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

КВАЛІФІКАЦІЙНА РОБОТА

на тему: «АПАРАТНО-ПРОГРАМНИЙ КОМПЛЕКС ЗАХИЩЕНОГО КЕРУВАННЯ БПЛА»

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

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

_____ *Данило ГРУША*
(підпис) *Ім'я, ПРІЗВИЩЕ здобувача*

Виконав: здобувач вищої освіти гр. КІД-41
Данило ГРУША
Ім'я, ПРІЗВИЩЕ

Керівник: Ігор БУЧЕНКО
науковий ступінь, вчене звання Ім'я, ПРІЗВИЩЕ

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

Київ 2026

ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ

Навчально-науковий інститут Інформаційних технологій

Кафедра Комп'ютерної інженерії

Ступінь вищої освіти бакалавр

Спеціальність 123 Комп'ютерна інженерія

Освітньо-професійна програма «Комп'ютерна інженерія»

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

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

Груша Данило Романович
(прізвище, ім'я, по батькові здобувача)

1. Тема кваліфікаційної роботи: «Апаратно-програмний комплекс захищеного керування БПЛА» керівник кваліфікаційної роботи: Ігор БУЧЕНКО, старший викладач ,
(Ім'я, ПРІЗВИЩЕ, науковий ступінь, вчене звання)
затверджені наказом Державного університету інформаційно-комунікаційних технологій від «20» лютого 2026р. №51
2. Строк подання кваліфікаційної роботи«14» травня 2026р.
3. Вихідні дані до кваліфікаційної роботи: науково-технічна література, стандарти та протоколи захищеного зв'язку БПЛА, архітектура апаратно-програмних комплексів керування, VPN-технології та криптографічні алгоритми захисту даних, методи мережевої фільтрації й контролю доступу, засоби емуляції та аналізу трафіку, тестування і порівняльне оцінювання ефективності системи.
4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)
    1. Теоретичні основи захисту каналів зв'язку безпілотних систем
    2. Практична розробка емульованого середовища захищеного керування
    3. Аналітичне оцінювання ефективності та перспективи розвитку системи
5. Перелік ілюстративного матеріалу: *презентація*
    1. Мета, об'єкт, предмет дослідження
    2. Актуальність роботи
    3. Базова архітектура керування БПЛА
    4. Запропоноване рішення
    5. Практична реалізація стенду
    6. Налаштування OpenWrt-вузла
    7. Демонстрація роботи системи
    8. Аналіз трафіку
    9. Оцінка ефективності

10. Методи покращення системи
11. Практичне впровадження
12. Висновки
13. Публікації та апробація роботи

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

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Збір даних	22.02-20.03.2026	Виконано
2	Обробка матеріалу	20.03-30.03.2026	Виконано
3	Розробка теоретичної частини	31.03-10.04.2026	Виконано
4	Практична розробка емульованого середовища захищеного керування	11.04-20.04.2026	Виконано
5	Висновки за результатами аналізу	21.04-26.04.2026	Виконано
6	Оформлення пояснювальної записки	26.04-28.04.2026	Виконано
7	Розробка презентації	29.04-01.05.2026	Виконано
8	Передзахист	08.05-11.05.2026	Виконано
9	Здача в деканат	08.05-14.05.2026	Виконано

Здобувач(ка) вищої освіти _____ Данило ГРУША
(підпис) *(Ім'я, ПРІЗВИЩЕ)*

Керівник кваліфікаційної роботи _____ Ігор БУЧЕНКО
(підпис) *(Ім'я, ПРІЗВИЩЕ)*

РЕФЕРАТ

Текстова частина кваліфікаційної роботи на здобуття освітнього ступеня бакалавра: 86 стор., 33 рис., 6 табл., 30 джерел.

Мета роботи – підвищення безпеки та надійності керування БПЛА шляхом використання тунелювання трафіку, криптографічного захисту, фільтрації доступу та сегментації каналів зв'язку.

Об'єкт дослідження – процеси передавання команд керування, телеметрії та службового трафіку у системах дистанційного керування БПЛА.

Предмет дослідження – засоби й технології побудови апаратно-програмних комплексів захищеного керування БПЛА на основі VPN, криптографічних алгоритмів, мережевої фільтрації та емуляційних середовищ.

Короткий зміст роботи: У роботі представлено теоретичні основи побудови каналів зв'язку безпілотних систем, архітектуру мережевої взаємодії БПЛА різних класів, особливості протоколів обміну даними, типові вразливості безпровідних каналів та методи протидії перехопленню, підміні сигналів і засобам радіоелектронної боротьби. Розглянуто криптографічні підходи до захищеного передавання даних у реальному часі, принципи інкапсуляції трафіку, сегментації потоків та побудови наземних станцій керування.

Особливу увагу приділено практичній реалізації емульованого середовища захищеного керування БПЛА із використанням WireGuard, VPN-шлюзу (на базі OpenWrt 23.05.5), брандмауера та алгоритмів ізоляції прикладного рівня. Проведено тестування встановлення з'єднання, аналіз стійкості до перехоплення трафіку, оцінювання впливу криптографічних перетворень на затримки та пропускну здатність, а також визначено перспективи інтеграції рішення у серійні платформи й адаптації до умов радіоелектронної протидії.

КЛЮЧОВІ СЛОВА: БПЛА, ЗАХИЩЕНЕ КЕРУВАННЯ, VPN, WIREGUARD, ТЕЛЕМЕТРІЯ, КРИПТОГРАФІЧНИЙ ЗАХИСТ, МЕРЕЖЕВА ФІЛЬТРАЦІЯ, ТУНЕЛЮВАННЯ, VPN-ШЛЮЗ, ЕМУЛЯЦІЯ, РЕБ.

ЗМІСТ

ВСТУП 8
1. ТЕОРЕТИЧНІ ОСНОВИ ЗАХИСТУ КАНАЛІВ ЗВ'ЯЗКУ БЕЗПІЛОТНИХ СИСТЕМ 9
1.1 Архітектура мережевої взаємодії БПЛА різних класів 9
1.2 Вразливості та моделювання загроз безпровідних каналів 17
1.3 Технології захищеного передавання даних у реальному часі 25
1.4 Інфраструктура наземних станцій управління (GCS) 31
1.5 Нормативна база та стандарти безпеки в галузі БпАК 39
2. ПРАКТИЧНА РОЗРОБКА ЕМУЛЬОВАНОГО СЕРЕДОВИЩА ЗАХИЩЕНОГО КЕРУВАННЯ 46
2.1 Проєктування архітектури тестового стенду 46
2.2 Побудова апаратно-програмного комплексу 53
2.3 Алгоритми обробки та захисту даних 61
2.4 Тестування безпеки та надійності комплексу 67
3. АНАЛІТИЧНЕ ОЦІНЮВАННЯ ЕФЕКТИВНОСТІ ТА ПЕРСПЕКТИВИ РОЗВИТКУ СИСТЕМИ 75
3.1 Аналітичне та практичне оцінювання продуктивності захищеної системи 75
3.2 Оптимізація та адаптація системи захисту 80
3.3 Напрямки практичного впровадження та масштабування 86
ВИСНОВКИ 93
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ 94
Додаток А. ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ (Презентація) 96

ВСТУП

Розвиток безпілотних літальних апаратів зумовив зростання вимог до надійності каналів керування, стабільності телеметричного обміну та захисту переданих даних. Сучасні БПЛА різних класів використовують складні схеми мережевої взаємодії, у межах яких одночасно передаються команди оператора, службова телеметрія, відеопотоки та діагностична інформація. Для важких ударних платформ пріоритетом є дальність і стійкість зв’язку, тоді як для швидкісних FPV-систем критичними залишаються мінімальна затримка та швидкість реакції каналу.

Поряд із технічним розвитком суттєво зростає кількість загроз для безпровідних систем керування. До найбільш небезпечних належать перехоплення команд, підміна сигналів, спроби несанкціонованого доступу до відеопотоків, а також навмисне створення перешкод засобами радіоелектронної боротьби. Вразливість відкритих каналів передавання даних підтверджує необхідність використання сучасних методів криптографічного захисту.

Особливого значення набуває вибір технологій захищеного передавання даних у реальному часі. Для систем керування БПЛА важлива не лише криптографічна стійкість, оскільки затримки впливають на ефективність польоту. Тому необхідний баланс між швидкодією, пропускною здатністю та рівнем безпеки.

Практична частина роботи спрямована на створення тестового апаратно-програмного комплексу захищеного керування БПЛА. Планується реалізація архітектури з наземною станцією, VPN-тунелюванням WireGuard, VPN-шлюзом із брандмауером та алгоритмами захисту телеметрії. Передбачено тестування з’єднання, аналіз стійкості до перехоплення трафіку та оцінювання можливості подальшого впровадження у реальні системи.

1. ТЕОРЕТИЧНІ ОСНОВИ ЗАХИСТУ КАНАЛІВ ЗВ'ЯЗКУ БЕЗПІЛОТНИХ СИСТЕМ

1.1 Архітектура мережевої взаємодії БПЛА різних класів

Архітектура мережевої взаємодії безпілотних літальних апаратів визначається їхнім функціональним призначенням, масогабаритними характеристиками, типом корисного навантаження та сценаріями застосування, у сучасних умовах можна виділити кілька основних класів БПЛА, які широко використовуються та мають суттєві відмінності у підходах до організації зв’язку, серед них особливу увагу привертають важкі ударні коптери та швидкісні FPV-дрони, різниця між цими класами полягає не лише у фізичних характеристиках, але й у принципах побудови мережевої інфраструктури, де для важких платформ визначальними є стабільність і дальність зв’язку, тоді як для FPV-систем критичною стає мінімальна затримка та миттєва реакція на дії оператора.

У загальному вигляді архітектура мережевої взаємодії БПЛА включає кілька ключових компонентів, серед яких [1]:

— канал керування C2 (Command & Control);

— канал телеметрії;

— канал передачі відео (аналоговий або цифровий);

— допоміжні канали.

Останні забезпечують навігацію, ретрансляцію та резервування зв’язку, взаємодія цих складових формує комплексну багаторівневу систему, яка може адаптуватися до умов експлуатації, рівня завад та вимог до безпеки, більш детальне розуміння архітектурних особливостей досягається через аналіз конкретних представників різних класів, зокрема важкого коптера «Вампір» та FPV-дрона «Shrike».

Технічний аналіз важких ударних коптерів (на прикладі комплексу «Вампір»). Важкі коптери типу комплексу «Вампір» належать до класу багатоцільових

безпілотних платформ із високою вантажопідйомністю та тривалим часом польоту, гексакоптерна конфігурація з шістьма гвинтами забезпечує підвищену стабільність у польоті та можливість роботи в складних умовах, до складу комплексу входить бортовий сегмент, який представлений носієм і розвідником типу «UKROPTER», наземна станція управління, навігаційні маяки, периферійні пристрої у вигляді моніторів і пультів керування, а також енергетична підсистема на базі акумуляторів конфігурації 6S4P, що забезпечує тривалу автономну роботу (див. рис. 1.1).



Рисунок 1.1 – Зовнішній вигляд важкого ударного коптера типу «Вампір»

Мережева архітектура такого комплексу має багатокомпонентний характер і реалізується через кілька незалежних каналів зв’язку, канал керування C2 забезпечує передачу команд від оператора до апарата та характеризується підвищеними вимогами до надійності, відносно помірними вимогами до затримки та обов’язковою наявністю захисту від несанкціонованого доступу, телеметричний канал передає інформацію про стан апарата, включаючи координати, швидкість, рівень заряду акумулятора та інші параметри польоту, передача відбувається періодично і допускає певні затримки без критичного впливу на систему [2-3].

Особливе місце займає відеоканал, зокрема тепловізійний потік, який є критично важливим для нічних операцій, передача такого потоку супроводжується значними обсягами даних, що вимагає стабільного каналу зв’язку та застосування цифрових технологій передачі, специфіка тепловізійного відео полягає у необхідності збереження якості зображення, тоді як допустима затримка може бути дещо більшою порівняно з FPV-системами, для оптимізації передачі застосовуються алгоритми стиснення, однак це створює додаткові вимоги до обробки даних.

Відеоканал у таких системах є потенційною точкою витоку інформації, оскільки передача незахищеного потоку може бути перехоплена, у зв’язку з цим виникає необхідність або криптографічного захисту, або фізичної ізоляції каналу від інших потоків, що досягається через сегментацію мережі або використання окремих частотних діапазонів (див. рис. 1.2).

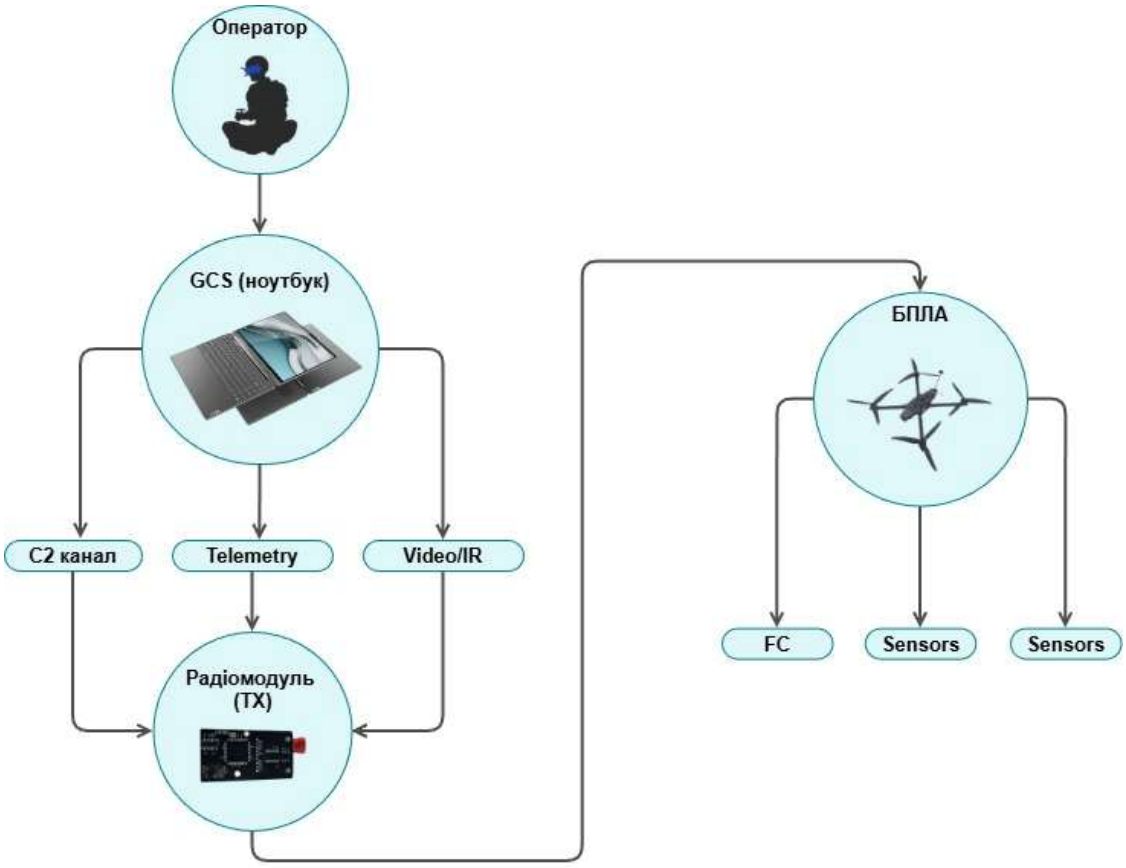


Рисунок 1.2 – Схема передавання даних між
оператором і бортовими модулями важкого БПЛА

З точки зору мережевої моделі, для комплексу «Вампір» характерна багаторівнева архітектура, яка включає основний канал зв’язку, резервні канали та допоміжні системи навігації, використання потужних акумуляторів і обчислювальних ресурсів дозволяє реалізовувати складні алгоритми обробки даних, включаючи шифрування, фільтрацію та аналіз трафіку, що є суттєвою перевагою порівняно з легкими безпілотними системами [1-2].

Пріоритетами для таких платформ є:

— стабільність роботи;

— можливість використання багатоканальної архітектури;

— застосування складних криптографічних методів;

— підвищені вимоги до захисту інформаційних потоків.

Специфіка швидкісних FPV-систем (на прикладі дрона «Shrike»). Швидкісні FPV-дрони, зокрема апарати типу «Shrike», належать до окремого класу безпілотних систем, орієнтованих передусім на високу швидкість, маневреність та мінімальну затримку керування, їхні характеристики формуються з урахуванням динамічних сценаріїв застосування, де вирішальним фактором стає швидкість реакції оператора, типовий радіус дії становить близько 5 км із можливістю розширення через використання ретрансляторів, тривалість польоту може перевищувати 40 хвилин залежно від ємності акумуляторів, а загальна архітектура системи має компактний і мінімалістичний характер, що пояснюється обмеженими апаратними ресурсами та необхідністю мінімізувати затримки у передачі сигналів (див. рис. 1.3).



Рисунок 1.3 – Зовнішній вигляд швидкісного FPV-дрона типу «Shrike»

Архітектурна структура взаємодії у таких системах зазвичай включає пульт керування типу TX12, який використовується як універсальний інтерфейс введення команд, радіоканал керування, відеоканал, що реалізується у вигляді аналогового або низькозатримкового цифрового сигналу, при цьому на практиці часто перевага надається саме аналоговому відео через його здатність забезпечувати стабільніший

прийом у складних умовах та кращу проникність сигналу, що особливо важливо при польотах на низьких висотах або у забудованій місцевості, уся система орієнтована на досягнення ультранизької затримки, де навіть незначне її збільшення може призвести до втрати контролю або помилок у керуванні, що безпосередньо впливає на ефективність застосування апарата (див. рис. 1.4) [1].



Рисунок 1.4 – Універсальний пульт керування типу TX12 та ноутбук-трансформер

На відміну від важких коптерів, телеметрія у FPV-дронах має другорядне значення і використовується обмежено, оскільки основним каналом сприйняття інформації оператором залишається відеопотік, передача команд здійснюється у реальному часі без буферизації або додаткової обробки, що дозволяє досягти максимальної швидкості реакції, але водночас зменшує можливості для додаткового захисту або контролю цілісності даних, що створює певний компроміс між швидкодією та безпекою.

Важливим елементом розширення функціональних можливостей таких систем є ретранслятори, які дозволяють збільшити дальність зв’язку та забезпечити роботу за межами прямої видимості, вони можуть реалізовуватися як наземні або

повітряні вузли, що приймають сигнал і передають його далі, у такий спосіб формується багаторівнева структура передачі даних, однак використання ретрансляції супроводжується появою додаткових затримок, оскільки сигнал проходить через проміжні вузли, а також може призводити до погіршення якості сигналу внаслідок втрат під час повторної передачі, разом із цим зростає складність маршрутизації та з'являються додаткові точки вразливості, які можуть бути використані для атак або перехоплення [1].

Особливості передачі даних у FPV-дронах визначаються використанням легких протоколів і мінімальної обробки інформації, у багатьох випадках шифрування або не застосовується, або використовується частково, що дозволяє зменшити накладні витрати та затримки, але одночасно підвищує ризики несанкціонованого доступу до каналів зв'язку, таким чином формується характерний компроміс, де висока швидкість досягається за рахунок зниження рівня захищеності.

Огляд стеків протоколів обміну даними (MAVLink, ExpressLRS, Crossfire). Протоколи обміну даними у системах БПЛА визначають формат і спосіб передачі інформації між компонентами, серед найбільш поширених рішень виділяється MAVLink, який є легковаговим протоколом для передачі телеметрії та команд управління, його перевагами є універсальність, підтримка більшістю систем, зокрема програмного забезпечення типу QGroundControl та ArduPilot, а також простота інтеграції, серед недоліків слід відзначити відсутність вбудованого шифрування, що вимагає застосування додаткових засобів захисту, наприклад тунелювання або використання захищених каналів передачі (див. рис. 1.5) [2-4].

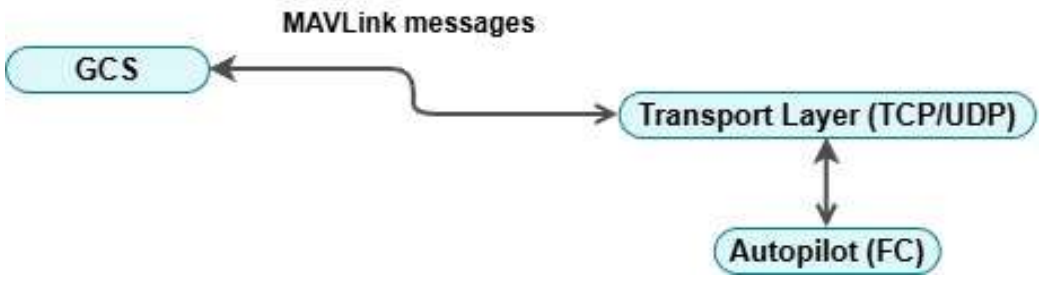


Рисунок 1.5 – Принцип обміну телеметрією через MAVLink

Іншим поширеним рішенням є ExpressLRS, сучасний протокол радіоуправління, орієнтований на FPV-дрони, який забезпечує високу швидкість передачі даних, низьку затримку та адаптивну зміну частоти, що дозволяє ефективно працювати у динамічних умовах і при наявності завад, його перевагами є велика дальність, стійкість до перешкод та оптимізація під швидкісні сценарії, водночас обмеженням виступає недостатній рівень криптографічного захисту, оскільки основний акцент зроблено на продуктивності [5].

Протокол Crossfire також широко використовується у системах радіоуправління та забезпечує стабільний зв’язок на великих дистанціях, він оптимізований для передачі керуючих сигналів, підтримує телеметрію та інтегрується з професійним обладнанням, його перевагами є висока надійність і якість сигналу, серед недоліків відзначається пропрієтарний характер, що обмежує гнучкість налаштування та створює залежність від виробника (див. табл. 1.1).

Таблиця 1.1 – Порівняльна характеристика протоколів та засобів передавання даних у системах БПЛА

Протокол	Призначення	Тип даних	Затримка	Захист	Сфера застосування
MAVLink	Телеметрія та команди керування	Пакети телеметрії, статус, waypoint	Низька	Відсутній вбудований	Autopilot, GCS, ArduPilot, PX4
ExpressLRS	Радіоканал керування FPV	RC-команди, базова телеметрія	Дуже низька	Базовий	FPV-дрони, швидкісні системи
Crossfire	Дальній радіозв’язок	RC-команди, телеметрія	Дуже низька	Частковий	Професійні FPV та long-range
TLS/VPN	Захищене тунелювання трафіку	Будь-які IP-дані	Середня	Високий	Захист каналів GCS ↔ БПЛА

Загальний підхід до вибору протоколів визначається класом БПЛА та вимогами до системи, для важких платформ доцільним є використання MAVLink у поєднанні з додатковими засобами захисту, тоді як для FPV-дронів перевага

надається ExpressLRS або Crossfire через їхню низьку затримку та здатність працювати на великих дистанціях, у практичних реалізаціях часто застосовується комбінований підхід, при якому різні протоколи використовуються для різних каналів, наприклад MAVLink для телеметрії із захищеним тунелюванням, а швидкі протоколи — для критичного керування, у результаті архітектура мережевої взаємодії формується з урахуванням балансу між швидкістю, надійністю та безпекою, що визначається конкретним призначенням і класом безпілотної системи [2-3; 6].

1.2 Вразливості та моделювання загроз безпровідних каналів

Безпровідні канали зв’язку становлять фундаментальну основу функціонування безпілотних літальних апаратів, водночас формуючи одну з найбільш уразливих площин для реалізації атак, оскільки використання відкритого радіоефіру означає, що будь-який сигнал може бути перехоплений, проаналізований і потенційно змінений за наявності відповідних технічних засобів, відсутність фізичного контролю середовища передачі суттєво відрізняє такі системи від провідних мереж, де доступ до каналу обмежується інфраструктурою, радіосигнали ж поширюються у просторі без чітких меж, що створює умови для їх доступності сторонніми суб’єктами, у контексті БПЛА ця особливість проявляється по-різному залежно від класу апарата, для важких платформ характерними є ризики перехоплення телеметрії та відеопотоку, тоді як для швидкісних FPV-дронів критичними стають затримки та порушення керування, моделювання загроз передбачає аналіз потенційних сценаріїв атак, оцінювання їхнього впливу на систему та визначення рівня стійкості до таких впливів, серед основних категорій загроз виділяються перехоплення і підміна сигналів, радіоелектронне подавлення та несанкціонований доступ до відеоданих, кожен із

цих напрямів формує окремий вектор впливу, що може призвести до порушення функціонування всієї системи керування.

Методики перехоплення та підміни керуючих сигналів (Spoofing). Перехоплення та підміна керуючих сигналів, відомі як spoofing, належать до найбільш небезпечних загроз, оскільки дозволяють безпосередньо впливати на поведінку апарата або навіть повністю перехоплювати керування, принцип реалізації такої атаки включає кілька послідовних етапів, спочатку здійснюється пасивне перехоплення сигналу за допомогою приймального обладнання, після чого відбувається аналіз структури переданих даних із визначенням формату пакетів, частоти передачі та типу протоколу, на основі отриманої інформації формується генерація підроблених пакетів, які імітують легітимні команди, далі відбувається передача цих сигналів до БПЛА, при цьому для досягнення ефекту використовуються або більша потужність передавача, або менша відстань до об’єкта, що дозволяє «перебити» оригінальний сигнал, особливістю безпровідного керування є відсутність необхідності повторного підключення, як у випадку мереж Wi-Fi, оскільки команди передаються постійно, і система приймає той сигнал, який має кращі параметри прийому, що значно спрощує реалізацію атаки (див. рис. 1.6) [7].

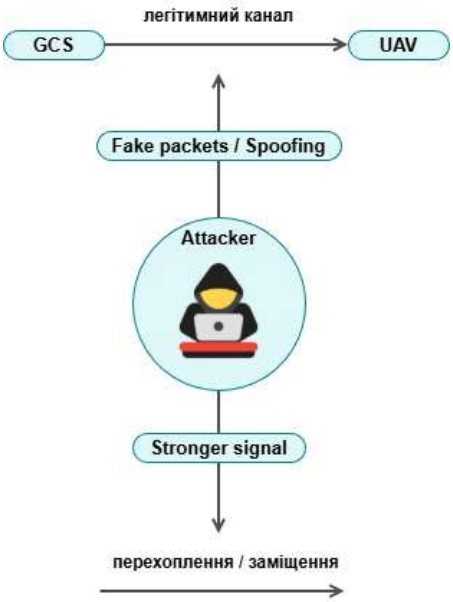


Рисунок 1.6 – Перехоплення каналу між GCS та БПЛА

Методики реалізації spoofing можуть включати повторну передачу перехоплених пакетів, що відома як replay-атака, вставку нових команд у потік даних, підміну навігаційних сигналів у вигляді GPS spoofing, а також імітацію протоколів обміну даними, зокрема таких як MAVLink, вплив подібних атак залежить від типу безпілотної системи, для важких коптерів можливе спотворення маршруту, зміна координат або режимів польоту, що призводить до порушення виконання місії, для FPV-дронів навіть короткочасне втручання може спричинити втрату контролю через високу залежність від безперервності сигналу, серед основних причин вразливості виступають передача даних у відкритому вигляді, відсутність механізмів автентифікації, використання статичних ключів або їх повна відсутність, що є типовим для багатьох легких протоколів радіоуправління [8].

