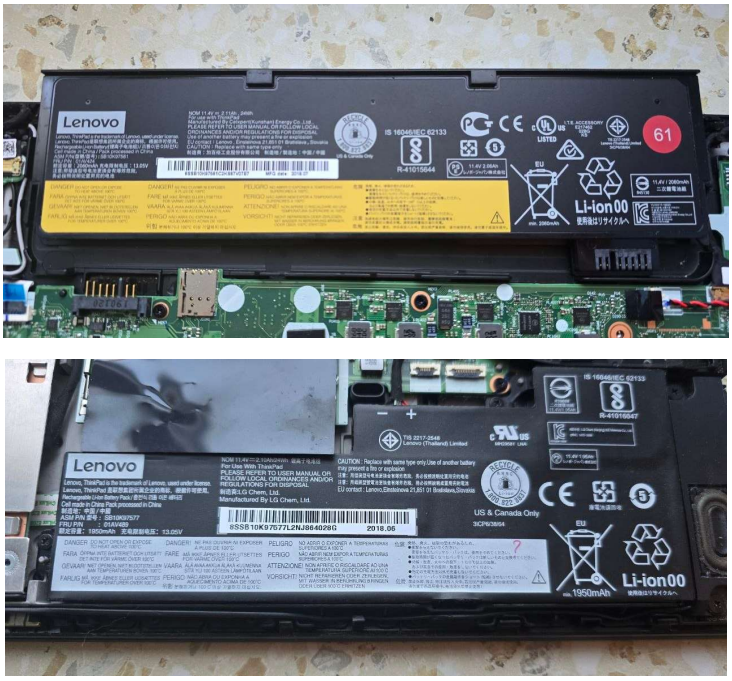


Рисунок 2.20 – Внутрішня незмінна та зовнішня how-swap батарея

З точки зору обчислювальних ресурсів використаний процесор Intel Core i5-8350U забезпечує достатню продуктивність для виконання задач, пов’язаних із обробкою VPN-трафіку, включаючи криптографічні операції шифрування та дешифрування, що є критично важливими для функціонування захищених каналів зв’язку, а підтримка багатопотоковості дозволяє ефективно обробляти декілька одночасних підключень без суттєвого зниження продуктивності системи.

Додатково враховано аспекти безпеки та масштабованості, оскільки наявність модуля TPM 2.0 та підтримка сучасного BIOS/UEFI середовища дозволяє підвищити рівень захисту системи, тоді як можливість розширення оперативної пам’яті створює передумови для подальшого масштабування інфраструктури у разі зростання навантаження, що робить обрану платформу придатною не лише для тестового використання, а й для розгортання у більш складних сценаріях.

Важливим аспектом є також енергоефективність та альтернативні способи живлення, оскільки використання інтерфейсу USB Type-C дозволяє підключати зовнішні джерела живлення, зокрема павербанки, що значно підвищує автономність системи та дозволяє підтримувати її роботу навіть у польових умовах або при повній відсутності стаціонарного електроживлення, а поєднання внутрішнього акумулятора, змінної батареї та зовнішніх джерел живлення формує багаторівневу систему енергозабезпечення.

Мобільність обраної платформи також має суттєве значення, оскільки використання безпровідного мережевого інтерфейсу дозволяє швидко змінювати місце розташування сервера без необхідності прокладання кабельної інфраструктури, що спрощує розгортання системи у різних середовищах та дозволяє адаптувати її до змінних умов експлуатації, а інтегрований дисплей, клавіатура та інші компоненти ноутбука забезпечують можливість локального адміністрування без необхідності підключення додаткових периферійних пристроїв.

Окремої уваги заслуговує ремонтопридатність та доступність компонентів, оскільки серія ThinkPad традиційно відзначається уніфікованістю деталей, що спрощує обслуговування та дозволяє швидко замінювати несправні елементи без значних витрат, а наявність великої кількості сумісних запчастин на ринку забезпечує додаткову надійність експлуатації у довгостроковій перспективі (див. табл. 2.2).

Таблиця 2.2 – Технічні характеристики серверної платформи (Lenovo ThinkPad T480)

Компонент	Характеристика	Значення
Процесор	Модель	Intel Core i5-8350U
Оперативна пам’ять	Тип / обсяг	DDR4 / 16 ГБ
Накопичувач	Тип / обсяг	NVMe SSD / 256 ГБ
Мережеві інтерфейси	Тип	Ethernet, Wi-Fi
Безпека	Модуль	TPM 2.0
Живлення	Технологія	Power Bridge (2 акумулятори)
Автономність	Час роботи	до 10–12 годин
Розширення	Можливості	RAM, акумулятори

У сукупності всі ці фактори формують ефективну, економічно доцільну та гнучку платформу для реалізації системи віддаленого доступу, яка поєднує у собі функціональність повноцінного VPN-сервера з високим рівнем автономності, мобільності та адаптивності до різних умов використання, що дозволяє розглядати таке рішення як практичну альтернативу традиційним серверним інфраструктурам у випадках, де критичними є швидкість розгортання, енергонезалежність та мінімізація витрат.

Практична реалізація VPN-сервера. Практична реалізація серверної частини системи віддаленого доступу здійснюється відповідно до концепції "bare metal", що передбачає встановлення операційної системи безпосередньо на фізичне обладнання без використання віртуалізації, завдяки чому досягається мінімізація накладних витрат ресурсів і забезпечується максимально можливий рівень продуктивності, при цьому обрана апаратна платформа у вигляді ноутбука Lenovo ThinkPad T480 використовується як повноцінний серверний вузол, на який інсталюється операційна система Ubuntu Server 24.04.3 LTS, що характеризується стабільністю, наявністю довгострокової підтримки та оптимізованим використанням системних ресурсів, а відсутність графічного інтерфейсу додатково зменшує навантаження на процесор і оперативну пам’ять, спрямовуючи доступні ресурси виключно на виконання серверних задач.

Після завершення інсталяції операційної системи виконується базова мережева конфігурація, яка включає призначення статичної IP-адреси,

налаштування маршрутизації та забезпечення доступу до мережі Інтернет через безпровідний інтерфейс Wi-Fi, що відповідає вимогам мобільності обраної платформи та дозволяє використовувати сервер у різних мережевих середовищах без прив’язки до кабельної інфраструктури, а стабільність мережевого підключення забезпечує коректну взаємодію із зовнішніми клієнтами та іншими мережевими вузлами.

Наступним етапом виконується розгортання програмного компонента VPN, у ролі якого використовується OpenVPN Access Server, що забезпечує створення захищених тунелів, керування підключеннями клієнтів та маршрутизацію мережевого трафіку, при цьому після встановлення сервіс інтегрується у систему керування службами операційної системи, що дозволяє контролювати його стан за допомогою стандартних інструментів адміністрування, а перевірка працездатності здійснюється шляхом виконання команди перевірки статусу служби, результати якої підтверджують коректний запуск та функціонування сервера у штатному режимі.

У процесі роботи сервер використовує виділений внутрішній адресний простір "172.16.0.0/26" для призначення IP-адрес клієнтам, що підключаються, завдяки чому формується ізольоване віртуальне мережеве середовище, у межах якого здійснюється обмін даними, а механізми маршрутизації забезпечують передачу трафіку між клієнтами VPN та ресурсами локальної мережі, до якої підключений сервер, що дозволяє реалізувати доступ до внутрішніх сервісів незалежно від фізичного місця розташування користувача (див. рис. 2.21).



Рисунок 2.21 – Робота VPN-сервера у терміналі

Налаштування клієнтської сторони. Налаштування клієнтської сторони системи віддаленого доступу здійснюється у межах ізольованого середовища, що моделює реальні умови використання та дозволяє відтворити типові сценарії підключення користувачів, при цьому для досягнення такого ефекту застосовується середовище віртуалізації VirtualBox, у якому розгортаються клієнтські операційні системи, зокрема:

1. Одна віртуальна машина на базі Windows 11.
2. Інша на базі Ubuntu Desktop.

Все це дозволяє перевірити роботу системи у кросплатформеному контексті та оцінити сумісність програмного забезпечення з різними середовищами.

Ізоляція клієнтів від основної локальної мережі досягається шляхом підключення їх не до внутрішньої інфраструктури, а до альтернативного джерела доступу до Інтернету, зокрема через мобільну точку доступу, що створюється за допомогою смартфона, завдяки чому виключається можливість прямого доступу до серверного вузла та ресурсів локальної мережі без використання VPN-з’єднання, а перевірка цього обмеження підтверджується відсутністю відповіді при спробі звернення до локальної IP-адреси сервера, що свідчить про коректну ізоляцію мережевого середовища.

Додатковим підтвердженням використання зовнішньої мережі виступає аналіз мережевих параметрів клієнтських систем, де призначені IP-адреси не належать ані до підмережі локального маршрутизатора, ані до внутрішнього VPN-діапазону, що вказує на використання мобільного інтернет-з’єднання, а перевірка публічної IP-адреси через відповідні онлайн-сервіси демонструє належність до оператора мобільного зв’язку, що остаточно підтверджує відсутність прямого доступу до серверної інфраструктури (див. рис. 2.22).

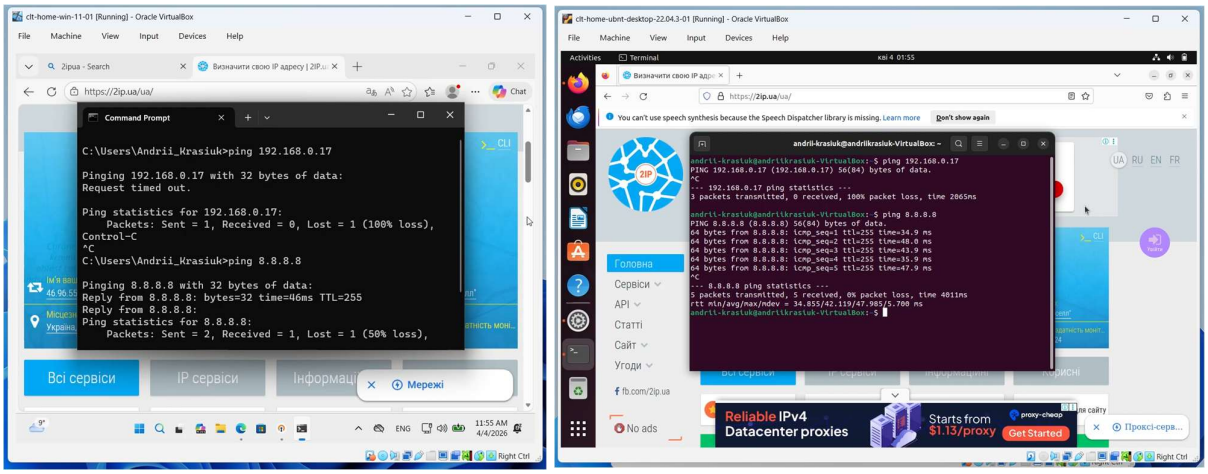


Рисунок 2.22 – Клієнтська сторона без доступу до внутрішніх ресурсів

Після формування ізольованого середовища виконується встановлення клієнтського програмного забезпечення OpenVPN Connect на кожній із віртуальних машин, що забезпечує можливість створення захищеного з'єднання з VPN-сервером, а для підключення використовуються індивідуальні конфігураційні файли формату ".ovpn", попередньо згенеровані на сервері, які містять усі необхідні параметри, включаючи адресу сервера, криптографічні ключі та сертифікати (див. рис. 2.23).

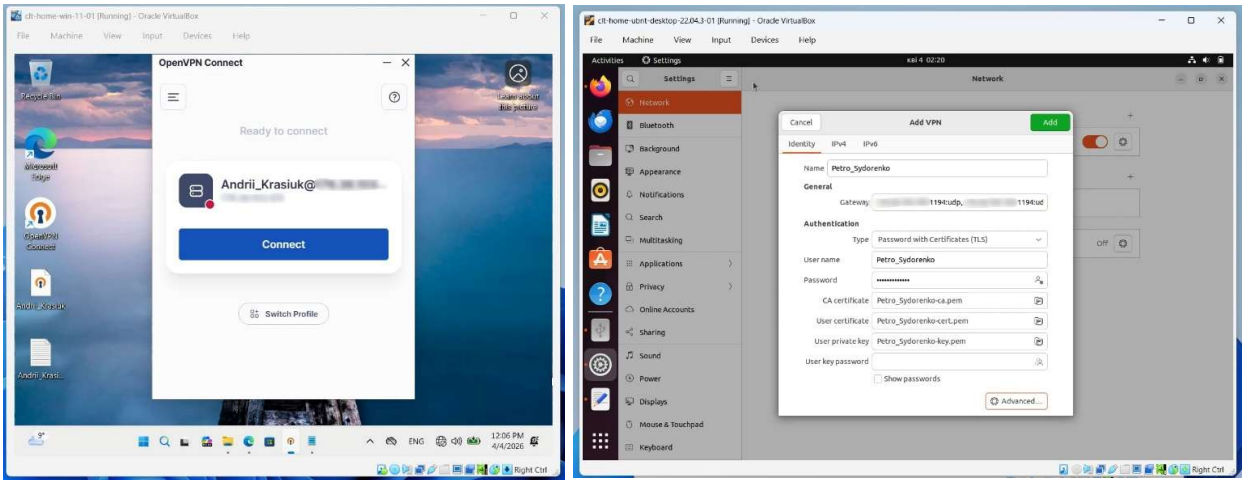


Рисунок 2.23 – Процес конфігурації клієнтів
шляхом імпорту конфігураційних файлів

Для практичної перевірки застосовуються сформовані архіви доступу для окремих користувачів, у складі яких знаходяться як конфігураційні файли, так і супровідні інструкції з даними для автентифікації, включаючи логін, тимчасовий пароль та секретний ключ для налаштування багатофакторної автентифікації, що реалізується за допомогою мобільного застосунку типу Google Authenticator, при цьому смартфон використовується не як окремий клієнт підключення, а як засіб генерації одноразових кодів, необхідних для завершення процедури входу (див. рис. 2.24).

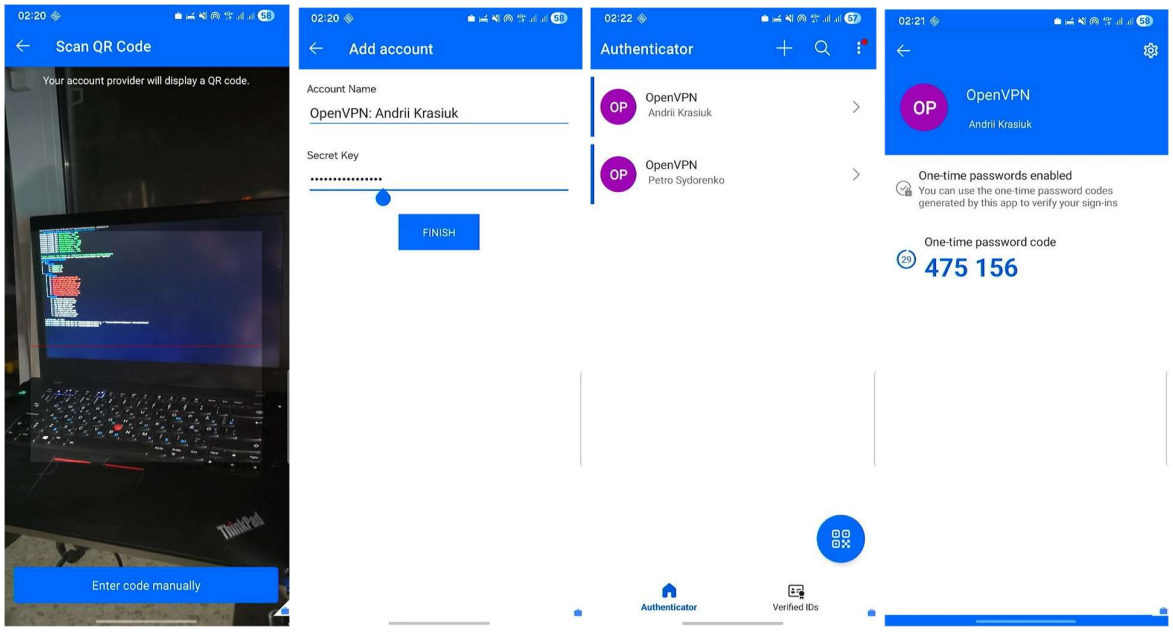


Рисунок 2.24 – Додавання другого фактору через згенерований код MFA

Під час підключення у клієнтському застосунку імпортується відповідний конфігураційний файл, після чого виконується автентифікація шляхом введення облікових даних та одноразового шестизначного коду, який генерується у мобільному застосунку та має обмежений час дії, а після успішного проходження цієї процедури встановлюється захищене VPN-з’єднання, у межах якого клієнт отримує внутрішню IP-адресу з виділеного діапазону та стає повноцінним учасником віртуальної мережі (див. рис. 2.25).

