УДК 621.372(075) **DOI:10.30837/rt.2025.3.222.16**

Д.Г. ФОКІН, М.О. ЄВДОКИМЕНКО, д-р техн. наук

АНАЛІЗ МЕТОДІВ ПРОТОКОЛЬНОЇ СТЕГАНОГРАФІЇ В ПРОГРАМНО-КОНФІГУРОВАНИХ МЕРЕЖАХ

Вступ

Приховані канали зв’язку, що забезпечують передачу даних шляхом маскування їх під легітимний мережний трафік, становлять серйозну загрозу для цілісності, конфіденційності та контролю інформаційних систем. На відміну від традиційних методів витоку інформації, ці канали функціонують без потреби в ескалації привілеїв або ініціації користувачем підозрілої активності, оскільки ґрунтуються на легітимній поведінці мережних протоколів або експлуатації часових характеристик передачі даних.

З огляду на стрімке впровадження програмно-конфігурованих мереж (Software-Defined Network, SDN), проблема прихованих каналів набуває нових вимірів. Характерні ознаки SDN – централізоване керування, програмованість логіки маршрутизації та чітке розділення площини управління й площини передачі – створюють не лише додаткові вектори загроз, але й потенційні засоби для протидії. З одного боку, гнучкість SDN-архітектури може бути використана зловмисниками для побудови стійких і малопомітних прихованих каналів, зокрема шляхом маніпуляції заголовками протоколів, міжпротокольної взаємодії або зміни порядку пакетів. З іншого боку, глобальна видимість трафіку, притаманна SDN-контролерам, відкриває перспективи для впровадження ефективних механізмів централізованого моніторингу, виявлення аномалій і адаптивного реагування.

Мережна стеганографія в контексті SDN розглядається як потужний інструмент для реалізації прихованого управління зловмисним програмним забезпеченням, ексфільтрації конфіденційних даних, обходу систем контролю трафіку та здійснення координації дій у межах складних атак. Її застосування дозволяє підтримувати зв'язок із скомпрометованими вузлами навіть у сегментованих або ізольованих середовищах, мінімізуючи ризик виявлення завдяки використанню дозволених протоколів і портів.

Актуальність дослідження методів протокольної стеганографії в SDN-середовищах визначається як еволюцією атакуючих технік, що адаптуються до сучасних архітектур, так і обмеженнями традиційних засобів стегоаналізу, які здебільшого орієнтовані на класичні мережеві моделі. Відсутність комплексних підходів до виявлення багатошарових і динамічних прихованих каналів у SDN залишається суттєвою прогалиною у сфері інформаційної безпеки.

У роботі проведено систематизацію сучасних підходів до реалізації протокольної стеганографії в SDN, а також здійснено порівняльний аналіз методів їх виявлення й нейтралізації з урахуванням особливостей архітектури. Такий підхід дозволяє не лише узагальнити наявні дослідження, а й виокремити вразливі елементи інфраструктури, що потребують посиленої уваги при проєктуванні захищених мереж на базі SDN.

Аналіз методів стеганографії в програмно-конфігурованих мережах

Традиційно приховані канали класифікуються на основі принципу їхньої роботи – як канали на основі зберігання (storage-based channels) та канали на основі синхронізації (timing-based channels). Такий поділ відображає механізм передачі стеганограм між відправником і отримувачем: перші маніпулюють значеннями полів протоколів, тоді як інші використовують часові характеристики трафіку, тобто замість того, щоб вбудувати дані в заголовки пакетів або корисне навантаження, кодують інформацію за допомогою затримки пакетів або швидкості передачі пакетів. Проте з погляду практичної реалізації в мережах доцільнішим є підхід, що враховує рівень мережної моделі, для якого розроблено конкретний метод.

У цьому огляді класифікацію методів стеганографії здійснено відповідно до мережних рівнів, у межах яких вони реалізуються (канальний, мережний, транспортний, тощо).

Між непомітністю та пропускною здатністю прихованого каналу, як правило, існує обернена залежність: високопродуктивні канали частіше генерують аномальний трафік, що підвищує ймовірність виявлення, тоді як малошвидкісні канали забезпечують вищий рівень маскування у фоновому трафіку. У середовищі програмно-конфігурованих мереж (SDN) захисники можуть використовувати глобальне бачення мережі, яким володіє контролер, для ідентифікації нетипових шаблонів у поведінці трафіку. Наприклад, OpenFlow-додатки здатні виявляти аномальні значення Flow Label або присутність ненульових зарезервованих полів, які зазвичай не використовуються у легітимному трафіку.

Водночас централізований характер SDN створює нові можливості і для зловмисників. Завдяки централізованому контролю можливо координувати складні сценарії прихованого зв'язку які дозволяють обходити обмеження традиційного фільтрування та сегментації мережі.

Особливий інтерес становить міжпротокольна стеганографія (inter-protocol steganography) [1], що використовує взаємодію між кількома протоколами для організації стійких і важковиявних каналів. У таких підходах одночасно можуть застосовуватись різні типи каналів – наприклад, канал на основі синхронізації у періоди низького навантаження на мережу, і канал на основі заголовків у моменти інтенсивного трафіку, що забезпечує ефективну адаптацію до змін умов. Централізоване управління в SDN може слугувати як інструментом для виявлення подібної активності, так і, у випадку компрометації контролера, засобом для реалізації високорівневої координації між прихованими компонентами атак.

З огляду на динамічну природу сучасних мереж і широкі можливості SDN, дослідження методів стеганографії повинні враховувати не лише окремі протоколи, а й їхню взаємодію, змінність поведінки трафіку, а також потенціал як для приховування, так і для виявлення таких каналів у масштабах всієї мережної інфраструктури.

