НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ОДЕСЬКА ЮРИДИЧНА АКАДЕМІЯ»
ФАКУЛЬТЕТ КІБЕРБЕЗПЕКИ ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Кафедра кібербезпеки

КВАЛІФІКАЦІЙНА РОБОТА

на тему:

«АНОНІМІЗАЦІЯ ТА ЗАХИСТ ІНТЕРНЕТ-ТРАФІКУ З ВИКОРИСТАННЯМ ТЕХНОЛОГІЙ VPN ТА TOR: ПОРІВНЯЛЬНІЙ АНАЛІЗ»

Виконав: здобувач 4 курсу
Рівень бакалавр
Галузь знань 12 «Інформаційні технології»,
спеціальність 125 «Кібербезпека»
Колєв Максим Васильович

Керівник: доктор філософії
Слатвінська Валерія Миколаївна

Рецензент: к.т.н., доцент
Ахмаметьєва Ганна Валеріївна

Одеса – 2025

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ОДЕСЬКА ЮРИДИЧНА АКАДЕМІЯ»
Факультет **кібербезпеки та інформаційних технологій**
Кафедра **кібербезпеки**
Галузь знань: **12 Інформаційні технології**
Спеціальність: **125 Кібербезпека**
Назва освітньої програми: **Кібербезпека**

ЗАТВЕРДЖУЮ
Завідувач кафедри кібербезпеки
професор С.А. Горбаченко
_____ «____ »_____ 20__ року

З А В Д А Н Н Я
на кваліфікаційну (бакалаврську) роботу
здобувачу Колєву Максиму Васильовичу

1. Тема роботи: «Анонімізація та захист інтернет-трафіку з використанням технологій VPN та TOR: порівняльній аналіз»
Керівник роботи: доктор філософії Слатвінська Валерія Миколаївна
затверджені наказом НУ ОЮА від «18» березня 2025 року № 548-6
2. Строк подання здобувачем роботи «19» травня 2025 року
3. Дата видачі завдання «16» вересня 2024 року

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів бакалаврської роботи	Строк виконання етапів роботи	Примітка

Здобувач_____
(підпис) (прізвище та ініціали)
Керівник роботи_____
(підпис) (прізвище та ініціали)

АНОТАЦІЯ

Кваліфікаційна робота на тему «Анонімізація та захист інтернет-трафіку з використанням технологій VPN та TOR» на здобуття першого (бакалаврського) рівня вищої освіти за спеціальністю 125 Кібербезпека, освітньою програмою: Кібербезпека, містить 15 рисунків, 2 додатки, 20 літературних джерел за переліком посилань. Робота виконана на 43 сторінках загального тексту і 41 сторінці основного тексту.

Метою роботи є налаштування свого VPN-сервера для захисту трафіка.

Розроблено метод захисту трафіка за допомогою VPS-сервера, протоколу WireGuard та Python. Програмно реалізовано клієнт для підключення до VPN-сервера. На основі розробленої системи проведено експериментальні дослідження роботи VPN-сервера.

Результати роботи можуть бути використані при створенні особистого VPN.

Можливими напрямками подальших досліджень є вдосконалення функціональних можливостей систем захисту трафіка.

КІБЕРБЕЗПЕКА, VPN, WIREGUARD, TOR, PYTHON, ЗАХИСТ ТРАФІКУ.

ABSTRACT

Qualification work on the topic «Anonymization and protection of Internet traffic using VPN and TOR technologies» for obtaining the first (bachelor's) level of higher education in the specialty 125 Cybersecurity, educational program: Cybersecurity, contains 15 figures, 2 appendices, 20 literary sources according to the list of references. The work is completed with 43 pages of general text and 41 pages of the main text.

The purpose of the work is to configure your own VPN server to protect traffic.

A method for protecting traffic using a VPS server, the WireGuard protocol and Python have been developed. A client for connecting to a VPN server has been implemented in software. Based on the developed system, experimental studies of the VPN server have been conducted.

The results of the work can be used when creating a personal VPN.

Possible areas of further research are improving the functionality of traffic protection systems.

CYBERSECURITY, VPN, WIREGUARD, TOR, PYTHON, TRAFFIC PROTECTION.

ЗМІСТ

ВСТУП

Актуальність теми обумовлена стрімким зростанням загроз у сфері кібербезпеки, посиленням контролю за інтернет-активністю користувачів та необхідністю забезпечення конфіденційності даних. У сучасному цифровому середовищі дедалі частіше виникають загрози зламу, витоку інформації, фішингових атак і незаконного стеження, що вимагає ефективних інструментів захисту персональної інформації.

Крім того, у багатьох країнах впроваджуються механізми державного контролю та цензури, що обмежують доступ до певних ресурсів. У таких умовах VPN та Tor стають ключовими технологіями, які дозволяють користувачам зберігати анонімність і обходити блокування.

Розвиток цифрових технологій, зокрема штучного інтелекту та методів аналізу трафіку, створює нові виклики у сфері конфіденційності, що підсилює потребу в ретельному аналізі ефективності різних підходів до захисту даних. В умовах зростаючої популярності дистанційної роботи, використання публічних мереж Wi-Fi та загальної діджиталізації суспільства питання безпечного з’єднання стає ще більш актуальним.

Мета дослідження - проведення порівняльного аналізу технологій VPN та Tor у контексті анонімізації та захисту інтернет-трафіку. Дослідження спрямоване на виявлення переваг і недоліків кожної з цих технологій, оцінку їхньої ефективності в забезпеченні конфіденційності, швидкості та безпеки передачі даних.

У межах дослідження передбачається розглянути принципи роботи VPN і Tor, їхні основні функціональні особливості, вразливості та потенційні ризики використання. Також буде проаналізовано, в яких випадках одна з технологій є більш доцільною для застосування, враховуючи потреби користувачів у захисті особистих даних, обході цензури та анонімності в мережі.

Завдання дослідження:

1. Проаналізувати принципи роботи технологій VPN і Tor, розглянути їхню архітектуру, механізми шифрування та маршрутизації інтернет-трафіку.

Дослідити рівень анонімності та безпеки, який забезпечують VPN та Tor, визначити їхні сильні та слабкі сторони у контексті захисту персональних даних користувачів.

2. Оцінити вплив цих технологій на швидкість та стабільність інтернет-з'єднання, порівняти їхню ефективність для повсякденного використання, включаючи перегляд веб-сторінок, потокове відео, завантаження файлів тощо.

3. Виявити потенційні ризики та вразливості, пов’язані з використанням VPN і Tor.

4. Порівняти VPN і Tor за ключовими параметрами, визначити їхню доцільність у різних сценаріях використання (обхід цензури, захист конфіденційності, безпека у публічних мережах тощо).

Об’єкт дослідження – є процеси анонімізації та захисту інтернет-трафіку з використанням технологій VPN і Tor. Дослідження зосереджене на механізмах забезпечення конфіденційності, принципах функціонування цих технологій, а також на їхньому впливі на безпеку користувачів у цифровому середовищі.

Предмет дослідження – порівняльний аналіз технологій VPN та Tor у контексті забезпечення анонімності та захисту інтернет-трафіку. Дослідження охоплює їхні технічні характеристики, принципи роботи, рівень безпеки та конфіденційності, швидкість і стабільність з’єднання, а також можливі вразливості та ризики. Окрему увагу приділено ефективності цих технологій у різних сценаріях використання, їхньому впливу на захист персональних даних користувачів та здатності протистояти стеженню, цензурі та кіберзагрозам.

Основні висновки бакалаврської роботи були опубліковані та апробовані у тезах «VPN та TOR як сучасні технології анонімізації та захисту інтернет-трафіку» на науковій конференції «VIII Міжнародна науково-практична конференція «Modern trends of social transformations of society in conditions of sustainable development»».

1. АНАЛІЗ СУЧАСНИХ ТЕХНОЛОГІЙ АНОНІМІЗАЦІЇ ТА ЗАХИСТУ ІНТЕРНЕТ-ТРАФІКУ

1.1 Технологія VPN: принципи роботи, переваги та недоліки

Virtual Private Network (VPN) – це технологія, що забезпечує безпечне та зашифроване з’єднання через менш захищені мережі, зазвичай Інтернет. Вона дозволяє користувачам передавати дані конфіденційно та безпечно, ніби їхні пристрої безпосередньо підключені до приватної мережі. VPN маскує IP-адресу користувача та шифрує трафік, гарантуючи анонімність і захист даних.

Сьогодні VPN широко використовується як у корпоративному, так і в особистому середовищі. Підприємства застосовують цю технологію для безпечного доступу віддалених співробітників до внутрішніх ресурсів компанії, тоді як індивідуальні користувачі використовують VPN для захисту конфіденційності в Інтернеті, безпеки даних у загальнодоступних Wi-Fi мережах і обходу географічних обмежень контенту.

VPN розширює приватну мережу через загальнодоступну, створюючи віртуальний зашифрований тунель між пристроєм користувача та VPN-сервером [1].

VPN-тунелювання – це ключова технологія, що забезпечує безпечну передачу даних через загальнодоступні мережі, такі як Інтернет. Воно створює захищений «тунель» для пакетів даних, гарантуючи їхню конфіденційність і цілісність. Це робить VPN-тунелювання незамінним інструментом як для організацій, так і для окремих користувачів, які потребують надійного та захищеного зв’язку через потенційно вразливі мережі.

Існує два типи тунелювань:

Добровільне тунелювання – ініціюється клієнтським пристроєм, наприклад, коли віддалений користувач підключається до корпоративної мережі через VPN-клієнт. У цьому випадку користувач самостійно встановлює з'єднання з VPN-сервером для захисту свого інтернет-трафіку.

Примусове тунелювання – ініціюється мережевим пристроєм, таким як VPN-шлюз. У цьому випадку весь інтернет-трафік користувача автоматично перенаправляється через VPN-тунель без потреби в ручному підключенні. Це забезпечує централізований контроль і підвищену безпеку для корпоративних або урядових мереж.

Тунелювання VPN передбачає кілька основних процесів, які забезпечують безпечний зв’язок:

1. Інкапсуляція: це процес загортання пакетів даних в інший рівень пакетів. Інкапсуляція є фундаментальною для VPN-тунелювання, оскільки вона дозволяє передавати пакети даних приватної мережі через загальнодоступні мережі. Процес інкапсуляції передбачає розміщення оригінальних пакетів даних у нових пакетах, які надають інформацію про маршрутизацію, необхідну для проходження загальнодоступної мережі.

2. Шифрування: після інкапсуляції даних вони шифруються для запобігання несанкціонованому доступу. Шифрування перетворює читабельні дані (відкритий текст) у закодовану форму (зашифрований текст), яку можуть декодувати лише авторизовані сторони. Кожен пакет, який проходить через тунель VPN, шифрується, що гарантує, що навіть у разі перехоплення дані залишаються конфіденційними.

