КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ БУДІВНИЦТВА І АРХІТЕКТУРИ

Автоматизації і інформаційних технологій

(факультет)

Кафедра кібербезпеки та комп'ютерної інженерії

(назва випускової кафедри)

КВАЛІФІКАЦІЙНА РОБОТА
ЗДОБУВАЧА СТУПЕНЯ ВИЩОЇ ОСВІТИ МАГІСТР

на тему:

Комплексна діагностика мережі за допомогою засобів аналізу трафіку

Сарапин Вадим Євгенійович

(прізвище, ім'я та по батькові здобувача повністю)

Київ 2025 р.

КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ БУДІВНИЦТВА І АРХІТЕКТУРИ

Автоматизації і інформаційних технологій

(факультет)

Кафедра кібербезпеки та комп'ютерної інженерії

(назва кафедри)

ЗАТВЕРДЖУЮ
Завідувач кафедри
Делембовський М.М.
„___" _____20___ року

КВАЛІФІКАЦІЙНА РОБОТА ЗДОБУВАЧА СТУПЕНЯ ВИЩОЇ ОСВІТИ МАГІСТР

Комплексна діагностика мережі за допомогою засобів аналізу трафіку

(назва)

Я як здобувач вищої освіти КНУБА розумію і підтримую політику закладу з академічної доброчесності. Я не надавав(-ла) і не одержував(-ла) недозволену допомогу під час підготовки цієї роботи. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело.

Здобувач Сарапин Вадим Євгенійович
(прізвище, ім'я та по батькові повністю)
125 «Кібербезпека та захист інформації»
(спеціальність)
Безпека інформаційних і комунікаційних систем
(освітня програма)
Група БІКСм-24
Керівник Шабала Є.Є.
(прізвище та ініціали)
Кандидат технічних наук, доцент
(вчене звання, науковий ступінь)
Рецензент Терентьєв О.О.
(прізвище та ініціали)
Ідентичність підтверджую

КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ БУДІВНИЦТВА І АРХІТЕКТУРИ

Факультет:	автоматизації і інформаційних технологій
Випускова кафедра:	кібербезпеки та комп'ютерної інженерії
Ступінь вищої освіти:	магістр
Спеціальність:	125 «Кібербезпека та захист інформації»
Освітня програма:	Безпека інформаційних і комунікаційних систем

ЗАТВЕРДЖУЮ

Завідувач кафедри

Делембовський М.М.

„___"_____20___ року

З А В Д А Н Н Я
ДО ВИКОНАННЯ КВАЛІФІКАЦІЙНОЇ РОБОТИ
ЗДОБУВАЧА СТУПЕНЯ ВИЩОЇ ОСВІТИ МАГІСТР

Сарапин Вадим Євгенійович

(прізвище, ім'я та по батькові здобувача)

1. Тема роботи **«Комплексна діагностика мережі за допомогою засобів аналізу трафіку»**

затверджена наказом ректора КНУБА № 1635/23.2/25 від «30» вересня 2025 року

2. Керівник роботи Шабала Євгенія Євгенівна

кандидат технічних наук, доцент

(прізвище, ім'я та по батькові, науковий ступінь, вчене звання)

3. Термін подання здобувачем роботи до захисту грудень 2025 року

4. Зміст пояснювальної записки за розділами:

Р. 1. Аналіз сучасного стану діагностики мереж

Р. 2. Технології та алгоритми мережевої діагностики

Р. 3. Експериментальне дослідження та аналіз мережевих аномалій і затримок

5. Графічний матеріал за розділами:

Р. 1. Актуальність, мета, завдання, обєкт, предмет дослідження, наукова новизна, ключові аспекти важливості КМ, проблеми діагностики мережі, класифікація систем діагностики КМ.

Р. 2. Класифікація мережевих відмов, моделі мережевого трафіку, порівняння Пуассонівського та Марківського процесів, методи для аналізу трафіку та виявлення аномалій, життєвий цикл аналізу мережевого трафіку.

Р. 3. Топологія тестового полігону, опис мережевих атак, моделювання мережевих атак, визначення та призначення основних метрик для оцінки систем виявлення аномалій, результати виконання команд в консолі, результати аналізу мережевого трафіку за допомогою Wireshark, результати аналізу мережевих розмов та окремих пакетів TCP/TLS з використанням Wireshark, Візуалізація мережевого трафіку з виявленими TCP-помилками, висновки.

6. Консультанти розділів кваліфікаційної випускної роботи:

Розділ	Прізвище, ініціали та посада	Перевірив	
		дата	підпис
Розділ 1.			
Розділ 2.			
Розділ 3.			

7. Календарний план виконання роботи:

Види робіт та їх зміст	Дата виконання
Розділ 1. Аналіз сучасного стану діагностики мереж	Вересень 2025
Розділ 2. Технології та алгоритми мережевої діагностики	Жовтень 2025
Розділ 3. Експериментальне дослідження та аналіз мережевих аномалій і затримок	Листопад 2025
Остаточне оформлення роботи	Грудень 2025
Направлення роботи для перевірки на плагіат	Грудень 2025
Попередній захист роботи	Грудень 2025
Направлення роботи на рецензування	Грудень 2025

8. Дата видачі завдання _____

Керівник _____ _____

Здобувач _____ _____

АНОТАЦІЯ

Сарапин В. Є. «Комплексна діагностика мережі за допомогою засобів аналізу трафіку».

Кваліфікаційна робота магістра за спеціальністю 125 «Кібербезпека та захист інформації», освітня програма «Безпека інформаційних і комунікаційних систем». – Київський національний університет будівництва та архітектури. – Київ, 2025.

У роботі досліджено теоретичні засади діагностики комп'ютерних мереж, сучасні архітектури мереж, типи мережевих відмов, показники якості обслуговування (QoS), а також методи аналізу мережевого трафіку та виявлення аномалій. Розглянуто математичні моделі мережевого трафіку, зокрема пуассонівські та марківські процеси, як основу для моделювання мережевих процесів.

У практичній частині роботи реалізовано тестовий полігон комп'ютерної мережі, виконано захоплення та аналіз трафіку за допомогою інструментів Wireshark, tcpdump, NetFlow, змодельовано мережеві атаки типу SYN Scan та UDP Flood, а також проведено експериментальне дослідження мережевих аномалій і затримок. Проаналізовано результати роботи системи виявлення аномалій та оцінено її ефективність за основними метриками.

Отримані результати підтверджують, що запропонований підхід до комплексної діагностики мережі дозволяє своєчасно виявляти аномалії, мережеві відмови та підвищувати рівень безпеки і стабільності мережевої інфраструктури.

Ключові слова: Мережева діагностика, аномалії трафіку, QoS, Візантійські відмови, SPAN-порт.

SUMMARY

Sarapyn V. Y. «Comprehensive Network Diagnostics Using Traffic Analysis Tools».

Master's qualification thesis in specialty: 125 "Cybersecurity and Information Protection", educational program: "Security of Information and Communication Systems". – Kyiv National University of Civil Engineering and Architecture. – Kyiv, 2025.

The thesis investigates the theoretical foundations of computer network diagnostics, modern network architectures, types of network failures, Quality of Service (QoS) indicators, as well as methods of traffic analysis and anomaly detection. Mathematical models of network traffic, including Poisson and Markov processes, are considered as the basis for modeling network behavior.

In the practical part of the work, a test network environment was deployed, traffic was captured and analyzed using Wireshark, tcpdump, and NetFlow, network attacks such as SYN Scan and UDP Flood were simulated, and an experimental study of network anomalies and delays was conducted. The effectiveness of the anomaly detection system was evaluated using key performance metrics.

The obtained results confirm that the proposed comprehensive network diagnostics approach enables timely detection of anomalies and network failures, and improves the security and stability of network infrastructure.

Keywords: Network diagnostics, traffic anomalies, QoS, Byzantine faults, SPAN-port.

РЕЗЮМЕ (SUMMARY) *до кваліфікаційної випускової роботи здобувача*	*Сарапин Вадим Євгенійович* *Sarapyn Vadym Yevheniyovych*

ЗВО	Київський національний університет будівництва і архітектури		
Тема (українською та англійською)	Комплексна діагностика мережі за допомогою засобів аналізу трафіку Comprehensive network diagnostics using traffic analysis tools		
Освітній ступінь	магістр		
Факультет	Автоматизації і інформаційних технологій		
Випускова кафедра	Кібербезпеки та комп'ютерної інженерії		
Спеціальність	125 «Кібербезпека та захист даних»		
Освітня програма	Безпека інформаційних і комунікаційних систем		
Керівник	к.т.н., доцент Шабала Євгенія Євгенівна		
Обсяг роботи:	*Пояснювальна записка, стор.*	*Розділів*	*Презентація, кількість слайдів*
	113	три	21
Розділ 1	Аналіз сучасного стану діагностики мереж		
Розділ 2	Технології та алгоритми мережевої діагностики		
Розділ 3	Експериментальне дослідження та аналіз мережевих аномалій і затримок		
Висновки по роботі	Робота присвячена актуальній проблемі забезпечення ефективної діагностики комп'ютерних мереж шляхом аналізу трафіку та виявлення аномалій. Запропонований підхід поєднує теоретичні основи, сучасні технології та практичні методи, що дозволяє підвищити рівень безпеки та стабільності мережевої інфраструктури.		
Ключові слова: *Keywords:*	Мережева діагностика, аномалії трафіку, QoS, Візантійські відмови, SPAN-порт Network diagnostics, traffic anomalies, QoS , Byzantine faults, SPAN port		

Здобувач _____/ _____

Керівник _____/ _____

"___" _____ 20__

ЗМІСТ

ВСТУП

У сучасному цифровому середовищі, де комп'ютерні мережі стали основою функціонування бізнесу, державних установ і повсякденного життя, питання їхньої стабільності, безпеки та ефективності набувають критичного значення. Зростання складності мережевих архітектур, поява нових типів загроз і збільшення обсягів трафіку створюють серйозні проблеми для систем адміністрування та моніторингу.

Актуальність теми обумовлена необхідністю забезпечення безперервної роботи мережевих сервісів, своєчасного виявлення аномалій та запобігання відмовам, які можуть мати катастрофічні наслідки для організацій. Проблематика полягає в тому, що традиційні методи діагностики часто не здатні оперативно реагувати на складні, багатофакторні загрози, особливо в умовах розподілених систем і високої динаміки трафіку.

Комплексний підхід до мережевої діагностики, який поєднує глибоке теоретичне розуміння архітектури мереж, знання типів відмов, показників якості обслуговування та сучасних алгоритмів виявлення аномалій, є необхідною умовою для створення ефективних систем моніторингу.

Зростання складності архітектур, поява нових типів загроз та збільшення обсягів трафіку створюють серйозні проблеми для систем моніторингу та адміністрування.

Метою дослідження є розробка підходу до комплексної діагностики комп'ютерної мережі на основі аналізу трафіку, що дозволяє своєчасно виявляти аномалії, затримки та потенційні загрози з метою підвищення надійності та безпеки мережевої інфраструктури. Для досягнення цієї мети поставлено низку завдань: аналіз архітектури мереж, класифікація типів відмов, дослідження показників якості обслуговування (QoS), застосування сучасних засобів збору трафіку (tcpdump, NetFlow, PF_RING), моделювання атак та оцінка ефективності алгоритмів виявлення аномалій.

Об’єктом дослідження виступає корпоративна комп’ютерна мережа з багаторівневою системою безпеки, а предметом — методи та засоби діагностики мережевих аномалій на основі аналізу трафіку.

Необхідність розробки обумовлена недостатньою гнучкістю існуючих рішень, високим ризиком втрати даних та збоїв у роботі сервісів через несвоєчасне реагування на інциденти. Запропонований підхід дозволяє зменшити час виявлення критичних інцидентів, знизити кількість хибних спрацювань, підвищити точність класифікації трафіку та забезпечити глибший рівень аналітики без втручання в роботу мережі.

Наукова новизна роботи полягає в інтеграції математичних моделей трафіку, концепції візантійських відмов та сучасних засобів захоплення пакетів у єдину систему діагностики, здатну адаптуватися до змін у мережевому середовищі та забезпечувати високоточне виявлення складних багатофазних атак. Задача візантійських генералів - це уявний експеримент, який стосується ключового питання інформатики: чи можливо сформувати консенсус у комп’ютерній мережі, що складається з незалежних географічно розподілених вузлів [25] Було реалізовано інтеграцію концепції візантійських відмов у контексті мережевої діагностики, що дозволяє моделювати складні сценарії логічної нестабільності вузлів, зокрема ситуації, коли пристрої поширюють суперечливу або неправдиву інформацію про маршрутизацію.

Однією з центральних задач діагностики є встановлення причин виникнення несправностей та їх усунення [25]. Такий підхід дозволяє не лише фіксувати очевидні порушення, а й виявляти приховані, складні загрози, що проявляються у вигляді нестабільності, затримок або некоректної поведінки окремих вузлів.

Використання математичних моделей трафіку, адаптивних алгоритмів класифікації та точкових сенсорів у критичних зонах мережі забезпечує глибоку аналітику та своєчасне реагування на інциденти. Тому поєднання теоретичних знань із практичними інструментами діагностики формує основу для побудови надійної, масштабованої та безпечної мережевої інфраструктури.

1. АНАЛІЗ СУЧАСНОГО СТАНУ ДІАГНОСТИКИ МЕРЕЖ

1.1. Архітектура сучасних комп’ютерних мереж

Комп'ютерна мережа — це система, що складається з двох або більше взаємопов'язаних кінцевих пристроїв (комп'ютерів, серверів, смартфонів, сенсорів), які можуть обмінюватися даними, ресурсами та інформацією за допомогою встановлених протоколів зв'язку. Сучасна мережа — це не просто кабелі та пристрої; це складна інфраструктура, що забезпечує глобальну зв'язаність та спільне використання ресурсів.

Комп'ютерні мережі перетворилися з інструменту обміну файлами на основу будь-якої сучасної економічної, соціальної та наукової діяльності. Їхня роль є критичною:

Рис. 1.1 Ключові аспекти важливості комп’ютерних мереж

Однією з ключових переваг комп'ютерних мереж є можливість спільного використання ресурсів. Це означає, що користувачі можуть отримувати доступ до централізованих ресурсів, таких як принтери, бази даних та обчислювальні потужності серверів, незалежно від їхнього фізичного розташування.

Мережі забезпечують доступ до централізованих баз даних, що дозволяє користувачам отримувати актуальну інформацію та спільно працювати над проє-

ктами. Це особливо важливо для великих організацій, де дані повинні бути узгодженими та доступними для всіх. Мережі дозволяють користувачам використовувати обчислювальні потужності серверів для виконання складних завдань, таких як аналіз даних, моделювання та рендеринг. Це особливо корисно для організацій, які потребують великих обчислювальних ресурсів, але не можуть дозволити собі придбати та обслуговувати власні сервери. Комп'ютерні мережі дозволяють адміністраторам централізовано контролювати безпеку, конфігурацію та оновлення всіх пристроїв. Це значно спрощує управління мережею та підвищує її безпеку.

Централізоване управління дозволяє адміністраторам встановлювати політики безпеки для всієї мережі, такі як паролі, антивірусне програмне забезпечення та брандмауери. Це допомагає захистити мережу від зовнішніх загроз та внутрішніх порушень. Адміністратори можуть централізовано конфігурувати всі пристрої в мережі, такі як комп'ютери, сервери та мережеве обладнання. Це забезпечує узгодженість конфігурації та зменшує ризик помилок. Централізоване управління дозволяє адміністраторам встановлювати оновлення програмного забезпечення на всі пристрої в мережі одночасно. Це забезпечує, що всі пристрої працюють з останньою версією програмного забезпечення, що містить виправлення помилок та покращення безпеки. Мережі забезпечують роботу критично важливих бізнес-додатків (ERP, CRM) та засобів комунікації (електронна пошта, VoIP, відеоконференції). Це дозволяє організаціям ефективно управляти своїми бізнес-процесами та підтримувати зв'язок з клієнтами, партнерами та співробітниками.

ERP-системи дозволяють організаціям управляти всіма аспектами свого бізнесу, від фінансів та бухгалтерського обліку до управління ланцюгом поставок та виробництва. Мережі забезпечують доступ до ERP-систем для всіх користувачів, що дозволяє їм спільно працювати над проєктами та отримувати актуальну інформацію.

CRM-системи дозволяють організаціям управляти своїми відносинами з клієнтами. Мережі забезпечують доступ до CRM-систем для всіх користувачів,

що дозволяє їм відстежувати взаємодію з клієнтами, управляти продажами та надавати підтримку.