Зважаючи на гетерогенність мережевих середовищ та варіативність підходів до прихованої передачі даних, доцільним є розгляд методів стеганографії в контексті окремих рівнів моделі OSI. Такий підхід дозволяє точніше визначити вразливості, пов'язані з кожним рівнем, оцінити вплив прихованого каналу на легітимний трафік, а також підібрати релевантні методи детектування та протидії. Нижче подано структурований огляд методів стеганографії, згрупованих за рівнями OSI-моделі, що дозволяє краще зрозуміти їхню специфіку, можливості застосування та ризики для безпеки SDN-інфраструктур.

Фізичний та канальний рівні

На фізичному та канальному рівнях моделі OSI стеганографія переважно використовує можливості протоколів локальних мереж, зокрема Ethernet та IEEE 802.11. Такі методи є особливо актуальними в локальних сегментах інфраструктур SDN, які здебільшого функціонують на базі Ethernet-комутаторів.

У роботі [1] описано метод PadSteg – міжпротокольний стеганографічний підхід, що використовує поле доповнення в Ethernet-кадрах для прихованої передачі інформації. За стандартом Ethernet кожен кадр повинен мати щонайменше 64 байти, з яких 46 – це мінімальний розмір поля даних. Якщо передаються менші обсяги даних, кадр автоматично доповнюється нульовими байтами. Однак уразливість під назвою Etherleak полягає в тому, що не всі реалізації драйверів коректно обнуляють ці додаткові байти – вони можуть містити залишкову пам'ять. PadSteg цілеспрямовано створює короткі Ethernet-кадри, у доповнення яких вбудовуються приховані дані, при цьому використовується протокол ARP разом з транспортними протоколами, як-от TCP або ICMP. У найкращому випадку метод забезпечує до 45 байт на кадр, однак для збереження непомітності використовується лише природне доповнення. В умовах реального навантаження пропускна здатність методу становить у середньому 32 біт/с, що є досить ефективним показником для стеганографії.

У статті [2] представлено прихований канал на основі зберігання, який не передбачає безпосереднього трафіку між учасниками. Один з вузлів мережі використовується як так званий "Dead Drop" – проміжний носій повідомлення. Відправник вбудовує дані в ARP-відповіді, які потрапляють до кешу, адрес цього вузла, а отримувач зчитує їх за допомогою SNMP. Через відсутність прямої комунікації канал дуже важко виявити, однак його пропускна здатність вкрай низька – у межах кількох бітів на хвилину.

В роботі [3] описано реалізацію внутрішньосмугового (in-band) прихованого каналу шляхом маніпуляції протоколом LLDP, який використовується в SDN для визначення топології мережі. Зловмисники підробляють LLDP-пакети, змушуючи SDN-контролер помилково вважати, що два віддалені OpenFlow-комутатори безпосередньо з'єднані. У такий спосіб створюється логічний "тунель" між вузлами, який використовується для передавання прихованих повідомлень через площину управління. Пропускна здатність такого каналу обмежена – зазвичай лише кілька байтів за один обмін LLDP, однак його цінність полягає у здатності функціонувати поза площиною даних.

У статті [4] описано метод StegoFrameOrder, який реалізує прихований канал на основі синхронізації у безпроводових мережах Wi-Fi. Він ґрунтується на маніпуляції порядком надходження кадрів у мережі з доступом за схемою CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance). Стеганограма кодується через черговість передачі кадрів між різними вузлами: наприклад, першим передає "0", другим – "1". Перевага методу полягає у відсутності змін до самих кадрів; використовується лише часовий аспект. Середня пропускна здатність становить до 10 біт/с у типовому середовищі IEEE 802.11.

Іншим важливим прикладом є метод HICCUPS (Hidden Communication System for Corrupted Networks), описаний у [5]. Він реалізує прихований канал шляхом наміру пошкоджувати CRC у кадрах. У звичайних умовах такі кадри відкидаються, однак у HICCUPS попередньо налаштовані вузли вважають їх сигналом прихованого повідомлення, інтерпретуючи тип помилки як біт "0" або "1". Оскільки в безпроводових мережах кадри часто зазнають ушкоджень через перешкоди, невелике зростання частоти "помилок" може бути непоміченим. Проте через високу ймовірність втрати та необхідність обмеження кількості навмисно пошкоджених кадрів, пропускна здатність становить лише кілька біт/с.

Усі згадані методи демонструють, що навіть на нижчих рівнях OSI-моделі – фізичному та канальному – існує широкий спектр технік прихованої передачі даних, які можуть бути реалізовані в умовах SDN. Особливої уваги потребують протоколи Ethernet, ARP, LLDP, а також механізми доступу до середовища у Wi-Fi, що можуть бути використані як носії прихованих повідомлень без зміни змісту легітимного трафіку.

Мережний рівень

Методи мережного рівня стеганографії приховують дані у полях заголовків протоколів IP, зокрема IPv4 та IPv6, або використовують службові протоколи цього рівня, як-от ICMP. Ці методи особливо актуальні в площині даних SDN, зокрема при передачі між маршрутизаторами та комутаторами, що оперують IP-трафіком.

У класичній роботі [6] запропоновано один із перших відомих методів стеганографії в стеку TCP/IP, що використовує 16-бітне поле Identification в заголовку IPv4. Це поле, призначене для фрагментації, може залишатися незмінним або містити довільне значення, якщо пакет не фрагментується. Таким чином, зловмисник може вбудовувати повідомлення у вигляді бітів у кожен пакет. Пропускна здатність методу теоретично досягає 1600 біт/с за умов високої інтенсивності трафіку. Простота реалізації та висока продуктивність роблять його актуальним і сьогодні, хоча відстеження повторюваних значень поля може дати змогу виявити канал за допомогою систем виявлення вторгнень.

Інша класична праця [7] описує використання зарезервованих або необов'язкових бітів у протоколах IPv4. Наприклад, стеганографічні біти можуть бути закодовані у зарезервованому прапорці або у полі ToS/DSCP, яке часто не використовується в типових налаштуваннях

QoS. Пропускна здатність тут становить 1–3 біт на пакет, але такі канали мають високу непомітність, якщо політики QoS не змінюють значення полів при транзиті.