3. Автентифікація: мережі VPN повинні автентифікувати пристрої на обох кінцях тунелю, щоб переконатися, що дані надсилаються та отримуються з надійного джерела. Цю автентифікацію можна досягти за допомогою різних засобів, таких як паролі, цифрові сертифікати або передові криптографічні методи. Автентифікація надає спосіб підтвердити особу сторін, що спілкуються, а також може включати автентифікацію користувача для подальшого підвищення безпеки.

4. Протоколи тунелювання: для створення та керування тунелем VPN використовуються різні протоколи. Ці протоколи визначають, як пакети інкапсулюються, шифруються та передаються через тунель [2]. Існують такі протоколи, як Internet Protocol Security (IPSec), протокол тунелювання рівня 2 (L2TP), протокол тунелювання «точка-точка» (PPTP), SSL, TLS, OpenVPN, Secure Shell (SSH).

Internet Protocol Security (IPSec) – це набір протоколів, що забезпечує захист Інтернет-з’єднання шляхом перевірки сеансу та шифрування кожного пакета даних. Він працює у двох режимах:

Транспортний режим – шифрує лише вміст пакета даних, залишаючи заголовки відкритими.

Тунельний режим – шифрує весь пакет даних, що забезпечує вищий рівень безпеки.

IPSec часто використовується у поєднанні з іншими протоколами для покращення безпеки VPN-з’єднань.

Протокол тунелювання рівня 2 (L2TP) – це протокол тунелювання, який зазвичай працює разом з IPSec для створення захищених VPN-з’єднань. L2TP формує тунель між двома кінцевими точками підключення, а IPSec забезпечує шифрування та захист переданих даних.

Протокол тунелювання «точка-точка» (PPTP) створює VPN-тунель і використовує протокол PPP (протокол «точка-точка») для шифрування даних. Це один із найдавніших і найпоширеніших VPN-протоколів, підтримуваний Windows, MacOS і Linux. Однак через слабші механізми безпеки PPTP вважається менш захищеним порівняно з іншими протоколами.

Протоколи SSL і TLS використовуються для створення VPN-з’єднання, у якому веб-браузер виконує роль клієнта. Вони зазвичай застосовуються для забезпечення безпечного доступу до веб-додатків, наприклад, в онлайн-магазинах. Веб-браузери автоматично підтримують SSL і TLS, що робить їх зручними для користувачів. Ознакою безпечного з’єднання є префікс "https" у URL-адресі.

OpenVPN – це VPN-протокол з відкритим вихідним кодом, який використовує SSL/TLS для захищеного зв’язку. Він підходить як для з’єднань «точка-точка», так і для «місце-місце» і є одним із найбільш безпечних і гнучких VPN-рішень завдяки підтримці потужного шифрування та автентифікації.

Secure Shell (SSH) використовується для створення зашифрованих VPN-тунелів, через які передаються дані. SSH-з’єднання ініціюються клієнтом SSH, а інформація передається через захищений тунель між локальним портом і

віддаленим сервером. Це забезпечує високий рівень безпеки при роботі з віддаленими серверами.

Існує два основних типів VPN:

Віддалений доступ

Цей тип VPN дозволяє користувачам підключатися до приватної мережі з будь-якого місця через Інтернет, забезпечуючи безпечний і зашифрований зв’язок. Він корисний як для бізнесу, так і для особистого користування.

Для корпоративних користувачів: співробітники можуть віддалено підключатися до корпоративної мережі, отримуючи доступ до внутрішніх ресурсів, файлів та систем, що забезпечує ефективну роботу поза офісом.

Для особистого користування: приватні користувачі застосовують VPN для обходу регіональних обмежень, доступу до заблокованих веб-сайтів і підвищення рівня конфіденційності та безпеки в Інтернеті.

Місце-місце

Цей тип VPN зазвичай використовується великими організаціями для з’єднання мереж кількох офісів у різних локаціях. Він поділяється на інтранет та екстранет, з яких перший використовується для об’єднання декількох офісів однієї компанії в єдину безпечну корпоративну мережу, а другий застосовується для встановлення безпечного з’єднання між мережами різних компаній, що дозволяє безпечно обмінюватися даними та співпрацювати.

Обидва типи VPN забезпечують безпечний зв’язок між розподіленими мережами, дозволяючи компаніям ефективно керувати своїми ресурсами та забезпечувати безпечний обмін даними [3].

Технологія VPN має свої переваги та недоліки, які зазначенні у таблиці 1.1.

Таблиця 1.1 – Переваги та недоліки VPN

Переваги	Недоліки
Надійний захист інтернет-трафіку завдяки шифруванню	Можливе зниження швидкості інтернет-з'єднання

Продовження таблиці 1.1

Переваги	Недоліки
Забезпечення конфіденційності та анонімності в мережі	Безкоштовні або дешеві VPN можуть бути ненадійними та збирати дані користувачів
Приховування реальної IP-адреси	Висока вартість преміальних VPN-сервісів
Надає захист активістам у ворожому середовищі	VPN не захищає від збору персональних даних у соціальних мережах
Запобігає обмеженню пропускної здатності на основі певних дій	Деякі пристрої не можуть підтримувати VPN-з’єднання
Захист від DDoS-атак	Використання VPN заборонене або обмежене в деяких країнах
Створення безпечного середовища для віддаленої роботи	Не забезпечує захист від добровільного розголошення особистої інформації

Таким чином ми розуміємо що використання надійного VPN є гарним рішенням для бізнесу або особистого використання [4].

VPN і Tor вирішують різні завдання в забезпеченні безпеки та конфіденційності. VPN є універсальним рішенням для повсякденного використання, пропонуючи високу швидкість, захист у публічних мережах і обхід географічних обмежень. Однак його анонімність обмежена залежністю від провайдера. Tor забезпечує неперевершену анонімність завдяки децентралізації та цибулевій маршрутизації, але поступається в швидкості та зручності, що робить його ідеальним для специфічних завдань, таких як обхід цензури чи анонімний доступ до прихованих сервісів.

Вибір між VPN і Tor залежить від пріоритетів користувача: для швидкості, стрімінгу та корпоративного використання краще обрати VPN, тоді як для максимальної анонімності та роботи в умовах цензури – Tor. Комбіноване

використання цих технологій може забезпечити найвищий рівень захисту, але вимагає ретельного вибору провайдера та правильного налаштування. У сучасному цифровому світі, де загрози конфіденційності зростають, обидві технології залишаються незамінними для безпечної роботи в Інтернеті.

1.2 Технологія TOR: принципи роботи, переваги та недоліки

Tor – це браузер, створений для забезпечення анонімності в мережі Інтернет, але перш за все це розподілена мережа, що складається з децентралізованої спільноти серверів, які добровільно розгортаються по всьому світу. Ці вузли є ключовими точками з’єднання в системі. Вузол може бути маршрутизатором, комп’ютером, мобільним пристроєм або іншим обладнанням, що забезпечує передачу даних через мережу Tor.

Вузли виконують роль проміжних станцій між користувачем та його кінцевим пунктом призначення. Вони спрямовують інтернет-трафік через шифровані маршрути, приховуючи реальну IP-адресу користувача та ускладнюючи спроби стеження. Мережа Tor використовує каталог вузлів для визначення оптимального маршруту передачі даних, який зазвичай проходить через щонайменше три окремі вузли:

1. Вузол входу (вузол охорони) – перша точка доступу користувача до мережі Tor.

2. Середній вузол – передає зашифровані пакети між іншими вузлами, додаючи додатковий рівень анонімності.

3. Вузол виходу – останній етап маршруту, після якого дані потрапляють у відкритий Інтернет.

Tor щоразу випадково вибирає новий маршрут для передавання даних, змінюючи набір вузлів при кожному відвідуванні нового сайту. Така рандомізація ускладнює відстеження трафіку, оскільки можливі шляхи передачі постійно змінюються. Чим більше вузлів у мережі, тим складніше будь-якому спостерігачеві

простежити повний шлях даних, що суттєво підвищує рівень конфіденційності та безпеки користувачів.

Tor використовує процес шифрування цибулевої маршрутизації, щоб запобігти веб-сайтам та іншим службам дізнатися про місцезнаходження користувача Tor (через його IP-адресу) або перехопити вміст надісланого повідомлення. Дані шифруються при вході в клієнт Tor для кожного вузла. Таким чином, налаштування з кількома вузлами створює більший рівень безпеки, ніж проксі-сервер, що використовує один вузол. На відміну від інших постачальників серверів, які спрямовують трафік через один окремий вузол, Tor, надсилаючи інформацію через кілька вузлів, ефективно змушує тих, хто відстежує інформацію, втратити з поля зору її походження. Tor не тільки неправильно спрямовує сайти, прагнучи зібрати інформацію від користувачів, які їх відвідують; він йде ще далі, захищаючи не лише шлях запитів користувача від вузла до вузла, але й корисні дані, які містять ці запити, і місцезнаходження користувача [5].

Tor також може забезпечити анонімність серверів завдяки прихованим сервісам, які є Tor-клієнтами або вузлами із спеціально налаштованим серверним програмним забезпеченням. Замість розкриття IP-адреси сервера (а отже, і його реального місцезнаходження), такі сервіси використовують доменну зону .onion, яка є унікальною для мережі Tor. Tor автоматично розпізнає ці псевдо-домени верхнього рівня (TLD) та анонімно маршрутизує трафік між користувачами та прихованими сервісами. Щоб отримати доступ до таких ресурсів, необхідно використовувати Tor-клієнт, оскільки звичайні браузери та інтернет-протоколи не підтримують з'єднання з доменами .onion [6].

Однією зі слабких сторін, якщо не найбільшою слабкістю, браузера Tor є користувач. Оскільки браузер попередньо налаштовано з урахуванням безпеки, налаштування не рекомендується. Насправді, найкраще, що може зробити користувач Tor, – це не змінювати жодних налаштувань браузера, оскільки будь-яке налаштування може вилити інформацію з Tor. Атака «людина посередині»(man-in-the-middle) – це спосіб обійти безпеку користувачів Tor шляхом включення служби захоплення між користувачем Tor і пунктом призначення. Національні держави

мають ресурси для таких типів атак на Tor, але навіть у цьому випадку скомпрометувати Tor дуже важко. Кожен із цих методів вимагає більше ресурсів і часу, ніж коли-небудь буде надано звичайним злочинцям, якщо не існує особливих ситуацій, таких як зв’язок із тероризмом. Навіть тоді кількість агентств, які мають доступ до таких ресурсів, дуже мала. Також недоліком є те, що оскільки трафік користувача проходить через кілька вузлів і багаторазово шифрується, швидкість з'єднання може бути нижчою, ніж зазвичай [7].