Бездротові мережі та хмарні сервіси дозволяють співробітникам працювати з будь-якої точки, що є основою сучасної гібридної роботи. Це підвищує продуктивність та задоволеність співробітників. Бездротові мережі дозволяють користувачам підключатися до мережі без використання кабелів. Це забезпечує більшу гнучкість та мобільність, оскільки користувачі можуть працювати з будь-якої точки в зоні покриття бездротової мережі. Хмарні сервіси дозволяють організаціям зберігати дані та запускати додатки в хмарі. Це забезпечує більшу гнучкість та масштабованість, оскільки організації можуть отримувати доступ до своїх даних та додатків з будь-якої точки світу. Мережа слугує транспортним середовищем для мільярдів підключених сенсорів, що збирають дані для промисловості, медицини та побуту. Це дозволяє організаціям отримувати цінну інформацію про свої операції та приймати більш обґрунтовані рішення. IoT-сенсори можуть використовуватися для моніторингу обладнання, відстеження запасів та оптимізації виробничих процесів.

Сучасна комп'ютерна мережа — це не просто сукупність з'єднаних пристроїв, а комплексна, динамічна система, спроєктована для забезпечення високої доступності, безпеки та масштабованості. Розуміння її архітектури є першочерговим кроком до успішної діагностики та аналізу трафіку, оскільки структура мережі диктує, де і як слід захоплювати та інтерпретувати дані. За формою представлення комп’ютерних мереж розрізняють фізичну та логічну архітектуру. Фізична архітектура – форма представлення комп’ютерної мережі у вигляді взаємодіючих апаратних засобів. Приклад фізичної архітектури зображено на рис. 1.1.

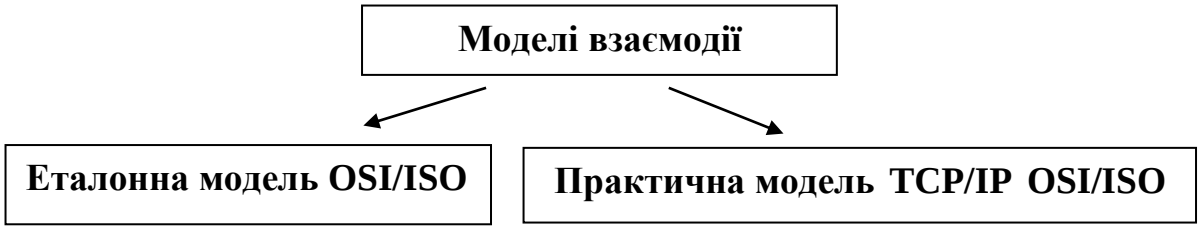


Рис. 1.2 Моделі взаємодії

Семирівнева модель OSI (Open Systems Interconnection) є теоретичною основою, яка стандартизує функції мережевої взаємодії. Вона критично важлива для аналізу трафіку, оскільки кожен рівень відповідає за формування певної одиниці даних (PDU — Protocol Data Unit):

Рис. 1.3 Рівні OSI

Робота моделі починається з Фізичного рівня (Рівень 1), який відповідає за передачу бітів по кабелях. Над ним розташований Канальний рівень (Рівень 2), що керує передачею даних у межах локальної мережі (Кадри – Ethernet, Wi-Fi) і є ключовим для аналізу локальних помилок. Далі йде Мережевий рівень (Рівень 3), завданням якого є маршрутизація даних між мережами (Пакети – IP), що має вирішальне значення для діагностики шляху. Транспортний рівень (Рівень 4) забезпечує надійну або швидку доставку даних (Сегменти – TCP/UDP), будучи ключовим для аналізу продуктивності. Вище знаходяться Сеансовий рівень (Рівень 5), що керує сеансами зв'язку, Рівень представлення (Рівень 6), який форматує дані (шифрування, стиснення), і, нарешті, Прикладний рівень (Рівень 7), який забезпечує взаємодію з користувачем (HTTP, SMTP).

На відміну від OSI, модель TCP/IP (Transmission Control Protocol/Internet Protocol) є практичним стандартом, що використовується в Інтернеті та більшості

корпоративних мереж. Вона має лише чотири рівні: Прикладний, Транспортний, Міжмережевий та Рівень мережевих інтерфейсів.

Центральною концепцією, яку повинен розуміти фахівець з діагностики, є інкапсуляція. Коли дані рухаються від прикладного рівня до фізичного, до них на кожному кроці додається заголовок протоколу (заголовок TCP, заголовок IP та заголовок Ethernet). Аналізатор трафіку виконує деінкапсуляцію, розкриваючи ці заголовки для вивчення.

Рис. 1.4 Ієрархічна архітектура

Сучасні корпоративні мережі будуються на принципах ієрархії для підвищення надійності, передбачуваності та спрощення управління [41].

Більшість архітектур, особливо великих LAN (Local Area Network), використовують трирівневу модель (часто спрощену до дворівневої для менших мереж):

1. Рівень Доступу (Access Layer): підключає кінцевих користувачів та пристрої (ПК, IP-телефони, принтери) до мережі. Тут працюють комутатори, які забезпечують підключення та застосування політик доступу.

2. Рівень Розподілу (Distribution Layer): слугує буфером між рівнями доступу та ядра. Він виконує функції маршрутизації (IP-адресації), агрегації трафіку з багатьох комутаторів доступу та впроваджує політики безпеки, VLAN та QoS (якість обслуговування).

3. Рівень Ядра (Core Layer): забезпечує високошвидкісний транспорт між розподільчими рівнями. Його головна функція — швидка комутація/маршрутизація з мінімальною обробкою пакетів, щоб уникнути затримок.

Класифікація за географічним охопленням

Хоча архітектурні принципи схожі, класифікація за розміром визначає технології, що використовуються:

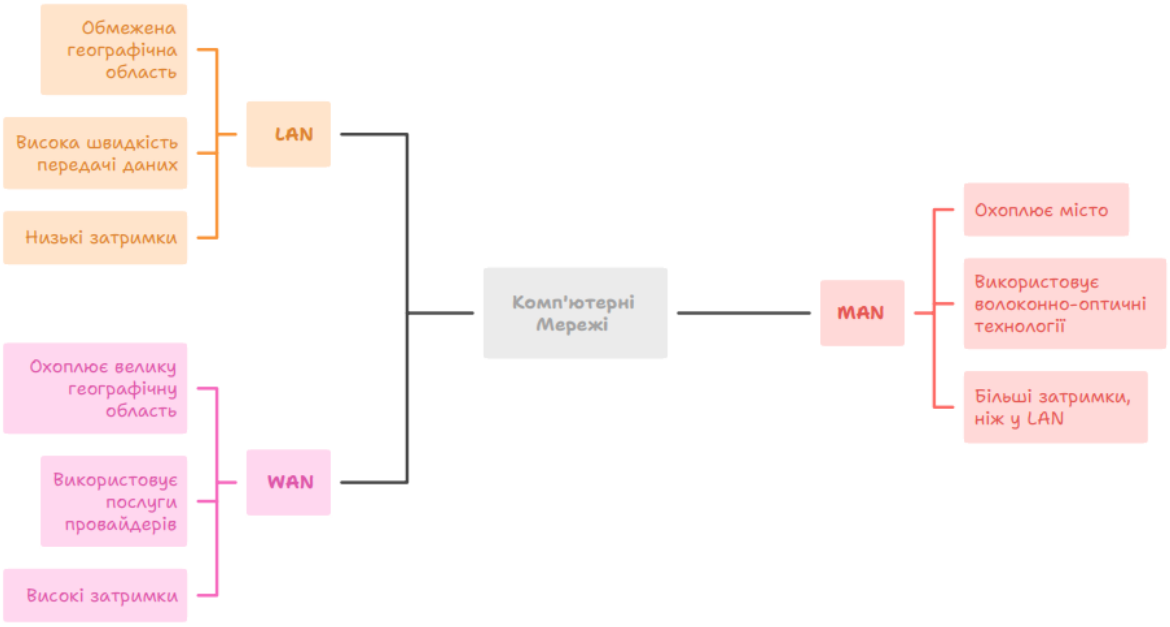


Рис. 1.5 Типи компютерних мереж

Локальна мережа (LAN) обмежена будівлею або кампусом, і її основні характеристики — високі швидкості (наприклад, Gigabit Ethernet) та мінімальні затримки. Муніципальна мережа (MAN) охоплює територію міста та часто використовує високошвидкісні волоконно-оптичні технології (Metro Ethernet) для з'єднання кількох LAN. Найбільш масштабна глобальна мережа (WAN) з'єднує віддалені локальні мережі, покладаючись на послуги провайдерів, таких як MPLS, VPN або орендовані канали. Через великі відстані у WAN спостерігаються значно вищі затримки, що ускладнює комплексне діагностування.

Топологію комп'ютерної мережі слід розуміти, як схему з'єднання вузлів без врахування відстані між ними і їх територіального розміщення. Найпростіші варіанти топології, які відповідають окремим (не поєднаним між собою) мережам, показані на рис.1.5.

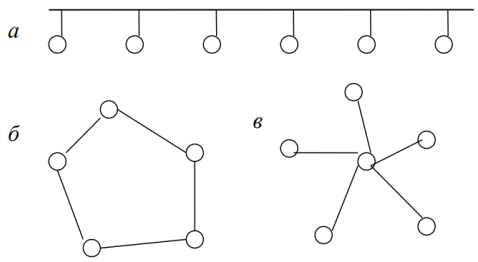


Рис. 1.6. Основні варіанти топології КМ: а – шинна (спільна шина); б – кільцева ("кільце); в – зіркоподібна ("зірка").

Перші мережі з більшою за два кількістю комп'ютерів було створено за цими варіантами топології. У випадку мережі з двома вузлами усі ці варіанти будуть мати однаковий вигляд. Найбільш відомими і розповсюдженими у наш час є мережі, що побудовані за технологією Ethernet, у яких топологія є шинною.

Наступним кроком щодо поліпшення роботи Ethernet мереж була поява комутаторів, які виконують ту ж функцію, що й концентратори, але мають пам'ять на кожному порту. У цій пам'яті можуть затримуватись пакети даних у разі зайнятості шини, що дозволяє уникати колізій. При цьому топологія фізичних зв'язків з появою з'єднувальних пристроїв таких, як концентратори та комутатори, буде зіркоподібною або деревоподібною (але не кільцевою), як показано на рис. 1.6.

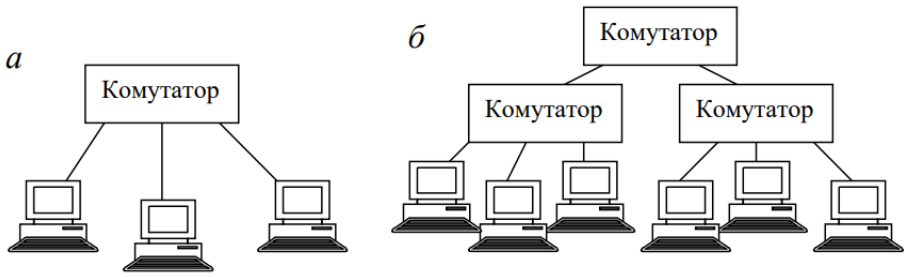


Рис. 1.7. Топологія фізичних зв'язків мережі Ethernet: а – зіркоподібна; б – деревоподібна.

Впровадження комутаторів чи концентраторів не змінює топологію логічних зв'язків мережі, яка залишається шинною [1].

У мережі із зіркоподібною топологією кожен абонент, що посилає і (чи) приймає інформацію, приєднаний одним чи двома виділеними каналами зв’язку до єдиного центрального вузла, через який проходить весь мережевий трафік. Кожен вузол підключається окремим кабелем до загального пристрою, який має

назву концентратор та розташовується в центрі мережі. У функції концентратора входить спрямування переданої комп'ютером інформації одному чи всім іншим комп'ютерам мережі. Деревоподібні мережі будуються на базі техніки кабельного телебачення, тобто з використанням таких засобів зв’язку, як кінцеві частотні ретранслятори, розщеплювачі-об’єднувачі, двонапрямлені посилювачі, відгалужувачі, радіочастотні модеми, фільтри тощо. [2]

Схему об'єднаної мережі, яка складається з декількох окремих мереж, зображено на рис.1.7, де жирною лінією позначені шини. Шиною можуть бути не тільки комутатори чи їх з'єднання, але й радіо ефір, як у мережі ALOHA, ідеї якої покладено у технологію Ethernet. Назва Ethernet походить від Ether (ефір) та network (мережа) на честь мережі ALOHA. Маршрутизатори можна з'єднувати між собою за будь-яким варіантом топології, включаючи кільця [1].

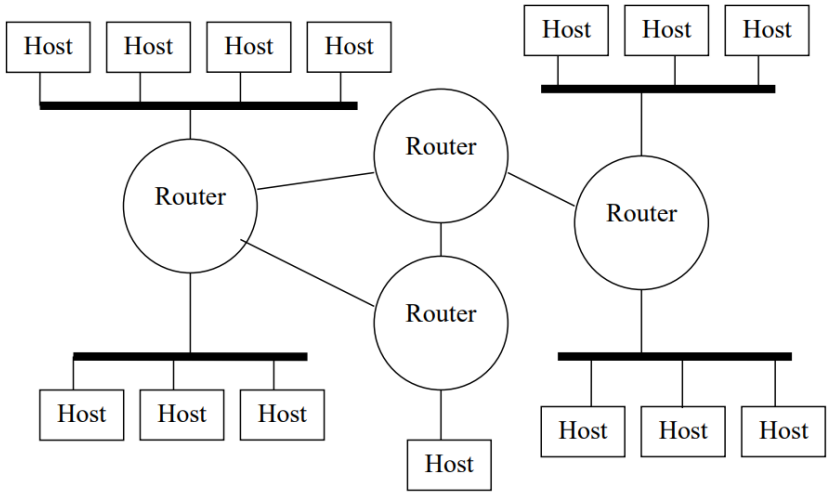


Рис. 1.8. Топологія складної мережі

На рисунку зображена структурна схема комп’ютерної мережі, побудована за ієрархічним принципом із використанням маршрутизаторів (Router) для з’єднання окремих локальних сегментів, у кожному з яких розміщено кілька вузлів (Host). Host (вузол, клієнт, робоча станція) – це комп’ютери користувачів або сервери, під’єднані до локального сегмента мережі. Вони підключені через комутатор або концентратор (switch/hub), який позначено товстою горизонтальною лінією. Кожен Host має свою IP-адресу, але всі в межах однієї підмережі мають спільну адресу мережі.

Router (маршрутизатор) – це активний мережевий пристрій, який з’єднує між собою кілька мереж різних підмережних сегментів. Він працює на 3-му рівні моделі OSI (мережевому рівні) і визначає, куди передавати пакети даних, використовуючи таблицю маршрутизації. Кожен маршрутизатор має кілька інтерфейсів – кожен інтерфейс підключений до певної мережі.

Таким чином, у фізичному сенсі саме маршрутизатор забезпечує логічне з'єднання і передачу даних між різними мережевими сегментами, на основі чого вибудовуються моделі взаємодії вузлів. Ці моделі визначають, як кінцеві пристрої використовують ресурси, незалежно від того, чи заснована ця взаємодія на ієрархії клієнт-сервер, чи на децентралізованому підході однорангової мережі (P2P).

За організацію взаємодії між комп’ютерами мережі поділяють на одноранговi (per to per або P2P) та ієрархічні мережі, які ще називають мережами з виділеним сервером. Одноранговa мережа складається виключно з робочих станцій, кожна з яких має особисто вирішувати питання автентифікації користувачів, керування доступом до власних ресурсів, встановлення та підтримки програмного забезпечення тощо. В одноранговій мережі всі робочі станції рів- 12 ноправні. Робоча станція може надавати доступ до таких своїх ресурсів, як дисковий простір та периферійне обладнання. Перевагою P2P мереж є простота розгортання і налаштування, однак при великій кількості робочих станцій процес адміністрування такої мережі суттєво ускладняється, а захищеність і надійність знижуються. Місце застосування одноранговых мереж – домашні мережі та мережі малих офісів.

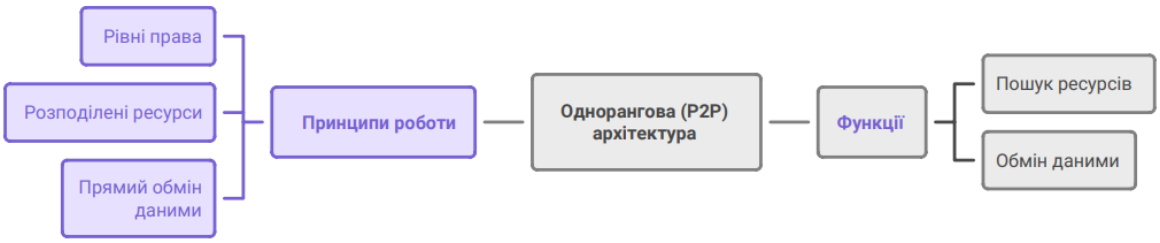


Рис. 1.9. Одноранговa (P2P) архітектура

Застосування клієнт-серверної архітектури передбачає наявність виділеного комп’ютера або комп’ютерів, що виконують специфічні функції в мережі й надають послуги іншим комп’ютерам. Ці пристрої називають серверами (Server – той,

хто надає послуги). Однією з найбільш важливих є функція керування доступом до мережі та її ресурсів. Таку задачу вирішують, наприклад, Active Directory сервер від Microsoft, ZENworks від Micro Focus та інші. Крім того в корпоративній мережі актуальними є функції файлового, поштового сервера, сервера друку. Слід звернути увагу, що клієнт-серверна архітектура є основною архітектурою мережі Інтернет, оскільки основні ресурси цієї мережі зберігаються на веб-серверах [3].

