Хмельницький національний університет
Факультет інформаційних технологій
Кафедра комп'ютерної інженерії та інформаційних систем

КВАЛІФІКАЦІЙНА РОБОТА

бакалавр
Освітній рівень

Програмно-технічні засоби VPN сервера на основі одноплатної комп'ютерної системи Raspberry Pi4
Назва теми

КвРКІ 210237.21.02.65 ПЗ
Шифр

Галузь знань *12 «Інформаційні технології»*
Шифр, назва

Спеціальність *123 «Комп'ютерна інженерія»*
Шифр, назва

Освітня програма *«Комп'ютерна інженерія та програмування»*
Назва

Виконала: студентка IV курсу, група КІ2-21-2 _____ Дар'я КОРЖОВА
Підпис, Ініціали, прізвище

Керівник _____ Сергій ЛИСЕНКО
Підпис, дата, Ініціали, прізвище

Нормоконтролер _____ Тетяна КИСІЛЬ
Підпис, дата, Ініціали, прізвище

До захисту допускаю:
зав. кафедри комп'ютерної інженерії та інформаційних систем _____ Ольга ПАВЛОВА
Підпис, Ініціали, прізвище

«9» червня 2025 р.

Хмельницький 2025

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Кафедра КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ ТА ІНФОРМАЦІЙНИХ СИСТЕМ

Освітній рівень БАКАЛАВР

Галузь знань 12 ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ

Спеціальність 123 КОМП'ЮТЕРНА ІНЖЕНЕРІЯ

Освітня програма «КОМП'ЮТЕРНА ІНЖЕНЕРІЯ ТА ПРОГРАМУВАННЯ»

ЗАТВЕРДЖУЮ

Зав. кафедри Ольга ПАВЛОВА

" 10 " 01 2025 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ БАКАЛАВРА

Дар'ї КОРЖОВІЙ

Прізвище, ім'я, по батькові студента

1. Тема проекту (роботи) Програмно-технічні засоби VPN сервера на основі одноплатної комп'ютерної системи Raspberry Pi4

Керівник проекту (роботи) Сергій ЛИСЕНКО, д.т.н., проф.

Прізвище, ім'я, по батькові, науковий ступінь, вчене звання

Затверджена наказом ректора університету від 01.03.2025 р. № 5

2. Строк подання студентом проекту (роботи) на кафедру 01.06.2025 р.

3. Вихідні дані до проекту (роботи) Завдання на кваліфікаційну роботу

4. Зміст пояснювальної записки (перелік питань, які потрібно розробити)

Програмно-технічні засоби vpn сервера на основі одноплатної комп'ютерної системи raspberry pi4 та постановка задачі щодо її удосконалення

Обгрунтування вибору компонентів та середовища реалізації

Програмно-технічний засіб VPN сервера на основі одноплатної комп'ютерної системи Raspberry PI4

5. Перелік графічного матеріалу (із зазначенням обов'язкових креслень)

Принцип роботи

Схема роботи програмної частини

Принципова схема апаратної частини

6. Консультанти розділів дипломного проекту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Нормоконтроль	Тетяна КИСІЛЬ, доцент кафедри КІС		
Антиплагіат	Андрій НІЧЕПОРУК, доцент кафедри КІС		

7. Дата видачі завдання «10» 01 2025 р.

КАЛЕНДАРНИЙ ПЛАН

№з/п	Назва етапів (розділів) дипломного проекту (роботи)	Термін виконання етапів проекту (роботи)	Примітка
1	Вибір напряму дослідження та узгодження тематики кваліфікаційної роботи з керівником	10.01.2025	виконано
2	Ознайомлення з предметною областю; формулювання мети та задач дослідження; визначення об'єкта та предмета дослідження	01.02.2025	виконано
3	Робота над розділом 1 – дослідження предметної області та постановка задачі	01.03.2025	виконано
4	Робота над розділом 2 – вибір компонентів для проектування системи адаптивного застосування моніторингових елементів розвідувального БПЛА	01.04.2025	виконано
5	Робота над розділом 3 – проектування системи адаптивного застосування моніторингових елементів розвідувального БПЛА	29.04.2025	виконано
6	Оформлення пояснювальної записки згідно вимог	25.05.2025	виконано
7	Попередній захист ВКР	26.05.2025	виконано
8	Захист ВКР на засіданні ЕК	Червень 2025 року	

Студентка Підпис Дар'я КОРЖОВА Ініціали, прізвище

Керівник роботи Підпис Сергій ЛИСЕНКО Ініціали, прізвище

№ рядка	формат	Позначення	Найменування	Кіл. листів	№ екз	Примітка
			Текстові документи			
1		КвРКІ 210237.21.02.65 ПЗ	Пояснювальна записка	59		
			Графічні матеріали			
2		КвРКІ 210237.21.02.65 Е8	Принцип роботи	1		
3		КвРКІ 210237.21.02.65 Е8	Схема роботи	1		
			програмної частини			
4		КвРКІ 210237.21.02.65 Е8	Принципова схема	1		
			апаратної частини			

КвРКІ 190186.19.01.08 ВП

Зм	Арк	№ докум	Підпис	Дата
Розробив		Коржова		
Перевір.		Лисенко		
Н. контр.		Кисіль		
Затв.		Павлова		

Відомість проекту

Літера	Аркуш	Аркушів
У	1	1

ХНУ, КІ2-21-2

АНОТАЦІЯ

Тема кваліфікаційної роботи: «Програмно-технічний засіб VPN сервера на основі одноплатної комп'ютерної системи Raspberry PI4».

Автор роботи: Дар'я КОРЖОВА.

Керівник роботи: Лисенко Сергій Миколайович.

Пояснювальна записка: 59 с., 26 рис., 2 табл., 3 дод., 52 джерела.

Графічна частина: 3 креслення.

VPN, RASPBERRY PI4, Сервер

Метою дипомної роботи є розробка програмно-технічного засобу VPN сервера з урахуванням можливостей одноплатної комп'ютерної системи Raspberry Pi4.

Об'єктом дослідження є процес створення та функціонування VPN-сервера на базі одноплатної комп'ютерної системи Raspberry Pi 4.

Предметом дослідження є сукупність програмно-технічних засобів, їхня взаємодія, ефективність та особливості реалізації для побудови VPN-сервера на Raspberry Pi 4.

Зважаючи на складність предметної області та різнорідність завдань, у роботі було застосовано комбінований методологічний підхід, що являє собою сукупність найбільш релевантних методів дослідження.

Підпис студента

09.06.2025р

Дата

ЗМІСТ

ВСТУП

У сучасному світі, де інформаційні технології розвиваються стрімко, питання захисту даних і забезпечення безпечного доступу до мережевих ресурсів стає все більш важливим. Щодня мільйони користувачів стикаються з проблемами, пов'язаними з загрозами конфіденційності, витоками особистої інформації, стеженням у публічних мережах, а також потребою в безпечному віддаленому доступі до корпоративних чи домашніх інфраструктур. У цьому контексті VPN (Virtual Private Network) виступає як один з ключових інструментів, що дає змогу створювати захищені канали зв'язку через незахищені мережі.

Водночас, з огляду на популяризацію недорогих, енергоефективних та доступних платформ, зростає інтерес до використання одноплатних комп'ютерів, зокрема Raspberry Pi4, як апаратної основи для створення компактних серверних рішень. Raspberry Pi4 – це універсальний пристрій, який поєднує достатню обчислювальну потужність з мініатюрними розмірами та низьким енергоспоживанням, що робить його привабливим варіантом для домашніх та навіть деяких корпоративних задач, зокрема у сфері побудови VPN-серверів.

Використання Raspberry Pi як бази для VPN-сервера дозволяє створити автономну, мобільну, економічно доступну систему для захищеного віддаленого доступу. Проте, незважаючи на очевидні переваги, така реалізація має ряд обмежень, пов'язаних з продуктивністю, тепловими режимами, ресурсомісткістю програмного забезпечення та складністю налаштування. Зважаючи на це, особливо важливим стає грамотний підхід до оптимізації конфігурацій, а також автоматизації розгортання та обслуговування VPN-системи на цій платформі.

Актуальність даної роботи зумовлена потребою у доступному, ефективному та безпечному VPN-рішенні, яке можливо реалізувати в умовах обмежених ресурсів і без спеціалізованого серверного обладнання. Така потреба особливо відчутна в українських реаліях - як для приватних користувачів, так і для освітніх

закладів, громадських організацій, журналістів, волонтерів, які працюють у зонах з підвищеним ризиком доступу до чутливої інформації.

Метою дипомної роботи є розробка програмно-технічного засобу VPN сервера з урахуванням можливостей одноплатної комп'ютерної системи Raspberry Pi4.

Об'єктом дослідження є процес створення та функціонування VPN-сервера на базі одноплатної комп'ютерної системи Raspberry Pi 4.

Предметом дослідження є сукупність програмно-технічних засобів, їхня взаємодія, ефективність та особливості реалізації для побудови VPN-сервера на Raspberry Pi 4.

Таким чином, дана робота спрямована на розв'язання задачі, що має високу практичну значущість у сучасних умовах цифрової трансформації, зокрема в контексті безпечної роботи з даними на рівні малих офісів, домашніх мереж та мобільних користувачів.

1 ПРОГРАМНО-ТЕХНІЧНІ ЗАСОБИ VPN СЕРВЕРА НА ОСНОВІ ОДНОПЛАТНОЇ КОМП'ЮТЕРНОЇ СИСТЕМИ RASPBERRY PI4 ТА ПОСТАНОВКА ЗАДАЧІ ЩОДО ЇЇ УДОСКОНАЛЕННЯ

1.1 Аналіз предметної області і виявлення наявних проблем і завдань

В умовах інтенсивного розвитку інформаційних технологій питання забезпечення безпеки даних набуває особливої актуальності. Це має ключове значення для збереження приватної інформації, гарантування конфіденційності в публічних мережах, а також для обмеження доступу до приватних або корпоративних мережевих ресурсів. Одним з найбільш ефективних рішень для цих завдань є використання віртуальної приватної мережі (VPN), яка формує зашифрований канал передачі даних між клієнтським пристроєм та мережею, забезпечуючи захист трафіку від зовнішніх загроз та уможливлюючи обхід регіональних обмежень доступу до інформації (принцип функціонування якої ілюструє рисунок 1.1).

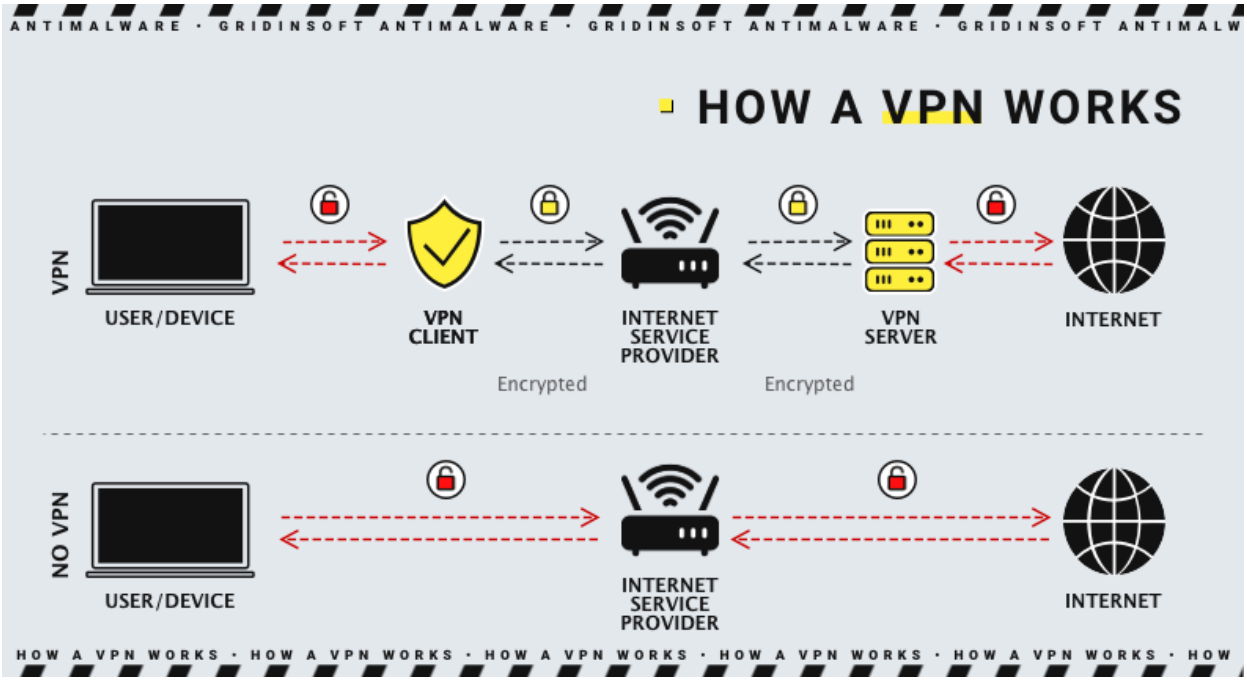


Рисунок 1.1 – Як працює VPN [1]

Останнім часом спостерігається зростання популярності одноплатних комп'ютерів, зокрема Raspberry Pi 4, як апаратної платформи для створення VPN-серверів. Дана апаратна платформа характеризується енергоефективністю, компактними розмірами, низькою вартістю та достатньою продуктивністю для вирішення основних мережевих завдань, зокрема для створення мобільних або автономних VPN-серверів. Такий підхід є особливо актуальним в Україні для задоволення потреб домашніх користувачів, представників освітньої сфери, журналістів, волонтерських організацій та військовослужбовців.

Мета дослідження охоплює аналіз, розробку та практичне застосування програмних (операційна система, VPN-серверне програмне забезпечення, засоби адміністрування) та апаратних (конфігурація мережі, забезпечення безпеки, оптимізація продуктивності та енергоспоживання) компонентів. Детальний аналіз предметної області дозволяє ідентифікувати низку актуальних проблем та сформулювати відповідні завдання дослідження.

Апаратною основою для розроблюваного сервера слугує Raspberry Pi 4 (зовнішній вигляд плати з основними компонентами показано на рисунку 1.2). Для успішної реалізації поставлених завдань необхідний аналіз її технічних можливостей. Це передбачає детальне вивчення та оцінку технічних характеристик платформи, її обчислювальної продуктивності, максимальної кількості одночасних підключень (що є критичним параметром при експлуатації сервера у великих мережевих інфраструктурах) та показників енергоефективності. Окремої уваги потребує проблема перегріву Raspberry Pi 4 при тривалих інтенсивних навантаженнях, що зумовлює необхідність розробки та впровадження ефективної системи охолодження для забезпечення стабільного функціонування пристрою.

На сучасному ринку представлений широкий спектр програмних рішень для VPN-серверів. Значна частина цих рішень розповсюджується за моделлю відкритого програмного коду, що надає можливість для детального аналізу та порівняння їхніх функціональних характеристик. Це, у свою чергу, сприятиме обґрунтованому вибору оптимального програмного забезпечення або розробці

кастомізованого рішення, що поєднуватиме переваги різних підходів у забезпеченні безпеки, продуктивності, простоти конфігурування та підтримки необхідних протоколів.

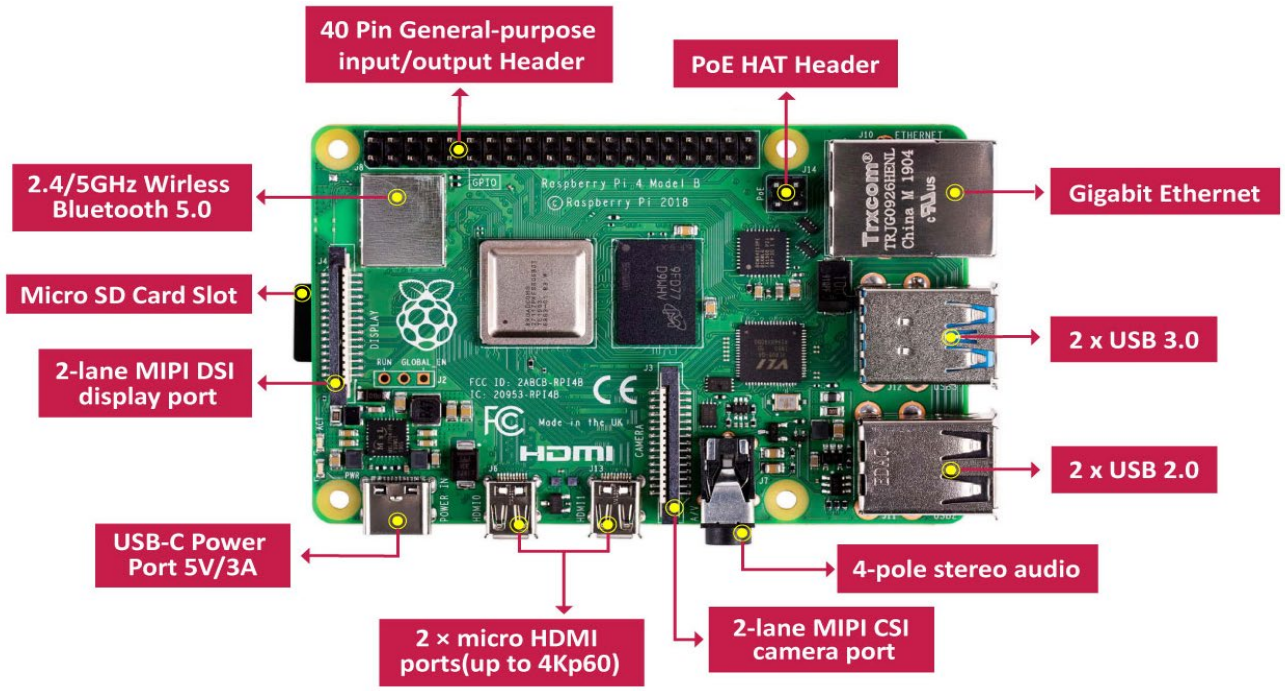


Рисунок 1.2 – Плата Raspberry Pi4 [2]

Важливим етапом є вибір оптимальної операційної системи для Raspberry Pi 4, сумісної з обраним VPN-серверним програмним забезпеченням. Пріоритетними кандидатами є дистрибутиви на базі Linux, такі як Raspberry Pi OS та Ubuntu Server. Обрана операційна система потребуватиме подальшого налаштування для забезпечення належного рівня безпеки, конфігурації мережевих параметрів та ефективного управління системними ресурсами.

Окремої уваги потребує конфігурація мережевих параметрів. Це включає налаштування мережевих інтерфейсів Raspberry Pi 4. Доцільним є також дослідження питань, пов'язаних із використанням динамічних IP-адрес та технології Dynamic DNS (DDNS), принцип роботи якої ілюструє рисунок 1.3.

Забезпечення високого рівня безпеки є ключовою вимогою до VPN-сервера. Це передбачає вибір стійких протоколів шифрування та алгоритмів автентифікації,

а також конфігурування брандмауера (firewall) для захисту сервера. Необхідно також дослідити та впровадити методи протидії несанкціонованому доступу та DDoS-атакам, а також реалізувати надійні механізми автентифікації та авторизації користувачів.

Система повинна надавати користувачам можливість управління власними обліковими записами, включно з їх створенням, видаленням та зміною параметрів автентифікації. Для адміністраторів сервера необхідно забезпечити інструменти моніторингу стану системи, систему сповіщень про інциденти, а також засоби управління процесами, проведення діагностики та встановлення оновлень програмного забезпечення.

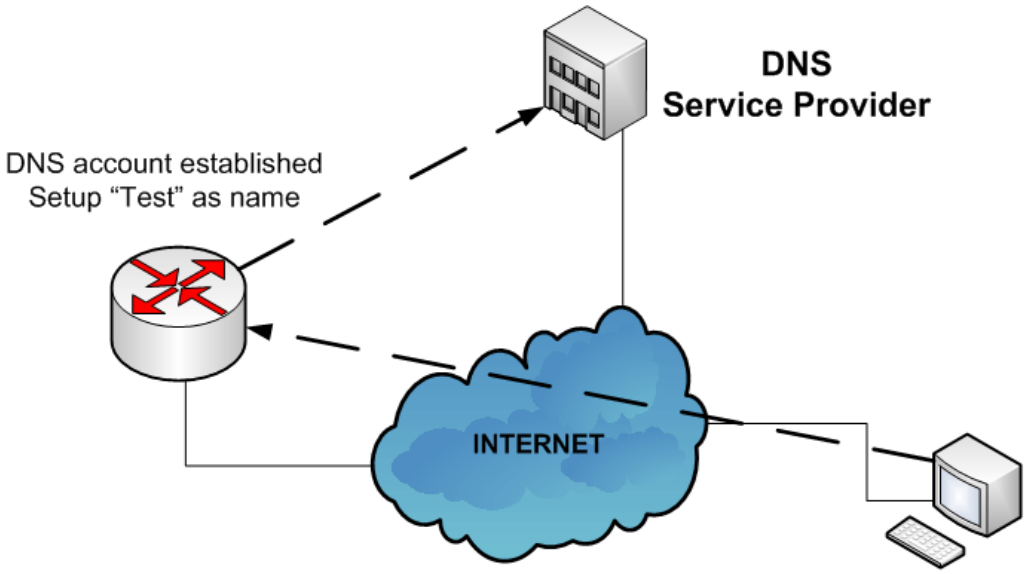


Рисунок 1.3 – Принцип роботи Dynamic DNS [3]

Таким чином, центральним завданням дослідження є розробка ефективного VPN-рішення на платформі Raspberry Pi 4, яке б враховувало технічні обмеження даної платформи та відповідало сучасним вимогам щодо безпеки та доступності. Кінцевою метою є досягнення оптимального балансу між рівнем захищеності, продуктивністю, простотою конфігурування та експлуатації, а також економічною

ефективністю для задоволення потреб широкого кола користувачів, від приватних осіб до елементів критичної інфраструктури.

1.2 Порівняльний аналіз переваг та недоліків існуючих рішень

Розвиток глобальної мережі Інтернет супроводжується зростанням попиту користувачів на ефективні засоби захисту персональних даних та забезпечення конфіденційності. Це стимулювало розробку та впровадження значної кількості VPN-сервісів, інфраструктура яких налічує тисячі серверів. Сукупна кількість серверів, що експлуатуються провайдерами VPN-послуг, включно з малими сервісами та корпоративними мережами, оцінюється у понад мільйон одиниць. Для розгортання VPN-серверів у локальних або малих мережах, рішення на базі одноплатних комп'ютерів (ОПК) представляють значний інтерес завдяки їхній економічній ефективності, компактності та доступності.