Технологія Tor є потужним інструментом для забезпечення анонімності та обхіду цензури, але вона має свої обмеження, зокрема низьку швидкість і залежність від безпеки вихідних вузлів. Завдяки децентралізованій архітектурі та цибулевій маршрутизації Tor залишається одним із найнадійніших рішень для захисту конфіденційності в Інтернеті. Однак для максимальної безпеки користувачі повинні дотримуватися рекомендованих практик і розуміти потенційні ризики, пов’язані з неправильним використанням або атаками на мережу.

1.3 Порівняльний аналіз VPN та TOR

VPN чудово підходить для повсякденної роботи, наприклад, для роботи в публічних мережах Wi-Fi, обходу географічних обмежень та/або доступу до платформ для стрімінгу. Це щось середнє між швидкістю та безпекою. Tor дійсно ідеальний для максимальної анонімності, хоча в таких випадках працюють журналісти, активісти та користувачі для роботи з конфіденційною інформацією. Але через свою низьку швидкість, Tor, безумовно, не підходить для потокового передавання відео або завантаження великих файлів.

Рисунок 1.1. – Спільне VPN та TOR

Таблиця 1.2 – Порівняльний аналіз VPN та TOR

Особливість	VPN	TOR
Анонімність	Досить висока, але залежить від провайдера	Висока, завдяки маршрутизації
Швидкість	Висока (залежить від сервера)	Низька (через багатошарове шифрування)
Шифрування	Так, але залежить від протоколу	Так, багатошарове шифрування
Вартість	Часто платна	Безкоштовна
Обхід блокувань	Ефективний	Обмежений (деякі сайти блокують TOR-користувачів)
Доступність	Широка (багато платних та безкоштовних сервісів)	Відкрита, але потрібно встановлювати спеціальний браузер

VPN чудово підходить для повсякденної роботи, наприклад, для роботи в публічних мережах Wi-Fi, обходу географічних обмежень та/або доступу до платформ для стрімінгу. Це щось середнє між швидкістю та безпекою. Tor дійсно ідеальний для максимальної анонімності, хоча в таких випадках працюють журналісти, активісти та користувачі для роботи з конфіденційною інформацією. Але через свою низьку швидкість, Tor, безумовно, не підходить для потокового передавання відео або завантаження великих файлів.

Очевидно, що проксі-VPN і TOR мають свої плюси і мінуси. Якщо пріоритетом для людини є вибір між швидкістю та безпекою, то VPN має перевагу над проксі-серверами. Для тих, хто любить свою анонімність незалежно від швидкості, тоді TOR буде правильним вибором. У сучасному світі, можливо, ніхто не має простого рішення для кожної роботи. Тому пропонується використовувати кілька стратегій: можна, наприклад, інтегрувати використання VPN з мережею TOR, щоб мати оптимальний захист конфіденційності. Такий підхід забезпечить як найвищий рівень анонімності, так і достатній захист персональних даних в мережі одночасно.

Таким чином, технології VPN і TOR мають важливе значення для сучасного користувача, який хоче забезпечити свою конфіденційність і безпечно працювати в Інтернеті [8].

VPN і Tor є потужними інструментами для захисту конфіденційності, але вони вирішують різні завдання. VPN забезпечує швидке та зручне рішення для повсякденного використання, такого як стрімінг, обхід географічних обмежень і захист у публічних мережах. Tor, завдяки своїй децентралізованій архітектурі та цибулевій маршрутизації, пропонує неперевершену анонімність, але за рахунок швидкості та зручності.

Вибір між VPN і Tor залежить від потреб користувача: для швидкості та універсальності – VPN, для максимальної анонімності – Tor. У сучасному світі, де загрози конфіденційності зростають, комбіноване використання цих технологій може забезпечити оптимальний баланс між безпекою, анонімністю та продуктивністю. Користувачам слід ретельно обирати VPN-провайдера, уникати витоків інформації в Tor та регулярно оновлювати програмне забезпечення для захисту від вразливостей.

2 ПРОЕКТУВАННЯ СИСТЕМИ ЗАХИСТУ ТА АНОНІМІЗАЦІЇ ІНТЕРНЕТ-ТРАФІКУ

2.1 Аналіз загроз для інтернет-трафіку

Основою безпеки в Інтернеті є занепокоєння щодо заходів захисту даних під час передачі через сукупність взаємопов’язаних мереж. Ми припускаємо, що потік даних від джерела до місця призначення відбувається через канал зв’язку. Система безпеки полягає в тому, щоб обмежити доступ лише тим сторонам, які мають право доступу відповідно до політики безпеки.

Існують декілька загроз для інтернет-трафіку:

Перехоплення інтернет трафіку (Sniffing)

Перехоплення пакетів – це процес отримання доступу до даних, які передаються через комп’ютерну мережу, з метою їхнього аналізу або подальшого використання. За аналогією, у телефонній мережі це схоже на прослуховування дротової лінії. Для цього використовуються спеціальні програми або пристрої, відомі як сніфери пакетів. Ці інструменти можуть бути підключені до мережі для моніторингу та «прослуховування» всього мережевого трафіку, що проходить через неї. У сфері адміністрування мереж сніфери застосовуються для виявлення помилок у пакетах даних, аналізу продуктивності мережі, виявлення вузьких місць і забезпечення ефективної передачі інформації. Проте, незважаючи на легальні застосування, ці ж інструменти широко використовуються хакерами та кіберзлочинцями. З їх допомогою зловмисники можуть незаконно збирати інформацію про мережу або користувачів, яку потім використовують для проникнення в систему. За допомогою сніферів можна отримати такі дані, як логіни й паролі, IP-адреси, імена користувачів, типи використовуваних протоколів, а також інші технічні деталі, які можуть бути корисними для здійснення атак. Сніфінг пакетів широко застосовується у сфері аналізу мережевих вторгнень, у процесах керування трафіком, а також при несанкціонованому зборі інформації. Програмне забезпечення для аналізу паролів є одним із найпоширеніших інструментів сніфінгу.

Однією з основних проблем, пов’язаних із перехопленням пакетів, є те, що цей процес надзвичайно важко виявити. Хоча виявлення можливе за певних умов, через складність і технічну специфіку таке виявлення на практиці проводиться вкрай рідко. Популярність аналізу пакетів серед зловмисників пояснюється тим, що цей метод дозволяє бачити майже все, що відбувається в мережі, роблячи його потужним засобом як для забезпечення безпеки, так і для порушення конфіденційності [9].

Атака типу «людина посередині» (Man-In-The-Middle), яка також відома під скороченнями MIM, MiM, MitM або MITMA, є видом криптографічної атаки, що передбачає втручання зловмисника в канал зв’язку між двома сторонами, які обмінюються інформацією. У ході такої кібератаки третя сторона, яка діє зловмисно, непомітно вбудовується в комунікаційний процес і отримує можливість повністю контролювати передавання даних. Це означає, що зловмисник може читати, перехоплювати, змінювати, підмінювати або навіть видозмінювати трафік, що передається між жертвами, залишаючись при цьому невидимим для обох сторін. Такий тип атаки особливо небезпечний тим, що, якщо зловмисник не буде автентифікований або ідентифікований у системі, він не залишає явних слідів свого втручання. Таким чином, жертви не підозрюють, що їхня комунікація була скомпрометована. Це робить атаку MitM особливо складною для виявлення та захисту від неї, оскільки на перший погляд передавання даних відбувається так, ніби нічого підозрілого не відбувається [10].

DNS Spoofing (підміна DNS) – це різновид кібератаки, націлений на систему доменних імен (DNS), яка є критично важливою складовою інфраструктури Інтернету. DNS перетворює зрозумілі для користувачів доменні імена, такі як www.example.com, у числові IP-адреси, які розуміють комп’ютери та мережеві пристрої. Атака DNS Spoofing полягає в тому, що зловмисник втручається в процес вирішення DNS-запиту, змінюючи відповідь на нього таким чином, щоб замість правильної IP-адреси користувачу поверталася підроблена.

У результаті комп’ютери або пристрої жертви починають асоціювати правильне доменне ім’я з фальшивою IP-адресою, що веде на шкідливий,

фішинговий або небажаний сайт. Таким чином, користувачі можуть бути перенаправлені на сайти, які виглядають як справжні, але призначені для крадіжки особистих даних, паролів, фінансової інформації або для встановлення шкідливого програмного забезпечення. DNS Spoofing є серйозною загрозою, оскільки може бути використаний для широкомасштабного обману користувачів без їх відома, а вразливості в DNS-системі можуть зробити такі атаки ефективними та важко виявлюваними [11].

Для наочної ілюстрації впливу цих загроз і способів їхнього усунення за допомогою VPN і Tor нижче наведено спрощену схему, яка показує, як незахищений трафік стає вразливим до атак і як VPN і Tor захищають дані користувача.

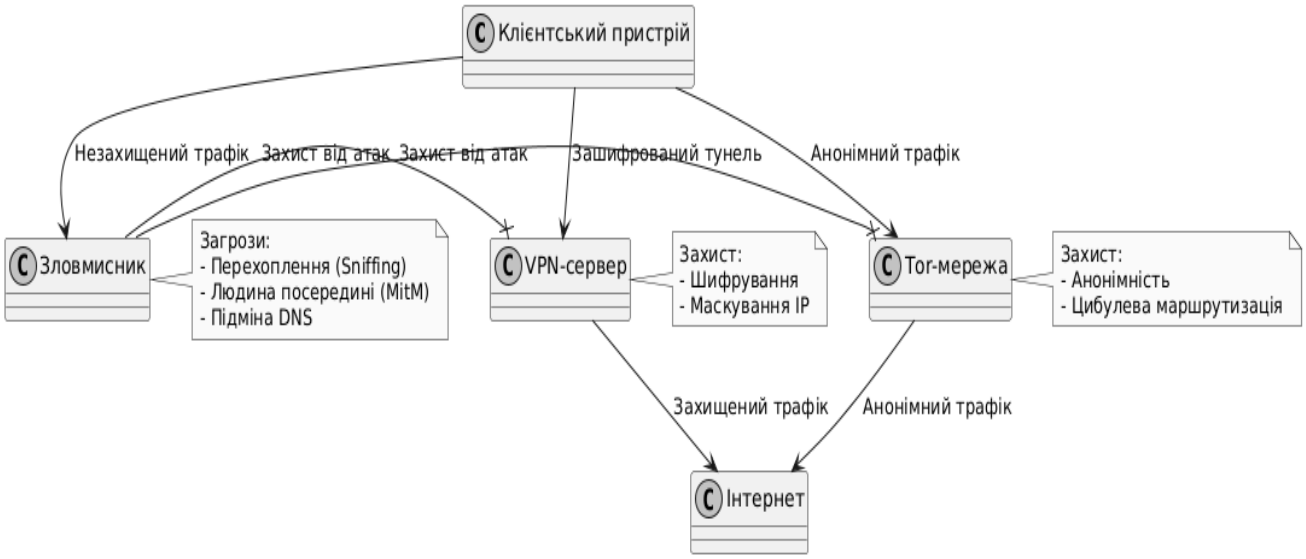


Рисунок 3.1. – Спрощена схема загроз для інтернет-трафіку та захисту за допомогою VPN і Tor

Загрози для інтернет-трафіку, такі як перехоплення (Sniffing), атаки «людина посередині» (MitM) і підміна DNS (DNS Spoofing), створюють серйозні ризики для конфіденційності та безпеки даних. VPN і Tor є ефективними інструментами для захисту від цих загроз, але їхня ефективність залежить від сценарію використання. VPN забезпечує швидке та зручне рішення для захисту даних у публічних мережах, обходу географічних обмежень і протидії сніфінгу та DNS Spoofing, але його

анонімність обмежена. Tor пропонує неперевершену анонімність завдяки цибулевій маршрутизації, що робить його ідеальним для захисту від усіх трьох загроз у ситуаціях, де конфіденційність є пріоритетом, але поступається в швидкості.