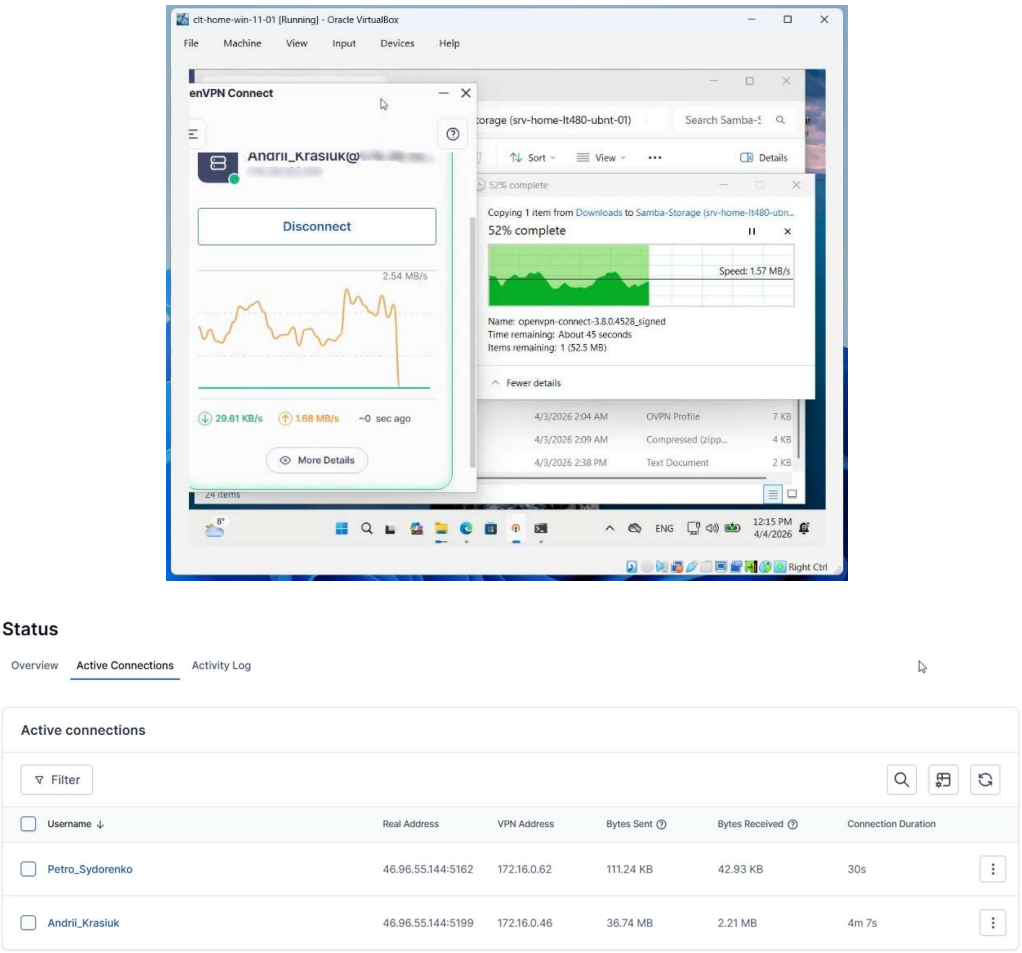


Рисунок 2.25 – Клієнтська сторона зі доступом до внутрішніх ресурсів

Такий підхід забезпечує повну ізоляцію клієнтів до моменту встановлення захищеного з’єднання, відтворює реальні умови роботи віддалених користувачів, дозволяє перевірити функціонування системи у різних операційних середовищах та підтверджує коректність реалізації механізмів автентифікації і тунелювання, а також створює основу для подальшої демонстрації доступу до внутрішніх ресурсів після успішного підключення.

Демонстрація роботи системи. Для перевірки функціональності системи віддаленого доступу реалізовано тестовий сценарій, що імітує типовий доступ користувача до внутрішніх ресурсів організації, серед яких одним із найбільш поширених є файловий сервер, тому у якості такого ресурсу на сервері розгортається файловий сервіс на базі Samba, що дозволяє організувати спільний доступ до директорій у локальній мережі, при цьому попередньо створюється

каталог за шляхом "/opt/samba/shared", якому призначаються відповідні права доступу, після чого у конфігураційному файлі Samba додається опис ресурсу з визначенням його імені, параметрів доступу та обмеженням на використання лише для певної групи користувачів.

Додатково формується окрема група користувачів та створюється спеціалізований обліковий запис "vpnuser", який використовується виключно для доступу до файлового ресурсу, причому важливо враховується, що Samba використовує власну систему автентифікації, тому пароль для доступу до мережевого ресурсу встановлюється окремо від системного облікового запису, а після внесення відповідних змін служба перезапускається для застосування конфігурації, у результаті чого формується ізольований файловий ресурс із контрольованим доступом (див. рис. 2.26).

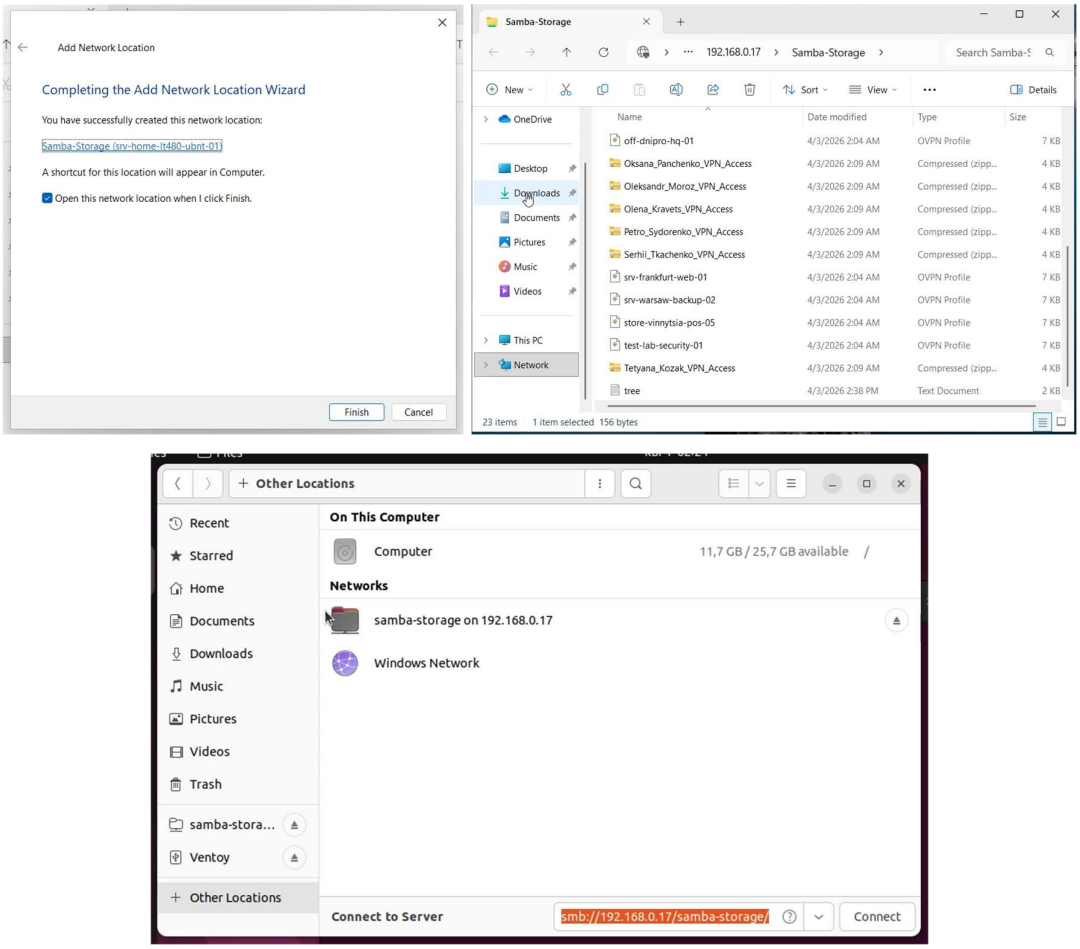


Рисунок 2.26 – Файловий сервер на базі Samba як внутрішній ресурс

Побудована таким чином схема доступу дозволяє реалізувати багаторівневу модель захисту, у межах якої спочатку виконується автентифікація на рівні VPN із використанням сертифікатів, пароля та багатофакторної автентифікації, після чого здійснюється додаткова авторизація безпосередньо у файловій системі, що забезпечує розмежування доступу навіть серед користувачів, які вже підключені до захищеної мережі, а така структура підвищує загальний рівень безпеки та дозволяє гнучко керувати правами доступу.

У процесі демонстрації роботи системи спочатку фіксується відсутність доступу до внутрішніх ресурсів без встановленого VPN-з'єднання, що підтверджується неможливістю звернення до локальної IP-адреси сервера, після чого виконується підключення через клієнт OpenVPN Connect, у результаті якого клієнт отримує внутрішню IP-адресу з VPN-пулу та інтегрується у віртуальну мережу, а перевірка з'єднання здійснюється шляхом відправлення мережевих запитів до сервера, що підтверджує доступність внутрішньої інфраструктури.

Після встановлення з'єднання виконується спроба доступу до файлового ресурсу за мережевим шляхом виду "\\192.168.0.17\Samba-Storage", у відповідь на що система запитує облікові дані користувача, які вводяться відповідно до попередньо створеного облікового запису, після чого відкривається доступ до спільної директорії, де стає можливим виконання операцій із файлами, включаючи їх створення, копіювання та редагування, а успішне виконання цих дій підтверджує коректність роботи захищеного тунелю та правильність налаштування як серверної, так і клієнтської частини системи (див. рис. 2.27).

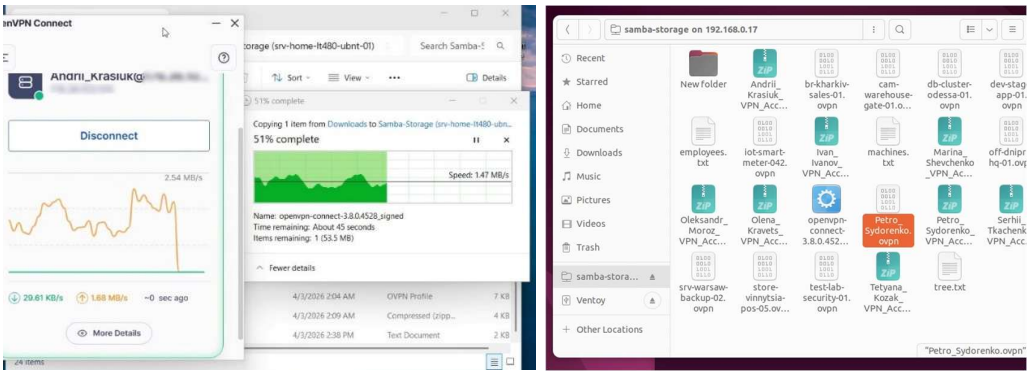


Рисунок 2.27 – Робота зі файловим сервером через VPN

Реалізований сценарій демонструє повноцінний цикл доступу до внутрішніх ресурсів через захищене VPN-з’єднання з урахуванням багаторівневої автентифікації та авторизації, що дозволяє не лише перевірити працездатність системи, а й наочно підтвердити ефективність застосованих механізмів захисту та можливість їх використання у реальних умовах, де доступ до ресурсів може бути обмежений відповідно до ролей користувачів та політик безпеки.

Аналіз продуктивності та стабільності системи. Загальна оцінка роботи реалізованої системи здійснювалася шляхом аналізу швидкості передачі даних та рівня навантаження на апаратні ресурси сервера на базі Lenovo ThinkPad T480, при цьому основним критерієм виступала саме пропускна здатність VPN-з’єднання, оскільки через нього проходить увесь користувацький трафік і саме воно визначає практичну ефективність використання системи в реальних умовах експлуатації, зокрема під час копіювання файлів, доступу до мережевих ресурсів та роботи з віддаленими сервісами, і хоча OpenVPN не орієнтований на досягнення максимально можливих швидкостей передачі даних, а більше спрямований на забезпечення захищеного доступу, отримані результати показали, що навіть при використанні шифрування рівня AES-256-GCM забезпечується достатня пропускна здатність для типових задач, таких як робота з документами, передача архівів та взаємодія з файловими ресурсами [26].

При цьому суттєвий вплив на швидкість передачі має тип мережевого підключення, оскільки сервер функціонує через Wi-Fi інтерфейс у діапазоні 5 ГГц, що забезпечує теоретичну швидкість до приблизно 800 Мбіт/с, однак реальна продуктивність залежить від якості сигналу, рівня перешкод та загального навантаження мережі, додатково обмеження накладаються і з боку клієнтського підключення, особливо у випадку використання мобільного інтернету, де під час тестування швидкість становила близько 20–27 Мбіт/с на завантаження та приблизно 8–9 Мбіт/с на відвантаження, що фактично формує верхню межу пропускної здатності для VPN-тунелю, і в реальних умовах під час передачі даних спостерігалася стабільна швидкість у межах приблизно 1.3–2.5 Мбіт/с, з піковими

значеннями до 3.5 Мбіт/с, що узгоджується з обмеженнями каналу та накладними витратами на шифрування (див. рис. 2.28).

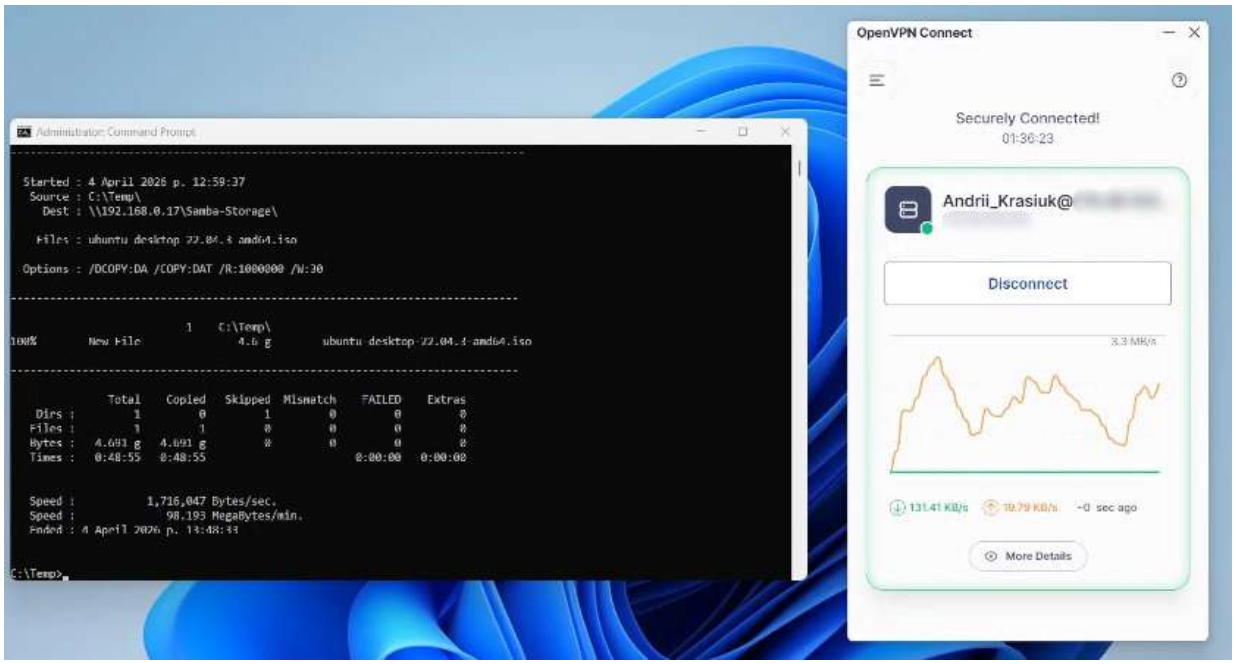


Рисунок 2.28 – Ручне тестування швидкості передачі

Аналіз навантаження на процесор виконувався за допомогою стандартної утиліти "htop" в середовищі Ubuntu Server, де було зафіксовано, що під час активної передачі даних, зокрема при копіюванні великого файлу розміром близько 4.6 ГБ, завантаження одного з ядер процесора Intel Core i5-8350U не перевищувало 10–12%, що свідчить про достатній запас обчислювальної потужності для обробки криптографічних операцій шифрування та дешифрування, а також дозволяє зробити висновок про можливість обслуговування більшої кількості клієнтів без суттєвого впливу на продуктивність системи (див. рис. 2.29).