Радіоелектронне подавлення, або jamming, являє собою інший клас загроз, спрямований не на підміну сигналу, а на повне або часткове порушення його передачі, принцип дії полягає у створенні потужних радіоперешкод у робочому діапазоні частот, що призводить до деградації якості зв’язку або його повної втрати, залежно від потужності джерела завад і відстані до БПЛА можливе як локальне, так і широкомасштабне подавлення, вплив такої атаки також відрізняється залежно від типу системи, важкі коптери зазвичай мають резервні канали зв’язку або алгоритми повернення у разі втрати сигналу, що дозволяє частково компенсувати вплив, однак при тривалому подавленні можливе повне порушення управління, для FPV-дронів навіть короткочасне зникнення сигналу означає втрату контролю, оскільки управління здійснюється в режимі реального часу без буферизації, додатковим фактором ризику є використання ретрансляторів, які можуть стати окремими точками впливу, оскільки їхнє подавлення призводить до розриву всього каналу зв’язку [8-9].

Несанкціонований доступ до відеопотоків формує ще один важливий напрям загроз, пов’язаний із витоком інформації, у багатьох системах відео передається без шифрування або з мінімальним рівнем захисту, що дозволяє стороннім особам здійснювати перехоплення сигналу за допомогою відповідного обладнання, особливо це стосується аналогових FPV-систем, де сигнал транслюється у

відкритому вигляді, у випадку цифрових систем ризик зменшується, однак за відсутності належного шифрування або автентифікації доступ до відеоданих також залишається можливим, витік відео може призвести до розкриття позицій, маршрутів руху або інших критичних параметрів, що створює загрозу не лише для окремого апарата, а й для всієї операції.

Узагальнення вразливостей безпровідних каналів зв'язку дозволяє визначити, що основними факторами ризику є:

— відкритість середовища передачі;

— відсутність фізичного контролю;

— використання спрощених протоколів без вбудованого захисту;

— обмежені апаратні ресурси.

Все це ускладнює впровадження складних механізмів безпеки, у таких умовах моделювання загроз стає необхідним етапом проєктування систем, оскільки дозволяє заздалегідь оцінити потенційні сценарії атак і визначити ефективні методи протидії, серед яких застосування шифрування, автентифікації, нумерації пакетів, перевірки цілісності та ізоляції каналів, поєднання цих підходів формує основу для побудови захищених каналів зв'язку, здатних протистояти основним типам атак у безпровідному середовищі.

Атаки на відмову в обслуговуванні (Jamming/РЕБ) та їх вплив на різні типи БПЛА. Атаки на відмову в обслуговуванні у безпровідних системах реалізуються переважно у вигляді радіоелектронного подавлення, відомого як jamming, метою якого є блокування або суттєве погіршення зв'язку між оператором і безпілотним літальним апаратом, або між окремими вузлами системи, принцип дії базується на створенні перешкод у радіочастотному спектрі шляхом генерації шуму або сигналів, що перекривають корисний канал, у результаті чого приймач втрачає можливість коректно декодувати передані дані через перевищення рівня шуму над сигналом, у практичних умовах це призводить до деградації якості зв'язку або повної його втрати, що особливо критично для систем, які працюють у режимі реального часу.

Залежно від способу реалізації виділяються різні типи радіоелектронного подавлення, широкосмугове подавлення охоплює весь діапазон частот і створює суцільний шум у ефірі, що унеможливлює передачу даних на будь-якому каналі, вузькосмугове подавлення спрямоване на конкретну частоту або канал, що робить його більш енергоефективним і точковим, імпульсне подавлення створює періодичні завади, які порушують цілісність передачі пакетів і викликають помилки декодування, інтелектуальне подавлення адаптується до поведінки системи зв’язку, аналізує її параметри та динамічно змінює характеристики завад для досягнення максимального ефекту, у сукупності ці підходи дозволяють впливати як на прості, так і на складні системи передачі даних [10].

Вплив радіоелектронного подавлення суттєво відрізняється залежно від класу безпілотного апарата, для важких платформ, подібних до комплексів типу «Вампір», характерне часткове зниження якості зв’язку при наявності завад, однак такі системи часто оснащуються резервними каналами або алгоритмами автономної поведінки, що дозволяє зменшити критичність короткочасних втрат сигналу, інерційність польоту та більша стабільність конструкції також сприяють збереженню керованості у складних умовах, при тривалому впливі або високій потужності завад можливий перехід у автономний режим або повна втрата зв’язку з наземною станцією, що впливає на виконання місії [8; 10].

Для швидкісних FPV-дронів ситуація є значно більш критичною, оскільки управління здійснюється у режимі реального часу і залежить від безперервного надходження сигналу, навіть короткочасні затримки або втрати пакетів можуть призвести до миттєвої втрати контролю, у більшості випадків відсутні резервні канали зв’язку, а сильна залежність від стабільного відеосигналу додатково ускладнює ситуацію, оскільки порушення відеоканалу фактично позбавляє оператора можливості орієнтації, таким чином навіть відносно слабкі завади можуть мати критичні наслідки для таких систем (див. рис. 1.7).

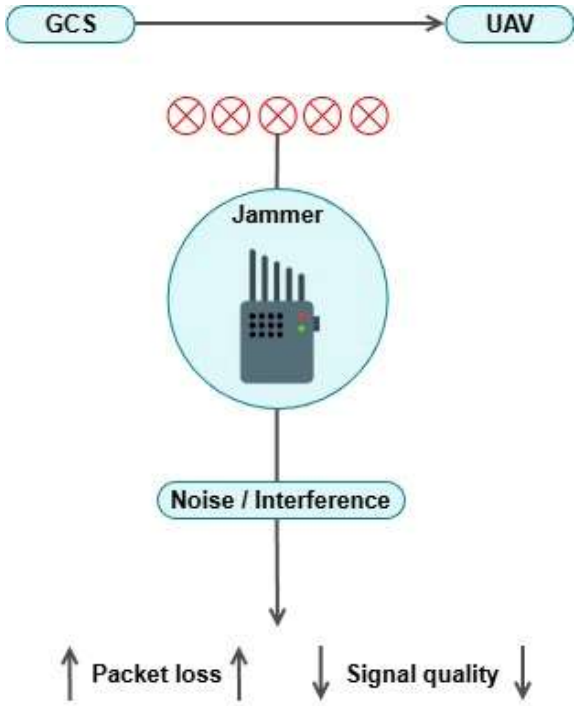


Рисунок 1.7 – Вплив радіоелектронного придушення на канал керування

Ефективність атак радіоелектронного подавлення визначається низкою факторів, серед яких важливими є потужність передавача завад, що прямо впливає на радіус дії, відстань до безпілотного апарата, яка визначає рівень сигналу, що надходить до приймача, тип використовуваного протоколу зв'язку та його стійкість до перешкод, а також наявність адаптивних механізмів, таких як зміна частоти або використання резервних каналів, у сукупності ці параметри визначають, наскільки ефективно система може протистояти впливу завад [9-10].

Методи протидії радіоелектронному подавленню включають використання технологій частотного перестрибування, що дозволяє змінювати робочий діапазон і уникати завад, застосування різних частотних діапазонів для розподілу каналів зв'язку, впровадження резервних каналів, зокрема на базі альтернативних технологій передачі даних, а також використання автономних алгоритмів, які дозволяють апарату виконувати базові функції у разі втрати зв'язку, додатково застосовується пріоритезація трафіку, при якій критичні дані передаються з вищим пріоритетом, що дозволяє зберегти керованість навіть при обмеженій пропускній

здатності каналу, аналіз затримок і стабільності сигналу використовується для оцінки стану каналу та адаптації параметрів передачі.

Ризики несанкціонованого доступу до курсових та тепловізійних відеопотоків. Відеопотік у системах безпілотних апаратів є одним із найбільш інформативних каналів, оскільки забезпечує оператору безпосереднє візуальне сприйняття ситуації, водночас він часто залишається одним із найменш захищених, що створює значні ризики несанкціонованого доступу, курсове відео використовується для орієнтації у просторі, а тепловізійні потоки застосовуються у нічних або складних умовах видимості, що підвищує їхню цінність з точки зору потенційного перехоплення.

Основними ризиками є можливість перехоплення відеосигналу, що дозволяє визначити місцезнаходження апарата або оператора, витік тактичної інформації, включаючи маршрути руху та об’єкти спостереження, аналіз поведінки оператора, а також потенційна підміна відеопотоку або вставка стороннього зображення, що може призвести до дезорієнтації, у FPV-дронах вразливість підсилюється використанням аналогових систем передачі, відсутністю шифрування та простотою прийому сигналу, у важких системах застосовується цифрове відео, що дозволяє впроваджувати шифрування, однак збільшений обсяг даних і висока цінність інформації роблять такі канали привабливою ціллю для атак (див. рис. 1.8).

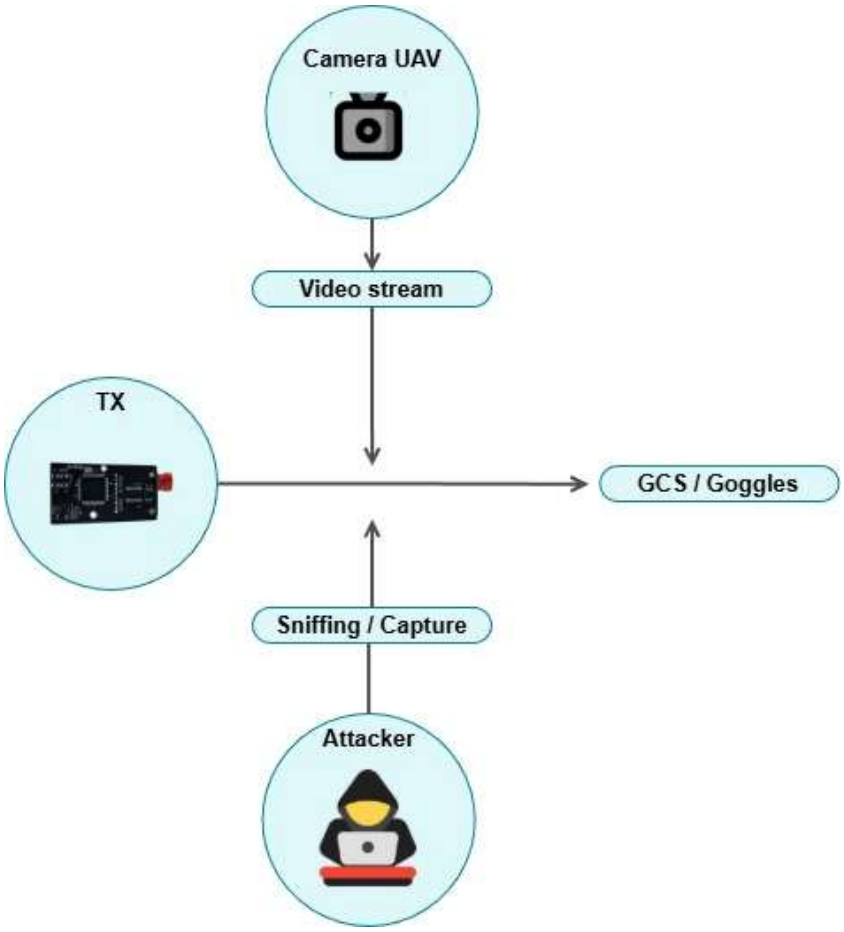


Рисунок 1.8 – Ризики несанкціонованого доступу до відеопотоку БПЛА

Методи атак на відеоканали включають:

1. пасивне прослуховування сигналу;
2. аналіз спектра для визначення параметрів передачі;
3. використання спеціалізованих приймачів для декодування відео;
4. вплив на інфраструктуру передачі.

Все це може призводити до порушення або перехоплення потоку, методи захисту передбачають використання шифрування відеоданих, застосування протоколів із вбудованими механізмами захисту, сегментацію мережі з розділенням каналів керування та передачі даних, обмеження доступу до каналів, а також оптимізацію потужності сигналу для зменшення ймовірності перехоплення.

Розділення каналів керування, телеметрії та відео дозволяє локалізувати потенційні атаки та зменшити їхній вплив на всю систему, у такій архітектурі

порушення одного каналу не призводить до повної втрати функціональності, що підвищує загальну стійкість, поєднання методів шифрування, тунелювання, сегментації та моніторингу стану каналів формує комплексний підхід до захисту безпровідних систем, який дозволяє підвищити рівень безпеки та забезпечити надійне функціонування безпілотних апаратів навіть у складних умовах радіоелектронного впливу [10-11].

1.3 Технології захищеного передавання даних у реальному часі

Передавання даних у реальному часі в системах безпілотних літальних апаратів визначається як критичний процес, від якого залежить можливість керування, моніторингу стану та оперативного прийняття рішень, при цьому відкритий характер безпровідного середовища створює умови для виникнення широкого спектра загроз, що вимагає впровадження спеціалізованих механізмів захисту, складність реалізації яких зумовлюється необхідністю дотримання жорстких обмежень, серед яких мінімальна затримка, обмежена пропускна здатність каналу, а також обчислювальні обмеження бортового обладнання, додатково враховується потреба у безперервності передачі, оскільки навіть короткочасні розриви можуть призвести до втрати керування або некоректної інтерпретації даних, у таких умовах технології захисту повинні забезпечувати:

1. баланс між криптографічною стійкістю;
2. ефективністю використання каналу;
3. швидкістю обробки інформації.

Все це зумовлює використання комбінованих підходів, серед яких виділяються інкапсуляція і тунелювання трафіку, застосування криптографічних алгоритмів та сегментація мережевих потоків [30].

Принципи інкапсуляції та побудови специфічних тунелів (на прикладі TLS/SSL). Інкапсуляція виступає базовим принципом побудови захищених каналів,

який полягає у вкладенні одного протоколу в інший із додаванням службової інформації, у контексті безпілотних систем такий підхід означає обгортання телеметричних і керуючих даних у захищений транспортний шар, що дозволяє приховати їхній вміст і структуру від сторонніх спостерігачів, на відміну від класичних VPN-рішень, які створюють універсальні тунелі для різних типів трафіку, у системах БПЛА доцільним є використання спеціалізованих тунелів, оптимізованих під конкретні протоколи, наприклад MAVLink або інші IoT-орієнтовані формати, що дозволяє зменшити накладні витрати та забезпечити необхідну швидкодію, надмірна універсальність стандартних VPN може призводити до збільшення затримок, що є неприйнятним для систем реального часу.

У якості основи для побудови захищених тунелів широко використовується TLS/SSL, який забезпечує:

— шифрування даних;
— автентифікацію сторін;
— контроль цілісності переданої інформації.

Процес встановлення з’єднання включає:

— ініціалізацію взаємодії;
— обмін параметрами шифрування;
— генерацію спільного ключа;
— формування захищеного каналу.

Далі передавання даних відбувається у зашифрованому вигляді, у практичному застосуванні для БПЛА телеметричні пакети або команди управління інкапсулюються у TLS-тунель, передаються мережею та на приймальній стороні дешифруються для подальшої обробки наземною станцією або автопілотом, перевагами такого підходу є незалежність від прикладного протоколу, можливість інтеграції з існуючими системами та високий рівень захисту, до недоліків належать збільшення розміру пакетів і додаткові обчислювальні витрати, що можуть впливати на затримку, у важких платформах використання TLS є доцільним у

повному обсязі, тоді як для FPV-систем необхідна оптимізація або часткове застосування через обмеження за затримкою [12].

Криптографічні методи захисту інформації забезпечують конфіденційність, цілісність і автентичність даних, однак їх застосування у системах реального часу потребує врахування компромісу між стійкістю алгоритмів і швидкістю їх виконання, симетричні алгоритми, такі як AES, забезпечують високу швидкість шифрування і є придатними для потокової передачі даних, асиметричні методи використовуються переважно для обміну ключами та автентифікації через їхню більшу обчислювальну складність, у сучасних реалізаціях часто застосовується комбінований підхід, при якому асиметричні алгоритми використовуються на етапі встановлення з'єднання, після чого передача даних здійснюється із застосуванням симетричного шифрування, додатково використовуються механізми контролю цілісності, такі як MAC або HMAC, які дозволяють виявляти підміну або пошкодження пакетів, важливим аспектом є також використання нумерації пакетів і часових міток для захисту від повторних атак, вибір конкретних алгоритмів визначається вимогами до системи, для важких платформ допустиме використання більш складних схем шифрування, тоді як для FPV-дронів перевага надається легковаговим алгоритмам або частковому захисту лише критичних команд (див. рис. 1.9) [12-13].

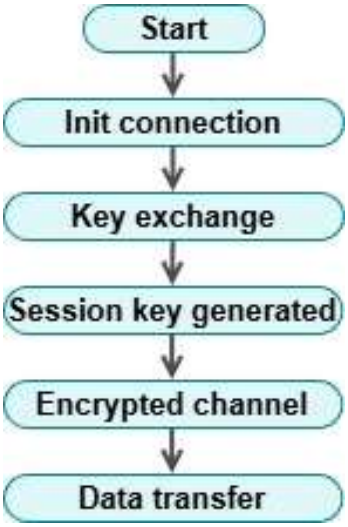


Рисунок 1.9 – Послідовність встановлення захищеного з'єднання

Сегментація мережевих потоків передбачає логічне або фізичне розділення різних типів даних, що передаються між компонентами системи, зокрема виділення окремих каналів для керування, телеметрії та відео, такий підхід дозволяє ізолювати критичні дані від менш важливих, зменшити навантаження на окремі канали та підвищити загальну стійкість системи, у рамках моделі Control Plane і Data Plane керуючі сигнали передаються окремо від інформаційних потоків, що дозволяє забезпечити їхню пріоритетність і мінімізувати затримки, сегментація також сприяє локалізації можливих атак, оскільки компрометація одного каналу не призводить до повного порушення функціонування системи, у поєднанні з механізмами пріоритезації трафіку та контролю якості обслуговування це дозволяє оптимізувати використання обмежених ресурсів каналу.

Криптографічні методи захисту інформації: баланс між стійкістю та швидкістю обробки. Криптографічний захист інформації у системах передавання даних безпілотних літальних апаратів розглядається як базовий механізм забезпечення конфіденційності, цілісності та автентичності інформаційних потоків, однак у середовищах реального часу він повинен бути максимально оптимізований з урахуванням обмежень обчислювальних ресурсів, пропускної здатності каналів і вимог до затримки передачі, що формує необхідність пошуку компромісу між криптостійкістю та швидкістю обробки даних, оскільки надмірно складні алгоритми можуть призводити до неприйнятного збільшення латентності та зниження ефективності керування апаратом, особливо у випадку швидкісних систем, де навіть незначне зростання затримки безпосередньо впливає на точність маневрування та стабільність польоту.

У загальному випадку криптографічні методи поділяються на:

— симетричні;

— асиметричні;

— гібридні.

Причому симетричні алгоритми шифрування, такі як AES (Advanced Encryption Standard), характеризуються високою швидкістю обробки та відносно низькими обчислювальними витратами, що робить їх придатними для безперервного

шифрування потокових даних, включаючи телеметрію та керуючі команди, тоді як асиметричні алгоритми, зокрема RSA або алгоритми на основі еліптичних кривих (ECC), забезпечують зручність організації безпечного обміну ключами та високий рівень криптографічної стійкості, але їх використання для безпосереднього шифрування великих обсягів даних є неефективним через значні обчислювальні витрати, що призводить до зростання затримок і перевантаження бортових процесорів [15; 17].

Поєднання цих підходів реалізується у вигляді гібридних криптографічних схем, де асиметричні алгоритми застосовуються на етапі встановлення захищеного з'єднання для генерації та обміну сесійними ключами, після чого передавання даних здійснюється із використанням симетричних алгоритмів, що дозволяє досягти оптимального балансу між безпекою та продуктивністю, при цьому у контексті безпілотних систем така схема є найбільш доцільною, оскільки початкові затрати на встановлення з'єднання компенсуються швидкою передачею даних у подальшому.

Важливим аспектом є врахування специфіки різних класів БПЛА, де для важких платформ з великою вантажопідйомністю та значними енергетичними ресурсами допустиме використання більш складних криптографічних алгоритмів із довшими ключами, наприклад AES-256 у режимах GCM або CBC, що забезпечує високий рівень захисту навіть при потенційному перехопленні трафіку, і водночас не створює критичних затримок у системі через меншу чутливість до латентності, тоді як для швидкісних FPV-дронів необхідно застосовувати більш легкі алгоритмічні рішення, такі як ChaCha20 або інші потокові шифри, які оптимізовані для швидкої обробки та мінімального навантаження на процесор, що дозволяє зберегти низький рівень затримки при передачі керуючих сигналів [14-16].

З урахуванням впливу криптографії на продуктивність системи слід відзначити, що додавання механізмів шифрування призводить до збільшення розміру передаваних пакетів за рахунок службових заголовків, ініціалізаційних векторів та кодів автентифікації, що у середньому може становити від 16 до 32 байтів на пакет, у результаті чого зростає навантаження на канал зв'язку та

зменшується ефективна пропускна здатність, а також виникає додаткова затримка, яка в окремих випадках може досягати 5–20% від загального часу передачі, що особливо критично для систем із вузькосмуговими каналами та високими вимогами до швидкодії [15].

Крім того, криптографічні операції впливають на енергоспоживання бортових систем, оскільки виконання алгоритмів шифрування та дешифрування потребує значних обчислювальних ресурсів, що може скорочувати тривалість автономної роботи апарата, особливо у випадку використання малопотужних мікроконтролерів або одноплатних комп'ютерів, у зв'язку з чим виникає необхідність оптимізації алгоритмів та використання апаратного прискорення, наприклад реалізації AES-NI або спеціалізованих криптографічних модулів, які дозволяють суттєво зменшити навантаження на центральний процесор [15].

Оптимізаційні підходи до застосування криптографії у безпілотних системах включають використання потокових шифрів, які забезпечують швидку обробку даних у реальному часі, адаптивне шифрування, при якому рівень захисту змінюється залежно від типу передаваних даних, а також вибір оптимальної довжини ключа, що дозволяє забезпечити достатній рівень безпеки без надлишкових обчислювальних витрат, при цьому важливо враховувати, що надмірне зменшення довжини ключа може призвести до зниження криптостійкості системи та підвищення ризику компрометації даних.

Окрему увагу приділяють забезпеченню цілісності та автентичності інформації, що реалізується за допомогою механізмів MAC (Message Authentication Code) або режимів шифрування з автентифікацією, таких як AES-GCM, які дозволяють не лише захистити дані від несанкціонованого доступу, але й виявляти спроби їх модифікації або підміни, що є критично важливим для захисту керуючих команд, оскільки навіть незначне втручання у такі дані може призвести до некоректної роботи системи або повної втрати контролю над апаратом.

У практичних реалізаціях систем захищеного керування БПЛА доцільно використовувати протоколи, що поєднують шифрування та автентифікацію, наприклад TLS із застосуванням сучасних наборів шифрів, таких як AES-GCM або

ChaCha20-Poly1305, що дозволяє забезпечити необхідний рівень захисту при збереженні прийнятної затримки передачі, при цьому для критичних каналів керування доцільно забезпечувати пріоритетність обробки та передачі, що дозволяє мінімізувати вплив криптографічних операцій на швидкість реакції системи [15-16].

Логічним продовженням застосування криптографічних методів є їх інтеграція з механізмами сегментації мережевих потоків, де керуючі команди, телеметрія та відеодані обробляються окремо з урахуванням їх пріоритетності та вимог до затримки, що дозволяє оптимізувати використання ресурсів та підвищити загальну ефективність системи, оскільки критичні дані отримують мінімальні затримки, тоді як менш важливі потоки можуть передаватися з використанням більш складних механізмів захисту.

Загалом криптографічні методи захисту інформації у безпілотних системах формують основу забезпечення безпеки передавання даних у реальному часі, а їх ефективність визначається правильним вибором алгоритмів, режимів роботи та параметрів налаштування з урахуванням специфіки конкретного класу апарата, що дозволяє досягти необхідного балансу між рівнем захищеності та продуктивністю системи без критичного впливу на її функціональні характеристики та здатність до виконання поставлених завдань.

1.4 Інфраструктура наземних станцій управління (GCS)

Інфраструктура наземних станцій управління або Ground Control Station формується як центральний вузол взаємодії між оператором та безпілотною платформою, у межах якого здійснюється (див. рис. 1.10):

— передача керуючих команд;

— прийом і обробка телеметричних даних;

— декодування;

— відображення відеопотоків;

— контроль стану всієї системи в реальному часі.

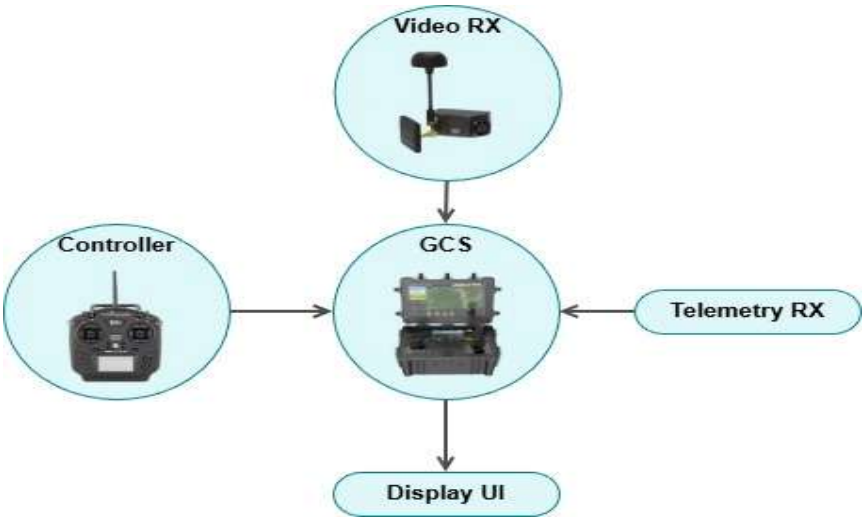


Рисунок 1.10 – Типова структура наземної станції керування GCS

При цьому функціональне навантаження таких станцій виходить за межі простого інтерфейсу керування і охоплює:

1. складні процеси маршрутизації трафіку;
2. інтеграції периферійного обладнання;
3. реалізації механізмів захисту каналів зв’язку;
4. забезпечення стабільності роботи (в умовах змінного радіоефіру).

Все це особливо актуально для сучасних комплексів, де різнорідні компоненти об’єднуються в єдину керуючу екосистему з високими вимогами до надійності та швидкодії.