Рис. 1.10. Цикл клієнт-серверної взаємодії

Особливості клієнт-серверної архітектури не обмежуються лише наданням основних корпоративних послуг, таких як файлообмін чи пошта. Вона також є критичною для забезпечення масштабованості та централізованої безпеки великих систем. Завдяки чіткому розділенню ролей, мережу легко розширювати, додаючи нові сервери для розподілу навантаження (наприклад, використовуючи ферми веб-серверів або балансувальники навантаження). Це гарантує, що зростаюча кількість клієнтів не призведе до зниження продуктивності.

Централізований контроль також дозволяє ефективно впроваджувати політики безпеки, керувати обліковими записами користувачів та проводити регулярне резервне копіювання даних, мінімізуючи ризики втрати інформації.

На противагу цьому, однорангова архітектура (P2P), незважаючи на свою децентралізовану природу, знайшла широке застосування у специфічних нішах.

Вона ідеально підходить для невеликих робочих груп (до 10-15 пристроїв), де немає потреби у виділеному адміністраторі або високій ієрархії управління.

Основні переваги P2P – простота налаштування, низька вартість і висока відмовостійкість на рівні даних: якщо один вузол виходить з ладу, інші вузли можуть продовжувати обмінюватися даними. Однак ця модель створює значні проблеми з безпекою та контролем, оскільки кожен користувач самостійно відповідає за спільний доступ до своїх ресурсів і захист свого пристрою, що робить її непридатною для більшості корпоративних середовищ, які вимагають жорсткого централізованого контролю.

1.2. Основні проблеми в процесах діагностики мереж

Забезпечення стабільності функціонування комп'ютерної мережі в підприємстві є важливою умовою для ефективності його операцій. Таким чином, будь-які відмови, незалежно від їх походження, порушують цю стабільність і потребують негайної ідентифікації причин та усунення проблем. Отже, швидка реакція та виявлення кореневих причин відмови є ключовими аспектами для запропонованої системи.

Існують два типи систем виявлення: системи виявлення аномалій і системи виявлення ознак. Однак, головною проблемою систем виявлення функцій є їх призначення для виявлення конкретних типів атак, які зазвичай є найнебезпечнішими на момент створення системи. Коли з'являються нові атаки або параметри атак змінюються, системам виявлення потрібно знову адаптуватися.

Системи виявлення аномалій часто базуються на складних моделях нормального інтернет-трафіку, що припускають статистичну однорідність трафіку. Однак не завжди враховується контекст, в якому застосовуються ці припущення, а також умови їх використання. Це може призвести до необхідності перенавчання алгоритмів у випадку навіть незначних змін у структурі трафіку чи надання послуг [4].

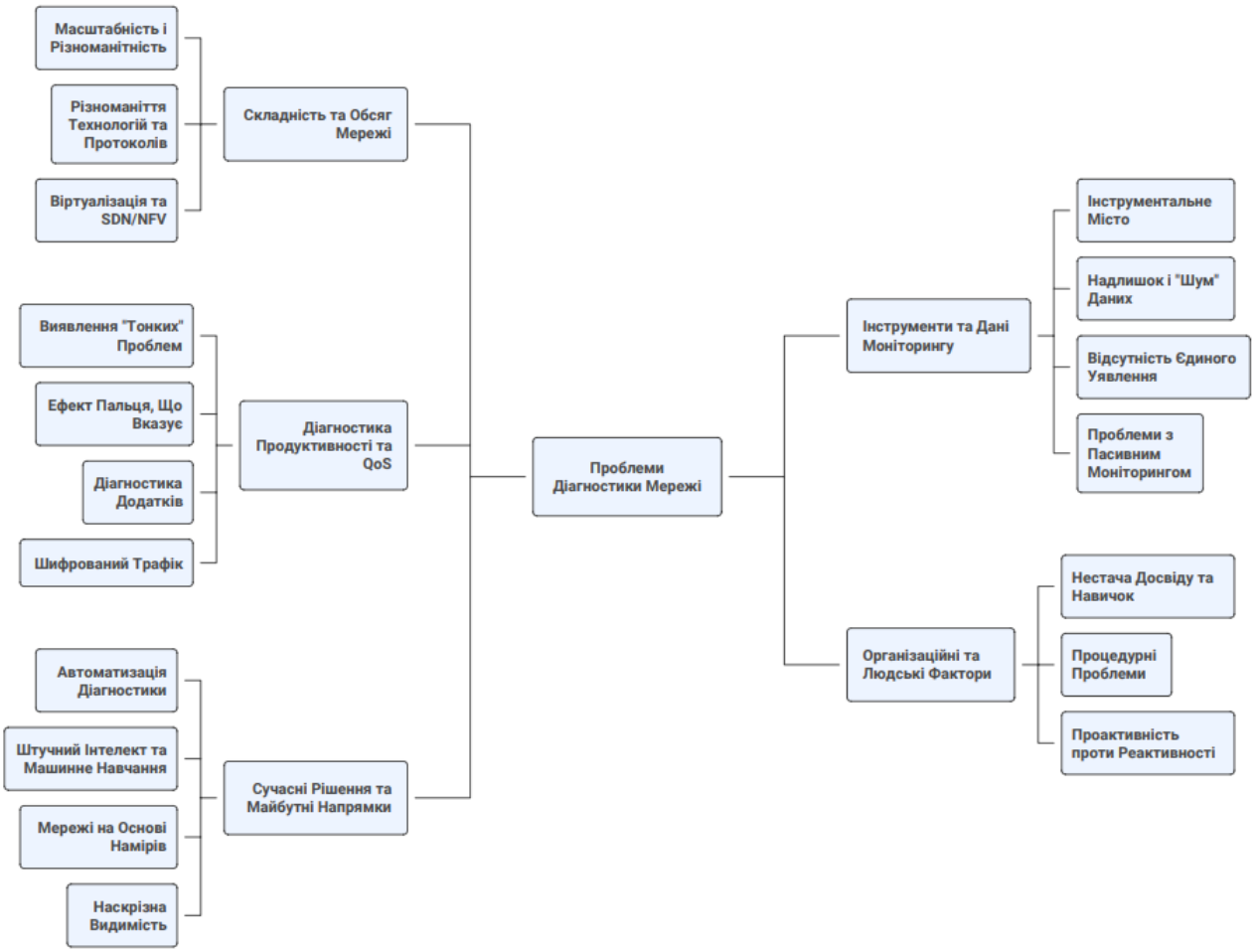


Рис. 1.11. Проблеми та рішення в діагностиці мережі

Опис блок-схеми:

1. Складність та обсяг мережі (ключова проблема 1). Цей розділ пов'язаний з проблемами, що виникають через фізичні та логічні характеристики сучасних мереж. Сюди входить масштабність і різноманітність тому що є проблема моніторингу та управління великими, географічно розподіленими мережами (наприклад, глобальними корпоративними мережами чи хмарними середовищами).

Різноманіття технологій та протоколів пов’язано з тим, що є труднощі в уніфікації діагностичних процесів через широкий спектр використовуваних технологій (наприклад, Ethernet, Wi-Fi, 5G) та комунікаційних протоколів.

Віртуалізація та SDN/NFV (Software-Defined Networking / Network Function Virtualization) полягає в тому, що є складність відстеження шляху трафіку та ло-

калізації несправностей у віртуалізованих, програмно-визначених мережевих середовищах, де межі традиційного апаратного обладнання розмиті.

2. Інструменти та дані моніторингу (ключова проблема 2). Цей блок описує проблеми, пов'язані безпосередньо із засобами збору та аналізу інформації:

Інструментальне місто (Tool Sprawl) пов’язане з тим, що є надмірне використання великої кількості окремих, неінтегрованих інструментів для різних завдань (наприклад, окремо для моніторингу продуктивності, окремо для логів, окремо для безпеки).

Надлишок і "шум" даних вказує на те, що є проблема з обробкою занадто великого обсягу даних моніторингу (логів, метрик), серед якого важко виділити релевантні сигнали про реальні проблеми [5].

Відсутність єдиного уявлення – це про неможливість ефективно корелювати дані з різних джерел (мережеве обладнання, сервери, додатки) для формування цілісної картини інциденту.

Проблеми з пасивним моніторингом повязано з обмеженням моніторингу, який лише пасивно збирає інформацію, не дозволяючи проактивно імітувати дії користувачів чи тестувати мережу.

3. Діагностика продуктивності та QoS (ключова проблема 3). Цей блок стосується проблем, пов'язаних з якістю обслуговування (Quality of Service) та кінцевим досвідом користувача.

Виявлення "тонких" проблем пов’язано із складністю локалізації нерегулярних або мінімальних погіршень (наприклад, незначні затримки або періодичні втрати пакетів), які важко відтворити та діагностувати.

Ефект пальця, що вказує (Finger-Pointing) стосується міжфункціональних конфліктів, коли команди (мережевики, системні адміністратори, розробники додатків) звинувачують одна одну у виникненні проблем із продуктивністю.

Діагностика додатків спричинене складністю визначення, чи проблема полягає у мережевій інфраструктурі, чи у самому коді, конфігурації прикладного програмного забезпечення [26].

Шифрований трафік обумовлене неможливістю глибокої інспекції пакетів і детального аналізу їхнього вмісту (наприклад, для виявлення шкідливого програмного забезпечення чи проблем протоколу) через використання *SSL/TLS* шифрування.

4. Організаційні та людські фактори (додаткова проблема). Хоча не виділений як окрема гілка з вищим рівнем, цей блок часто є частиною розділу проблем і включає декілька проблема, які спричинені людським фактором.

Нестача досвіду та навичок повязана із дефіцитом кваліфікованих фахівців, здатних керувати та діагностувати нові, складні архітектури (наприклад, *SDN* та хмарні мережі).

Процедурні проблеми обумовлені відсутністю стандартизованих, документованих процедур для управління змінами, реагування на інциденти та усунення несправностей [27].

Проактивність проти реактивності виникає через домінування реактивних підходів (усунення проблеми після її виникнення) над проактивними (прогнозування та запобігання збоям).

5. Сучасні рішення та майбутні напрямки (фокус на інноваціях). Цей блок представляє способи подолання вищезазначених проблем за допомогою нових технологій:

Автоматизація діагностики пов’язана із використанням програмних інструментів для автоматичного виконання рутинних діагностичних завдань та збору даних [28].

Штучний інтелект та машинне навчання (AIOps) стосується застосування алгоритмів для обробки "шуму" даних, кореляції подій, виявлення аномалій і прогнозування потенційних збоїв.

Мережі на основі намірів (мережі на основі намірів) повязані з розробкою нової архітектури, де мережа самостійно налаштовується та оптимізується, що спрощує перевірку відповідності її стану заданим бізнес-вимогам.

Наскрізна видимість (End-to-End Visibility) досягається впровадженням інструментів, що забезпечують повний і уніфікований огляд шляху трафіку від кінцевого користувача до бекенд-сервісу, долаючи "Ефект пальця, що вказує".

Задачі пошуку та локалізації відмов відрізняються на різних етапах розробки та експлуатації мережі [29]. На етапі архітектурного проектування головною задачею є розробка системи вбудованих засобів контролю та діагностики, які забезпечують стійкість до відмов, а також самовідновлення технічних та програмних компонентів. На стадії реалізації архітектурних рішень, які включають налаштування взаємодії окремих підсистем, задачею мережної діагностики є перевірка відповідності їх функціонування заданим специфікаціям. В процесі експлуатації мережі діагностичні задачі зводяться до стеження за рівнем завантаженості різних компонентів, їх ефективної роботи та пропускної спроможності, а також до прогнозування змін архітектурних рішень, або проведенню більш детального автономного дослідження окремих компонентів [6].

1.3. Класифікація та порівняльний аналіз сучасних засобів аналізу трафіку

Облік мережевого трафіку є актуальним, і для його реалізації існує ряд програмних і технічних засобів. Зокрема, він реалізується в комерційних цілях при наданні послуг доступу до мережі. Однак зібрані при цьому дані не завжди можна вважати досить об'єктивними, так як обидві сторони - постачальники послуг і абоненти мережі - прагнуть змістити показники трафіку в свою користь або ж іноді невірно ідентифікувати об'єкт свого інтересу до мережі. На підставі даних про трафік в багатьох випадках можуть бути зроблені висновки про фактори, що визначають активність користувачів, а також про об'єкти їх найбільшого інтересу. Таким чином, облік мережевого трафіку фактично є частиною політики щодо забезпечення інформаційної та економічної безпеки фірм і організацій [7].

Аналіз мережевого трафіку (Network Traffic Analysis, NTA) є основою для забезпечення продуктивності, безпеки та надійності будь-якої сучасної комп'ютерної мережі. Інструменти NTA – це програмні та апаратні рішення, призначені

для захоплення, декодування, агрегування та інтерпретації даних, що передаються мережею. Через різноманіття мережевих завдань та технологій, ці інструменти класифікують за декількома ключовими ознаками, що визначають їхню глибину аналізу, функціональне призначення та спосіб розгортання.

Рішення для моніторингу мережі дозволяють здійснювати безперервне спостереження за мережевою інфраструктурою в режимі реального часу.

Швидке виявлення та усунення проблем має вирішальне значення для забезпечення продуктивності та безпеки вашої мережі. Моніторинг допомагає виявити всі проблеми продуктивності, включаючи ті, які не можуть бути зафіксовані стандартними мережевими командами. Це дозволяє запобігати збоям, мінімізувати час простою та покращувати ефективність роботи.

Основні переваги моніторингу мережі:

- Виявлення кіберзагроз та злочинної діяльності. Моніторинг мережі дозволяє своєчасно виявляти кіберзагрози, такі як вторгнення, спам, фішингові атаки та інші види злочинної діяльності. Це допомагає запобігти потенційним атакам і захистити мережу від шкідливих дій. Завдяки аналізу трафіку можна ідентифікувати підозрілі активності та запроваджувати відповідні заходи захисту.
- Підвищення ефективності мережі. Мережевий моніторинг допомагає оптимізувати використання ресурсів, аналізуючи пропускну здатність і навантаження на мережу. Він дозволяє визначити, які пристрої або сервіси споживають найбільше трафіку, та запровадити заходи для покращення продуктивності. Це забезпечує рівномірний розподіл ресурсів і підвищує швидкість роботи мережі.
- Відновлення роботи мережі після збоїв. У разі відмови обладнання або програмного забезпечення моніторинг дозволяє швидко ідентифікувати проблему та оперативно її усунути. Це скорочує час простою й допомагає мінімізувати негативний вплив на бізнес-процеси.
- Покращення безпеки мережі. Моніторинг допомагає виявляти вразливості в мережевій інфраструктурі та вчасно їх усувати, запобігаючи можливим кібератa-

кам. Регулярний аналіз мережевого трафіку дозволяє визначати потенційні точки входу для зловмисників і посилювати

захист.

- Підтримка прийняття рішень. Мережевий моніторинг надає цінні аналітичні дані, які допомагають приймати обґрунтовані рішення щодо конфігурації мережі, вибору обладнання та відповідності вимогам безпеки. Завдяки зібраній інформації можна покращити керування мережевою інфраструктурою та оптимізувати її роботу.

- Виявлення вторгнень у реальному часі. Коли виникає загроза, моніторинг мережі дозволяє негайно відреагувати та вжити заходів для припинення шкідливої активності. Аналіз мережевого трафіку дозволяє виявити джерело атаки та вжити необхідних заходів для захисту від майбутніх інцидентів.

- Скорочення часу простою, виявлення проблем до їх виникнення. Простій через збій системи може бути дорогим і призводити до значних фінансових втрат. Використання технології моніторингу мережі допомагає зменшити час простою, адже потенційні проблеми можна виявити та усунути до того, як вони спричинять серйозні збої в роботі.

- Захист організації від кібератак. Зі зростанням кількості кіберзагроз хакери постійно шукають слабкі місця в мережах компаній. Використовуючи інструменти моніторингу, можна ідентифікувати слабкі місця, що ускладнює несанкціонований доступ для зловмисників.