Для максимального захисту рекомендується комбінувати VPN і Tor, використовуючи надійних провайдерів і правильні налаштування. У сучасному цифровому світі, де загрози для інтернет-трафіку постійно зростають, свідоме використання цих технологій дозволяє користувачам ефективно захищати свої дані та зберігати конфіденційність.

2.2 Структурно-функціональна схема системи захисту

У цій роботі буде розглядатися VPN-система та Python клієнт до неї. Система базується на сучасному VPN-протоколі WireGuard, який відзначається високою продуктивністю, простотою налаштування та надійністю шифрування.

Компоненти системи

Клієнтський пристрій: підключається до VPN-серверу через WireGuard. Весь інтернет-трафік шифрується та перенаправляється на сервер.

VPN-сервер (VPS): запускає WireGuard, приймає вхідні підключення клієнтів. Встановлює маршрутизацію та NAT для виходу в Інтернет.

iptables + NAT: маскує вихідний трафік клієнта, дозволяє безперешкодно передавати його до глобальної мережі.

IPforwarding: вмикає пересилання трафіку між інтерфейсами wg0 (VPN) та eth0 (інтернет).

Інтернет (WAN): кінцева точка, до якої надсилається зашифрований клієнтський трафік.

Схема зображена на рисунку 2.2.

Структурно-функціональна схема системи захисту на базі VPN із протоколом WireGuard і Python-клієнтом демонструє ефективне рішення для забезпечення безпеки та конфіденційності в Інтернеті. Завдяки високій продуктивності, надійному шифруванню та простоті налаштування система ідеально підходить для

захисту даних у публічних мережах, обходу географічних обмежень і безпечного доступу до корпоративних ресурсів. Однак користувачі повинні враховувати залежність від надійності VPS, можливі вразливості конфігурації та юридичні обмеження в деяких країнах.

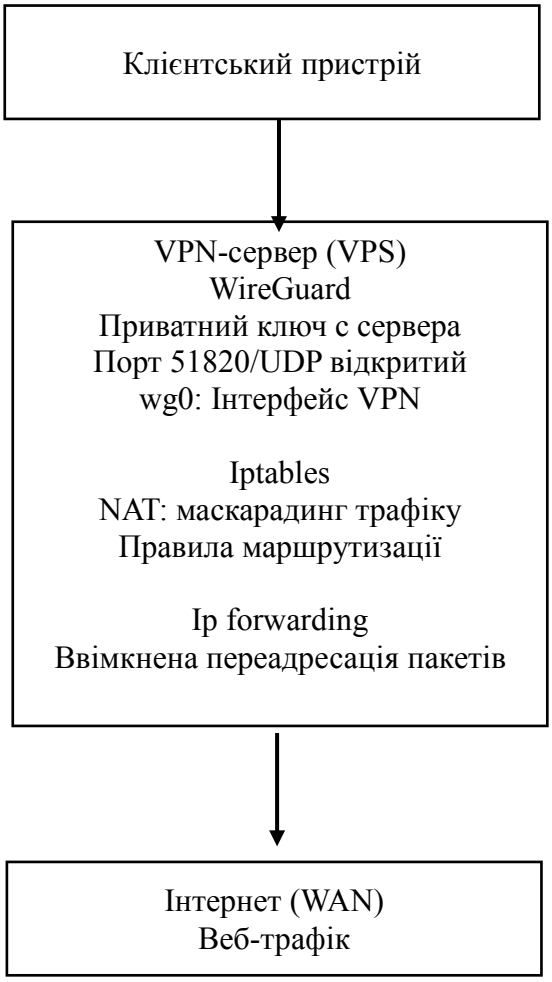


Рисунок 2.2. Структурно-функціональна схема системи захисту

Для максимального захисту рекомендується використовувати надійного провайдера з політикою no-logs, регулярно оновлювати програмне забезпечення та правильно налаштовувати iptables і IP Forwarding. У порівнянні з Tor, система на базі WireGuard пропонує кращу швидкість і зручність, але поступається в анонімності. Для специфічних завдань, що вимагають високої анонімності, можна комбінувати VPN із Tor, спрямовуючи трафік через VPN-сервер до мережі Tor.

Таким чином, система є гнучким і потужним інструментом для сучасних потреб безпеки в цифровому середовищі.

2.3 Методи підвищення рівня анонімності та безпеки

Одним із базових інструментів для підвищення конфіденційності є використання віртуальних приватних мереж. VPN забезпечує шифрування трафіку між користувачем та віддаленим сервером, що дозволяє приховати справжню IP-адресу та зробити трафік недоступним для аналізу з боку провайдера або сторонніх осіб. Особливо важливо, аби VPN-провайдер не вів логів та підтримував сучасні протоколи шифрування, зокрема WireGuard, OpenVPN чи IPSec [12]. Більш просунуті користувачі можуть самостійно розгорнути VPN-сервер на хмарній інфраструктурі або власному обладнанні, що суттєво підвищує контроль над усіма компонентами з’єднання.

Ще одним дієвим методом є використання мережі Tor яка забезпечує багаторівневе шифрування шляхом переспрямування трафіку через декілька випадково обраних вузлів по всьому світу. Завдяки цьому навіть при спробах стеження неможливо встановити маршрут з’єднання між відправником і кінцевим ресурсом. Хоча Tor не гарантує абсолютну безпеку, він є одним із найефективніших інструментів для анонімного доступу до Інтернету [13]. Варто враховувати, що внаслідок багаторівневої маршрутизації швидкість з’єднання у мережі Tor суттєво нижча, що робить її непридатною для роботи з потоковим відео або великими обсягами даних.

На рівні операційної системи також можна вжити заходів для забезпечення цифрової анонімності. Існують спеціалізовані дистрибутиви, такі як Tails OS, Whonix чи Qubes OS, які розроблені з урахуванням потреб користувачів, що прагнуть повної конфіденційності. Наприклад, Tails запускається з флеш-накопичувача, не залишаючи слідів на комп’ютері, а весь інтернет-трафік автоматично проходить через Tor [14]. Whonix забезпечує повну ізоляцію середовищ, де одне віртуальне середовище відповідає за маршрутизацію через Tor,

а друге – за роботу користувача. Qubes OS, у свою чергу, застосовує концепцію мікровіртуалізації, завдяки якій кожен застосунок запускається у власному ізольованому середовищі, мінімізуючи ризики витоку інформації [15].

Особливу увагу слід приділити інструментам цифрової гігієни під час роботи в мережі. Анонімність у браузері можливо підвищити шляхом використання браузерів із фокусом на приватність, таких як Brave або Mozilla Firefox із відповідними розширеннями (NoScript, uBlock Origin, Privacy Badger). Також важливо регулярно очищувати cookies, кеш, локальне сховище, вимикати WebRTC, блокувати Canvas fingerprinting та інші сучасні методи трекінгу [16].

Комунікація в мережі також має здійснюватися із застосуванням шифрованих каналів. Для приватного спілкування рекомендується використовувати месенджери, що реалізують наскрізне шифрування, зокрема Signal, Session або Threema. На відміну від популярних платформ на зразок Telegram чи WhatsApp, зазначені рішення не зберігають метаданих, не мають прив’язки до номера телефону (у випадку Session), і не залежать від централізованих серверів [17].

Нарешті, слід пам’ятати про поведінкову гігієну. Анонімність втрачається у момент, коли користувач, навіть перебуваючи у Tor-мережі чи за VPN, авторизується у своїх акаунтах соціальних мереж або залишає унікальні сліди в мережі. Розділення особистої та анонімної цифрової ідентичності, уникнення реєстрації з використанням особистих даних, а також свідоме обмеження цифрової активності, що дозволяє ідентифікувати користувача, є критично важливими елементами цифрової безпеки [18].

Для наочної ілюстрації методів підвищення анонімності та безпеки нижче наведено спрощену вертикальну схему, яка показує, як клієнтський пристрій використовує різні інструменти для захисту даних і забезпечення конфіденційності в мережі Інтернет.

Рисунок 2.3. – Спрощена вертикальна схема методів підвищення анонімності та безпеки

Таким чином, досягнення високого рівня анонімності та безпеки в Інтернеті вимагає системного підходу, який передбачає не лише використання відповідного програмного забезпечення, але й зміни цифрових звичок та свідоме управління особистою інформацією в мережі.

Взаємодія мережі Tor та VPN: комбінація для підвищення конфіденційності

Існує два основних способи використання VPN у поєднанні з Tor: Tor через VPN (Tor over VPN) та VPN через Tor (VPN over Tor). Обидва варіанти мають різні цілі та застосовуються залежно від потреб користувача.

Tor через VPN (Tor over VPN) є найпоширенішим сценарієм. У цьому випадку спочатку встановлюється VPN-з’єднання, а вже потім через нього запускається Tor-браузер. Такий підхід дозволяє приховати сам факт використання Tor від інтернет-провайдера, оскільки весь трафік шифрується і спрямовується на VPN-сервер. З погляду провайдера, це виглядає як звичайне VPN-з’єднання. Водночас вихід з Tor-мережі залишається анонімним, і реальна IP-адреса користувача не розкривається навіть перед першими вузлами Tor. Це значно знижує ризики deanonymization-атак та блокує можливість спостереження за мережею Tor з боку провайдера або на рівні державної цензури [12].

Проте цей варіант має і свої недоліки. По-перше, користувач змушений довіряти VPN-провайдеру, оскільки саме він бачить справжню IP-адресу користувача та обсяг його зашифрованого трафіку. У випадку компрометації VPN-сервера або вимоги передачі логів, ця інформація може бути використана проти користувача. Тому вкрай важливо використовувати VPN-сервіси з перевіреною репутацією, що не зберігають логи та розміщені в юрисдикціях із сильними законами про конфіденційність.