Апаратно-програмні вимоги до GCS (на прикладі комплекту «UKROPTER»: ноутбук-трансформер, зовнішні монітори). Апаратно-програмна структура наземної станції визначає її здатність ефективно виконувати ці функції, оскільки саме від обчислювальних ресурсів, архітектури підключення пристроїв і якості програмного забезпечення залежить швидкість реакції оператора на зміну ситуації та точність передавання команд, у сучасних реалізаціях як базовий обчислювальний вузол застосовується ноутбук-трансформер або портативна

робоча станція, яка поєднує мобільність і достатню продуктивність для обробки потокових даних, включаючи відео високої роздільної здатності та телеметричні пакети, при цьому обов’язковими вимогами виступають наявність багатоядерного процесора, достатнього обсягу оперативної пам’яті на рівні не менше 8–16 ГБ, використання твердотільного накопичувача для швидкого доступу до даних і підтримка сучасних мережевих інтерфейсів, що забезпечують стабільне підключення до радіомодемів та інших комунікаційних пристроїв.

Доповненням до основного обчислювального вузла виступають зовнішні монітори, які дозволяють розділяти інформаційні потоки та одночасно відображати різні типи даних, зокрема телеметрію, карту місцевості та відеопотік, що підвищує ефективність сприйняття інформації оператором і зменшує ймовірність помилок під час керування, паралельно використовуються пульти керування радіоконтролерного типу, які забезпечують безпосереднє введення команд і формують сигнали управління, а також комутаційні модулі, до яких належать радіомодеми, антени, мережеві адаптери та інші засоби передачі даних, що разом формують фізичний рівень інфраструктури та забезпечують зв’язок із безпілотною платформою (див. рис. 1.11).

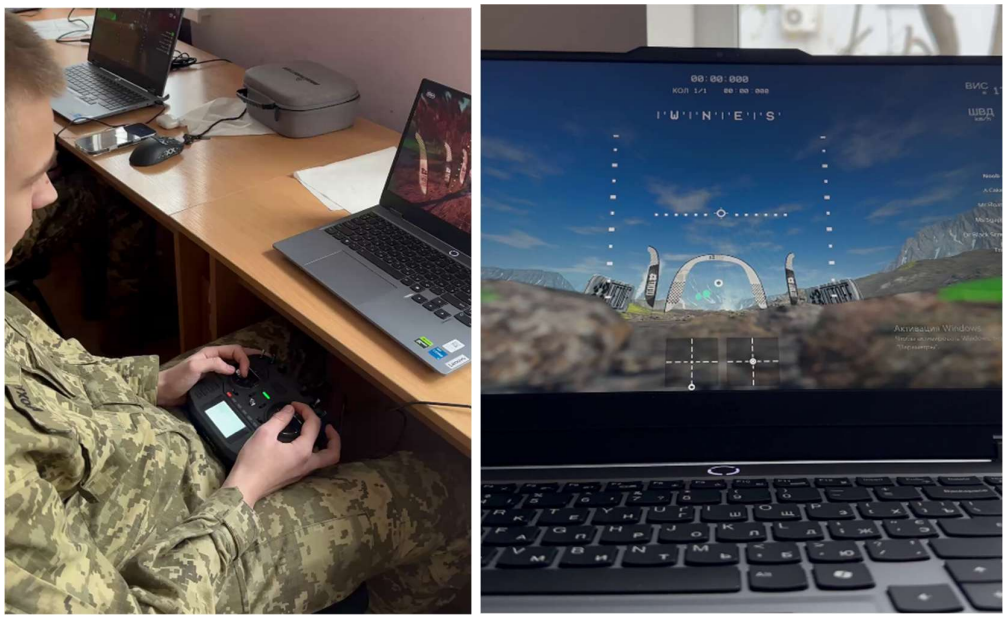


Рисунок 1.11 – Тренування керуванням БПЛА в симуляторі
зі пультом керування типу TX12 та ноутбуком-трансформером

Програмна складова наземної станції включає декілька взаємопов’язаних рівнів, серед яких виділяється:

— програмне забезпечення керування польотом;

— відображення параметрів польоту;

— формування та передачу команд;

— обробку отриманих даних.

При цьому мережеві служби виконують функції:

— маршрутизації трафіку;

— організації тунелювання;

— реалізації механізмів шифрування та контролю доступу.

Все це що дозволяє забезпечити захищене передавання інформації навіть у відкритому радіоефірі, додатково використовуються інструменти моніторингу та діагностики, які дозволяють аналізувати якість з’єднання, виявляти помилки передачі, фіксувати події у журналах та оцінювати стабільність функціонування системи в цілому.

З урахуванням експлуатації в польових умовах до наземної станції висуваються підвищені вимоги щодо надійності та автономності роботи, оскільки система повинна функціонувати безперервно в умовах обмеженого доступу до енергоресурсів, впливу зовнішніх факторів та можливих завад у каналах зв’язку, одночасно важливою є здатність до обробки даних у реальному часі, що забезпечує оперативне прийняття рішень, а також реалізація механізмів захисту інформації, які запобігають перехопленню або підміні керуючих сигналів, додатково враховується можливість масштабування системи шляхом підключення нових модулів і пристроїв, що дозволяє адаптувати інфраструктуру до різних сценаріїв застосування.

Особливості реалізації наземної станції залежать від класу безпілотного апарата, оскільки для важких багатофункціональних платформ характерна складна архітектура з підтримкою декількох каналів зв’язку, одночасною обробкою телеметрії та відеопотоків, а також використанням розширених засобів аналізу даних, що потребує більш потужного апаратного забезпечення та складнішого

програмного стеку, тоді як для швидкісних FPV-систем притаманна спрощена конфігурація з акцентом на мінімізацію затримки та забезпечення максимально швидкої реакції оператора, де пріоритет надається швидкодії та стабільності відеосигналу, а не глибокій обробці даних.

Маршрутизація даних у межах наземної станції передбачає розподіл інформаційних потоків між різними компонентами системи, де керуючі команди, телеметрія та відео передаються окремими каналами або логічно розділяються за допомогою механізмів пріоритезації, що дозволяє уникнути перевантаження каналу та забезпечити стабільну передачу критично важливих даних, у цьому контексті використання принципів Quality of Service дозволяє встановлювати різні рівні пріоритету для пакетів, що забезпечує першочергову доставку команд управління та мінімізує вплив затримок на керованість апарата.

Додатковим елементом інфраструктури є системи моніторингу якості каналу зв’язку, які дозволяють оцінювати такі параметри, як рівень сигналу, затримка передачі, втрати пакетів та стабільність з’єднання, на основі яких формується уявлення про поточний стан мережі та можливі загрози її функціонуванню, аналіз цих показників дає змогу своєчасно виявляти деградацію каналу, адаптувати параметри передачі даних або переходити до альтернативних режимів роботи, включаючи використання резервних каналів або автономних алгоритмів керування.

Маршрутизація даних між оператором та периферійним обладнанням (окуляри Skyzone, пульти TX12). Маршрутизація даних у межах наземної станції управління формується як процес організації передавання інформації між оператором, обчислювальним ядром системи та периферійними пристроями, де кожен тип даних має власні вимоги до затримки, пропускної здатності та надійності, у зв’язку з чим виникає необхідність чіткого розподілу потоків і визначення правил їх обробки, при цьому в типовій архітектурі безпілотних систем виділяються керуючі сигнали, телеметричні дані, відеопотоки та службова інформація, які циркулюють між компонентами системи та потребують узгодженої маршрутизації з урахуванням їх пріоритетності та критичності.

У структурі периферійного обладнання особливе місце займають пристрої відображення та керування, зокрема відеоокуляри типу Skyzone, які призначені для прийому відеосигналу та його відображення у режимі реального часу з мінімальною затримкою, що є критично важливим для забезпечення коректного сприйняття ситуації оператором, а також радіоконтролери типу TX12, які виконують функцію генерації керуючих сигналів і передачі їх у систему через відповідні радіомодулі, взаємодіючи із наземною станцією як джерело команд управління.

У загальній схемі маршрутизації інформаційні потоки проходять послідовний шлях від оператора до безпілотної платформи та назад, де оператор формує команди за допомогою пульта керування, після чого сигнали надходять до наземної станції, яка виконує їх обробку, розподіл і подальшу передачу через радіоканал до БПЛА, одночасно з цим відеопотік може спрямовуватись безпосередньо на пристрої відображення або проходити через обчислювальний вузол для додаткової обробки, тоді як телеметричні дані передаються у зворотному напрямку та відображаються на моніторах або інтерфейсі оператора, що формує багатоканальну структуру обміну інформацією (див. рис. 1.12).

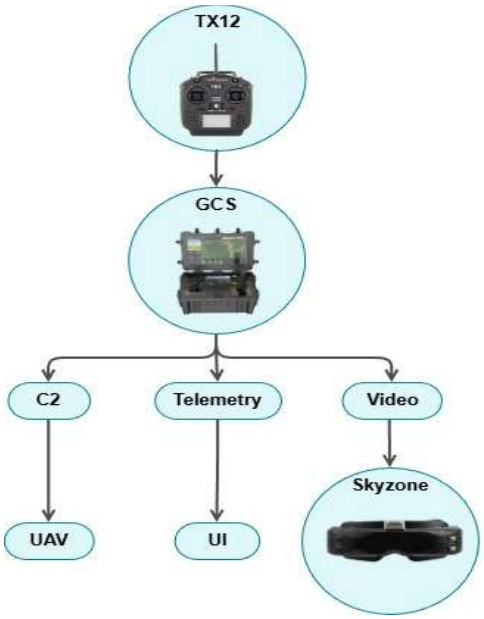


Рисунок 1.12 – Маршрутизація даних між ноутбуком-трансформером, окулярами Skyzone та пультом TX12

Ефективна маршрутизація передбачає використання логічного розділення потоків із застосуванням різних портів, протоколів і механізмів пріоритезації, що дозволяє уникнути конкуренції за ресурси каналу зв’язку та забезпечити стабільну передачу критично важливих даних, у цьому контексті застосування підходів Quality of Service дозволяє встановлювати високий пріоритет для керуючих пакетів, середній для телеметрії та нижчий для відеоданих, що сприяє зменшенню впливу затримок на процес керування та підвищує загальну стабільність системи.

Особливості маршрутизації значною мірою залежать від класу безпілотної системи, оскільки у швидкісних FPV-дронах відеопотік часто передається безпосередньо до окулярів оператора, минаючи складні етапи обробки в наземній станції, що дозволяє досягти мінімальної затримки та забезпечити швидку реакцію, при цьому маршрутизація в таких системах максимально спрощена і орієнтована на прямий зв’язок між передавачем і приймачем, тоді як у складніших багатофункціональних платформах усі інформаційні потоки можуть проходити через наземну станцію, де виконується їх фільтрація, обробка та розподіл між різними пристроями, що потребує застосування більш складних механізмів сегментації та управління трафіком.

У процесі експлуатації можуть виникати різноманітні проблеми маршрутизації, серед яких виділяються:

— конфлікти портів;
— перевантаження каналів;
— збільшення затримок;
— втрати пакетів.

Все це негативно впливає на якість зв’язку та керованість апарата, у зв’язку з цим виникає необхідність застосування методів оптимізації, до яких належать:

1. пріоритезація керуючих пакетів;
2. буферизація відеоданих;
3. використання окремих каналів (для різних типів інформації);
4. впровадження адаптивних алгоритмів маршрутизації.

Для останніх вкрай важливо змінювати параметри передачі залежно від поточного стану мережі.

Системи моніторингу якості каналу та діагностики стану зв'язку. Паралельно з маршрутизацією функціонують системи моніторингу якості каналу зв’язку, які забезпечують безперервний контроль параметрів передачі даних та дозволяють оцінювати стан мережі у реальному часі, до основних показників належать:

— затримка передачі;

— втрати пакетів;

— рівень сигналу;

— ефективна пропускна здатність.

Де, в свою чергу, аналіз яких дає можливість виявляти деградацію каналу, оцінювати вплив зовнішніх завад і своєчасно реагувати на зміну умов зв’язку.

Інструменти моніторингу можуть бути інтегровані безпосередньо у програмне забезпечення наземної станції, забезпечуючи відображення телеметричних параметрів, статусу з’єднання та попереджень про можливі проблеми, або реалізовуватись у вигляді спеціалізованих мережевих аналізаторів, що дозволяють здійснювати глибокий аналіз пакетів, виявляти аномалії та досліджувати поведінку трафіку, додатково використовується логування подій, яке забезпечує накопичення інформації для подальшого аналізу після завершення місії та дозволяє оцінити ефективність роботи системи.

Функціональне призначення систем моніторингу охоплює:

1. виявлення проблем у реальному часі;
2. прогнозування можливих відмов;
3. підтримку процесу прийняття рішень оператором;
4. забезпечення загального рівня безпеки.

Оскільки аналіз параметрів каналу дозволяє виявляти ознаки стороннього втручання, зокрема спроби підміни сигналів або радіоелектронного подавлення, що робить моніторинг невід’ємною складовою захищених систем керування.

Діагностика стану каналу передбачає комплексний аналіз параметрів з’єднання, включаючи:

— перевірку доступності вузлів;

— оцінку затримок;

— визначення джерел перешкод;

— локалізацію проблемних ділянок.

Все це дозволяє оперативно реагувати на зміни та забезпечувати стабільність роботи системи навіть у складних умовах експлуатації.

Рівень складності систем моніторингу також залежить від класу безпілотної платформи, оскільки для багатофункціональних апаратів застосовуються розширені засоби аналізу з підтримкою обробки великих обсягів телеметричних і відеоданих, тоді як у швидкісних FPV-системах використовується більш спрощений підхід, орієнтований на візуальний контроль та швидке реагування без глибокої аналітики.

Інтеграція механізмів маршрутизації та моніторингу у складі наземної станції формує єдину систему управління інформаційними потоками, що забезпечує ефективну взаємодію між оператором, безпілотною платформою та периферійними пристроями, при цьому правильна організація цих процесів дозволяє підвищити надійність керування, зменшити вплив зовнішніх факторів та забезпечити необхідний рівень захищеності системи в умовах реального часу.

1.5 Нормативна база та стандарти безпеки в галузі БпАК

Розвиток безпілотних авіаційних комплексів супроводжується поступовим формуванням нормативної бази, яка визначає вимоги до безпеки, надійності та захищеності каналів зв’язку, оскільки сучасні системи застосовуються не лише у цивільних сценаріях, а й у критичних та спеціалізованих середовищах, де порушення роботи або компрометація інформації може призводити до значних

технічних і операційних наслідків, при цьому регуляторне поле охоплює кілька рівнів, серед яких міжнародні стандарти, галузеві рекомендації, національні нормативні документи та внутрішні технічні політики розробників, що у сукупності формують комплексний підхід до забезпечення безпеки функціонування БпАК, зокрема в частині мережевої взаємодії, де визначальними стають вимоги до надійності радіоканалів, криптографічного захисту та забезпечення цілісності й автентичності переданих даних, що безпосередньо впливає на архітектурні рішення, вибір протоколів та реалізацію механізмів захисту.

Стандарти надійності критичних радіоканалів. У контексті стандартів надійності критичних радіоканалів канал керування, який позначається як C2, розглядається як центральний елемент системи, через який передаються команди управління польотом, тому до нього висуваються підвищені вимоги щодо стабільності функціонування, допустимих затримок і стійкості до зовнішніх впливів, при цьому ключовими характеристиками виступають доступність каналу у визначений проміжок часу, затримка передачі сигналу, рівень втрат пакетів, здатність протистояти завадам та забезпечення безперервності зв’язку, у системах реального часу ці параметри мають жорсткі обмеження, що формує необхідність застосування спеціалізованих підходів до їх забезпечення, у міжнародній практиці регулювання таких аспектів спирається на рекомендації телекомунікаційних організацій, стандарти якості обслуговування та принципи управління радіочастотним ресурсом, які передбачають використання резервування каналів, розподіл по різних частотних діапазонах і механізми автоматичного перемикання у разі деградації основного каналу.

Підвищення надійності радіоканалів досягається за рахунок впровадження технічних механізмів, серед яких важливе місце займає частотне перестрибування, що дозволяє змінювати робочу частоту передачі сигналу у динамічному режимі та зменшувати вплив перешкод і спроб перехоплення, доповненням до цього виступає адаптивна модуляція, яка змінює параметри передачі залежно від поточного стану каналу, забезпечуючи баланс між швидкістю та стійкістю, також використовується

випереджувальна корекція помилок, яка дозволяє відновлювати частину втрачених даних без необхідності повторної передачі, що особливо важливо для систем із обмеженою пропускною здатністю, додатково впроваджуються резервні канали, які можуть базуватися на альтернативних технологіях зв’язку, що забезпечує підвищену стійкість до відмов, при цьому для різних класів безпілотних систем ці підходи застосовуються з різною інтенсивністю, оскільки важкі платформи можуть дозволити багатоканальність і складні алгоритми обробки, тоді як легкі системи обмежені у ресурсах і змушені мінімізувати накладні витрати.

У сфері сертифікації та використання засобів шифрування нормативна база визначає:

— вимоги до алгоритмів;

— процедур управління ключами;

— способів інтеграції криптографічних механізмів.

Оскільки захист інформації передбачає не лише конфіденційність, а й гарантії цілісності та автентичності, у міжнародній практиці застосовуються стандартизовані алгоритми шифрування, які проходять перевірку на стійкість до відомих типів атак, а також визначаються вимоги до довжини ключів і режимів роботи, важливим аспектом виступає процедура обміну ключами, яка повинна забезпечувати захищене встановлення з’єднання навіть у відкритому середовищі, у цьому контексті поширеним підходом є використання гібридних схем, де асиметричні алгоритми застосовуються для початкової автентифікації та обміну ключами, після чого передача даних здійснюється із використанням швидших симетричних алгоритмів.

Сертифікація криптографічних засобів передбачає перевірку їх відповідності встановленим вимогам, що включає:

— аналіз стійкості алгоритмів;

— коректність реалізації;

— здатність функціонувати у заданих умовах експлуатації.

Все це, у практичних системах БпАК, означає необхідність використання перевірених бібліотек та протоколів, а також забезпечення захищеного зберігання

ключів і регулярного їх оновлення, додатково враховується обчислювальне навантаження, яке створюють криптографічні операції, оскільки надмірна складність алгоритмів може призводити до збільшення затримок та зниження ефективності системи в цілому, тому вибір засобів шифрування здійснюється з урахуванням балансу між рівнем безпеки та продуктивністю.

Окремим напрямом нормативного регулювання виступають практики забезпечення цілісності команд управління, які спрямовані на запобігання несанкціонованим змінам або підміні переданих даних, оскільки навіть за наявності шифрування можливі атаки, спрямовані на повторну передачу або модифікацію пакетів, тому застосовуються механізми контролю цілісності, що базуються на використанні криптографічних хеш-функцій і кодів автентифікації повідомлень, які дозволяють перевіряти відповідність отриманих даних очікуваним значенням, додатково впроваджується нумерація пакетів або використання часових міток, що забезпечує захист від повторних атак, а також дозволяє визначати порядок передачі даних.

У практичній реалізації систем управління безпілотними апаратами механізми контролю цілісності інтегруються безпосередньо у протоколи обміну даними, що забезпечує перевірку кожного отриманого пакета перед його обробкою, у разі виявлення невідповідності пакет відкидається, а подія може фіксуватися у журналі безпеки для подальшого аналізу, такий підхід дозволяє локалізувати потенційні атаки та запобігти їх впливу на роботу системи, при цьому ефективність контролю залежить від правильного налаштування параметрів перевірки та узгодженості між усіма компонентами системи.

З урахуванням різноманіття класів безпілотних систем підходи до забезпечення цілісності можуть відрізнятися, оскільки для складних платформ із багатоканальною архітектурою можливе застосування повноцінних механізмів криптографічного захисту із перевіркою кожного етапу передачі, тоді як для легких систем із жорсткими обмеженнями по затримці допускається використання спрощених схем, які забезпечують базовий рівень захисту без значного впливу на швидкодію, при цьому навіть у спрощених реалізаціях важливим залишається

дотримання принципу перевірки достовірності джерела сигналу та контролю послідовності пакетів.

Вимоги до сертифікації та використання засобів шифрування. Криптографічний захист у сучасних безпілотних авіаційних комплексах формується як невід'ємна складова архітектури безпеки, що регламентується як міжнародними, так і національними стандартами, при цьому основні вимоги до засобів шифрування визначають необхідність використання перевірених алгоритмів, достатньої довжини криптографічних ключів, стійкості до актуальних типів атак, а також забезпечення безпечного зберігання ключового матеріалу і контролю доступу до нього, у такому контексті криптографія розглядається не лише як інструмент забезпечення конфіденційності, а і як комплексний механізм, що охоплює автентифікацію сторін взаємодії, перевірку цілісності переданих даних і захист від несанкціонованого втручання у процес керування.

Вимоги до сертифікації та використання засобів шифрування. У міжнародній практиці сформовано набір стандартів, які визначають вимоги до реалізації криптографічних засобів, серед яких виділяються:

1. підходи до управління інформаційною безпекою;
2. вимоги до криптографічних модулів;
3. рекомендації щодо вибору алгоритмів шифрування.

По суті ці документи описують процедури тестування, рівні довіри до реалізацій та критерії оцінювання стійкості, що дозволяє уніфікувати підхід до побудови захищених систем, одночасно національні регуляторні механізми встановлюють додаткові обмеження і вимоги, які передбачають застосування сертифікованих засобів, відповідність державним стандартам та контроль використання криптографії у системах із підвищеними вимогами до безпеки, у результаті формується багаторівнева модель регулювання, що поєднує глобальні рекомендації і локальні нормативні обмеження.

Практичне застосування криптографічних засобів у системах БпАК передбачає використання протоколів захищеної передачі даних, сертифікатів для автентифікації та симетричних алгоритмів для ефективного шифрування

інформації, при цьому важливим аспектом виступає правильна інтеграція цих механізмів у існуючу архітектуру зв’язку, що включає інкапсуляцію службових і керуючих даних, організацію тунелів передачі та розмежування доступу до різних типів трафіку, одночасно необхідно враховувати обмеження обчислювальних ресурсів бортового обладнання і вимоги до мінімальної затримки, що характерні для систем реального часу, у зв’язку з чим вибір алгоритмів і параметрів шифрування здійснюється з урахуванням компромісу між рівнем захисту та продуктивністю.

У різних класах безпілотних систем цей баланс реалізується по-різному, оскільки для важких платформ із розвиненою обчислювальною базою допускається повноцінне шифрування усіх каналів зв’язку із застосуванням складних алгоритмів і повним циклом криптографічної обробки, при цьому збільшення затримки залишається у допустимих межах і не впливає критично на виконання завдань, тоді як для швидкісних систем із жорсткими вимогами до часу реакції доцільним стає часткове шифрування або використання полегшених алгоритмів, що дозволяє мінімізувати накладні витрати і зберегти необхідну швидкодію, така диференціація підходів відповідає загальній логіці адаптивного застосування криптографії залежно від умов експлуатації і характеристик конкретної платформи.

У межах реалізації захищених каналів передачі даних доцільним є використання тунелювання на базі стандартних протоколів, що забезпечують шифрування, автентифікацію і контроль цілісності, при цьому трафік телеметрії та керування може інкапсулюватися у захищений контейнер, який передається через відкриті мережі, додатково застосовується ізоляція каналів, що дозволяє розділити критичні та допоміжні потоки даних і обмежити можливість їх перехресного впливу, у сукупності такі підходи забезпечують відповідність сучасним вимогам безпеки та спрощують інтеграцію захищених механізмів у існуючі системи.

Питання забезпечення цілісності команд управління розглядається як один із критичних аспектів функціонування безпілотних систем, оскільки порушення цілісності переданих даних може призвести до некоректного виконання команд, втрати керованості або зміни поведінки апарата, у цьому контексті основними

загрозами виступають модифікація пакетів, повторна передача раніше перехоплених даних і підміна легітимних команд, що потребує впровадження механізмів перевірки достовірності інформації на кожному етапі передачі.

Серед основних методів забезпечення цілісності використовуються коди автентифікації повідомлень, які дозволяють перевіряти незмінність даних і підтверджувати їх походження, цифрові підписи, що забезпечують як цілісність, так і автентичність, нумерація пакетів або використання часових міток для запобігання повторним атакам, а також контрольні суми як базовий рівень перевірки, у сучасних протоколах обміну даними частина цих механізмів може бути інтегрована на рівні самого протоколу, що дозволяє виконувати перевірку структури пакетів і достовірності повідомлень ще до їх обробки системою керування, однак у більшості випадків таких засобів недостатньо без додаткового криптографічного захисту.

Алгоритм обробки отриманих даних передбачає послідовну перевірку їх достовірності, при якій після отримання пакета виконується контроль коду автентифікації, у разі невідповідності пакет відкидається без подальшої обробки, у випадку успішної перевірки здійснюється аналіз порядкового номера або часової мітки, що дозволяє виявити повторно передані або застарілі дані, після проходження цих етапів пакет може бути дешифрований і переданий до системи керування для виконання, така послідовність забезпечує багаторівневий захист і мінімізує ймовірність успішної атаки.

Ефективне забезпечення цілісності команд управління потребує комплексного підходу, що включає поєднання криптографічних механізмів, сегментації каналів, контролю доступу та постійного моніторингу стану мережі, при цьому важливим аспектом залишається здатність системи виявляти аномалії у трафіку і своєчасно реагувати на них, що дозволяє підвищити стійкість до атак і забезпечити стабільне функціонування у складних умовах експлуатації.

2. ПРАКТИЧНА РОЗРОБКА ЕМУЛЬОВАНОГО СЕРЕДОВИЩА ЗАХИЩЕНОГО КЕРУВАННЯ

2.1 Проєктування архітектури тестового стенду

Практична реалізація емульованого середовища захищеного керування орієнтована на відтворення реалістичної поведінки системи безпілотного літального апарата без використання фізичного носія, оскільки в умовах обмеженого доступу до апаратних ресурсів саме програмна емуляція дозволяє отримати повноцінний дослідницький результат із високим рівнем достовірності. Основою такого підходу виступає:

— застосування програмних засобів емуляції;

— стандартних мережевих інструментів дистрибутивів Linux;

— відкритих протоколів передачі даних.

Все це разом формує контрольоване середовище, у межах якого можливо відтворити повний цикл взаємодії між оператором та бортовою системою, починаючи від формування керуючих команд і завершуючи отриманням телеметричної інформації.

Передача даних у такій системі здійснюється через захищений канал, що дозволяє досліджувати вплив криптографічних механізмів на продуктивність, затримки та загальну надійність функціонування, при цьому ключовою концепцією є використання підходу Software In The Loop, який забезпечує емуляцію роботи автопілота та генерацію реального мережевого трафіку, ідентичного тому, що формується у фізичних системах. Взаємодія між компонентами організовується через мережеві інтерфейси з використанням протоколів TCP та UDP, механізмів тунелювання та контролю доступу, що дозволяє досліджувати поведінку системи на різних рівнях мережевої моделі.