Найбільш поширеними програмними реалізаціями VPN є OpenVPN, WireGuard та IPsec. Також заслуговує на увагу комбінація протоколів L2TP/IPsec, схематичний принцип роботи якої ілюструє рисунок 1.4. Незважаючи на відносну простоту конфігурування, дане рішення характеризується як менш надійне та певною мірою застаріле порівняно з альтернативними протоколами.

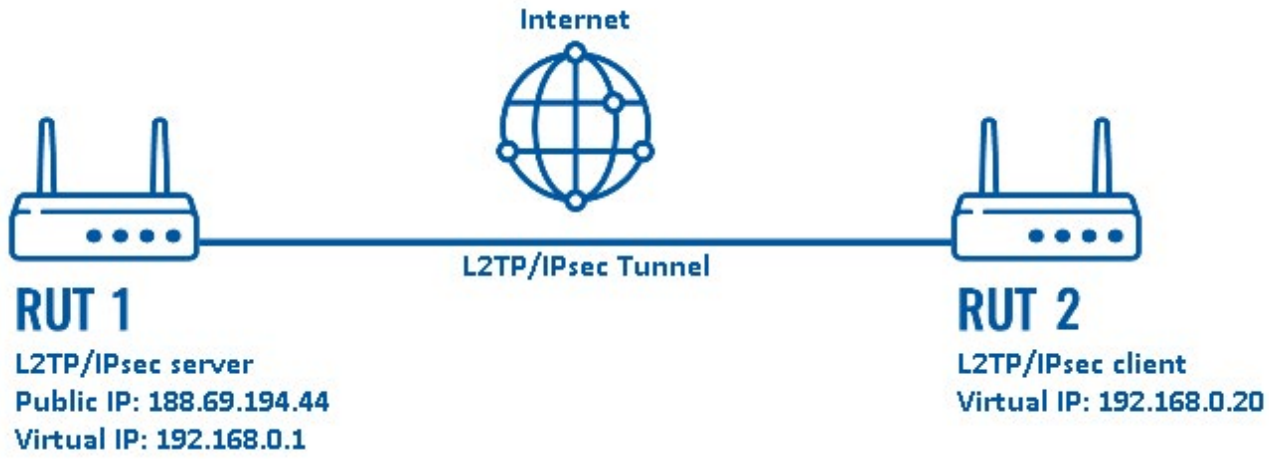


Рисунок 1.4 – Принцип роботи L2TP/IPsec [4]

Кожен із зазначених протоколів володіє специфічними характеристиками, що обумовлюють доцільність його застосування у певних операційних середовищах. Детальний аналіз їхніх переваг та недоліків, з подальшим узагальненням, має на меті ідентифікацію потенційних вразливостей та обмежень VPN-рішень на базі Raspberry Pi 4.

OpenVPN (принцип функціонування якого ілюструє рисунок 1.5) традиційно розглядається як один із провідних VPN-протоколів. Він характеризується високим рівнем безпеки, підтвердженим тривалим періодом експлуатації, та універсальністю, що дозволяє його застосування як у корпоративних, так і в домашніх мережах. Процес конфігурації з боку користувача оцінюється як середньої складності, вимагаючи певного рівня знань у галузі мережевих технологій, що може становити труднощі для непідготовлених користувачів. До недоліків протоколу відносять потенційне зниження продуктивності при використанні на платформах з обмеженими ресурсами, таких як Raspberry Pi, а також ускладнене проходження NAT (Network Address Translation) та відносно тривалий час ініціалізації тунелю, що може бути критичним для певних сценаріїв використання.

Незважаючи на зазначені обмеження, OpenVPN залишається високофункціональним рішенням. Функціонування через порт 443 (стандартний для HTTPS) забезпечує можливість маскування трафіку під звичайний веб-трафік, що ускладнює його ідентифікацію та блокування. Забезпечується високий рівень криптографічного захисту завдяки використанню бібліотеки OpenSSL та підтримці сучасних алгоритмів і протоколів, таких як AES-256, TLS та HMAC. До переваг протоколу також належать широка кросплатформність (підтримка операційних систем Windows, Linux, macOS, Android, iOS, а також мережевих маршрутизаторів) та значна гнучкість конфігурування, включно з підтримкою протоколів TCP/UDP, різних типів маршрутизації та статичних маршрутів.

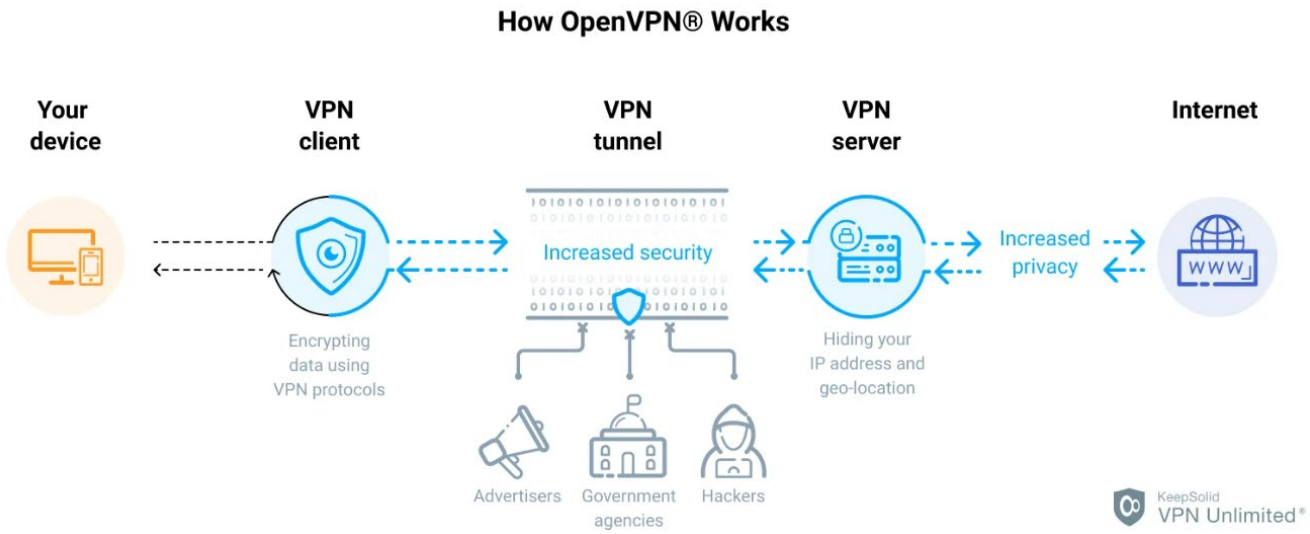


Рисунок 1.5 – OpenVPN [5]

Альтернативою є протокол WireGuard (принцип роботи якого ілюструє рисунок 1.6), що характеризується значно простішою архітектурою. Даний сучасний VPN-протокол розроблений "з нуля" та відзначається мінімалістичною кодовою базою (близько 4000 рядків коду порівняно з приблизно 100 000 рядків у OpenVPN). Це сприяє зменшенню ймовірності виникнення помилок та вразливостей, а також спрощує їх ідентифікацію та усунення. За показниками продуктивності WireGuard часто демонструє переваги, зокрема менші затримки та вищу швидкість передачі даних. Конфігурація клієнта здійснюється за допомогою єдиного файлу, а також реалізовано ефективнішу підтримку NAT traversal. Зазначені особливості забезпечують ефективне функціонування WireGuard на пристроях з обмеженими обчислювальними ресурсами.

Простота архітектури та відносна новизна протоколу визначають як його переваги, так і певні обмеження. До недоліків можна віднести обмежену підтримку на деяких платформах, меншу кількість доступних параметрів конфігурування та відсутність механізмів динамічної IP-адресації та автентифікації на основі центрів сертифікації (CA). Водночас, протокол перебуває на стадії активного розвитку, що передбачає його подальше вдосконалення.

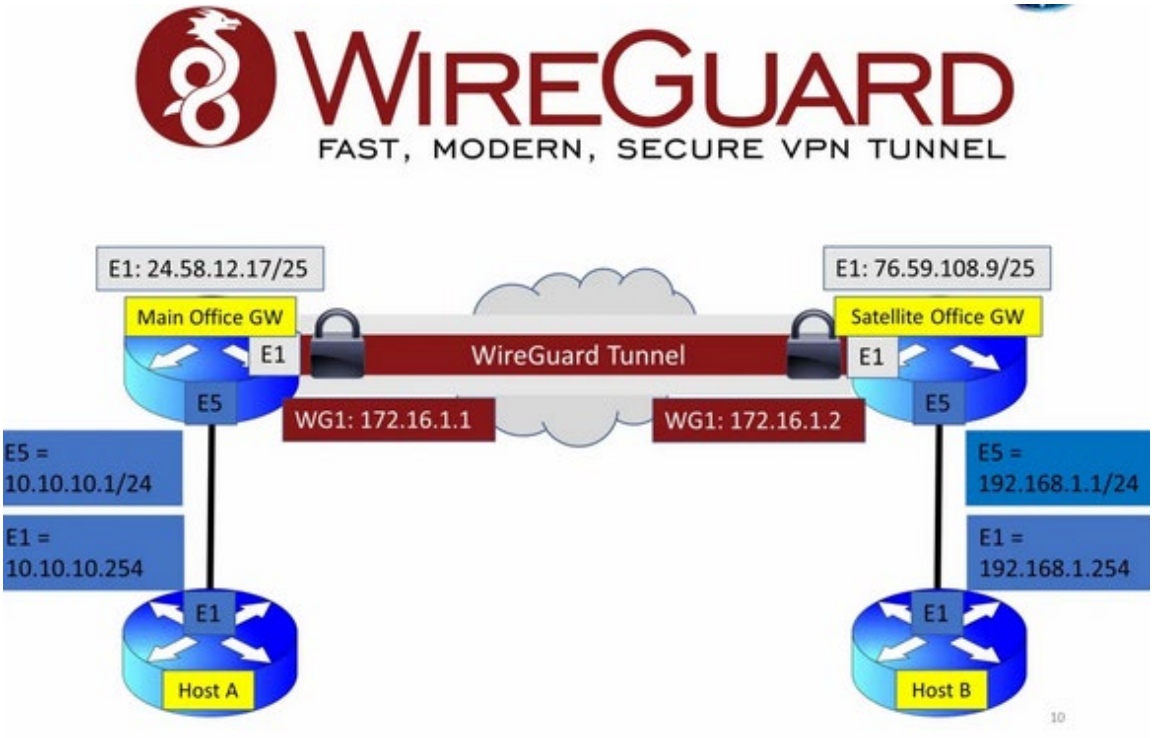


Рисунок 1.6 – WireGuard [6]

Ще одним важливим набором протоколів є IPsec (принцип роботи якого для сценарію site-to-site VPN ілюструє рисунок 1.7), що являє собою надійне та перевірене часом рішення для створення захищених VPN-з'єднань. IPsec інтегрований на рівні операційних систем, таких як Windows, Android та macOS, забезпечуючи нативну підтримку. Він забезпечує високу стабільність довготривалих з'єднань та потужні механізми безпеки, включно з підтримкою алгоритмів шифрування AES, хешування SHA2, груп Діффі-Геллмана (DH), еліптичної кривої Curve25519, а також методів автентифікації за допомогою сертифікатів, попередньо узгоджених ключів (PSK) та протоколу EAP.

Незважаючи на гнучкість, конфігурування IPsec є складним процесом, що вимагає глибоких технічних знань та ретельного налаштування параметрів, що може бути перешкодою для деяких користувачів. Функціонування IPsec на пристроях з обмеженою продуктивністю, таких як Raspberry Pi, може бути менш ефективним за відсутності апаратного прискорення криптографічних операцій. На відміну від OpenVPN, трафік IPsec не маскується під стандартний веб-трафік, що підвищує ймовірність його блокування в мережах з обмеженим доступом.

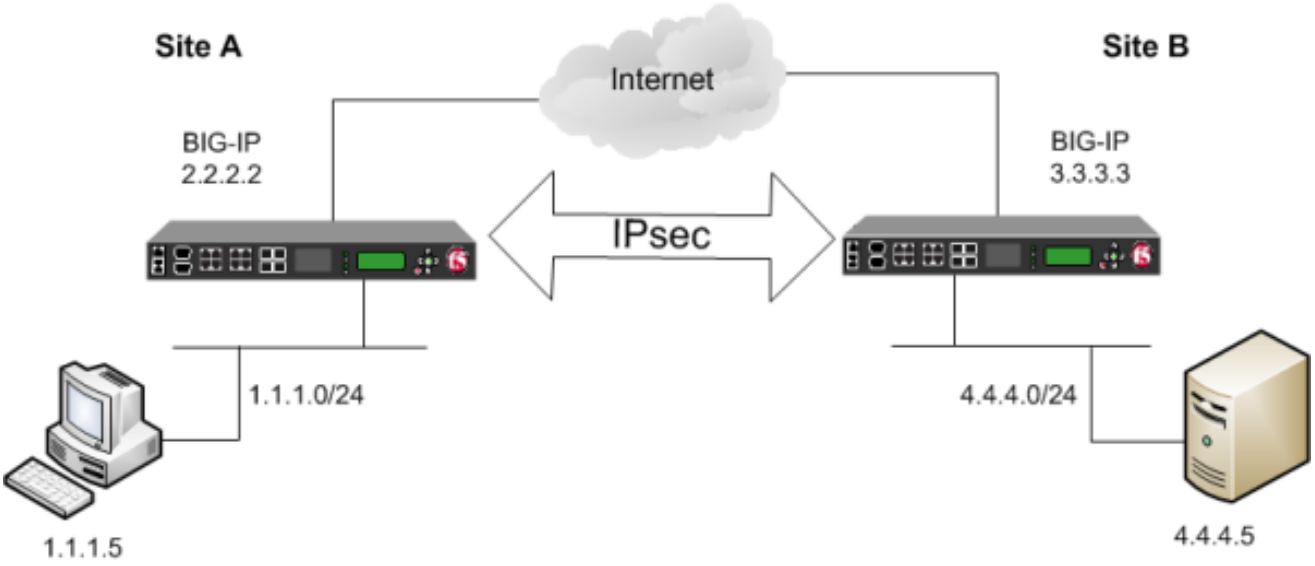


Рисунок 1.7 – Як працює IPsec

Узагальнення основних характеристик розглянутих протоколів представлено у вигляді порівняльної таблиці 1.1.

Таблиця 1.1 - Порівняльна характеристика протоколів

Характеристика	OpenVPN	WireGuard	IPsec
Продуктивність	Середня	Висока	Середня/Низька
Безпека	Висока	Висока	Висока
Налаштування	Складне	Дуже просте	Дуже складне
Маскування	Відмінне	Середнє	Обмежене
Швидкість	Середня	Дуже висока	Висока
Аудит коду	Високий	Спрощений	Складний
Кросплатформність	Відмінна	Помірна	Висока

Апаратна платформа Raspberry Pi 4 забезпечує можливість реалізації VPN-сервера з використанням різних протоколів; вибір конкретного програмного рішення визначається пріоритетними вимогами, такими як продуктивність, рівень безпеки, простота розгортання та експлуатації, або масштабованість.

Таким чином, проведений аналіз свідчить про відсутність універсального VPN-рішення, здатного повною мірою задовольнити увесь спектр потенційних вимог користувачів. Кожен із розглянутих протоколів має певні обмеження та недоліки, що можуть впливати на вибір користувача залежно від конкретних умов та завдань.

1.3 Підходи до вирішення задачі за темою дослідження

Вибір методології дослідження визначає планування подальших етапів роботи над обраною темою та структурування отриманої інформації. Він ґрунтується на меті роботи – розробці ефективного, безпечного та зручного у використанні VPN-сервера на базі одноплатного комп’ютера Raspberry Pi 4.

Зважаючи на складність предметної області та різнорідність завдань, у подальшій роботі буде застосовано комбінований методологічний підхід, що являє собою сукупність найбільш релевантних методів дослідження. Далі буде послідовно розглянуто та обґрунтовано кожен із компонентів обраного підходу.

Основним компонентом є системний підхід. Він передбачає розгляд VPN-сервера як інтегрованого елемента ширшої інформаційної системи. У цьому контексті Raspberry Pi 4 виступає апаратною платформою, програмне забезпечення – функціональним модулем, а користувач – ключовим суб'єктом взаємодії. На даному етапі зазначені компоненти розглядатимуться у взаємозв'язку з іншими елементами системи.

З метою підвищення гнучкості, масштабованості та спрощення технічного обслуговування системи буде використано модульний підхід. Компоненти VPN-сервера, зокрема система управління ключами, підсистема моніторингу та механізми захисту, будуть представлені як окремі логічні модулі. Такий підхід дозволить, за необхідності, здійснювати модифікацію окремих частин системи без порушення її загальної працездатності.

Для розробки рішення ідентифікованої проблеми, а саме відсутності універсального та оптимізованого механізму конфігурування VPN-сервера для цільової платформи, будуть застосовані методи аналізу та синтезу. Ключовим елементом аналізу є дослідження наявних програмних рішень, результати якого представлено у попередньому розділі. Завданням синтезу є вибір та інтеграція VPN-протоколу, який демонструватиме переваги за критеріями продуктивності, зручності конфігурування та швидкодії, а також забезпечуватиме оптимальну сумісність з апаратною платформою Raspberry Pi 4, відповідно до мети дослідження.

Для забезпечення коректної роботи та подальшого впровадження VPN-сервера на базі одноплатної комп'ютерної системи Raspberry Pi 4 необхідним є його тестування. Дане завдання реалізується шляхом застосування методу експериментального дослідження. Цей етап передбачає тестування розробленого прототипу в умовах, що імітують реальну експлуатацію. Планується розгортання тестового середовища для виявлення можливих суттєвих недоліків та помилок конфігурації. У процесі експерименту будуть вимірюватися ключові характеристики, такі як швидкість передачі даних, температурні показники пристрою під навантаженням, стабільність з'єднання та рівень завантаження центрального процесора. Доцільним є проведення порівняльних випробувань різних конфігурацій (наприклад, з використанням протоколів OpenVPN та WireGuard) для визначення напрямків подальшої оптимізації прототипу.

Після успішного налагодження функціонування сервера в режимі ручного адміністрування, буде застосовано метод автоматизації. Розроблені та верифіковані скрипти автоматизації (наприклад, на Bash або Python), стандартизовані шаблони конфігурацій та механізми автоматичного запуску сервісів дозволять мінімізувати вплив людського фактору та знизити ймовірність операційних помилок при розгортанні та експлуатації сервера.

1.4 Постановка задачі

На основі проведеного аналізу предметної області, компаративного аналізу існуючих рішень та визначених методологічних підходів, сформульовано основну мету дослідження – розробка програмно-технічного рішення для VPN-сервера на базі Raspberry Pi 4, що характеризуватиметься оптимальним поєднанням безпеки, продуктивності та простоти впровадження.

На поточному етапі дослідження визначено низку послідовних завдань, деталізація яких наведена нижче.

Першочерговим завданням є повторний аналіз наявних програмних рішень для створення VPN-серверів, з розширенням переліку досліджуваних альтернатив, з метою всебічного дослідження можливих конфігурацій за ключовими критеріями (продуктивність, безпека, простота конфігурування), неодноразово зазначеними в роботі. При цьому необхідним є врахування специфікацій апаратної платформи Raspberry Pi 4, зокрема її обчислювальних та ресурсних обмежень.

Наступним кроком, на основі отриманих результатів аналізу, є обґрунтування вибору VPN-протоколу для серверної реалізації. Це, у свою чергу, уможливить детальне визначення вимог до апаратних ресурсів, що забезпечать стабільне та ефективне функціонування VPN-сервера. До таких елементів належать обсяг оперативної пам'яті, тип системи живлення, компоненти системи охолодження, мережеві інтерфейси та інші.

Після виконання зазначених завдань здійснюється перехід до розробки архітектури програмно-технічного комплексу VPN-сервера. Даний етап передбачає визначення ключових функціональних компонентів, таких як модуль криптографічного захисту, система моніторингу стану та підсистема управління криптографічними ключами.

Наступними взаємопов'язаними завданнями є безпосередня практична реалізація серверного рішення та його подальше тестування в умовах, наближених до реальної експлуатації. Даний етап включає тестування за умов стандартного та

пікового навантаження для виявлення потенційних вразливостей та помилок конфігурації, здатних призвести до відмови системи. Це також дозволить здійснити кількісну оцінку пропускної здатності сервера, стабільності з'єднання, проаналізувати використання ресурсів та оцінити швидкість ініціалізації тунелю.

Окремим підзавданням, що випливає з попереднього етапу та визначає практичну значущість отриманих результатів, є верифікація рівня безпеки та захищеності розробленої системи. Це включає аналіз конфігурації механізмів шифрування, процедур автентифікації користувачів та стійкості до несанкціонованого доступу.

З огляду на потенційні труднощі, що виникають у користувачів при конфігуруванні аналогічних систем, передбачається розробка детальних інструкцій з розгортання та експлуатації запропонованого рішення. Вимогою до зазначених інструкцій є надання чіткої, покрокової інформації, викладеної доступною мовою без надмірного використання вузькоспеціалізованої термінології. Це забезпечить можливість використання системи широким колом користувачів, включно з тими, хто не володіє глибокими технічними знаннями або перебуває на початковому етапі їх освоєння.

Таким чином, визначений комплекс завдань охоплює як теоретичний аналіз, так і практичну реалізацію проєкту, забезпечуючи комплексний підхід до вирішення науково-практичної проблеми, визначеної в дослідженні.

1.5 Висновки до першого розділу

У першому розділі було проведено детальне дослідження проблематики функціонування VPN-серверів, ідентифіковано ключові вимоги та перспективи їх розвитку, що уможливило визначення основних проблемних аспектів у процесі їх використання.

У рамках даного дослідження також було здійснено поглиблений аналіз сучасних VPN-протоколів, зокрема OpenVPN, WireGuard та IPsec. Компаративний

аналіз таких параметрів, як продуктивність, рівень безпеки, швидкість передачі даних, сумісність з різними операційними системами та доцільність застосування в умовах обмежених апаратних ресурсів, мав суттєвий вплив на процес формулювання завдань для подальшої практичної реалізації.