Опції IPv4, такі як Record Route та Timestamp, також демонструють потенціал для прихованої передачі інформації. У роботі [8] автори кодують байти у фальсифікованих IP-адресах маршруту, а в [9] – у полі переповнення опції Timestamp. Обидва методи мають обмеження у вигляді максимального розміру опцій IPv4 (до 40 байт), але забезпечують десятки або навіть сотні біт/пакет при правильній реалізації.

ICMP-повідомлення, зокрема Echo-запити (ping), давно використовуються як транспортний механізм для стеганографії. Утиліта Loki (1996) і сучасні приклади, як Pingback (2021) [10], демонструють вбудовування повідомлень у поля даних, Identifier або Sequence. Частота ping-пакетів визначає пропускну здатність, яка може досягати 800 біт/с при помірному трафіку.

Для IPv6 характерні нові поля, що відкривають додаткові можливості. 8-бітне поле Traffic Class є аналогом ToS/DSCP в IPv4 і може кодувати біти, якщо мережа не застосовує активні політики QoS. Поле Flow Label (20 біт), як зазначено у [11], прямо визначене як потенційне джерело прихованих каналів. У роботі [12] показано, що за умови достатнього обсягу трафіку і належної рандомізації значень, можна досягти тисяч біт/с без очевидного порушення легітимності трафіку.

Також у [12] розглядається використання опцій-доповнень в розширених заголовках IPv6. Пропускна здатність таких каналів теоретично сягає 2048 біт/пакет, хоча на практиці застосовуються значно менші обсяги, щоби уникнути виявлення засобами глибокого аналізу пакетів.

У роботі [13] описано стеганографічний метод на основі варіативності IP-адреси. Змінюючи хост-частину IPv6-адреси, можна кодувати дані, які зчитуються за допомогою аналізу відправлених адрес.

Крім цього, на мережному рівні можливе створення прихованих каналів на основі синхронізації. Наприклад, у [14] Він модулює секретні дані в трафік VoNR (Voice over New Radio), змінюючи міжпакетні затримки. Ці модифікації ретельно розроблені, щоб імітувати статистичну поведінку (зокрема, кумулятивну функцію розподілу) природних тремтінь IPD у реальному трафіку VoNR.

Сукупно методи мережного рівня охоплюють широкий спектр векторів стеганографії – від модифікації зарезервованих бітів до складних схем зміни IP-адрес і затримок між пакетами. У контексті SDN їхня ефективність або вразливість визначається архітектурною здатністю до централізованого моніторингу, обробки потоків та динамічної зміни політик обробки трафіку.

Транспортний рівень

Методи транспортного рівня зосереджуються на протоколах TCP, UDP, SCTP та інших. Стеганографія на цьому рівні реалізується шляхом модифікації полів заголовків транспортних протоколів або шляхом маніпулювання характерними особливостями їх функціонування. У SDN, зокрема в площині управління, протокол OpenFlow функціонує поверх TCP, а трафік площини даних зазвичай реалізується через TCP або UDP. Таким чином, приховані канали цього рівня найчастіше проявляються у трафіку площини даних.

У класичній роботі [6] запропоновано використання поля Initial Sequence Number (ISN) в TCP як носія прихованого повідомлення. При встановленні TCP-з’єднання обидві сторони обмінюються 32-бітовими випадковими початковими номерами. Прихований відправник може ініціювати TCP-сеанс, задавши ISN таким чином, щоб він кодував певну послідовність бітів. Одержувач, який контролює трафік SYN (synchronize), вилучає значення ISN і реконструює приховане повідомлення. Пропускна здатність методу становить до 16 біт на з’єднання, що за умови одного з’єднання на секунду дає лише 16 біт/с. Цей підхід ефективний за умови великої кількості короткоживучих з’єднань.

У тій же роботі розглянуто використання поля Acknowledgement Number TCP. Зловмисник може кодувати дані в ACK-пакетах, змінюючи 32-бітове значення підтвердження відповідно до бітової послідовності повідомлення. Такий метод вимагає контролю над формуванням трафіку та скоординованості між відправником і одержувачем, і найкраще підходить для штучно створених з'єднань. Теоретично, кожен пакет може нести до 32 біт прихованої інформації, але на практиці це може викликати підозру при перехресній перевірці з очікуваними значеннями підтвердження.

Методи, що використовують опції TCP, також заслуговують на увагу. Наприклад, у 32-бітному полі опції Timestamp можливо кодувати дані в молодших бітах значення часової мітки. Інший підхід – використання опції No-Operation (NOP), яка зазвичай застосовується для вирівнювання, як прихованого бітового сигналу. У роботі [15] продемонстровано, що комбінація змін у кількох полях TCP та IP, включаючи опції, може забезпечити приховану передачу до 31 біта на пакет. Однак це граничний випадок, що значно ускладнює непомітність каналу.

У статті [16] запропоновано метод приховання даних у змінюваній довжині UDP-пакетів. Оскільки довжина UDP-пакету визначається як сума 8 байт заголовка і довжини навантаження, контрольований відправник може кодувати біти, трохи змінюючи довжину пакета (на 1 байт для представлення "0" або "1"). В експериментальних умовах досягнуто пропускної здатності до 456 біт/с. Цей метод залишається малопомітним, якщо варіації довжини інтегруються у природний розподіл розмірів пакетів і використовуються з випадковими інтервалами.

Протокол SCTP, хоча й не широко розповсюджений, має низку унікальних характеристик, що роблять його перспективним для стеганографії. У роботі [17] розглянуто використання полів Initiate Tag, Payload Protocol Identifier та Heartbeat Info для прихованої передачі. Крім того, можливості multistreaming (багатопотоковості), multihoming (підтримки кількох маршрутів) та часткової надійності в SCTP дозволяють створювати складні сценарії кодування, включно з використанням повторних передач або зміни маршруту як сигналу. У випадку SDN ці можливості є більш теоретичними, оскільки більшість середовищ не використовує SCTP, і будь-яка його поява може одразу викликати підозру, за винятком телекомунікаційних систем, де протокол є стандартом.