- Контроль над мережею. Моніторинг забезпечує детальну видимість мережевого трафіку, дозволяючи контролювати дії користувачів, обмежувати доступ до певних ресурсів і блокувати небажані підключення. Це сприяє кращому керуванню мережею та підвищенню рівня безпеки.

- Зменшення експлуатаційних витрат. Моніторинг допомагає виявляти та усувати проблеми ще до того, як вони призведуть до серйозних збоїв або потребуватимуть значних фінансових витрат на їх усунення. Також це сприяє ефективнішому використанню ресурсів, що знижує загальні витрати на підтримку мережі.

- Запобігання втраті даних через зловмисні атаки. Один із ключових аспектів безпеки – захист даних від несанкціонованого доступу. Моніторинг дозволяє виявляти спроби хакерських атак, підозрілі дії та витоки інформації. Використання сучасних інструментів моніторингу допомагає підтримувати актуальний рівень безпеки [8].

Класифікація інструментів аналізу трафіку

1. За принципом роботи (глибина аналізу). Аналізатори повних пакетів (також відомі як Packet Sniffers або Packet Capturers) – це найбільш потужні інструменти в арсеналі мережевого інженера та спеціаліста з кібербезпеки. Вони представляють найвищий ступінь деталізації в класифікації засобів аналізу трафіку за принципом роботи.

Повний пакет – це кожен біт даних, який подорожує мережею, включаючи заголовки всіх рівнів (від фізичного до прикладного) та корисне навантаження (payload) – фактичний вміст, що передається.

Принцип дії полягає в тому, що спочатку відбувається захоплення, інструмент переводить мережеву карту в нерозбірливий режим (*promiscuous mode*), що дозволяє їй захоплювати не лише пакети, адресовані безпосередньо цій машині, а й увесь трафік, що проходить через сегмент мережі (за умови використання SPAN-порту або мережевого TAP). Захоплені пакети зберігаються у спеціальному бінарному форматі (наприклад, .pcap або .cap). Саме це повне зберігання вмісту відрізняє їх від моніторів потоків (NetFlow).

Потім відбувається декодування та інтерпретація, де програмне забезпечення (аналізатор) може декодувати та візуалізувати кожен пакет, розбиваючи його на компоненти відповідно до мережевих протоколів (IP, TCP, HTTP, DNS тощо) на всіх семи рівнях моделі OSI.

Класифікація, яка представлена нижче ґрунтується на тому, який обсяг інформації про трафік інструмент захоплює та обробляє, що безпосередньо впливає на його функціональність і вимоги до ресурсів.

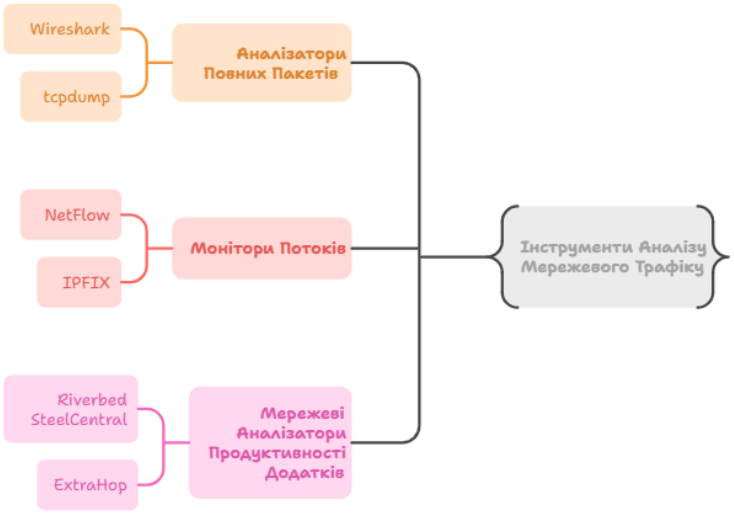


Рис. 1.12 Класифікація інструментів аналізу трафіку за принципом роботи

Wireshark – один із найпопулярніших інструментів для аналізу мережевого трафіку. Він дозволяє перехоплювати пакети та детально їх аналізувати, забезпечуючи інтуїтивно зрозумілий графічний інтерфейс [10]. Також цей інструмент здатний декодувати та візуалізувати тисячі протоколів. Дозволяє застосовувати складні фільтри, відстежувати повні TCP-сесії та аналізувати затримки на рівні пакетів та застосовується для ручної діагностики складних несправностей, налагодження протоколів, вивчення мережевих атак.

Tcpdump – консольний (CLI) інструмент захоплення пакетів, поширений у середовищах Unix/Linux. Він ефективно захоплює пакети у файл (.pcap) для подальшого аналізу (часто за допомогою Wireshark) або відображає їх у консолі. Використовується для швидкого, точкового аналізу на серверах, для захоплення трафіку на віддалених серверах, моніторингу активності на мережевих шлюзах.

NetFlow – протокол, розроблений компанією Cisco, який є найпоширенішим стандартом для збору інформації про мережеві потоки. За допомогою нього фіксуться ключові характеристики кожного потоку: IP-адреси, порти, протокол, час початку/завершення та обсяг даних. Він використовується для моніторингу використання смуги пропускання, виявлення "найбільших споживачів" (top talkers), планування навантаження та білінг [30].

IPFIX (IP Flow Information Export) – стандартизований IETF протокол, який є наступником NetFlow v9. На відміну від NetFlow, IPFIX є більш гнучким і дозволяє експортувати додаткові, довільно визначені поля даних, не обмежуючись фіксованим набором. Застосовується так само, як і NetFlow, але з ширшими можливостями налаштування та розширення.

2. За призначенням (функціональна класифікація). Класифікація за функціональністю визначає, які завдання інструмент розв'язує насамперед.

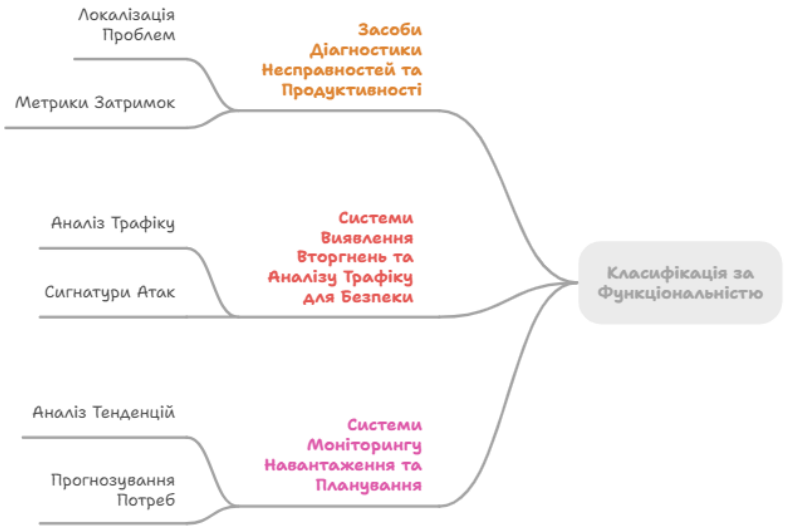


Рис. 1.13 Класифікація інструментів аналізу трафіку за призначенням

Ця класифікація визначає, які конкретні завдання та бізнес-цілі покликаний вирішувати той чи інший інструмент.

Засоби діагностики несправностей та продуктивності орієнтовані на оперативну підтримку мережі та забезпечення якості обслуговування (QoS). Їхня головна мета — швидко визначити, чому мережа або додаток працює повільно чи має збої, визначити точне місце виникнення несправності: чи це фізичний обрив, помилка маршрутизації, занадто велике навантаження на комутаторі, чи проблема на кінцевому сервері. Це дозволяє скоротити час простою (Mean time to repair, MTTR).

Системи виявлення вторгнень та аналізу трафіку для безпеки використовують мережевий трафік як джерело інформації про кіберзагрози. Вони є критично важливими компонентами інфраструктури безпеки (наприклад NIDS/NDR). Ін-

струменти постійно моніторять трафік, шукаючи аномалії або зловмисну активність. На відміну від діагностичних засобів, вони не шукають, чому повільно, а шукають, чому небезпечно. Інструменти порівнюють захоплений трафік із базами даних відомих зразків шкідливого програмного забезпечення, експлойтів та атак. Коли виявляється збіг з сигнатурою (наприклад, спроба використання відомої уразливості HTTP-протоколу), система генерує попередження або автоматично блокує трафік (у випадку систем NDR).

Системи моніторингу навантаження та планування допомагають менеджерам мережі приймати рішення щодо розвитку інфраструктури та розподілу ресурсів. Інструменти збирають і зберігають історичні дані про використання смуги пропускання та обсяги трафіку (зазвичай через NetFlow/IPFIX). Аналіз статистики дозволяє зрозуміти, як змінюється навантаження мережі з часом.

3. Вибір типу розгортання впливає на вартість, масштабованість та простоту обслуговування. Ця класифікація визначає, як інструмент фізично інтегрується в інфраструктуру, а також впливає на його вартість, продуктивність та гнучкість.

Рис. 1.14 Класифікація інструментів аналізу трафіку за типом розгортання

Апаратні пристрої – це виділене фізичне обладнання (сервери або спеціалізовані пристрої), які містять апаратне та програмне забезпечення, оптимізоване для аналізу трафіку. Апаратне забезпечення часто включає спеціалізовані мережеві карти та оптимізовану пам'ять, що дозволяє захоплювати та обробляти трафік на високих швидкостях (10 Гбіт/с, 40 Гбіт/с і вище) без втрати пакетів. Вони використовуються на критично важливих високошвидкісних магістралях (backbone),

у великих дата-центрах або на межі мережі (WAN edge), де втрата навіть частки трафіку є неприпустимою. Прикладом апаратним рішень є рішення від Gigamon, NetScout, спеціалізовані апаратні Probe від Riverbed, власні апаратні платформи NDR.

Програмні інструменти аналізу трафіку можуть бути безкоштовними (наприклад, nTop або Wireshark) або платними. Останні характеризуються ширшими функціональними можливостями, і навіть клієнти комерційних коштів можуть скористатися підтримкою постачальника. Очевидно, що підприємствам і розвиненим користувачам краще використовувати платні версії. Для початку проектування системи моніторингу мережі рекомендується провести аналіз середовища та додатків у якому вони будуть запускатися та скласти список поточних та майбутніх вимог до моніторингу інфраструктури. Чи будуть це інструменти для класичної мережевої архітектури розгортання мережі або це буде “хмара” - залежить від оцінки бюджету та можливостей інвестувати в моніторинг [9]. Програмні рішення ідеально підходять для віртуальних і приватних хмарних середовищ, локальних мереж (LAN), а також для менших компаній або сегментів, які не вимагають максимальної швидкості.

Прикладом програмних інструментів є Wireshark, tcpdump, Zeek/Suricata (встановлені на віртуальних серверах), VMware vRealize Network Insight.

Нарешті, існують хмарні рішення моніторингу мережі, які надають можливість віддаленого моніторингу трафіку без потреби встановлення додаткового обладнання чи програмного забезпечення. Ці рішення дозволяють адміністраторам моніторити мережу з будь-якої точки з доступом до Інтернету, що зручно для дистанційного адміністрування та моніторингу мережі [10]. Хмарні рішення підходять для гібридних і мультихмарних інфраструктур. Вони забезпечують єдиний центр управління, здатний збирати та корелювати дані з локальної мережі, приватних і публічних хмар одночасно. Це дозволяє забезпечити наскрізну видимість. Хмарні рішення використовують компанії, що активно використовують хмарні обчислення та потребують моніторингу трафіку між різними хмарними провайде-

рами та власними дата-центрами. Прикладом хмарних сервісів є ThousandEyes (Cisco), AppDynamics (Cisco), Kentik (аналіз NetFlow/Flow в хмарі), ExtraHop (хмарні сенсори).

1.4. Огляд протоколів і технологій для моніторингу та збору даних

Ефективна діагностика та забезпечення безпеки комп'ютерних мереж неможливі без якісного збору даних про трафік. Протоколи та технології, що використовуються для цього, діляться на дві основні категорії: ті, що захоплюють повний вміст пакетів, і ті, що збирають метадані (потоки). Вибір методу визначає глибину аналізу, масштабованість рішення та його економічну ефективність.

Рис. 1.15. Протоколи збору метаданих трафіку (Flow Data)

NetFlow — це протокол, який забезпечує видимість мережевого трафіку шляхом збору та експорту інформації про IP-потоки в мережі. Він включає такі деталі, як IP-адреси джерела та призначення, порти, типи протоколів та обсяги передачі даних [11]. Більшість сучасних маршрутизаторів і комутаторів підтримують NetFlow або подібні протоколи, такі як sFlow або IPFIX. Для маршрутизаторів, що обробляють транзитний IP-трафік, достатній обсяг процесора і пам'яті є критично важливим для керування вибіркою потоку разом із завданнями маршрутизації. У середовищах з високою про-

пускною здатністю - обробка трафіку в діапазоні декількох гігабіт на секунду - часто необхідне спеціальне обладнання для моніторингу або надійні колектори на основі програмного забезпечення.

Для отримання, зберігання та обробки експортованих записів потоку використовують колектор NetFlow. Колектори повинні мати достатній обсяг пам'яті, щоб зберігати історичні дані і підтримувати аналіз тенденцій. Для мереж з терабайтами щомісячного трафіку необхідно використовувати колектори з декількома терабайтами пам'яті і можливостями швидкого дискового вводу/виводу, щоб ефективно обробляти навантаження.

Після налаштування конфігурації наступним кроком є збір та обробка даних NetFlow для отримання цінної інформації. Це передбачає експорт записів потоку, управління великими обсягами даних та інтеграцію аналітичних інструментів для змістовного аналізу.

Експорт даних NetFlow вимагає злагодженої роботи мережевих пристроїв і колекторів, щоб забезпечити безперебійну роботу і точність даних. Маршрутизатори і комутатори постійно генерують записи потоку, і важливо оптимізувати спосіб експорту цих записів, щоб впоратися з великими обсягами трафіку, уникаючи при цьому втрати даних.

Оскільки NetFlow часто використовує UDP для експорту даних, під час перевантаження мережі може відбуватися втрата пакетів. Щоб зменшити цей ризик, налаштуйте пристрої на надсилання записів потоку за кількома адресатами. Така надмірність гарантує, що якщо один колектор вийде з ладу, дані все одно зможуть потрапити до іншої системи. Багато адміністраторів використовують первинні та вторинні колектори, розташовані в різних регіонах, щоб підтримувати видимість даних під час збоїв.

Також важливо виділити достатньо буферного простору на мережевих пристроях, щоб впоратися зі сплесками трафіку без втрати записів потоку. Крім того, слід дотримуватися балансу з часом експорту - дані слід надсилати швидко, щоб їх було видно, але без перевантаження системних ресурсів. Зазвичай, потоки екс-

портуються, коли у них закінчується тайм-аут або коли заповнюється кеш потоку [13].

Щоб перетворити дані NetFlow на практично корисну інформацію, необхідно інтегрувати їх із системами аналітики. Такі інструменти забезпечують безперервний моніторинг мережі й дозволяють оперативно реагувати на будь-які відхилення або підозрілі зміни в трафіку. Проте ефективна інтеграція потребує уваги до сумісності форматів, продуктивності обробки даних і підтримки роботи в реальному часі.

Деякі аналітичні платформи можуть напряму підключати базу даних до NetFlow-колекторів і здійснювати аналіз у режимі реального часу. Інші — використовують періодичний експорт даних у форматах CSV чи JSON. Вибір підходу залежить від вимог до швидкості моніторингу, звітності та обсягів даних.

Щоб зробити потік NetFlow більш інформативним, доцільно збагачувати його зовнішніми джерелами — наприклад, записами DNS, даними WHOIS чи каналами розвідки загроз (Threat Intelligence). Це допомагає додати контекст до IP-адрес і доменів. Також геолокаційна інформація та дані з автономних систем можуть бути корисними для виявлення маршрутів і закономірностей у трафіку.

Потокова аналітика в реальному часі дозволяє одразу реагувати на події в мережі або інциденти безпеки. Обробляючи трафік у момент його надходження, можна налаштувати автоматичне виявлення аномалій, контроль порогових значень і сповіщення. Це особливо важливо для швидкої ідентифікації загроз чи несправностей маршрутизації.

Інформаційні панелі (дашборди) допомагають представити результати аналізу у зручній візуальній формі. Для мережевих інженерів важливі детальні графіки трафіку й пропускної здатності в реальному часі, тоді як керівники зазвичай віддають перевагу оглядовим зведенням про загальну ефективність та довгострокові тенденції.

Із ростом обсягів даних зростає й потреба в оптимізації аналітичних запитів. Індексування ключових полів — IP-адрес, протоколів, часових позначок — а та-

кож попереднє агрегування основних метрик суттєво підвищує швидкодію аналітичних систем і зменшує навантаження на інфраструктуру.