Було також досліджено технічні характеристики апаратної платформи Raspberry Pi 4, оскільки дана розробка є відносно новою та характеризується підвищеною продуктивністю центрального процесора і збільшеним об'ємом оперативної пам'яті, що забезпечує можливість функціонування ресурсомістких програмних застосунків та операційних систем.

У результаті проведених дослідницьких операцій було чітко сформульовано подальші завдання, що підлягають вирішенню в процесі написання дипломної роботи. Даний перелік завдань слугуватиме методологічною основою для подальшої науково-дослідної діяльності та сприятиме концентрації уваги на пріоритетних напрямках дослідження, що є важливим підґрунтям для початку розробки практичної частини роботи. Розроблена методологія також забезпечить дотримання наміченого плану та, за потреби, дозволить інтегрувати нові підзавдання з метою вдосконалення майбутнього проєкту.

2 ОБҐРУНТУВАННЯ ВИБОРУ КОМПОНЕНТІВ ТА СЕРЕДОВИЩА РЕАЛІЗАЦІЇ

2.1 Апаратне середовище реалізації

Сучасні потреби у безпечному переданні даних у комп'ютерних мережах сприяли широкому застосуванню технології віртуальних приватних мереж (VPN). Її можна реалізувати на різних апаратних платформах, проте найпопулярнішим рішенням є одноплатні комп'ютери. Серед них лідером вважається Raspberry Pi 4 Model B.

У цьому розділі детально проаналізовано апаратне забезпечення, необхідне та достатнє для створення потужного та ефективного VPN-сервера на основі цієї платформи. Розуміння компонентів, їх характеристик та взаємодій важливе для усвідомлення потенціалу, обмежень та оптимальних умов роботи цих систем.

Основа апаратного середовища - це власне одноплатний комп'ютер Raspberry Pi 4 Model B. Він побудований на системі-на-кристалі (SoC) Broadcom BCM2711, що містить чотирьохядерний процесор ARM Cortex-A72. Тактова частота процесора варіюється від 1,5 ГГц до 1,8 ГГц у пізніших версіях. Архітектура Cortex-A72 забезпечує значний приріст продуктивності у порівнянні з попередніми поколіннями Raspberry Pi. Це особливо актуально при виконанні обчислювально складних задач, таких як шифрування та дешифрування трафіку, що критично важливо для роботи VPN.

Чотири фізичні ядра дозволяють ефективно обробляти численні одночасні запити від VPN-клієнтів, забезпечуючи стабільність сервера при підключенні кількох користувачів одночасно. Вбудований графічний процесор VideoCore VI SoC, хоча і не є безпосередньо ядром VPN-сервера, відповідає за загальну функціональність системи, наприклад, налаштування та керування через графічний інтерфейс користувача операційної системи.

Оперативна пам'ять відіграє важливу роль в апаратному забезпеченні. Raspberry Pi 4 Model B доступний у конфігураціях SDRAM LPDDR4 з об'ємом 2

ГБ, 4 ГБ та 8 ГБ. Вибір потрібного об'єму залежить від очікуваного навантаження на VPN-сервер.

Для невеликої кількості підключень (до 5-10 клієнтів) та використання VPN-протоколів, що не вимагають значних ресурсів (наприклад, OpenVPN або WireGuard з базовими налаштуваннями шифрування), може бути достатньо моделі з 2 ГБ оперативної пам'яті. Проте, для більш комфортної роботи, можливості запускати інші сервіси на Raspberry Pi та підтримки більшої кількості клієнтів або ресурсомістких налаштувань VPN, рекомендується вибирати модель з 4 ГБ або 8 ГБ оперативної пам'яті.

Великий обсяг оперативної пам'яті покращує кешування даних, зменшує звернення до постійного сховища та прискорює обробку запитів, що позитивно впливає на загальну продуктивність та швидкість VPN-сервера.

Мережеві можливості Raspberry Pi 4 мають велике значення для функціонування VPN-сервера. Пристрій оснащений портом Gigabit Ethernet (RJ45), який забезпечує швидке та стабільне дротове з'єднання з локальною мережею та Інтернетом. Пропускна здатність мережевого інтерфейсу безпосередньо впливає на максимальну швидкість передачі даних через VPN-тунель.

Фактична пропускна здатність Gigabit Ethernet на Raspberry Pi 4 близька до теоретичного максимуму, що дозволяє обслуговувати клієнтів на високих швидкостях, обмежених лише швидкістю інтернет-каналу та обчислювальною потужністю процесора, що використовується для шифрування.

Окрім дротового підключення, Raspberry Pi 4 має вбудований дводіапазонний модуль Wi-Fi (IEEE 802.11ac), що підтримує частоти 2,4 ГГц та 5 ГГц, а також Bluetooth 5.0. Хоча дротове з'єднання Ethernet рекомендується як основний спосіб підключення VPN-сервера до мережі, Wi-Fi може бути використаний для початкового налаштування пристрою або в ситуаціях, коли дротове підключення недоступне.

Система зберігання даних на Raspberry Pi 4 традиційно організовується з допомогою карти пам'яті microSD. На цю карту записується операційна система,

програмне забезпечення для VPN-сервера, файли налаштувань та лог-файли. Підбір microSD карти є важливим етапом. Рекомендується застосовувати карти від відомих виробників з високим класом швидкості (наприклад, Class 10, UHS-I U3 або Application Performance Class A1/A2) для забезпечення швидкого старту системи та оперативного опрацювання файлів.

Об'єм карти залежить від потреб: для простої установки операційної системи та VPN-сервера може вистачити 16 ГБ або 32 ГБ, але для зберігання логів протягом тривалого часу та можливості встановлення додаткового ПЗ краще обрати карту на 64 ГБ або більше.

Важливо розуміти, що microSD карти мають обмежений ресурс перезаписів, тому для інтенсивного логування або частого запису даних доцільно розглянути можливість підключення зовнішнього USB-накопичувача (SSD чи HDD) для зберігання даних, що часто змінюються, залишаючи microSD карту переважно для завантаження операційної системи.

Raspberry Pi 4 обладнаний двома портами USB 3.0 та двома портами USB 2.0. Порти USB 3.0 забезпечують значно більшу швидкість передачі даних, що робить їх ідеальними для підключення швидкісних зовнішніх накопичувачів. Використання зовнішнього SSD через USB 3.0 може суттєво збільшити загальну продуктивність системи та надійність зберігання даних в порівнянні з microSD картою.

Живлення Raspberry Pi 4 реалізується через порт USB Type-C. Важливо застосовувати якісний блок живлення, що забезпечує стабільну напругу 5В та струм не менше 3А. Недостатнє або нестабільне живлення може призвести до збоїв у роботі, перезавантажень та навіть пошкодження файлової системи на microSD карті. Тому вибір надійного джерела живлення критичний для стабільної роботи VPN-сервера в режимі 24/7.

Для забезпечення стабільної роботи Raspberry Pi 4 під навантаженням, типовим для VPN-сервера (постійне шифрування/дешифрування даних), потрібно приділити увагу системі охолодження. Процесор BCM2711 може нагріватися,

особливо при поганій вентиляції або інтенсивних обчисленнях. Перегрів може призвести до спрацьовування механізму тротлінгу, коли процесор автоматично знижує свою тактову частоту для зменшення температури, що негативно впливає на продуктивність VPN-сервера.

Отже, рекомендується застосування пасивного охолодження у вигляді радіаторів, які встановлюються на SoC та інші мікросхеми, що можуть нагріватися. Для більш інтенсивних навантажень або експлуатації в умовах підвищеної температури навколишнього середовища доцільним є використання активного охолодження, наприклад, корпусу з вбудованим вентилятором або окремого вентилятора, який підключається до GPIO-пінів Raspberry Pi.

Правильно організоване охолодження дозволяє підтримувати оптимальний температурний режим, уникнути тротлінгу та забезпечувати максимальну продуктивність процесора протягом тривалого часу.

Додаткове периферійне обладнання для функціонування VPN-сервера, як правило, не потрібне після початкового налаштування. Для первинної інсталяції операційної системи та конфігурації може знадобитися клавіатура, миша та монітор, які підключаються відповідно через USB-порти та один з двох портів micro-HDMI, якими обладнаний Raspberry Pi 4.

Після завершення налаштування VPN-сервер може працювати в режимі "безголового" режиму (headless mode), тобто без підключеного монітора, клавіатури та миші, а керування ним реалізується віддалено через мережу, наприклад, за допомогою протоколу SSH

Розглядаючи апаратне забезпечення для VPN-сервера на Raspberry Pi 4, важливо також враховувати характеристики мережевої інфраструктури, до якої він буде підключений. Швидкість та стабільність інтернет-каналу є визначальними факторами для продуктивності VPN-з'єднання.

Навіть найпотужніший сервер не зможе забезпечити високу швидкість, якщо інтернет-з'єднання повільне або нестабільне. Крім того, маршрутизатор, до якого підключений Raspberry Pi, повинен підтримувати функцію перенаправлення портів

(port forwarding), щоб зовнішні VPN-клієнти могли встановлювати з'єднання з сервером, що знаходиться в локальній мережі.

Порівнюючи Raspberry Pi 4 з іншими одноплатними комп'ютерами, слід відмітити, що саме ця модель пропонує одне з найкращих співвідношень ціни, продуктивності та енергоефективності для задач VPN-сервера невеликого масштабу. Велика спільнота користувачів та розробників, а також доступність великої кількості програмного забезпечення та інструкцій, роблять процес налаштування відносно простим навіть для користувачів з обмеженим досвідом.

Хоча є потужніші одноплатні комп'ютери, вони часто мають вищу вартість або менш розвинену програмну підтримку. Для домашнього використання або невеликого офісу, де не потрібна підтримка сотень одночасних підключень, Raspberry Pi 4 з відповідним об'ємом оперативної пам'яті та належним охолодженням є цілком адекватним апаратним рішенням. Його низьке енергоспоживання також є значною перевагою, дозволяючи експлуатувати сервер цілодобово без значних витрат на електроенергію.

Підсумовуючи аналіз апаратного забезпечення, можна стверджувати, що одноплатний комп'ютер Raspberry Pi 4 Model B є життєздатною та економічно ефективною платформою для реалізації програмно-технічних засобів VPN-сервера. Його чотирьохядерний процесор ARM Cortex-A72, достатній об'єм оперативної пам'яті LPDDR4 (особливо в конфігураціях 4 ГБ та 8 ГБ), гігабітний Ethernet-порт та підтримка швидких USB 3.0 накопичувачів створюють необхідну апаратну базу.

Ключовими факторами для успішної реалізації є правильний вибір об'єму оперативної пам'яті, використання швидкої та надійної microSD карти (або зовнішнього SSD), забезпечення адекватного живлення та ефективного охолодження. При дотриманні цих умов Raspberry Pi 4 здатен забезпечити стабільну та безпечну роботу VPN-сервера, задовольняючи потреби значної категорії користувачів у захищеному доступі до мережевих ресурсів.

Ретельний підхід до вибору та конфігурації апаратних компонентів дозволить максимально розкрити потенціал цієї компактної платформи для створення надійного вузла віртуальної приватної мережі.

2.2 Функційні вимоги

Визначення функціональних вимог до програмно-технічних засобів VPN-сервера, що реалізовується на основі одноплатного комп'ютера Raspberry Pi 4, є основоположним етапом проектування, який закладає фундамент для розробки, тестування та подальшого функціонування системи.

Функціональні вимоги визначають конкретні дії, операції та завдання, які система зобов'язана виконувати, а також взаємодію системи з користувачами та іншими системами. Для VPN-сервера на специфічній платформі, як-от Raspberry Pi 4, ці вимоги повинні не лише охоплювати стандартний набір функцій віртуальної приватної мережі, але й брати до уваги апаратні обмеження та потенційні сценарії використання пристрою.

Однією з ключових функціональних вимог є забезпечення безпечного тунелювання мережевого трафіку. Це означає, що програмно-технічні засоби мають бути здатні створювати зашифровані канали зв'язку між VPN-клієнтами та сервером.

Дані, що передаються через ці тунелі, повинні бути надійно захищені від несанкціонованого доступу, перехоплення та зміни третіми особами. Система мусить підтримувати сучасні та перевірені криптографічні протоколи тунелювання.

Наприклад, обов'язковою є підтримка протоколу OpenVPN, який є галузевим стандартом де-факто завдяки своїй гнучкості, високому рівню безпеки та доступності клієнтського програмного забезпечення для широкого спектра операційних систем. Разом з тим, з огляду на обмежені обчислювальні ресурси

Raspberry Pi 4, функціональною вимогою може бути також підтримка новітнього протоколу WireGuard.

WireGuard відомий своєю простотою, високою продуктивністю та меншим споживанням ресурсів у порівнянні з OpenVPN, що робить його особливо привабливим для використання на одноплатних комп'ютерах. Реалізація обох протоколів надасть користувачеві вибір залежно від конкретних потреб у безпеці та швидкодії. Функція тунелювання повинна гарантувати інкапсуляцію IP-пакетів та їх безпечне транспортування через публічні мережі, на кшталт Інтернету.

Наступною важливою функціональною вимогою є автентифікація користувачів та пристроїв. Система повинна надавати надійні механізми для перевірки ідентичності клієнтів, які намагаються підключитися до VPN-сервера. Це потрібно для запобігання несанкціонованому доступу до віртуальної приватної мережі та ресурсів, які вона захищає.

Функціонал автентифікації має підтримувати різні методи. Одним з основних методів є використання цифрових сертифікатів на базі інфраструктури відкритих ключів (PKI). Цей підхід забезпечує високий рівень безпеки, оскільки кожен клієнт та сервер мають унікальні сертифікати та приватні ключі. Система повинна дозволяти генерувати, відкликати та управляти цими сертифікатами.

Іншим можливим методом автентифікації є використання попередньо узгоджених ключів (Pre-Shared Keys, PSK), який є простішим у налаштуванні, але менш масштабованим та безпечним для великої кількості користувачів. Також може бути передбачена можливість автентифікації за допомогою логіна та пароля, можливо, в поєднанні з іншими методами для двофакторної автентифікації, що значно підвищує рівень захисту облікових записів.

Забезпечення конфіденційності та цілісності даних є невід'ємною функціональною вимогою. Програмно-технічні засоби VPN-сервера повинні використовувати сильні алгоритми шифрування для захисту даних, що передаються. Необхідна підтримка сучасних симетричних та асиметричних алгоритмів шифрування, як-от AES (Advanced Encryption Standard) з різною

довжиною ключа (наприклад, AES-256), ChaCha20 (особливо актуально для WireGuard).

Окрім конфіденційності, система повинна гарантувати цілісність даних, тобто забезпечувати виявлення будь-яких випадкових або навмисних змін у переданих пакетах. Для цього використовуються хеш-функції та коди автентифікації повідомлень (MAC), наприклад, HMAC-SHA256.

Вибір конкретних алгоритмів та параметрів шифрування повинен бути доступний для конфігурації адміністратором системи, дозволяючи знаходити баланс між рівнем безпеки та продуктивністю, зважаючи на можливості Raspberry Pi 4.

Система має виконувати функцію керування IP-адресацією для VPN-клієнтів. Після успішної автентифікації та встановлення VPN-з'єднання кожному клієнту має бути призначено унікальну IP-адресу з заздалегідь визначеного пулу адрес. Ця IP-адреса використовуватиметься для ідентифікації клієнта всередині віртуальної приватної мережі.

Функціонал може бути аналогічним DHCP-серверу, але діяти у контексті VPN-тунелю. Система повинна дозволяти адміністратору конфігурувати діапазон IP-адрес, що видаються, час оренди IP-адреси та інші пов'язані параметри. Також може бути корисним функція призначення статичних IP-адрес певним клієнтам на основі їхніх сертифікатів або інших ідентифікаторів.

Важливим блоком функціональних вимог є забезпечення безпеки самого VPN-сервера та керування доступом. Програмно-технічні засоби повинні містити механізми контролю доступу, що дають змогу визначати, які ресурси локальної мережі будуть доступні VPN-клієнтам. Це може реалізовуватися через налаштування правил маршрутизації та брандмауера. Наприклад, система повинна дозволяти обмежувати доступ клієнтів лише до певних серверів або сервісів у локальній мережі, або ж надавати повний доступ до всієї мережі (split tunneling vs full tunneling).

Система повинна бути захищеною від типових мережевих атак, спрямованих на VPN-сервери, як-от DDoS-атаки (в межах можливостей платформи), спроби підбору паролів або експлуатації відомих вразливостей у програмному забезпеченні VPN.

Ведення журналів (логування) та моніторинг стану є критично важливими функціональними вимогами для адміністрування та забезпечення безпеки. Система повинна реєструвати всі важливі події, такі як спроби підключення (успішні та невдалі), помилки, зміни конфігурації, IP-адреси підключених клієнтів та обсяг переданого трафіку.

Ці журнали є необхідними для аналізу роботи сервера, виявлення потенційних проблем безпеки, діагностики несправностей та аудиту. Формат логів повинен бути зрозумілим та придатним для автоматизованої обробки. Окрім логування, система повинна надавати адміністратору засоби для моніторингу поточного стану VPN-сервера в реальному часі.

Це включає відображення списку активних підключень, завантаження процесора та оперативної пам'яті Raspberry Pi, використання мережевого інтерфейсу. Такий моніторинг допоможе вчасно виявляти перевантаження системи або нетипову активність.

З точки зору керування та зручності використання, програмно-технічні засоби повинні надавати зручний та безпечний інтерфейс для конфігурування VPN-сервера. Враховуючи, що Raspberry Pi часто використовується в режимі без графічного інтерфейсу, основним способом конфігурації може бути командний рядок через SSH-з'єднання.

Проте, для спрощення налаштування, особливо для менш досвідчених користувачів, бажаною є наявність веб-інтерфейсу адміністрування. Цей інтерфейс повинен бути захищений паролем та, можливо, іншими засобами автентифікації, та дозволяти керувати основними параметрами сервера, користувачами, сертифікатами та переглядати журнали. Також система повинна дозволяти легко

генерувати конфігураційні файли для VPN-клієнтів, що спрощує процес їх підключення.

Сумісність з клієнтськими операційними системами є важливою функціональною вимогою. VPN-сервер мусить підтримувати підключення від клієнтів, що працюють на найпоширеніших настільних та мобільних операційних системах, як-от Windows, macOS, Linux, Android та iOS. Це досягається шляхом використання стандартизованих VPN-протоколів та надання відповідних конфігураційних файлів або інструкцій для налаштування клієнтського програмного забезпечення.

Вимоги до продуктивності та масштабованості для VPN-сервера на Raspberry Pi 4 повинні бути реалістичними, враховуючи апаратні можливості платформи. Функціонально система повинна бути здатною обробляти певну кількість одночасних VPN-підключень без суттєвої деградації продуктивності.

Ця кількість залежить від обраного VPN-протоколу, рівня шифрування та інтенсивності використання каналу клієнтами. Наприклад, для WireGuard очікується підтримка більшої кількості клієнтів та вища пропускна здатність порівняно з OpenVPN на тому самому обладнанні.

Система повинна забезпечувати максимально можливу пропускну здатність для зашифрованого трафіку, наближену до можливостей мережевого інтерфейсу Raspberry Pi 4 та його процесора при виконанні криптографічних операцій. Програмне забезпечення VPN повинно бути оптимізоване для ефективного використання обмежених ресурсів процесора та оперативної пам'яті Raspberry Pi.

Надійність та доступність є невід'ємними функціональними аспектами. VPN-сервер повинен стабільно функціонувати протягом тривалого часу без збоїв та непередбачуваних перезавантажень. Програмне забезпечення має бути стійким до помилок та здатним коректно обробляти нештатні ситуації, наприклад, тимчасові розриви мережевого з'єднання з боку клієнта або сервера.

Функціонал повинен сприяти автоматичному відновленню з'єднань після усунення проблем з мережею, якщо це підтримується клієнтським програмним

забезпеченням. Хоча реалізація повноцінних механізмів високої доступності (кластеризації) на одному Raspberry Pi є неможливою, система повинна бути спроектована таким чином, щоб мінімізувати час простою та забезпечувати швидке відновлення роботи після можливих збоїв.

З огляду на специфіку платформи Raspberry Pi 4, окремою функціональною вимогою є ефективне використання системних ресурсів. Усі компоненти програмно-технічних засобів VPN-сервера повинні бути розроблені або сконфігуровані таким чином, щоб мінімізувати навантаження на центральний процесор, споживання оперативної пам'яті та операції введення-виведення на microSD карту.

Це особливо важливо для забезпечення стабільної роботи пристрою, який може одночасно виконувати й інші завдання, окрім функцій VPN-сервера. Вибір легко вагового програмного забезпечення та його ретельне налаштування є ключовими для досягнення цієї мети. Крім того, програмне забезпечення повинно бути сумісним та легко розгортатися на ARM-архітектурі, яку використовує Raspberry Pi.

Отже, функціональні потреби до програмно-технічного комплексу VPN-сервера на Raspberry Pi 4 охоплюють низку ключових моментів: від основних операцій безпечного тунелювання та перевірки автентичності до процесів керування, спостереження, продуктивності та стабільності. Вдале втілення цих потреб дасть змогу розробити дієве та захищене VPN-рішення, яке враховує особливості та обмеження популярної одноплатної комп'ютерної платформи, надаючи користувачам надійний інструмент для забезпечення безпеки їх онлайн-діяльності.

2.3 Нефункційні вимоги

Після окреслення функціональних особливостей програмно-технічних засобів VPN-сервера на базі Raspberry Pi 4, важливим є формулювання

нефункційних потреб. Ці вимоги визначають якісні характеристики системи, розкриваючи не те, що система виконує, а як вона це робить.

Нефункційні потреби задають критерії оцінки продуктивності, надійності, безпеки, зручності використання, супроводжуваності та інших атрибутів якості, що є критично важливими для успішної та ефективної експлуатації VPN-сервера, особливо на платформі з обмеженими ресурсами, якою є Raspberry Pi 4. Ігнорування цих вимог може призвести до створення системи, яка, хоч і виконує заявлені функції, проте є повільною, нестабільною, вразливою до атак або складною в управлінні.