Архітектура тестового стенду формується за принципом функціональної декомпозиції, що передбачає розділення системи на незалежні логічні компоненти, кожен із яких виконує окрему функцію, пов’язану з генерацією, передачею або обробкою даних, завдяки чому забезпечується можливість масштабування, ізоляції помилок та гнучкої зміни параметрів експерименту без впливу на інші частини системи, а також реалізується можливість незалежного тестування окремих модулів.

У структурі тестового стенду виділяються три основні рівні, які утворюють цілісну систему взаємодії. Перший рівень — операторський, що включає наземну станцію керування (GCS), реалізовану на базі програмного забезпечення Mission Planner, яка відповідає за формування керуючих команд, відображення телеметрії та моніторинг стану системи, при цьому GCS функціонує як клієнт, який ініціює мережеве з’єднання з бортовою частиною. Другий рівень — мережевий, що забезпечує транспортування даних між вузлами системи та включає механізми тунелювання WireGuard, маршрутизацію, контроль доступу за допомогою Firewall та трансляцію мережевих адрес (NAT), причому саме на цьому рівні реалізується криптографічний захист трафіку та обмеження несанкціонованого доступу. Третій рівень — бортовий, представлений емулятором автопілота або мережевим сервісом, який приймає керуючі команди, обробляє їх та формує відповідну телеметрію, у ролі такого вузла може виступати як повноцінний SITL-емулятор, так і TCP-сервер, що імітує роботу телеметричного порту (див. рис. 2.1).

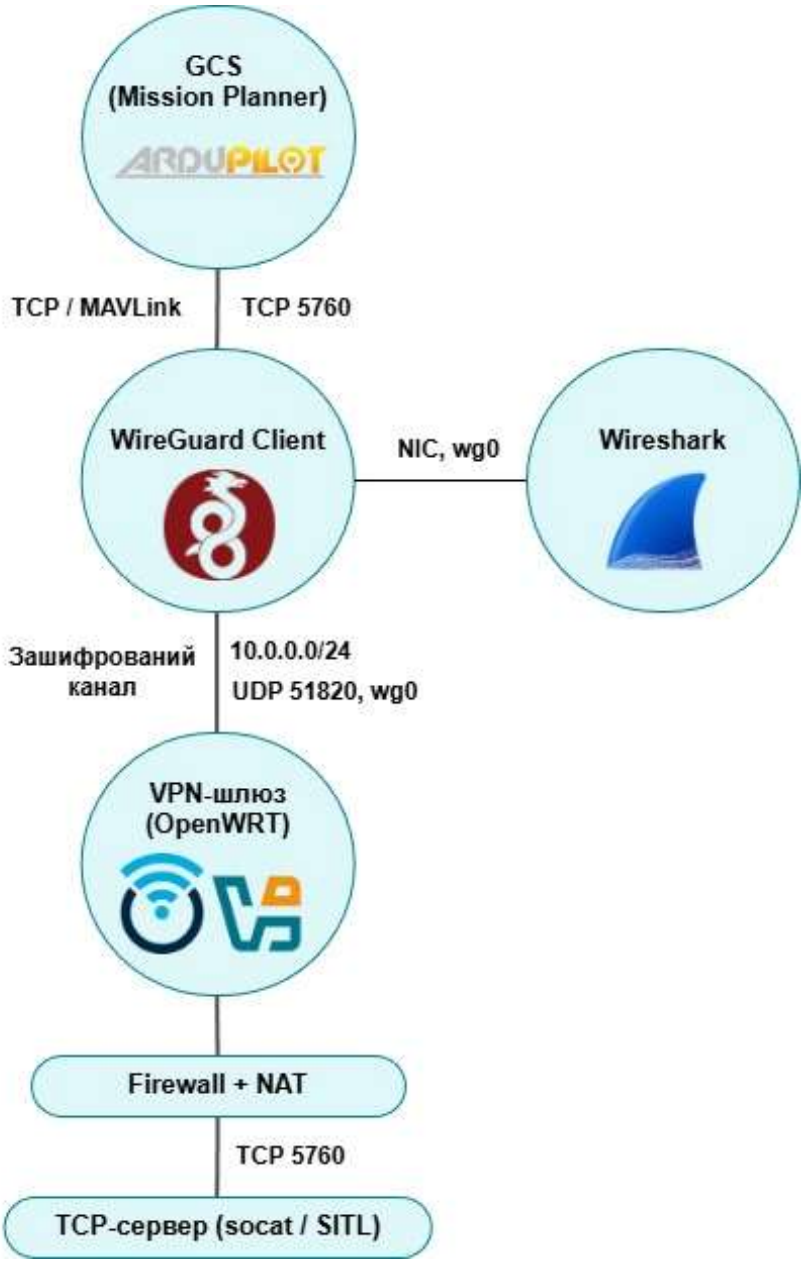


Рисунок 2.1 – Архітектура апаратно-програмного комплексу захищеного керування БПЛА

Загальна архітектура системи реалізується у вигляді послідовного ланцюга взаємодії компонентів, де наземна станція керування формує потік даних і передає його через TCP-з’єднання, використовуючи протокол MAVLink, після чого трафік інкапсулюється у VPN-клієнті WireGuard, який створює зашифрований канал передачі, далі зашифровані пакети надходять до проміжного вузла, реалізованого у вигляді віртуальної машини на базі OpenWrt 23.05.5, де виконується функція VPN-

шлюзу з налаштованими правилами Firewall та механізмами NAT, після чого трафік передається до бортового вузла, також реалізованого у вигляді окремої віртуальної машини, де працює утиліта socat або емулятор SITL (див. табл. 2.1) [18; 22].

Таблиця 2.1 – Склад апаратно-програмного тестового стенду

Компонент	Призначення	ОС	Інтерфейси	Примітка
GCS (Mission Planner)	Передавання команд та прийом телеметрії	Windows	TCP 5760	Клієнтська станція оператора
OpenWRT	VPN-шлюз, NAT, Firewall	OpenWrt 23.05.5	UDP 51820, wg0	Центральний вузол безпеки
WireGuard Client	Тунелювання трафіку	Windows/Linux	wg0	Шифрування каналу
TCP Server (socat)	Емуляція бортового порту	OpenWrt 23.05.5	TCP 5760	Імітація автопілота
Wireshark	Аналіз пакетів	Windows/Linux	NIC, wg0	Контроль трафіку
ArduPilot SITL	Повна емуляція автопілота	Linux	TCP/UDP	Альтернатива socat

Логіка взаємодії між компонентами відповідає клієнт-серверній моделі, у межах якої наземна станція ініціює TCP-з’єднання, після чого трафік інкапсулюється у VPN-тунель, передається через захищений канал, приймається на стороні бортового вузла та обробляється відповідно до заданої логіки, при цьому можливе як формування відповіді, так і імітація відсутності реакції для дослідження поведінки прикладного рівня.

Така організація системи дозволяє досягти високого рівня ізоляції між прикладним та мережевим рівнями, забезпечити контрольоване середовище для дослідження криптографічного захисту, а також створити гнучку платформу для подальшого аналізу продуктивності, безпеки та стійкості систем дистанційного керування безпілотними апаратами.

Апаратна та мережева схема взаємодії. Для дослідження процесів передавання даних та механізмів криптографічного захисту використовується

архітектура типу «клієнт — захищений шлюз — сервер», у межах якої всі мережеві взаємодії проходять через проміжний VPN-вузол на базі OpenWrt 23.05.5, що виконує функції маршрутизації, контролю доступу та обробки трафіку. Така організація дозволяє ізолювати прикладний рівень від транспортного та дослідити вплив шифрування без втручання у логіку роботи наземної станції керування.

Мережева взаємодія формується як послідовний ланцюг передачі даних, у якому початковою точкою виступає наземна станція керування Mission Planner, що функціонує як клієнт і ініціює TCP-з'єднання на стандартний телеметричний порт 5760. Даний порт використовується як типовий інтерфейс обміну даними для протоколу MAVLink, що забезпечує впорядковану передачу пакетів, контроль доставки та збереження їхньої послідовності, завдяки чому формується стабільний потік керуючих команд і телеметрії. Сформований TCP-трафік не передається напряму, а інкапсулюється у віртуальний мережевий інтерфейс VPN-клієнта на базі WireGuard, де відбувається автоматичне шифрування та автентифікація даних із використанням UDP-порту 51820.

Передача зашифрованого трафіку здійснюється через мережу до проміжного вузла — Linux VPN-шлюзу, реалізованого у вигляді віртуальної машини з дистрибутивом OpenWrt 23.05.5, де розгорнуто механізми Firewall та NAT. На цьому етапі виконується:

— дешифрування пакетів;

— перевірка на відповідність правилам доступу;

— журналювання мережевої активності.

Наявність брандмауера дозволяє реалізувати політику фільтрації, відповідно до якої кожен пакет проходить перевірку перед подальшою обробкою, унаслідок чого можливе як відхилення небажаного трафіку, так і пропуск дозволених з'єднань до наступного рівня.

Після проходження через шлюз трафік спрямовується до TCP-сервера, реалізованого за допомогою утиліти socat, яка виконує роль імітатора бортового телеметричного порту. Даний сервер відкриває порт 5760, приймає вхідні з'єднання та обробляє отримані дані на базовому рівні, передаючи їх у консоль або

відкидаючи, що дозволяє моделювати поведінку бортового контролера без необхідності використання повноцінного автопілота. У деяких сценаріях замість такого імітатора може застосовуватись середовище ArduPilot SITL, однак у даній конфігурації використовується спрощений підхід із метою фокусування на мережевих аспектах (див. рис. 2.2).

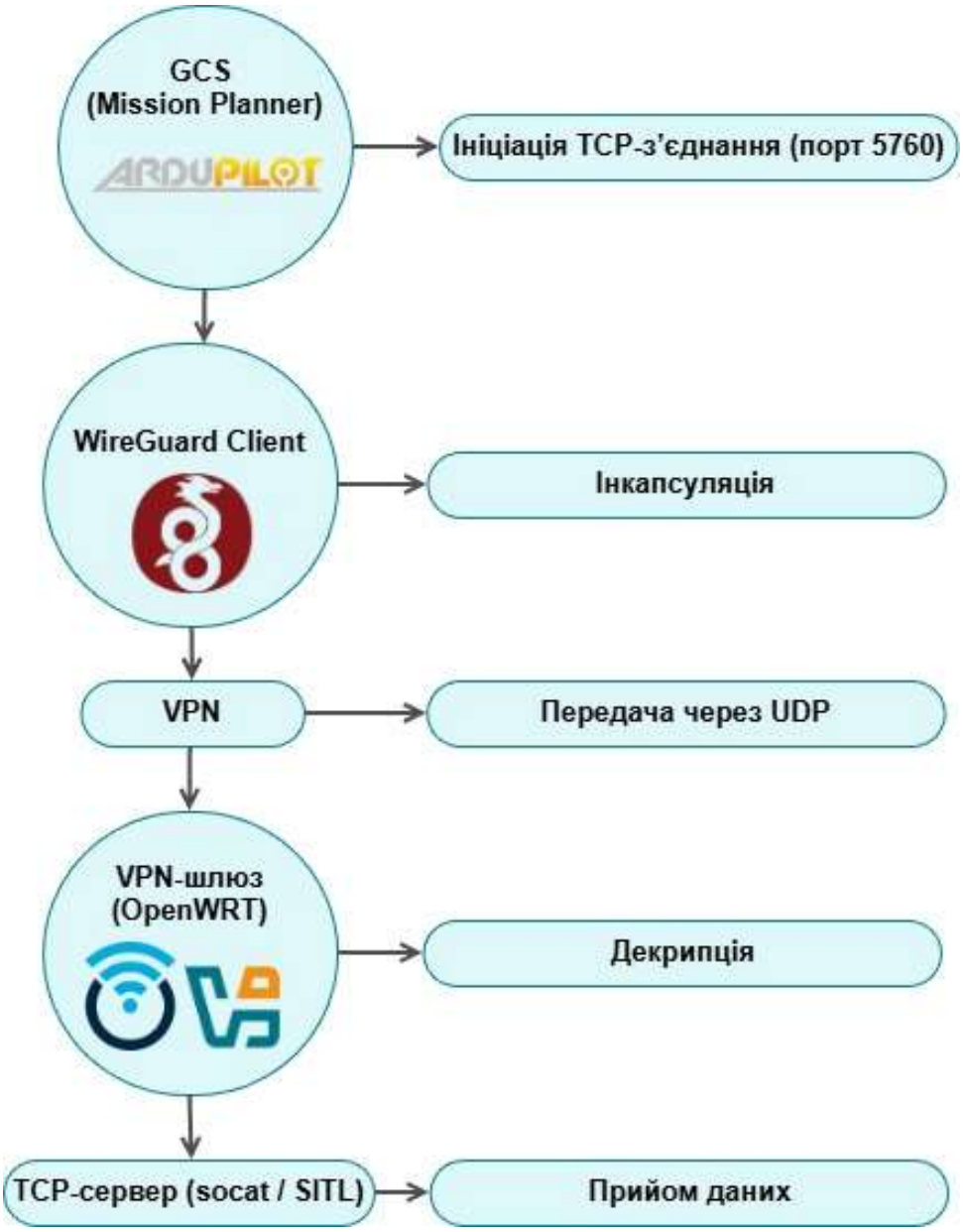


Рисунок 2.2 – Апаратна та мережева схема взаємодії компонентів стенду

Розгортання VPN та TCP-сервера в межах однієї віртуальної машини дозволяє спростити тестове середовище без втрати функціональності, оскільки обидва компоненти працюють як логічно відокремлені модулі, що взаємодіють через мережевий стек операційної системи. Такий підхід зменшує складність інфраструктури, водночас зберігаючи можливість дослідження процесів маршрутизації, шифрування та фільтрації.

У розгалуженій логіці ключовим етапом є перевірка правил доступу на VPN-шлюзі, де пакет може бути або прийнятий і переданий далі, або відхилений у разі невідповідності політикам безпеки. Після успішного проходження цього етапу здійснюється дешифрування, і лише у випадку відсутності помилок цілісності дані передаються до прикладного рівня.

Характерною особливістю функціонування системи є відсутність сигналу Heartbeat під час встановлення з’єднання між Mission Planner і бортовим вузлом, що пояснюється використанням утиліти socat, яка не реалізує повноцінний протокол MAVLink і не генерує службові пакети. У результаті наземна станція ініціює з’єднання, передає дані, однак не отримує підтвердження активності автопілота, що призводить до виникнення тайм-ауту та повідомлення про відсутність Heartbeat. Така поведінка не є помилкою мережевої архітектури, а відображає обмеження імітаційного середовища, водночас дозволяючи досліджувати транспортний рівень без впливу прикладної логіки.

Для контролю мережевих процесів та аналізу структури трафіку застосовується Wireshark, який забезпечує можливість перехоплення пакетів, ідентифікації TCP-з’єднань, аналізу їхнього вмісту та перевірки наявності службових повідомлень (див. рис. 2.3).



Рисунок 2.3 – Схема руху телеметричного трафіку між вузлами

Використання такого інструменту дозволяє верифікувати факт інкапсуляції трафіку, оцінити його структуру та підтвердити відсутність відкритих даних у зашифрованому каналі.

2.2 Побудова апаратно-програмного комплексу

Побудова апаратно-програмного комплексу емульованого середовища захищеного керування базується на використанні реального програмного стеку, що забезпечує генерацію, передачу та аналіз мережевого трафіку у форматі, максимально наближеному до реальних умов функціонування безпілотних систем, унаслідок чого створюється можливість формування повноцінних мережевих

пакетів, оцінювання впливу криптографічного захисту та дослідження поведінки системи при зміні параметрів каналу зв'язку. Архітектурно комплекс реалізовано як взаємодію трьох ключових компонентів:

1. наземна станція керування виступає джерелом команд;
2. VPN-шлюз забезпечує захист і транспортування;
3. бортовий емулятор виконує функцію приймача.

Все це дозволяє відтворити логіку клієнт-серверної моделі без необхідності використання фізичного обладнання.

Загальна структура комплексу формується як послідовний ланцюг передачі даних, у якому наземна станція керування Mission Planner ініціює TCP-з'єднання на порт 5760, що відповідає стандартному інтерфейсу телеметрії, після чого сформований трафік передається до клієнта WireGuard, де відбувається його інкапсуляція та шифрування з подальшою передачею через UDP-порт 51820. Отриманий зашифрований потік приймається VPN-шлюзом, реалізованим на базі OpenWrt 23.05.5, де виконуються функції фільтрації пакетів, трансляції адрес та контролю доступу, після чого дані передаються до TCP-сервера, організованого за допомогою утиліти socat, що забезпечує прийом з'єднань і базову обробку трафіку [18; 22; 29].

Принцип функціонування комплексу визначається послідовністю етапів, у межах яких наземна станція ініціює TCP-з'єднання, сформований трафік інкапсулюється у VPN-тунель, передається захищеним каналом, проходить процедуру дешифрування на шлюзі, перевіряється за правилами брандмауера та передається до TCP-сервера, де або обробляється, або відкидається залежно від сценарію експерименту. Така організація дозволяє реалізувати контрольований процес маршрутизації пакетів, у якому кожен етап може бути проаналізований окремо, включаючи перевірку факту проходження трафіку через VPN, відповідність портів, а також реакцію системи на відсутність або наявність вхідних даних.

Алгоритмічна модель роботи комплексу має послідовний характер і може бути інтерпретована як процес, у якому спочатку відбувається ініціація з'єднання,

далі формується TCP-пакет, виконується його інкапсуляція у VPN, здійснюється передача через UDP-канал, після чого на стороні шлюзу виконується дешифрування, перевірка правил доступу та передача до кінцевого вузла. У розгалуженій формі така модель передбачає перевірку джерела пакета, визначення факту його надходження через VPN, аналіз цільового порту, після чого приймається рішення щодо передачі даних до TCP-сервера або їх відхилення, що забезпечує базову реалізацію політики безпеки [29].

Налаштування середовища мережевої емуляції. Оскільки основною метою дослідження є аналіз безпеки передачі даних, бортова частина системи може бути реалізована у двох варіантах, серед яких повноцінний емулятор автопілота або спрощений мережевий емулятор, причому у даному випадку використовується другий підхід, що дозволяє зосередитися на транспортному рівні без ускладнення системи додатковою логікою. Для реалізації такого підходу застосовується утиліта socat, яка відкриває TCP-порт 5760 та приймає вхідні з'єднання, перенаправляючи всі отримані дані до системного пристрою "/dev/null", що фактично створює ефект «чорної діри», де трафік приймається, але не обробляється (див. рис. 2.4) [18].

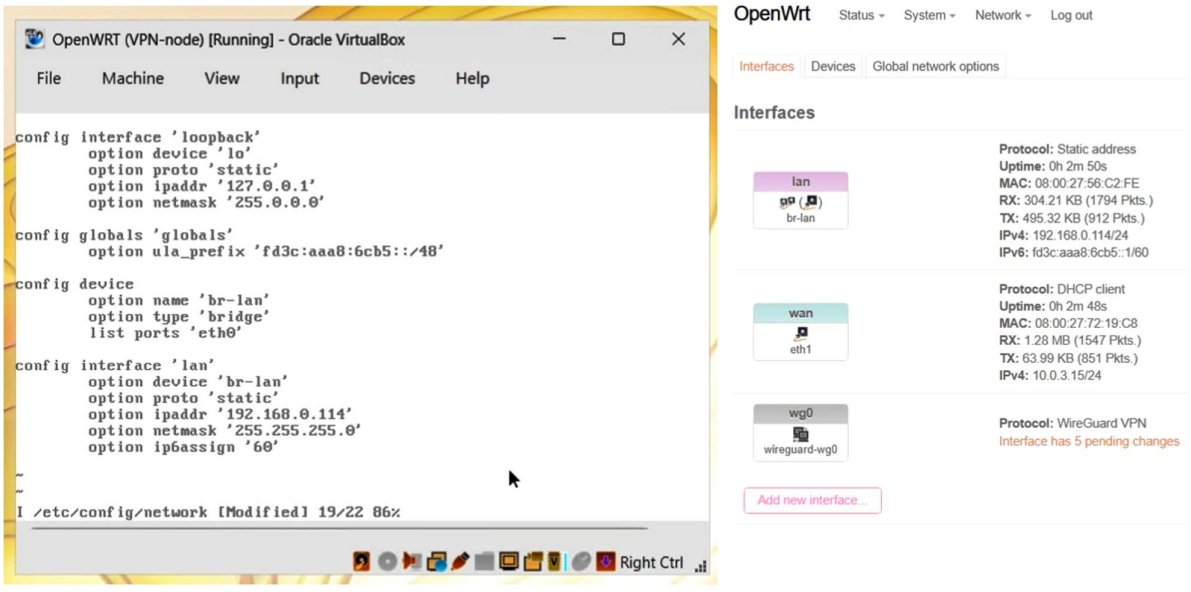


Рисунок 2.4 – Налаштування мережевих інтерфейсів у віртуальній машині OpenWrt 23.05.5

Функціонально така конфігурація означає, що процес відкриває порт, очікує підключення клієнта, встановлює TCP-сесію та приймає дані, після чого відбувається їхнє відкидання без виконання будь-яких операцій прикладного рівня, що дозволяє підтвердити працездатність мережевого інтерфейсу без необхідності реалізації повного протоколу автопілота.

У контексті мережевої емуляції така модель може бути представлена як взаємодія, де наземна станція передає дані через TCP, сервер приймає їх як "listener" і перенаправляє у нульовий пристрій, що забезпечує контрольований сценарій без зворотного зв’язку (див. рис. 2.5).



Рисунок 2.5 – Робота терміналу під час запуску
середовища емуляції ArduPilot SITL

Переваги такого підходу полягають у мінімальному використанні обчислювальних ресурсів, відсутності необхідності розгортання складних емуляторів, швидкості запуску та можливості чіткого розділення транспортного та прикладного рівнів, що особливо важливо для дослідження процесів маршрутизації, шифрування та перехоплення трафіку. Алгоритм роботи TCP-

сервера у такому середовищі включає етап ініціалізації, відкриття порту, очікування з’єднання, прийом TCP-сесії, обробку вхідних даних шляхом їх перенаправлення до "/dev/null" та створення нового процесу для обробки наступних підключень у разі використання параметра "fork" [20].

У більш деталізованій інтерпретації логіка роботи передбачає перевірку доступності порту, прийом нового з’єднання, аналіз необхідності обробки даних та прийняття рішення щодо їх передачі у систему або відкидання, причому у даному випадку реалізується саме сценарій відкидання, що дозволяє уникнути впливу прикладного рівня на результати дослідження. Незважаючи на відсутність повноцінної логіки автопілота, така конфігурація повністю відтворює поведінку мережевого інтерфейсу, забезпечує можливість встановлення TCP-з’єднань та дозволяє виконувати аналіз трафіку, що є достатнім для дослідження затримок, маршрутизації, шифрування та стійкості до перехоплення.

Конфігурація захищених тунелів та брандмауера. Захист передачі даних у сформованому середовищі реалізується через створення віртуального приватного тунелю на базі WireGuard, який забезпечує конфіденційність, цілісність та автентифікацію переданих пакетів, при цьому логіка роботи побудована таким чином, що прикладний трафік залишається прозорим для користувацьких застосунків і одночасно повністю захищеним на рівні мережевої інфраструктури, де базова конфігурація VPN включає інтерфейс із приватним ключем, внутрішньою адресацією та портом прослуховування, а також опис віддаленого вузла через його публічний ключ і дозволені IP-адреси, що формує замкнений криптографічний контур обміну даними між клієнтом і шлюзом у межах логічно ізольованої мережі (див. табл. 2.2) [14; 19].

Таблиця 2.2 – Основні параметри конфігурації захищеного тунелю та брандмауера

Параметр	Значення	Опис
VPN-протокол	WireGuard	Захищений тунель рівня L3
ListenPort	51820/UDP	Основний порт тунелю
Tunnel IP	10.0.0.0/24	Внутрішня адресація вузлів
AllowedIPs	10.0.0.0/24	Дозволені адреси маршрутизації
Firewall rule 1	Allow wg0 → TCP 5760	Доступ лише через VPN
Firewall rule 2	Drop WAN → TCP 5760	Блокування прямого доступу
NAT	Enabled	Маршрутизація між сегментами
Keepalive	25 s	Підтримка активного каналу

У такій конфігурації процес передачі даних набуває чітко визначеної послідовності, де первинний TCP-трафік, сформований наземною станцією керування на порту 5760, передається до віртуального мережевого інтерфейсу тунелю, після чого виконується інкапсуляція в UDP-пакети з використанням порту 51820, далі відбувається передача через зовнішню мережу у вигляді зашифрованого потоку, який не містить ознак початкового протоколу, а на стороні VPN-шлюзу здійснюється зворотна операція дешифрування та відновлення початкового TCP-сеансу з подальшою передачею на локальний сервіс, що забезпечує повну прозорість транспортного рівня для прикладного програмного забезпечення при одночасному приховуванні внутрішньої структури трафіку від сторонніх спостерігачів (див. рис. 2.6).

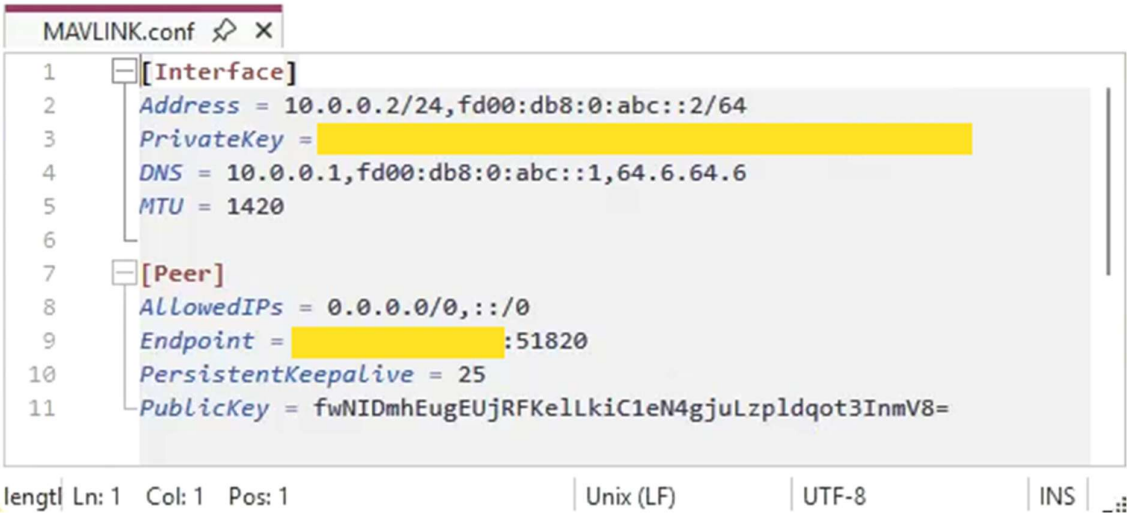


Рисунок 2.6 – Конфігураційний файл WireGuard-тунелю

Логічно така схема може бути представлена у вигляді послідовного ланцюга обробки, де формування пакета відбувається на рівні GCS, після чого виконується його передача через TCP-з'єднання, інкапсуляція у VPN-тунель, транспортування у вигляді UDP-потоку, прийом на шлюзі, дешифрування, перевірка політик доступу та передача до кінцевого сервісу, причому на кожному етапі зберігається можливість контролю та аналізу трафіку, що дозволяє досліджувати як вплив шифрування, так і поведінку системи в умовах різних мережевих сценаріїв.