Після того, як моніторинг і оповіщення будуть впроваджені, необхідно зібрати ключові показники в таблиці, щоб зробити аналіз швидшим і більш дієвим.

Source IP	Destination IP	Protocol	Bytes Transferred	% of Total Traffic
192.168.1.100	203.0.113.50	TCP/443	2.3 GB	15.2%
10.0.0.45	198.51.100.25	TCP/80	1.8 GB	11.7%
172.16.0.200	203.0.113.75	UDP/53	890 MB	5.8%

Рис. 1.16. Ключові показники трафіку

Такі таблиці висвітлюють ключові моделі трафіку, наприклад, які IP-адреси споживають найбільше пропускної здатності, задіяні протоколи та тривалість потоків. Крім того, таблиці розподілу протоколів і порівняння продуктивності можуть виявити зміни з часом і потенційні проблеми в різних сегментах мережі. Зосередьтеся на показниках, які впливають на прийняття рішень, таких як процентні зміни або часові тенденції, щоб доповнити ваші системи виявлення аномалій і оповіщення [13].

Отже, NetFlow є ключовим протоколом для масштабованого моніторингу мережі, оскільки він замінює ресурсомістке захоплення повних пакетів на ефективний збір метаданих, де кожен унікальний потік трафіку ідентифікується за основними параметрами (IP-адреси, порти та протокол). Еволюція протоколу від фіксованого формату NetFlow v5 до розширюваного шаблонного формату NetFlow v9 забезпечила його адаптивність і це дозволяє успішно використовувати технологію для широкого спектру завдань, включаючи моніторинг навантаження, точний білінг та проактивне виявлення мережевих аномалій.

IPFIX — це стандартизована версія NetFlow, визначена IETF (Internet Engineering Task Force). Цей протокол забезпечує загальний формат для експорту

інформації про потік, що робить його сумісним з різними мережевими пристроями та постачальниками [12]. Він функціонально є прямим наступником та розширенням пропрієтарного протоколу NetFlow v9 компанії Cisco і відіграє критично важливу роль у сучасному широкомасштабному моніторингу мережевого трафіку [31].

IPFIX, як і NetFlow, не захоплює вміст кожного пакету, а збирає метадані про кожен мережевий потік (Flow). Потік — це односпрямована послідовність пакетів, що має спільні ключові атрибути. Хоча базові атрибути збігаються з NetFlow (вихідна/цільова IP-адреса, порти, протокол), IPFIX фокусується на гнучкості.

Рис. 1.17. Протокол IPFIX

Процес роботи IPFIX складається з експортера, який є пристрієм (маршрутизатор, комутатор, зонд), що спостерігає за трафіком, агрегує дані потоку та створює запис IPFIX; колектора, тобто центрального сервера, що приймає, зберігає та обробляє записи IPFIX для подальшого аналізу і записів потоку. Записи потоку – дані про потік, які надсилаються від експортера до колектора. Головна перевага IPFIX над старими версіями NetFlow полягає у його шаблонному форматі та відкритій структурі [32].

Основна відмінність IPFIX, успадкована від NetFlow v9, полягає у використанні шаблонів. Шаблони – це спеціальні записи, які визначають структуру та типи даних, що будуть міститися у подальших записах потоку. Наприклад шаблон

може вказувати, що запис складається з вихідної IP-адреси (4 байти), цільового порту (2 байти) та обсягу байтів (8 байтів [33]).

Експортер спочатку надсилає колектору цей шаблон. Далі він надсилає лише записи даних, посилаючись на номер шаблону. Це дозволяє колектору точно декодувати дані, не отримуючи опис структури з кожним пакетом. Якщо необхідно почати збирати нові дані, експортер просто створює новий шаблон і надсилає його колектору. Це забезпечує динамічне розширення без необхідності оновлювати весь протокол (як це було при переході від NetFlow v5 до v9). Експортер постійно спостерігає за мережевим трафіком. Коли пристрій бачить перший пакет нового потоку, він ініціює створення запису. При проходженні наступних пакетів, що належать до того ж потоку, експортер оновлює лічильники (кількість пакетів, загальний обсяг байтів) та час активності. Потік вважається завершеним і експортується, коли у потоці не було пакетів протягом певного часу, потік триває надто довго (запобігає надто великим записам) і коли відбулося явне завершення**,** наприклад, отримання пакетів FIN або RST у TCP-сесії [34].

Завдяки своїй гнучкості та надійності IPFIX використовується для більш складних завдань, ніж традиційний моніторинг смуги пропускання. Він забезпечує єдиний формат збору даних, незалежно від виробника мережевого обладнання. IPFIX гарно підходить для збору інформації в середовищах SDN/NFV, де необхідне відстеження потоків між віртуальними машинами та контейнерами. Він дозволяє експортувати не лише базові дані, а й додаткові метадані, які можуть свідчити про небезпечну активність (наприклад аномальне використання протоколів чи нетипові обсяги даних). Завдяки надійному транспортному протоколу та можливості детального експорту, IPFIX використовується для точного розрахунку вартості трафіку.

У той час як NetFlow та IPFIX домінують як універсальні стандарти для збору мережевих метаданих (Flow Data), існують альтернативні та пропрієтарні протоколи, які пропонують унікальні переваги або використовуються як аналоги

на обладнанні конкретних виробників. Ці альтернативи часто оптимізовані для високої продуктивності або мають вузькоспеціалізовані функції.

sFlow є одним із найвідоміших альтернативних протоколів, що відрізняється від NetFlow/IPFIX фундаментальним принципом роботи – методом вибірки. На відміну від NetFlow, який, по суті, намагається проаналізувати кожну сесію, sFlow обробляє лише невелику, випадково обрану частину пакетів.

Принцип полягає в тому, що адміністратор задає коефіцієнт вибірки (наприклад 1:1000). Це означає, що мережевий пристрій (комутатор або маршрутизатор) створює запис потоку лише для кожного тисячного пакету, що проходить. Потім відбувається запис, який містить метадані (заголовки пакетів) для обраного пакету, а також статистику лічильників інтерфейсу (використання смуги пропускання, помилки). Цей запис надсилається на колектор sFlow. Колектор sFlow використовує статистичні алгоритми для екстраполяції отриманої вибірки на весь обсяг трафіку.

jFlow – протокол для збору метаданих мережевого трафіку, розробленим компанією Juniper Networks. За своєю суттю, це прямий функціональний аналог технології NetFlow від Cisco, призначений для використання на маршрутизаторах та комутаторах Juniper. Як і NetFlow, jFlow фіксує ключові атрибути кожного мережевого потоку, такі як IP-адреси джерела та призначення, номери портів, протоколи та обсяг переданих даних. Це дозволяє пристроям Juniper ефективно генерувати та експортувати агреговану інформацію про трафік, уникаючи ресурсомісткого захоплення повних пакетів.

Основна функціональність jFlow повністю відповідає цілям моніторингу, притаманним іншим Flow-технологіям. Протокол використовується для забезпечення детальної видимості мережі, що критично важливо для цілей моніторингу трафіку та аналізу навантаження на інтерфейсах. Зібрані дані дають змогу мережевим адміністраторам визначати тенденції використання смуги пропускання, ідентифікувати "найбільших споживачів" (top talkers) та проводити базове вияв-

лення аномалій, що можуть свідчити про DoS-атаки чи неправильну поведінку мережевих програм.

Компанія Juniper, як і інші великі виробники, поступово інтегрує у своє обладнання підтримку відкритого стандарту IPFIX. Це означає, що хоча jFlow залишається доступним для забезпечення зворотної сумісності та специфічних потреб Juniper, сучасні мережеві архітектури все частіше переходять до використання IPFIX [35]. Це дозволяє користувачам Juniper безперешкодно інтегрувати свої пристрої у гетерогенні мережі, використовуючи універсальні системи збору та аналізу метаданих, які не залежать від конкретного виробника обладнання.

Протокол NetStream є фірмовою технологією, розробленою компанією Huawei Technologies, що виконує ту саму фундаментальну функцію, що й NetFlow від Cisco чи jFlow від Juniper: збір та експорт метаданих про мережеві потоки. Суть NetStream полягає в агрегації ключових параметрів трафіку (наприклад IP-адреси, порти та протокол) на маршрутизаторах та комутаторах Huawei. Завдяки цьому механізму, мережеві пристрої можуть ефективно генерувати стислі звіти про активність, замість того, щоб перехоплювати та обробляти кожен пакет.

Основна функціональність NetStream охоплює ключові потреби мережевого менеджменту, забезпечуючи моніторинг трафіку та планування потужності у мережах Huawei. Зібрані метадані дозволяють адміністраторам отримувати комплексну картину використання мережевих ресурсів, ідентифікувати джерела найбільшого навантаження та аналізувати розподіл трафіку між різними додатками та користувачами. Це критично важливо для оптимізації продуктивності та забезпечення того, щоб ресурси мережі відповідали поточним та прогнозованим бізнес-вимогам.

Як і Flow-рішення, NetStream створює необхідність забезпечення сумісності у гетерогенних мережах. Хоча він є стандартним інструментом у екосистемі Huawei, зовнішні системи моніторингу повинні мати підтримку його специфічного формату. Сучасне обладнання Huawei також підтримує відкритий стандарт IPFIX, що дозволяє інтегрувати його дані з універсальними колекторами та аналі-

тичними платформами, долаючи обмеження власницьких протоколів і забезпечуючи єдиний підхід до моніторингу.

Для забезпечення надійності, продуктивності та безпеки сучасних мережевих інфраструктур необхідним є застосування ефективних технологій моніторингу та діагностики. Для збору інформації про мережеву активність протоколи розділяються на основні групи, виходячи з принципу роботи та функціонального призначення.

Рис. 1.18. Класифікація протоколів моніторингу

SNMP (простий протокол керування мережею) є найважливішим протоколом для активного моніторингу працездатності мережевих пристроїв. Він дозволяє адміністраторам централізовано збирати оперативну інформацію та керувати пристроями.

Протокол SNMP використовує модель "менеджер-агент". Агентом виступає програмне забезпечення, вбудоване в кожен мережевий пристрій (маршрутизатор, комутатор, сервер). Менеджером є центральна станція моніторингу, яка надсилає запити до агентів.

Ключовим елементом SNMP є MIB (Management Information Base – база керувальної інформації). Це ієрархічна структура даних (подібна до дерева), що містить об'єкти (OID), які представляють конфігурацію та статистику пристрою (наприклад поточний рівень завантаження процесора, кількість вхідних пакетів на інте-

рфейсі). Менеджер використовує команди GET та SET для взаємодії з цими об'єктами.

SNMP надає важливі метрики для оцінки стану пристрою. Він вимірює використання CPU та пам'яті пристрою. Високі показники можуть свідчити про перевантаження або циклічні помилки конфігурації. Також в нього є можливість збору даних про обсяг трафіку (вхідного/вихідного), а також про кількість помилок (CRC Errors, Collisions, Discards). Велика кількість помилок на інтерфейсі часто вказує на фізичну проблему (несправний кабель, неправильний duplex).

Команди GET є синхронними (потребують запиту). Але для критичних ситуацій використовується механізм SNMP Traps. Trap – це асинхронне повідомлення, яке агент негайно надсилає менеджеру, коли на пристрої відбувається важлива подія (наприклад, збій живлення, вихід інтерфейсу з ладу, перевищення порогового значення завантаження CPU). Traps дозволяють системі моніторингу проактивно реагувати на критичні події, не чекаючи наступного планового запиту GET.

Syslog – це стандартизований, простий та універсальний протокол для ведення журналів подій. Syslog дозволяє будь-якому пристрою, від маршрутизатора до операційної системи, відправляти свої повідомлення, попередження та помилки на центральний сервер-колектор Syslog. Syslog фіксує всі зміни конфігурації та помилки, пов'язані з нею (наприклад "Не вдалося застосувати ACL", "Збій автентифікації"). Журнали Syslog фіксують спроби входу, успішні та невдалі авторизації, події фаєрвола (блокування трафіку), що є основою для систем SIEM (Security Information and Event Management).

ICMP (протокол керування повідомленнями Інтернету) не є протоколом збору даних у прямому сенсі, але він є базовим інструментом діагностики доступності та шляху трафіку.

Протокол ICMP використовує Ping для швидкої перевірки зв'язності та базової продуктивності мережі. Його механізм роботи простий: він надсилає запит ICMP Echo Request до цільового хоста і очікує відповіді ICMP Echo Reply. Голов-

на роль Ping полягає у вимірюванні доступності пристрою (показуючи, чи він активний) та визначенні часу відгуку (Round Trip Time, RTT), який є показником базової затримки мережі. Оскільки це найпростіший і найшвидший спосіб підтвердити або спростувати проблему зв'язку на мережевому рівні, Ping завжди є першою перевіркою, яку виконують інженери при виникненні будь-якої мережевої несправності [36].

Ping підтверджує лише наявність з'єднання та базову затримку, але інший фундаментальний інструмент Traceroute (або tracert) використовує ICMP-повідомлення для визначення повного шляху проходження трафіку, що є наступним кроком у діагностиці. Механізм Traceroute послідовно відправляє пакети, маніпулюючи їхнім полем Time-to-Live (TTL), і на кожному етапі отримує від маршрутизатора відповідь "час життя пакета вичерпано" (TTL Exceeded). Це дозволяє йому відобразити всі проміжні вузли (хопи).

Основна роль Traceroute – локалізація місця збою: якщо трафік зупиняється на певному маршрутизаторі, адміністратор точно знає, де шукати проблему, а також отримує можливість виміряти затримку на кожному окремому сегменті шляху, що є безцінним для виявлення "вузьких місць" у мережі.

Таким чином, у контексті комплексної діагностики, ці протоколи функціонують як система швидкого реагування. Їхня спільна робота мінімізує час, необхідний для ізоляції та локалізації проблеми, що робить їх важливими для будь-якої стратегії управління продуктивністю та забезпеченням надійності корпоративної мережі [37].

2. ТЕХНОЛОГІЇ ТА АЛГОРИТМИ МЕРЕЖЕВОЇ ДІАГНОСТИКИ

2.1. Моделі мережевих відмов

Мережеві відмови – це порушення або припинення нормального функціонування мережевих компонентів, що призводить до деградації, зниження доступності або повної недоступності мережевих сервісів для кінцевих користувачів. Мережеві відмови можуть бути класифіковані за різними критеріями, що допомагає точно визначити їхню першопричину та розробити відповідну стратегію відновлення.

Рис. 2.1. Класифікація мережевих відмов за природою виникнення

Апаратні відмови пов'язані з фізичним збоєм обладнання, що призводить до порушення його працездатності. Збій живлення відбувається через нестабільність або повне припинення подачі електроенергії, що є однією з найбільш поширених причин відключення мережевих вузлів [38]. Перегрів відбувається через надмірне тепловиділення, спричинене недостатнім охолодженням або високим навантаженням, що викликає нестабільну роботу або автоматичне аварійне відключення компонентів (CPU, ASIC).

Програмні відмови мають логічний характер і пов'язані з дефектами у програмному забезпеченні, що керує мережевим обладнанням. Витоки пам'яті – це ситуація, коли процес не звільняє раніше виділену пам'ять, що з часом

призводить до повного вичерпання системних ресурсів і, як наслідок, до уповільнення роботи або аварійного перезавантаження пристрою. Баги в операційній системі повязані з критичними помилками у мікропрограмі або мережевій операційній системі (наприклад, Cisco IOS, Juniper JunOS), що можуть викликати несподівану поведінку, циклічні помилки чи аварійне завершення процесів.

Людський фактор охоплює помилки, спричинені діями мережевих інженерів або операційного персоналу, і є однією з найбільш частих причин інцидентів. Неправильна конфігурація пов’язана з введенням некоректних команд, що призводять до логічних збоїв у роботі мережі (наприклад, помилки у маршрутизації, неправильні списки доступу, невідповідність параметрів тунелів). Випадкове відключення – це фізична помилка, така як ненавмисне відключення кабелю живлення, лінії зв'язку або неправильне обслуговування.