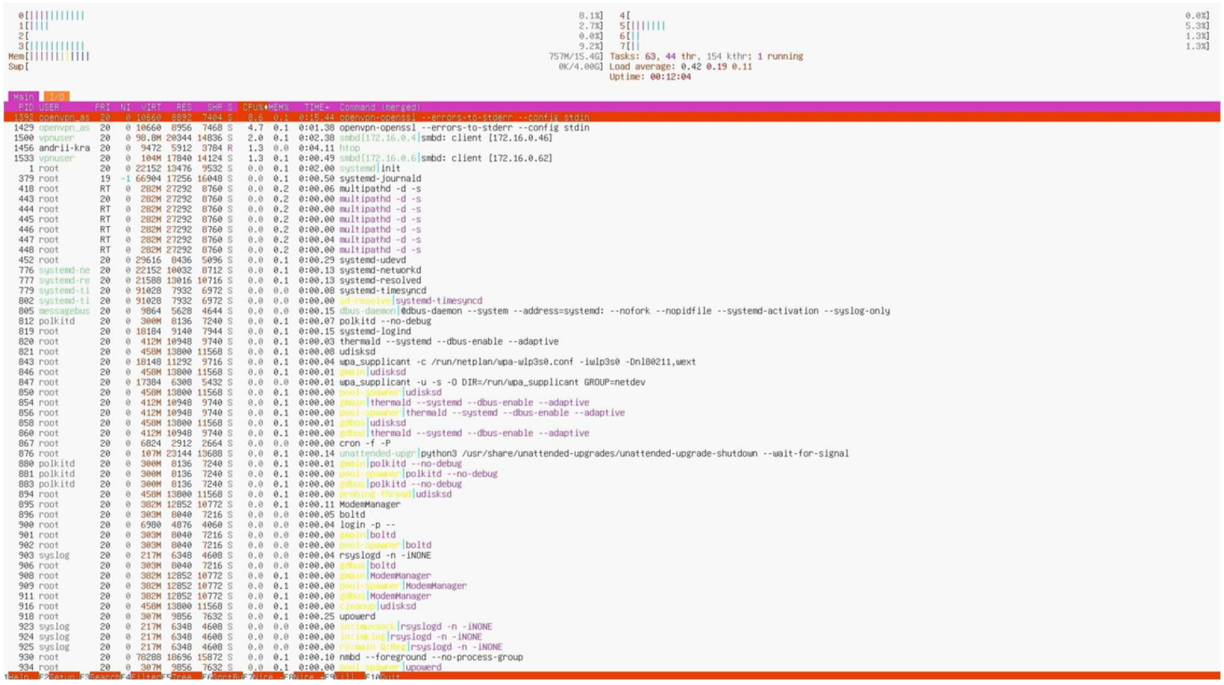
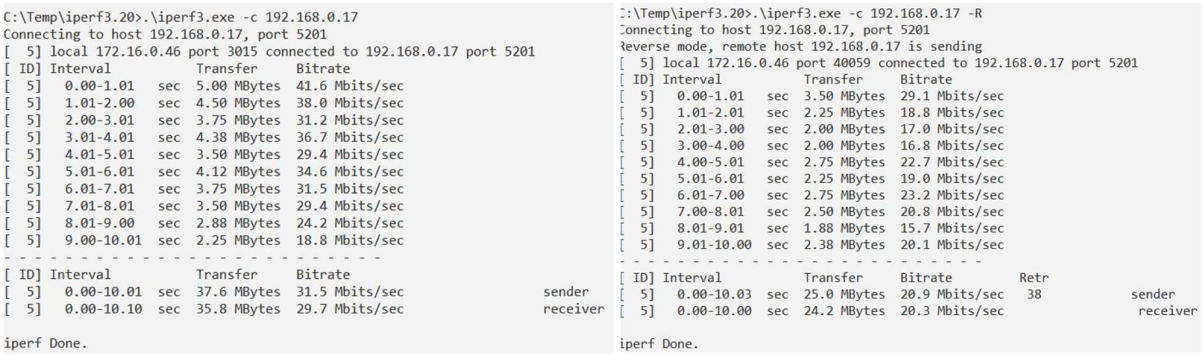


Рисунок 2.29 – Моніторинг навантаження системи зі двома активними клієнтами

Практичний експеримент із копіюванням дистрибутива операційної системи Ubuntu Desktop продемонстрував, що передача файлу обсягом 4.6 ГБ через VPN-тунель із використанням мобільного інтернету займає приблизно до однієї години, що відповідає очікуваним показникам для даного типу з’єднання і підтверджує придатність системи для виконання типових задач віддаленого доступу, хоча для сценаріїв із високими вимогами до пропускної здатності, таких як потокове відео або передача великих обсягів даних у реальному часі, таке рішення не є оптимальним.

Додатково було проведено тестування пропускної здатності за допомогою утиліти "iperf3", що дозволило оцінити швидкість передачі даних у двох напрямках між клієнтом та сервером, при цьому результати показали середню швидкість на рівні приблизно 29–31 Мбіт/с у напрямку від клієнта до сервера та близько 20–21 Мбіт/с у зворотному напрямку, що підтверджує стабільність каналу та узгоджується з характеристиками мережевого середовища, а також демонструє відсутність критичних втрат пакетів або значних коливань пропускної здатності (див. рис. 2.30).



```
C:\Temp\iperf3.20>.\iperf3.exe -c 192.168.0.17
Connecting to host 192.168.0.17, port 5201
[  5] local 172.16.0.46 port 3015 connected to 192.168.0.17 port 5201
[ ID] Interval           Transfer     Bitrate
[  5]   0.00-1.01   sec  5.00 MBytes  41.6 Mbits/sec
[  5]   1.01-2.00   sec  4.50 MBytes  38.0 Mbits/sec
[  5]   2.00-3.01   sec  3.75 MBytes  31.2 Mbits/sec
[  5]   3.01-4.01   sec  4.38 MBytes  36.7 Mbits/sec
[  5]   4.01-5.01   sec  3.50 MBytes  29.4 Mbits/sec
[  5]   5.01-6.01   sec  4.12 MBytes  34.6 Mbits/sec
[  5]   6.01-7.01   sec  3.75 MBytes  31.5 Mbits/sec
[  5]   7.01-8.01   sec  3.50 MBytes  29.4 Mbits/sec
[  5]   8.01-9.00   sec  2.88 MBytes  24.2 Mbits/sec
[  5]   9.00-10.01  sec  2.25 MBytes  18.8 Mbits/sec
- - - - - - - - - - - - - - - - - - - - - - - - -
[ ID] Interval           Transfer     Bitrate
[  5]   0.00-10.01  sec  37.6 MBytes  31.5 Mbits/sec                  sender
[  5]   0.00-10.10  sec  35.8 MBytes  29.7 Mbits/sec                  receiver

iperf Done.
```

```
C:\Temp\iperf3.20>.\iperf3.exe -c 192.168.0.17 -R
Connecting to host 192.168.0.17, port 5201
Reverse mode, remote host 192.168.0.17 is sending
[  5] local 172.16.0.46 port 40059 connected to 192.168.0.17 port 5201
[ ID] Interval           Transfer     Bitrate
[  5]   0.00-1.01   sec  3.50 MBytes  29.1 Mbits/sec
[  5]   1.01-2.01   sec  2.25 MBytes  18.8 Mbits/sec
[  5]   2.01-3.00   sec  2.00 MBytes  17.0 Mbits/sec
[  5]   3.00-4.00   sec  2.00 MBytes  16.8 Mbits/sec
[  5]   4.00-5.01   sec  2.75 MBytes  22.7 Mbits/sec
[  5]   5.01-6.01   sec  2.25 MBytes  19.0 Mbits/sec
[  5]   6.01-7.00   sec  2.75 MBytes  23.2 Mbits/sec
[  5]   7.00-8.01   sec  2.50 MBytes  20.8 Mbits/sec
[  5]   8.01-9.01   sec  1.88 MBytes  15.7 Mbits/sec
[  5]   9.01-10.00  sec  2.38 MBytes  20.1 Mbits/sec
- - - - - - - - - - - - - - - - - - - - - - - - -
[ ID] Interval           Transfer     Bitrate         Retr
[  5]   0.00-10.03  sec  25.0 MBytes  20.9 Mbits/sec   38            sender
[  5]   0.00-10.00  sec  24.2 MBytes  20.3 Mbits/sec                 receiver

iperf Done.
```

Рисунок 2.30 – Тестування швидкості передачі через iperf3

Оцінка стабільності системи здійснювалася шляхом тривалого тестування з’єднання під навантаженням, зокрема під час безперервного копіювання файлів протягом більше ніж однієї години, у ході якого не було зафіксовано розривів з’єднання або критичних збоїв у роботі VPN-сервера, що свідчить про надійність реалізованого рішення та коректність налаштування як серверної, так і клієнтської частини системи, а також підтверджує можливість стабільної роботи навіть при використанні безпровідного підключення.

Отримані результати дозволяють зробити висновок, що навіть використання відносно застарілого апаратного забезпечення за умови правильної конфігурації програмного середовища та мережевих параметрів дає змогу реалізувати ефективну, стабільну та функціональну систему віддаленого доступу, яка повністю відповідає вимогам до безпеки та забезпечує достатній рівень продуктивності для більшості практичних задач, пов’язаних із доступом до внутрішніх ресурсів організації.

3. АНАЛІЗ ЕФЕКТИВНОСТІ ТА ПЕРСПЕКТИВИ РОЗВИТКУ СИСТЕМИ

3.1 Методи оптимізації безпеки та продуктивності системи

На основі отриманих результатів функціонування системи формується можливість проведення додаткового аналізу, який дозволяє не лише узагальнити ефективність запропонованого рішення, але й виявити його потенційні слабкі місця та визначити напрями подальшої оптимізації з урахуванням реальних умов експлуатації, включаючи обмежені апаратні ресурси серверної платформи, використання застарілого обладнання у вигляді Lenovo ThinkPad T480, а також нестабільність мережевого середовища, що проявляється при роботі через Wi-Fi та мобільні канали зв’язку, де спостерігаються коливання затримок, пропускної здатності та загальної якості з’єднання, унаслідок чого особлива увага приділяється практичним аспектам забезпечення стабільності, безпеки та ефективності функціонування системи в умовах змінного навантаження та непередбачуваних мережевих факторів.

Оптимізація навантаження на апаратну платформу сервера. Під час проведеного тестування встановлено, що одним із основних обмежуючих факторів продуктивності системи виступає центральний процесор серверної платформи, незважаючи на те, що у процесі вимірювання навантаження за участю одного клієнта, який виконує передачу великого обсягу даних, рівень використання процесора знаходився в межах приблизно 10–12%, така ситуація не відображає повної картини при масштабуванні системи, оскільки зі збільшенням кількості одночасних підключень та інтенсивності обміну даними навантаження зростатиме пропорційно, що особливо актуально в умовах обмеження ліцензії на кількість клієнтів, яка не дозволила провести більш глибоке експериментальне дослідження, однак навіть за наявними результатами можна зробити висновок про потенційне зростання навантаження при реальному використанні.

Суттєвий внесок у використання обчислювальних ресурсів здійснюють криптографічні операції, які виконуються під час шифрування та дешифрування трафіку у VPN-тунелі, зокрема застосування алгоритму AES-256-GCM забезпечує високий рівень захисту переданих даних, однак одночасно створює додаткове навантаження на процесор Intel Core i5-8350U, що використовується у складі серверної платформи, і така залежність між рівнем криптографічного захисту та обчислювальними витратами безпосередньо впливає на загальну пропускну здатність системи та її здатність обслуговувати більшу кількість клієнтів.

З метою оптимізації використання апаратних ресурсів доцільним є перехід до менш ресурсоємного алгоритму шифрування, зокрема AES-128-GCM, який забезпечує достатній рівень криптографічного захисту для більшості корпоративних сценаріїв використання, при цьому значно зменшує навантаження на процесор та підвищує ефективність обробки трафіку, а практичний досвід застосування подібних рішень демонструє, що різниця у рівні безпеки між AES-128 та AES-256 у реальних умовах експлуатації є незначною для більшості задач, тоді як приріст продуктивності може бути відчутним, що дозволяє досягти більш збалансованого співвідношення між безпекою та швидкодією системи (див. рис. 3.1) [27].

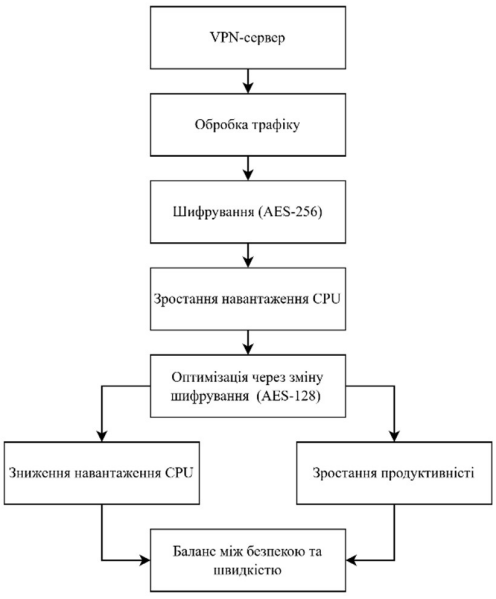


Рисунок 3.1 – Оптимізація навантаження на апаратну платформу сервера

Оптимізація передачі даних у межах VPN-з'єднання. Аналіз результатів тестування пропускної здатності, зокрема із застосуванням утиліти "iperf3", дозволив оцінити ефективність передачі даних через VPN-тунель при різних умовах функціонування системи, при цьому було встановлено, що середня швидкість передачі від клієнта до сервера становить приблизно 30 Мбіт/с, тоді як у зворотному напрямку від сервера до клієнта — близько 20 Мбіт/с, що загалом відповідає характеристикам мережевого середовища та підтверджує стабільність каналу зв'язку, однак одним із визначальних факторів, що безпосередньо впливає на ефективність передачі даних, виступає вибір транспортного протоколу, оскільки OpenVPN підтримує роботу як через TCP, так і через UDP, і проведений аналіз показав, що використання TCP у межах VPN-тунелю може призводити до суттєвого зниження продуктивності через явище, відоме як TCP over TCP meltdown, яке виникає у випадку інкапсуляції одного TCP-з'єднання в інше, що спричиняє надмірну кількість повторних передач пакетів, збільшення затримок та загальну деградацію швидкості передачі даних (див. рис. 3.2) [28].

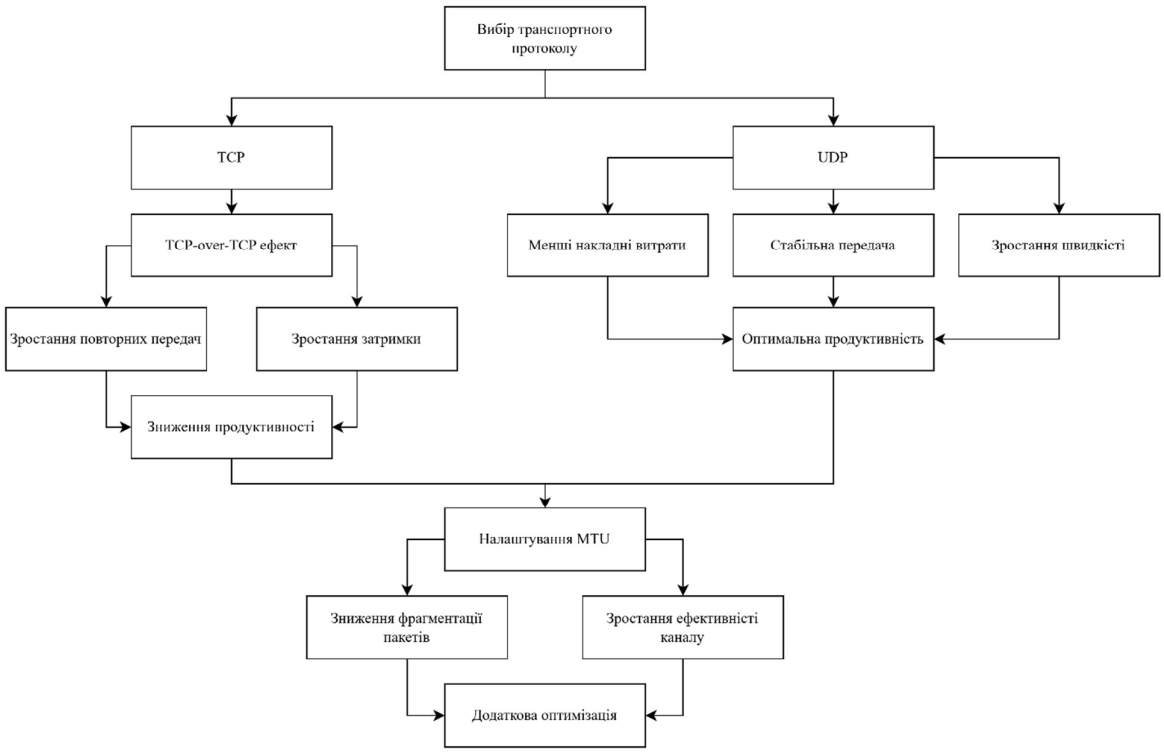


Рисунок 3.2 – Оптимізація передачі даних у межах VPN-з'єднання

На відміну від цього, використання UDP дозволяє уникнути подвійного механізму контролю доставки пакетів, зменшує накладні витрати та забезпечує більш ефективне використання пропускної здатності каналу, що особливо важливо в умовах нестабільних мереж, таких як безпровідні або мобільні з'єднання, унаслідок чого саме UDP доцільно розглядати як оптимальний варіант для реалізації VPN-з'єднань у більшості практичних сценаріїв, а додаткове налаштування параметрів передачі, зокрема значення MTU, дозволяє зменшити фрагментацію пакетів і підвищити ефективність використання каналу, що має особливе значення при роботі через мережі з підвищеним рівнем затримок та варіативною якістю зв'язку [28].