Паралельно із впровадженням тунелю реалізується механізм мережевої фільтрації, який базується на принципі мінімальних привілеїв і обмежує доступ до критичних сервісів виключно через захищений інтерфейс, де правило доступу формалізується як дозвіл вхідного трафіку лише з VPN-інтерфейсу "wg0" до локального пристрою на порт 5760, тоді як будь-які інші спроби доступу автоматично відхиляються, що створює жорстку модель контролю, у якій відсутня можливість обходу тунелю або прямого підключення до сервісу керування (див. рис. 2.7).

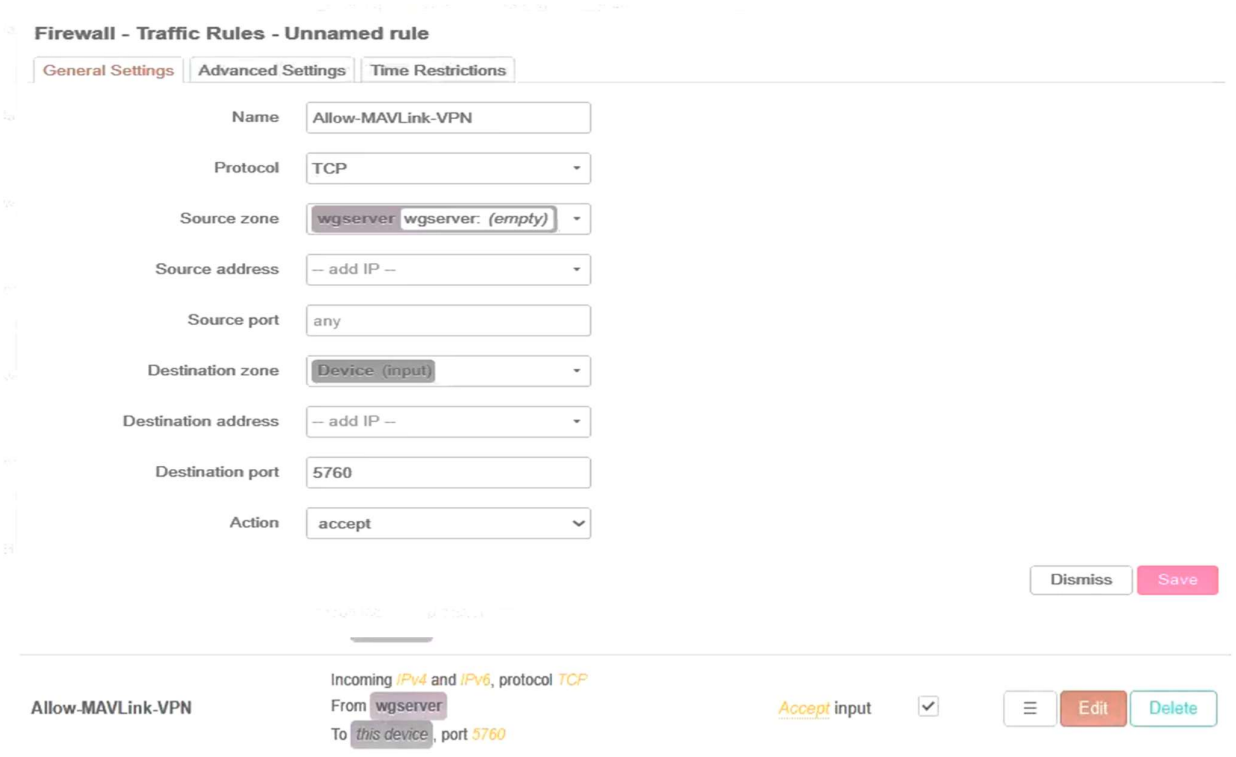


Рисунок 2.7 – Налаштування правил брандмауера UFW / iptables в OpenWRT

У контексті обробки пакетів така політика реалізується як послідовний алгоритм перевірок, де кожен вхідний пакет спочатку аналізується за джерелом надходження, після чого у разі невідповідності VPN-інтерфейсу відразу відкидається без подальшої обробки, при успішному проходженні першого етапу виконується перевірка цільового порту, де дозволеним вважається виключно порт 5760, після чого пакет передається до наступного етапу обробки, а у випадку невідповідності — також відхиляється, що формує чітку логіку прийняття рішень і мінімізує навантаження на систему за рахунок раннього відсікання небажаного трафіку (див. рис. 2.8).



Рисунок 2.8 – Схема фільтрації пакетів на VPN-шлюзі

З точки зору безпекового ефекту запропонована конфігурація дозволяє досягти ізоляції сервісу керування від відкритого мережевого середовища, унеможливлює сканування портів ззовні, запобігає прямому підключенню до бортового вузла без проходження криптографічного тунелю та забезпечує централізований контроль усіх вхідних з’єднань, при цьому будь-яка спроба

взаємодії з системою поза межами VPN автоматично завершується відмовою без розкриття додаткової інформації про внутрішню структуру мережі.

Загальна інтегрована модель функціонування формується як єдиний конвеєр, у якому генерація трафіку на стороні наземної станції переходить у фазу шифрування через WireGuard, далі здійснюється передача через UDP-канал, прийом на VPN-шлюзі супроводжується фільтрацією згідно з правилами брандмауера, після чого дані передаються до TCP-сервера, де виконується їх прийом або відкидання залежно від логіки обробки, причому така організація дозволяє поєднати реалістичність мережевої взаємодії з високим рівнем контролю та безпеки.

Сформований апаратно-програмний комплекс, який поєднує VPN-тунелювання, механізми фільтрації та імітаційний сервер, створює повноцінне середовище для дослідження процесів передачі даних у системах керування БПЛА, де використання реального програмного стеку забезпечує достовірність отриманих результатів, а спрощення бортового вузла дозволяє сфокусувати увагу саме на транспортному та криптографічному рівнях, у результаті чого досягається можливість ефективного аналізу поведінки захищених каналів зв’язку в умовах, максимально наближених до реальних, без необхідності розгортання складної фізичної інфраструктури.

2.3 Алгоритми обробки та захисту даних

У межах побудованого емуляційного середовища захищеного керування безпілотною системою визначальним є алгоритмічний рівень організації обробки даних, який задає порядок проходження, трансформації та контролю мережевих пакетів у режимі реального часу, оскільки на відміну від класичних інформаційних систем, де допускається буферизація або відкладена обробка, у системах дистанційного керування критичним є забезпечення безперервності та

детермінованості обміну, що обумовлює необхідність побудови чітко структурованого конвеєра обробки. Такий конвеєр реалізується як послідовність взаємопов'язаних етапів, у межах яких кожен пакет проходить:

1. перевірку політик доступу;
2. маршрутизацію;
3. інкапсуляцію;
4. криптографічний захист;
5. передачу.

Після чого на приймальній стороні виконується зворотний процес дешифрування, перевірки цілісності та передавання до прикладного рівня, що дозволяє забезпечити ізоляцію функцій і локалізацію потенційних вразливостей у конкретних вузлах системи [29].

Логіка функціонування такого конвеєра передбачає, що вхідний пакет, сформований на рівні наземної станції керування, спочатку потрапляє до механізмів мережевої фільтрації, де виконується перевірка відповідності встановленим правилам доступу, після чого у випадку дозволу здійснюється передача до транспортного рівня з формуванням TCP-сесії, далі пакет маршрутизується до віртуального мережевого інтерфейсу тунелю, де виконується його інкапсуляція та шифрування, після чого передача відбувається через захищений канал, а на приймальній стороні відбувається зворотна декрипція з використанням криптографічної пари ключів, що забезпечує відновлення початкового змісту даних і їх подальшу передачу до прикладного сервісу [29].

Алгоритм шифрування телеметричного потоку. Алгоритм шифрування телеметричного потоку реалізується на мережевому рівні операційної системи та функціонує прозоро відносно прикладного програмного забезпечення, що означає відсутність необхідності модифікації логіки роботи наземної станції керування, яка продовжує використовувати стандартне TCP-з'єднання для обміну даними, тоді як усі криптографічні операції виконуються на рівні мережевої інфраструктури. Початковим етапом процесу є формування TCP-запиту на встановлення з'єднання з використанням стандартного порту 5760, який традиційно застосовується для

передачі телеметричних даних, після чого сформований пакет передається до стеку мережевих протоколів і маршрутизується до віртуального інтерфейсу VPN.

На наступному етапі відбувається інкапсуляція TCP-пакета у структуру UDP, що дозволяє здійснювати його передачу через тунельований канал, після чого виконується шифрування з використанням алгоритму ChaCha20-Poly1305, який забезпечує одночасно конфіденційність і контроль цілісності даних завдяки інтегрованому механізму автентифікації повідомлень, далі сформований зашифрований пакет передається через мережу у вигляді потоку UDP-пакетів на порт 51820, що відповідає стандартній конфігурації тунелю [16].

На стороні приймача, представленій VPN-шлюзом, виконується зворотний процес дешифрування з використанням відповідних криптографічних ключів, після чого відновлений TCP-пакет передається до локального сервісу, який реалізує прийом з’єднань, у результаті чого забезпечується повна прозорість передачі для прикладного рівня, де програмне забезпечення не взаємодіє безпосередньо з механізмами шифрування, але отримує всі переваги захищеного каналу.

Послідовність виконання алгоритму може бути представлена як ланцюг операцій, що включає генерацію TCP-запиту, маршрутизацію до VPN-інтерфейсу, інкапсуляцію у UDP, шифрування, передачу через мережу, дешифрування та передачу до прикладного рівня, причому у випадку виникнення помилки на будь-якому етапі пакет не передається далі, що забезпечує контроль цілісності та запобігає використанню пошкоджених або модифікованих даних (див. рис. 2.9).

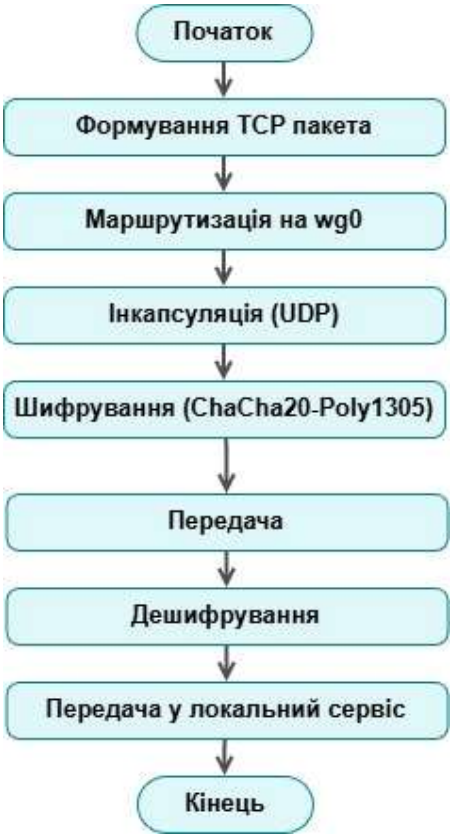


Рисунок 2.9 – Алгоритм шифрування телеметричного потоку

Серед характерних особливостей такого підходу відзначається прозорість для прикладного рівня, мінімальні затримки завдяки використанню потокового шифру, автоматична перевірка цілісності на основі MAC-функції Poly1305, а також відсутність необхідності внесення змін у логіку роботи наземної станції керування, що дозволяє інтегрувати захист без впливу на функціональність системи, при цьому реалізація відповідає концепції вбудованого захисту, де криптографічні механізми інтегруються безпосередньо у транспортний рівень і функціонують незалежно від прикладного програмного забезпечення [16].

Алгоритм ізоляції прикладного рівня. Поряд із криптографічним захистом у системі реалізується додатковий механізм безпеки, що базується на логічній ізоляції прикладного рівня взаємодії, яка спрямована на запобігання встановленню повноцінного обміну даними навіть у випадку успішного встановлення мережевого з’єднання, причому така ізоляція спирається на особливості функціонування

протоколу обміну, що використовується у системах керування безпілотними літальними апаратами, де ключовим елементом є службові повідомлення типу Heartbeat, які сигналізують про активність та готовність бортового контролера до взаємодії.

Логіка роботи алгоритму передбачає, що на першому етапі відбувається ініціалізація TCP-з'єднання клієнтом, який формує запит на підключення до адреси внутрішньої мережі VPN із використанням порту 5760, після чого запит передається через захищений тунель до VPN-шлюзу, де приймається як стандартне вхідне з'єднання, далі виконується повна процедура TCP-handshake із послідовністю SYN, SYN-ACK, ACK, у результаті чого з'єднання встановлюється на транспортному рівні та переходить у стан готовності до передачі даних.

Після встановлення з'єднання прикладний рівень переходить у режим очікування службових пакетів Heartbeat, які повинні надходити від бортового вузла як підтвердження коректності взаємодії, однак у межах реалізованого середовища генерація таких пакетів навмисно не виконується, у результаті чого відсутність цієї сигнатури призводить до блокування подальшого обміну на прикладному рівні, що формує чітку розмежувальну межу між фактом наявності мережевого доступу та можливістю виконання керуючих операцій [4].

Послідовність роботи алгоритму може бути представлена як умовно розгалужений процес, де після встановлення TCP-з'єднання виконується перевірка наявності Heartbeat-повідомлень, при їх отриманні система переходить до штатного режиму взаємодії з можливістю передачі телеметрії та команд керування, тоді як у разі їх відсутності формується стан очікування з подальшим завершенням сесії або повідомленням про помилку, що унеможливлює будь-які подальші дії з боку клієнта (див. рис. 2.10).

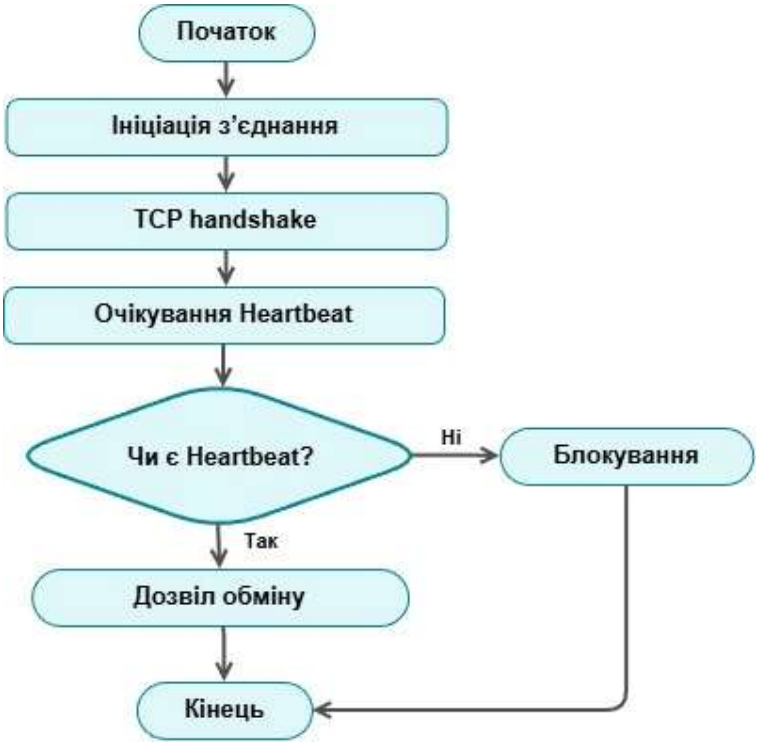


Рисунок 2.10 – Алгоритм ізоляції прикладного рівня

Практична інтерпретація такого підходу полягає у тому, що навіть за умови встановлення VPN-з'єднання та успішного проходження всіх мережевих перевірок сторонній суб'єкт не отримує можливості ініціювати повноцінний обмін даними, оскільки відсутність коректної відповіді на рівні протоколу взаємодії блокує логіку роботи прикладного програмного забезпечення, у результаті чого передані пакети залишаються без підтвердження і не призводять до виконання будь-яких команд.

З точки зору моделювання сценаріїв загроз така схема дозволяє розглянути ситуацію, у якій потенційний зловмисник отримує доступ до мережевого каналу та навіть встановлює TCP-з'єднання через VPN, однак не володіє можливістю сформувати валідні службові повідомлення, що відповідають очікуваному формату протоколу, у результаті чого система не переходить у стан активної взаємодії та не надає доступу до функцій керування, що значно знижує ризик несанкціонованого впливу.

Додатковим ефектом такого підходу є формування багаторівневої моделі захисту, у якій криптографічний рівень забезпечує конфіденційність і цілісність переданих даних, мережевий рівень контролює доступ через механізми

тунелювання та фільтрації, а прикладний рівень обмежує саму можливість взаємодії через перевірку логічної коректності протоколу, причому кожен із цих рівнів функціонує незалежно від інших, створюючи додаткові бар'єри для потенційних атак.

Поєднання зазначених механізмів дозволяє сформувати систему, у якій навіть повний компроміс окремого рівня не призводить до порушення загальної безпеки, оскільки відсутність підтвердження на рівні протоколу блокує виконання команд незалежно від стану мережевого з'єднання, а така організація взаємодії забезпечує контроль легітимності доступу на кожному етапі обробки даних і створює умови для стійкого функціонування системи в умовах потенційних загроз перехоплення, підміни або несанкціонованого підключення.

2.4 Тестування безпеки та надійності комплексу

Етап тестування сформованого емуляційного середовища захищеного керування орієнтований на практичну перевірку працездатності захищеного каналу зв'язку та оцінювання стійкості системи до аналізу і перехоплення мережевого трафіку, при цьому на відміну від теоретичного моделювання використовується експериментальний підхід із контрольованими умовами, де всі компоненти, включаючи наземну станцію керування, VPN-шлюз і мережеві утиліти, функціонують у передбачуваному середовищі, що дозволяє ізолювати вплив криптографічних механізмів і мережевих параметрів від сторонніх факторів та забезпечити коректність інтерпретації результатів.

Загальна структура тестування формується як послідовний процес, у межах якого виконується ініціалізація з'єднання з боку клієнта, передача трафіку через VPN-тунель, обробка пакетів на Linux-шлюзі, розгорнутому у вигляді віртуальної машини, а також паралельний аналіз мережевої активності на фізичному інтерфейсі, де фіксуються всі етапи проходження даних, що дозволяє оцінити

поведінку системи в умовах постійного обміну пакетами, характерного для телеметричних потоків (див. рис. 2.11).

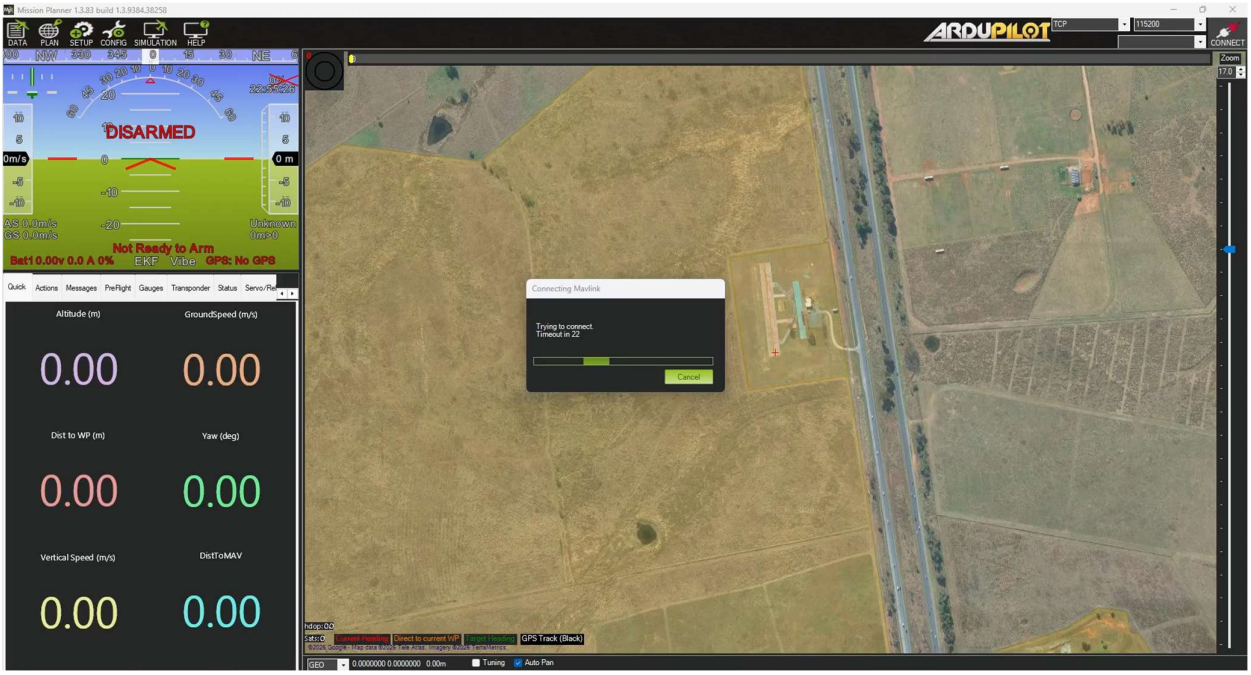


Рисунок 2.11 – Робота Mission Planner у режимі очікування з’єднання

У процесі експерименту наземна станція керування формує TCP-трафік на порт 5760, який передається до віртуального мережевого інтерфейсу тунелю, де виконується його інкапсуляція та шифрування, після чого у вигляді UDP-пакетів на порт 51820 здійснюється передача через мережу до VPN-шлюзу, на якому відбувається дешифрування, фільтрація відповідно до правил брандмауера та передача до локального TCP-сервера, що виконує функцію прийому з’єднань, при цьому паралельно здійснюється захоплення трафіку за допомогою аналізатора, що дозволяє дослідити структуру переданих даних та оцінити ефективність шифрування.

Постійне повторення циклу передачі забезпечує накопичення достатньої кількості даних для аналізу, що дозволяє виявити характерні закономірності функціонування системи, зокрема стабільність встановлення з’єднань, поведінку при відсутності відповіді прикладного рівня та особливості представлення зашифрованого трафіку на мережевому рівні.

Симуляція встановлення з'єднання в зашифрованому каналі. Основною метою експерименту є перевірка коректності встановлення мережевого з’єднання через захищений тунель та оцінка поведінки системи на прикладному рівні в умовах відсутності повноцінної відповіді від бортового вузла, де наземна станція керування ініціює підключення до вузла з адресою 10.0.0.1 із використанням порту 5760, а на стороні VPN-шлюзу функціонує TCP-лісенер, реалізований за допомогою утиліти, який приймає з’єднання, але не генерує відповідь прикладного рівня.

У межах виконання тесту відбувається послідовність дій, що включає формування TCP-запиту, його передачу через VPN-тунель, дешифрування на шлюзі та встановлення TCP-з’єднання із сервером, після чого система переходить у стан очікування відповіді прикладного рівня, що дозволяє оцінити поведінку клієнтського програмного забезпечення в умовах часткової доступності сервісу.

У ході експерименту фіксується характерна поведінка системи, яка дозволяє чітко розмежувати функціонування різних рівнів, де на мережевому рівні відсутність помилки типу "Connection Refused" свідчить про доступність порту, коректність налаштування брандмауера та успішне проходження пакетів через VPN-тунель, а також підтверджує здатність сервера приймати вхідні з’єднання, що вказує на працездатність механізмів маршрутизації та тунелювання.

На прикладному рівні після встановлення з’єднання система переходить у стан очікування службових повідомлень, при цьому фіксується повідомлення про встановлення з’єднання без подальшої ініціалізації протоколу, що пов’язано з відсутністю відповідних пакетів, у результаті чого повноцінний обмін даними не відбувається, а взаємодія залишається обмеженою рамками транспортного рівня (див. рис. 2.12).

```
root@OpenWrt:~# opkg install socat
Installing socat (1.7.4.4-1) to root...
Downloading https://downloads.openwrt.org/releases/23.05.5/packages/x86_64/package
s/socat_1.7.4.4-1_x86_64.ipk
Configuring socat.
root@OpenWrt:~# socat TCP-LISTEN:5760,fork /dev/null
```

Рисунок 2.12 – Термінал socat під час приймання TCP-запиту

Інтерпретація отриманих результатів дозволяє зробити висновок про навмисно сформований ефект, який демонструє коректність архітектури системи, де мережевий рівень забезпечує доступність каналу, криптографічний рівень гарантує захист переданих даних, а прикладний рівень залишається ізольованим у разі відсутності необхідних службових сигналів, що підтверджує реалізацію принципу, відповідно до якого наявність мережевого доступу не означає можливості повного контролю системи (див. табл. 2.3).

Таблиця 2.3 – Результати тестування встановлення з’єднання та перехоплення трафіку

Сценарій	Статус	Час з’єднання	Видимість payload	Результат
Відкритий канал TCP	Успішно	~1 с	Повна	Дані доступні для аналізу
WireGuard тунель	Успішно	~1.5 с	Відсутня	Дані зашифровані
Заблокований порт	Відмова	Timeout	Немає	Доступ заборонено
Sniffing відкритого каналу	Успішно	Реальний час	Повна	Можливе перехоплення
Sniffing VPN-каналу	Успішно	Реальний час	Немає	Лише UDP-потік
Відсутній Heartbeat	Частково	TCP OK	Немає	Транспортний рівень працює

Таким чином, у процесі тестування формується чітке розділення рівнів доступу, де доступ до мережі та встановлення TCP-з’єднання є допустимими операціями, тоді як доступ до функцій керування обмежується логікою прикладного протоколу, що забезпечує додатковий рівень безпеки та унеможливлює несанкціоноване втручання навіть за умови успішного встановлення з’єднання.

Аналіз стійкості до перехоплення трафіку. Етап тестування орієнтований на оцінювання захищеності каналу зв’язку від пасивного аналізу мережевого трафіку та визначення можливості витоку інформації при перехопленні пакетів на фізичному рівні, при цьому перехоплення виконується безпосередньо на фізичному

мережевому інтерфейсі під час активної спроби встановлення з'єднання з боку наземної станції управління, яка передає дані через VPN-тунель, а сам процес аналізу базується на фіксації пакетів та дослідженні їх структури у контрольованому середовищі.