VPN через Tor (VPN over Tor) – менш поширена, але цікава схема, яка реалізується складніше і підтримується не всіма VPN-сервісами. У цьому випадку користувач спочатку підключається до мережі Tor, а вже потім встановлює VPN-з’єднання поверх неї. Таким чином, VPN-провайдер не бачить справжню IP-адресу користувача, лише адресу вихідного вузла Tor. Такий підхід підвищує анонімність навіть перед самим VPN-сервером, що може бути корисно в умовах, коли користувач не довіряє жодному централізованому провайдеру [13].

Однак конфігурація VPN over Tor є технічно складнішою та менш гнучкою. До того ж не всі VPN підтримують з’єднання через Tor, і навіть якщо таке можливо, швидкість з’єднання значно знижується через подвійне шифрування та маршрутизацію.

Загалом, поєднання Tor та VPN дозволяє створити багаторівневу систему захисту, де один інструмент компенсує слабкі сторони іншого. Наприклад, Tor забезпечує анонімність за рахунок багаторівневого шифрування і розподіленої

мережі, але не шифрує трафік після виходу з мережі (на вихідному вузлі). VPN, у свою чергу, шифрує трафік на всьому шляху до сервера, але є централізованим рішенням, що потребує довіри до провайдера. Разом вони дозволяють користувачу підвищити як анонімність, так і цілісність переданої інформації, особливо при правильному налаштуванні.

У контексті інформаційної безпеки та цифрових прав, комбінування Tor і VPN вважається одним із найефективніших підходів до мінімізації ризиків стеження, блокування доступу, а також протидії масовому нагляду.

Підвищення рівня анонімності та безпеки в Інтернеті вимагає системного підходу, що поєднує технологічні інструменти (VPN, Tor, спеціалізовані ОС, шифровані месенджери) та поведінкову гігієну. VPN забезпечує швидке та зручне рішення для захисту даних і обходу обмежень, але його анонімність обмежена залежністю від провайдера. Tor пропонує найвищий рівень анонімності завдяки децентралізації, але поступається в швидкості та зручності. Спеціалізовані операційні системи, такі як Tails, Whonix і Qubes OS, ідеально підходять для користувачів із високими вимогами до конфіденційності, але потребують технічних знань. Практики цифрової гігієни, включаючи використання захищених браузерів і месенджерів, а також розділення цифрових ідентичностей, є критично важливими для уникнення деанонімізації.

Комбінація VPN і Tor, зокрема в конфігурації Tor через VPN, є оптимальним рішенням для максимальної анонімності та безпеки, особливо в умовах цензури чи стеження. Однак користувачі повинні обирати надійних провайдерів, уникати витоків інформації та дотримуватися суворих правил цифрової безпеки. У сучасному цифровому середовищі, де загрози конфіденційності постійно зростають, правильне поєднання цих методів дозволяє ефективно захищати дані та зберігати анонімність у мережі.

Схема взаємодії VPN та TOR знаходиться у ДОДАТКУ А.

3 РЕАЛІЗАЦІЯ ТА ТЕСТУВАННЯ СИСТЕМИ ЗАХИСТУ ТРАФІКУ

3.1 Обґрунтування вибору інструментів та технологій

Для реалізації VPN-системи був використаний протокол WireGuard. WireGuard сьогодні вважається однією з найефективніших альтернатив традиційним VPN-протоколам, таким як IKEv2/IPSec та OpenVPN. На відміну від них, він був спроєктований із нуля – з урахуванням сучасних технологій та потреб користувачів. Його кодова база складається лише з приблизно 4000 рядків, що робить WireGuard надзвичайно легким для аудиту безпеки. Для порівняння, OpenVPN містить понад 100 000 рядків коду. Така лаконічність не лише спрощує використання, а й мінімізує потенційні вразливості, підвищує швидкість роботи та загальну продуктивність. Завдяки цьому WireGuard наразі вважається найшвидшим VPN-протоколом серед усіх доступних.

Для наочної ілюстрації роботи системи нижче наведено структурно-функціональну схему, яка відображає взаємодію між клієнтським пристроєм, VPS-сервером і мережею Інтернет через VPN-тунель.

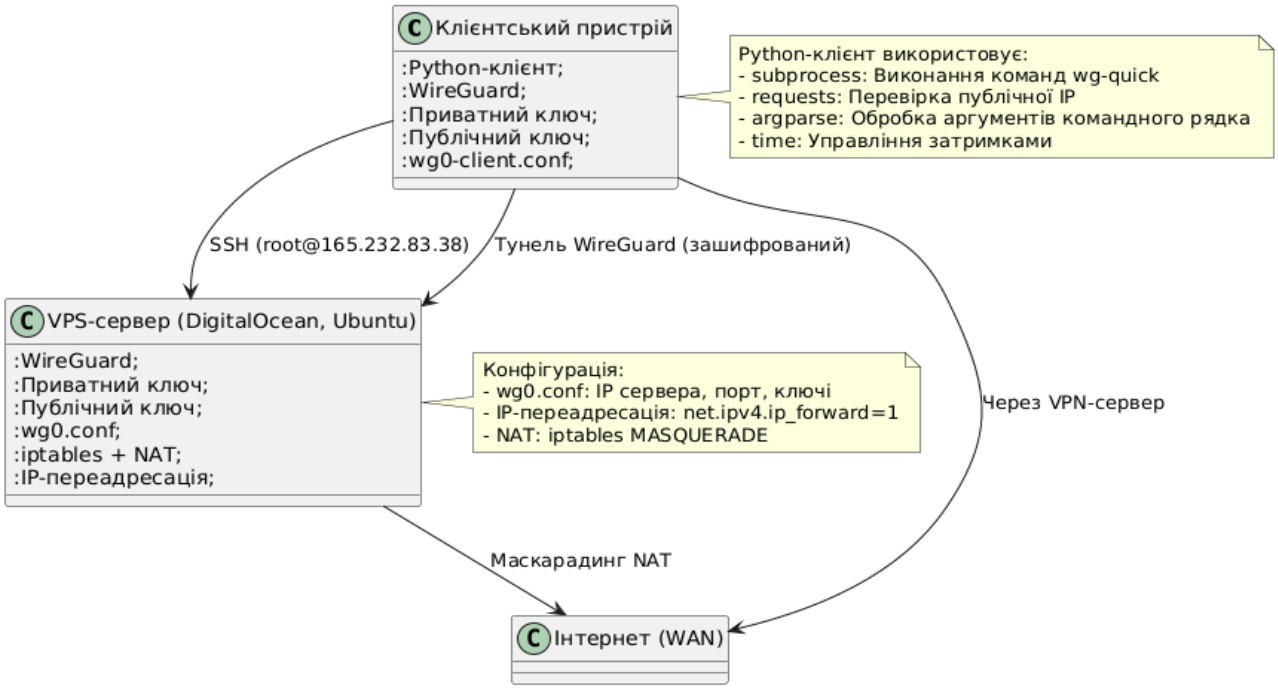


Рисунок 3.1 – Структурно-функціональна схема VPN-системи на базі WireGuard

Ця діаграма чітко ілюструє архітектуру VPN-системи, підкреслюючи простоту та ефективність WireGuard, а також роль Python-клієнта в автоматизації управління з'єднанням.

Як працює WireGuard VPN:

Генерація ключів. При першому запуску WireGuard створює пару криптографічних ключів: приватний залишається на вашому пристрої, а публічний передається VPN-серверу.

Встановлення з'єднання. Для початку з'єднання ваш пристрій надсилає VPN-серверу свій публічний ключ. У відповідь сервер надсилає свій публічний ключ – так створюється захищений автентифікований тунель.

Шифрування та автентифікація. Усі дані, що передаються через тунель, шифруються та автентифікуються, що забезпечує їхню конфіденційність і цілісність.

Оптимальна маршрутизація. WireGuard використовує просту й ефективну схему маршрутизації: кожному пристрою призначається унікальна IP-адреса в межах VPN-мережі, яка використовується для пересилання трафіку.

Рукостискання при з'єднанні. WireGuard не підтримує постійного з'єднання. Замість цього він виконує швидкі «рукостискання» під час кожного підключення, що дозволяє стабільно працювати навіть за умов нестабільного інтернету.

Автоматичне перепідключення. У разі втрати з'єднання WireGuard автоматично й швидко відновлює тунель без втручання користувача.

Просте налаштування. WireGuard створений із фокусом на зручність: достатньо встановити застосунок, згенерувати ключі й налаштувати один конфігураційний файл.

Статичні IP-адреси. Кожному пристрою надається постійна внутрішня IP-адреса, що спрощує адміністрування та налаштування мережі [19].

WireGuard був встановлений на VPS сервері з операційною системою Ubuntu від DigitalOcean. VPS (Virtual Private Server) – це віртуальний сервер, що працює на фізичному обладнанні провайдера, а не розміщується безпосередньо у користувача. Попри це, він надає ті самі функціональні можливості, що й окремий фізичний

сервер: повний контроль над системою, root-доступ, встановлення програмного забезпечення, налаштування мережі тощо [20].

Підключення до VPN-серверу здійснюється за допомогою Python клієнта. Для нього були використані такі бібліотеки, як subprocess, requests, time та argparse. subprocess це бібліотека Python, яка дозволяє виконувати системні команди (наприклад, запускати програми або скрипти). Вона використовується для виклику команд у терміналі. requests – популярна бібліотека для виконання HTTP-запитів. У даному випадку вона використовується для отримання поточної публічної IP-адреси з API. argparse – бібліотека для обробки аргументів командного рядка. time – стандартна бібліотека Python для роботи з часом. Використовується для пауз (затримок) між операціями.

Вибір WireGuard, VPS від DigitalOcean на Ubuntu та Python-клієнта з бібліотеками subprocess, requests, time і argparse для реалізації VPN-системи був обґрунтованим завдяки їхній ефективності, безпеці та зручності. WireGuard забезпечує високу швидкість і надійне шифрування, VPS пропонує гнучкість і контроль, а Python-клієнт спрощує управління з’єднанням. У порівнянні з Tor, система надає кращу продуктивність і зручність, але поступається в анонімності. Для максимального захисту рекомендується комбінувати VPN із Tor і дотримуватися практик цифрової безпеки. Ця система є оптимальним рішенням для захисту трафіку в сучасному цифровому середовищі, поєднуючи простоту, швидкість і надійність.