Оптимізація процесу надання доступу користувачам. У процесі реалізації системи було впроваджено механізм автоматизованого створення клієнтських конфігурацій із використанням bash-скриптів, що суттєво вплинуло на ефективність адміністрування та масштабування інфраструктури, і порівняльний аналіз показує, що ручне створення конфігурацій є доволі трудомістким процесом, оскільки для групи з приблизно десяти користувачів такий підхід може займати від десяти до п'ятнадцяти хвилин з урахуванням необхідності введення параметрів, генерації сертифікатів та перевірки коректності налаштувань, тоді як автоматизований підхід дозволяє виконати аналогічний обсяг операцій протягом кількох секунд за рахунок використання попередньо підготовлених сценаріїв.

Скрипти здійснюють зчитування списку користувачів із текстового файлу, після чого автоматично створюють облікові записи, застосовують відповідні політики безпеки та експортують готові конфігураційні файли, що забезпечує не лише значне скорочення часу розгортання, але й мінімізацію людських помилок, які можуть виникати при ручному налаштуванні, а також стандартизацію всього процесу створення доступів, що у свою чергу спрощує подальше адміністрування системи та дозволяє швидко масштабувати її у випадку збільшення кількості користувачів, і така автоматизація розглядається не лише як інструмент підвищення ефективності, але і як важливий елемент забезпечення безпеки,

оскільки забезпечує уніфікований підхід до налаштування параметрів доступу та зменшує ймовірність виникнення конфігураційних помилок (див. рис. 3.3).

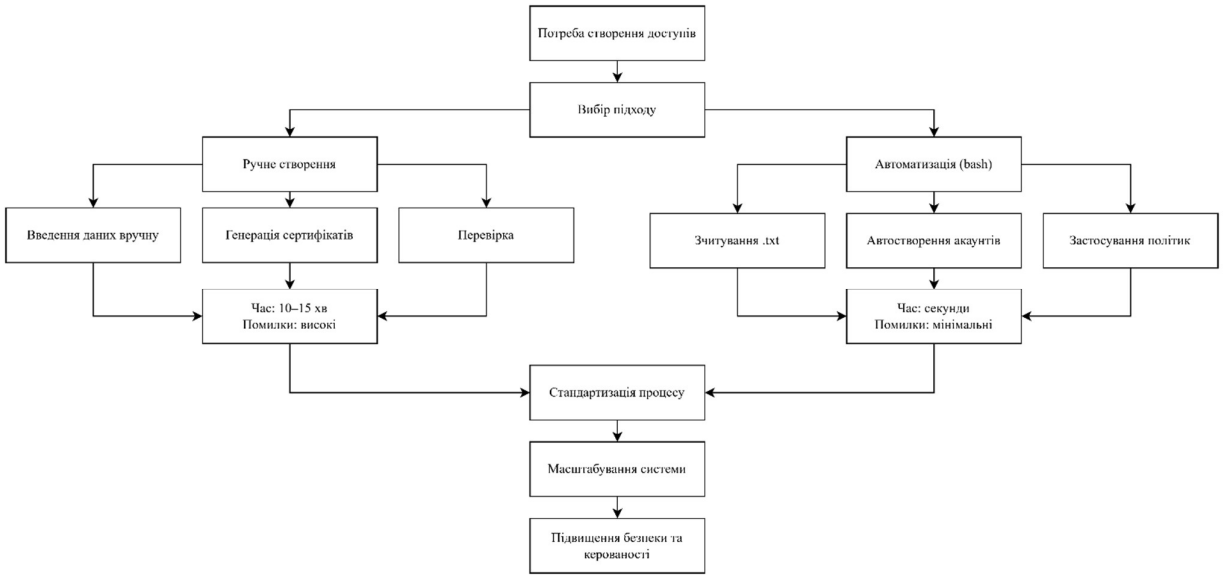


Рисунок 3.3 – Оптимізація процесу надання доступу користувачам

3.2 Порівняння запропонованого рішення з існуючими підходами

У межах подальшого аналізу доцільно розглянути розроблену систему віддаленого доступу на базі VPN у порівнянні з альтернативними підходами, що застосовуються в сучасних корпоративних мережах, що дозволяє сформувати комплексне уявлення про її переваги та обмеження з урахуванням реальних умов експлуатації, включаючи:

— обмежені апаратні ресурси;

— нестабільне електроживлення;

— використання безпровідних каналів зв’язку.

При цьому основна увага спрямовується на визначення ефективності рішення, його адаптивності та доцільності застосування у практичних сценаріях.

Порівняння характеристик VPN-протоколів. У межах проведеного аналізу розглянуто основні VPN-протоколи, що активно використовуються у сучасних мережевих інфраструктурах, серед яких виділяються OpenVPN, WireGuard та IPsec, що дозволяє оцінити їх функціональні можливості, рівень безпеки, продуктивність та придатність до використання в різних сценаріях, включаючи корпоративні середовища з підвищеними вимогами до контролю доступу та захисту даних (див. табл. 3.1).

Таблиця 3.1 – Порівняння VPN-рішень за ключовими характеристиками

Критерій	OpenVPN (реалізація)	WireGuard	IPsec	Комерційні VPN
Рівень безпеки	Високий	Високий	Високий	Високий
Продуктивність	Середня	Висока	Висока	Висока
Гнучкість налаштування	Висока	Низька	Середня	Висока
Підтримка MFA	Так	Обмежена	Так	Так
Простота розгортання	Середня	Висока	Низька	Висока
Вартість	Низька (open-source)	Низька	Низька	Висока

OpenVPN розглядається як універсальне рішення, що базується на використанні протоколу TLS для встановлення захищеного з’єднання, при цьому його архітектура забезпечує підтримку сертифікатної моделі автентифікації, можливість реалізації багатофакторної автентифікації, гнучке налаштування маршрутів та політик доступу, а також здатність працювати у складних мережевих умовах, включаючи NAT і міжмережеві екрани, зокрема через використання TCP-порту 443, що дозволяє маскувати VPN-трафік під звичайний HTTPS, водночас до недоліків відноситься підвищене навантаження на процесор, що обумовлено інтенсивним використанням криптографічних операцій і було підтверджено під час практичного тестування на обмеженій апаратній платформі, однак така особливість не знижує загальної ефективності протоколу у випадках, де пріоритетом виступає безпека та керованість.

WireGuard представляє сучасний підхід до реалізації VPN, орієнтований на мінімалізм, високу продуктивність та простоту конфігурації, завдяки чому забезпечується низьке навантаження на апаратні ресурси та висока швидкість передачі даних, що робить його привабливим для сценаріїв із великими обсягами трафіку або обмеженими обчислювальними ресурсами, однак відсутність вбудованої підтримки багатофакторної автентифікації, використання статичної моделі ключів та обмежена гнучкість у реалізації складних політик доступу створюють певні обмеження для використання у корпоративних середовищах, де необхідна розширена система контролю доступу.

IPsec виступає класичним рішенням мережевого рівня, що забезпечує високий рівень безпеки та інтеграцію на рівні операційних систем і мережевого обладнання, дозволяючи реалізовувати захищені тунелі без необхідності встановлення додаткового програмного забезпечення на клієнтських пристроях, однак складність конфігурації, проблеми з роботою через NAT та обмежена гнучкість у порівнянні з іншими рішеннями знижують його привабливість для швидкого розгортання та адаптації у динамічних середовищах.

Порівняльний аналіз демонструє, що OpenVPN забезпечує найбільш збалансоване поєднання безпеки, гнучкості та керованості серед розглянутих протоколів, що робить його доцільним вибором для реалізації системи віддаленого доступу, навіть з урахуванням дещо нижчої продуктивності у порівнянні з більш легковаговими альтернативами.

Оцінка переваг і недоліків використання OpenVPN. На основі проведеної практичної реалізації формується об’єктивна оцінка використання OpenVPN як основи системи віддаленого доступу, що дозволяє виділити як його сильні сторони, так і певні обмеження, які необхідно враховувати при впровадженні.

До основних переваг відноситься висока гнучкість конфігурації, що дозволяє налаштовувати маршрутизацію, використовувати різні режими роботи, такі як TUN і TAP, а також адаптувати систему під конкретні вимоги мережевої інфраструктури, додатково слід відзначити розвинені механізми безпеки, що включають використання TLS, сертифікатної автентифікації та підтримку

багатофакторної автентифікації через зовнішні інструменти, що забезпечує високий рівень захисту доступу до ресурсів, ще одним важливим аспектом виступає можливість автоматизації процесів адміністрування, що реалізується через використання утиліт командного рядка та інтеграцію зі скриптами, що дозволяє централізовано керувати користувачами та швидко створювати конфігурації, також значною перевагою є широка сумісність із різними операційними системами, включаючи Windows, Linux та мобільні платформи, а також здатність працювати через NAT і міжмережеві екрани без складних додаткових налаштувань.

До недоліків відноситься підвищене навантаження на процесор, особливо при використанні складних алгоритмів шифрування, таких як AES-256-GCM, що було підтверджено під час тестування за допомогою системних утиліт моніторингу, ще одним обмеженням виступає ліцензійна політика деяких реалізацій, зокрема обмеження кількості одночасних підключень у безкоштовних версіях серверного програмного забезпечення, а також більша складність конфігурації у порівнянні з сучасними мінімалістичними рішеннями, що потребує більш глибокого розуміння принципів роботи мережевих протоколів та механізмів безпеки.

Узагальнена оцінка дозволяє зробити висновок, що незважаючи на наявні обмеження, OpenVPN залишається одним із найбільш придатних рішень для корпоративних середовищ, де критично важливими є контроль доступу, гнучкість налаштування та високий рівень захисту інформації, що повністю відповідає вимогам до побудованої системи віддаленого доступу.

Аналіз ефективності використання старого апаратного забезпечення. Аналіз ефективності використання старого апаратного забезпечення у контексті реалізації системи віддаленого доступу дозволяє розглядати застосування ноутбука бізнес-класу як серверної платформи як цілком обґрунтоване технічне рішення, зокрема використання моделі Lenovo ThinkPad T480 демонструє поєднання енергоефективності, автономності та достатньої продуктивності для виконання мережевих сервісів. До основних переваг такого підходу належить низьке

енергоспоживання процесора з тепловим пакетом близько 15 Вт, що забезпечує стабільну роботу навіть при тривалому навантаженні, а також можливість автономної роботи завдяки конструктивним особливостям системи живлення, що дозволяє підтримувати функціонування сервера без постійного підключення до електромережі, при цьому інтегрованість компонентів, таких як дисплей, клавіатура та вказівні пристрої, фактично формує повноцінний локальний інтерфейс керування без необхідності використання додаткового обладнання на зразок KVM-рішень, що значно спрощує експлуатацію та обслуговування

Додатковою перевагою виступає достатній рівень обчислювальної продуктивності, який забезпечується чотириядерним процесором із підтримкою багатопотоковості, що дозволяє ефективно обробляти VPN-трафік та супутні сервіси, включаючи файловий сервер, при цьому використання швидкого твердотільного накопичувача NVMe, зокрема моделі Samsung PM981, забезпечує високу швидкість доступу до даних і позитивно впливає на продуктивність файлових операцій у мережевому середовищі. Сукупність цих характеристик формує збалансовану платформу, здатну виконувати функції серверного вузла без необхідності придбання спеціалізованого обладнання.

Разом із перевагами спостерігаються і певні обмеження, серед яких варто виділити обмежену масштабованість, що пов’язана з апаратними характеристиками мобільної платформи, відсутність спеціалізованого апаратного прискорення криптографічних операцій у порівнянні з професійними серверними рішеннями, а також потенційні ризики перегріву при тривалих пікових навантаженнях, хоча проведене тестування показало відсутність критичних відхилень у роботі навіть при інтенсивному використанні. Отримані практичні результати підтверджують стабільність функціонування системи, прийнятний рівень швидкості передачі даних та відсутність збоїв у процесі тривалої експлуатації, що свідчить про можливість ефективного використання застарілого обладнання за умови правильної конфігурації програмного забезпечення.

3.3 Вартість реалізації та перспективи розвитку системи

У межах аналізу вартості реалізації та перспектив подальшого розвитку системи доцільно розглянути економічну складову розробленого рішення разом із визначенням можливих напрямків його масштабування та вдосконалення, при цьому основна увага зосереджується на ефективності використання наявного апаратного забезпечення, застосуванні програмного забезпечення з відкритим вихідним кодом та потенціалі адаптації системи до умов реальної експлуатації з обмеженими ресурсами.

Оцінка вартості побудови системи з використанням існуючого обладнання. Оцінка вартості побудови системи на базі існуючого обладнання демонструє суттєві переваги такого підходу, оскільки основою апаратної платформи виступає ноутбук бізнес-класу Lenovo ThinkPad T480, який було обрано з урахуванням технічних характеристик і практичної доцільності використання, конфігурація якого включає процесор Intel Core i5-8350U з чотирма фізичними ядрами та підтримкою восьми потоків обробки, 16 ГБ оперативної пам’яті стандарту DDR4, а також твердотільний накопичувач NVMe об’ємом 256 ГБ, зокрема моделі Samsung PM981, що забезпечує достатню швидкодію для виконання серверних задач. В умовах сучасного ринку вживаної техніки вартість подібної конфігурації становить орієнтовно від 10 000 до 11 000 гривень, причому зростання ціни у порівнянні з попередніми періодами пояснюється підвищеним попитом на енергоефективні мобільні рішення в умовах нестабільного електроживлення та дефіцитом окремих компонентів.

Забезпечення автономності системи досягається за рахунок підтримки технології Power Bridge, що передбачає використання двох акумуляторів, внутрішнього ємністю приблизно 24 Вт·год та зовнішнього до 72 Вт·год, що в сумі формує близько 96 Вт·год та дозволяє забезпечити від 10 до 12 годин автономної роботи сервера під керуванням Linux, при цьому для досягнення тривалішої автономності на рівні кількох діб доцільним є використання додаткових змінних

акумуляторів, наприклад моделей Lenovo 61++ або 01AV428, у кількості двох-трьох одиниць, вартість кожної з яких становить у середньому від 2600 до 2800 гривень, що формує загальну суму близько 8000 гривень при використанні трьох батарей (див. рис. 3.4).



Рисунок 3.4 – Додаткова батарея 01AV428 для підтримки системи

Додатковим джерелом резервного живлення виступає саморобний повербанк із підтримкою USB Type-C та Power Delivery потужністю до 45 Вт, зібраний на базі восьми акумуляторів формату 18650, зокрема Samsung INR18650-35E, сумарна вартість якого становить приблизно 980 гривень, включаючи корпус із контролером та самі елементи живлення, при цьому фактична ємність у межах приблизно 24 000 мА·год або близько 80 Вт·год забезпечує додатково до 8–10 годин автономної роботи, що значно підвищує загальну стійкість системи до перебоїв електроживлення (див. рис. 3.5).