Таким чином, методи стеганографії транспортного рівня охоплюють як внутрішньопротокольні канали в TCP, так і канали на основі особливостей UDP або специфічних властивостей SCTP. Їх пропускна здатність значно варіюється залежно від протоколу та методики, але всі вони мають спільну ознаку – вони потребують ретельного налаштування для уникнення виявлення, зокрема у контрольованих середовищах SDN, де доступний централізований моніторинг трафіку.

Сеансовий рівень

Сеансовий рівень відповідає за встановлення, підтримку та контроль сеансів між кінцевими точками. На практиці багато функцій сеансового рівня обробляються протоколами вищого рівня, але деякі протоколи та механізми можна розглядати як такі, що працюють на даному рівні. У цьому контексті інформація може бути прихована у протоколах ініціації сеансу та сигнальних повідомленнях, а також у маніпулюванні поведінкою сеансу.

Session Initiation Protocol (SIP) і супровідний Session Description Protocol (SDP) є класичними прикладами протоколів сеансового рівня, які можуть передавати приховані дані. У статті [18] продемонстрували, що SIP-повідомлення, які використовуються для встановлення VoIP-дзвінків, можуть бути використані для створення прихованих каналів. Вони показали, що багато полів, маркерів і параметрів у SIP/SDP мають гнучкі або невизначені формати, які можна використовувати для вбудовування інформації. Наприклад, необов'язкові маркери, як-от ідентифікатори SIP «tag» і «branch», які мають бути випадковими; поля заголовка, які не мають суворих обмежень по довжині, можуть бути розширені. Такі

методи забезпечують пропускну здатність каналу від низької до помірної – в експериментах від кількох десятків до кількох сотень біт на секунду.

Окрім конкретних протоколів, сеансовий рівень охоплює керування з'єднаннями між кінцевими точками. Приховані канали можна побудувати, використовуючи початок, синхронізацію та завершення сеансу. Наприклад, шляхом навмисного ініціювання та розривання з'єднань у шаблоні (або введення певних затримок між подіями сеансу) можна передати повідомлення. В статті [19] показали, що навіть без будь-якого контролю над контролером або комутаторами SDN хости можуть створити прихований канал синхронізації, ретельно організовуючи пакетні спалахи (packet bursts), які запускають певну поведінку контролера. По суті, сама взаємодія SDN з площиною керування стає частиною прихованого каналу – наприклад, час, потрібний контролеру для встановлення правила потоку або відповіді на нове з'єднання, може кодувати біти. Ця техніка використовувала те що SDN контролер бачить нові запити на сеанс, щоб створити канал зв'язку, невидимий для звичайного моніторінгу трафіка.

Хоча методи стеганографії на сеансовому рівні є порівняно менш дослідженими у порівнянні з мережним або транспортним рівнями, існуючі дослідження формують концептуальну основу того, що будь-який протокол, який керує сеансом або діалогом може бути використаний для передачі інформації. Таким чином, сеансовий рівень пропонує приховані канали через метадані та динаміку сеансів – від метаданих викликів VoIP до синхронізації налаштувань потоку SDN.

Рівень представлення

На цьому рівні відбувається шифрування та перетворення форматів даних (наприклад, TLS забезпечує конфіденційність і цілісність переданої інформації). У статті [20] автори проаналізували сім методів прихованого каналу в TLS, зокрема приховану передачу через довжину запису, вектор ініціалізації та поле типу вмісту. Експерименти підтвердили ефективність цих підходів: найвищу пропускну здатність показав метод маніпулювання вектором ініціалізації. Водночас стандартні системи виявлення вторгнень (Snort, Bro/Zeek, Suricata) не здатні «з коробки» розпізнати такі TLS-канали, що свідчить про їхню непомітність [20].

Робота [21] демонструють розширення стеганографії рівня представлення на сучасні протоколи. Так, у протоколі QUIC (що лежить в основі HTTP/3) зловмисники можуть використовувати його поля для укриття даних. Зокрема, показано можливість використання необов'язкового біта spin bit у заголовку QUIC для сигналізації прихованих біт між клієнтом і сервером. Хоча пропускна здатність такого каналу невелика (передається лише 1 біт інформації на кожен пакет, що дає десятки біт/с), його трафік майже не відрізнити від легітимного HTTP/3-обміну, що ускладнює виявлення. Інший підхід використовує механізм міграції з'єднання в QUIC: функцію Server Preferred Address можна експлуатувати для прихованої передачі даних під виглядом перемикання сервером адреси підключення. Дослідники реалізували такий канал (інструмент QUIC-Exfil) і показали, що сучасні міжмережеві екрани не здатні відрізнити фальшиву міграцію від легітимної. Перевага методів на основі QUIC – використання зашифрованих заголовків і динаміки протоколу, що робить їх малопомітними для засобів DPI. Недоліком є складність реалізації та залежність від підтримки протоколом певних опцій (наприклад, наявності spin bit) і можливість контрзаходів у майбутніх версіях протоколів.

Прикладний рівень та площина управління SDN

Починаючи з прикладного рівня моделі OSI відкриваються широкі можливості для реалізації прихованих каналів через зловживання функціональністю високорівневих протоколів. У середовищі SDN до цього рівня також відносяться протоколи управління мережею, зокрема OpenFlow. Стеганографічні методи на прикладному рівні охоплюють як маніпуляції структурованими протоколами на кшталт HTTP або DNS, так і специфічні для SDN підходи, пов'язані з контролером і комутаторами.

В роботі [22] описано вразливість у процедурі встановлення з'єднання між OpenFlow-комутатором та контролером, яка дозволяє створити прихований канал у повідомленні Features Reply. Зокрема, Datapath ID (унікальний 64-бітовий ідентифікатор пристрою) або інші поля відповіді, як-от MAC-адреси портів чи імена інтерфейсів, можуть бути використані для кодування повідомлень. Оскільки контролер OpenFlow за замовчуванням довіряє отриманим значенням без обов'язкової перевірки їх достовірності, ці поля можуть бути використані зловмисником-комутатором для прихованої передачі інформації контролеру. Пропускна здатність обмежується обсягом інформації, що вміщується в одному Features Reply (кілька десятків байтів), але цього достатньо для передачі конфіденційних параметрів або тригерів.