3.2 Реалізація системи захисту трафіку

Першим кроком для реалізації системи захисту трафіку була оренда VPS-сервера від DigitalOcean на операційній системі Ubuntu, який знаходиться у Нідерландах. Клієнтська частина також реалізувалась на Ubuntu. Підключення до серверу відбувалось за допомогою команди ssh root@165.232.83.38. Перш за все на сервер був встановлений протокол WireGuard та згенеровані публічний та приватний ключі (див. рис.3.3, рис. 3.4).

```
root@vpnserver:~# apt update && apt install wireguard -y
```

Рисунок 3.3 – Встановлення протоколу WireGuard на VPS-сервері

```
root@vpnserver:~# wg genkey | tee server_private.key | wg pubkey > server_public.key
root@vpnserver:~# cat server_private.key
GOmekq4g9kld6pgeXuoKfePbhDAQKm+xGBoDWoncw0M=
root@vpnserver:~# cat server_public.key
LqNIBXYAo/5JvkgGytUOLH0Vj3SFRvOzjeNXqoGhb0M=
```

Рисунок 3.4 – Генерація публічного та приватного ключів

Для використання WireGuard, сервер і кожен клієнт мають згенерувати власну пару ключів. Після цього вони обмінюються публічними ключами, щоб мати змогу встановити захищене з'єднання між собою. Наступним була створена конфігурація для сервера на рис. 3.5, де [Interface] – інтерфейс серверу, у якому була записана внутрішня IP-адреса, порт та приватний ключ сервера. [Peer] – підключений клієнт, де був записаний публічний ключ клієнта та IP-адреси, які сервер буде перенаправляти.

```
  GNU nano 4.8                    /etc/wireguard/wg0.conf
[Interface]
Address = 10.0.0.1/24
ListenPort = 51820
PrivateKey = GOmekq4g9kld6pgeXuoKfePbhDAQKm+xGBoDWoncw0M=

[Peer]
PublicKey = +cEJUxDVIcyBuaog1ZvVvYBnUvWvSDNYKAbtkT146jE=
AllowedIPs = 10.0.0.2/32
```

Рисунок 3.5 – Конфігурація /etc/wireguard/wg0.conf для сервера

Наступним кроком – налаштування клієнта на робочому комп'ютері, на якому буде проходити з'єднання з VPN-сервером. Для цього теж був встановлений

WireGuard, згенеровані публічний та приватний ключі та створена конфігурація. На рис. 3.6 показана конфігурація /etc/wireguard/wg0-client.conf для клієнту, де були записані публічний ключ сервера, з яким встановлюється з'єднання, IP-адреса клієнта у внутрішній VPN-мережі, приватний ключ клієнта IP-адреса та порт сервера, DNS та зазначено що весь трафік іде через VPN.

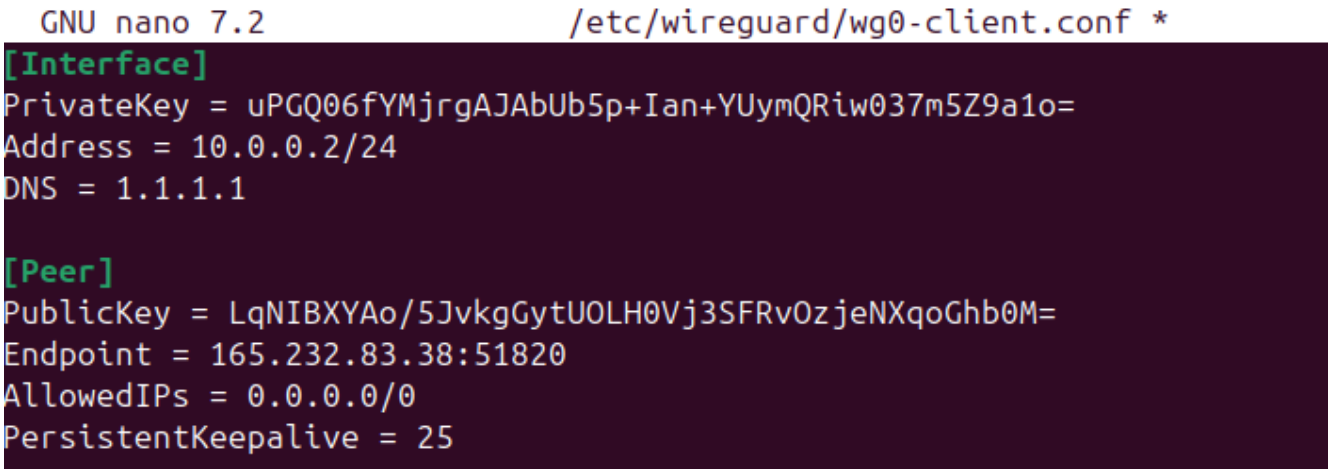


Рисунок 3.6 – Конфігурація /etc/wireguard/wg0-client.conf для клієнта

Далі було створено Python-управління VPN підключенням, лістинг коду якого знаходиться у ДОДАТКУ Б. Таким чином вже можна було підключитися до VPN-серверу через клієнт, спочатку запустивши VPN на сервері за допомогою команди sudo wg-quick up wg0 та на клієнті командою sudo wg-quick up wg0-client, але при такому підключенні не буде доступу в інтернет. Для цього потрібно налаштувати мережеві правила (NAT) на сервері. У конфігурації sysctl.conf було знайдено та розкоментовано рядок #net.ipv4.ip_forward=1, як це показано на рис. 3.7 та 3.8.

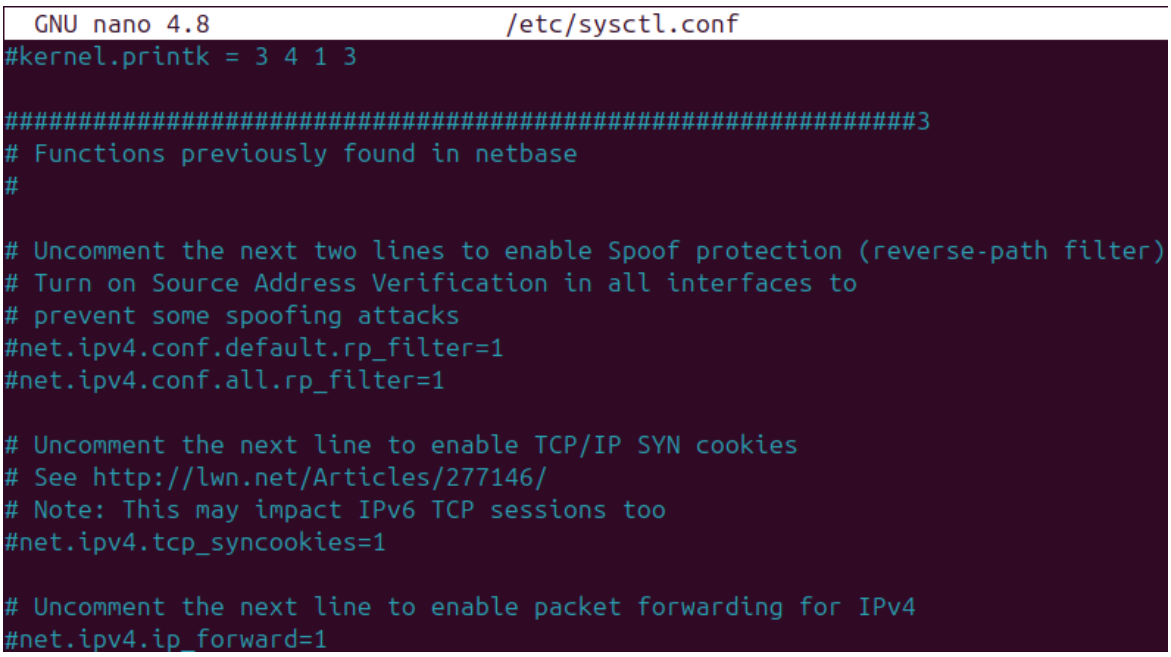


Рисунок 3.7 – Закоментований рядок #net.ipv4.ip_forward=1

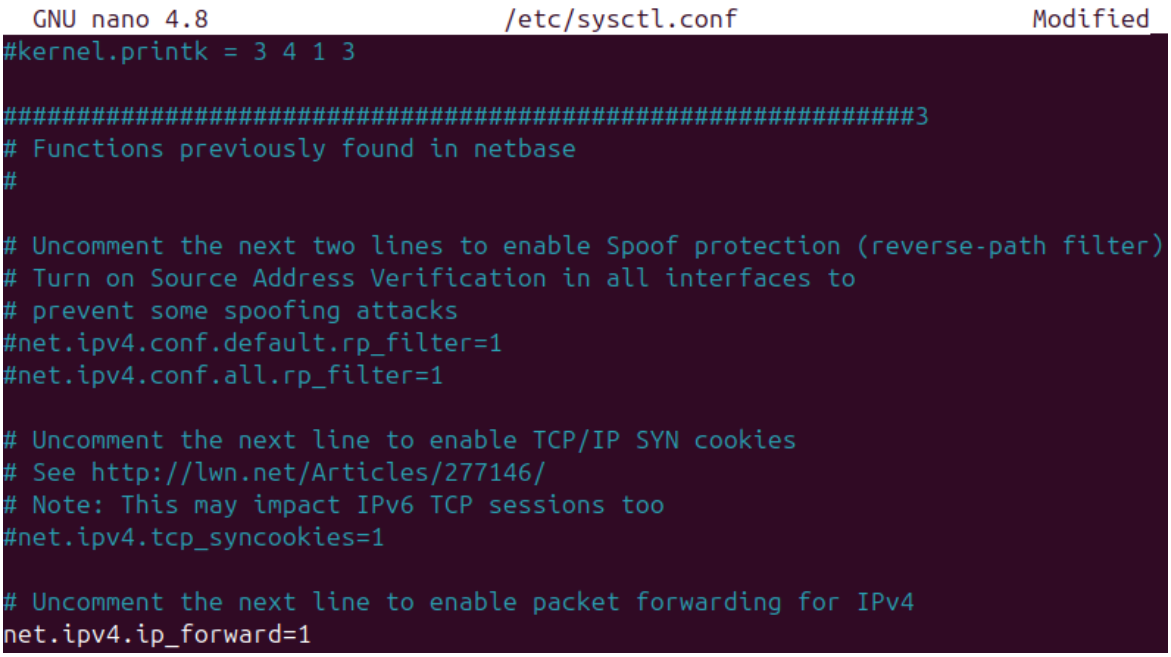


Рисунок 3.8 – Розкоментований рядок net.ipv4.ip_forward=1

Після цього застосувавши команду sudo sysctl -p було дозволено переадресацію IP. Наступним кроком для доступу в інтернет через VPN-сервер було введено наступні команди:

1. SERVER_INTERFACE=$(ip route get 8.8.8.8 | grep -oP 'dev \K\S+')

ip route get 8.8.8.8 – запитує маршрут до публічної IP-адреси.

Система відповідає, через який інтерфейс буде здійснено вихід у зовнішню мережу.

grep -oP 'dev \K\S+' – витягує назву інтерфейсу.

Результат зберігається у змінну SERVER_INTERFACE.

У результаті SERVER_INTERFACE буде містити ім’я мережевого інтерфейсу, через який сервер виходить в Інтернет.