Рисунок 3.5 – DIY-павербанк як резервне живлення для системи

Сукупний кошторис реалізації системи формує приблизну вартість на рівні 19 500–20 000 гривень з урахуванням ноутбука, додаткових акумуляторів та резервного джерела живлення, що є значно нижчим порогом входження у порівнянні з класичними серверними рішеннями, де лише серверне обладнання може коштувати від 30 000 гривень, а джерело безперебійного живлення додатково від 10 000 до 20 000 гривень, не враховуючи супутні витрати на інфраструктурую (див. табл. 3.2).

Таблиця 3.2 – Кошторис побудови системи на базі існуючого обладнання

Компонент	Кількість	Ціна за одиницю (грн)	Загальна вартість (грн)	Примітка
Ноутбук Lenovo ThinkPad T480	1	10500	10500	Основний сервер
Додаткові батареї Lenovo 61++ (01AV428)	3	2700	8100	Підвищення автономності
DIY павербанк 45W (8 x 18650)	1	1000	1000	Резервне живлення
Програмне забезпечення (OpenVPN Access Server)	1	0	0	Відкрите ПЗ
Мережеве обладнання	1	0	0	Наявна інфраструктура
Разом	—	—	19600	Орієнтовна загальна вартість

Все це дозволяє розглядати запропоноване рішення як економічно доцільне для малих організацій, навчальних лабораторій або автономних середовищ (див. рис. 3.6).

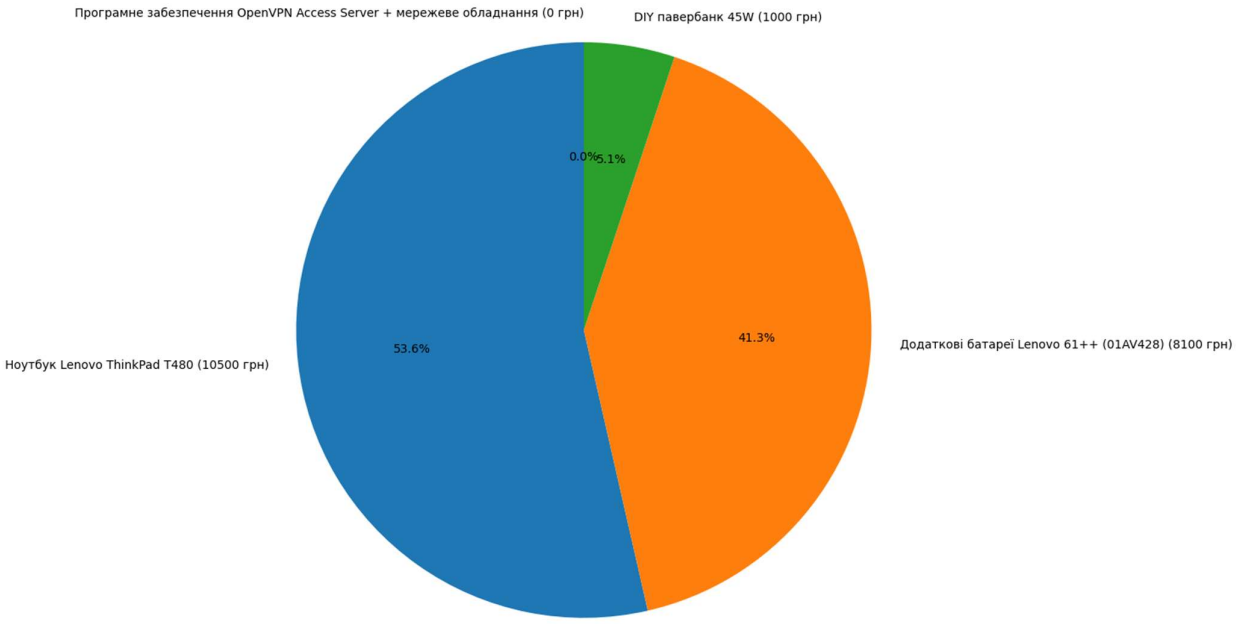


Рисунок 3.6 – Вартість побудови системи

Економічна доцільність використання OpenVPN та відкритого програмного забезпечення. Економічна доцільність додатково підсилюється використанням програмного забезпечення з відкритим вихідним кодом, зокрема операційної системи Ubuntu Server 24.04.3 LTS, яка забезпечує безкоштовне використання, довгострокову підтримку та регулярні оновлення безпеки, а також VPN-рішення на базі OpenVPN Access Server, безкоштовна версія якого дозволяє організувати до двох одночасних підключень із наявністю веб-інтерфейсу адміністрування, що є достатнім для тестових або невеликих розгортань, при цьому застосування відкритого програмного забезпечення дозволяє уникнути витрат на ліцензування, знизити загальний бюджет системи та спрямувати фінансові ресурси на підвищення автономності та надійності апаратної складової [30].

Можливості масштабування системи для великої кількості користувачів. Перспективи розвитку системи визначаються можливістю її масштабування як на програмному, так і на апаратному рівні, при цьому апаратна платформа Lenovo ThinkPad T480 підтримує розширення оперативної пам’яті до 64 ГБ за неофіційними даними, можливість встановлення більш ємного накопичувача, а

також підтримку технологій віртуалізації, що відкриває додаткові сценарії використання, включаючи розгортання декількох сервісів одночасно. Подальший розвиток може передбачати перехід на платну версію OpenVPN Access Server для збільшення кількості користувачів і централізованого управління або використання OpenVPN Community Edition як повністю безкоштовного рішення з ширшими можливостями конфігурації, що дозволяє адаптувати систему під більші навантаження (див. рис. 3.7).



Рисунок 3.7 – Можливість масштабування через апгрейд системи

Наявний потенціал системи дозволяє масштабувати її до десятків користувачів без суттєвих змін архітектури за умови оптимізації налаштувань і раціонального використання ресурсів, що у поєднанні з низькою вартістю, автономністю та гнучкістю формує практичне рішення для широкого спектра задач у середовищах із обмеженими ресурсами та підвищеними вимогами до незалежності від зовнішньої інфраструктури.

Інтеграція з системами централізованого управління доступом. У межах подальшого розвитку системи доцільно розглянути інтеграцію з централізованими засобами управління доступом, оскільки на поточному етапі адміністрування користувачів здійснюється через локальні облікові записи та bash-скрипти, що

забезпечує базову функціональність, однак має обмеження з точки зору масштабованості та керованості, а тому перспективним напрямком виступає впровадження централізованих механізмів автентифікації, зокрема інтеграція з каталогами Active Directory або LDAP, що дозволяє реалізувати єдину точку керування обліковими записами, автоматизувати процеси надання та відкликання доступу, зменшити вплив людського фактору та забезпечити відповідність корпоративним стандартам безпеки, при цьому практична цінність такого підходу проявляється у можливості оперативного блокування доступу до VPN у разі змін статусу користувача, наприклад при звільненні співробітника, коли обліковий запис деактивується у централізованій системі і відповідні права доступу припиняються без необхідності ручного втручання (див. рис. 3.8).

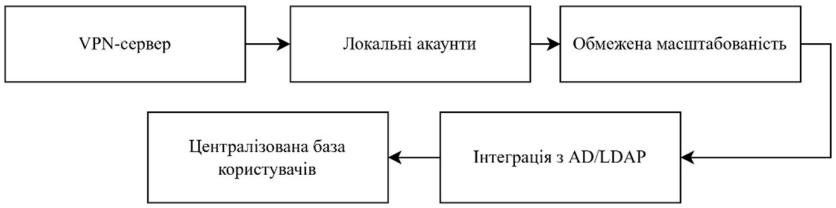


Рисунок 3.8 – Інтеграція з Active Directory

Перспективи розвитку системи з урахуванням сучасних технологій та хмарних рішень. Розгляд перспектив розвитку системи з урахуванням сучасних технологій дозволяє визначити декілька напрямків еволюції архітектури, серед яких особливу увагу привертає використання гібридної інфраструктури, що поєднує локальний сервер, реалізований на базі ноутбука Lenovo ThinkPad T480, із резервним вузлом у хмарному середовищі у вигляді VPS, що забезпечує додатковий рівень відмовостійкості та можливість швидкого відновлення у разі втрати доступу до локального обладнання, при цьому синхронізація конфігурацій між локальним та віддаленим середовищем дозволяє підтримувати актуальність даних і забезпечує доступ до системи з різних географічних точок (див. рис. 3.9).

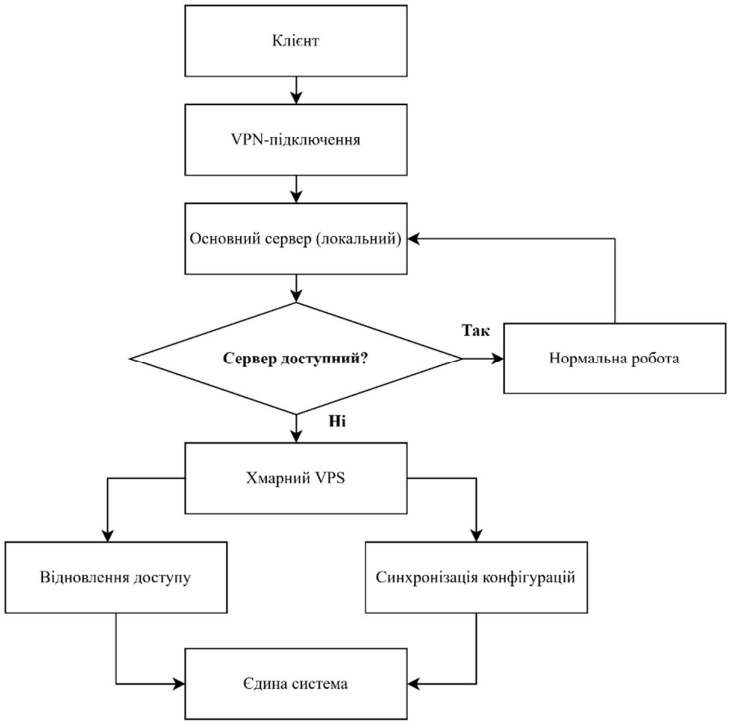


Рисунок 3.9 – Гібридна архітектура VPN

Додаткові напрями розвитку включають впровадження контейнеризації із використанням технологій на зразок Docker, що дозволяє ізолювати сервіси, спростити розгортання та підвищити гнучкість управління компонентами системи, а також інтеграцію систем моніторингу, таких як Prometheus і Grafana, що забезпечують збір метрик, візуалізацію стану системи та своєчасне виявлення потенційних проблем, при цьому можливості вдосконалення не обмежуються зазначеними підходами і можуть включати подальшу оптимізацію мережевих параметрів, впровадження додаткових механізмів безпеки та адаптацію до нових вимог експлуатації, що формує основу для поступового розвитку системи у напрямку більш складних та масштабованих рішень.

ВИСНОВКИ

У процесі проєктування сформовано архітектуру системи підключення користувачів до внутрішніх ресурсів на основі клієнт-серверної моделі із застосуванням VPN, де центральним елементом визначено сервер на базі фізичного обладнання у вигляді ноутбука Lenovo ThinkPad T480, використання якого обґрунтовано поєднанням достатньої продуктивності процесора, енергоефективності, наявності SSD-накопичувача та можливості автономної роботи від акумулятора, що забезпечує обробку підключень і взаємодію з внутрішньою мережею, а організація взаємодії між клієнтами та сервісами через захищений канал дозволяє реалізувати контрольований доступ до ресурсів.

У межах побудови системи виконано розгортання серверної частини на основі OpenVPN, налаштовано мережеві параметри та підготовлено конфігурації для підключення користувачів, при цьому процес встановлення захищеного тунелю реалізує передачу трафіку через зашифроване з’єднання, а описаний алгоритм функціонування охоплює етапи ініціалізації, узгодження параметрів, автентифікації та подальшої взаємодії з внутрішніми ресурсами.

Реалізація також включає налаштування клієнтських пристроїв, використання сертифікатів і автоматизацію створення конфігурацій за допомогою bash-скриптів, що спрощує адміністрування, а проведене тестування підтвердило стабільність роботи, коректність маршрутизації та доступність ресурсів, при цьому отримані результати демонструють достатній рівень продуктивності навіть за умов обмежених ресурсів і використання безпровідних та мобільних мереж.

Аналіз ефективності показав можливість оптимізації системи шляхом зниження навантаження на процесор, вибору ефективних параметрів передачі даних і впровадження багаторівневої автентифікації, а порівняння з альтернативними підходами підтвердило доцільність використання OpenVPN у поєднанні з наявним обладнанням, що забезпечує баланс між безпекою, гнучкістю, вартістю та можливістю подальшого масштабування й інтеграції.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Virtual Private Networks (VPNs). Cisco. URL: https://www.cisco.com/c/en/us/products/security/vpn-endpoint-security-clients/what-is-vpn.html (дата звернення: 25.02.2026).
2. Remote Access Security. NIST. URL: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-46r2.pdf (дата звернення: 28.02.2026).
3. Zero Trust Architecture Overview. Microsoft. URL: https://learn.microsoft.com/en-us/security/zero-trust/zero-trust-overview (дата звернення: 03.03.2026).
4. Why Use VPN for Business. Fortinet. URL: https://www.fortinet.com/resources/cyberglossary/vpn (дата звернення: 06.03.2026).
5. What is VPN Tunneling?. Cloudflare. URL: https://www.cloudflare.com/learning/network-layer/what-is-tunneling/ (дата звернення: 10.03.2026).
6. Types of VPN Protocols. ZoogVPN. URL: https://zoogvpn.com/vpn-protocols/?srsltid=AfmBOorO_SFy1bs0ftKlR-KAVeCH2hv58-dgAhhrwA9HTgXKiegqwNZw (дата звернення: 14.03.2026).
7. Encryption and Authentication in VPN. IBM. URL: https://www.ibm.com/topics/encryption (дата звернення: 18.03.2026).
8. OpenVPN Technology Overview. OpenVPN Inc.. URL: https://openvpn.net/community-resources/how-openvpn-works/ (дата звернення: 22.03.2026).
9. Client-Server Architecture Explained. Oracle. URL: https://www.oracle.com/database/what-is-client-server-architecture/ (дата звернення: 26.03.2026).
10. Network Client-Server Model. GeeksforGeeks. URL: https://www.geeksforgeeks.org/client-server-model/ (дата звернення: 30.03.2026).
11. Server Infrastructure Basics. Red Hat. URL: https://www.redhat.com/en/topics/infrastructure (дата звернення: 03.04.2026).
12. OpenVPN in Client-Server Model. OpenVPN Inc.. URL: https://openvpn.net/vpn-server-resources/ (дата звернення: 07.04.2026).
13. Network Security Threats. Cisco. URL: https://www.cisco.com/c/en/us/products/security/common-cyberattacks.html (дата звернення: 11.04.2026).
14. Data Encryption Techniques. IBM. URL: https://www.ibm.com/topics/data-encryption (дата звернення: 15.04.2026).
15. Authentication Methods Explained. Okta. URL: https://www.okta.com/identity-101/authentication/ (дата звернення: 19.04.2026).
16. Certificates in VPN. DigiCert. URL: https://www.digicert.com/what-is-an-ssl-certificate (дата звернення: 23.04.2026).

17. VPN Software Comparison. TechRadar. URL: https://www.techradar.com/vpn/best-vpn (дата звернення: 27.04.2026).
18. OpenVPN Advantages. OpenVPN Inc.. URL: https://openvpn.net/faq/ (дата звернення: 01.05.2026).
19. Why Choose OpenVPN. Comparitech. URL: https://www.comparitech.com/blog/vpn-privacy/openvpn-guide/ (дата звернення: 01.05.2026).
20. Hub-and-Spoke Network Topology. Microsoft. URL: https://learn.microsoft.com/en-us/azure/architecture/reference-architectures/hybrid-networking/hub-spoke (дата звернення: 02.05.2026).
21. Bare Metal Server Explained. IBM. URL: https://www.ibm.com/topics/bare-metal (дата звернення: 03.05.2026).
22. OpenVPN Access Server Setup Guide. OpenVPN Inc.. URL: https://openvpn.net/vpn-server-resources/installing-openvpn-access-server/ (дата звернення: 03.05.2026).
23. Linux Network Configuration. Ubuntu. URL: https://ubuntu.com/server/docs/network-configuration (дата звернення: 04.05.2026).
24. TLS Handshake Explained. Cloudflare. URL: https://www.cloudflare.com/learning/ssl/what-happens-in-a-tls-handshake/ (дата звернення: 04.05.2026).
25. Bash Scripting Basics. GNU. URL: https://www.gnu.org/software/bash/manual/ (дата звернення: 05.05.2026).
26. VPN Performance Testing. Speedtest by Ookla. URL: https://www.speedtest.net/insights/blog/vpn-performance/ (дата звернення: 06.05.2026).
27. AES Encryption Overview. NIST. URL: https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.197.pdf (дата звернення: 06.05.2026).
28. TCP vs UDP Performance. Cloudflare. URL: https://www.cloudflare.com/learning/ddos/glossary/tcp-vs-udp/ (дата звернення: 07.05.2026).
29. Multi-Factor Authentication (MFA). Microsoft. URL: https://learn.microsoft.com/en-us/azure/active-directory/authentication/concept-mfa-howitworks (дата звернення: 07.05.2026).
30. Open Source vs Commercial Software. Red Hat. URL: https://www.redhat.com/en/topics/open-source/what-is-open-source (дата звернення: 07.05.2026).
31. Красюк А.В., Торошанко Я.І. Автоматизована система захищеного доступу до внутрішніх ресурсів на основі OpenVPN. «Сучасний стан та перспективи розвитку IoT» : VII Науково-техн. конф., м. Київ, 20 квітня 2026 р. С.84-86.
32. Красюк А.В., Торошанко Я.І. Оптимізація та масштабування системи vpn доступу на основі OpenVPN в умовах обмежених ресурсів. «Застосування програмного забезпечення в ІКТ» : VII Всеукр. науково-техн. конф., м. Київ, 23 квітня 2026 р. (подано до публікації).