Дослідження [23] представляє прихований канал синхронізації, реалізований через архітектуру SDN. Автори демонструють, як два ізольовані OpenFlow-комутатори можуть обмінюватися даними через централізований контролер без прямого канального з'єднання. Зловмисник модифікує поведінку одного з комутаторів таким чином, щоб викликати у контролера певну відповідь (наприклад, переналаштування маршруту або відправлення Barrier Request). Інший комутатор, що спостерігає за інтенсивністю або часом реакції контролера, розшифровує приховані біти. У прототипі, реалізованому за допомогою операційної системи для SDN-контролера Open vSwitch та ONOS (Open Network Operating System), досягнуто пропускної здатності до 20 біт/с. Цей канал є типовим каналом синхронізації, де не вміст, а порядок і час повідомлень використовуються як носії.

Ще один значний напрямок прикладного рівня – HTTP та DNS-стеганографія. Протокол HTTP, широко використовуваний у інтерфейсі прикладного рівня SDN-контролерів Northbound API, може бути модифікований для прихованої передачі інформації. В роботі [24] розглянуто приховування даних в полях заголовків, використовуючи допустимі але непомітні модифікації. Наприклад, варіювання пропусків (табуляція/пробіл) у заголовку може кодувати біти 0/1, так само як і зміна регістру літер. У [25] розглядаються методи кодування секретних бітів шляхом вставлення даних у параметри URL та інші поля HTTP-запитів. Крім того, запропоновано метод прихованої передачі інформації через контрольовані часові інтервали між HTTP-запитами, що утворює канал синхронізації.

Щодо DNS, в [26] продемонстровано кодування повідомлення в іменах доменів, що запитуються. Секретні дані розбиваються на фрагменти та вставляються у вигляді довгих піддоменів при DNS-запитах. Також автори розглядають вбудовування інформації в поля ресурсних записів – наприклад, TXT-записи здатні нести до ~64 КБ текстових даних, що дозволяє передавати великі повідомлення у відповідях DNS. Також можуть застосовуватися поля адресних записів. Поле Time-To-Live у DNS-відповіді, що зазвичай визначає час кешування, теж може слугувати каналом. Прихований відправник може варіювати значення TTL у відповіді за певною схемою, щоб закодувати біти [27]. В статті [28] автори розглядають канал синхронізації. Попередньо домовившись про інтерпретацію порядку відправник і отримувач здатні передавати повідомлення, просто варіюючи черговість DNS-запитів.

Отже, стеганографія прикладного рівня в SDN є ефективною завдяки широкому спектру доступних механізмів і слабкому моніторингу окремих елементів протоколів, однак якісний аналіз трафіку та машинне навчання здатні значно ускладнити приховану передачу даних.

Порівняльний аналіз методів стеганографії

З метою систематизації аналізованих методів протокольної стеганографії в табл. 1 продемонстровано ключові характеристики, зокрема пропускну здатність, рівень непомітності, технічні особливості впровадження та основні недоліки кожного методу. Такий підхід дозволяє не лише порівняти методи за ефективністю, а й виявити найбільш уразливі рівні мережної архітектури до прихованих каналів у середовищі SDN. Це особливо важливо для вибору пріоритетних напрямів подальших досліджень у сфері виявлення й нейтралізації стеганографічних загроз.

Таблиця 1

Порівняльні характеристики методів стеганографії в SDN за рівнями моделі OSI

Метод	Пропускна здатність	Особливість	Недолік	Непомітність
Фізичний/Канальний рівень				
StegoFrameOrder	~10 біт/с	Немає зміни кадрів	Потрібна координація між вузлами	висока
HICCUPS	кілька біт/с	Маскування під звичайні помилки	Висока ймовірність втрати кадрів	висока
PadSteg	~32 біт/с (в середньому)	Використання Etherleak доповнення	Потребує вразливості NIC Etherleak	висока
Dead Drop (ARP)	лічені біти/хв	Немає прямої взаємодії між відправником і отримувачем	Потрібні контроль над проміжним вузлом і узгоджені дії сторін	дуже висока
LLDP Tunnel	кілька байтів/обмін	Канал на основі площини управління	Маніпуляція топологією	середня
Мережний рівень				
IPv4 ID Field	до ~1600 біт/с	Поле заголовка IPv4 високої ємності	Повторювані значення ID	середня
IPv4 Reserved/ToS Bits	~1–3 біт на пакет	Вбудовування у невикористовувані біти заголовків	Політики QoS можуть змінювати або очищати біти з інформацією	висока
IPv4 Record Route Option	десятки–сотні біт/пакет	Опція рідко використовується	Великий заголовок запускає DPI	середня
IPv4 Timestamp Overflow	десятки–сотні біт/пакет	Вбудовування в поле переповнення	Обмежений розмір опціонального поля	середня
ICMP Echo (Ping)	до ~800 біт/с	Використання частих ping-пакетів	Вразливість до аналізу ping-трафіку	середня
IPv6 Traffic Class	8 біт/пакет	Непомітний якщо не використовується QoS	Вразливість до повторного маркування	висока
IPv6 Flow Label	до кількох тисяч біт/с	Використання 20-бітового поля Flow Label	Потрібен значний трафік і ретельна рандомізація	висока
IPv6 Extension Headers	до ~2048 біт/пакет	Приховування великих обсягів інформації	Великий розмір опцій легко помічається	середня
IPv6 Address Variation	кілька біт/с (залежно від частоти)	Кодування через хост-частину	Потреба часто змінювати IP-адресу	середня
VoNR IPD Timing	низька (залежить від трафіку)	Маскування природніх коливань затримок VoNR-трафіку	Вимагає точної імітації статистичних характеристик VoNR-трафіку	висока
Транспортний рівень				
TCP ISN	~16 біт/с (при 1 SYN/с)	Використовує ISN під час встановлення з’єднання	Вимагає великої кількості коротких з’єднань	середня
TCP ACK Number	до 32 біт/пакет	Велика ємність на пакет	Дуже помітний, неправильний ACK руйнує сеанс	низька
TCP Timestamp	кілька біт/пакет	Вбудовування в Timestamp	Аномальна динаміка Timestamp викликає підозру	середня
TCP NOP	~1 біт/пакет	Використання опції No-Operation як носія біта	Додаткові NOP у заголовку роблять структуру опцій TCP незвичною	середня
UDP Length Modulation	до ~456 біт/с	Кодування бітів шляхом зміни загальної довжини UDP-пакета на 1 байт	Потребує повного контролю над формуванням UDP-пакетів	висока
SCTP (Initiate Tag, PPI, Heartbeat тощо)	низька (обмежена випадками використання SCTP)	Використання унікальних полів і можливостей SCTP	SCTP трафік є підозрілим сам по собі	низька