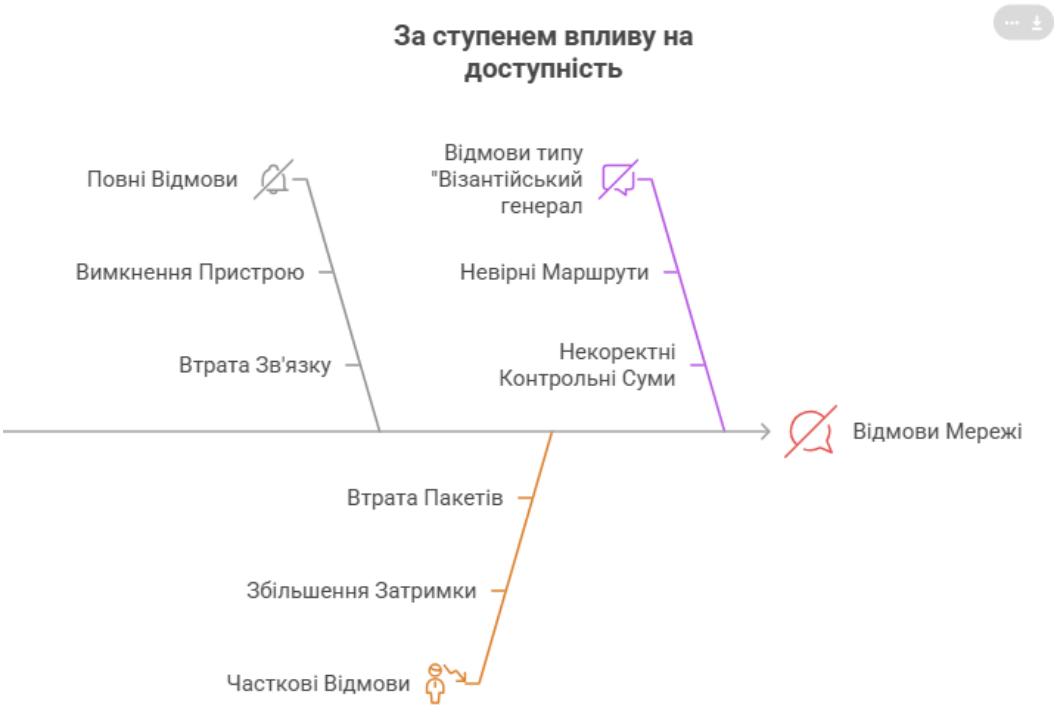


Рис. 2.2. Класифікація мережевих відмов за ступенем впливу на доступність

Повні відмови – це тип відмов, який характеризується повним припиненням роботи компонента або зв'язку, що робить його нездатним передавати чи обробляти трафік. Вимкнення пристрою пов’язано з фізичним або логічним

припиненням роботи мережевого вузла (маршрутизатора, комутатора). Це призводить до миттєвої втрати всіх сервісів, які він надавав.

Втрата зв'язку пов’язана з повною деградацією фізичного каналу (наприклад, обрив кабелю або відмова трансивера), внаслідок чого сусідні пристрої фіксують відсутність зв'язку на порту [39]. Повні відмови є найпростішими для виявлення, оскільки вони явно відображаються у системах моніторингу як зміна стану на "Down" (неактивний).

Часткові відмови – ці відмови є більш підступними, оскільки пристрій або зв'язок формально залишається доступним, але його продуктивність значно погіршується. Збільшення затримки пов’язано з різким зростанням часу проходження пакетів через мережу, що критично впливає на додатки реального часу (VoIP, відео). Це часто є наслідком перевантаження або виснаження ресурсів (CPU, пам'ять). Втрата пакетів вібувається тоді, коли пристрій не може обробити або переслати весь вхідний трафік, починаючи відкидати частину пакетів (наприклад через переповнення буферів). Це призводить до необхідності повторної передачі та погіршує якість сервісу.

Часткові відмови складніше діагностуються, оскільки вони вимагають постійного моніторингу метрик продуктивності, а не лише статусу доступності.

Існує кілька можливих причин, через які розподілена комп'ютерна система може впасти. Вони широко відомі як візантійські збої (також відомі як візантійські помилки). Візантійські збої - це, по суті, зрадники, які намагаються порушити зв'язок між лояльними генералами.

Застосовуючи цю концепцію до реальних комп'ютерних систем, це може бути програмна помилка, апаратна несправність і/або зловмисна атака. Іншими словами, візантійські збої не обов'язково повинні бути організованою атакою зловмисників. Це можуть бути просто проблеми, які заважають вузлам прийти до згоди щодо рішень для розподіленої мережі [15].

Відмови типу "Візантійський генерал" – це найбільш критичні та найважчі для виявлення відмови, оскільки вони порушують логічну цілісність мережі, а не

лише її фізичний стан. Пристрій, що не має фізичного збою, починає оголошувати некоректну або суперечливу інформацію про маршрутизацію (наприклад, оголошує себе найкоротшим шляхом до неіснуючої мережі). Це призводить до перенаправлення трафіку в "чорні діри" або створення маршрутних петель. Обладнання може передавати пакети з помилковими даними, що змушує приймаючий пристрій відкидати їх, імітуючи втрату пакетів.

Візантійські відмови можуть викликати ланцюгову реакцію, оскільки некоректна інформація поширюється протоколами, руйнуючи роботу всієї доменної області.

Проблема візантійських генералів моделює ситуацію, коли кілька вузлів у мережі мають домовитися про спільне рішення (наприклад, “передати дані”, “оновити конфігурацію”, “зберегти транзакцію”) навіть за умови, що частина вузлів поводиться ненадійно або зловмисно. Деякі вузли або маршрутизатори можуть збоїти, зависати, підміняти дані, або відсилати суперечливу інформацію, але система все одно повинна зберігати єдине узгоджене рішення для всіх надійних вузлів.

Нехай:

n – загальна кількість вузлів (процесів) у мережі;

f – кількість вузлів, що можуть поводитися візантійськи (тобто довільно, з помилками або навмисно);

$V = \{1, 2, ..., n\}$ – множина всіх вузлів;

$C \subseteq V$ – множина коректних вузлів, тобто $|C| = n - f$.

Кожен вузол $i \in V$ має початкове значення $v_i \in \{0,1\}$

Наприклад 0 = “не виконувати команду”, 1 = “виконати команду”.

Щоб система вважалася стійкою до візантійських відмов, рішення має задовільняти три умови:

1. **Agreement (узгодженість).**

   Всі правильні вузли мають прийти до однакового рішення:

$$\forall ij \in C, \quad d_i = d_j \tag{1}$$

2. Validity (достовірність).

Якщо всі правильні вузли почали з одного і того ж значення v, то воно і стане спільним рішенням:

$$\forall i \in C, (v_i = v) \Rightarrow (\forall i \in C, d_j = v) \quad (2)$$

3. Termination (завершення).

Кожен правильний вузол повинен дійти рішення за скінченне число кроків.

Для мереж без цифрових підписів (тобто коли зловмисний вузол може підробити повідомлення іншого):

$$n > 3f \quad (3)$$

або еквівалентно

$$f < \frac{n}{3} \quad (4)$$

Це означає: щоб гарантувати узгодження, кількість надійних вузлів повинна бути принаймні втричі більшою за кількість зламаних.

Для мереж **з** аутентифікацією (цифровими підписами**,** тобто коли неможливо підробити повідомлення):

$$n > 2f \quad (5)$$

або

$$f < \frac{n}{2} \quad (6)$$

Тут достатньо, щоб більшість вузлів була чесною.

Таблиця 2.1

Відображення на мережеві відмови

Тип відмови	Характеристика	Модель поведінки
Crash fault	Вузол перестає відповідати (вимкнувся або завис)	Втрата повідомлень
Omission fault	Деякі повідомлення губляться або не доходять	Часткова втрата даних
Timing	Повідомлення запізнюються або при-	Асинхронність

fault	ходять із затримкою	
Byzantine fault	Вузол діє довільно: підробляє, бреше, надсилає різним вузлам різні дані	Повна недовіра

Логічна модель процесу

Узгодження в мережі можна подати як послідовність раундів обміну повідомленнями:

$$m_i^{(r)} = f_i(s_i^{(r-1)}, M^{(r-1)}), \quad (7)$$

де:

$m_i^{(r)}$ — повідомлення, яке вузол iii відправляє на r-му раунді;

$M^{(r-1)}$) — набір повідомлень, отриманих у попередньому раунді;

$s_i^{(r)}$ — новий стан вузла після обробки повідомлень.

У візантійській ситуації для $i \notin C$:

$m_i^{(r)}$ =довільна функція (може порушувати протокол).

Необхідно знайти функцію прийняття рішення $D_i(M^{(r)})$, яка гарантує виконання умов agreement та validity.

Припустимо, у нас є мережа з n=10n вузлів. Якщо ми хочемо забезпечити стійкість до візантійських відмов без цифрових підписів:

$$f < \frac{10}{3} \Rightarrow f \leq 3 \quad (8)$$

Тобто максимум 3 вузли можуть поводитись неправильно і система все одно збереже спільне рішення.

У мережевих системах ця модель використовується для:

- розподілених баз даних (наприклад, Cassandra, etcd, Raft, PBFT);
- синхронізації конфігурацій маршрутизаторів або DNS-серверів;
- систем моніторингу, де вузли мають узгоджено визначити “чи є збій”.

Для стійкого до візантійських відмов розподіленого рішення необхідно виконати умову:

$$n = 3f + 1 \quad (9)$$

Таким чином, щоб витримати:

1 зламаний вузол → потрібно мінімум 4 вузли;

2 зламані вузли → мінімум 7;

3 зламані вузли → мінімум 10.

Математична модель візантійських генералів у мережевому контексті показує, яку кількість вузлів потрібно для досягнення узгодженості в умовах збоїв або атак. Вона лежить в основі сучасних протоколів відмовостійкості (BFT, PBFT, Paxos, Raft) і дозволяє формально описати, де межа між безпекою системи й неможливістю узгодження.

2.2. Показники якості обслуговування

В умовах постійного зростання обсягів трафіку реального часу принципи мережевої нейтральності виявилися недостатніми для гарантування якості послуг. Таким чином виникла потреба в розробці контрольованої площини даних, здатної здійснювати диференційоване управління ресурсами — процес, відомий як якість обслуговування (QoS).

QoS (Quality of service, укр. Якість обслуговування), у широкому значенні — якість послуг, які надає комунікаційна мережа. У вузькому технічному значенні в IT, цей термін означає - набір методів для управління ресурсами пакетних мереж.

QoS є необхідним для пакетних мереж, які використовуються для сервісів працюючих у режимі реального часу, насамперед VoIP. Мережеві протоколи, які обслуговують сервіси реального часу є чутливими до якості обслуговування, а саме до втрати пакетів даних, затримок у передачі пакетів та нерівномірності цих затримок [14].

Показники якості дозволяють об'єктивно оцінювати та контролювати різні аспекти послуг і продуктів. Для зручності аналізу та управління, показники якості часто класифікують на технічні, експлуатаційні та сервісні.

Технічні показники якості характеризують внутрішні властивості та характеристики продукту або послуги. Вони відображають технічні параметри, які впливають на функціональність та продуктивність.

Рис. 2.3. Показники якості

Експлуатаційні показники якості характеризують ефективність використання продукту або послуги в реальних умовах експлуатації. Вони відображають зручність, доступність та інші аспекти, що впливають на досвід користувача.

Сервісні показники якості характеризують якість обслуговування користувачів, включаючи підтримку, консультації та інші послуги, що надаються разом з продуктом або послугою.

Технічні, експлуатаційні та сервісні показники якості взаємопов'язані. Наприклад, низька пропускна здатність (технічний показник) може призвести до повільного часу відгуку (експлуатаційний показник), що, в свою чергу, може призвести до незадоволеності клієнтів (сервісний показник).

Критичними часовими параметрами є затримка (час передачі пакета) та її нестабільність — джитер (варіація затримок), які безпосередньо впливають на якість додатків реального часу. Надійність передачі оцінюється через втрати пакетів та загальну надійність системи (здатність працювати без збоїв тривалий час), тоді як ресурсна ефективність визначається пропускною здатністю (фактичною швидкістю передачі) та доступністю мережі (відсотком часу роботи без збоїв). Таким чином, управління QoS вимагає збалансованого контролю всіх цих метрик для забезпечення гарантованого рівня сервісу.

Рис. 2.4. Основні показники якості обслуговування

Пропускна здатність — це кількість даних, переданих мережею за одиницю часу:

$$T = \frac{D}{t}, \tag{10}$$

де

T — пропускна здатність (біт/с або байт/с),

D — обсяг переданих даних,

T — час передавання.

Реальна пропускна здатність завжди менша за номінальну через затримки, колізії, втрати пакетів і службовий трафік.

Ефективність каналу обраховується за формулою:

$$\eta = \frac{T_{\text{реальний}}}{T_{\text{номінальний}}} \times 100\% \quad (11)$$

Загальна затримка складається з кількох компонентів:

$$D_{\text{заг}} = D_{\text{перед}} + D_{\text{черги}} + D_{\text{обробки}} + D_{\text{поширення}}, \quad (12)$$

де

$D_{\text{перед}} = \frac{L}{R}$; — час передавання пакета довжиною L(біт) через канал зі швидкістю R(біт/с);

$D_{\text{черги}}$ — середній час очікування у черзі;

$D_{\text{обробки}}$ — час аналізу пакета маршрутизатором;

$D_{\text{поширення}} = \frac{d}{v}$ — час поширення сигналу, де d — відстань, v — швидкість сигналу ($\approx 2 \cdot 10^8$ м/с для оптоволокна).

Для інтерактивних сервісів (VoIP, відеоконференцій) допустимі затримки ≤ 150 мс.

Джитер — це варіація затримки між пакетами:

$$J = \frac{1}{N - 1} \sum_{i=1}^{N-1} \left| (D_{i+1} - D_i) \right|, \quad (13)$$

де

D_i — затримка i-го пакета,

N — кількість виміряних пакетів.

Високий джитер призводить до спотворень звуку чи відео. Для якісного VoIP — J<30 мс.

Частка втрачених пакетів від загальної кількості переданих обраховується наступною формулою:

$$P_{\text{loss}} = \frac{N_{\text{втрач}}}{N_{\text{заг}}} \times 100\% \quad (14)$$

де

$N_{\text{втрач}}$ — кількість втрачених пакетів,

$N_{заг}$ — кількість відправлених пакетів.

Причиною є перевантаження буферів, пошкодження кадрів, збої в маршрутизації.

Для реального часу допустимо $P_{loss} < 1\%$, для TCP — до $2-3\%$.

Ймовірність, що система доступна для користувача в будь-який момент часу:

$$A = \frac{T_{роб}}{T_{роб} + T_{відм}} \times 100\%, \tag{15}$$

де

$T_{роб}$ — середній час безвідмовної роботи,

$T_{відм}$ — середній час відновлення після відмови.

Класи доступності Таблиця 2.2

Рівень	Доступність	Простої на рік
99%	3.65 днів	
99.9%	8.76 год	
99.99%	52.6 хв	
99.999%	5.26 хв	

У корпоративних мережах зазвичай прагнуть до «п’яти дев’яток» (99.999%).

Надійність (Reliability) — це ймовірність безвідмовної роботи протягом часу t:

$$R(t) = e^{-\lambda t}, \tag{16}$$

де

λ — інтенсивність відмов (кількість відмов на одиницю часу).

Середній час безвідмовної роботи (MTBF):

$$MTBF = \frac{1}{\lambda}. \tag{17}$$

Середній час відновлення (MTTR):

$$MTBF = \frac{T_{заг.простоїв}}{N_{відмов}}. \tag{18}$$

Зв’язок із доступністю:

$$A = \frac{MTBF}{MTBF + MTTR}. \ (x) \qquad (19)$$

Чим менше MTTR і більше MTBF — тим вища надійність системи.

MTBF (Mean Time Between Failures) — середній час між відмовами. Це середній період, протягом якого система працює без збоїв. Чим більше MTBF, тим рідше система ламається.

MTTR (Mean Time To Repair) — середній час відновлення. Це середній час, необхідний для усунення відмови і повернення системи до робочого стану. Чим менше MTTR, тим швидше система відновлюється після збою.

А так як загальна доступність системи визначається формулою х, то:

А — коефіцієнт доступності (від 0 до 1 або у %), MTBF — середній час безвідмовної роботи, MTTR — середній час відновлення.

У чисельнику MTBF — це "час, коли система працює нормально". У знаменнику MTBF+MTTR— це повний цикл життя системи: період роботи + період ремонту. Тобто формула показує, яку частку часу система працює, а не ремонтується.

Приклад порівняння Таблиця 2.3

Показник	Сценарій 1	Сценарій 2
MTBF	1000 годин	2000 годин
MTTR	10 годин	2 години
Доступність А	1000/(1000+10)=0.990 → 99.0	2000/(2000+2)=0.999 → 99.9%

У другому випадку: система ламається рідше (MTBF↑), ремонтується швидше (MTTR↓), тому надійність (А) стає вищою.

У комп'ютерних мережах це означає:

Якщо маршрутизатор або сервер працює довше без збоїв — високий MTBF; Якщо у випадку збою адміністратор швидко перезапускає або замінює вузол — низький MTTR; У результаті користувачі майже не помічають збоїв, а доступність

мережі зростає. Надійна система — це та, яка рідко виходить з ладу (великий MTBF) і швидко відновлюється (малий MTTR).