Додаток А

Фрагмент масового створення конфігурацій машин для OpenVPN Access Server

```
#!/usr/bin/env bash

# ============================================================================
# Bulk creation of machine configs for OpenVPN Access Server (Autologin)
# ============================================================================

SACLI="/usr/local/openvpn_as/scripts/sacli"
INPUT_FILE="./machines.txt"
OUT_DIR="/home/$USER/openvpn/clients/machines"

# Output colors
GREEN='\033[0;32m'
BLUE='\033[0;34m'
NC='\033[0m' # No Color

mkdir -p "$OUT_DIR"

if [[ ! -f "$INPUT_FILE" ]]; then
    echo -e "${BLUE}Error: File $INPUT_FILE not found!${NC}"
    exit 1
fi

echo -e "${BLUE}--- Launch of mass profile creation for MACHINES (Autologin) ---
${NC}"

while IFS= read -r MACHINE_ID || [ -n "$MACHINE_ID" ]; do
    # Cleaning name from extra characters
    CLEAN_MACHINE=$(echo "$MACHINE_ID" | tr -d '\r' | tr ' ' '_')

    # Skip blank lines
    [[ -z "$CLEAN_MACHINE" ]] && continue

    echo -ne "Processing machine: ${GREEN}$CLEAN_MACHINE${NC}... "

    # Step 1 - Creating user
    sudo $SACLI --user "$CLEAN_MACHINE" --key "type" --value "user_connect"
UserPropPut > /dev/null

    # Step 2 - Activating Autologin mode (allows connection without password)
    sudo $SACLI --user "$CLEAN_MACHINE" --key "prop_autologin" --value "true"
UserPropPut > /dev/null

    # Step 3 - Export Autologin configuration directly to file
    sudo $SACLI --user "$CLEAN_MACHINE" GetAutologin >
"${OUT_DIR}/${CLEAN_MACHINE}.ovpn" 2>/dev/null

    echo -e "${GREEN}DONE${NC}"

done < "$INPUT_FILE"

# Final update of server configuration in memory
sudo $SACLI start > /dev/null

echo -e "${BLUE}------------------------------------------------------------${NC}"
echo -e "${GREEN}Process complete! All configs in: $OUT_DIR${NC}"
```

Додаток Б

Фрагмент масового створення конфігурацій співробітників для OpenVPN Access Server

```
#!/usr/bin/env bash

# ============================================================================
# Bulk creation of employees configs for OpenVPN Access Server (MFA Enabled)
# ============================================================================

SACLI="/usr/local/openvpn_as/scripts/sacli"
INPUT_FILE="./employees.txt"
DEFAULT_PASS="ChangeMe2026!"
OUT_DIR="/home/$USER/openvpn/clients/employees"

# Output colors
GREEN='\033[0;32m'
BLUE='\033[0;34m'
NC='\033[0m' # No Color

mkdir -p "$OUT_DIR"

if [[ ! -f "$INPUT_FILE" ]]; then
    echo -e "${BLUE}Error: File $INPUT_FILE not found!${NC}"
    exit 1
fi

echo -e "${BLUE}--- Launching mass profile creation for EMPLOYEES (MFA Enabled) --
-${NC}"

# Creating group for employees
sudo $SACLI --group "employees" --key "type" --value "group" GroupPropPut >
/dev/null

while IFS= read -r USERNAME || [ -n "$USERNAME" ]; do
    # Cleaning name from extra characters
    CLEAN_USER=$(echo "$USERNAME" | tr -d '\r' | tr ' ' '_')

    # Skip blank lines
    [[ -z "$CLEAN_USER" ]] && continue

    echo -ne "Creating profile for: ${GREEN}$CLEAN_USER${NC}... "

    # Step 1 - Creating user
    sudo $SACLI --user "$CLEAN_USER" --key "type" --value "user_connect"
UserPropPut > /dev/null

    # Step 2 - Setting password
    sudo $SACLI --user "$CLEAN_USER" --new_pass "$DEFAULT_PASS" SetLocalPassword >
/dev/null

    # Step 3 - Assignment to employees group
    sudo $SACLI --user "$CLEAN_USER" --key "group_name" --value "employees"
UserPropPut > /dev/null

    # Step 4 - MFA settings
    # 4.1 Secret generation
    sudo $SACLI --user "$CLEAN_USER" --lock 0 GoogleAuthRegen > /dev/null 2>&1

    # 4.2 Extracting secret (using jq)
    MFA_SECRET=$(sudo $SACLI UserPropGet | jq -r
".[\"$CLEAN_USER\"].pvt_google_auth_secret // \"ERROR\"")
```

```
    # 4.3 Enabling mandatory MFA
    sudo $SACLI --user "$CLEAN_USER" --key "prop_google_auth" --value "true"
UserPropPut > /dev/null

    # Step 5 - Export and packaging
    OVPN_FILE="${OUT_DIR}/${CLEAN_USER}.ovpn"
    README_FILE="${OUT_DIR}/${CLEAN_USER}-README.txt"
    ZIP_FILE="${OUT_DIR}/${CLEAN_USER}_VPN_Access.zip"

    # 5.1 Generating .ovpn
    sudo $SACLI --user "$CLEAN_USER" GetUserlogin > "$OVPN_FILE" 2>/dev/null

    # 5.2 Creating clear README
    cat <<EOF > "$README_FILE"
Data for accessing the network via OpenVPN Connect:
------------------------------------------------
User:         $CLEAN_USER
Password:     $DEFAULT_PASS
MFA code:     $MFA_SECRET

INSTRUCTION:
1. Install Google Authenticator app on your smartphone.
2. Select "Enter Setup Key" and enter the "MFA Code" listed above.
3. Import the .ovpn file into the OpenVPN Connect app.
4. When connecting, enter your password and then the 6-digit code from Google
Authenticator.
EOF

    # 5.3 Creating individual archive for employee
    zip -j "$ZIP_FILE" "$OVPN_FILE" "$README_FILE" > /dev/null

    # 5.4 Deleting temporary files
    rm -f "$OVPN_FILE" "$README_FILE"

    echo -e "${GREEN}DONE${NC}"

done < "$INPUT_FILE"

# Final configuration update
sudo $SACLI start > /dev/null

echo -e "${BLUE}----------------------------------------${NC}"
echo -e "${GREEN}Process complete! All archives in: $OUT_DIR${NC}"
```

Додаток В

ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ

(Презентація)

ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ
ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
Кафедра комп'ютерної інженерії



КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня бакалавра

НА ТЕМУ: «АВТОМАТИЗОВАНА СИСТЕМА ПІДКЛЮЧЕННЯ КОРИСТУВАЧІВ ДО ВНУТРІШНІХ РЕСУРСІВ ЗА ДОПОМОГОЮ OPENVPN»

Виконав студент групи КІД-41
Красюк Андрій Валерійович
Керівник кваліфікаційної роботи:
Торошанко Ярослав Іванович,
канд. техн. наук, доц., с.н.с.

Київ – 2026

Мета, об’єкт, предмет дослідження

Мета роботи – оптимізація та спрощення процесу підключення користувачів до внутрішніх ресурсів за рахунок використання клієнт-серверної моделі та автоматизації управління доступом

Об’єкт дослідження – процес організації захищеного віддаленого доступу до внутрішніх ресурсів у комп’ютерних мережах

Предмет дослідження – засоби та технології побудови, налаштування й оптимізації VPN-систем на основі клієнт-серверної архітектури

Актуальність роботи

У роботі розглянуто теоретичні основи організації захищеного віддаленого доступу, принципи функціонування VPN, механізми тунелювання, шифрування та автентифікації, а також архітектуру клієнт-серверних систем. Проаналізовано сучасні VPN-протоколи, зокрема OpenVPN, WireGuard та IPSec, і визначено їх переваги та обмеження з урахуванням вимог до безпеки, продуктивності та гнучкості.

Практична частина включає реалізацію автоматизованої системи підключення користувачів із використанням OpenVPN із розгортанням серверної частини на базі старого апаратного забезпечення — ноутбука Lenovo ThinkPad T480, що обґрунтовано його енергоефективністю, достатньою продуктивністю та автономністю. Виконано налаштування системи, автоматизацію створення клієнтських конфігурацій, впровадження багаторівневої автентифікації, а також проведено тестування продуктивності, стабільності та ефективності роботи в умовах обмежених ресурсів і нестабільних мереж.

Архітектура запропонованого рішення

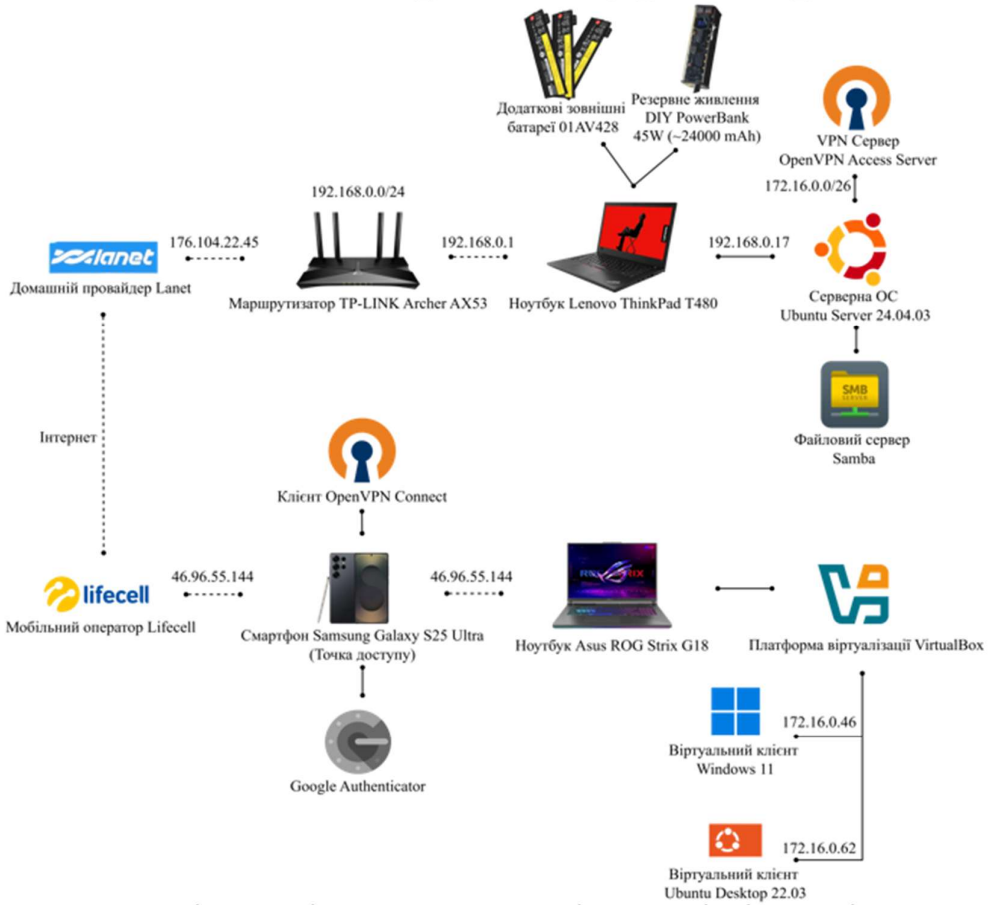


Рис. 2.3 – Архітектура підключення користувачів до внутрішніх ресурсів

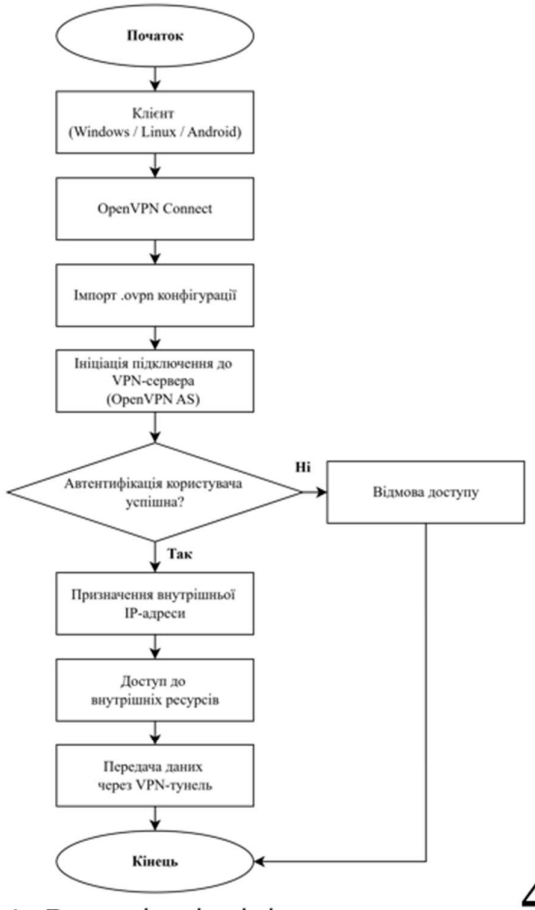


Рис. 2.4 – Взаємодія клієнтів із сервером

Практична реалізація (сервер на базі вживаного обладнання)

Табл. 2.2 – Технічні характеристики серверної платформи (Lenovo ThinkPad T480)

Компонент	Характеристика	Значення
Процесор	Модель	Intel Core i5-8350U
Оперативна пам’ять	Тип / обсяг	DDR4 / 16 ГБ
Накопичувач	Тип / обсяг	NVMe SSD / 256 ГБ
Мережеві інтерфейси	Тип	Ethernet, Wi-Fi
Безпека	Модуль	TPM 2.0
Живлення	Технологія	Power Bridge (2 акумулятори)
Автономність	Час роботи	до 10–12 годин
Розширення	Можливості	RAM, акумулятори