Сеансовий рівень				
SIP/SDP Fields	десятки–сотні біт/с	Гнучкі невикористані поля заголовка	Можливе втручання SIP-інфраструктури	середня
Session Timing (SDN)	кілька біт/с	На основі факту початку/завершення сеансу	Необхідна точна синхронізація та попередня домовленість про інтерпретацію часових інтервалів	висока
Рівень представлення				
TLS Record Length	до сотень біт/с	Зміна довжини TLS-записів	Потребує контролю над фрагментацією записів	висока
TLS Initialization Vector	найвища серед TLS-методів (до сотень біт/с)	Маніпуляція вектором ініціалізації	Вимагає втручання в криптографічний процес TLS	висока
TLS Content Type Field	~1 біт/запис	Використання поля Content Type як носія	Невірне або неочікуване значення Content Type може не підтримуватися отримувачем	середня
QUIC Spin Bit	десятки біт/с	Передача даних через необов'язковий Spin Bit у заголовку QUIC (HTTP/3)	Залежність від реалізації QUIC (наявність та незаблокованість Spin Bit у конкретній мережі)	дуже висока
QUIC Connection Migration	кілька байтів на сеанс	Кодування в Server Preferred Address	Складність реалізації та залежність від підтримки цієї функції	висока
Прикладний рівень та площина управління SDN				
OpenFlow Features Reply	десятки байт на з'єднання	Комутатор кодує дані в полях повідомлення Features Reply	Одноразовий канал вимагає компрометації мережевого обладнання	висока
SDN Inter-switch Timing Channel	~20 біт/с	Два ізольовані OpenFlow-комутатори обмінюються інформацією через спільний SDN-контролер	Покладається на передбачувану реакцію контролера, яка може змінюватися	середня
HTTP Headers	кілька біт на запит	Зміни пробілів та регістру кодують біти	Загроза фільтрації/нормалізації проміжними проксі	висока
HTTP Parameters	кілька байт на запит	Вбудовування у поля запитів або URL	Незвично закодовані параметри можуть виділятися	середня
HTTP Timing Channel	~1–2 біт/с (дуже низька)	Кодування бітів в інтервалах між запитами	Потрібна точна синхронізація між сторонами	висока
DNS TXT Records	до ~64 КБ/відповідь (дуже висока)	Вбудовування в TXT ресурс	Простий для виявлення	низька
DNS TTL	кілька біт/запит	Вбудовування в значення TTL	Повторювані TTL зміни можуть виділятися	середня
DNS Query Ordering	біт/хвилини (дуже низька)	Кодування бітів в порядок виконання DNS-запитів	Для передачі даних потрібна значна кількість упорядкованих запитів	висока

Порівняльний аналіз методів протокольної стеганографії, узагальнений у табл. 1, дозволяє зробити низку важливих висновків. Зокрема, на фізичному та канальному рівнях методи прихованої передачі завдяки мінімальному впливу на легітимний трафік забезпечують дуже високий рівень непомітності, проте мають надзвичайно низьку пропускну здатність і часто вимагають специфічних умов.

Натомість на мережному рівні вбудовування даних у поля заголовків IP або службові протоколи дозволяє суттєво підвищити пропускну здатність прихованого каналу при збереженні помірної чи навіть високої непомітності за обережної реалізації. Водночас нетипові

шаблони (повторювані ідентифікатори, надмірні опції тощо) на цьому рівні можуть привертати увагу засобів глибокого аналізу трафіку або бути зміненими мережними політиками, особливо з огляду на централізований контроль у SDN.

Методи транспортного рівня здатні забезпечувати як низьку, так і значну пропускну здатність, але всі вони потребують ретельного налаштування, щоб зміни не порушили нормальний перебіг з'єднань і не привертали уваги систем моніторингу. Вбудовування даних у керівні поля TCP (номери послідовності/підтвердження, опції) або незначна модифікація довжини UDP-пакетів залишається непомітною в межах типової динаміки трафіку, проте грубе втручання (некоректні ACK, надлишкові опції тощо) легко викривається; більш того, у SDN централізований аналіз трафіку полегшує виявлення таких аномалій.

Сеансовий рівень також може слугувати для прихованої передачі даних – через метадані з'єднань і синхронізацію подій. Такі канали характеризуються низькою пропускною здатністю, але забезпечують високий рівень непомітності, маскуючись під нормальну поведінку сеансів; утім, вони вимагають точної синхронізації, і будь-які нетипові затримки чи незвичні службові поля можуть бути помічені системами керування сеансами (наприклад, SIP-проксі або SDN-контролером).