У ході експерименту формується типовий мережевий сценарій, у якому трафік від GCS, що первинно генерується у вигляді TCP-пакетів на порт 5760, проходить через механізм тунелювання, де відбувається його інкапсуляція у UDP-пакети з використанням порту 51820, після чого передається через фізичний інтерфейс, доступний для моніторингу, і саме на цьому рівні здійснюється захоплення трафіку для подальшого аналізу, що дозволяє оцінити реальний вигляд пакетів з точки зору стороннього спостерігача (див. рис. 2.13).

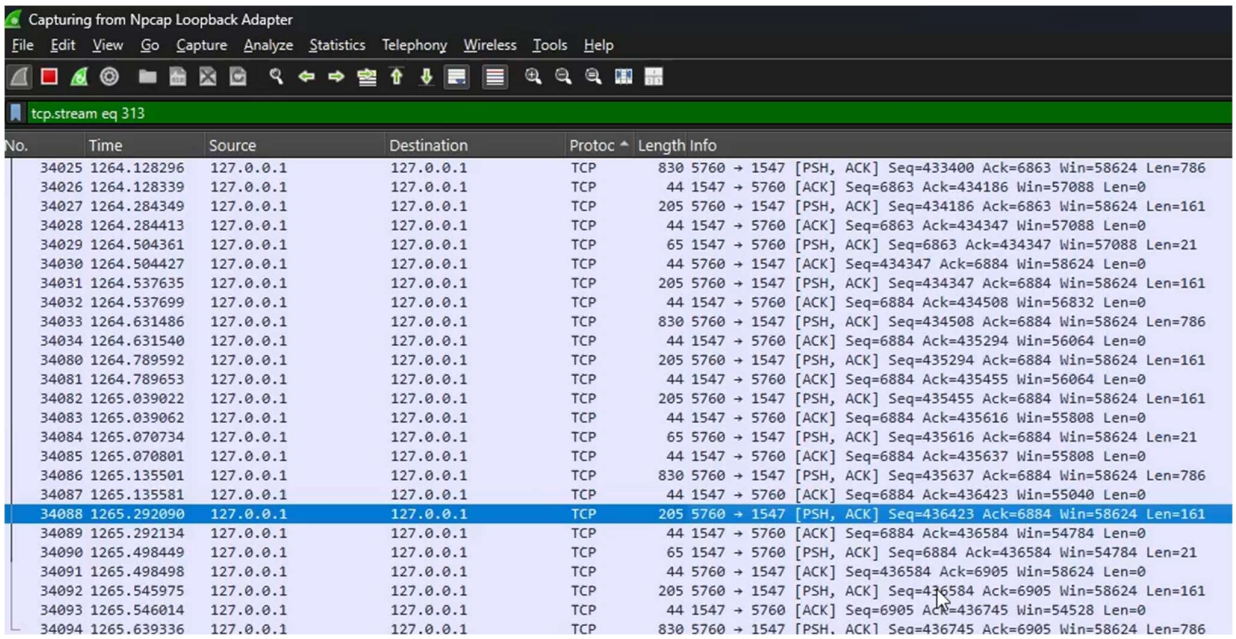


Рисунок 2.13 – Захоплення трафіку у Wireshark на фізичному інтерфейсі

Результати аналізу демонструють, що весь трафік, який спостерігається у мережі, має уніфікований вигляд UDP-пакетів, асоційованих виключно з портом 51820, що відповідає роботі WireGuard, при цьому будь-які ознаки первинного TCP-з'єднання або структури прикладного протоколу повністю відсутні, що свідчить про повну інкапсуляцію початкових даних, унаслідок чого зловмисник не

має можливості визначити, який саме протокол використовується на вищому рівні (див. рис. 2.14) [11].

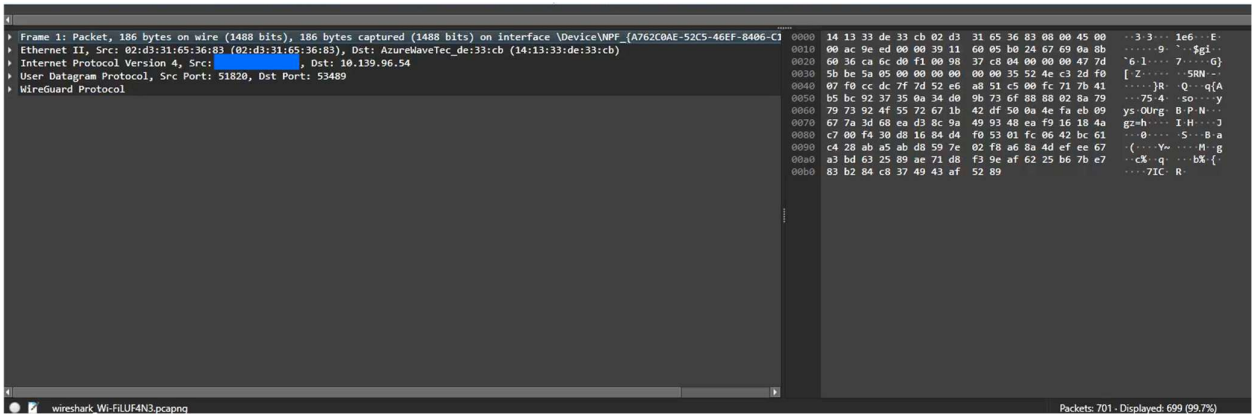


Рисунок 2.14 – UDP-пакети WireGuard у мережевому дампі

Подальший аналіз вмісту пакетів на рівні корисного навантаження дозволяє встановити відсутність будь-яких читабельних або структурованих даних, включаючи текстові сигнатури чи службові заголовки, натомість спостерігається високий рівень ентропії, що проявляється у вигляді псевдовипадкової послідовності байтів, характерної для зашифрованих даних, унаслідок чого неможливо ідентифікувати тип інформації, що передається, її призначення або внутрішню структуру (див. рис. 2.15).

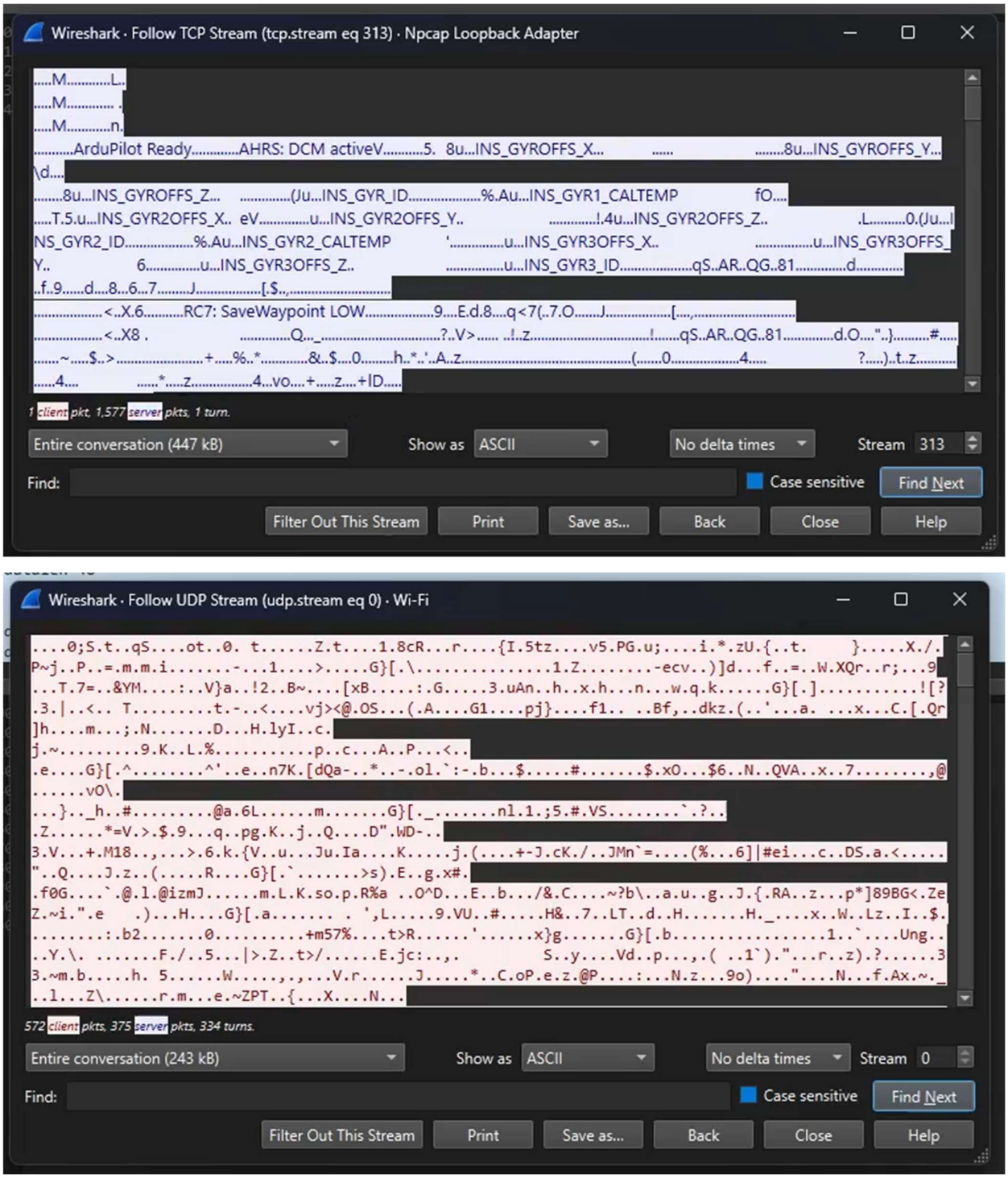


Рисунок 2.15 – Зіставлення відкритого та зашифрованого трафіку на основі отриманого дампу зі Wireshark

З точки зору алгоритму аналізу перехопленого трафіку процес можна інтерпретувати як послідовність перевірок, де після отримання пакета визначається тип транспортного протоколу, далі виконується спроба ідентифікації прикладного

рівня за сигнатурами, після чого аналізується структура payload, і якщо у випадку відкритого каналу можливе виявлення структурованих даних або заголовків протоколу, то у випадку використання VPN-тунелювання всі ці етапи не дають результату через відсутність доступу до незашифрованого вмісту.

Інтерпретація отриманих результатів дозволяє зробити висновок про повну ізоляцію прикладного трафіку від рівня мережевого спостереження, оскільки інкапсуляція приховує не лише вміст пакетів, але й сам факт використання конкретного прикладного протоколу, унаслідок чого навіть при повному контролі каналу зв’язку неможливо визначити, чи передаються керуючі команди, телеметрія або інші типи даних.

З практичної точки зору така поведінка системи означає, що перехоплення трафіку не надає зловмиснику жодної корисної інформації, оскільки відсутні як відкриті дані, так і метадані, що дозволяють зробити висновки про структуру або логіку роботи системи, унаслідок чого будь-які спроби аналізу зводяться до роботи з зашифрованим потоком без можливості його декодування.

Узагальнення результатів тестування підтверджує, що мережевий рівень функціонує коректно, забезпечуючи стабільне встановлення TCP-з’єднання без помилок, криптографічний рівень гарантує повну інкапсуляцію та шифрування трафіку, що унеможливлює його аналіз, а прикладний рівень залишається ізольованим, що додатково обмежує можливість несанкціонованої взаємодії, при цьому сукупність цих факторів формує багаторівневу модель захисту, у якій кожен рівень незалежно контролює доступ і обробку даних.

Отримані результати також підтверджують доцільність використання емульованого середовища для дослідження захищених каналів зв’язку, оскільки воно дозволяє відтворити реальні умови передачі даних і одночасно забезпечити повний контроль над процесом експерименту, що створює можливість детального аналізу ефективності застосованих механізмів захисту та їх впливу на загальну стійкість системи до перехоплення та аналізу трафіку.

3. АНАЛІТИЧНЕ ОЦІНЮВАННЯ ЕФЕКТИВНОСТІ ТА ПЕРСПЕКТИВИ РОЗВИТКУ СИСТЕМИ

3.1 Аналітичне та практичне оцінювання продуктивності захищеної системи

У межах оцінювання ефективності емульованого середовища захищеного керування доцільним є поєднання аналітичного підходу з практичними результатами, отриманими під час експериментального тестування, що дозволяє сформувати цілісне уявлення про вплив криптографічного захисту на продуктивність системи та рівень її безпеки, при цьому основна увага зосереджується на дослідженні змін характеристик трафіку в умовах переходу від відкритого до інкапсульованого середовища, а також на визначенні компромісу між захищеністю та ефективністю передачі даних.

Методологія дослідження базується на поєднанні аналізу мережевого трафіку, зафіксованого під час роботи системи, з розрахунком накладних витрат, що виникають унаслідок застосування криптографічних механізмів, а також на візуальному спостереженні змін структури пакетів, що дозволяє не лише підтвердити працездатність запропонованої архітектури, але й кількісно оцінити вплив захисту на ключові параметри функціонування каналу зв’язку [30].

Оцінка впливу криптографічних перетворень на затримки та пропускну здатність. У відкритому режимі передавання даних телеметричні пакети мають відносно невеликий розмір, що залежить від типу повідомлення та в середньому знаходиться в діапазоні від 17 до 74 байт, при цьому після застосування механізму тунелювання відбувається інкапсуляція, яка призводить до збільшення загального розміру пакета за рахунок додавання службових заголовків та криптографічних даних, і загальна довжина пакета може бути описана співвідношенням, у якому сумарний розмір дорівнює сумі корисного навантаження та накладних витрат [23].

У цьому випадку корисне навантаження відповідає безпосередньо даним протоколу телеметрії, тоді як накладні витрати становлять приблизно 32 байти, що є характерним для використаного механізму тунелювання, і на практичному прикладі можна показати, що для пакета розміром 32 байти після інкапсуляції загальний розмір становить 64 байти, що означає збільшення у два рази та відповідно зменшення частки корисного навантаження до 50 відсотків, тоді як для пакета розміром 64 байти сумарний розмір становить 96 байтів, що відповідає збільшенню приблизно на 50 відсотків (див. табл. 3.1) [23].

Таблиця 3.1 – Вплив інкапсуляції на розмір пакета та ефективність каналу

Payload (байт)	Overhead (байт)	Загальний розмір	Збільшення	Ефективність
32	32	64	100%	50%
64	32	96	50%	66.7%
128	32	160	25%	80%
256	32	288	12.5%	88.9%
512	32	544	6.25%	94.1%

Таке зростання розміру пакетів безпосередньо впливає на ефективність використання пропускної здатності каналу, оскільки частина ресурсу витрачається на передачу службових даних, і ефективна пропускна здатність визначається як добуток реальної швидкості каналу на відношення корисного навантаження до загального розміру пакета, унаслідок чого при передачі невеликих пакетів ефективність може знижуватися до рівня 50–70 відсотків (див. рис. 3.1).

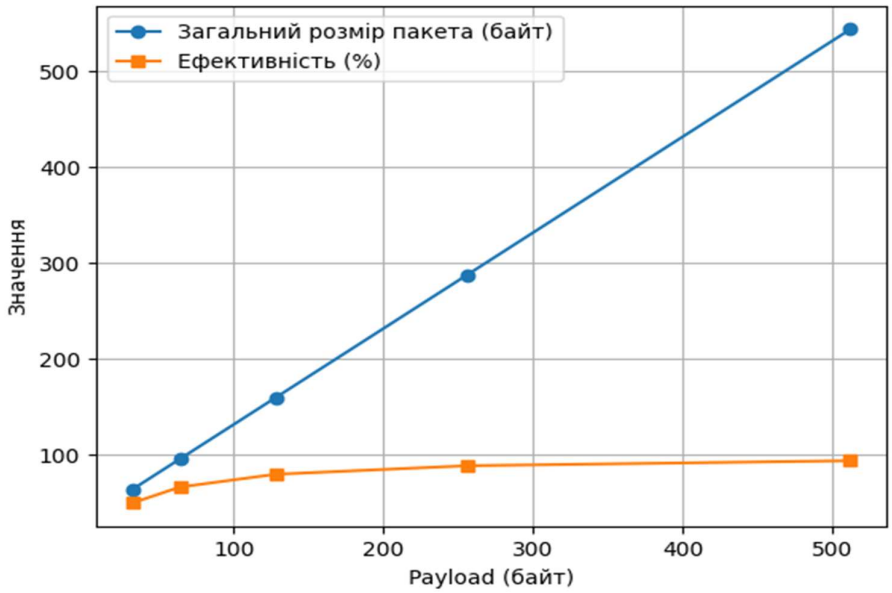


Рисунок 3.1 – Вплив криптографічного оверхеду на розмір пакета

Поряд із цим враховується вплив криптографічних перетворень на затримки передачі даних, які формуються як сума часу, необхідного для виконання операцій шифрування, дешифрування, обробки пакетів у мережевому стеку та їх маршрутизації, при цьому затримка проходження одного пакета включає етап формування, інкапсуляції, шифрування, передачі через мережу, дешифрування та передачі на прикладний рівень, однак ці затримки мають стабільний характер і не накопичуються критичним чином у процесі роботи системи.

Аналіз отриманих результатів дозволяє встановити, що накладні витрати найбільш помітні для малих пакетів, де відносна частка службових даних є максимальною, водночас вплив на канал залишається передбачуваним, а затримки перебувають у допустимих межах для задач дистанційного керування, що дозволяє сформулювати висновок про відсутність критичного впливу криптографічного захисту на функціонування системи навіть у випадку використання каналів із обмеженою пропускною здатністю.

Порівняльний аналіз безпеки: відкритий трафік vs інкапсульовані тунелі. Порівняльне оцінювання безпеки базується на аналізі двох принципово різних сценаріїв передачі даних, де у першому випадку використовується відкритий канал

без застосування криптографічного захисту, а у другому — передача здійснюється через зашифрований тунель, при цьому основою для висновків виступають результати перехоплення та подальшого аналізу мережевого трафіку, що дозволяє безпосередньо оцінити рівень доступності інформації для стороннього спостерігача [24].

У сценарії відкритого каналу структура переданих пакетів залишається повністю доступною для аналізу, що означає можливість безперешкодного читання як службових заголовків, так і корисного навантаження, унаслідок чого зловмисник отримує доступ до критично важливих даних, включаючи координати, параметри польоту, службову телеметрію та інші характеристики функціонування системи, причому сам факт відсутності будь-якого шифрування робить процес перехоплення не лише можливим, але й ефективним з точки зору подальшого використання отриманої інформації [25].

Такий підхід формує середовище, у якому будь-який вузол, що має доступ до каналу зв’язку, здатен здійснювати моніторинг або аналіз переданих даних, а також виконувати активні дії, спрямовані на модифікацію або підміну пакетів, що відкриває можливість реалізації атак із втручанням у процес передачі, включаючи зміну керуючих команд або нав’язування хибної телеметрії.

У сценарії використання захищеного тунелю передача даних набуває принципово іншого вигляду, оскільки всі пакети проходять етап інкапсуляції та перетворюються у UDP-потік, у межах якого корисне навантаження повністю зашифроване, що унеможливлює прямий доступ до вмісту повідомлень, при цьому навіть у разі перехоплення трафіку аналіз обмежується лише спостереженням за зашифрованими пакетами без можливості визначення їх змісту або призначення [24].

У такому середовищі зловмисник стикається з криптографічно захищеним потоком даних, де відсутні будь-які ознаки прикладного протоколу, а структура пакетів прихована за рахунок інкапсуляції, що значно ускладнює або повністю унеможливлює проведення аналізу, спрямованого на отримання корисної інформації.

Порівняння типових загроз дозволяє встановити, що у відкритому каналі перехоплення трафіку надає повний доступ до даних, тоді як у тунельованому середовищі доступ обмежується лише зашифрованим потоком, який не піддається інтерпретації, при цьому атаки типу втручання в канал передачі у відкритому середовищі можуть призводити до підміни даних або модифікації повідомлень, тоді як у захищеному тунелі такі спроби блокуються механізмами криптографічного контролю цілісності (див. рис. 3.2).

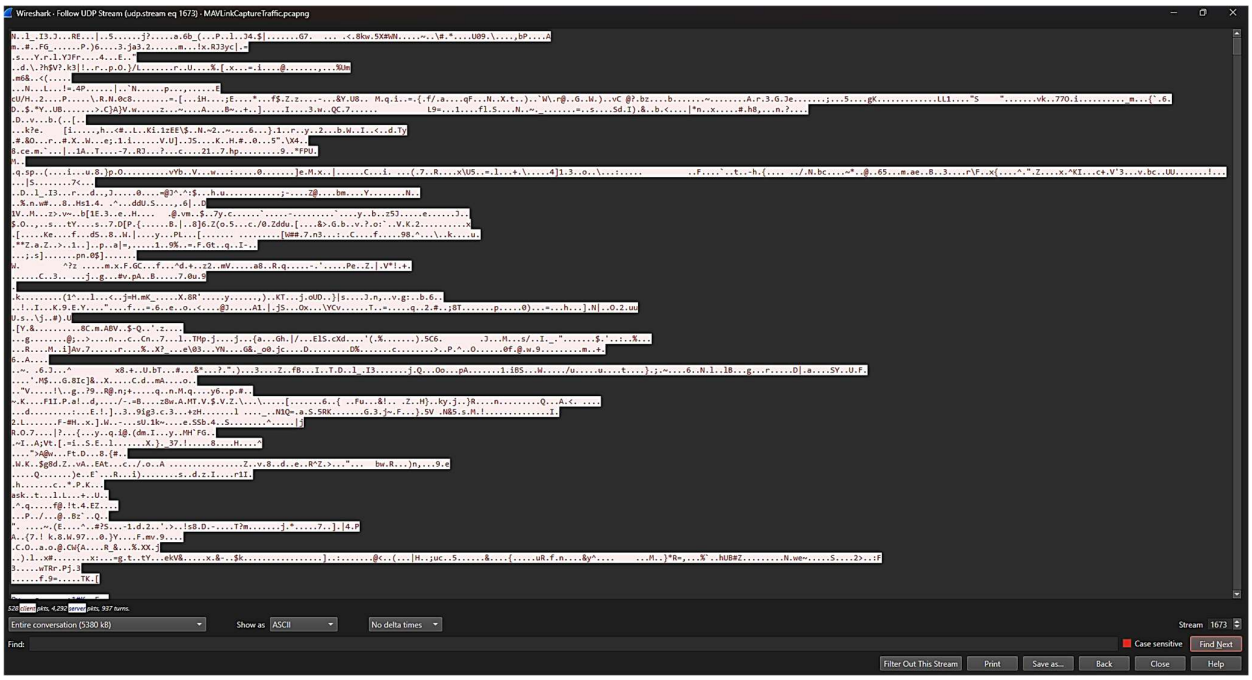


Рисунок 3.2 – Неможливість аналізу трафіку в інкапсульованому каналі

Аналогічно ситуація спостерігається для атак повторного відтворення пакетів, де у відкритому каналі можливе повторне використання перехоплених даних з метою впливу на систему, тоді як у тунельованому середовищі такі пакети не проходять перевірку і відхиляються, що виключає можливість їх повторного застосування.

Інтерпретація результатів порівняльного аналізу демонструє принципову відмінність між підходами, де відкритий канал не забезпечує жодного рівня захисту і дозволяє здійснювати як пасивний, так і активний вплив на систему, тоді як

використання тунелювання формує ізольоване середовище передачі, у якому дані захищені як від перехоплення, так і від модифікації.

Узагальнення аналітичних та практичних результатів дозволяє встановити, що продуктивність системи зберігається на прийнятному рівні навіть з урахуванням накладних витрат, а затримки не досягають значень, здатних вплинути на стабільність керування, при цьому впровадження криптографічного захисту є обґрунтованим, оскільки незначне зниження ефективності компенсується суттєвим підвищенням рівня безпеки.

Сформована модель захисту забезпечує конфіденційність переданих даних, гарантує їх цілісність та обмежує доступ до системи керування, що відповідає вимогам до побудови захищених каналів зв’язку у задачах дистанційного управління, унаслідок чого навіть за умов перехоплення трафіку відсутня можливість отримання або використання будь-якої корисної інформації.

3.2 Оптимізація та адаптація системи захисту

У процесі проєктування захищеної системи керування безпілотними літальними апаратами одним із найважливіших завдань є досягнення збалансованого співвідношення між рівнем криптографічного захисту, швидкодією мережевої взаємодії та стабільністю функціонування всієї системи. Для класичних інформаційних мереж допустимими є певні додаткові затримки, підвищене навантаження на процесор або збільшення службового трафіку, оскільки такі фактори рідко впливають критично на робочий процес. У системах дистанційного керування БПЛА ситуація суттєво відрізняється, оскільки навіть незначне зростання затримки передачі команд, короткочасне перевантаження обчислювального вузла чи втрата окремих пакетів можуть безпосередньо впливати на стабільність польоту, точність маневрування та загальну безпеку виконання місії [29].

Особливістю бортових платформ є використання енергоефективних процесорів із обмеженими обчислювальними ресурсами, компактних одноплатних комп'ютерів або вбудованих контролерів, для яких надмірно складні алгоритми шифрування можуть створювати відчутний «оверхед». У зв'язку з цим система захисту повинна не лише гарантувати конфіденційність і цілісність переданих даних, а й зберігати здатність працювати у режимі реального часу без деградації каналу керування [29].

Паралельно із вибором криптографічних механізмів значну увагу приділено мережевій фільтрації, оскільки навіть добре зашифрований канал може втрачати ефективність у разі перевантаження стороннім трафіком, спробами сканування портів, масовими запитами або навмисними атаками на вузол зв'язку. Саме тому оптимізація системи захисту розглядається як поєднання двох взаємопов'язаних напрямів: використання швидких та стійких алгоритмів шифрування з мінімальним впливом на затримку, а також впровадження суворих механізмів контролю доступу, які блокують небажані пакети ще до етапу їх подальшої обробки.

В*ибір оптимального балансу «стійкість шифрування — мінімальна затримка».* Одним із базових рішень у межах побудови захищеного каналу зв'язку став вибір VPN-технології та криптографічного алгоритму, здатного забезпечити високий рівень безпеки без критичного зростання затримок. Для реалізації запропонованої архітектури обрано WireGuard із використанням алгоритму ChaCha20-Poly1305, що зумовлено технічними особливостями систем керування БПЛА та характеристиками обчислювальних платформ, які застосовуються на борту.

Традиційні VPN-рішення, зокрема OpenVPN, у більшості випадків використовують алгоритми сімейства AES, найчастіше AES-256. Такий підхід забезпечує високий рівень криптографічної стійкості та широко застосовується в корпоративних мережах, однак його ефективність значною мірою залежить від наявності апаратного прискорення AES-NI. На сучасних серверних процесорах це не створює проблем, проте на малопотужних ARM-платформах, одноплатних комп'ютерах або бортових модулях без спеціалізованих інструкцій навантаження

суттєво зростає. Наслідком стає підвищене використання процесора, зростання часу шифрування пакетів та нестабільні мікрозатримки.