Продуктивність VPN-сервера на Raspberry Pi 4 є одним з ключових блоків нефункційних потреб, оскільки апаратні можливості одноплатного комп'ютера накладають значні обмеження.

Першочерговою вимогою до продуктивності є пропускна здатність. Вона визначає обсяг даних, який програмно-технічні засоби VPN-сервера можуть обробити (зашифрувати, передати через тунель та розшифрувати) за одиницю часу, зазвичай вимірюється в мегабітах за секунду (Мбіт/с). Для Raspberry Pi 4, обладнаного гігабітним Ethernet-портом, теоретична верхня межа передачі даних висока, однак реальна пропускна здатність VPN-сервера буде значно обмежена потужністю центрального процесора Broadcom BCM2711, який виконує криптографічні перетворення.

Необхідно встановити очікувані рівні пропускної здатності для різних сценаріїв використання та конфігурацій. Наприклад, при використанні протоколу OpenVPN з алгоритмом шифрування AES-256-GCM, система має забезпечувати пропускну здатність, достатню для комфортного веб-серфінгу, роботи з електронною поштою та передачі невеликих файлів для кількох одночасних користувачів.

Для більш ефективного протоколу WireGuard, який відомий меншими накладними витратами, очікувана пропускна здатність на тому ж апаратному забезпеченні Raspberry Pi 4 має бути суттєво вищою, потенційно дозволяючи

потокове передавання відео високої чіткості або швидший обмін великими файлами. Вимога повинна також враховувати, що збільшення кількості одночасних підключень або використання складніших конфігурацій маршрутизації може знижувати загальну доступну пропускну здатність для кожного окремого клієнта.

Тісно пов'язаною з пропускною здатністю є вимога до затримки, яку вносить VPN-сервер. Затримка – це час, потрібний для проходження пакету даних від клієнта до кінцевого ресурсу через VPN-тунель і назад, понад затримку, що існує в мережі без VPN. Криптографічні операції та обробка пакетів на сервері неминуче додають певну затримку.

Для програмно-технічних засобів на Raspberry Pi 4 ця додаткова затримка має бути мінімізована наскільки це можливо. Низька затримка є критично важливою для інтерактивних додатків, таких як онлайн-ігри, VoIP-телефонія та відеоконференцзв'язок. Система повинна прагнути до того, щоб додаткова затримка, внесена VPN-сервером, не перевищувала значень, які б робили такі додатки некомфортними у використанні. Наприклад, додаткова затримка в кілька десятків мілісекунд може бути прийнятною, але сотні мілісекунд вже негативно вплинуть на досвід користувача.

Місткість з'єднань визначає максимальну кількість одночасних VPN-клієнтів, яку сервер на Raspberry Pi 4 може підтримувати без значного падіння продуктивності для кожного з них. Ця вимога залежить від обсягу доступної оперативної пам'яті (LPDDR4 SDRAM, яка може бути 2GB, 4GB або 8GB на Raspberry Pi 4) та завантаження процесора.

Кожне активне VPN-з'єднання споживає певну кількість пам'яті для зберігання стану сесії, буферів даних та інших структур. Процесорний час також розподіляється між обробкою трафіку для всіх активних клієнтів. Нефункційна вимога повинна визначати цільову кількість одночасних підключень, наприклад, від 5 до 15 активних користувачів, залежно від моделі Raspberry Pi 4 та обраного VPN-протоколу.

Система повинна коректно обробляти ситуації, коли кількість запитів на підключення перевищує її номінальну місткість, можливо, шляхом відхилення нових з'єднань замість повного колапсу сервісу.

Використання ресурсів є надзвичайно важливою нефункційною вимогою для Raspberry Pi 4. Програмно-технічні засоби VPN-сервера повинні бути оптимізовані для ефективного споживання обмежених ресурсів: центрального процесора (CPU), оперативної пам'яті (RAM) та пропускної здатності системи вводу-виводу (особливо при роботі з microSD картою).

Середнє завантаження CPU під час типової роботи VPN-сервера з очікуваною кількістю клієнтів не повинно перевищувати, наприклад, 50-70%, щоб залишався резерв для пікових навантажень та інших системних процесів. Споживання оперативної пам'яті самим VPN-сервісом та пов'язаними з ним процесами має бути таким, щоб операційна система та інші можливі фонові служби могли стабільно функціонувати.

Надмірне використання ресурсів може призвести до перегріву Raspberry Pi 4, спрацювання механізму тротлінгу (зниження тактової частоти процесора) і, як наслідок, до значного погіршення продуктивності та стабільності. Оптимізація використання ресурсів також позитивно впливає на енергоспоживання пристрою.

Стосовно масштабованості, для одноплатного комп'ютера Raspberry Pi 4 вертикальна масштабованість (збільшення потужності шляхом заміни компонентів) практично відсутня. Горизонтальна масштабованість (додавання нових вузлів) для простого VPN-сервера зазвичай не розглядається.

Тож, вимога до масштабованості тут радше стосується здатності програмного забезпечення ефективно використовувати доступні ресурси різних моделей Raspberry Pi 4 (з різним обсягом RAM) та можливості гнучкого налаштування параметрів VPN-сервера (наприклад, сили шифрування, кількості робочих процесів) для адаптації до різного очікуваного навантаження в межах одного пристрою. Система повинна дозволяти адміністратору конфігурувати її таким

чином, щоб досягти найкращого балансу між безпекою, продуктивністю та кількістю підтримуваних клієнтів на конкретній апаратній конфігурації.

Стійкість до атак є критично важливою. VPN-сервер, будучи доступним з Інтернету, є потенційною мішенню для різноманітних атак. Система повинна демонструвати належний рівень стійкості до поширених мережевих атак, таких як спроби сканування портів, атаки типу "відмова в обслуговуванні" (DoS) та розподілені атаки типу "відмова в обслуговуванні" (DDoS) – наскільки це можливо в межах обчислювальних ресурсів Raspberry Pi 4.

Це може включати механізми обмеження швидкості для IP-адрес, що генерують надмірну кількість запитів, або інтеграцію з файрволом для блокування шкідливого трафіку. Також важлива стійкість до атак на протоколи автентифікації, наприклад, шляхом впровадження захисту від перебору паролів.

Якість реалізації конфіденційності та цілісності даних також є нефункційною вимогою. Хоча функціональні вимоги визначають необхідність шифрування, нефункційні уточнюють, що використовуються криптографічні алгоритми та протоколи повинні бути актуальними, визнаними криптографічною спільнотою як сильні та не мати відомих критичних вразливостей.

Довжина ключів шифрування повинна відповідати сучасним стандартам безпеки (наприклад, AES-256, ключі RSA не менше 2048 біт, або еліптичні криві відповідної стійкості). Генерація та управління криптографічними ключами мають здійснюватися безпечним чином, мінімізуючи ризик їх компрометації. Система повинна уникати використання застарілих або слабких шифрів та хеш-функцій.

Безпека конфігурації означає, що програмно-технічні засоби VPN-сервера повинні мати безпечну конфігурацію за замовчуванням ("secure by default"). Це мінімізує ризик неправильного налаштування системи недосвідченим користувачем, що могло б призвести до створення вразливостей. Будь-які інтерфейси управління, чи то командний рядок через SSH, чи веб-інтерфейс, повинні бути захищені надійними механізмами автентифікації та авторизації. Доступ до конфігураційних файлів та критичних системних параметрів має бути

суворо обмежений. Паролі за замовчуванням, якщо такі використовуються під час початкового встановлення, повинні бути унікальними або система повинна вимагати їх негайної зміни.

Управління вразливостями стосується здатності системи бути постійно актуальною та безпечною. Використане програмне забезпечення (операційна система Raspberry Pi OS, VPN-серверне програмне забезпечення, допоміжні утиліти) мусить бути вчасно оновлюване для виправлення виявлених вразливостей.

Процедура оновлення має бути максимально легкою та безпечною, щоб адміністратори могли виконувати її регулярно, не докладаючи великих зусиль або ризикуючи порушенням роботи сервера. Це також передбачає використання програмних компонентів, що мають активну підтримку з боку розробників або спільноти.

Середній час безвідмовної роботи (MTBF) є показником надійності. Для системи на базі Raspberry Pi 4, що використовує компоненти споживчого класу, очікування надзвичайного MTBF, як у серверного обладнання корпоративного рівня, є нереальним. Але, система мусить бути спроєктована та налаштована так, щоб забезпечити тривалу стабільну роботу без апаратних чи програмних збоїв протягом тижнів або навіть місяців безперервної експлуатації за умови стабільного живлення та достатнього охолодження. Це охоплює стабільність операційної системи, самого VPN-сервісу, а також відсутність витоків пам'яті або інших проблем, які можуть накопичуватись з часом.

Середній час відновлення (MTTR) характеризує швидкість, з якою система може бути відновлена до робочого стану після збою. Він включає час, потрібний для діагностики проблеми, її усунення та перезапуску сервісів.

Для Raspberry Pi 4 час перезавантаження операційної системи та автоматичного запуску VPN-сервісу мусить бути мінімальним, наприклад, в межах кількох хвилин. Якщо відновлення потребує втручання адміністратора, наявність чітких діагностичних повідомлень та логів може суттєво скоротити MTTR.

Можливість швидкого відновлення з резервної копії конфігурації також сприяє зменшенню цього показника.

Доступність – це відсоток часу, протягом якого VPN-сервер є працездатним та доступним для користувачів. Для VPN-сервера на Raspberry Pi, що може використовуватися для особистих потреб або малого бізнесу, вимога до доступності може бути встановлена на рівні, скажімо, 99% або 99.5%.

Це означає, що сумарний час простою протягом року не повинен перевищувати певну кількість годин. Досягнення високої доступності потребує не лише надійності самого програмного та апаратного забезпечення, але й стабільності електроживлення та мережевого з'єднання.

Відмовостійкість на одноплатній системі, на кшталт Raspberry Pi 4, обмежена через відсутність апаратного резервування. Але, програмне забезпечення VPN-сервера має намагатись коректно обробляти незначні помилки, такі як тимчасові проблеми з мережею або короткочасні помилки читання з microSD карти (якщо це можливо відстежити та обробити на програмному рівні), без повного краху сервісу. Наприклад, при тимчасовому розриві з'єднання з клієнтом, сервер повинен бути готовий прийняти повторне підключення після відновлення зв'язку.

Особливу увагу в контексті надійності варто приділити цілісності даних на сховищі, тобто на microSD карті. Відомо, що microSD карти мають обмежений ресурс циклів запису та менш надійні, ніж SSD або HDD. Отож, нефункціональною вимогою є мінімізація операцій запису на microSD карту, особливо для даних, що часто оновлюються, як-от детальні логи.

Логування має бути налаштоване так, щоб записувати тільки критично важливу інформацію, або ж система має підтримувати можливість перенаправлення логів на зовнішній USB-накопичувач чи мережеве сховище. Важливою також є стійкість файлової системи до раптових вимкнень живлення, хоча це значною мірою залежить від операційної системи та обраної файлової системи. Простота встановлення та початкового налаштування є важливою

вимогою, особливо з огляду на те, що Raspberry Pi часто використовується ентузіастами та користувачами з різним рівнем технічної підготовки.

Процедура розгортання VPN-сервера має бути максимально автоматизованою або супроводжуватись чіткими покроковими інструкціями. Наявність готових скриптів встановлення або пакетів для популярних дистрибутивів Raspberry Pi OS значно спрощує цей процес. Початкова конфігурація повинна надавати розумні значення за замовчуванням, які забезпечують базовий рівень безпеки та функціональності "з коробки".

Простота управління та адміністрування стосується щоденних завдань, таких як додавання та видалення користувачів VPN, генерування та відкликання сертифікатів, зміна параметрів сервера, моніторинг активних підключень та перегляд логів.

Інтерфейс управління, чи то командний рядок, чи опціональний веб-інтерфейс, має бути інтуїтивно зрозумілим та ефективним. Команди повинні мати логічний синтаксис, а веб-інтерфейс (якщо його реалізовано) – чітку структуру та навігацію. Важливо, щоб адміністратор міг легко отримати необхідну інформацію про стан сервера та швидко вносити необхідні зміни.

Документація є невід'ємною складовою зручності використання. Має бути доступна повна, точна та актуальна документація, що охоплює всі аспекти встановлення, конфігурації, адміністрування, усунення несправностей та найкращі практики безпеки для VPN-сервера на Raspberry Pi 4. Документація повинна бути написана зрозумілою мовою та містити практичні приклади. Це можуть бути як офіційні посібники, так і якісні матеріали від спільноти.

Простота освоєння визначає, наскільки швидко новий адміністратор може навчитися ефективно працювати з системою. Добре структурована документація, інтуїтивний інтерфейс та логічна організація конфігураційних файлів сприяють швидкому освоєнню. Якщо система використовує стандартні та добре відомі інструменти та підходи, це також полегшує навчання.

Система (включно з операційною системою та безпосередньо програмним забезпеченням VPN-сервера) мусить легко оновлюватись для встановлення патчів безпеки, виправлення помилок та отримання нових функцій. Процес оновлення має бути безпечним, тобто мінімізувати ризик порушення працездатності сервера чи втрати конфігурації. Бажано, щоб оновлення не потребували складних ручних маніпуляцій.

Діагностованість означає, що система має надавати достатньо інформації для швидкого виявлення та аналізу причин проблем або збоїв. Це досягається за допомогою детального та структурованого логування подій, помилок та стану системи. Наявність вбудованих засобів діагностики або можливість легко підключити стандартні інструменти моніторингу та аналізу логів (наприклад, для Raspberry Pi OS) є важливою.

Конфігурованість в контексті супроводжуваності означає, що параметри системи повинні бути легко доступні для зміни через конфігураційні файли або інтерфейс управління. Структура конфігураційних файлів має бути зрозумілою та добре документованою. Це дозволяє адміністраторам адаптувати поведінку сервера до специфічних потреб без необхідності модифікації вихідного коду програмного забезпечення.

Модульність програмного забезпечення VPN-сервера, якщо це можливо (наприклад, використання окремих компонентів для автентифікації, шифрування, управління мережею), може спростити процес супроводу. Заміна або оновлення окремого модуля є менш ризикованим та трудомістким, ніж робота з монолітною системою.

Для VPN-сервера, що тісно прив'язаний до апаратної платформи Raspberry Pi 4, вимоги до переносимості є менш критичними, ніж для універсального програмного продукту. Але, бажано, щоб програмне забезпечення VPN-сервера базувалося на відкритих стандартах та технологіях, які широко підтримуються на різних ARM-сумісних платформах. Це може полегшити міграцію на іншу одноплатну систему в майбутньому або використання досвіду налаштування на

подібних пристроях. Основний акцент тут на сумісності з операційною системою Raspberry Pi OS та її стандартними бібліотеками.

Енергоспоживання є однією з сильних сторін Raspberry Pi. Нефункційна вимога полягає в тому, щоб програмно-технічні засоби VPN-сервера не призводили до надмірного збільшення енергоспоживання пристрою. Система повинна залишатись енергоефективною як в режимі простою, так і під навантаженням. Низьке енергоспоживання дозволяє експлуатувати сервер цілодобово з мінімальними витратами на електроенергію та зменшує тепловиділення. Очікуване енергоспоживання під типовим навантаженням VPN-сервера має бути в межах кількох ват.

Теплові показники напряму корелюють з продуктивністю та стабільністю. Процесор Raspberry Pi 4 відчутно нагрівається під час інтенсивних навантажень, наприклад, при безперервних криптографічних операціях. Нефункціональна вимога передбачає стабільну роботу системи в умовах звичайної домашньої або офісної температури, уникнення перегріву, що викликає тротлінг або збої.

Це може потребувати пасивного (радіатори) або навіть активного (вентилятор) охолодження. Програмне забезпечення не повинно створювати надмірного теплового навантаження, яке неможливо компенсувати стандартними засобами охолодження Raspberry Pi. Адміністратор повинен мати змогу контролювати температуру процесора.

Насамкінець, ретельне формулювання та врахування нефункціональних вимог – ключовий момент для створення високоякісного, надійного та ефективного програмно-апаратного комплексу VPN-сервера на базі Raspberry Pi 4. Ці вимоги гарантують, що система не лише виконуватиме свої основні завдання, але й робитиме це з належним рівнем продуктивності, безпеки та зручності для кінцевого користувача, використовуючи переваги та враховуючи обмеження обраної апаратної платформи. Вони слугують основою для вибору архітектурних рішень, компонентів програмного забезпечення та для подальшого тестування готового продукту.

2.4 Висновки до другого розділу

Проведений у даному розділі комплексний аналіз апаратного середовища, функційних та нефункційних вимог до програмно-технічних засобів VPN-сервера на основі одноплатної комп'ютерної системи Raspberry Pi 4 дозволяє сформулювати обґрунтовані висновки щодо специфіки розробки та впровадження таких рішень.

Було встановлено, що Raspberry Pi 4, завдяки своєму процесору ARM Cortex-A72, достатньому обсягу оперативної пам'яті LPDDR4 та наявності гігабітного Ethernet-порту, є принципово придатною платформою для розгортання VPN-серверів, орієнтованих на потреби індивідуальних користувачів, домашніх мереж або невеликих офісів.

Водночас, виявлені апаратні обмеження, зокрема щодо обчислювальної потужності для інтенсивних криптографічних операцій та надійності microSD карт як основного сховища, диктують необхідність ретельного підбору програмного забезпечення та оптимізації його конфігурацій. Забезпечення належного живлення та ефективного охолодження також визначено як критичні фактори для стабільної та довготривалої експлуатації.

Деталізація функційних вимог окреслила ключовий набір можливостей, якими повинен володіти VPN-сервер. До них належать, передусім, гарантування безпечного тунелювання трафіку з використанням сучасних протоколів, таких як OpenVPN та WireGuard, реалізація надійних механізмів багатофакторної автентифікації користувачів та пристроїв, забезпечення конфіденційності та цілісності даних шляхом застосування стійких алгоритмів шифрування, а також гнучке управління IP-адресацією клієнтів.

Окрім цього, наголошено на важливості функцій моніторингу, детального логування подій для цілей безпеки та діагностики, і забезпечення сумісності з широким спектром клієнтських операційних систем.

Розгляд нефункційних вимог акцентував увагу на якісних атрибутах системи, що є вирішальними для її практичної цінності. Пріоритетними визначено вимоги до продуктивності, зокрема до пропускної здатності зашифрованого каналу та мінімальної внесеної затримки, які мають бути реалістично збалансовані з обчислювальними можливостями Raspberry Pi 4.

Високий рівень безпеки, що виражається у стійкості до типових атак та безпечній конфігурації за замовчуванням, є обов'язковою умовою. Надійність системи, що включає стабільність роботи та швидкість відновлення після можливих збоїв, має враховувати особливості використання компонентів споживчого класу.

Зручність встановлення, налаштування та подальшого адміністрування, підкріплена якісною документацією, суттєво впливає на доступність рішення для широкого кола користувачів. Не менш важливою є супроводжуваність системи, що передбачає простоту оновлення та діагностики.

Узагальнюючи, успішна реалізація програмно-технічних засобів VPN-сервера на Raspberry Pi 4 можлива за умови синергетичного врахування всіх трьох аспектів: адекватного апаратного забезпечення, повноцінного набору функціональних можливостей та досягнення необхідних якісних показників, визначених нефункційними вимогами.

Такий підхід дозволяє створити збалансоване, економічно ефективне та безпечне рішення, що відповідає сучасним потребам у захисті мережевого трафіку в специфічних сценаріях використання.

3 ПРОГРАМНО-ТЕХНІЧНИЙ ЗАСІБ VPN СЕРВЕРА НА ОСНОВІ ОДНОПЛАТНОЇ КОМП'ЮТЕРНОЇ СИСТЕМИ RASPBERRY PI4

3.1 Принцип роботи програмно-технічного засобу VPN-сервера на основі одноплатного комп'ютера Raspberry Pi 4

Архітектура та системні компоненти VPN-сервера є багатокомпонентною структурою (алгоритм роботи представлений на рисунку 3.1). Технологія віртуальної приватної мережі (VPN) за своєю суттю створює інкапсульований та зашифрований тунель для передачі даних через недовірені мережі, як-от Інтернет, чим гарантує конфіденційність інформації від несанкціонованого доступу.

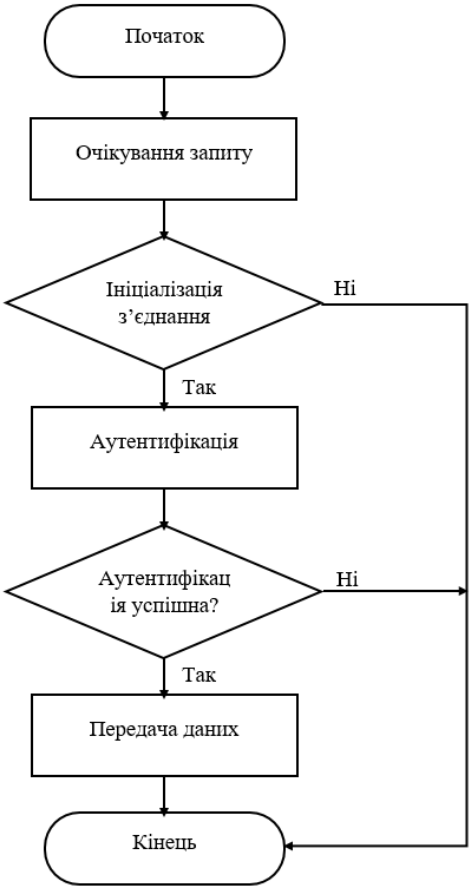


Рисунок 3.1 – Алгоритм роботи засобу

Система на базі Raspberry Pi 4 включає в себе апаратну платформу, програмний стек, мережеву інфраструктуру та підсистему безпеки. Апаратна частина, представлена комп'ютером Raspberry Pi 4 з чотириядерним процесором ARM Cortex-A72, об'ємом оперативної пам'яті 4 або 8 ГБ та набором інтерфейсів (USB, Ethernet, HDMI), володіє обчислювальними ресурсами, достатніми для обробки криптографічних операцій та маршрутизації трафіку на швидкостях, що задовольняють вимоги стабільної роботи VPN-з'єднань.