Рис. 2.19 – Апаратна платформа сервера: Lenovo ThinkPad T480

Особливості реалізованого рішення

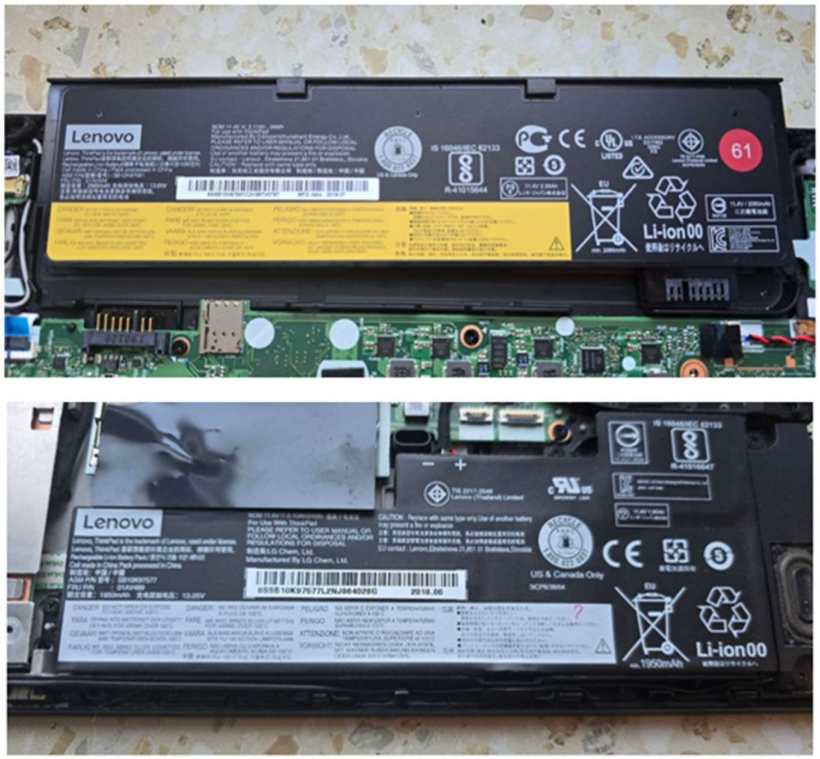


Рис. 2.20 – Внутрішня незмінна та зовнішня how-swap батарея



Рис. 3.5 – DIY-павербанк як резервне живлення для системи

Алгоритми у реалізованому рішенні

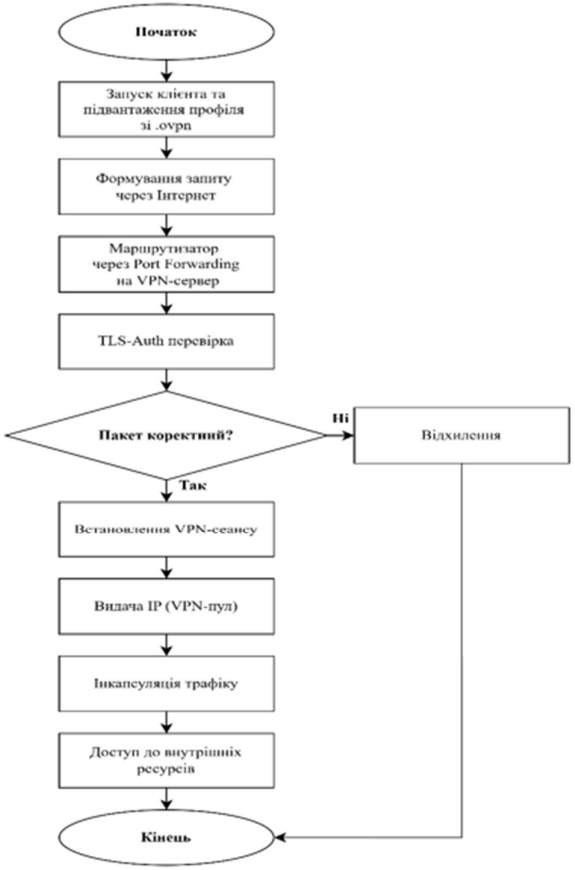


Рис. 2.15 – Алгоритм підключення користувача до внутрішньої мережі через VPN

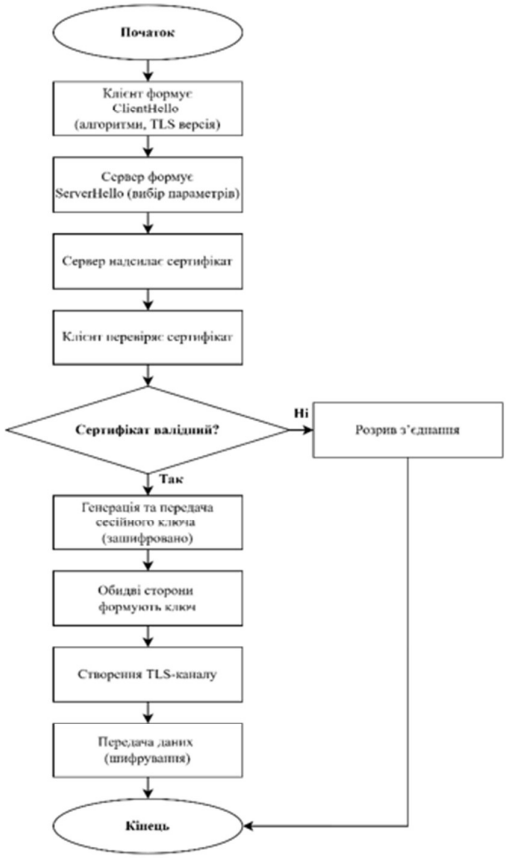


Рис. 2.16 – Процес встановлення захищеного TLS-з'єднання

Алгоритм у реалізованому рішенні

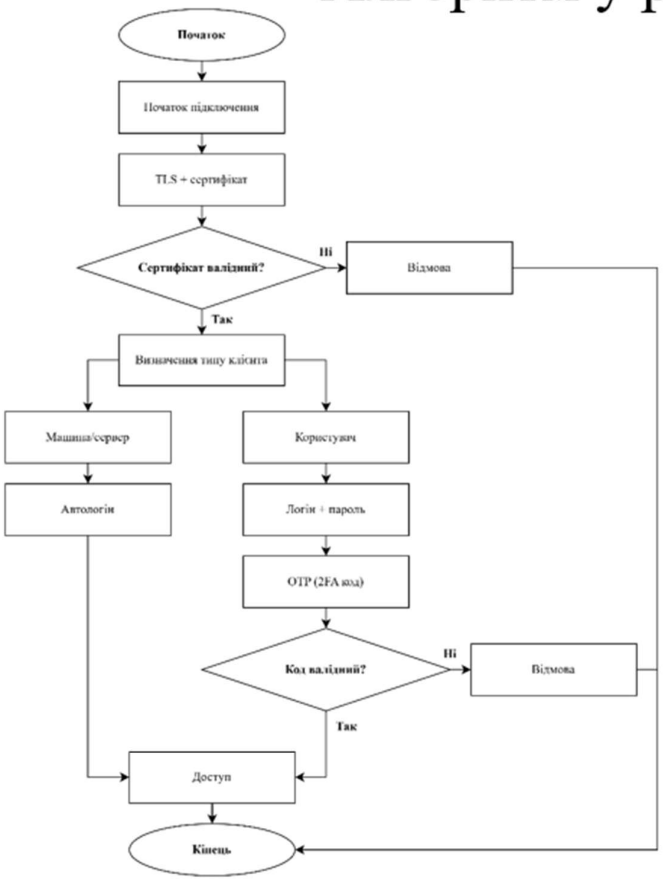


Рис. 2.17 – Алгоритм автентифікації користувача на основі сертифікатів, ключів та другого фактора

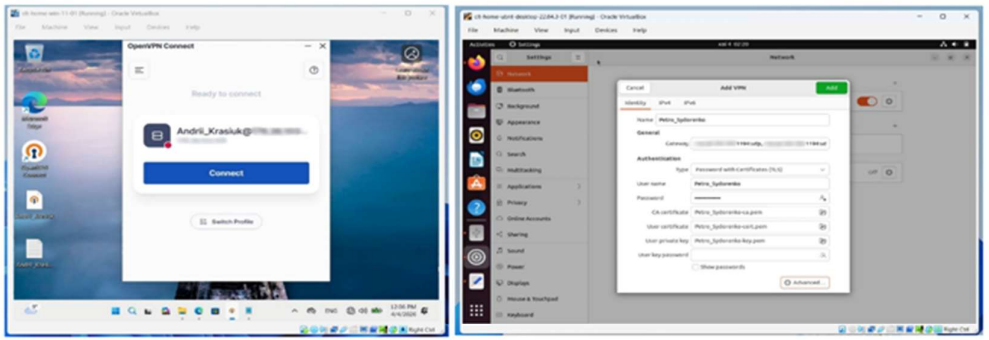


Рис. 2.23 – Процес конфігурації клієнтів шляхом імпорту конфігураційних файлів

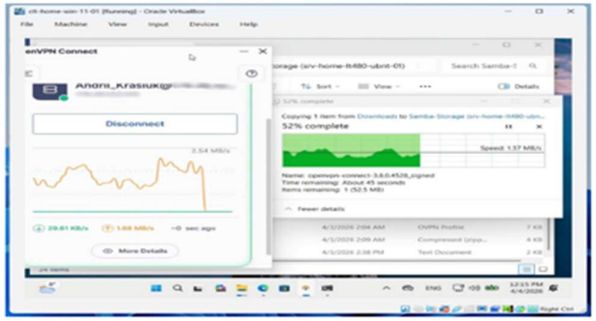




Рис. 2.25 – Клієнтська сторона зі доступом до внутрішніх ресурсів

Інтеграція автоматизації у формуванні конфігурацій

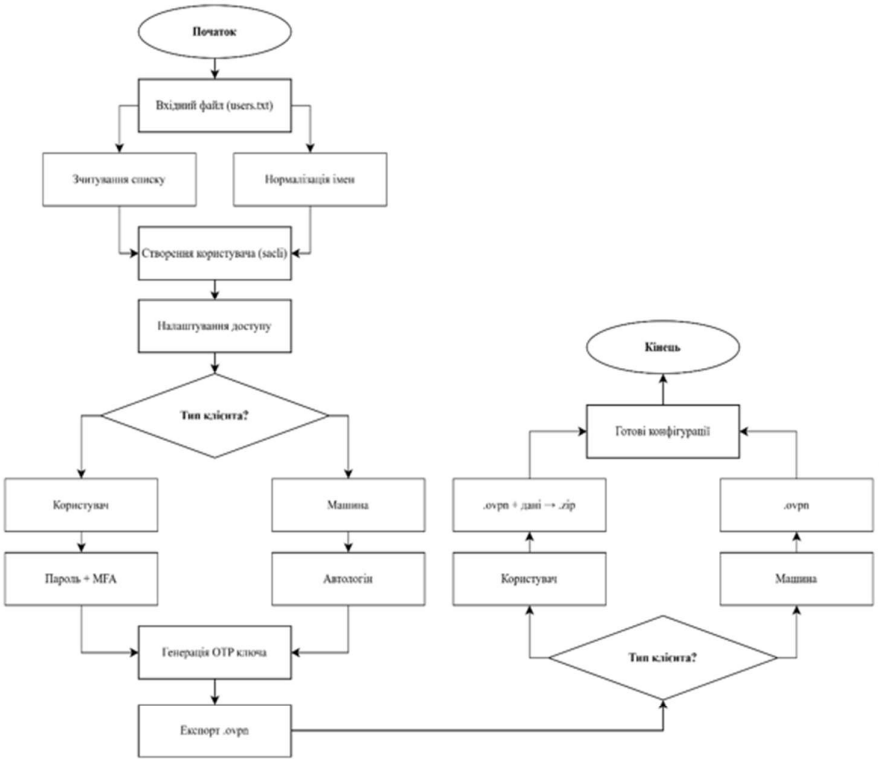


Рис. 2.18 – Алгоритм автоматизованого створення клієнтських конфігурацій за допомогою bash-скриптів

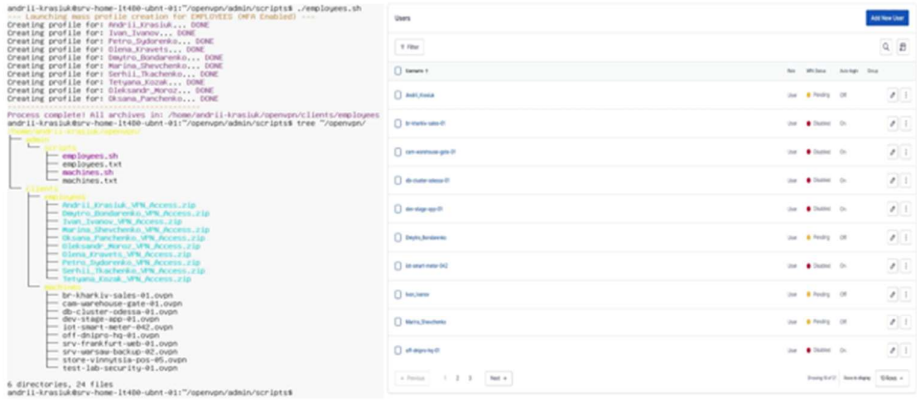


Рис. 2.12 – Генерація клієнтських конфігурацій

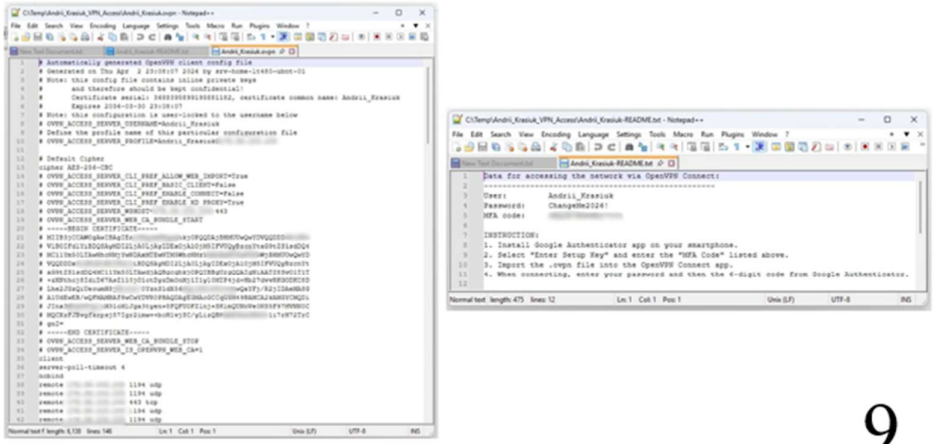


Рис. 2.13 – Формування унікальних файлів зі детальною інформацією для співробітника