На відміну від цього, ChaCha20 є потоковим шифром, спеціально орієнтованим на швидке програмне виконання без залежності від спеціалізованих апаратних модулів. Алгоритм демонструє стабільну продуктивність як на x86-процесорах, так і на ARM-платформах, що особливо важливо для компактних бортових систем. Обчислення виконуються рівномірно, без різких пікових навантажень, а отже зменшується ризик короткочасного перевантаження процесора під час активного обміну телеметрією [14; 16].

Комбінація ChaCha20 з механізмом автентифікації Poly1305 забезпечує одночасно шифрування та контроль цілісності пакета. Після формування корисного навантаження дані проходять інкапсуляцію, шифруються, доповнюються тегом автентичності та передаються каналом зв’язку. На приймальній стороні виконується перевірка цілісності, після чого пакет дешифрується та надходить у прикладний сервіс. Така схема усуває ризики непомітної модифікації пакетів, повторного використання даних або впровадження сторонніх команд у телеметричний потік [14; 16].

Для систем реального часу ключовим параметром залишається загальна затримка доставки пакета від оператора до бортового вузла. Будь-який VPN додає власний службовий цикл обробки, що включає маршрутизацію, інкапсуляцію, шифрування та дешифрування. Перевагою WireGuard є мінімалістична архітектура ядра та мала кількість проміжних операцій, завдяки чому додатковий latency залишається незначним і стабільним навіть за інтенсивного потоку пакетів [24].

Якщо порівнювати підходи узагальнено, OpenVPN з AES-256 забезпечує дуже високий рівень захисту, проте створює більший процесорний оверхед, складнішу архітектуру обробки та вищі затримки на слабкому обладнанні. WireGuard з ChaCha20-Poly1305 також забезпечує високий рівень стійкості, але водночас зменшує навантаження на CPU, спрощує конфігурацію та мінімізує затримки передачі команд (див. табл. 3.2).

Таблиця 3.2 – Порівняння відкритого каналу, OpenVPN та WireGuard для БПЛА

Критерій	Відкритий канал	OpenVPN	WireGuard
Захист даних	Низький	Високий	Високий
Затримка	Мінімальна	Середня / вища	Низька
Навантаження CPU	Низьке	Високе	Низьке
Складність налаштування	Низька	Висока	Низька
Придатність до FPV	Висока, але ризикована	Середня	Висока
Стійкість до перехоплення	Низька	Висока	Висока
Робота на слабких ARM	Висока	Середня	Висока
Рекомендація для БПЛА	Небажано	Допустимо	Оптимально

Таким чином, обраний криптографічний підхід дозволяє досягти необхідного рівня конфіденційності та цілісності без критичного впливу на часові характеристики системи керування, що робить його найбільш доцільним для використання у безпілотних платформах з обмеженими ресурсами живлення та продуктивності.

Методи мережевої фільтрації та відсікання аномального трафіку. Окрім безпосередньо криптографічного захисту важливим елементом безпеки системи керування безпілотним літальним апаратом залишається контроль мережевого доступу та фільтрація трафіку на ранніх етапах його обробки, оскільки саме такий підхід дозволяє мінімізувати зайве навантаження на вузли системи, знизити ризик перевантаження службових процесів і підвищити загальну стійкість до зовнішніх впливів. Для БПЛА подібне рішення має особливе значення через обмеженість обчислювальних ресурсів бортових пристроїв, енергоспоживання та необхідність стабільної роботи каналу керування в режимі реального часу, де навіть другорядні процеси можуть впливати на затримку передачі команд або телеметрії.

У межах розробленої архітектури центральним елементом контролю мережевої взаємодії виступає Linux-шлюз, реалізований у тестовому середовищі як окрема віртуальна машина на базі OpenWrt 23.05.5. Саме цей вузол виконує функції

маршрутизації, прийому VPN-з’єднань, дешифрування трафіку, застосування правил доступу та перенаправлення пакетів до локального сервісу. Завдяки такому підходу формується єдина точка контролю, у межах якої можуть централізовано реалізовуватися механізми безпеки без втручання у логіку роботи наземної станції керування або бортового програмного забезпечення.

Основою конфігурації мережевого захисту є політика мінімальних привілеїв, відповідно до якої дозволяється лише той трафік, що є необхідним для штатного функціонування системи, а всі інші запити автоматично блокуються. Подібна модель вважається однією з найбільш ефективних для критичних інфраструктур, оскільки система не намагається визначати всі можливі загрози окремо, а спочатку закриває доступ до всього непотрібного та відкриває лише вузький перелік дозволених сервісів [28].

Типове правило доступу для розробленого середовища може бути сформоване таким чином:

1. джерело трафіку — VPN-інтерфейс "wg0";
2. призначення — локальний пристрій (або сам шлюз);
3. порт — 5760, дія — ACCEPT.

У практичному сенсі це означає, що з’єднання до сервісу керування допускаються виключно з уже встановленого WireGuard-тунелю, тоді як будь-які прямі звернення з фізичної мережі, зовнішнього інтерфейсу або інших адрес не отримують доступу до службового порту.

Логіка обробки пакета у такій схемі є послідовною та прозорою. Після надходження мережевого пакета спочатку перевіряється інтерфейс, з якого він отриманий. Якщо джерелом не є VPN-інтерфейс, пакет негайно відкидається політикою DROP без передачі у прикладний рівень. Якщо пакет надійшов через "wg0", виконується перевірка порту призначення. У випадку невідповідності встановленому значенню 5760 пакет також блокується. Лише за умови відповідності обом параметрам відбувається передача даних до локального сервісу, яким у тестовому середовищі виступає socat або емулятор SITL (див. рис. 3.3).

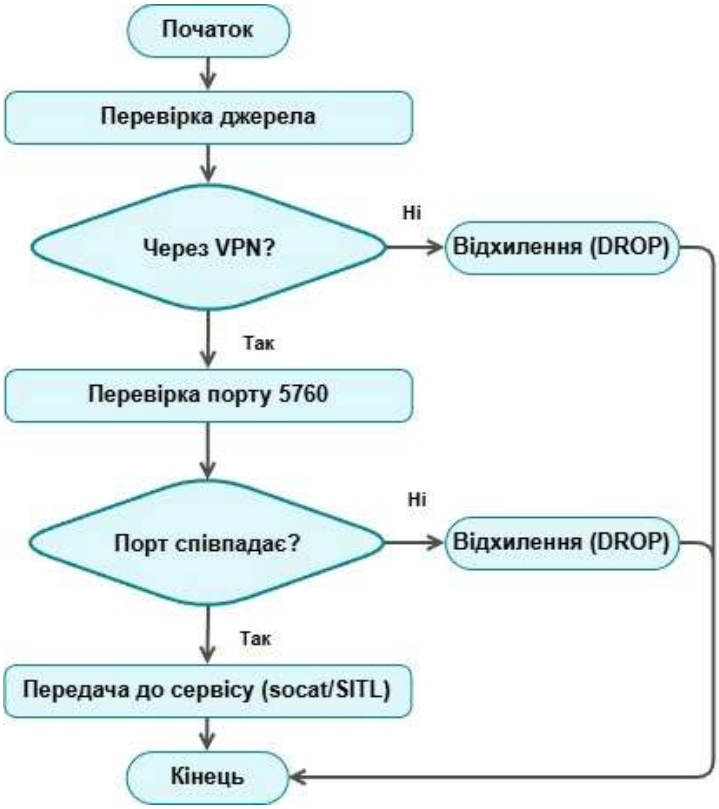


Рисунок 3.3 – Відсікання небажаних пакетів на рівні VPN-шлюзу

Завдяки цьому виключається зайва обробка стороннього трафіку та зменшується навантаження на процесор шлюзу.

Практична користь мережевої фільтрації особливо проявляється під час протидії аномальному трафіку. До таких сценаріїв належать:

1. автоматизоване сканування портів;
2. масові спроби підключення до службових інтерфейсів;
3. випадковий широкомовний шум локальної мережі;
4. елементи примітивних DDOS-атак.

Останні можуть бути спрямовані не на злам системи, а на виснаження її ресурсів. Якщо подібні пакети досягають прикладного рівня, кожен із них споживає частину часу процесора, пам’яті та мережевого стеку. Для звичайного серверного обладнання це часто не створює критичних наслідків, однак для компактного вузла зв’язку БПЛА навіть помірне перевантаження може впливати на стабільність каналу керування.

Саме тому небажаний трафік доцільно відсікати ще на рівні шлюзу, до передачі в автопілот або супутній сервіс. У такому випадку бортова система працює лише з легітимними пакетами, а процесорні ресурси використовуються для виконання основного завдання — підтримання польоту, обробки телеметрії та виконання команд оператора. Додатковою перевагою стає зменшення ризику зависань прикладних сервісів через надлишкову кількість з’єднань або помилково сформовані пакети.

Мережевий захист у запропонованій архітектурі працює у тісному поєднанні з криптографічним рівнем. Спочатку зовнішній трафік проходить через правила Firewall, далі легітимні пакети потрапляють до WireGuard, де здійснюється автентифікація та дешифрування, після чого дозволені дані передаються до TCP-служби та прикладного рівня. Така багаторівнева модель означає, що зловмиснику недостатньо лише виявити порт або мережевий маршрут, оскільки без коректного VPN-сеансу трафік буде заблокований ще до спроби подальшої взаємодії.

З практичної точки зору використання описаної архітектури забезпечує зниження навантаження на систему, підвищення стійкості до зовнішніх атак, ізоляцію критичних сервісів керування та загальне зростання надійності комплексу. Поєднання швидкого шифрування ChaCha20-Poly1305 у складі WireGuard із суворими правилами фільтрації на VPN-шлюзі (OpenWrt) формує збалансовану систему безпеки, у якій захист каналу зв’язку не створює надмірних затримок, а небажаний трафік усувається ще до його потрапляння у внутрішнє середовище. Саме така модель найбільш повно відповідає вимогам продуктивності, стабільності та надійності, що ставляться до сучасних систем керування БПЛА.

3.3 Напрямки практичного впровадження та масштабування

Розроблена модель захищеного каналу керування безпілотним літальним апаратом, реалізована у вигляді емульованого середовища, має не лише

дослідницьке, але й безпосереднє прикладне значення, оскільки побудована архітектура орієнтована на подальше перенесення у реальні апаратні комплекси без суттєвої зміни базової логіки функціонування. Основною перевагою запропонованого підходу є апаратно незалежна структура, за якої криптографічні механізми, маршрутизація трафіку, фільтрація пакетів та контроль доступу винесені в окремий мережевий вузол, що дозволяє адаптувати систему до різних платформ, типів БПЛА та сценаріїв застосування. Такий принцип забезпечує можливість масштабування рішення від експериментального стенду до серійних зразків без необхідності глибокого втручання у штатне програмне забезпечення автопілота.

Практичне впровадження подібної моделі доцільно розглядати у двох основних напрямах. Перший пов'язаний з інтеграцією системи у серійні апаратні платформи, де емульований віртуальний шлюз замінюється компактним фізичним обчислювальним модулем на борту. Другий стосується подальшого розвитку криптографічної стійкості, автоматизації управління ключами та адаптації каналу зв'язку до умов активної протидії, зокрема засобів радіоелектронної боротьби, перехоплення або аналізу сигнатур мережевого трафіку. Саме така дворівнева стратегія дозволяє одночасно забезпечити реальне впровадження існуючого рішення та сформувати резерв для його модернізації в майбутньому [30].

Архітектурні рішення для інтеграції моделі у серійні апарати («UKROPTER»). Перехід від емуляційного середовища до реальної апаратної реалізації потребує трансформації віртуального VPN-шлюзу (OpenWrt) у фізичний обчислювальний пристрій, здатний працювати безпосередньо у складі бортової системи БПЛА. У тестовому середовищі роль такого вузла виконувала віртуальна машина на базі OpenWrt 23.05.5, яка приймала VPN-з'єднання, застосовувала правила Firewall, здійснювала дешифрування трафіку та передавала дані локальному сервісу. У серійному апараті аналогічні функції доцільно покласти на окремий companion computer — допоміжний бортовий комп'ютер, що працює паралельно з польотним контролером.

Як приклади подібних платформ можуть використовуватися компактні одноплатні комп’ютери класу Raspberry Pi Zero, Orange Pi, Radxa Zero, NanoPi або спеціалізовані ARM-модулі промислового виконання. Такі пристрої поєднують достатню продуктивність для роботи WireGuard, низьке енергоспоживання, малу масу та можливість інтеграції у внутрішній простір дрона без критичного впливу на льотні характеристики (див. рис. 3.4) [26-27].



Рисунок 3.4 – Використання "companion computer" у складі БПЛА

У цій структурі радіомодем забезпечує фізичний безпровідний канал зв’язку із наземною станцією керування, companion computer виконує функції мережевого шлюзу, приймає зашифрований трафік, обробляє VPN-сеанс, перевіряє правила доступу та передає вже очищені дані до автопілота. Польотний контролер працює з телеметрією та командами у звичному форматі через MAVLink, не взаємодіючи напряму із зовнішнім середовищем.

Послідовність обробки трафіку у реальній системі виглядає таким чином: команда оператора надходить із наземної станції до радіомодему, після чого передається на companion computer; далі виконується автентифікація вузла, дешифрування пакета, перевірка правил Firewall, маршрутизація та передача через UART або USB до Pixhawk. У зворотному напрямку телеметрія від польотного контролера надходить на companion computer, інкапсулюється у WireGuard-тунель, шифрується та передається через радіоканал оператору.

Такий підхід забезпечує повну ізоляцію критичного автопілота від зовнішньої мережі. Навіть у випадку сканування портів, масового стороннього

трафіку або спроби встановлення несанкціонованого з’єднання контакт відбувається лише з companion computer, тоді як Pixhawk залишається доступним виключно через внутрішній інтерфейс. Це суттєво знижує ризики експлуатації вразливостей автопілота, оскільки мережевий доступ до нього фізично відсутній.

Додатковою перевагою є модульність архітектури. Криптографічний модуль винесено за межі польотного контролера, тому оновлення WireGuard, зміна політик безпеки, додавання журналювання або нових механізмів захисту не потребують перепрошивки автопілота. Завдяки цьому одна й та сама схема може використовуватися на різних моделях БПЛА незалежно від типу контролера, якщо він підтримує стандартний MAVLink-інтерфейс.

Важливим фактором є також практична відповідність між емуляційною та реальною моделлю. У лабораторному середовищі використовувався віртуальний VPN-шлюз, WireGuard-інтерфейс, TCP-сервіс та контроль трафіку через Firewall. У реальному апараті ці самі функції переходять на companion computer, а роль емулятора замінює фізичний польотний контролер. Логіка взаємодії при цьому зберігається: зовнішній канал не контактує напряму з автопілотом, а проходить через захищений проміжний вузол.

Таким чином, запропонована архітектура дозволяє безпосередньо перенести результати дослідження у серійні апарати типу «UKROPTER» або інші безпілотні платформи, забезпечуючи високий рівень кіберзахисту без ускладнення основної системи керування, без зміни принципів роботи автопілота та без критичного збільшення маси чи енергоспоживання комплексу.

Перспективи використання динамічної зміни ключів (Key Rotation) та захисту від РЕБ. У базовій конфігурації VPN WireGuard використовує статичні криптографічні ключі, що забезпечує простоту налаштування, високу швидкодію та стабільність роботи каналу зв’язку, однак в умовах підвищених ризиків, особливо під час активного застосування засобів радіоелектронної боротьби, виникає потреба у впровадженні додаткових механізмів захисту. Тривале використання незмінних ключів потенційно створює низку загроз, серед яких:

1. компрометація ключового матеріалу;

2. накопичення статистики трафіку;
3. виявлення шаблонів передавання даних;
4. підвищення ймовірності атак повторення.

За наявності достатнього часу спостереження противник може формувати поведінкові моделі роботи каналу, визначати активність системи та намагатися використовувати ці відомості для подальшого втручання [28].

Для підвищення стійкості системи доцільним є впровадження механізму динамічної ротації ключів, що передбачає періодичну зміну криптографічних параметрів у межах поточної сесії зв’язку. Такий підхід може реалізовуватися за таймером або за подіями мережевої активності. Після ініціалізації каналу зв’язку система працює зі встановленим набором ключів протягом визначеного інтервалу часу, наприклад десяти хвилин, після чого автоматично генерується новий ключовий матеріал, виконується захищений обмін параметрами між вузлами, оновлюється конфігурація тунелю та продовжується робота без критичного переривання сеансу. Скорочення часу життя одного ключа істотно знижує цінність перехоплених даних, ускладнює криптоаналіз, зменшує наслідки можливої компрометації та підвищує загальну стійкість системи до повторного використання раніше зафіксованих пакетів (див. рис. 3.5) [28].



Рисунок 3.5 – Алгоритм динамічної зміни ключів Key Rotation

Окремим напрямком розвитку є адаптація захищеного каналу до умов радіоелектронної боротьби. Окрім власне криптографічного захисту важливим стає приховування самого факту використання VPN-з’єднання, оскільки стандартні мережеві сигнатури можуть бути виявлені за характерними ознаками трафіку, стабільним використанням одного порту, повторюваною структурою пакетів або постійною інтенсивністю передавання даних. Для зниження ймовірності виявлення доцільно застосовувати зміну портів зв’язку, варіативність часових інтервалів передачі, модифікацію параметрів потоку, а також методи обфускації, що ускладнюють класифікацію трафіку засобами автоматизованого аналізу.

Практична багаторівнева модель протидії РЕБ може будуватися за послідовністю, у якій фізичний канал зв’язку доповнюється маскуванням мережевої активності, далі застосовується VPN-шифрування, після чого працюють механізми мережевої фільтрації та контроль прикладного рівня. У такій архітектурі

навіть у разі виявлення радіоканалу суттєво ускладнюється визначення структури переданих даних, логіки роботи системи та способів впливу на неї (див. рис. 3.6).



Рисунок 3.6 – Захист каналу зв'язку від РЕБ шляхом маскування трафіку

Поєднання ротації ключів, мережевого маскування, криптографічного захисту та фільтрації трафіку дозволяє сформувати стійку багаторівневу систему безпеки, здатну протидіяти як пасивному спостереженню, так і активному втручанню. Така архітектура зберігає переваги модульного підходу, оскільки функції захисту можуть виконуватися окремим обчислювальним вузлом без втручання у базову логіку автопілота або польотного контролера. Перенесення механізмів кіберзахисту на рівень companion computer забезпечує ізоляцію критичних компонентів, спрощує модернізацію системи та підвищує її експлуатаційну надійність.

Запропонований підхід демонструє не лише ефективність у межах емуляційного середовища, а й наявність чітких шляхів подальшого практичного впровадження у серійні комплекси керування БПЛА. Подальший розвиток доцільно спрямовувати на автоматизовану ротацію ключів, інтелектуальну зміну мережевих параметрів, адаптивне реагування на ознаки радіоелектронного впливу та інтеграцію засобів аналізу загроз у реальному часі, що дозволить істотно підвищити рівень кіберзахисту систем у бойових та експлуатаційних умовах.

ВИСНОВКИ

У ході роботи розроблено емульоване середовище захищеного керування БПЛА, яке відтворює основні процеси взаємодії між наземною станцією керування та бортовим вузлом. Побудована архітектура тестового стенду дозволила змоделювати передавання команд, телеметрії та службового трафіку через проміжний захищений шлюз із використанням стандартних мережевих засобів. Запропонований підхід підтвердив можливість створення функціонального дослідницького середовища без використання реального літального апарата.

У межах побудованого комплексу реалізовано VPN-тунелювання на базі WireGuard, налаштовано механізми маршрутизації та брандмауера, а також сформовано логіку обробки пакетів на транспортному рівні. Використання захищеного тунелю забезпечило конфіденційність передавання даних, а фільтрація доступу дозволила обмежити несанкціоновані підключення. Додатково впроваджено механізм ізоляції прикладного рівня, за якого мережеве з’єднання не гарантує доступу до функцій керування.

Проведене тестування показало коректність встановлення з’єднання через зашифрований канал та працездатність усіх ключових компонентів системи. Аналіз мережевого трафіку підтвердив, що під час перехоплення на фізичному інтерфейсі спостерігаються лише зашифровані UDP-пакети без доступу до змісту команд або телеметрії. Отримані результати довели ефективність багаторівневої моделі захисту, у якій поєднуються криптографія, мережева фільтрація та логічний контроль доступу.

Аналітичне оцінювання засвідчило, що криптографічний захист створює помірні накладні витрати, які не є критичними для систем керування БПЛА. Використання WireGuard із алгоритмом ChaCha20-Poly1305 забезпечує раціональний баланс між стійкістю та швидкодією, а застосування мережевої фільтрації підвищує загальну надійність комплексу. Перспективними напрямками подальшого розвитку визначено інтеграцію моделі у серійні платформи.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. U.S. Department of Defense. Unmanned Systems Integrated Roadmap 2017–2042. URL: https://apps.dtic.mil/sti/citations/AD1046149 (дата звернення: 25.02.2026).
2. PX4 Autopilot Documentation. Vehicle Telemetry and MAVLink Communication. URL: https://docs.px4.io/ (дата звернення: 28.02.2026).
3. ArduPilot Documentation. MAVLink Telemetry Overview. URL: https://ardupilot.org/ (дата звернення: 03.03.2026).
4. MAVLink Development Team. MAVLink Protocol Specification. URL: https://mavlink.io/en/ (дата звернення: 07.03.2026).
5. ExpressLRS Official Documentation. URL: https://www.expresslrs.org/ (дата звернення: 11.03.2026).
6. Team BlackSheep. TBS Crossfire System Overview. URL: https://www.team-blacksheep.com/ (дата звернення: 15.03.2026).
7. OWASP Foundation. Spoofing Attacks Explained. URL: https://owasp.org/ (дата звернення: 19.03.2026).
8. CISA. Wireless Communication Security Best Practices. URL: https://www.cisa.gov/ (дата звернення: 23.03.2026).
9. NATO Cooperative Cyber Defence Centre of Excellence. GPS Spoofing and Jamming Risks. URL: https://ccdcoe.org/ (дата звернення: 27.03.2026).
10. NIST. Guide to Industrial Wireless Security. URL: https://csrc.nist.gov/ (дата звернення: 31.03.2026).
11. Wireshark Foundation. Wireshark User Guide. URL: https://www.wireshark.org/docs/ (дата звернення: 04.04.2026).
12. IETF. The Transport Layer Security (TLS) Protocol Version 1.3. URL: https://datatracker.ietf.org/doc/html/rfc8446 (дата звернення: 08.04.2026).
13. OpenSSL Project. OpenSSL Documentation. URL: https://www.openssl.org/docs/ (дата звернення: 12.04.2026).
14. WireGuard Project. WireGuard: Fast, Modern, Secure VPN Tunnel. URL: https://www.wireguard.com/ (дата звернення: 16.04.2026).
15. WireGuard Project. WireGuard Cryptography Overview. URL: https://www.wireguard.com/protocol/ (дата звернення: 20.04.2026).
16. ChaCha20-Poly1305 for IETF Protocols. RFC 8439. URL: https://datatracker.ietf.org/doc/html/rfc8439 (дата звернення: 23.04.2026).
17. Cisco Systems. Network Segmentation Design Guide. URL: https://www.cisco.com/ (дата звернення: 26.04.2026).
18. OpenWrt Documentation. OpenWrt User Guide. URL: openwrt.org (дата звернення: 26.04.2026)
19. Netfilter Project. iptables / nftables Documentation. URL: https://netfilter.org/documentation/ (дата звернення: 28.04.2026).
20. Linux Manual Pages. socat Utility Manual. URL: https://man7.org/linux/man-pages/ (дата звернення: 28.04.2026).

21. Mininet Project. Network Emulation Testbed. URL: http://mininet.org/ (дата звернення: 29.04.2026).
22. Oracle. VirtualBox User Manual. URL: https://www.virtualbox.org/manual/ (дата звернення: 30.04.2026).
23. NIST. Performance Testing of Cryptographic Modules. URL: https://csrc.nist.gov/ (дата звернення: 30.04.2026).
24. Cloudflare Learning Center. What Is VPN Tunneling? URL: https://www.cloudflare.com/learning/ (дата звернення: 01.05.2026).
25. MITRE ATT&CK. Adversary-in-the-Middle Techniques. URL: https://attack.mitre.org/ (дата звернення: 03.05.2026).
26. Raspberry Pi Foundation. Raspberry Pi Zero Technical Specifications. URL: https://www.raspberrypi.com/ (дата звернення: 03.05.2026).
27. Orange Pi. Orange Pi Product Documentation. URL: http://www.orangepi.org/ (дата звернення: 05.05.2026).
28. NSA Cybersecurity Information. Network Hardening Guidance. URL: https://www.nsa.gov/Cybersecurity/ (дата звернення: 05.05.2026).
29. Груша Д. Р., Бученко І. А. Апаратно-програмний комплекс захищеного керування БПЛА з використанням сучасних мережевих технологій. «Сучасний стан та перспективи розвитку IoT» : VII Науково-техн. конф., м. Київ, 20 квітня 2026 р. С.106-109.
30. Груша Д. Р., Бученко І. А. Захищене передавання даних у реальному часі для систем керування БПЛА з використанням тунелювання та сегментації трафіку. «Застосування програмного забезпечення в ІКТ» : VII Всеукр. наук.-техн. конф., м. Київ, 23 квітня 2026 р. (подано до публікації).

Додаток А

ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ

(Презентація)

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



КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня бакалавра

НА ТЕМУ: «АПАРАТНО-ПРОГРАМНИЙ КОМПЛЕКС ЗАХИЩЕНОГО КЕРУВАННЯ БПЛА»

Виконав студент групи КІД-41
Груша Данило Романович
Керівник кваліфікаційної роботи:
Бученко Ігор Анатолійович,
старший викладач кафедри КІ

Київ – 2026

Мета, об’єкт, предмет дослідження

Мета роботи – підвищення безпеки та надійності керування БПЛА шляхом використання тунелювання трафіку, криптографічного захисту, фільтрації доступу та сегментації каналів зв’язку

Об’єкт дослідження – процеси передавання команд керування, телеметрії та службового трафіку у системах дистанційного керування БПЛА

Предмет дослідження – засоби й технології побудови апаратно-програмних комплексів захищеного керування БПЛА на основі VPN, криптографічних алгоритмів, мережевої фільтрації та емуляційних середовищ

Актуальність роботи

У роботі представлено теоретичні основи побудови каналів зв’язку безпілотних систем, архітектуру мережевої взаємодії БПЛА різних класів, особливості протоколів обміну даними, типові вразливості безпровідних каналів та методи протидії перехопленню, підміні сигналів і засобам радіоелектронної боротьби. Розглянуто криптографічні підходи до захищеного передавання даних у реальному часі, принципи інкапсуляції трафіку, сегментації потоків та побудови наземних станцій керування.