Функціонування сервера забезпечується операційними системами на ядрі Linux, наприклад Raspbian, що надають необхідне середовище для розгортання. Програмний компонент реалізується за допомогою спеціалізованих застосунків, серед яких домінують OpenVPN та WireGuard. Перший є визнаним стандартом для створення захищених тунелів, тоді як другий, будучи новішим протоколом, відзначається вищою продуктивністю та спрощеною архітектурою.

Мережева інфраструктура вимагає стабільного інтернет-з'єднання, яке може бути реалізовано через дротовий Ethernet або бездротовий Wi-Fi. Для забезпечення зовнішньої доступності сервера необхідна конфігурація статичної IP-адреси та механізму трансляції мережевих адрес (NAT). Фундаментальним елементом є підсистема безпеки, що покладається на криптографічні алгоритми. В OpenVPN переважно використовується симетричний шифр AES з довжиною ключа 128 або 256 біт, тоді як WireGuard застосовує фіксований набір сучасних криптографічних примітивів для досягнення високої швидкості без компрометації рівня захисту.

Принцип функціонування VPN-сервера полягає у послідовному виконанні операцій для створення та підтримки захищеного каналу між клієнтським пристроєм та сервером. На початковому етапі клієнт ініціює з'єднання, виконуючи процедуру "рукостискання" (handshake) для встановлення сеансу зв'язку (це та наступні пункти можна відстідкувати на рисунку 3.2). Під час цього процесу сервер верифікує автентичність клієнта, часто за допомогою цифрових сертифікатів, та відбувається взаємний обмін криптографічними ключами з використанням таких протоколів, як TLS або IPSec.

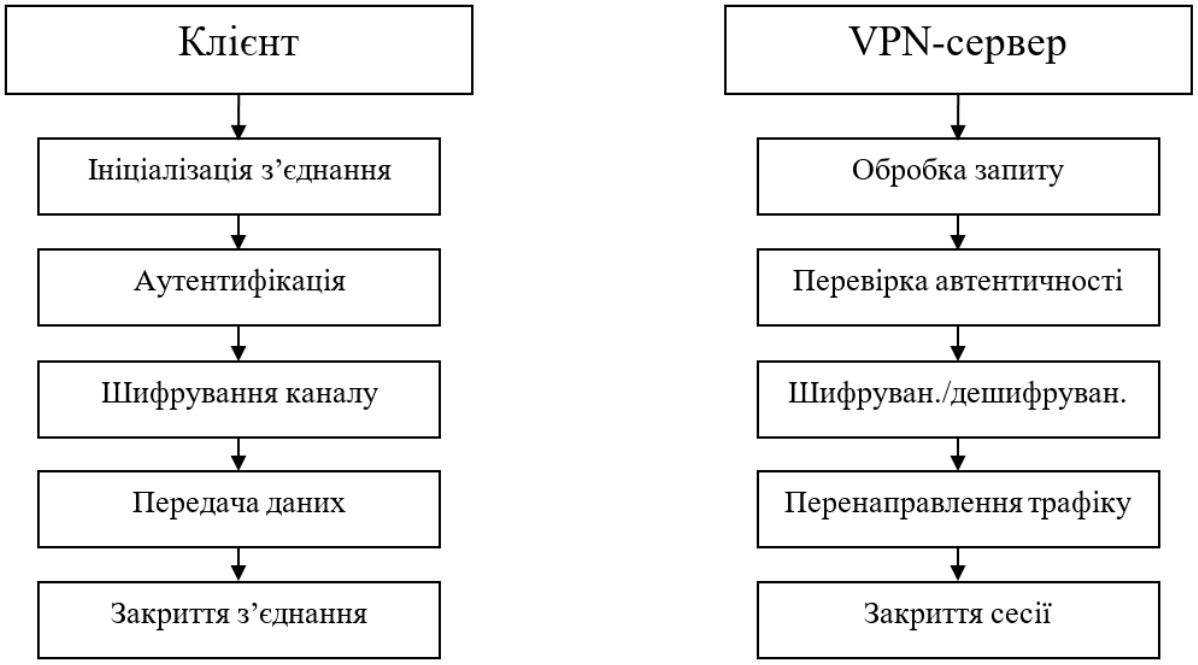


Рисунок 3.2 – Схема роботи пристрою на рівнях клієнт і сервер

Наступним кроком є аутентифікація самого користувача, яка може здійснюватися за допомогою різноманітних механізмів, включно з паролями, клієнтськими сертифікатами або одноразовими токенами (OTP), що є критично важливим для контролю доступу. Після успішної аутентифікації формується зашифрований канал.

Вибір алгоритмів шифрування залежить від протоколу: для OpenVPN це часто AES, а для WireGuard — ChaCha20, який є більш ефективним на пристроях з обмеженими ресурсами. Протягом встановленого з'єднання весь мережевий трафік клієнта інкапсулюється та шифрується, що унеможливлює його перехоплення та аналіз третіми сторонами. По завершенні сесії відбувається коректне закриття з'єднання, під час якого всі тимчасові криптографічні ключі, що використовувались для шифрування, безпечно знищуються для запобігання їхній подальшій компрометації.

Криптографічне забезпечення є наріжним каменем функціонування VPN-сервера, оскільки саме воно відповідає за захист даних. В рамках протоколу OpenVPN застосовується широкий спектр алгоритмів для забезпечення

конфіденційності та цілісності. Наприклад, симетричний блочний шифр AES-256 забезпечує високий рівень криптографічної стійкості, що робить його придатним для корпоративних стандартів безпеки. Протокол WireGuard, натомість, використовує іншу філософію, покладаючись на фіксований набір сучасних алгоритмів, зокрема Curve25519 для еліптично-кривих операцій обміну ключами та ChaCha20 для потокового шифрування даних. Важливою перевагою WireGuard є його мінімалістична кодова база, що суттєво зменшує поверхню атаки та спрощує аудит безпеки, знижуючи ймовірність наявності невиявлених вразливостей.

Процес взаємодії в системі можна описати як послідовний потік операцій. Клієнтський пристрій ініціює запит на з'єднання, проходить етапи аутентифікації, після чого встановлюється шифрований канал для передачі даних, і по завершенні роботи сесія закривається. З боку сервера на Raspberry Pi відбувається обробка вхідного запиту, що включає перевірку автентичності клієнта, виконання операцій шифрування вихідного та дешифрування вхідного трафіку, його подальше перенаправлення до цільового ресурсу в Інтернеті або локальній мережі, і, нарешті, закриття сесії за вимогою клієнта або по тайм-ауту.

Процедура розгортання такого VPN-сервера на платформі Raspberry Pi є методологічно простою. Вона передбачає попередню підготовку операційної системи, зазвичай Raspbian, шляхом її оновлення. Далі відбувається інсталяція обраного програмного забезпечення, наприклад OpenVPN або WireGuard. Наступним етапом є детальне налаштування конфігураційних файлів, що визначають параметри роботи сервера та генерують унікальні конфігурації для кожного клієнта. Після завершення конфігурації здійснюється запуск відповідної системної служби (сервісу), і фінальним кроком є налаштування правил мережевого екрана (фаєрвола) для дозволу вхідних з'єднань на відповідний порт VPN-сервера.

Таким чином, імплементація VPN-сервера на базі одноплатного комп'ютера Raspberry Pi 4 є техніко-економічно обґрунтованим рішенням для створення захищених каналів зв'язку. Принцип дії таких систем повністю ґрунтується на

застосуванні сучасних криптографічних методів для забезпечення конфіденційності та цілісності даних, а також на технологіях мережевого тунелювання. Наявність таких протоколів, як OpenVPN та WireGuard, надає гнучкий вибір між стабільними, багатофункціональними рішеннями та легковагими, високопродуктивними технологіями, що дозволяє оптимізувати систему під конкретні вимоги до безпеки та швидкодії.

3.2 Апаратна реалізація програмно-технічного засобу VPN-сервера на основі одноплатного комп'ютера Raspberry Pi 4.

У межах реалізації програмно-технічного засобу VPN-сервера було спроєктовано апаратну частину на основі одноплатного комп'ютера Raspberry Pi 4. Однією з ключових задач у межах даного етапу стало створення принципової електричної схеми та розведення друкованої плати з використанням системи автоматизованого проєктування KiCad. Цей розділ присвячено детальному опису всіх етапів розробки, починаючи з вибору елементної бази та побудови схеми, до візуалізації готової плати у тривимірному форматі.

Початковий етап передбачав аналіз функціональних вимог до VPN-сервера. Основною вимогою було забезпечення живлення одноплатного комп’ютера, організація доступу до нього через Ethernet та USB-інтерфейси, а також можливість апаратного перезавантаження через кнопку Reset. Крім цього, доцільним було включення світлодіодної індикації для відображення стану системи. Усі ці вимоги було враховано при розробці схеми у середовищі KiCad.

Після створення нового проєкту у KiCad було сформовано структуру проекту та відкрито редактор схем. Додатково були підключені бібліотеки компонентів, зокрема Device, Connector, Diode, Regulator_Linear, LED та інші, що містили необхідні елементи, зокрема стабілізатор AMS1117-3.3, світлодіоди, резистори, кнопки, USB-роз’єми та роз’єми RJ45. Як основний інтерфейс взаємодії з Raspberry

Pi 4 використовувався 40-контактний роз’єм GPIO, який було розміщено на схемі та пов’язано з іншими компонентами за допомогою сигнальних ліній.

У програмі KiCad компонент Conn_02x20_Odd_Even є одним з елементів, який зазвичай використовується для представлення підключень через дворядкові роз'єми з двома піновими рядами по 20 пінів у кожному. Цей компонент використовується для створення з'єднань на схемах, де необхідне підключення через такі роз'єми. Однак, важливо зазначити, що цей компонент не є прямим аналогом фізичного GPIO-з'єднання, яке є характерним для Raspberry Pi 4. В реальності, Conn_02x20_Odd_Even можна розглядати як зручний абстракційний компонент для відображення підключень через роз'єми, які використовуються для з'єднання з GPIO-пінами, але він не є точним представником конкретної моделі Raspberry Pi (рисунок 3.3).

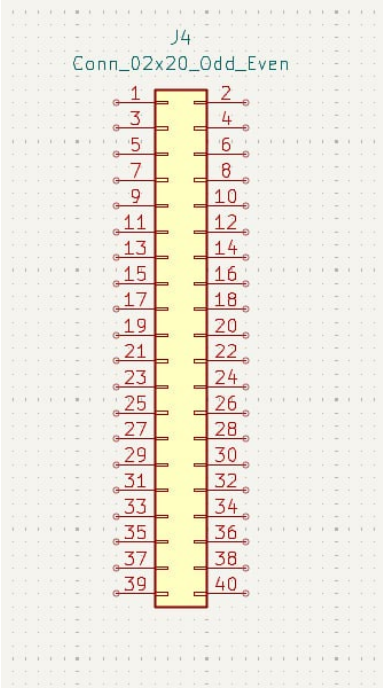


Рисунок 3.3 – Conn_02x20_Odd_Even

По-перше, компонент Conn_02x20_Odd_Even має два ряди пінів, які можуть бути використані для підключення як живлення, так і для введення/виведення сигналів. Піни в цьому компоненті повинні бути підписані таким чином, щоб

відображати функціональні можливості пінів GPIO на платі Raspberry Pi. Це означає, що для кожного піну на заголовку Conn_02x20_Odd_Even треба визначити, чи він буде використовуватись для 5V живлення, 3.3V, GND або для GPIO сигналів.

У процесі підписування пінів слід дотримуватися стандартного призначення пінів для Raspberry Pi 4. Наприклад, перші п'ять пінів на кожному ряді використовуються для живлення (5V і 3.3V), а також для загальної землі (GND). Важливо врахувати, що в заголовку Conn_02x20_Odd_Even піні мають певну нумерацію, яка може бути змінена відповідно до розташування пінів Raspberry Pi. Для точного дублювання підключення необхідно визначити функціональність кожного піну відповідно до документації Raspberry Pi 4 і забезпечити, щоб кожен пін на схемі мав правильне позначення та нумерацію (коректний елемент представлений на рисунку 3.4).

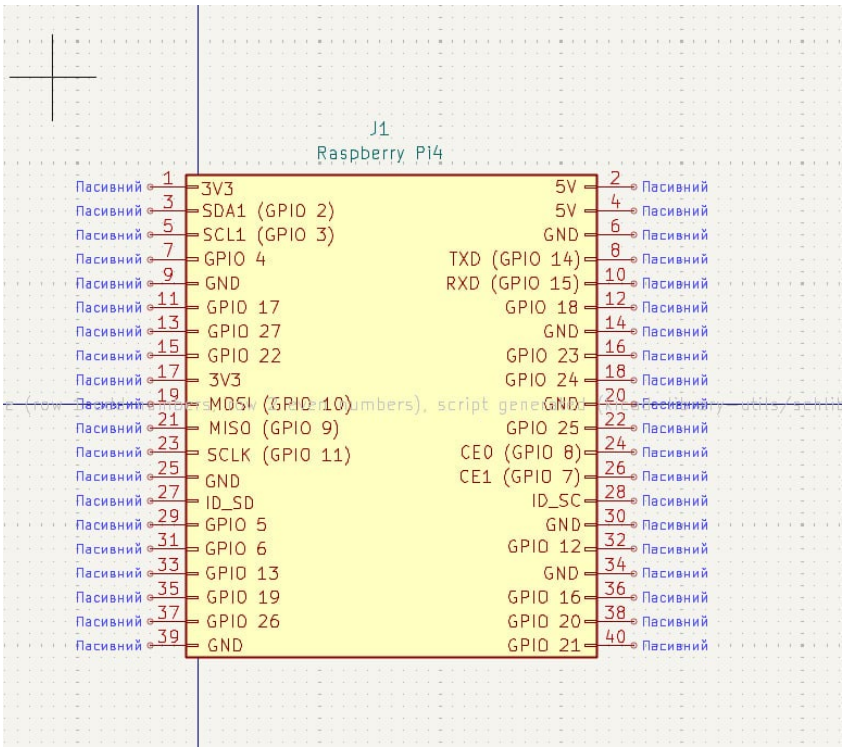


Рисунок 3.4 – Conn_02x20_Odd_Even, як Raspberry Pi4

Особливу увагу було приділено реалізації живлення. Оскільки Raspberry Pi 4 потребує стабільної напруги 3.3 В для частини компонентів, а живиться переважно від 5 В, було використано стабілізатор AMS1117-3.3 (стабілізатор та інші елементи можна побачити на рисунку 3.5). На вході та виході стабілізатора були розміщені конденсатори ємністю 10 мкФ і 100 нФ відповідно для забезпечення фільтрації та стабільності напруги. Важливим аспектом стало правильне підключення виводів стабілізатора: пін 1 було з'єднано із землею, пін 2 – з вихідною лінією 3.3 В, а пін 3 – із вхідною шиною 5 В. Додатково, на вихід стабілізатора було встановлено резистор навантаження на 1 МОм для забезпечення мінімального споживання струму та стабільності роботи при відсутності навантаження.

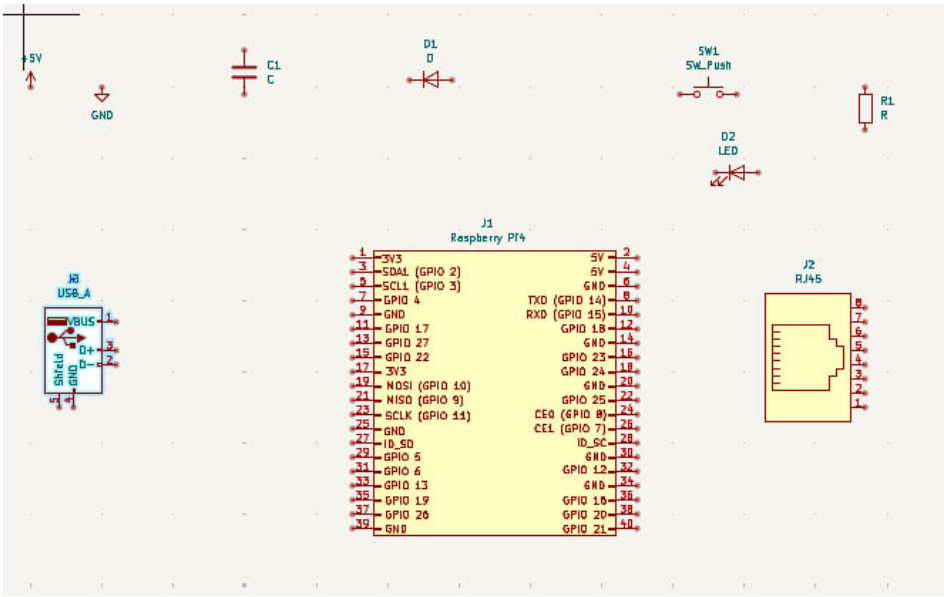


Рисунок 3.5 – Елементи майбутньої схеми

Світлодіодна індикація реалізувалася шляхом підключення світлодіода до одного з GPIO-виводів Raspberry Pi через резистор обмеження струму номіналом 330 Ом. Таким чином, при подачі логічної одиниці на відповідний GPIO-пін відбувається світіння діода, що дозволяє візуально ідентифікувати стан системи. Кнопка Reset підключалася до пінів GPIO3 (фізичний пін 5) та GND (пін 6), що є стандартною практикою для виклику апаратного перезавантаження Raspberry Pi. У

паралель з кнопкою було підключено захисний діод Шотткі для запобігання паразитним імпульсам.

USB-інтерфейс реалізувався у вигляді USB Type-A роз’єму, з’єднаного із шиною 5 В та землею. Лінії D+ та D− залишено непідключеними, оскільки Raspberry Pi вже має вбудований USB-хост, і в схемі передбачалося лише виведення роз’єму для користувача. Аналогічний підхід було застосовано і до роз’єму RJ45, який у даній реалізації не підключався безпосередньо до GPIO, оскільки Ethernet-інтерфейс уже інтегрований у апаратну частину Raspberry Pi. Для обох роз’ємів у схемі було зазначено коментарі "Not connected directly", що дозволяє уникнути помилок під час перевірки правил електричних з’єднань (ERC). Готова схема представлена на рисунку 3.6.

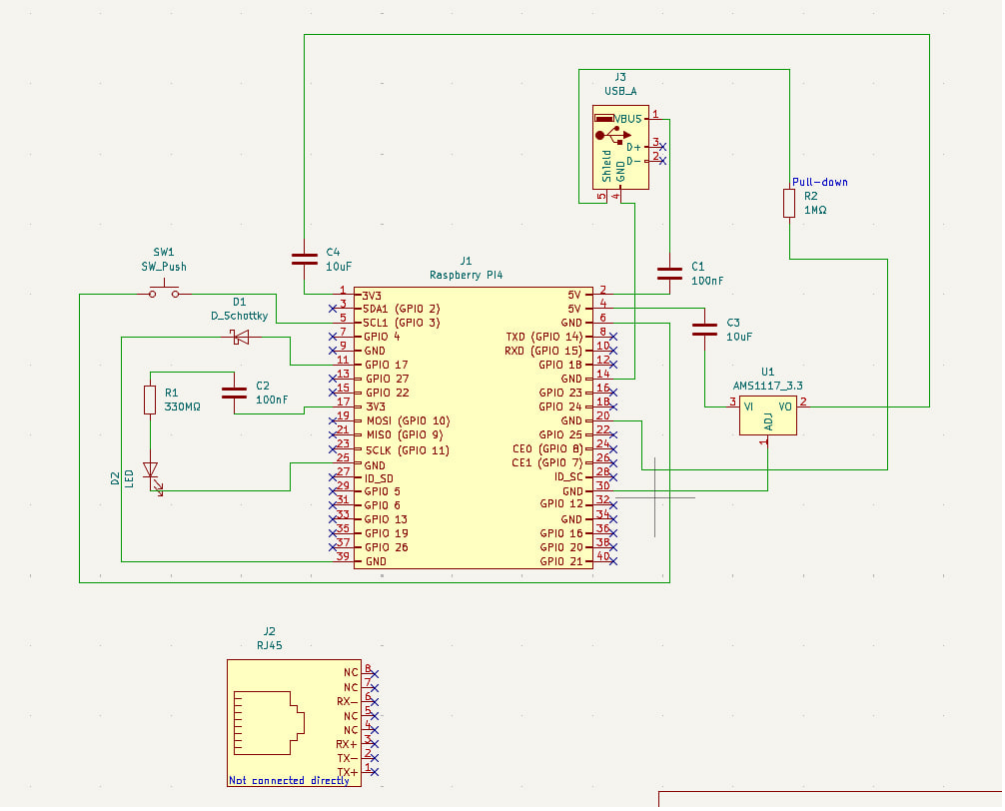


Рисунок 3.6 – Принципова схема апаратної частини програмно-технічного засобу VPN сервера на основі одноплатної комп'ютерної системи Raspberry Pi4

Після завершення побудови принципової схеми та перевірки її коректності за допомогою інструмента ERC було виконано прив’язку елементів схеми до відповідних посадкових місць (footprints) (таблиця 3.1). Для кожного компоненту було обрано корпус, відповідний до реального фізичного елемента. Наприклад, для стабілізатора AMS1117 обрано корпус SOT-223, для резисторів – тип R_0805, для конденсаторів – C_0805, для роз’ємів – відповідні THT або SMD-моделі. Прив’язка відбувалася у вікні Assign Footprints, що дозволило підготувати схему до формування друкованої плати.

Таблиця 3.1 – Елемент-Footprint

Елемент	Footprint
Raspberry Pi 4	Connector_PinHeader_2.54mm:PinHeader_2x20_P2.54mm_Vertical
LED	LED_THT:LED_D5.0mm або LED_SMD:LED_0805
R, C	Resistor_THT:R_Axial_DIN0207_L6.3mm_D2.5mm_P7.62mm (THT) або R_0805 (SMD)
USB-A	Connector_USB:USB_A_Female або USB_A_TH
AMS1117-3.3	Package_TO_SOT_SMD:SOT-223
SW_Push	Button_Switch_THT:SW_PUSH_6mm або SMD-варіант
RJ45 (не підключений)	Connector_RJ:RJ45_8 або MagJack
D_Schottky (THT)	Diode_THT:D_DO-41_SOD81_P10.16mm_Horizontal
D_Schottky (SMD)	Diode_SMD:D_SMA

Після оновлення PCB з принципової схеми відкрився редактор друкованої плати, де всі елементи з’явилися у вигляді розкиданих посадкових місць (рисунок 3.7). На цьому етапі було визначено контур плати, який малювався на шарі Edge.Cuts. Було створено прямокутну область із урахуванням просторового розміщення компонентів та можливості подальшого монтажу. Компоненти розміщувалися у межах цієї області відповідно до логіки з’єднання: роз’єми біля країв плати, елементи живлення компактно в одному секторі, індикатори та кнопки – ближче до доступних сторін плати.