На рівні представлення (TLS, QUIC) приховані канали використовують переваги шифрування та гнучких полів протоколів, забезпечуючи дуже високий рівень непомітності при значній пропускній здатності. Наприклад, маніпуляція довжиною TLS-записів або вектором ініціалізації, а також використання опціонального біта spin-bit у QUIC дозволяють передавати сотні біт/с непомітно для стандартних IDS. Втім, реалізація цих методів потребує складного втручання в криптографічні процеси і залежить від специфічних функцій протоколів, що ускладнює впровадження та може бути нейтралізовано оновленнями протоколів.

Нарешті, на прикладному рівні та в площині управління SDN спостерігається найширший спектр стеганографічних технік – від майже непомітних, але низької пропускної здатності (мінімальні варіації заголовків чи часових інтервалів HTTP-запитів) до високопродуктивних, проте легко виявлюваних (приховування значних обсягів даних у DNS або службових повідомленнях SDN). Різноманіття високорівневих протоколів робить такі канали ефективними та часто непоміченими традиційним моніторингом, однак ретельний аналіз трафіку й використання методів машинного навчання дозволяють викрити аномальні шаблони та суттєво обмежити їх.

Висновки

Здійснено комплексний огляд протокольних методів стеганографії з позицій їх релевантності до сучасних мережних архітектур, зокрема SDN. Запропонована класифікація методів за рівнями моделі OSI дозволила систематизувати наявні підходи та встановити залежності між рівнем реалізації протоколу, можливостями прихованої передачі, рівнем непомітності та операційними обмеженнями. Виявлено, що високопродуктивні приховані канали зазвичай генерують аномальні шаблони трафіку, які полегшують їх виявлення, тоді як низькошвидкісні канали забезпечують значно вищий рівень маскування у фоновому потоці. Проведений порівняльний аналіз продемонстрував, які саме рівні мережної моделі є найбільш уразливими до стеганографічних каналів у SDN-середовищі, що важливо для визначення пріоритетів захисту мережі.

Окрему увагу приділено особливостям архітектури SDN, які впливають на розвиток прихованих каналів і методи протидії їм. Глобальна видимість трафіку, яку забезпечує SDN-контролер, відкриває широкі можливості для централізованого моніторингу та виявлення аномалій (нетипового використання полів протоколів, нестандартної динаміки сеансів тощо). Водночас централізований характер управління може бути використаний зловмисниками: у разі компрометації контролера або змови мережевих вузлів стає можливим координувати складні багатокрокові приховані обміни даними, що обходять традиційні механізми фільтрації та сегментації трафіку. Продемонстровано перспективність міжпротокольної стеганогра-

фії, яка завдяки взаємодії кількох протоколів забезпечує підвищену стійкість прихованого каналу та адаптацію до змін мережевих умов. Таким чином, навіть у сегментованих або ізольованих середовищах SDN приховані канали здатні підтримувати нелегітимний зв'язок, використовуючи дозволені протоколи й порти та мінімізуючи ризик виявлення.

Отримані результати вказують на необхідність подальших досліджень, спрямованих на розвиток спеціалізованих засобів стегоаналізу для SDN. Зокрема, перспективним є впровадження централізованих систем виявлення прихованих каналів на рівні контролера, що використовують глобальне бачення мережі й методи машинного навчання для розпізнавання малопомітних аномалій у багатошаровому трафіку. Важливим напрямом є також дослідження міжпротокольних та багатошарових каналів з метою розробки ефективних методів їх нейтралізації до того, як такі техніки набудуть поширення у зловмисників. Таким чином, комплексний підхід, який поєднує врахування вразливостей усіх рівнів мережі, можливості SDN для глобального моніторингу і адаптивний аналіз трафіку, є ключовим для випередження розвитку стеганографічних загроз у майбутніх мережах.

Список літератури:

1. B. Jankowski, W. Mazurczyk, and K. Szczypiorski. PadSteg: Introducing inter-protocol steganography // Telecommunication Systems, preprint, 2011. doi: 10.1007/s11235-011-9616-z.
2. T. Schmidbauer et al. Introducing dead drops to network steganography using ARP-caches and SNMP-walks // Proc. 14th Int. Conf. Availability, Reliability and Security (ARES). 2019. P. 1–10. doi: 10.1145/3339252.3341488.
3. J. Hua, Z. Zhou, and S. Zhong. Flow misleading: Worm-hole attack in software-defined networking via building in-band Covert Channel // IEEE Trans. Inf. Forensics Security. 2021. Vol. 16. P. 1029–1043. doi: 10.1109/TIFS.2020.3013093.
4. K. Sawicki, G. Bieszczad, and Z. Piotrowski. StegoFrameOrder-AC layer covert network channel for wireless IEEE 802.11 networks // Sensors. 2021. Vol. 21, no. 18. P. 6268. doi: 10.3390/s21186268.
5. K. Szczypiorski. HICCUPS: Hidden Communication System for Corrupted Networks // Proceedings of the 10th International Multi-Conference on Advanced Computer Systems (ACS), Międzyzdroje, Poland, Oct. 2003, pp. 31–40.
6. C. H. Rowland. Covert channels in the TCP/IP protocol suite // First Monday, preprint, 1997. doi: 10.5210/fm.v2i5.528.
7. M. Wolf. Covert channels in LAN protocols // Lect. Notes Comput. Sci., 1989, pp. 89–101. doi: 10.1007/3-540-51754-5_33.
8. Z. Trabelsi and I. Jawhar. Covert File Transfer Protocol Based on the IP Record Route Option // Journal of Information Assurance and Security. 2010. Vol. 5, no. 1. P. 64–73.
9. P. Bedi and A. Dua. Network steganography using the overflow field of timestamp option in an IPv4 packet // Procedia Comput. Sci. 2020. Vol. 171. P. 1810–1818. doi: 10.1016/j.procs.2020.04.194.
10. A. Sharma. New Windows 'Pingback' malware uses ICMP for covert communication // BleepingComputer, May 4, 2021. Available: https://www.bleepingcomputer.com/news/security/new-windows-pingback-malware-uses-icmp-for-covert-communication/
11. S. Amante, B. Carpenter, S. Jiang, and J. Rajahalme. IPv6 Flow Label Specification, IETF, RFC 6437, Nov. 2011. Available: https://datatracker.ietf.org/doc/html/rfc6437
12. N. B. Lucena, G. Lewandowski, and S. Chapin. Covert channels in IPv6 // Privacy Enhancing Technologies, G. Danezis and D. Martin, Eds. Heidelberg : Springer, 2006. P. 147–166. doi: 10.1007/11767831_10.
13. M. Bobade and A. Sagar. Survey and design approach of protocol steganography in IPv6 // Int. J. Comput. Appl. 2013. Vol. 69, no. 7. P. 31–34. doi: 10.5120/11856-7623.
14. M. Wang, S. Cao, and Y. Wang.VoNR-IPD: A Novel Timing-Based Network Steganography for Industrial Internet // Security and Communication Networks. 2020. Vol. 2020, Article ID 8846230, 14 p., Jun. 2020. doi: 10.1155/2020/8846230.
15. G. Fisk, M. Fisk, C. Papadopoulos, and J. Neil. Eliminating steganography in internet traffic with active wardens // Information Hiding, F. A. P. Petitcolas, Ed. Heidelberg: Springer, 2003. P. 18–35. doi: 10.1007/3-540-36415-3_2.
16. A. S. Nair, A. Kumar, A. Sur, and S. Nandi. Length based network steganography using UDP protocol // Proc. 2011 IEEE 3rd Int. Conf. Commun. Softw. Netw., Xi'an, China, May 2011. P. 588–592. doi: 10.1109/ICCSN.2011.6014994.
17. W. Fraczek, W. Mazurczyk, and K. Szczypiorski. Stream control transmission protocol steganography // Proc. 2010 Int. Conf. Multimedia Inf. Netw. Secur., Nanjing, China, Nov. 2010. P. 829–834. doi: 10.1109/MINES.2010.176.