Особливу увагу приділено практичній реалізації емульованого середовища захищеного керування БПЛА із використанням WireGuard, VPN-шлюзу (на базі OpenWrt 23.05.5), брандмауера та алгоритмів ізоляції прикладного рівня. Проведено тестування встановлення з’єднання, аналіз стійкості до перехоплення трафіку, оцінювання впливу криптографічних перетворень на затримки та пропускну здатність, а також визначено перспективи інтеграції рішення у серійні платформи й адаптації до умов радіоелектронної протидії.

Базова архітектура керування БПЛА

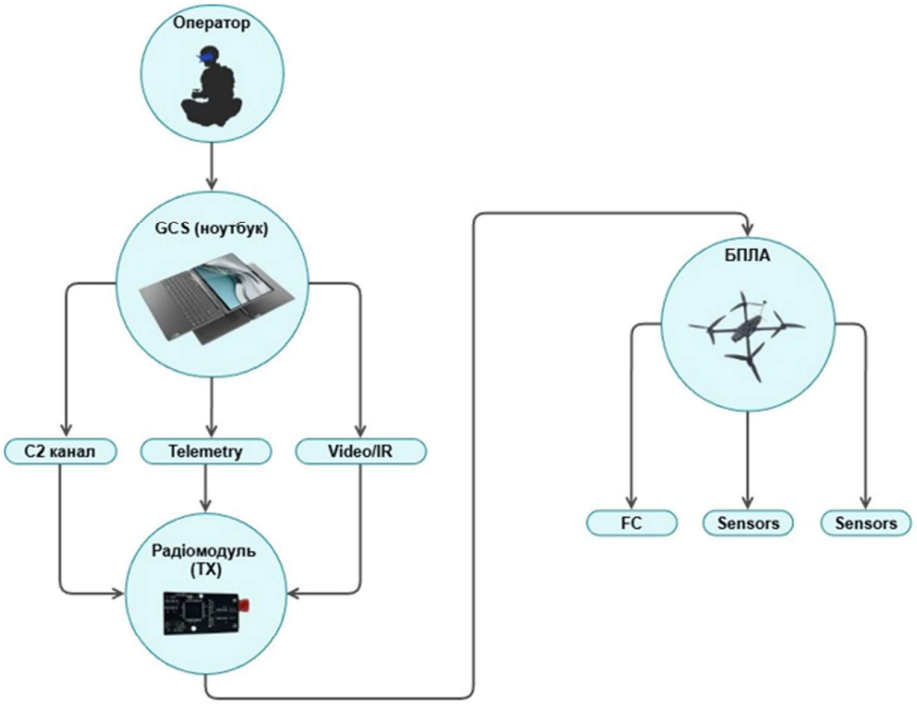


Рис. 1.2 – Схема передавання даних між оператором і бортовими модулями важкого БПЛА

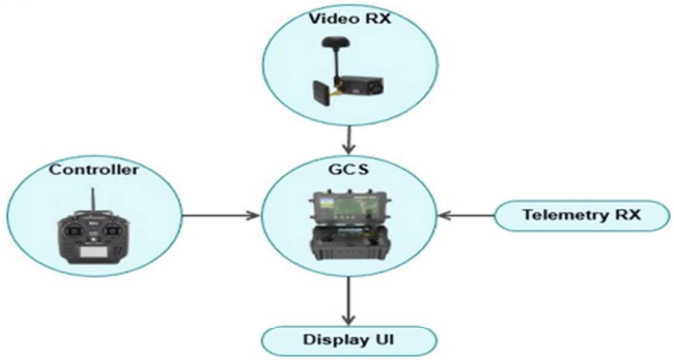


Рис. 1.10 – Типова структура наземної станції керування GCS

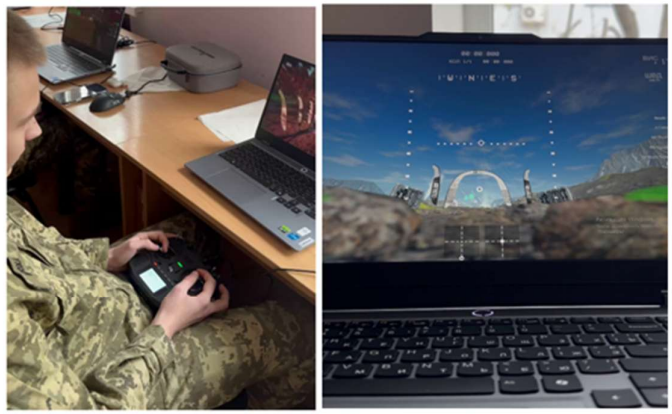


Рис. 1.11 – Тренування керуванням БПЛА в симуляторі зі пультом керування типу TX12 та ноутбуком-трансформером

Запропоноване рішення

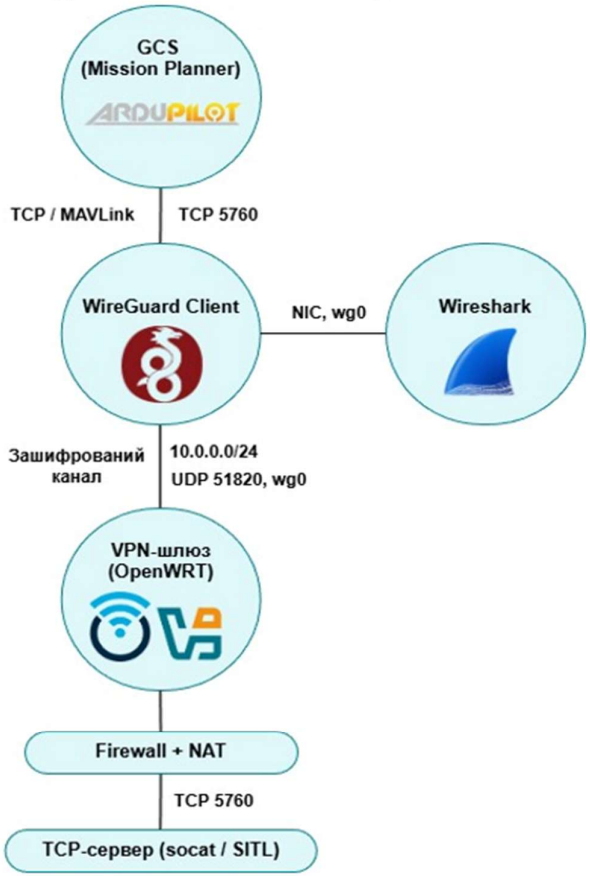


Рис. 2.1 – Архітектура апаратно-програмного комплексу захищеного керування БПЛА

Практична реалізація стенду

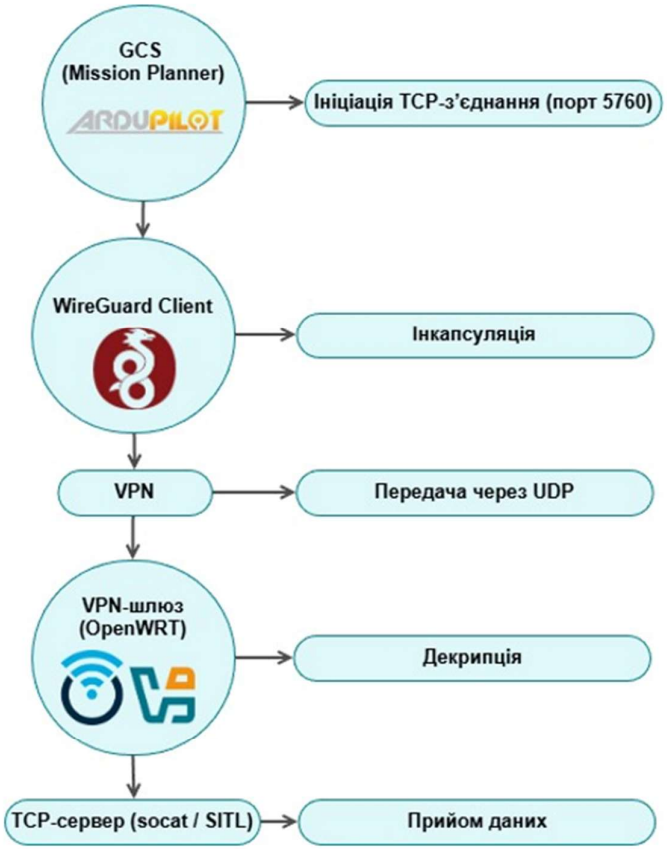


Рис. 2.2 – Апаратна та мережева схема взаємодії компонентів стенду



Рис. 2.3 – Схема руху телеметричного трафіку між вузлами

Налаштування OpenWrt-вузла

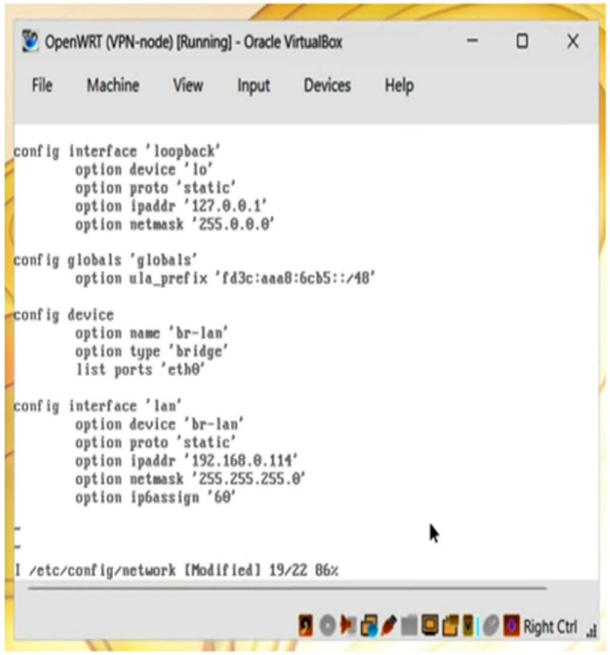




Рис. 2.4 – Налаштування мережевих інтерфейсів у віртуальній машині OpenWrt 23.05.5

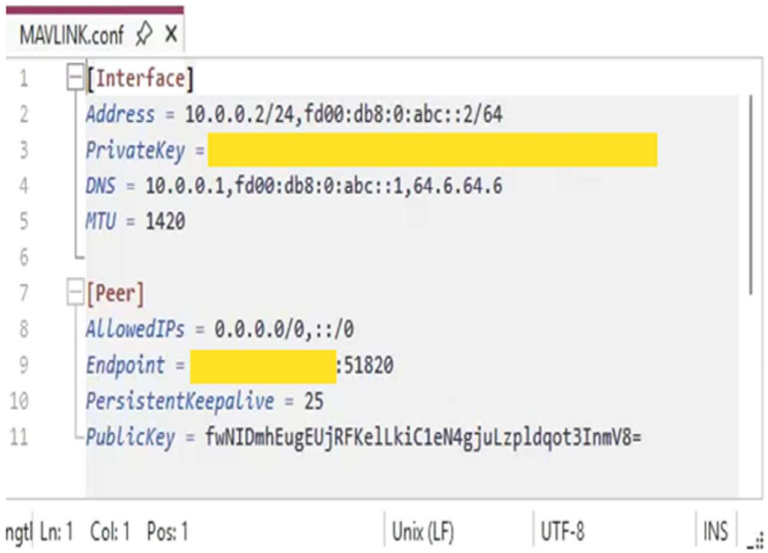


Рис. 2.6 – Конфігураційний файл WireGuard-тунелю

Демонстрація роботи системи

```
C:\Users\danya\Documents\Mission Planner\sitl\ArduCopter.exe
Starting sketch 'ArduCopter'
Starting SITL input
Using Irlock at port : 9005
bind port 5760 for SERIAL0
SERIAL0 on TCP port 5760
Loaded defaults from @ROMFS/default_params/copter.parm,C:\Users\danya\Documents\Mission Planner\sitl\default_params/copt
er.parm
Smoothing reset at 0.001
bind port 5762 for SERIAL1
SERIAL1 on TCP port 5762
bind port 5763 for SERIAL2
SERIAL2 on TCP port 5763
Loaded defaults from @ROMFS/default_params/copter.parm,C:\Users\danya\Documents\Mission Planner\sitl\default_params/copt
er.parm
validate_structures:538: Validating structures
Waiting for internal clock bits to be set (current=0x00)
Loaded defaults from @ROMFS/default_params/copter.parm,C:\Users\danya\Documents\Mission Planner\sitl\default_params/copt
er.parm
Loaded defaults from @ROMFS/default_params/copter.parm,C:\Users\danya\Documents\Mission Planner\sitl\default_params/copt
er.parm
Loaded defaults from @ROMFS/default_params/copter.parm,C:\Users\danya\Documents\Mission Planner\sitl\default_params/copt
er.parm
Loaded defaults from @ROMFS/default_params/copter.parm,C:\Users\danya\Documents\Mission Planner\sitl\default_params/copt
er.parm
Loaded defaults from @ROMFS/default_params/copter.parm,C:\Users\danya\Documents\Mission Planner\sitl\default_params/copt
er.parm
Loaded defaults from @ROMFS/default_params/copter.parm,C:\Users\danya\Documents\Mission Planner\sitl\default_params/copt
er.parm
New connection on SERIAL0
```

Рис. 2.5 – Робота терміналу під час запуску середовища емуляції ArduPilot SITL

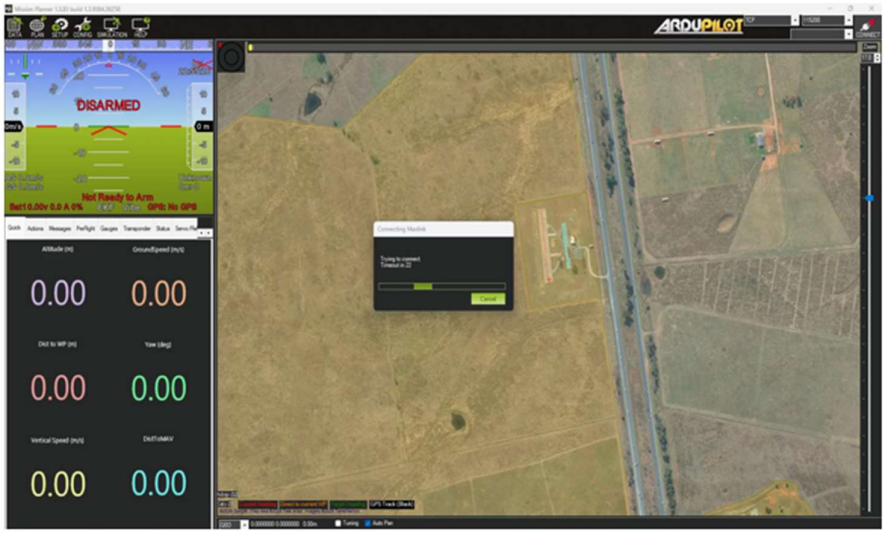


Рис. 2.11 – Робота Mission Planner у режимі очікування з’єднання

```
root@OpenWrt:~# opkg install socat
Installing socat (1.7.4.4-1) to root...
Downloading https://downloads.openwrt.org/releases/23.05.5/packages/x86_64/package
s/socat_1.7.4.4-1_x86_64.ipk
Configuring socat.
root@OpenWrt:~# socat TCP-LISTEN:5760,fork /dev/null
```

Рис. 2.12 – Термінал socat під час приймання TCP-запиту

Аналіз трафіку

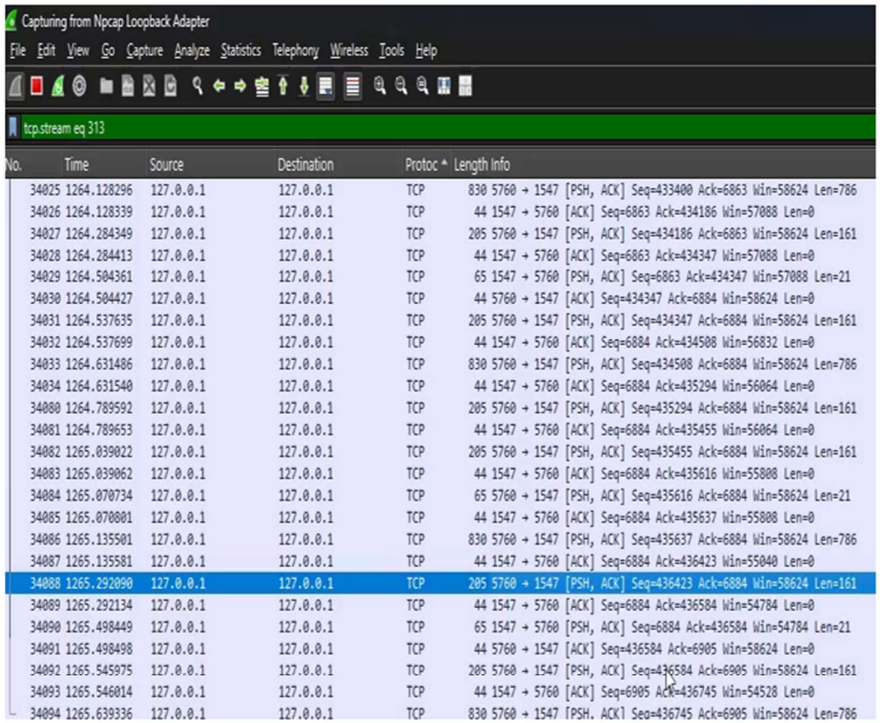


Рис. 2.13 – Захоплення трафіку у Wireshark на фізичному інтерфейсі

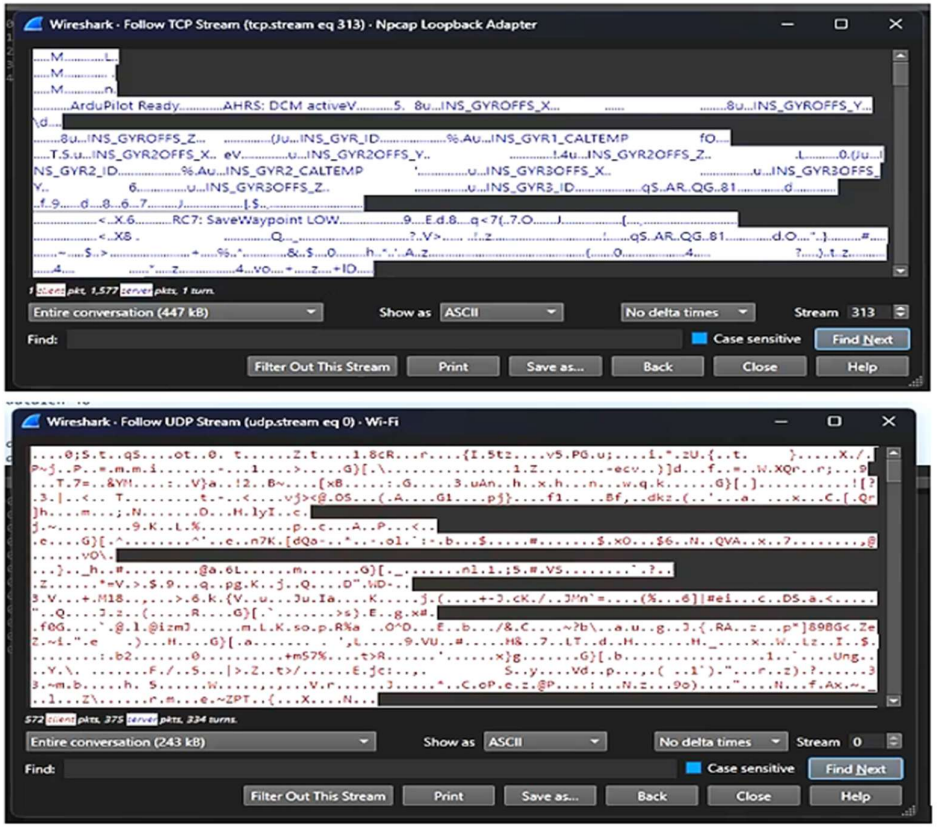


Рис. 2.15 – Зіставлення відкритого та зашифрованого трафіку на основі отриманого дампу зі Wireshark

Оцінка ефективності

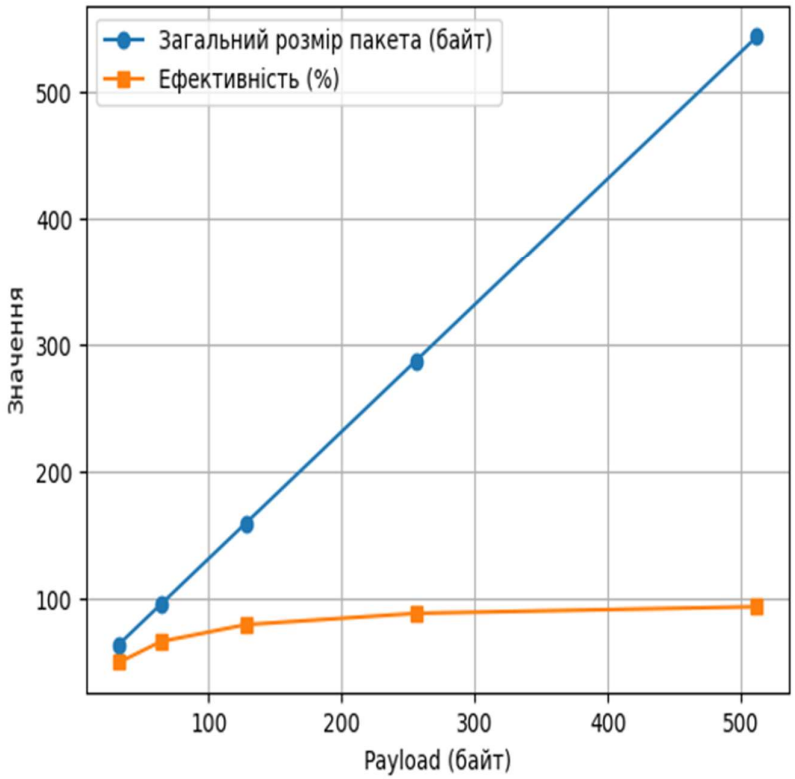


Рис. 3.1 – Вплив криптографічного оверхеду на розмір пакета

Табл. 3.1 – Вплив інкапсуляції на розмір пакета та ефективність каналу

Payload (байт)	Overhead (байт)	Загальний розмір	Збільшення	Ефективність
32	32	64	100%	50%
64	32	96	50%	66.7%
128	32	160	25%	80%
256	32	288	12.5%	88.9%
512	32	544	6.25%	94.1%

Табл. 3.2 – Порівняння відкритого каналу, OpenVPN та WireGuard для БПЛА

Критерій	Відкритий канал	OpenVPN	WireGuard
Захист даних	Низький	Високий	Високий
Затримка	Мінімальна	Середня / вища	Низька
Навантаження CPU	Низьке	Високе	Низьке
Складність налаштування	Низька	Висока	Низька
Придатність до FPV	Висока, але ризикована	Середня	Висока
Стійкість до перехоплення	Низька	Висока	Висока
Робота на слабких ARM	Висока	Середня	Висока
Рекомендація для БПЛА	Небажано	Допустимо	Оптимально

Методи покращення системи

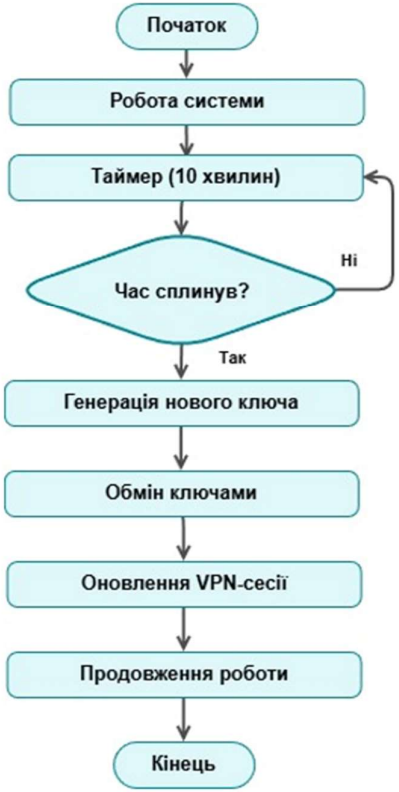


Рис. 3.5 – Алгоритм динамічної зміни ключів Key Rotation



Рис. 3.6 – Захист каналу зв'язку від РЕБ шляхом маскування трафіку

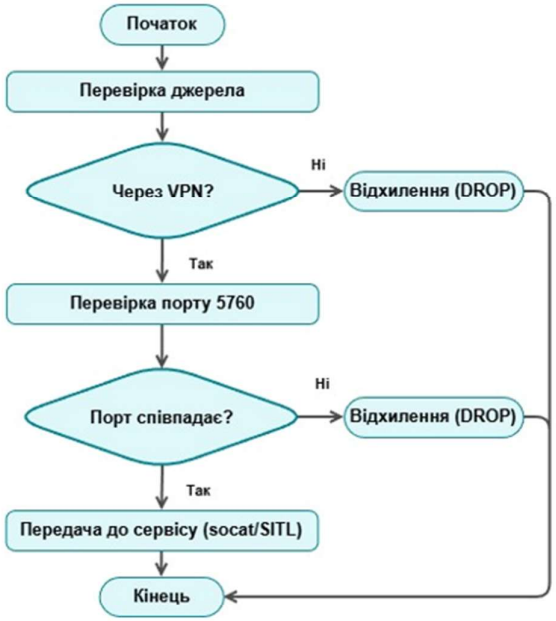


Рис. 3.3 – Відсікання небажаних пакетів на рівні VPN-шлюзу

Практичне впровадження



Рис. 3.4 – Використання "companion computer" у складі БПЛА

Висновки

У роботі розроблено емульоване середовище захищеного керування БПЛА, яке відтворює взаємодію між наземною станцією та бортовим вузлом без використання реального апарата. Реалізовано VPN-тунелювання на базі WireGuard, механізми маршрутизації, фільтрації та ізоляції прикладного рівня, що забезпечують конфіденційність передавання даних і обмеження несанкціонованого доступу.

Результати тестування підтвердили коректність роботи системи та ефективність захисту: у перехопленому трафіку доступні лише зашифровані пакети без можливості аналізу вмісту. Встановлено, що криптографічні накладні витрати є помірними і не впливають критично на роботу, а використаний підхід забезпечує баланс між безпекою та продуктивністю і має потенціал для подальшого впровадження.

Результати впровадження

1. Груша Д. Р., Бученко І. А. Апаратно-програмний комплекс захищеного керування БПЛА з використанням сучасних мережевих технологій. «Сучасний стан та перспективи розвитку IoT» : VII Науково-техн. конф., м. Київ, 20 квітня 2026 р. С.106-109.
2. Груша Д. Р., Бученко І. А. Захищене передавання даних у реальному часі для систем керування БПЛА з використанням тунелювання та сегментації трафіку. «Застосування програмного забезпечення в ІКТ» : VII Всеукр. наук.-техн. конф., м. Київ, 23 квітня 2026 р. (подано до публікації).