Трасування доріжок здійснювалося у режимі інтерактивного маршрутизатора. Для кожного з'єднання спочатку активувалася відповідна повітряна лінія (airwire), після чого прокладалася мідна доріжка по верхньому шарі (F.Cu).

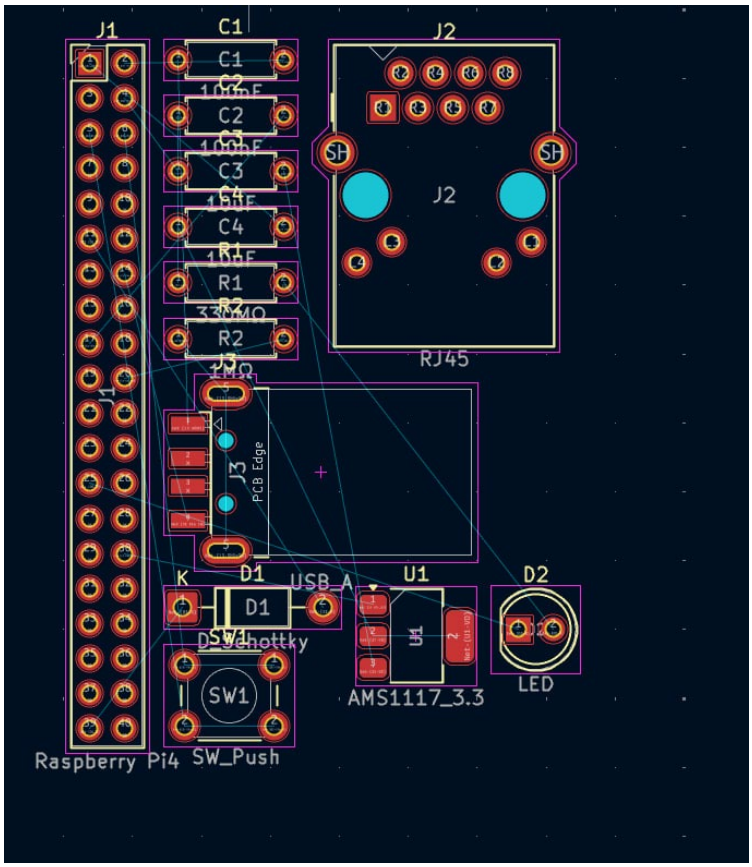


Рисунок 3.7 – Початок розведення плати

Для кожного з'єднання було обрано оптимальний маршрут із дотриманням мінімальної довжини та уникнення перетинів. Особливу увагу приділено заземленню: для лінії GND була створена заливка (polygon pour), яка охоплювала максимальну площу плати. Це не лише спростило розведення, але й підвищило електромагнітну сумісність та стабільність роботи схеми.

Після завершення трасування було виконано перевірку правил трасування (DRC). Жодних критичних помилок не виявлено, що свідчить про відповідність схеми вимогам до виробництва. Після цього були згенеровані виробничі файли – Gerber, які включають усю необхідну інформацію для виготовлення плати. Також сформовано Drill Files для свердління отворів у процесі виготовлення.

Для візуального контролю отриманої плати було використано інструмент 3D Viewer, вбудований у KiCad. У тривимірному перегляді було перевірено правильність розташування компонентів, відповідність посадкових місць фізичним моделям, а також загальний вигляд плати. Цей етап є важливим для перевірки ергономіки та фізичної сумісності плати із корпусом або іншими модулями. Результати 3D-візуалізації повністю відповідали очікуванням (рисунок 3.8).

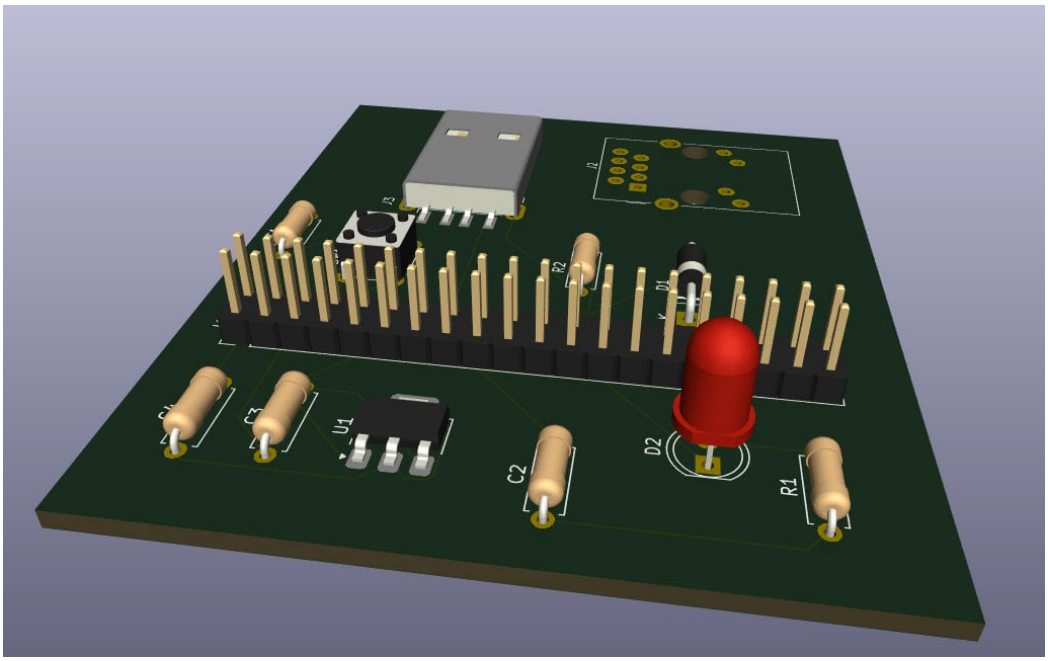


Рисунок 3.8 – 3D модель плати

Загалом, процес створення принципової схеми, розведення друкованої плати та її тривимірної візуалізації у середовищі KiCad довів свою ефективність у проєктуванні апаратної частини VPN-сервера. Інструменти, доступні у KiCad, дозволяють здійснювати повний цикл розробки: від побудови схеми до експорту файлів для виробництва. У результаті було створено функціональну апаратну платформу, здатну виконувати роль частини програмно-технічного комплексу для побудови захищених мережевих рішень на основі Raspberry Pi 4.

3.3 Налаштування та розгортання VPN-сервера у віртуальному середовищі.

Однією з надійних та поширених реалізацій VPN є програмне забезпечення OpenVPN, яке забезпечує високий рівень шифрування, підтримку різноманітних платформ та простоту інтеграції. У межах даного проєкту було реалізовано розгортання VPN-сервера з використанням OpenVPN у віртуальному середовищі з подальшим тестуванням клієнтського з'єднання (схема роботи представлена на рисунку 3.9).

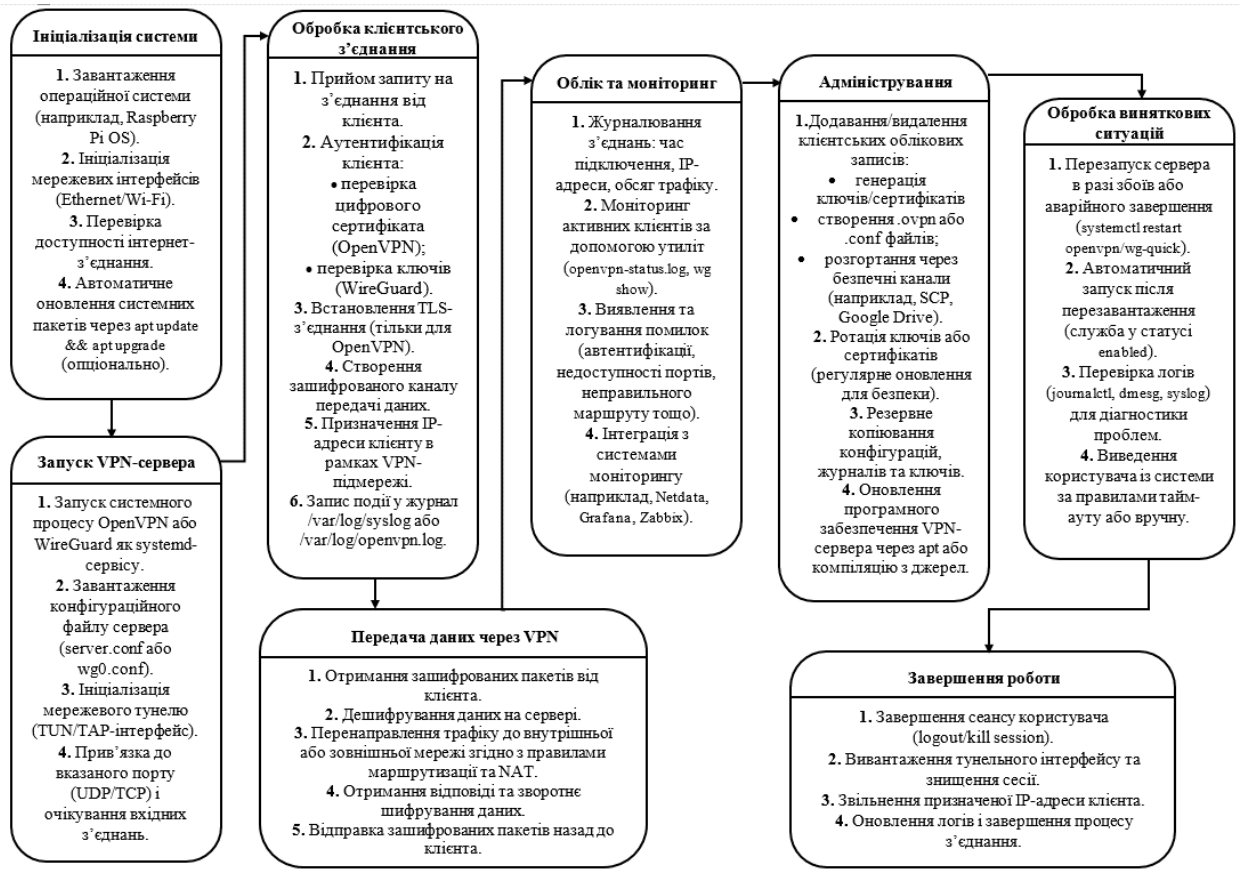


Рисунок 3.9 – Схема роботи програмної частини

Для створення тестового середовища використано віртуалізаційну платформу VirtualBox, що забезпечує запуск ізольованих операційних систем на основі гостьових образів. Основною операційною системою, обраною для розгортання сервера, є Raspbian - спеціалізований дистрибутив Linux, орієнтований на використання в одноплатних комп'ютерах Raspberry Pi. Образ цієї системи було

попередньо завантажено з офіційного джерела та встановлено у середовище VirtualBox. Після створення та запуску віртуальної машини відбулося налаштування мережевого інтерфейсу (рисунок 3.10). Це забезпечило повноцінне функціонування VPN-модуля у середовищі емуляції та дало змогу коректно здійснювати комунікацію між сервером і клієнтом.

Рисунок 3.10 – Обираю попередньо встановлений Raspbian

Особливу увагу було приділено налаштуванню мережевих адаптерів віртуальної машини (рисунок 3.11 та 3.12). Для забезпечення прямого підключення до локальної мережі використовувався режим "Bridged Adapter", який дозволяє віртуальній машині діяти як повноцінний мережевий пристрій з власною IP-адресою. Такий підхід є необхідним для моделювання ситуації, в якій сервер OpenVPN функціонує в умовах реально існуючої корпоративної або домашньої мережі.

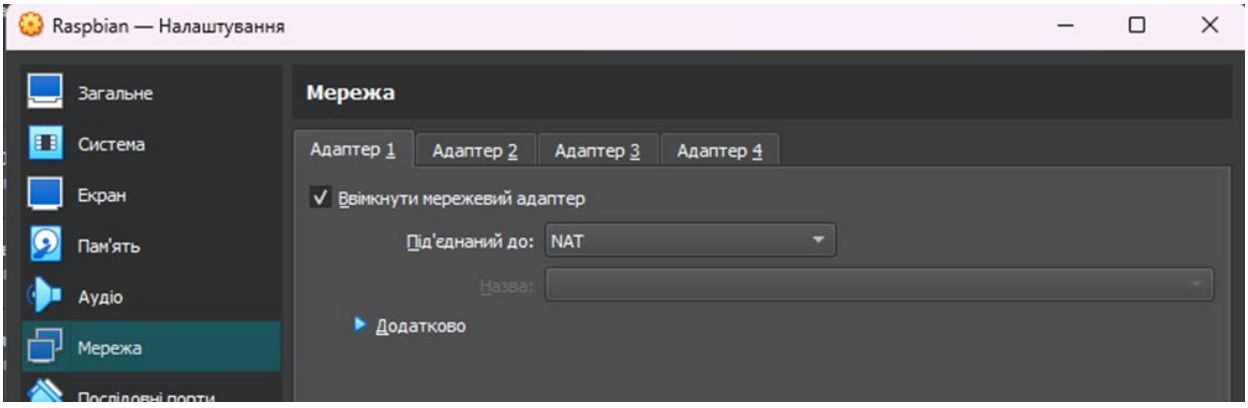


Рисунок 3.11 – Перевіряю налаштування 1

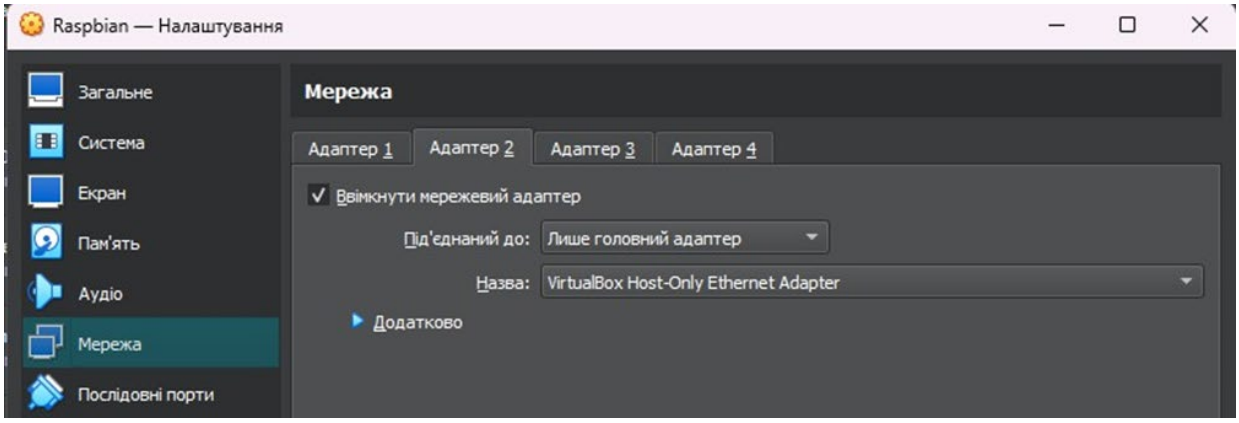


Рисунок 3.12 – Перевіряю налаштування 2

Після ініціалізації системи було запущено термінальне середовище для виконання команд. У ньому переходимо до директорії, де зберігається інсталяційний скрипт для OpenVPN, після чого здійснюється його виконання за допомогою команди sudo ./openvpn-install.sh (рисунок 3.13). Даний скрипт є автоматизованим рішенням, яке суттєво спрощує процес інсталяції та первинного налаштування сервера. Під час його виконання продовжую взаємодіяти з діалоговим інтерфейсом, у якому обираю параметри конфігурації.

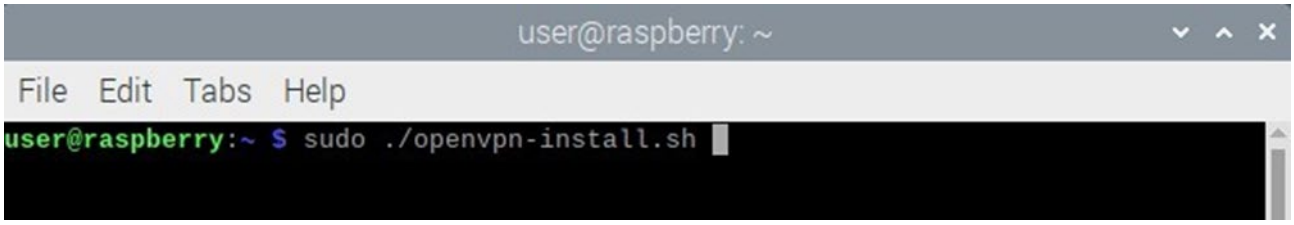


Рисунок 3.13 – Прописую команду в Terminal

На етапі налаштування IP-адреси обираємо одну з доступних адрес, переважно в межах приватного діапазону (наприклад, ті, що починаються з 192.168) (рисунок 3.14). Важливо, щоб ця адреса була дійсною у межах локальної мережі, до якої підключено віртуальну машину. Обрана адреса вказується як локальна IP-адреса VPN-сервера. Надалі скрипт автоматично формує необхідні ключі, сертифікати та конфігураційні файли.

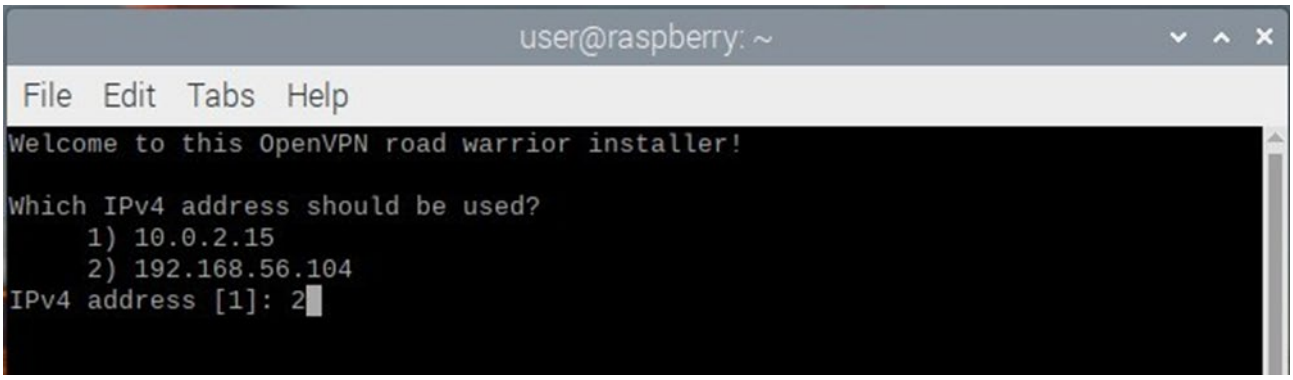


Рисунок 3.14 – Обираю IP-адресу

У подальших кроках залишаю всі параметри за замовчуванням, натискаючи клавішу Enter (рисунок 3.15). Такий підхід дозволяє уникнути помилок конфігурації і гарантує сумісність з типовими клієнтами OpenVPN. Після завершення роботи скрипта отримуємо готовий сервер із попередньо згенерованим конфігураційним файлом клієнта, який зберігається у відповідній директорії у вигляді файлу client.ovpn (рисунок 3.16).

Щоб перевірити, чи сервіс OpenVPN активний, у терміналі виконується команда systemctl status openvpn. У разі, якщо сервіс не запущений, активується команда systemctl start openvpn, що ініціалізує VPN-сервер (коректну роботу відображено на рисунку 3.17). У цьому контексті важливо, аби були права адміністратора, оскільки виконання вказаних команд потребує підвищеного рівня привілеїв.