18. W. Mazurczyk and K. Szczypiorski. Covert channels in SIP for VoIP signalling // Multimedia Communications, Services and Security, R. Choras, Ed. Heidelberg: Springer, 2008, pp. 65–76. doi: 10.1007/978-3-540-88873-4_6.

19. Y. Ji, Y. Wang, and Y. Zhang. Constructing SDN covert timing channels between hosts with unprivileged attackers // IEEE/ACM Trans. Netw. 2024. Vol. 32, no. 1. P. 1–14. doi: 10.1109/TNET.2024.3496997.

20. C. Heinz, M. Zuppelli, and L. Caviglione. Covert Channels in Transport Layer Security: Performance and Security Assessment // Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications. 2021. Vol. 12, no. 4. P. 22–36. doi: 10.22667/JOWUA.2021.12.31.022.

21. T. Grübl, W. Niu, J. von der Assen, and B. Stiller. QUIC-Exfil: Exploiting QUIC's Server Preferred Address Feature to Perform Data Exfiltration Attacks // Proc. ACM Asia Conf. on Computer and Communications Security (AsiaCCS). Hanoi, Vietnam, 2025. doi: 10.1145/3708821.3733872.

22. K. Thimmaraju, R. Krösche, L. Schiff, and S. Schmid. CVE-2018-1000155: Denial of service, improper authentication and authorization, and covert channel in the OpenFlow 1.0+ handshake. oss-sec mailing list, May 9, 2018. Available: https://seclists.org/oss-sec/2018/q2/99

23. R. Krösche, K. Thimmaraju, and S. Schmid. I DPID it my way! A covert timing channel in software-defined networks // Proc. 2018 IFIP Netw. Conf., Zurich, Switzerland, May 2018, pp. 1–9. doi: 10.23919/IFIPNetworking.2018.8696597.

24. S. Bistarelli, M. Ceccarelli, C. Luchini, I. Mercanti, and F. Santini. A Preliminary Study on the Creation of a Covert Channel with HTTP Headers // Proceedings of the Italian Conference on CyberSecurity (ITASEC 2024), Ancona, Italy, Jan. 2024. Available: https://ceur-ws.org/Vol-3731/paper34.pdf

25. E. Brown, B. Yuan, D. Johnson, and P. Lutz. Covert Channels in the HTTP Network Protocol: Channel Characterization and Detecting Man-in-the-Middle Attacks // Journal of Information Warfare. 2010. Vol. 9, no. 3. P. 1–12. Available: https://repository.rit.edu/other/781/

26. S. Smendowski. Hidden Communication Using Covert Channels // GitHub repository, 2021. Available: https://github.com/Smendowski/hidden-communication-using-covert-channels

27. W. A. Dimitrova and G. S. Panayotova. The Impacts of DNS Protocol Security Weaknesses // Journal of Communications. 2020. Vol. 15, no. 5. P. 1–9. Available: https://www.jocm.us/show-245-1596-1.html

28. M. Hildebrandt, R. Altschaffel, K. Lamshöft, M. Lange, M. Szemkus, T. Neubert, C. Vielhauer, Y. Ding, and J. Dittmann. Threat Analysis of Steganographic and Covert Communication in Nuclear I&C Systems // presented at the International Conference on Nuclear Security: Sustaining and Strengthening Efforts (ICONS 2020), Vienna, Austria, Feb. 2020. Available: https://conferences.iaea.org/event/181/contributions/15608/attachments/8569/11404/CN278_478-stealth_v006.pdf

Надійшла до редколегії 15.06.2025

Відомості про авторів:

Фокін Денис Геннадійович – Харківський національний університет радіоелектроніки, аспірант кафедри інфокомунікаційної інженерії ім. В.В. Поповського, Україна; e-mail: denys.fokin@nure.ua; ORCID: https://orcid.org/0009-0002-3282-842X

Євдокименко Марина Олександрівна – д-р техн. наук, професор, Харківський національний університет радіоелектроніки, професор кафедри інфокомунікаційної інженерії ім. В.В. Поповського, Україна; e-mail: maryna.yevdokymenko@ieee.org; ORCID: https://orcid.org/0000-0002-7391-3068