Залежність MTBF і MTTR на надійність Таблиця 2.4

Параметр	Значення	Вплив на надійність
MTBF ↑	Збільшується час безвідмовної роботи	Надійність зростає
MTBF ↓	Частіші відмови	Надійність зменшується
MTTR ↑	Довше триває відновлення після збою	Надійність зменшується
MTTR ↓	Швидке відновлення роботи	Надійність зростає

Отже, чим більше MTBF і менше MTTR, тим система рідше виходить з ладу і швидше відновлюється після збою. Велике MTBF означає, що періоди безвідмовної роботи довгі, а відмови трапляються рідко. Малий MTTR — що навіть якщо відмова сталася, вона усувається швидко, і простої мінімальні. Тому частка часу, коли система перебуває в робочому стані (доступність), стає максимальною.

2.3 Математичні моделі мережевого трафіку

Боротьба з перевантаженнями в КМ є важливою частиною задачі забезпечення QoS і керування трафіком зокрема [16]. Використання ефективних алгоритмів боротьби з перевантаженнями дає можливість підвищити як надійність, так і корисну пропускну здатність мережі. При перевантаженні продуктивність мережі (число оброблених пакетів) прагне до нуля, а час затримки – до нескінченності. Як правило, перевантаження може викликатися флуктуаціями потоків трафіку або виходом з ладу будь-якого елемента мережі. Така ситуація може призвести як до недотримання зобов’язань мережі щодо забезпечення якості обслуговування існуючих з’єднань, так і до неможливості встановлення нового з’єднання з потрібною якістю обслуговування [17].

Рис. 2.5. Моделі мережевого трафіку

Випадкові процеси

Ця категорія охоплює Пуассонівський процес та Марківські моделі, які є класичними підходами до моделювання мережевого трафіку. Пуассонівський процес припускає, що прихід пакетів або запитів є незалежними та рідкісними випадковими подіями, і використовується для моделювання трафіку з низькою інтенсивністю або в мережах, де події відбуваються випадковим чином. Марківські моделі (наприклад, ланцюги або процеси) враховують, що ймовірність наступного стану системи залежить лише від її поточного стану (властивість відсутності післядії), що дозволяє моделювати залежність між послідовними подіями в мережі, наприклад, стан зайнятості каналу або зміну швидкості передачі даних.

Пуассонівський процес описує послідовність випадкових подій (наприклад, прибуття пакетів), які відбуваються незалежно одна від одної, мають середню інтенсивність λ (подій за одиницю часу) та імовірність двох подій у тому самому малому інтервалі часу — незначна.

Нехай N(t) — кількість подій, що відбулися за інтервал часу [0,t].

$$P\{N(t)=k\}=\frac{(\lambda t)^k e^{-\lambda t}}{k!},\ k=0,1,2,... \qquad (20)$$

де λ — інтенсивність потоку (середня кількість подій за одиницю часу).

Середнє значення обчислюється за формулою:

$$E\left[N(t)\right] = \lambda t, \tag{21}$$

Дисперсія обчислюється за формулою:

$$Var\left[N(t)\right] = \lambda t, \tag{22}$$

Міжприхідний час (інтервал між подіями), розподілений за експоненційним законом обчислюється за формулою:

$$f_T(t) = \lambda e^{-\lambda t}, t \geq 0 \tag{23}$$

$$E[T] = \frac{1}{\lambda} \tag{24}$$

Пуассонівський процес використовується для моделювання випадкового прибуття запитів або пакетів, коли події не впливають одна на одну — наприклад, у простих або слабко завантажених мережах.

Марківський процес описує систему, яка переходить між станами з певними ймовірностями, де майбутній стан залежить лише від поточного стану, а не від усієї історії.

Дискретний марківський ланцюг можна представити через формулу.

Нехай X_n — стан системи в момент часу n.

Тоді:

$$P(X_{n+1} = j \mid X_n = i, X_{n-1}, \ldots, X_0) = P(X_{n+1} = j \mid X_n = i) \tag{25}$$

Це буде властивість відсутності післядії (memoryless property).

Матриця перехідних ймовірностей представлена наступним чином:

$$P = \begin{bmatrix} p_{11}\, p_{12} \cdots p_{1m} \\ p_{21}\, p_{22} \cdots p_{2m} \\ \cdots \quad \cdots \cdots \cdots \\ p_{m1} p_{m2} \cdots p_{mm} \end{bmatrix} \tag{26}$$

де

$$p_{ij} = P(X_{n+1} = j \mid X_n = i), \tag{27}$$

а сума по рядку дорівнює 1:

$$\sum_{j=1}^{m} p_{ij} = 1. \tag{28}$$

У сталому стані ймовірність перебування системи в стані і визначається як

$$\pi j = \sum_{i=1}^{m} \pi_i p_{ij}, \quad \sum_{j=1}^{m} \pi_i = 1 \tag{29}$$

У контексті мереж це описує, наприклад, ймовірність, що канал вільний або зайнятий, або зміну швидкості передавання в різних станах навантаження.

Безперервний марківський процес (Continuous-time Markov process) використовується, коли події можуть відбуватись у довільний момент часу.

Він визначається інтенсивностями переходів q_{ij}, які утворюють матрицю інтенсивностей Q:

$$Q = \begin{bmatrix} -q_1 & q_{12} & \dots & q_{1m} \\ q_{21} & -q_2 & \dots & q_{2m} \\ \dots & \dots & \dots & \dots \\ q_{m1} & q_{m2} & \dots & -q_m \end{bmatrix}, \tag{30}$$

Де

$$q_i = \sum_{j \neq i} q_{ij}. \tag{31}$$

Стаціонарний розподіл тоді визначається з рівняння:

$$\pi Q = 0, \quad \sum_i \pi i = 1. \tag{32}$$

Моделі мережевого трафіку Таблиця 2.5

Модель	Тип процесу	Основна властивість	Використання
Пуассонівський	Потік подій	Незалежні прибуття (експоненційний розподіл)	Моделювання простих потоків запитів або пакетів
Марксівський	Станова модель	Наступний стан залежить тільки від поточного	Моделювання черг, навантаження, зміни режимів роботи кана

Пуассонівські процеси — добре описують випадкові, неузгоджені події (наприклад, пакети від різних користувачів). Марківські процеси — ефективні для динамічних систем, де поточний стан впливає на наступний (наприклад, заванта-

ження маршрутизатора, буферизація, адаптація швидкості). Ці моделі є основою для аналітичних моделей черг M/M/1, M/M/n, M/G/1 тощо.

Моделювання взаємодії та обслуговування

Ця гілка зосереджена на Теорії черг, Моделях черг та Дисциплінах обслуговування, які є критично важливими для оцінки продуктивності мережевих пристроїв та протоколів. Теорія черг математично описує, як запити (наприклад, пакети) прибувають до системи (наприклад, маршрутизатор), очікують у черзі, якщо ресурс зайнятий, і отримують обслуговування. Моделі черг (наприклад, M/M/1,M/G/1) є конкретними реалізаціями, що дозволяють обчислити такі метрики, як середня затримка, довжина черги та пропускна здатність. Дисципліни обслуговування (наприклад, FIFO, пріоритетне обслуговування) визначають порядок, у якому запити, що очікують, отримують доступ до ресурсу.

Моделі самоподібного трафіку

Самоподібність є більш сучасним підходом, який виник через відкриття, що реальний агрегований мережевий трафік часто не відповідає простим пуассонівським моделям, а демонструє залежність на великих часових масштабах (ефект довгої пам'яті). Ці моделі описують трафік, який виглядає статистично схожим незалежно від того, на якому часовому масштабі його спостерігають. Ключовими елементами є сама самоподібність та параметр Херста (H), який кількісно оцінює ступінь цієї самоподібності ($0.5<H<1$ вказує на довготривалу залежність). Фрактальні рухи є математичним апаратом, що використовується для опису трафіку з цією властивістю, наприклад, фрактальний броунівський рух.

Самоподібний (фрактальний) трафік характеризується тим, що його статистичні властивості залишаються подібними при зміні часової шкали спостереження.

Тобто, якщо агрегувати потік даних у більші часові інтервали, форма варіацій трафіку залишається схожою.

Послідовність X(t) називають самоподібною із параметром H (показник Херста), якщо для будь-якого коефіцієнта масштабування $a>0$:

$$X(at) \overset{d}{=} a^{H} X(t), \tag{33}$$

де

$\overset{d}{=}$ означає рівність за розподілом, а $H \in (0.5,1)$ — параметр Херста.

Показник Херста H визначає ступінь довготривалої залежності у часовому ряді:

H=0.5 — випадковий процес без пам'яті (приблизно пуассонівський або броунівський рух);

0.5<H<1 — довготривала кореляція, властива реальному мережевому трафіку;

H→1 — сильна самоподібність (висока інерційність змін).

Коваріаційна функція самоподібного процесу має степеневе зменшення:

$$r(k) \sim k^{2H-2}, \; k \to \infty, \tag{34}$$

де $r(k)$ — автокореляційна функція між значеннями, розділеними лагом k.

На відміну від експоненційного спаду в пуассонівських моделях, степеневий спад означає, що залежності зберігаються на великих часових масштабах (ефект «довгої пам'яті»).

Для самоподібного процесу дисперсія середніх значень зменшується повільніше, ніж у звичайних процесах:

$$\mathrm{Var}(X^{(m)}) \sim m^{2H-2}, \tag{35}$$

де $X^{(m)}$ — процес, агрегований за інтервалами довжиною m.

Якщо H>0.5, дисперсія спадає повільніше → трафік залишається «бурхливим» навіть при усередненні.

Фрактальний броунівський рух — базова модель самоподібного трафіку. Він є гаусівським процесом $B_H(t)$ з такими властивостями:

$$E\left[B_H(t)\right] = 0, \tag{36}$$

$$\mathrm{Var}(B_H(t)) = \sigma^2 t^{2H}, \tag{37}$$

$$\mathrm{Cov}(B_H(t), B_H(s)) = \frac{\sigma^2}{2} \left(t^{2H} + s^{2H} - |t-s|^{2H}\right). \tag{38}$$

тут

σ^2 — дисперсія одиничного приросту, H — показник Херста, що визначає рівень самоподібності. При H=0.5 отримуємо класичний броунівський рух (без пам'яті), при H>0.5 — з'являється довготривала залежність (трафік «корельований» у часі).

Реальний трафік часто моделюють як приріст фрактального броунівського руху:

$$X(t) = (B_H(t+1) - B_H(t), \tag{39}$$

що утворює стаціонарний процес із автокореляційною функцією:

$$r(k) = \frac{1}{2}(|k+1|^{2H} - 2|k|^{2H} + |k-1|^{2H}). \tag{40}$$

Самоподібні моделі (FBM, FGN) краще описують реальні потоки даних у сучасних мережах — HTTP, відео, P2P, cloud-трафік. Вони дозволяють точніше прогнозувати черги, перевантаження, затримки, ніж класичні пуассонівські моделі. Основна властивість — довга пам'ять, яка впливає на продуктивність мереж навіть при агрегуванні трафіку.

Моделі прогнозування та оцінки продуктивності

Ця категорія включає практичні методи, які використовуються для прогнозування майбутніх потреб у мережевих ресурсах і оцінки ефективності їхнього використання. Авторегресійні моделі (наприклад, ARIMA, ARMA) використовують минулі значення трафіку для прогнозування майбутніх, припускаючи, що між ними існує лінійна залежність. Імітаційне моделювання передбачає побудову програмної моделі мережі (симулятора) для експериментального відтворення поведінки мережі в різних умовах, що є особливо корисним для складних сценаріїв, які важко проаналізувати аналітично.

Діагностичні метрики — це набір показників (наприклад, пропускна здатність, затримка, втрата пакетів), які використовуються для вимірювання, моніторингу та оцінки поточної продуктивності мережі.

2.4 Методи та алгоритми виявлення аномалій у мережевому трафіку

Аномалія мережі –це раптове та короткочасне відхилення від нормальної роботи мережі. Деякі аномалії навмисно викликані зловмисниками зі шкідливими намірами, як-от атака на відмову в обслуговуванні (DoS/DDoS) в IP-мережі. Швидке виявлення аномалій необхідне для своєчасного реагування на зміни стану в інформаційній системі.

Поняття «вторгнення» та «аномалії» зазвичай використовуються як синоніми в контексті IDS; однак обидва терміни мають певні відмінності. Вторгнення —це зловмисна діяльність, яка намагається скомпрометувати конфіденційність, цілісність і доступність, тоді як аномалія стосується моделей даних, які не відповідають очікуваній нормальній поведінці, тобто відхиленню від того, що вважається нормальним. Однак поняття аномалії залежить від сфери застосування та контексту, тобто аномалія не завжди є вторгненням. Наприклад, у мережевому домені відстежуються мережевий трафік, індекси продуктивності та журнали для виявлення збоїв у мережі (це не є вторгненням). Дуже часто поняття системи виявлення вторгнень та системи виявлення аномалій утотожнюються, хоч і мають відмінності. Підходи до виявлення аномалій базуються на методі використаного навчання, на статистичних методах та методі машинного навчання [18]. Мережеві аномалії можна класифікувати за різними критеріями. У цьому документі ми розглянемо три основні категорії:

Рис. 2.6. Аномалії в мережевому трафіку

Аномалії, пов’язані з поведінкою. Такі аномалії виникають, коли дії користувачів або пристроїв у мережі відхиляються від звичного сценарію. Прикладом є сканування портів, коли зловмисник намагається виявити відкриті порти на різних вузлах мережі для подальшого вторгнення. Інший тип поведінкових аномалій — горизонтальне переміщення (lateral movement), коли зловмисник, отримавши доступ до однієї системи, намагається поширитися на інші, використовуючи внутрішні облікові дані [40]. Такі відхилення часто виявляються за допомогою поведінкової аналітики трафіку (UBA/NTA).

Аномалії, пов’язані з обсягом. Цей тип аномалій характеризується раптовими змінами інтенсивності трафіку — його різким збільшенням або падінням. Найпоширенішими прикладами є сплески трафіку під час DDoS-атак, коли мережа або сервер перевантажуються великою кількістю запитів, що призводить до недоступності сервісу. Інша форма — раптове зниження трафіку (відмова служби), яке може свідчити про збої в роботі мережевого обладнання або порушення маршрутизації. Моніторинг обсягу допомагає виявляти такі ситуації в реальному часі.

Аномалії, пов’язані з протоколами. До цієї категорії належать порушення стандартів обміну даними, які можуть бути як результатом помилок, так і навмисних атак. Наприклад, неправильне використання заголовків пакетів (спотворені поля IP або TCP) може бути спробою приховати шкідливу активність чи обійти фільтрацію.

Інший приклад — нестандартні порти, коли служби запускаються на незвичних номерах портів для уникнення виявлення системами безпеки. Аналіз протоколів і сигнатурний контроль допомагають виявляти такі аномалії та запобігати потенційним атакам. Методи виявлення аномалій у мережевому трафіку представлені на Рис. 2.7.

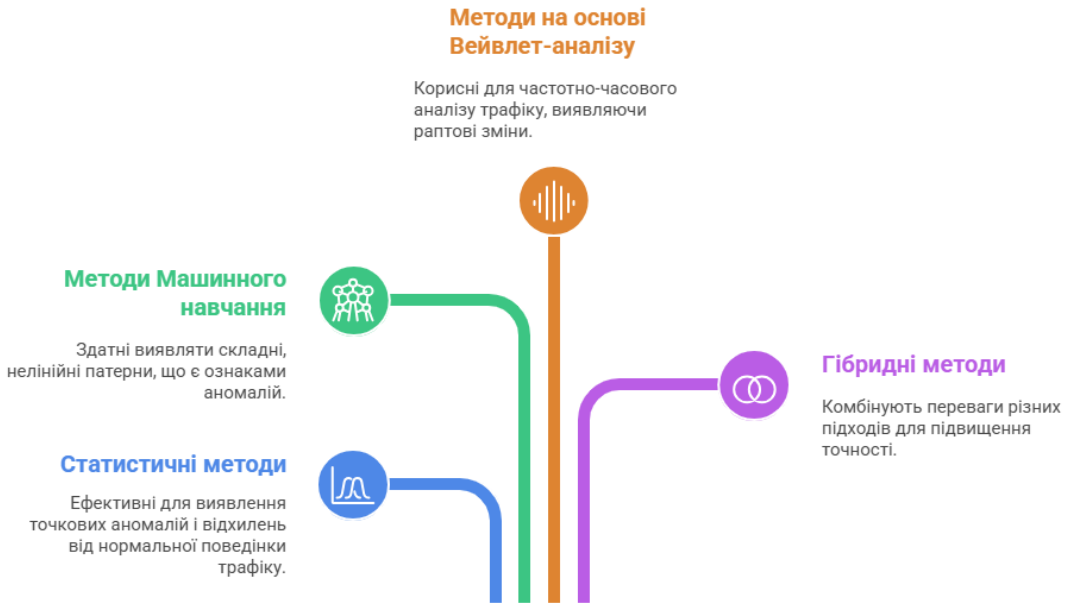


Рис. 2.7. Методи виявлення аномалій у мережевому трафіку

Статистичні методи виявлення аномалій ґрунтуються на побудові статистичної моделі нормальної поведінки мережевого трафіку та ідентифікації значних відхилень від цієї норми, що зазвичай вказує на точкові аномалії. Для цього використовують такі інструменти, як Z-оцінка, яка вимірює, на скільки стандартних відхилень показник трафіку відхилився від середнього, та метод експоненційного ковзного середнього (EMA), що динамічно зважує останні дані для підвищення чутливості до поступових змін. Складніші підходи включають аналіз часових рядів для прогнозування майбутнього трафіку та виявлення розбіжностей, а також ентропійний аналіз, який оцінює різноманітність характеристик трафіку (наприклад, IP-адрес); різке падіння ентропії може сигналізувати про однотипну атаку, як-от сканування портів. Статистичні моделі дозволяють виявляти аномалії мережевоготрафіку та робити прогнози щодо його кількісних показників [19].