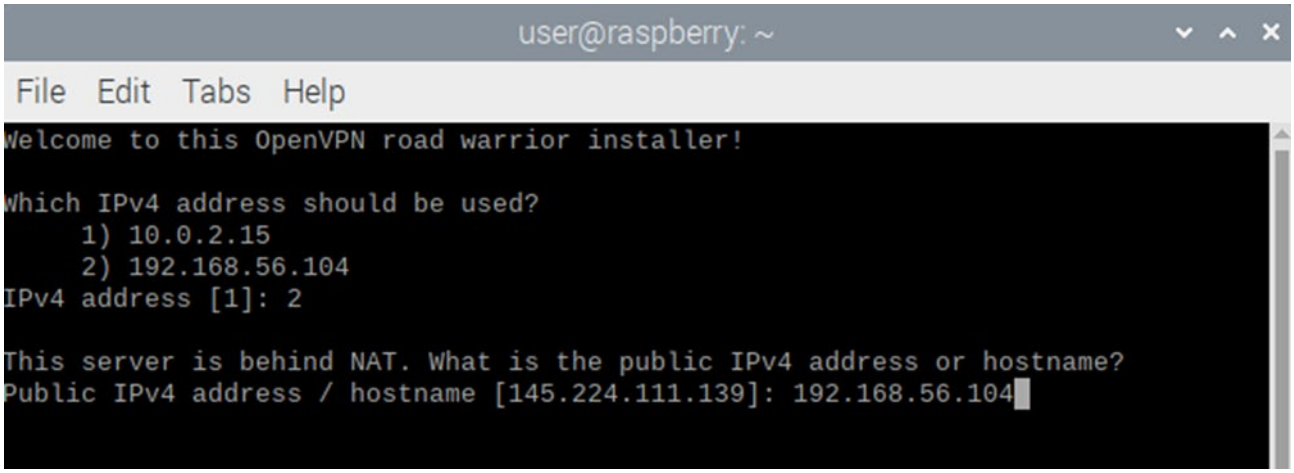


Рисунок 3.15 – Обираю IP-адресу

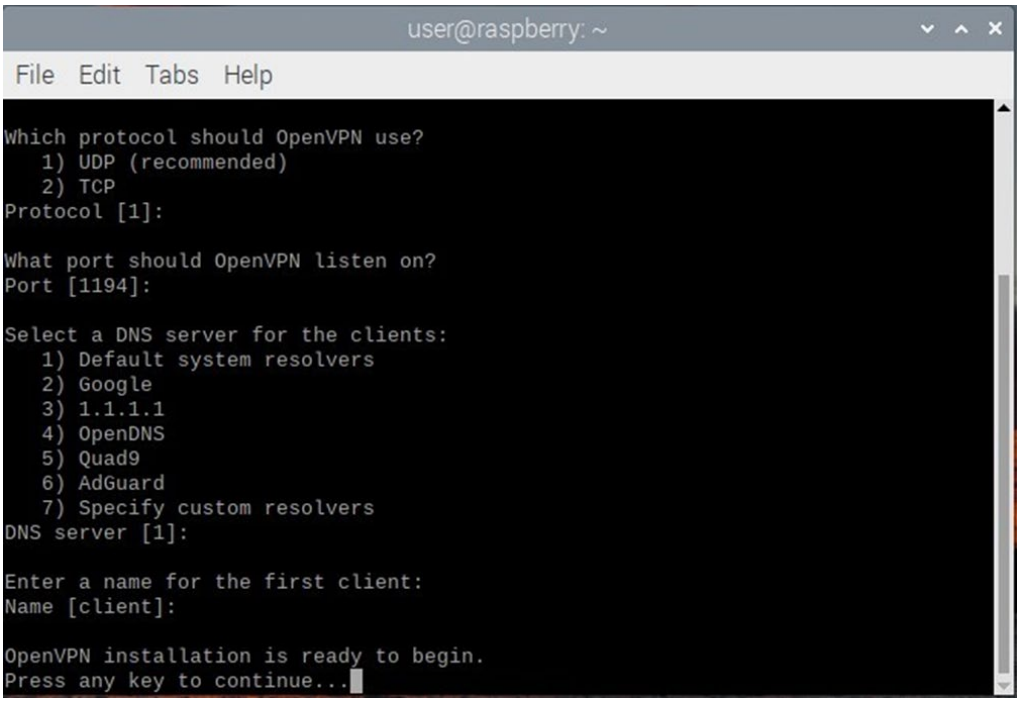


Рисунок 3.16 – Проставляю характеристики за замовчуванням

Файл client.ovpn містить усі необхідні дані для встановлення з’єднання з сервером. Він включає у себе налаштування тунелю, інформацію про сертифікати, параметри шифрування та IP-адресу VPN-сервера (папка з цим файлом зображена на рисунку 3.18). Оскільки віртуальна машина з Raspbian не підтримує гостьові доповнення VirtualBox, безпосередній обмін файлами між хостом і гостьовою системою ускладнений. Як альтернативний підхід було використано передавання файлу client.ovpn через хмарне сховище, зокрема Google Drive. Це дозволило завантажити файл на основну машину та використати його на стороні клієнта.

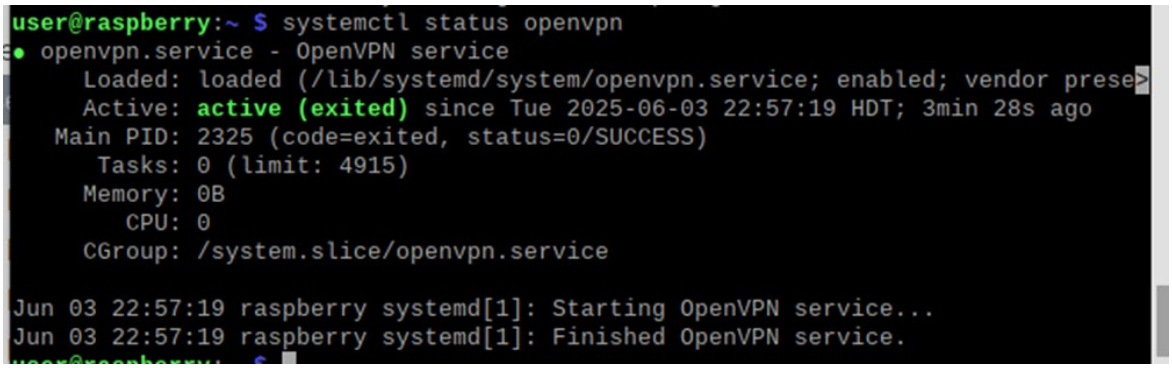


Рисунок 3.17 – Перевірка роботи

На клієнтському пристрої (в даному випадку на основному комп'ютері) виконується встановлення офіційного OpenVPN-клієнта, який завантажується з вебсайту OpenVPN. Після завершення інсталяції відкривається файл client.ovpn у відповідному інтерфейсі клієнтської програми (саме цей етап зображено на рисунку 3.19). Це дозволяє ініціювати з'єднання з сервером, розгорнутим у віртуальній машині. Успішне з'єднання підтверджується повідомленням про встановлення тунелю та отриманням IP-адреси у межах VPN.

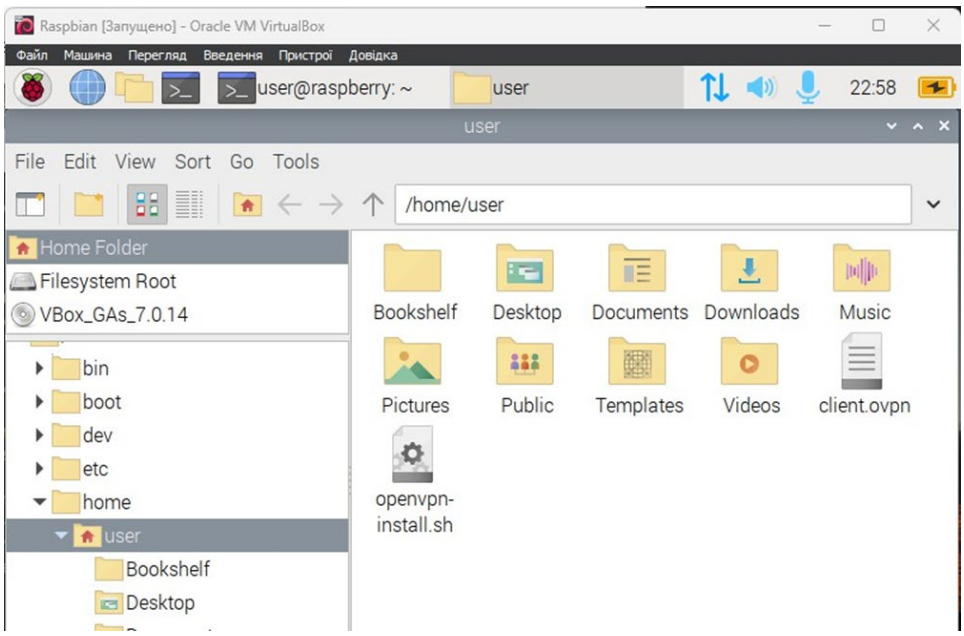


Рисунок 3.18 – Пошук client.ovpn

Даний підхід демонструє ефективність використання віртуальних машин у ролі середовища розгортання безпечної інфраструктури. Простота інтеграції, відкритий код та активна підтримка з боку спільноти роблять OpenVPN одним із провідних рішень для побудови VPN. Крім того, автоматизація налаштувань за допомогою скрипта openvpn-install.sh значно знижує поріг входу для нових користувачів, дозволяючи сфокусуватися на аналізі безпеки та налаштуванні політик доступу замість вирішення низькорівневих технічних проблем.

Рисунок 3.19 – Під’єдную client.ovpn з основного комп’ютера

У межах проєкту було успішно реалізовано повний цикл розгортання VPN-сервера: від інсталяції операційної системи до тестування підключення з клієнтського пристрою. Це дозволило створити повноцінне лабораторне середовище для подальшого аналізу та моделювання захищеного мережевого обміну, що є актуальним в умовах сучасної кіберзагрози та віддаленої роботи. У результаті підтверджено працездатність і надійність OpenVPN як компонента програмно-технічного засобу для побудови інформаційно-безпечних систем.

3.6. Висновки до третього розділу

Отже, на першому етапі було детально досліджено принцип роботи програмно-технічного засобу. Проаналізовано ключові архітектурні компоненти системи, включаючи апаратну платформу Raspberry Pi 4, програмне забезпечення, мережеву інфраструктуру та підсистему безпеки. Було встановлено, що функціонування VPN-сервера ґрунтується на створенні зашифрованого тунелю, що забезпечується послідовними етапами ініціалізації з'єднання, автентифікації клієнта, встановлення криптографічного каналу та безпечної передачі даних. Цей аналіз сформував необхідну теоретичну базу для подальшої практичної реалізації.

На другому етапі була здійснена апаратна реалізація програмно-технічного засобу. На основі проведеного аналізу було змодельовано прототип пристрою. В якості центрального елемента було обрано аналог одноплатного комп'ютера Raspberry Pi4, який було оснащено необхідними периферійними компонентами.

Завершальним етапом практичної роботи стало налаштування та розгортання VPN-сервера у віртуальному середовищі. Було виконано встановлення операційної системи Raspbian, інстальовано та налаштовано програмне забезпечення VPN-сервера, згенеровано конфігураційні файли для сервера та клієнтів, а також налаштовано правила мережевого екрана для забезпечення безпеки та маршрутизації трафіку.

ВИСНОВКИ

У даній роботі було проведено комплексне дослідження, розробку та апробацію програмно-технічного засобу VPN-сервера на основі одноплатної комп'ютерної системи Raspberry Pi 4. Робота послідовно охопила всі етапи створення продукту: від аналізу предметної області та постановки задачі до практичної реалізації та тестування.

У першому розділі було проведено глибокий аналіз предметної області, що дозволило виявити актуальні проблеми та завдання у сфері захисту даних та приватності в мережі. Через порівняльний аналіз переваг та недоліків існуючих комерційних та самостійно розгорнутих VPN-рішень було обґрунтовано необхідність створення доступного, гнучкого та безпечного інструменту. На основі цього аналізу було чітко сформульовано мету та задачі дослідження, спрямовані на удосконалення підходів до побудови персональних VPN-серверів.

У другому розділі було надано детальне обґрунтування вибору компонентів та середовища реалізації. Було доведено доцільність використання Raspberry Pi 4 як апаратного середовища завдяки його високій продуктивності, енергоефективності та низькій вартості. На основі аналізу були сформульовані чіткі функційні вимоги до системи (підтримка сучасних протоколів, автентифікація користувачів, керування трафіком) та нефункційні вимоги (безпека, надійність, простота розгортання). Цей етап заклав архітектурну та технічну основу для майбутньої розробки.

У третьому розділі було безпосередньо реалізовано програмно-технічний засіб. Було детально описано принцип його роботи, що включає створення захищеного тунелю та шифрування даних. Практична частина включала апаратну реалізацію — моделювання плати, та програмну — налаштування й розгортання VPN-сервера у віртуальному середовищі для безпечного тестування.

Таким чином, у ході виконання роботи було успішно пройдено всі заплановані етапи. Поставлені на початку роботи завдання були виконані.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Що таке VPN, і як це працює? URL: https://gridinsoft.ua/vpn (дата звернення: 02.05.2025)
2. Одноплатний комп'ютер Raspberry Pi 4 URL: https://www.kosmodrom.ua/news/odnoplatniy-komp-yuter-raspberry-pi-4-model-b.html (дата звернення: 02.05.2025)
3. What Is Dynamic DNS (DDNS)? -How it Works and Why Use It? URL: https://download.zone/what-is-dynamic-dns/ (дата звернення: 02.05.2025)
4. Configuration examples l2tp over ipsec scheme URL: https://wiki.teltonika-networks.com/view/File:Configuration_examples_l2tp_over_ipsec_scheme.png (дата звернення: 02.05.2025)
5. What is OpenVPN Protocol URL: https://www.vpnunlimited.com/help/vpn-protocols/open-vpn-protocol (дата звернення: 03.06.2025)
6. E0615: VPN Site-to-Site con WireGuard URL: https://www.desdeelreloj.com/e0615/ (дата звернення: 03.06.2025)
7. Configuring IPsec Using Manually Keyed Security Associations URL: https://techdocs.f5.com/kb/en-us/products/big-ip_ltm/manuals/product/bigip-tmos-tunnels-ipsec-11-6-0/9.html (дата звернення: 02.05.2025)
8. Gentile A. F., Macrì D., De Rango F., Tropea M., Greco E. A VPN performances analysis of constrained hardware open source infrastructure deploy in IoT environment. *Future Internet.* 2022. Vol. 14(9). P. 264.
9. Chua C. H., Ng S. C. Open-Source VPN Software: Performance Comparison for Remote Access. *Proceedings of the 5th International Conference on Information Science and Systems.* 2022. August. P. 29–34.
10. Islam M. Z., Khan M. A. R., Hossain M. I., Hossain R. Analysis the importance of VPN for Creating a Safe Connection Over the World of Internet. *International Journal of Advanced Research in Computer and Communication Engineering.* 2021. Vol. 10(10). P. 86–92.

11. Ezra P. J., Misra S., Agrawal A., Oluranti J., Maskeliunas R., Damasevicius R. Secured communication using virtual private network (VPN). *Cyber security and digital forensics: proceedings of ICCSDF 2021*. 2022. P. 309–319.

12. Abbas H., Emmanuel N., Amjad M. F., Yaqoob T., Atiquzzaman M., Iqbal Z., Ashfaq U. Security assessment and evaluation of VPNs: a comprehensive survey. *ACM Computing Surveys.* 2023. Vol. 55(13s). P. 1–47.

13. Huseynov E. Passwordless VPN using FIDO2 Security Keys: Modern authentication security for legacy VPN systems. 2022 *4th International Conference on Data Intelligence and Security (ICDIS).* 2022. August. P. 01–03.

14. Khan A., Gupta D., Dutta M. Fortifying Public Area Networks: Leveraging Ensemble Learning to Combat VPN Malicious Transmissions. *2024 3rd International Conference for Advancement in Technology (ICONAT).* 2024. September. P. 1–6.

15. Радченко І., Шеховцов О., Коваленко А., Ситник О. Формування кластерів на одноплатних комп'ютерах в IoT мережах. *Системи управління, навігації та зв'язку.* Збірник наукових праць. 2024. Вип. 2(76). С. 141–143.

16. Kanagachidambaresan G. R. *Role of Single Board Computers (SBCs) in Rapid IoT Prototyping.* Cham, Switzerland : Springer, 2021. P. 1–18.

17. Sinchangreed V., Watanyulertsakul E., Onrit S. Web services performance evaluation on single board computers for mobile applications and IoT devices. *Suranaree Journal of Science and Technology.* 2022. Vol. 29(2). P. 1-7

18. Shwetha N., Raghu J., Sahas V. G., Khan M., Siddesh G. M., Vinay H. N. Virtual private network (VPN) router using raspberry-PI. *International Journal of Engineering and Management Research.* 2023. Vol. 13(3). P. 209–213.

19. Hemachandran A. S., Nithyabharathi S. Compact Raspberry PI Server with Integrated UPS. *2024 International Conference on Sustainable Communication Networks and Application (ICSCNA).* 2024. December. P. 288–293.

20. Pamungkas D. B. P., Isnawaty I., Aksara L. F. Implementation of Samba Server Using OpenVPN Based on Single Board Computer (SBC) for Private Cloud Storage. *Journal of Applied Informatics and Computing.* 2024. Vol. 8(2). P. 316–325.

21. Naaman D. W., Rasheed B. H., Ahmed B. T., Salih A. Y., Mustafa S. H. Design a real-time communication system using 3cx software-based private branch exchange phone system on raspberry pi device. *Asian Journal of Research in Computer Science.* 2022. Vol. 13(4). P. 34–45.

22. He W. *Design of supervisory controllers and ultra-low power data loggers for hybrid power systems.* Doctoral dissertation, Memorial University of Newfoundland, 2024. P. 195

23. Steadman P., Jenkins P., Rathore R. S., Hewage C. Challenges in Implementing Artificial Intelligence on the Raspberry Pi 4, 5 and 5 with AI HAT. *The International Conference on Computing, Communication, Cybersecurity & AI.* Cham : Springer Nature Switzerland, 2024. July. P. 147–157.

24. Radu F., Tuyishime E., Cotfas P., Cotfas D., Balan T., Rekeraho A. Online Serial Laboratories. *Online Laboratories in Engineering and Technology Education: State of the Art and Trends for the Future.* Cham : Springer Nature Switzerland, 2025. P. 375–390.

25. Ortega G., Moreno J. J., Garzón E. M., Orts F. J., Donaire L. M., López L. O., Redondo A. DOMOTIC-BASED SETUPS USING A RASPBERRY PI TO INCREASE INTEREST IN COMPUTER ENGINEERING. *ICERI2023 Proceedings.* IATED, 2023. P. 8472–8477.

26. He W., Baig M. J. A., Iqbal M. T. An Open-Source Supervisory Control and Data Acquisition Architecture for Photovoltaic System Monitoring Using ESP32, Banana Pi M4, and Node-RED. *Energies.* 2024. Vol. 17(10). P. 2295.

27. Pitz M., Wege F., Eiling N., Vogel S., Bareis V., Monti A. Automated Deployment of Single-Board Computer Based Grid Measurement and Co-Simulation Equipment. *2024 Open Source Modelling and Simulation of Energy Systems (OSMSES).* 2024. September. P. 1–6.

28. Souza H. C. D. *Implantação de firewall em Raspberry Pi.* 2023.

29. Angelov K., Kogias P., Manchev N., Sadinov S. Development of a Universal Integrated Monitoring System for Communication Networks. *2024 5th International*

Conference on Communications, Information, Electronic and Energy Systems (CIEES). 2024. November. P. 1–5.

30. Bianconi L. *Towards automation of TLS-based VPN configuration.* Doctoral dissertation, Politecnico di Torino, 2023. P. 1-88

31. Damasco Jr L. B., Lacson N. R. F., Intong R. T. Safeguarding transactional data: A comprehensive investigation of VPN integration in point-of-Sale systems for enhanced security and connectivity. *Eximia.* 2023. Vol. 12. P. 570–589.

32. Zhang T., Gong B. Design and implementation of Android VPN client based on GMSSL. *International Conference on Computer, Artificial Intelligence, and Control Engineering (CAICE 2022).* SPIE, 2022. December. Vol. 12288. P. 52–64.

33. Hauser F., Häberle M., Menth M. P4sec: Automated deployment of 802.1 X, IPsec, and MACsec network protection in P4-based SDN. *IEEE Access.* 2023. Vol. 11. P. 56300–56309.

34. Prakash V., Jain C., Rathi R., Garg L., Shukla V. Setting up an OpenVPN Server on the Google Cloud Platform. *International Conference on Cryptology & Network Security with Machine Learning.* Singapore : Springer Nature Singapore, 2023. October. P. 675–686.

35. Barguil S., Lopez V., Manta-Caro C., De Lerma A. M. L., De Dios O. G., Echeverry E., Vilalta R. Field trial of programmable l3 vpn service deployment using sdn-based multi-domain service provisioning over ip/optical networks. *IEEE Network.* 2021. Vol. 35(6). P. 217–224.

36. Alemany Prats P. *Quality of service, security and trustworthiness for network slices.* 2023. P.1-198

37. Manso C., Vilalta R., Gifre L., Casellas R., Martínez R., Munoz R. Introducing end-to-end location awareness in packet-optical networks. *IET Conference Proceedings CP839.* Stevenage, UK : The Institution of Engineering and Technology, 2023. October. Vol. 2023(34). P. 60–63.

38. Gao H., Chen X., Zheng W., Chen Y., Xiao Y., Li Z. Demonstration of composable-ML-assisted autonomous lightpath configuration over a field-deployed SDM

network with 7-core fibers. *Optical Fiber Communication Conference.* Optica Publishing Group, 2023. March. P. Th4C–2.

39. Hajipour S., Hamza M., Renom L. G., Manso C., Casellas R., Martínez R., Vilalta R. Network Resource Allocation for Gaming Using MEC API and TeraFlowSDN. *2024 15th International Conference on Network of the Future (NoF).* 2024. October. P. 19–21.

40. Ariza J. A., Baez H. Understanding the role of single-board computers in engineering and computer science education: A systematic literature review. *Computer Applications in Engineering Education.* 2022. Vol. 30. Iss. 1. P. 304–329.

41. Lee E., Oh H., Park D. Big data processing on single board computer clusters: Exploring challenges and possibilities. *IEEE access.* 2021. Vol. 9. P. 142551–142565.

42. Álvarez J. L., Mozo J. D., Durán E. Analysis of single board architectures integrating sensors technologies. *Sensors.* 2021. Vol. 21(18). P. 6303.

43. Justus V., Kanagachidambaresan G. R. Intelligent single-board computer for Industry 4.0: Efficient real-time monitoring system for anomaly detection in CNC machines. *Microprocessors and Microsystems.* 2022. Vol. 93. P. 104629.

44. Coelho P., Bessa C., Landeck J., Silva C. The Potential of Low-Power, Cost-Effective Single Board Computers for Manufacturing Scheduling. *Procedia Computer Science.* 2023. Vol. 217. P. 904–911.

45. Hoffmann R. B., Griebler D., da Rosa Righi R., Fernandes L. G. Benchmarking parallel programming for single-board computers. *Future Generation Computer Systems.* 2024. Vol. 161. P. 119–134.

46. Sánchez P. M. S., Valero J. M. J., Celdrán A. H., Bovet G., Pérez M. G., Pérez G. M. A methodology to identify identical single-board computers based on hardware behavior fingerprinting. *Journal of Network and Computer Applications.* 2023. Vol. 212. P. 103579.

47. Lambropoulos G., Mitropoulos S., Douligeris C., Maglaras L. Implementing Virtualization on Single-Board Computers: A Case Study on Edge Computing. *Computers.* 2024. Vol. 13(2). P. 54.

48. Loubet P., Vincent A., Collin A., Dejous C., Ghiotto A., Jego C. Life cycle assessment of ICT in higher education: a comparison between desktop and single-board computers. *The International Journal of Life Cycle Assessment.* 2023. Vol. 28(3). P. 255–273.

49. Piccolo C., Foster S. W., Parker D., Seltzer C., Grinias J. P. The utilization of open-source microcontroller boards and single-board computers in liquid chromatography. *Journal of Separation Science.* 2024. Vol. 47(9-10). Art. no. 2400111.