2. sudo iptables -t nat -A POSTROUTING -o $SERVER_INTERFACE -j MASQUERADE

iptables -t nat – працює з таблицею NAT.

-A POSTROUTING – додає правило до ланцюга POSTROUTING, який застосовується до вихідного трафіку.

-o $SERVER_INTERFACE – правило застосовується лише до трафіку, що йде через інтерфейс $SERVER_INTERFACE.

-j MASQUERADE – активує маскарадинг: заміна IP-адреси відправника на зовнішню IP-адресу сервера.

У результаті коли VPN-клієнт надсилає запит до Інтернету, сервер переписує IP-адресу клієнта на свою – так, ніби це він сам зробив запит. Відповідь повертається, а сервер пересилає її назад клієнту.

Останнім кроком було збереження змін правил командами apt install iptables-persistent -y та netfilter-persistent save.

Тепер, запустивши VPN на сервері та на клієнті можна було повноцінно підключитися до VPN-серверу через клієнт за допомогою команди python3 vpn.py connect.

Реалізація системи захисту трафіку на базі WireGuard і Python-клієнта, розгорнутої на VPS-сервері від DigitalOcean, продемонструвала високу ефективність у забезпеченні безпеки та анонімності. Система успішно приховує IP-адресу клієнта, забезпечує швидке та надійне шифрування і є простою у налаштуванні завдяки автоматизації через Python-скрипт і systemctl. У порівнянні з

Tor, VPN пропонує вищу швидкість і зручність, але поступається в анонімності через централізовану природу. Для максимального захисту рекомендується комбінувати VPN із Tor, використовувати надійні сервери та дотримуватися практик цифрової безпеки. Ця система є потужним і гнучким рішенням для захисту трафіку в сучасному цифровому середовищі, але вимагає ретельного налаштування та усвідомленого підходу до використання.

3.3 Тестування та аналіз результатів

Для початку роботи з VPN-сервером потрібно увімкнути VPN на сервері командою sudo wg-quick up wg0 та на клієнті командою sudo wg-quick up wg0-client. Для зручності можна зробити автозапуск VPN на сервері при його старті командою sudo systemctl enable wg-quick@wg0. Після цього вводимо команду python3 vpn.py connect у термінал комп'ютера. На рис. 3.9 зображене підключення до VPN-сервера.

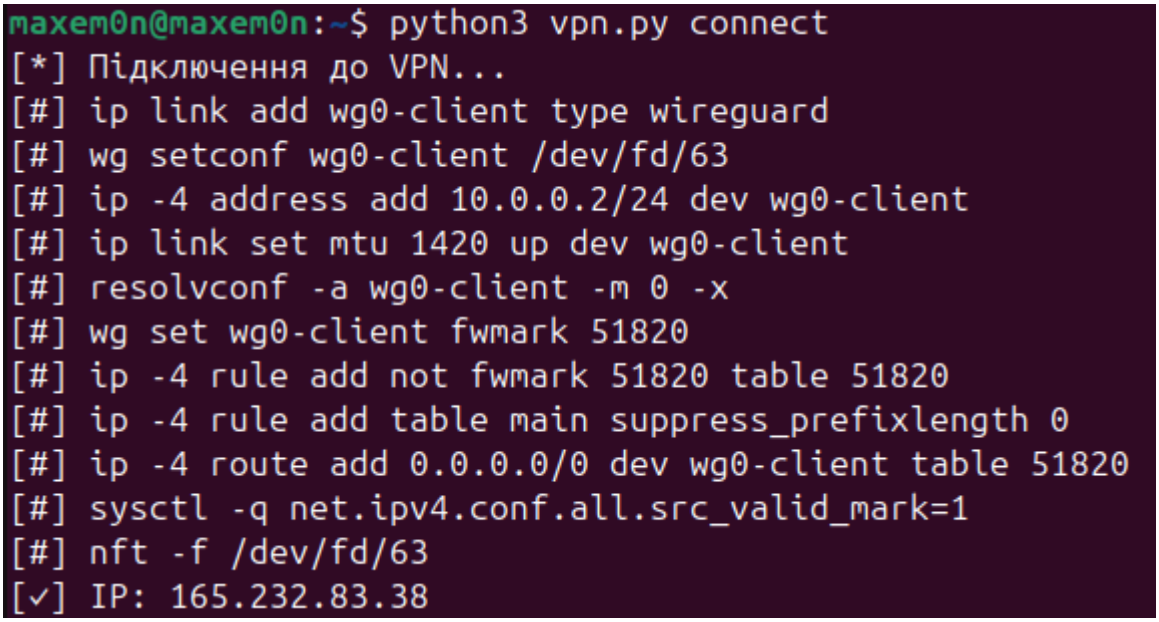


Рисунок 3.9 – Підключення до VPN-серверу

На ньому зображено що IP-адреса змінилась на 165.232.83.38. Також на рисунку 3.10 показана перевірка через сайт https://whatismyipaddress.com/. На ньому вказано що наша локація у Нідерландах.

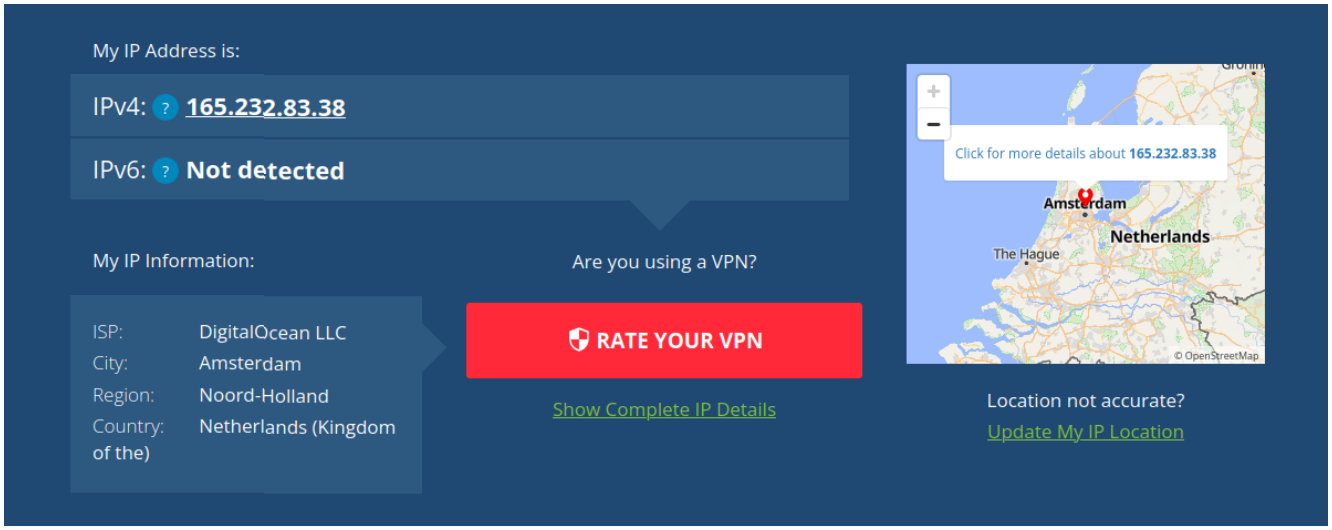


Рисунок 3.10 – Перевірка через сайт https://whatismyipaddress.com/

На рис. 3.11. показане відключення від VPN-серверу командою python3 vpn.py disconnect та реальну IP-адресу.

```
maxem0n@maxem0n:~$ python3 vpn.py disconnect
[*] Відключення від VPN...
[#] ip -4 rule delete table 51820
[#] ip -4 rule delete table main suppress_prefixlength 0
[#] ip link delete dev wg0-client
[#] resolvconf -d wg0-client -f
[#] nft -f /dev/fd/63
[✓] IP: 188.130.177.217
```

Рисуннок 3.11 – Відключення від VPN-серверу

На рисунку 3.12 зображена перевірка IP-адреси з відключеним VPN через сайт https://whatismyipaddress.com/.

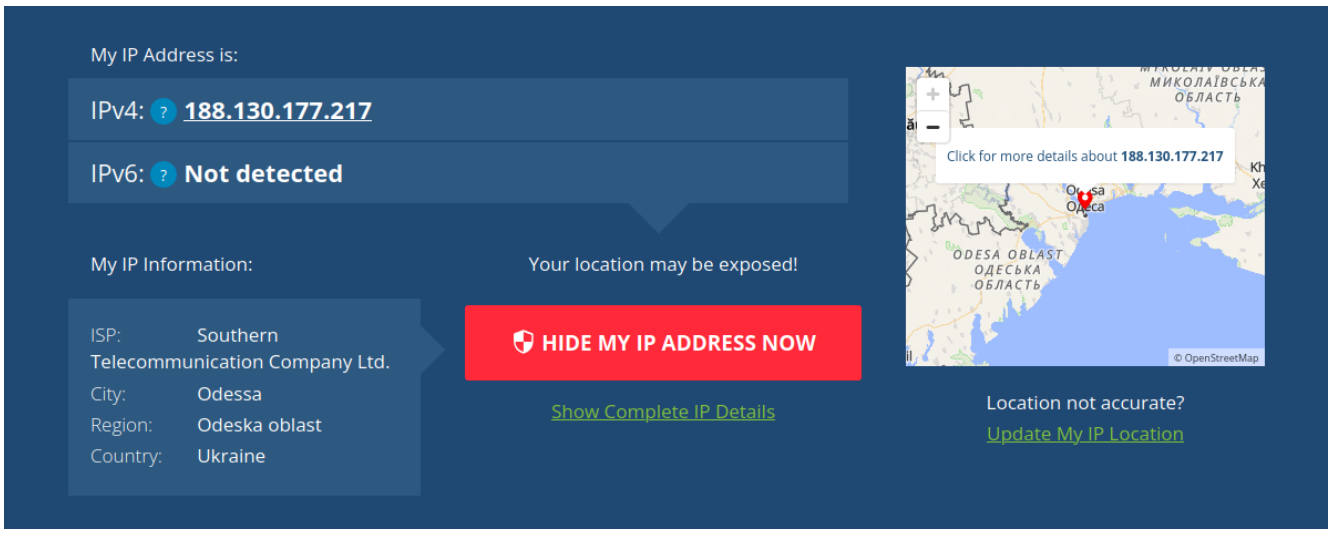


Рисунок 3.12 – Перевірка через сайт https://whatismyipaddress.com/ з відключеним VPN

Тестування VPN-сервера на базі WireGuard із Python-клієнтом показало його ефективність у приховуванні IP-адреси, захисті даних і забезпеченні доступу до Інтернету через віддалений сервер (у тесті – у Нідерландах). Система виявилася простою у використанні, швидкою та надійною, що робить її придатною для повсякденних сценаріїв, таких як захист у публічних мережах і обхід географічних обмежень. У порівнянні з Tor, VPN пропонує вищу швидкість і зручність, але поступається в анонімності через централізовану природу. Для максимального захисту рекомендується комбінувати VPN із Tor, використовувати надійні сервери та дотримуватися цифрової гігієни, щоб мінімізувати ризики деанонімізації. Ця система є потужним інструментом для забезпечення безпеки в сучасному цифровому середовищі, але потребує ретельного налаштування та усвідомленого підходу до використання.