Практичне використання ресурсів через встановлений тунель

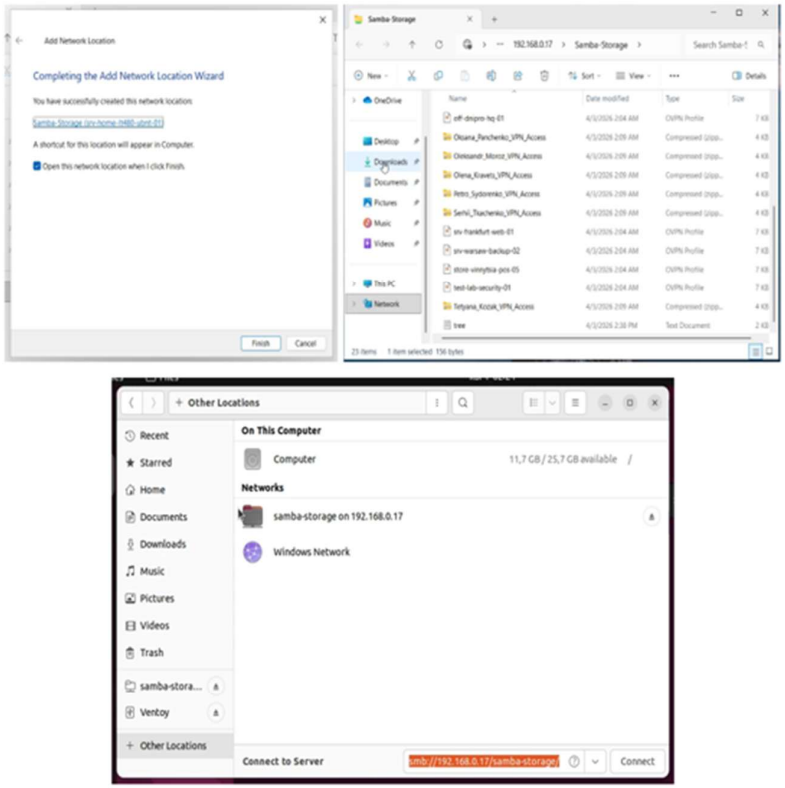


Рис. 2.26 – Файловий сервер на базі Samba як внутрішній ресурс

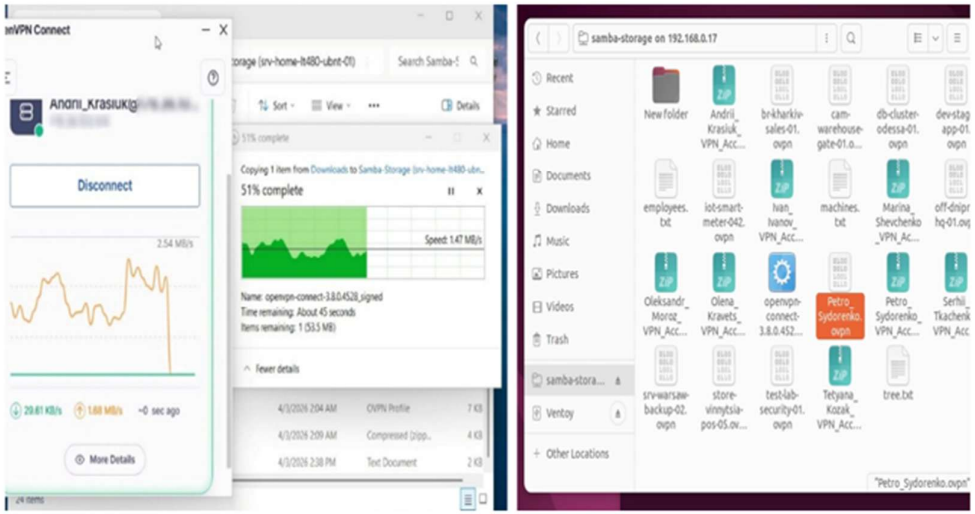


Рис. 2.27 – Робота зі файловим сервером через VPN

Навантаження під час використання ресурсів двома клієнтами

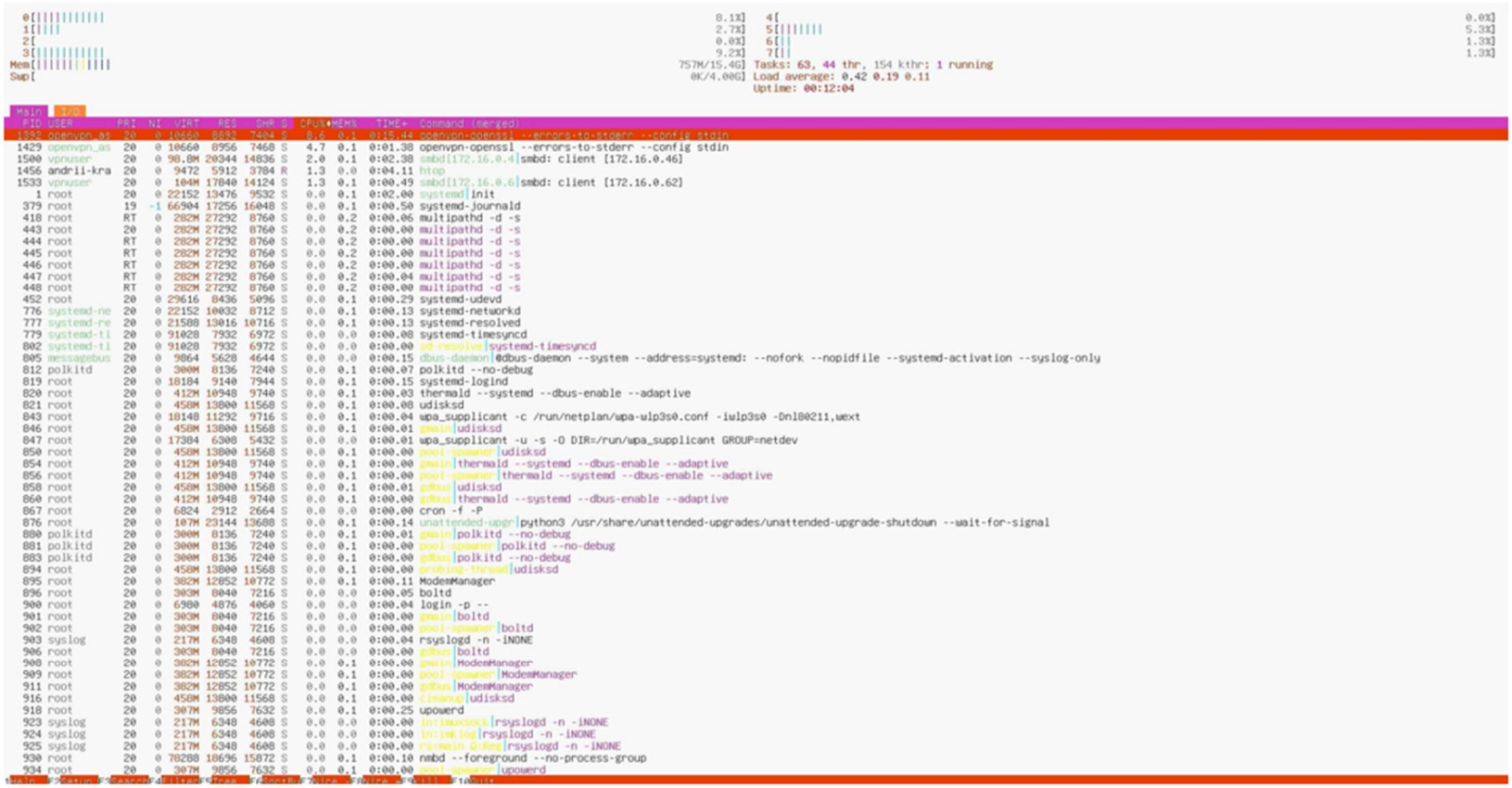


Рис. 2.29 – Моніторинг навантаження системи зі двома активними клієнтами

Аналіз продуктивності між клієнтом та сервером через iperf3

```
C:\Temp\iperf3.20>.\iperf3.exe -c 192.168.0.17
Connecting to host 192.168.0.17, port 5201
[  5] local 172.16.0.46 port 3015 connected to 192.168.0.17 port 5201
[ ID] Interval           Transfer     Bitrate
[  5]   0.00-1.01   sec  5.00 MBytes  41.6 Mbits/sec
[  5]   1.01-2.00   sec  4.50 MBytes  38.0 Mbits/sec
[  5]   2.00-3.01   sec  3.75 MBytes  31.2 Mbits/sec
[  5]   3.01-4.01   sec  4.38 MBytes  36.7 Mbits/sec
[  5]   4.01-5.01   sec  3.50 MBytes  29.4 Mbits/sec
[  5]   5.01-6.01   sec  4.12 MBytes  34.6 Mbits/sec
[  5]   6.01-7.01   sec  3.75 MBytes  31.5 Mbits/sec
[  5]   7.01-8.01   sec  3.50 MBytes  29.4 Mbits/sec
[  5]   8.01-9.00   sec  2.88 MBytes  24.2 Mbits/sec
[  5]   9.00-10.01  sec  2.25 MBytes  18.8 Mbits/sec
- - - - - - - - - - - - - - - - - - - - - - - - -
[ ID] Interval           Transfer     Bitrate
[  5]   0.00-10.01  sec  37.6 MBytes  31.5 Mbits/sec                  sender
[  5]   0.00-10.10  sec  35.8 MBytes  29.7 Mbits/sec                  receiver

iperf Done.
```

```
C:\Temp\iperf3.20>.\iperf3.exe -c 192.168.0.17 -R
Connecting to host 192.168.0.17, port 5201
Reverse mode, remote host 192.168.0.17 is sending
[  5] local 172.16.0.46 port 40059 connected to 192.168.0.17 port 5201
[ ID] Interval           Transfer     Bitrate
[  5]   0.00-1.01   sec  3.50 MBytes  29.1 Mbits/sec
[  5]   1.01-2.01   sec  2.25 MBytes  18.8 Mbits/sec
[  5]   2.01-3.00   sec  2.00 MBytes  17.0 Mbits/sec
[  5]   3.00-4.00   sec  2.00 MBytes  16.8 Mbits/sec
[  5]   4.00-5.01   sec  2.75 MBytes  22.7 Mbits/sec
[  5]   5.01-6.01   sec  2.25 MBytes  19.0 Mbits/sec
[  5]   6.01-7.00   sec  2.75 MBytes  23.2 Mbits/sec
[  5]   7.00-8.01   sec  2.50 MBytes  20.8 Mbits/sec
[  5]   8.01-9.01   sec  1.88 MBytes  15.7 Mbits/sec
[  5]   9.01-10.00  sec  2.38 MBytes  20.1 Mbits/sec
- - - - - - - - - - - - - - - - - - - - - - - - -
[ ID] Interval           Transfer     Bitrate         Retr
[  5]   0.00-10.03  sec  25.0 MBytes  20.9 Mbits/sec   38             sender
[  5]   0.00-10.00  sec  24.2 MBytes  20.3 Mbits/sec                  receiver

iperf Done.
```

Рис. 2.30 – Тестування швидкості передачі через iperf3

Методи оптимізації роботи реалізованого рішення

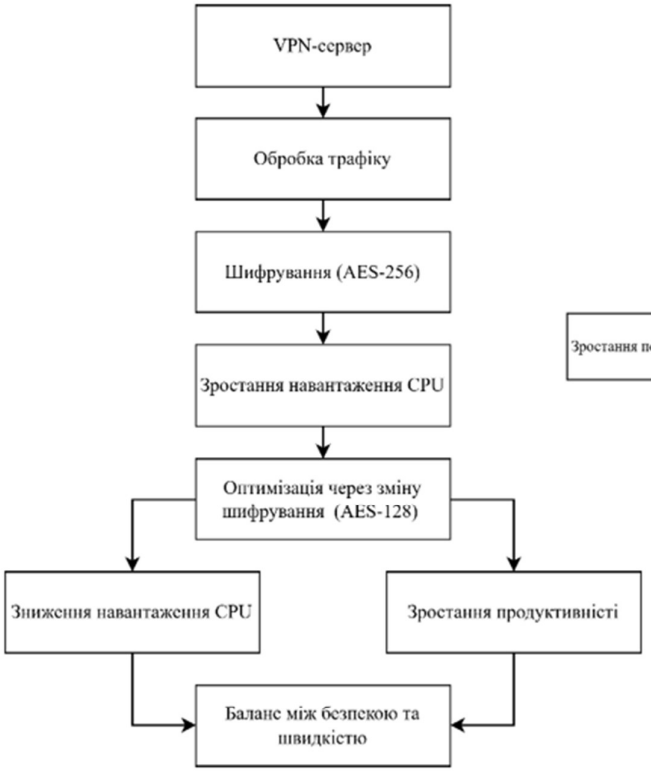


Рис. 3.1 – Оптимізація навантаження на апаратну платформу сервера

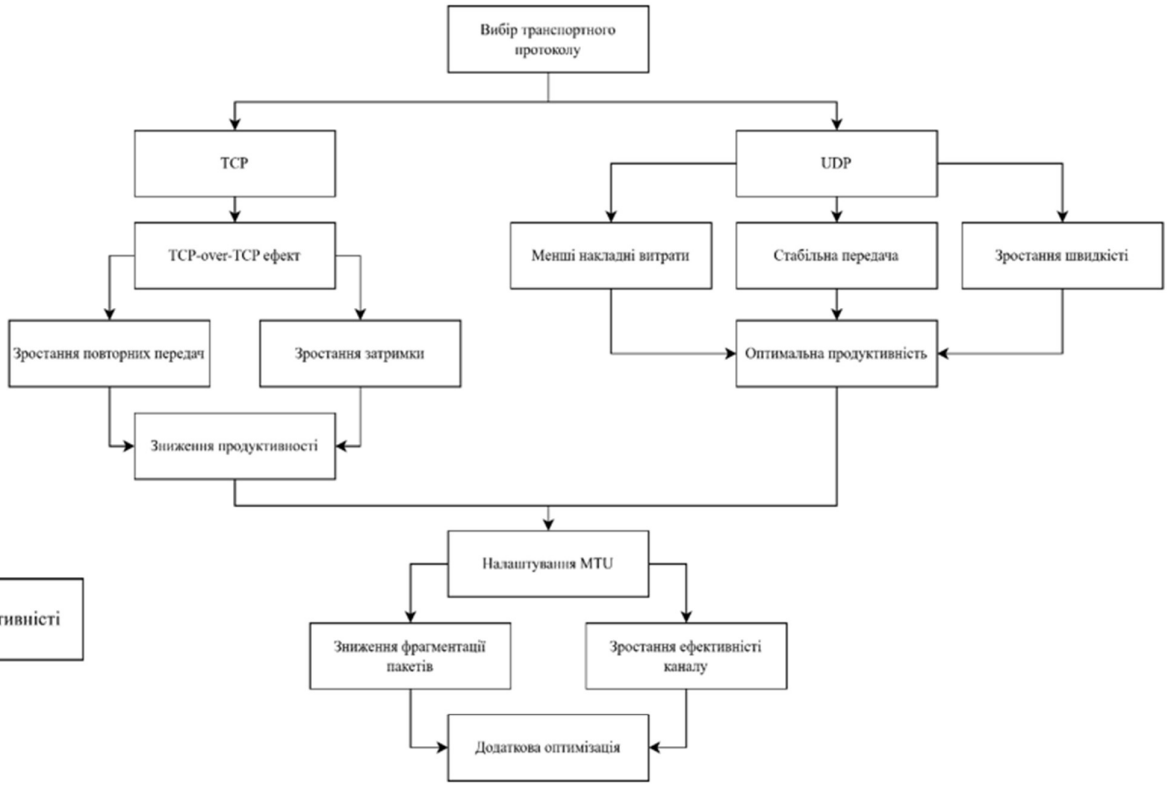


Рис. 3.2 – Оптимізація передачі даних у межах VPN-з'єднання

Економічна ефективність обраного рішення

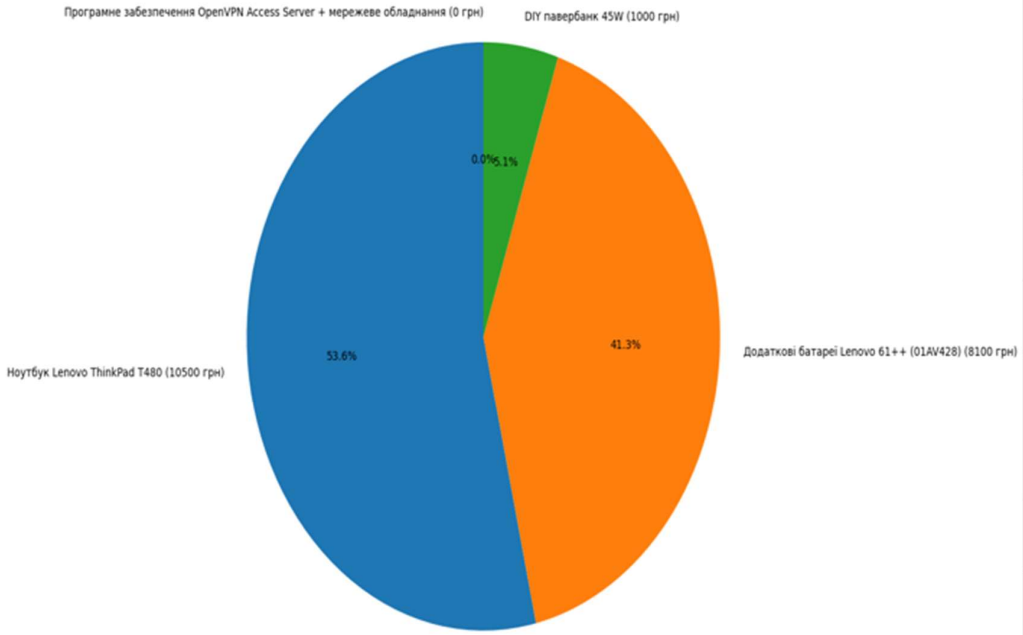


Рис. 3.6 – Вартість побудови системи

Табл. 3.2 – Кошторис побудови системи на базі існуючого обладнання

Компонент	Кількість	Ціна за одиницю (грн)	Загальна вартість (грн)	Примітка
Ноутбук Lenovo ThinkPad T480	1	10500	10500	Основний сервер
Додаткові батареї Lenovo 61++ (01AV428)	3	2700	8100	Підвищення автономності
DIY павербанк 45W (8 x 18650)	1	1000	1000	Резервне живлення
Програмне забезпечення (OpenVPN Access Server)	1	0	0	Відкрите ПЗ
Мережеве обладнання	1	0	0	Наявна інфраструктура
Разом	—	—	19600	Орієнтовна загальна вартість

Висновки

У роботі спроєктовано та реалізовано систему захищеного віддаленого доступу до внутрішніх ресурсів на основі VPN із використанням клієнт-серверної архітектури. Як серверну платформу обрано ноутбук Lenovo ThinkPad T480, що забезпечує достатню продуктивність, енергоефективність і автономність, а розгортання OpenVPN дозволило організувати захищений канал зв'язку з повним циклом встановлення з'єднання, автентифікації та доступу до ресурсів.

Практична реалізація, автоматизація конфігурацій і результати тестування підтвердили працездатність, стабільність і достатню продуктивність системи навіть за обмежених ресурсів. Проведена оптимізація та порівняння рішень довели доцільність обраного підходу, який забезпечує баланс між безпекою, вартістю та можливістю масштабування й подальшої інтеграції.

Результати впровадження

1. Красюк А.В., Торошанко Я.І. Автоматизована система захищеного доступу до внутрішніх ресурсів на основі OpenVPN. «Сучасний стан та перспективи розвитку IoT» : VII Науково-техн. конф., м. Київ, 20 квітня 2026 р. С.84-86.
2. Красюк А.В., Торошанко Я.І. Оптимізація та масштабування системи vpn доступу на основі OpenVPN в умовах обмежених ресурсів. «Застосування програмного забезпечення в ІКТ» : VII Всеукр. науково-техн. конф., м. Київ, 23 квітня 2026 р. (подано до публікації).