50. Wai E., Lee C. K. M. Seamless industry 4.0 integration: A multilayered cyber-security framework for resilient scada deployments in cpps. *Applied Sciences.* 2023. Vol. 13(21). P. 12008.

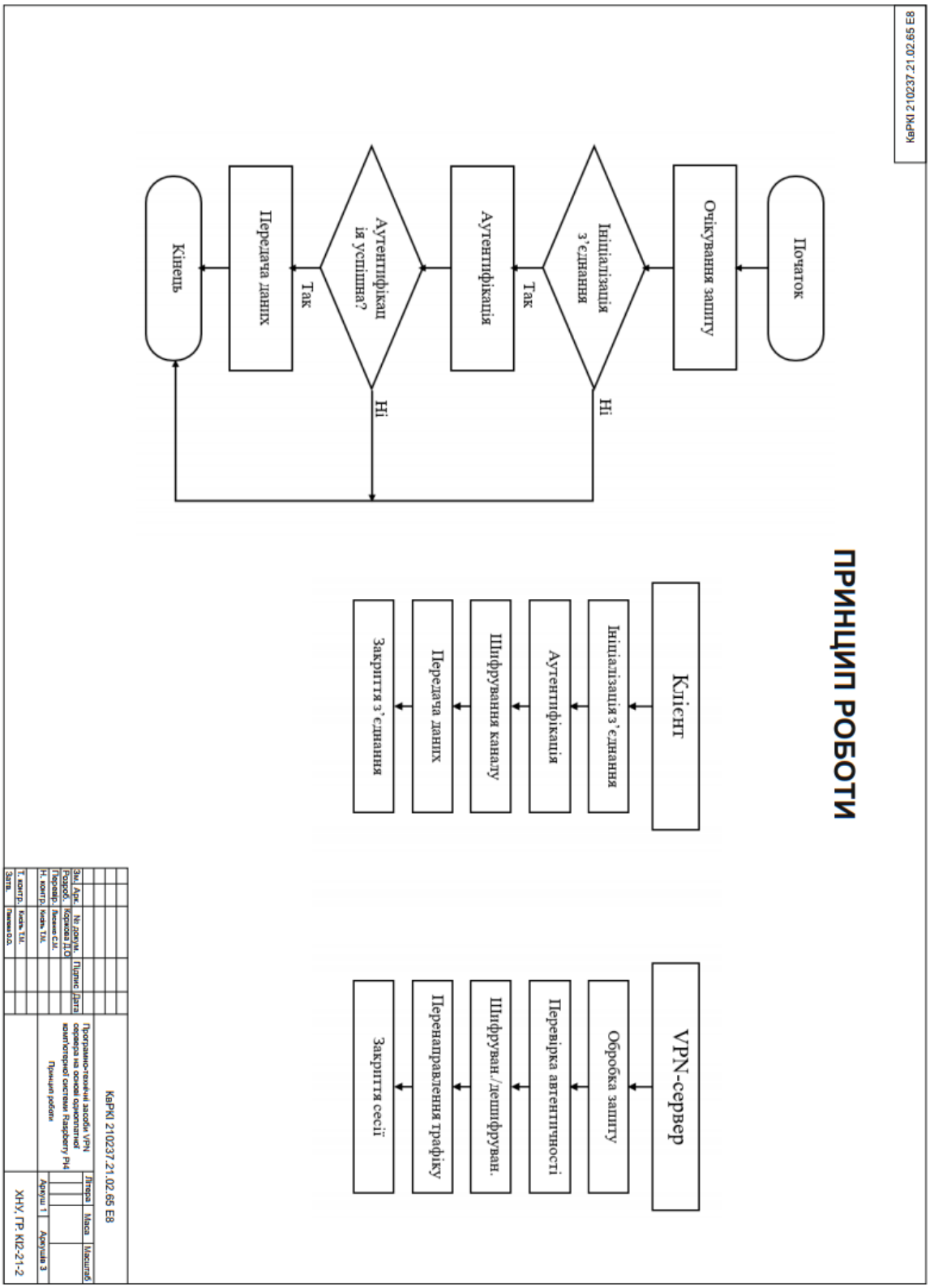
51. Bruno S., Giannoccaro G., Islam M. M., Iurlaro C., La Scala M., Menga M., Rodio C. Control and Power Hardware-in-the-Loop tests for low-inertia power systems. *2022 AEIT International Annual Conference (AEIT).* IEEE, 2022. October. P. 1–6.

52. Fadhilla C. A., Alfikri M. D., Kaliski R. Lightweight meta-learning BotNet attack detection. *IEEE Internet of Things Journal.* 2022. Vol. 10(10). P. 8455–8466.

Додаток А

(обов’язковий)

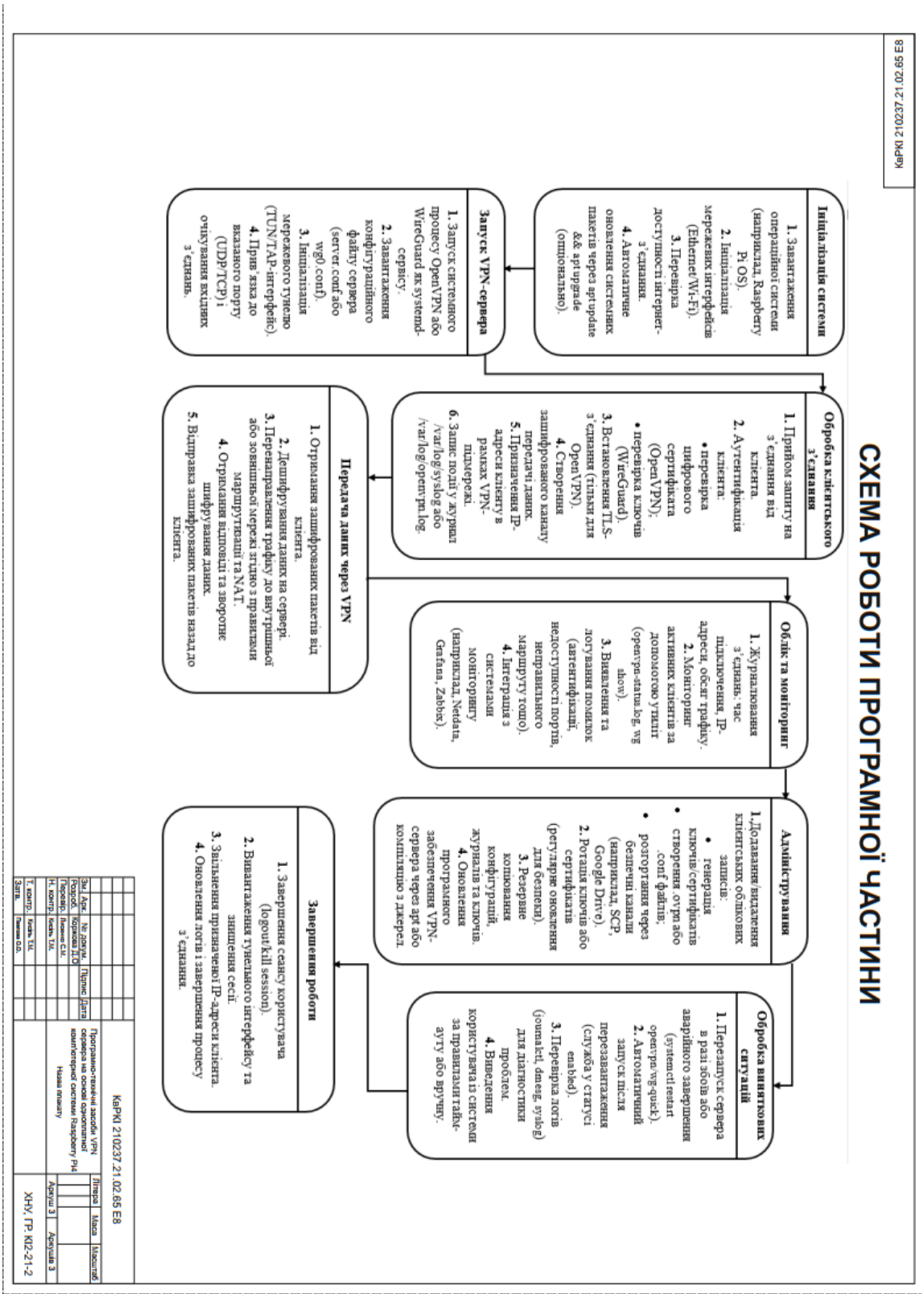
КОПІЯ КРЕСЛЕННЯ «ПРИНЦИП РОБОТИ»



Додаток Б

(обов’язковий)

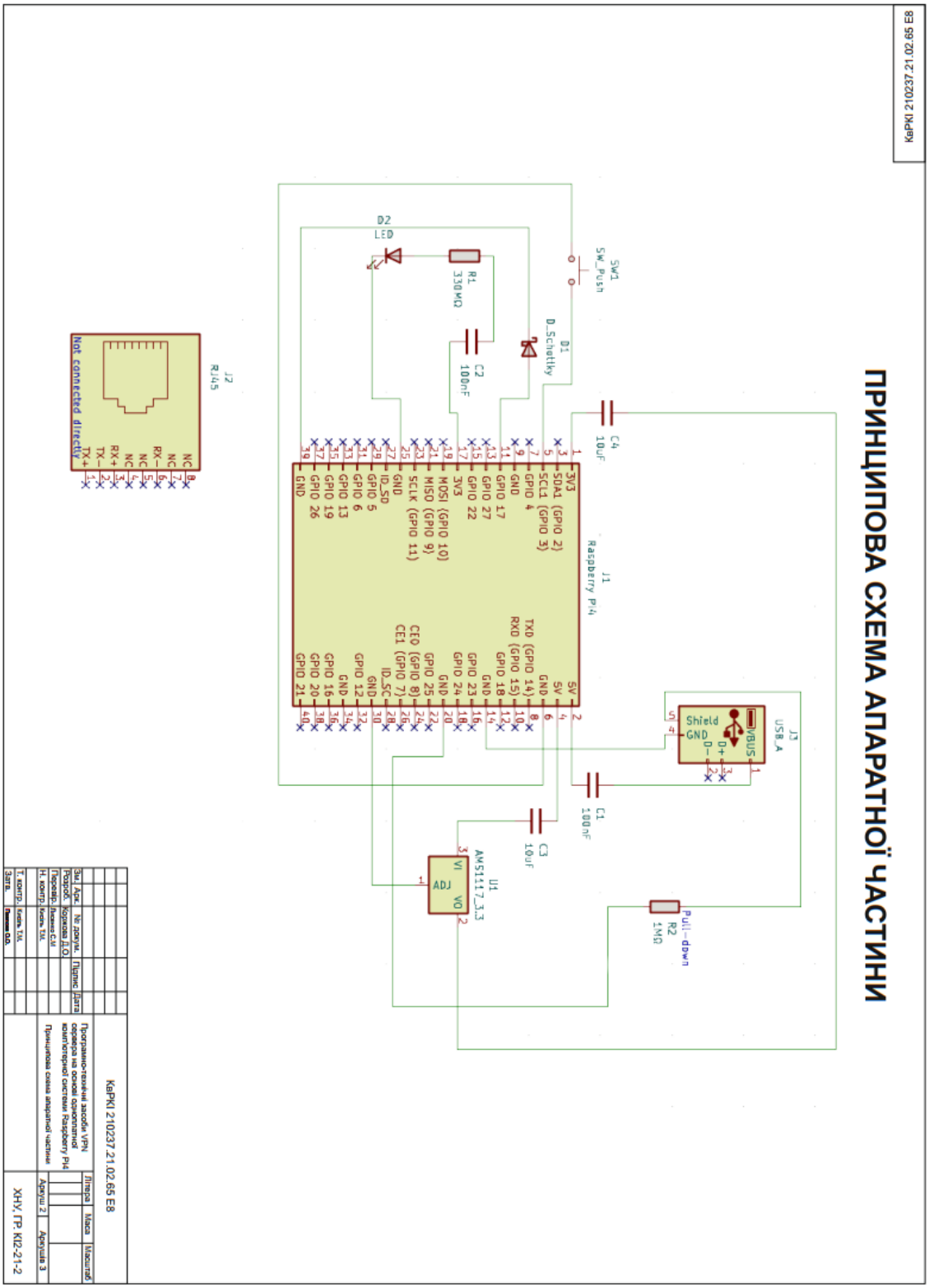
КОПІЯ КРЕСЛЕННЯ «СХЕМА РОБОТИ ПРОГРАМНОЇ ЧАСТИНИ»



Додаток В

(обов’язковий)

КОПІЯ КРЕСЛЕННЯ «ПРИНЦИПОВА СХЕМА АПАРАТНОЇ ЧАСТИНИ»



Протокол аналізу звіту подібності експертом

Заявляю, що я ознайомився (-лась) з Повним звітом подібності, який був згенерований Системою виявлення і запобігання плагіату щодо роботи:

Автор: Дар'я КОРЖОВА

Співавтор:

Назва: Коржова_Програмно-технічні засоби VPN сервера на основі одноплатної комп'ютерної системи Raspberry Pi4

Експерт:

Підрозділ: Кафедра комп'ютерної інженерії та інформаційних систем

Коефіцієнт подібності 1:4.8%

Коефіцієнт подібності 2:1.7%

Мікропробіли: 6

Заміна букв: 3

Інтервали: 0

Білі знаки: 0

Дата створення звіту: 2025-06-09 11:44:02.0

Після аналізу Звіту подібності констатую наступне:

☒ **Запозичення, виявлені в роботі є законними і не є плагіатом. Рівень подібності не перевищує допустимої межі. Таким чином робота незалежна і приймається.**

☐ **Запозичення не є плагіатом, але перевищено граничне значення рівня подібностей. Таким чином робота повертається на доопрацювання.**

☐ **Виявлено запозичення і плагіат або навмисні текстові спотворення (маніпуляції), як передбачувані спроби укриття плагіату, які роблять роботу невідповідною вимогам законодавства (Ст. 32. ЗУ Про вищу освіту, пункт 3.1, Ст. 42. ЗУ Про освіту) та вимог НАЗЯВО (Критерій 5), а також кодексу етики і процедурам. Таким чином робота не приймається.**

Обгрунтування:

2025-06-09 Доцент Андрій Нічепорук

Дата експерт

Anti-Plagiarism (UA) v-15.281 Educational

The maximum coincidence with one document 1.0%

Dictionaries check: en_US, ru_RU, ua_UA. **Errors in the documents: 11%**

	Document		Sum coincidence on the DB	
ID: 244288 Title: БКР Програмно-технічні засоби VPN сервера на основі одноплатної комп'ютерної системи Raspberry Pi4 Added in a DB: 2025-06-09 Authors: Дар'я КОРЖОВА Heads: Сергій ЛИСЕНКО Consultants: Opponents:	Symbols	Lexemes	Symbols	Lexemes
	93601	665	2332 (2%)	33 (5%)

Plagiarism sources

ID	Description	Plagiarism presence in the document	
		Symbols	Lexemes

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

РЕЦЕНЗІЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

Дипломник: Коржова Дар'я Олександрівна

Тема: Програмно-технічні засоби VPN сервера на основі одноплатної комп'ютерної системи Raspberry Pi4

Спеціальність: 123 «Комп'ютерна інженерія»

Обсяг кваліфікаційної роботи:

Кількість листів креслень ___3___ Кількість сторінок записки ___59___

1. Короткий зміст роботи та прийнятих рішень: Метою кваліфікаційної роботи програмно-технічний засіб VPN сервера на основі одноплатної комп'ютерної системи Raspberry Pi4

2. Висновок про відповідність роботи дипломному завданню: Робота повністю відповідає поставленому завданню.

3. Характеристика виконання кожного розділу, ступінь використання останніх досягнень науки і техніки і передових методів роботи: У першому розділі було проведено глибокий аналіз предметної області, що дозволило виявити актуальні проблеми та завдання у сфері захисту даних та приватності в мережі. Через порівняльний аналіз переваг та недоліків існуючих комерційних та самостійно розгорнутих VPN-рішень було обґрунтовано необхідність створення доступного, гнучкого та безпечного інструменту. На основі цього аналізу було чітко сформульовано мету та задачі дослідження, спрямовані на удосконалення підходів до побудови персональних VPN-серверів.

У другому розділі було надано детальне обґрунтування вибору компонентів та середовища реалізації. Було доведено доцільність використання Raspberry Pi 4 як апаратного середовища завдяки його високій продуктивності, енергоефективності та низькій вартості. На основі аналізу були сформульовані чіткі функційні вимоги до системи (підтримка сучасних протоколів, автентифікація користувачів, керування трафіком) та нефункційні вимоги (безпека, надійність, простота розгортання). Цей етап заклав архітектурну та технічну основу для майбутньої розробки.

У третьому розділі було безпосередньо реалізовано програмно-технічний засіб. Було детально описано принцип його роботи, що включає створення захищеного тунелю та шифрування даних. Практична частина включала апаратну реалізацію — моделювання плати, та програмну — налаштування й розгортання VPN-сервера у віртуальному середовищі для безпечного тестування.

4. Позитивні сторони роботи: висока практична цінність роботи.

5. Негативні сторони роботи:

6. Оцінка графічного оформлення та пояснювальної записки роботи: Пояснювальна записка оформлена коректно, згідно діючих стандартів оформлення документації.

7. Відгук про роботу в цілому: Робота виконана на належному науково-технічному рівні.

8. Інші зауваження: _____

9. Оцінка дипломної роботи: добре

Рецензент (прізвище, ім'я, по батькові, посада, місце роботи) _____

Мартинюк Валерій Володимирович,
доц. каф. АКІТТеР, ХНУ

"9" 06 2025 р. _____ (підпис)

Завідувачу кафедри КІІС
д-р. філософії, доц. Ользі ПАВЛОВІЙ

Дар'ї КОРЖОВОЇ
ПІБ здобувача вищої освіти

ФІТ, 4 курсу, групи КІ2-21-2

ЗАЯВА

З правилами чинного Положення «Про систему забезпечення академічної доброчесності у Хмельницькому національному університеті» від 01.07.2022, згідно з яким виявлення плагіату є підставою для відмови в допуску кваліфікаційної роботи до захисту та застосування заходів дисциплінарної та академічної відповідальності, ознайомлений (а). Про використання програмно-технічних засобів для перевірки кваліфікаційних робіт здобувачів вищої освіти на плагіат оповіщений(а) та надаю свою згоду на обробку та збереження університетом моєї роботи в інституційному репозитарії університету.

Також надаю університету право на передачу моєї роботи для обробки та збереження в базах даних програмно-технічних засобів (Strike-Plagiarism та Anti-Plagiarism) та використання роботи для виявлення плагіату в інших роботах, які перевіряються програмно-технічними засобами та користувачами, що мають доступ до цих програмно-технічних засобів, виключно в обмежених цілях для виявлення плагіату в текстах робіт.

Робота для перевірки університетом надається в друкованому та електронному варіанті. Електронна версія моєї роботи збігається (ідентична) з друкованою.

09.06 2025 року

РІШЕННЯ ЕКСПЕРТНОЇ КОМІСІЇ
КАФЕДРИ КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ ТА ІНФОРМАЦІЙНИХ СИСТЕМ
ПРО ДОПУСК КВАЛІФІКАЦІЙНОЇ РОБОТИ ДО ЗАХИСТУ

Підтверджуємо ознайомлення з результатом звіту подібності щодо роботи, генерованого системою виявлення текстових збігів/ідентичності/схожості:

Назва: Програмно-технічні засоби VPN сервера на основі одноплатної комп'ютерної системи Raspberry Pi4

Автор: Дар'я КОРЖОВА

Спеціальність: 123– Комп'ютерна інженерія

Освітня програма: освітньо-професійна

Науковий керівник: Сергій ЛИСЕНКО, д.т.н, професор

Після аналізу звіту подібності зроблено такий висновок:

№	Висновок	Позначка про відповідність
1	Запозичення, виявлені в роботі, є законними і не є плагіатом. Робота приймається до захисту.	відповідає
2	Виявлені запозичення не є плагіатом, розміщені в розділах, які не описують безпосередньо авторське дослідження, але кількість цитат перевищує обсяг, виправданий поставленою метою роботи Робота приймається до захисту, але має бути відкоригована. Відкоригований варіант має бути поданий на кафедру за 2 дні до захисту, разом із заявою щодо самостійності виконання письмової роботи та ідентичності друкованої та електронної версії роботи	
3	Виявлені запозичення не є плагіатом, але частково розміщені в розділах, які описують безпосередньо авторське дослідження, а кількість цитат перевищує обсяг, виправданий поставленою метою роботи. В зв'язку з цим мета роботи та поставлені завдання не були досягнені. Робота може бути допущена до захисту (наступного року) після того як буде відкоригована та допрацьована і успішно пройде повторну перевірку на академічний плагіат.	
4	Робота містить навмисні текстові спотворення, передбачувані спроби укриття запозичень або інші прояви академічного плагіату. Робота містить фабрикацію або фальсифікацію даних. Робота не допускається до захисту.	

Підтвердження:

Запозичення, виявлені в роботі, є законними і не є плагіатом, оскільки:

1) запозичення розміщені у вступній частині, розділах аналізу аналогів, теоретичних основ і постановки задачі, які мають описово-оглядовий характер, базуються на загальнодоступній технічній інформації або літературних джерелах;
2) деякі зафіксовані збіги пов'язані з діаграмами, назвами таблиць або схемці елементи не становлять самостійного авторського контенту;
3) в роботі використано посилання на використані джерела, що свідчить про належне оформлення запозичень відповідно до академічних стандартів та вимог до написання кваліфікаційних робіт;
4) система зафіксувала незначні модифікації, які пов'язані з технічним оформленням формул або вживанням скорочень, зокрема комбінування латинських та українських символів, що не є свідомим приховуванням плагіату.

Сумарний обсяг всіх запозичень, визначений системою виявлення збігів/ ідентичності/схожості StrikePlagiarism, складає 4.76% і адресується до 33 першоджерел; та системою Anti-Plagiarism складає 1%, що, з урахуванням наведених обґрунтувань, відповідає характеру наукового дослідження і свідчить на користь кваліфікаційної роботи.

Керівник роботи _____ Сергій ЛИСЕНКО

Гарант ОП _____ Андрій Нічепорук

Завідувач кафедри КІС _____ Ольга ПАВЛОВА