ВИСНОВКИ

У результаті проведеного дослідження можна стверджувати, що технології VPN та TOR відіграють ключову роль у забезпеченні цифрової безпеки та анонімності користувачів в умовах постійного зростання кіберзагроз і спроб стеження за інтернет-активністю. Проведений аналіз дозволив глибше зрозуміти принципи функціонування цих інструментів, їхні сильні та слабкі сторони, а також специфіку застосування в різних умовах. Зокрема, було встановлено, що сучасні загрози, такі як перехоплення трафіку (Sniffing), атаки «людина посередині» (MitM) і підміна DNS, створюють серйозні ризики для конфіденційності, що підкреслює необхідність використання шифрування та анонімізації. VPN і Tor ефективно протидіють цим загрозам, захищаючи дані від несанкціонованого доступу та маніпуляцій.

Практична частина роботи засвідчила, що створення власного VPN-сервера з використанням протоколу WireGuard є не лише ефективним, а й реалістичним способом підвищення рівня захисту інтернет-з'єднання навіть для звичайного користувача. Вибір WireGuard обґрунтовано його мінімалістичною кодовою базою (4000 рядків), яка знижує ризик вразливостей, високою швидкістю завдяки сучасним алгоритмам шифрування (ChaCha20, Poly1305) і простотою налаштування. VPS на базі Ubuntu від DigitalOcean забезпечує гнучкість, повний контроль і надійність, а Python-клієнт із бібліотеками subprocess, requests, argparse і time автоматизує управління, роблячи систему доступною для широкого кола користувачів. Розроблений Python-клієнт забезпечує зручність управління з'єднанням і наочно демонструє зміну публічної IP-адреси, що є підтвердженням роботи захищеного тунелю. Крім того, експериментальне тестування підтвердило надійність налаштованої системи.

Досягнення високого рівня безпеки вимагає не лише VPN і Tor, а й додаткових інструментів. Спеціалізовані операційні системи, такі як Tails OS (автоматична маршрутизація через Tor), Whonix (ізоляція мережевого й робочого середовищ) і Qubes OS (мікровіртуалізація), мінімізують ризики витоків даних. Шифровані

месенджери (Signal, Session, Threema) з наскрізним шифруванням і мінімальними метаданими забезпечують конфіденційність спілкування. Цифрова гігієна, включаючи використання браузерів із захистом приватності (Brave, Firefox із розширеннями NoScript, uBlock Origin) і блокування трекінгу (WebRTC, Canvas fingerprinting), є необхідною для уникнення деанонімізації. Порівняльний аналіз VPN та TOR дозволив виокремити ключові аспекти, які варто враховувати під час вибору інструментів анонімізації залежно від конкретних потреб користувача – чи то забезпечення високої швидкості, чи максимального рівня конфіденційності.

Комбіноване використання обох технологій продемонструвало найвищу ефективність з точки зору захисту трафіку та уникнення деанонімізації. Конфігурація Tor через VPN приховує використання Tor від провайдера, тоді як VPN через Tor підвищує анонімність перед самим VPN-сервером, хоча й ускладнює налаштування. Ці підходи компенсують слабкі сторони один одного, створюючи багаторівневу систему захисту.

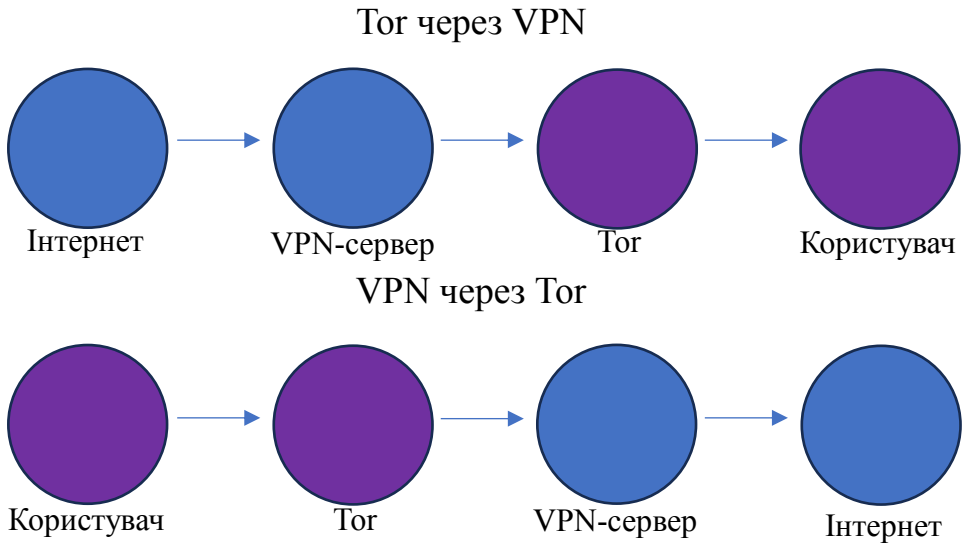
У підсумку, робота підтвердила доцільність та актуальність застосування комплексного підходу до захисту інтернет-трафіку, який поєднує технічні рішення з поведінковими стратегіями цифрової гігієни, а отримані результати можуть бути використані як основа для подальших розробок у сфері кібербезпеки та розширення функціоналу інструментів анонімності. Перспективи включають інтеграцію захисту DNS (DoH/DoT) для протидії підміні, розробку гібридних систем із підтримкою Tor і автоматизацію процесів безпеки, що може підвищити ефективність і доступність рішень для користувачів.

ПЕРЕЛІК ПОСИЛАНЬ

1. Muazam Ali. Virtual Private Network. URL: https://is.gd/g0EBNU
2. Dr. Eman Daraghmi. Virtual Private Network (VPN). URL: https://is.gd/ckrGjL
3. Types of Virtual Private Network (VPN) and its Protocols. URL: https://www.mppolytechnic.ac.in/mp-staff/notes_upload_photo/CS1Documents5.pdf
4. Arora, A., Garman, C. Improving the Performance and Security of Tor's Onion Services. Proceedings on Privacy Enhancing Technologies. 2025 № (1), URL: https://petsymposium.org/popets/2025/popets-2025-0029.php
5. Gudla, R., Vollala, S., Srinivasa, K. G., Amin, R. TCC: Time constrained classification of VPN and non-VPN traffic using machine learning algorithms. Wireless Networks, 2025. № 31(4), P. 3415–3429. DOI: https://doi.org/10.1007/s11276-025-03946-y URL: https://link.springer.com/article/10.1007/s11276-025-03946-y
6. Тимченко О.В., Гавриш Б.М. Аналіз анонімності користувачів мережі Tor. Наука і техніка сьогодні. № 3(44). 2025. С. 1501-1513. DOI: https://doi.org/10.52058/2786-6025-2025-3(44)-1501-1513 URL: http://perspectives.pp.ua/index.php/nts/article/view/21856
7. Barredo-Valenzuela, A. R., Portillo, S. P., & Suarez-Tangil, G. Snorkeling in dark waters: A longitudinal surface exploration of unique Tor Hidden Services (Extended Version) (Version 1). 2025. arXiv. DOI: https://doi.org/10.48550/ARXIV.2504.16836 URL: https://arxiv.org/abs/2504.16836v1
8. Василенко М., Рачук В., Слатвінська В. Аналіз несанкціонованого доступу в комп'ютерні мережі в контексті заходів щодо їх безпеки. Інформаційні технології та суспільство. 2024. № 2 (13). С. 12-16. DOI: https://doi.org/10.32689/maup.it.2024.2.2 URL: https://journals.maup.com.ua/index.php/it/article/view/3927
9. Nimisha P. Patel, Rajan G. Patel, Dr. Dhiren R. Patel. Packet Sniffing: Network Wiretapping. URL: https://is.gd/hjV3CI

10. Avijit Mallik. Man in the middle-attack: understandinf in simple words. Cyberspace: Jurnal Pendidikan Teknologi Informasi. 2019. № 2. 109. DOI: 10.22373/cj.v2i2.3453. URL: https://is.gd/hljNMa
11. Kukutla, Tejonath Reddy. A Deep Dive into DNS Spoofing and Security Measures. 2023. URL: https://is.gd/lqgeJl
12. PrivacyTools. PrivacyTools.io: Encryption against global mass surveillance. URL: https://www.privacytools.io
13. Tor Project. Tor: Overview. URL: https://www.torproject.org
14. Tails Project. Tails - The Amnesic Incognito Live System. URL: https://tails.boum.org
15. Qubes OS Docs. Qubes OS: A reasonably secure operating system. URL: https://www.qubes-os.org
16. EFF Privacy Badger. Stop trackers automatically. URL: https://privacybadger.org
17. PrivacyGuides. Secure and Private Software Recommendations. URL: https://www.privacyguides.org
18. OPSEC Handbook. Operational Security for Activists and Journalists. URL: https://opsecguide.org
19. WireGuard VPN. Що це і для чого? – SurfShark. URL: https://surfshark.com/uk/blog/wireguard-vpn
20. Revanth P S B.E. Blockchain-Secured and VPN Integrated Power Systems Management. Dandao Xuebao Journal of Ballistics, 2025. № 37(1), P. 25–34. https://doi.org/10.52783/dxjb.v37.166 URL: https://ballisticsjournal.com/index.php/journal/article/view/166

Додаток А. Схема взаємодії VPN та TOR



Додаток Б. Лістинг програмного коду для налаштування VPN-сервера

Vnp.py

```
import subprocess
import requests
import time

CONFIG_PATH = "/etc/wireguard/wg0-client.conf"

def check_ip():
    try:
        ip = requests.get("https://api.ipify.org", timeout=5).text
        print(f"[✓] IP: {ip}")
    except:
        print("[✗] Не вдалось отримати IP")

def connect_vpn():
    print("[*] Підключення до VPN...")
    subprocess.run(["sudo", "wg-quick", "up", CONFIG_PATH])
    time.sleep(2)
    check_ip()

def disconnect_vpn():
    print("[*] Відключення від VPN...")
    subprocess.run(["sudo", "wg-quick", "down", CONFIG_PATH])
    time.sleep(2)
    check_ip()

if __name__ == "__main__":
    import argparse

    parser = argparse.ArgumentParser()
    parser.add_argument("action", choices=["connect", "disconnect"])
    args = parser.parse_args()

    if args.action == "connect":
        connect_vpn()
    else:
        disconnect_vpn()
```