Z-оцінка — це показник, який відображає, на скільки стандартних відхилень певне спостереження x_i відрізняється від середнього значення μ. Якщо значення виходить за межі визначеного порогу, воно вважається аномальним.

Основна формула

$$Z_i = \frac{xi - \mu}{\sigma} \tag{41}$$

де:

x_i — поточне виміряне значення трафіку (наприклад, кількість пакетів за секунду, обсяг байтів, кількість з'єднань тощо);

μ — середнє значення трафіку за історичний період (норма);

σ — стандартне відхилення, що показує розкид значень трафіку навколо середнього.

Якщо $|Z_i| \leq 2| \rightarrow$ нормальна поведінка (відхилення незначні). Якщо $2 < |Z_i| \leq 3 \rightarrow$ підозріла активність, варто спостерігати. Якщо $|Z_i| > 3 \rightarrow$ аномалія, можливий інцидент у мережі.

Вейвлет-аналіз — це математичний інструмент, який дозволяє розкласти сигнал (трафік) на складові різних частот і часових масштабів. На відміну від звичайного перетворення Фур'є, яке показує лише частоти, вейвлети зберігають локалізацію в часі, тобто показують, коли саме виникає аномалія.

У контексті мережевого трафіку сигналом є часовий ряд:

$$x(t) = \text{кількість пакетів або байтів у момент часу } t.$$

Математично вейвлет-перетворення визначається як:

$$W(a,b) = \frac{1}{\sqrt{|a|}} \int_{-\infty}^{\infty} x(t) \psi^*\left(\frac{t-b}{a}\right) dt, \qquad (42)$$

де:

x(t) — аналізований сигнал (трафік);

ψ(t) — базова вейвлет-функція (наприклад, Мексиканський капелюх, Морле, Добеші);

a — масштаб (scale), який визначає рівень деталізації (малі a — високочастотні деталі, великі a — низькочастотні тенденції);

b — зсув (shift) у часі;

ψ^* — комплексно-спряжена функція вейвлету.

Результат W(a,b) показує, наскільки сильно сигнал x(t) містить компоненти певної частоти (масштабу) у момент часу b. Якщо |W(a,b)| різко зростає, це означає, що у цей момент часу відбулася аномалія — різкий стрибок або зміна струк-

тури трафіку. Малий масштаб (a) вказує на короткочасні аномалії (наприклад DDoS-сплеск). Великий масштаб вказує на довготривалі зміни (наприклад поступове зростання навантаження або деградація сервісу). Для практичного аналізу трафіку часто застосовується дискретне вейвлет-перетворення, яке розкладає сигнал на дві компоненти:

$$x(t) = Aj(t) + D_j(t), \tag{43}$$

де:

$A_j(t)$— апроксимаційна складова (повільні, фонові зміни трафіку);

$D_j(t)$ — детальна складова (високочастотні компоненти — потенційні аномалії).

Якщо енергія або амплітуда $D_j(t)$перевищує певний поріг T, фіксується аномалія:

$$| D_j(t)| > T \Rightarrow \text{аномальна подія}.$$

Поріг T часто визначають за статистичним критерієм:

$$T = \sigma\sqrt{2\ln N}, \tag{44}$$

де σ — стандартне відхилення коефіцієнтів $D_j(t)$, N — довжина вибірки сигналу.

Наприклад, при аналізі вхідного трафіку спостерігається різке збільшення високочастотної енергії $|D_j(t)|$ на малих масштабах. Це може вказувати на раптові сплески коротких з'єднань — типовий симптом DDoS-атаки або сканування портів**.** На більших масштабах відхилення можуть свідчити про аномальні тренди в навантаженні**.**

Отже, вейвлет-аналіз дозволяє виявляти аномалії на різних часових рівнях **і** є ефективним як для раптових**,** так і поступових змін у трафіку. За потребою він може комбінуватися з іншими методами (Z-score, статистичні тести, нейронні мережі).

А сама аномалія у вейвлет-просторі визначається як:

$$A = \{(a,b)| \ | W(a,b)| > T\}, \tag{45}$$

де A — множина часових і масштабних точок, у яких виявлено відхилення.

Методи машинного навчання (ML) є потужним інструментом для виявлення складних, нелінійних патернів аномалій у мережевому трафіку, використовуючи великі обсяги даних. Вони поділяються на три основні категорії: навчання з учителем, яке вимагає позначених даних для класифікації трафіку за допомогою таких алгоритмів, як випадковий ліс чи нейронні мережі (CNN, RNN); навчання без учителя, що ідеально підходить для виявлення zero-day атак, оскільки ідентифікує аномалії як точки, що не належать до жодного кластера або мають низьку локальну густину, використовуючи такі алгоритми, як K-середніх або Isolation Forest; і напівнавчання, де модель, наприклад автокодувальник, навчається лише на нормальних даних, а висока помилка відновлення нового зразка вказує на його аномальність.

Модель дерева рішень класифікує або оцінює об’єкти (наприклад, пакети, сесії, вузли мережі) на основі послідовності логічних правил виду “якщо – то” (if–then). Кожне правило ґрунтується на розбитті простору ознак (features) за певним критерієм, який мінімізує невизначеність або ентропію даних.

Нехай маємо множину спостережень (трафік)

$$D = \{(x_1, y_1), (x_2, y_2), ..., (x_N, y_N)\}, \tag{46}$$

де $x_i = (x_{i1}, x_{i2}, ..., x_{im})$— вектор ознак трафіку (наприклад швидкість потоку, кількість пакетів, час сесії, порт, протокол тощо); $y_i \in$ {норма,аномалія} є міткою класу.

Для кожного вузла дерево обирає ознаку x_j і порогове значення t_j, які мінімізують міру невпорядкованості — наприклад, ентропію або індекс Джині.

Ентропія розраховується за формулою:

$$H(D) = -\sum_{c \in C} p(c) \log_2 p(c), \tag{47}$$

де p(c) — частка елементів класу c у вибірці D.

Індекс Джині розраховується за формулою:

$$G(D) = 1 - \sum_{c \in C} p(c)^2. \tag{48}$$

Приріст інформації розраховується за формулою:

$$\Delta H = H(D) - \left(\frac{|D_L|}{|D|} H(D_L) + \frac{|D_R|}{|D|} H(D_R) \right), \quad (49)$$

де D_L,D_R — підвибірки після розбиття.

Вибирається розбиття з максимальним приростом інформації max ΔH. Кожен об'єкт x_i "спускається" по дереву до листового вузла, де зберігається кінцева оцінка:

$$y_i = f(x_i), \quad (50)$$

де f(·) є побудованою функцією прийняття рішення. Якщо у листі переважають аномальні зразки, система класифікує потік як аномальний. Якщо частота пакетів >t1, то можлива DDoS-активність. Якщо кількість унікальних IP за короткий час >t2, то це є скануванням портів. Якщо використовується нетиповий порт або протокол, то це свідчить про протокольну аномалію.

Гібридні методи поєднують переваги двох або більше підходів (наприклад, статистичних і машинного навчання) для підвищення загальної точності виявлення аномалій. Це дозволяє ефективніше фільтрувати очевидні аномалії, а потім аналізувати складні приховані патерни, мінімізуючи хибнопозитивні спрацьовування.

2.5. Методика покрокового аналізу захопленого трафіку на різних рівнях моделі OSI

1. Аналіз фізичного та канального рівнів (рівні 1-2 OSI). Цей етап зосереджується на перевірці основної фізичної та логічної зв'язності мережі. Аналізується цілісність Ethernet-кадрів, їхні заголовки та коректність обміну MAC-адресами.

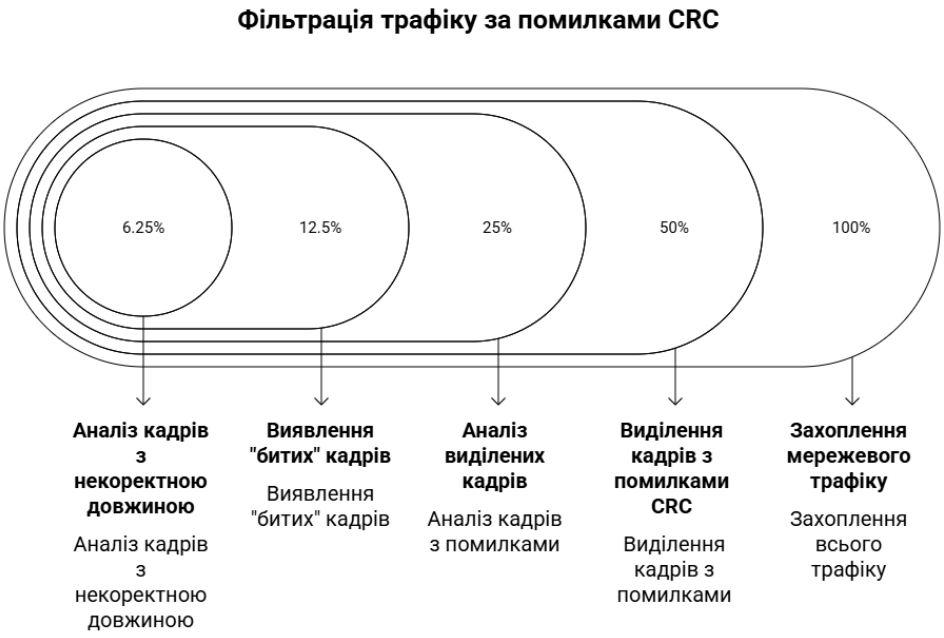


Рис. 2.8. Фільтрація трафіку за помилками CRC

Крок 1. Налаштування інструменту аналізу трафіку (наприклад Wireshark, tcpdump) для захоплення мережевого трафіку.

Крок 2. Застосування фільтрів для виділення кадрів з помилками CRC. У Wireshark це можна зробити за допомогою фільтру `eth.fcs_bad == 1` [42].

Крок 3. Аналіз виділених кадрів. Перевірка джерела та призначення кадрів, частоти виникнення помилок.

Крок 4.Виявлення "битих" кадрів (кадрів з некоректною структурою або невідповідністю довжини).

Крок 5. Аналіз кадрів з некоректною довжиною. Перевірка відповідності поля довжини кадру фактичній довжині даних.

Аналіз протоколу ARP для виявлення ARP-спуфінгу або надмірної кількості ARP-запитів, які можуть вказувати на перевантаження канального рівня полягає у наступних діях:

Крок 1. Фільтрація трафіку для виділення ARP-пакетів. У Wireshark це можна зробити за допомогою фільтру `arp`.

Крок 2. Аналіз ARP-запитів та відповідей. Перевірка відповідності MAC-адрес IP-адресам.

Крок 3. Виявлення ARP-спуфінгу. Це можна зробити шляхом пошуку декількох ARP-відповідей з різними MAC-адресами для однієї IP-адреси.

Крок 4. Аналіз кількості ARP-запитів. Надмірна кількість ARP-запитів може вказувати на перевантаження канального рівня або спроби ARP-спуфінгу.

Крок 5. Перевірка наявності "gratuitous ARP" пакетів (ARP-пакети, які відправляються без запиту). Їх надмірна кількість може вказувати на проблеми з конфігурацією мережі.

Для більш глибокого аналізу може знадобитися використання спеціалізованих інструментів для аналізу фізичного рівня, таких як кабельні тестери. Важливо враховувати контекст мережі при аналізі трафіку. Наприклад велика кількість помилок CRC може бути нормальною для мережі з великою кількістю бездротових з'єднань.

Приклад використання Wireshark:

1. Для захоплення трафіку необхідно запустити Wireshark та вибрати мережевий інтерфейс для захоплення трафіку.
2. Для фільтрації помилок CRC необхідно ввести фільтр eth.fcs_bad == 1 у поле фільтру.
3. Проаналізувати ARP-трафік ввівши фільтр arp у поле фільтру.
4. Перевірити на ARP-спуфінг переглянувши ARP-відповіді та перевірити, чи немає декількох відповідей з різними MAC-адресами для однієї IP-адреси.
5. Проаналізувати кількість ARP-запитів переглянувши статистику ARP-трафіку та перевірити, чи немає надмірної кількості ARP-запитів.

Аналіз мережевого рівня (рівень 3 OSI) полягає в тому, що на цьому етапі основна увага приділяється протоколу IP (Internet Protocol) та його допоміжним протоколам, що відповідають за маршрутизацію та логічну адресацію. Тут виявляються проблеми маршрутизації, високих затримок (затримки передачі), наявність DDoS-атак на мережевому рівні (наприклад ICMP-флуд) та виявляються неправильні налаштування мережевих пристроїв.

Рис. 2.9. Аналіз мережевого рівня

Покроковий алгоритм аналізу мережевого рівня (Рівень 3 OSI)

Крок 1. Збір даних. Спершу необхідно захопити мережевий трафік. Для цього необхідно використати інструменти захоплення мережевого трафіку, такі як Wireshark, tcpdump або Microsoft Network Monitor, для збору IP-пакетів. Захоплювати трафік потрібно на стратегічних точках мережі, таких як маршрутизатори, шлюзи та сервери. Залежно від мети аналізу, можна фільтрувати трафік за певними критеріями, наприклад, за IP-адресою, портом, протоколом (ICMP, TCP, UDP) або іншими параметрами. Це допоможе зосередитися на конкретних проблемах.

Крок 2. Аналіз заголовків IP-Пакетів. Тут необхідно переконатися, що IP-адреси джерела та призначення є валідними та відповідають очікуваним діапазонам адрес. Необхідно перевірити значення TTL у IP-пакетах. Низьке значення TTL (наприклад 1 або 2) може свідчити про петлеву маршрутизацію. Зменшення TTL з кожним переходом дозволяє оцінити кількість маршрутизаторів на шляху.

Неочікувано велика кількість переходів може вказувати на проблеми з маршрутизацією. Щоб проаналізувати пакети, потрібно перевірити поля "Flags" та "Fragment Offset" у заголовку IP. Велика кількість фрагментованих пакетів може свідчити про проблеми з MTU на шляху між джерелом та призначенням. Фрагментація може призвести до збільшення затримок та втрати пакетів.

Крок 3. Аналіз протоколу ICMP (Ping). Щоб перевірити доступність мережевих пристроїв та вимірювання затримок, необхідно використати команду ping для відправки ICMP Echo Request пакетів до цільового пристрою. Аналізуйте

відповіді ICMP Echo Reply. Високі значення RTT можуть свідчити про перевантаження мережі, проблеми з маршрутизацією або фізичні проблеми з мережевим обладнанням. Втрата пакетів може вказувати на перевантаження мережі, проблеми з маршрутизацією, помилки в конфігурації мережевих пристроїв або фізичні проблеми з мережевим обладнанням.

Щоб виявити, чи є DDoS-атаки на мережевому рівні необхідно проаналізувати кількість ICMP-пакетів, що надходять з однієї IP-адреси. Незвичайно велика кількість ICMP-пакетів з однієї IP-адреси може свідчити про ICMP-флуд.

Крок 4. Аналіз розподілу IP-адрес джерел/призначень. Щоб виявити незвичайної активності, які можуть свідчити про DDoS-атаки, сканування мережі або інші шкідливі дії необхідно проаналізувати розподіл IP-адрес джерел та призначень і звернути увагу на незвичайно велику кількість пакетів від однієї IP-адреси або до однієї IP-адреси. Використавши інструменти геолокації IP-адрес для визначення географічного розташування IP-адрес джерел та призначень можна виявити неочікуване географічне розташування, яке може свідчити про шкідливу активність.

Крок 5. Кореляція даних та діагностика. На цьому кроці необхідно зіставити результати аналізу заголовків IP-пакетів, аналізу ICMP та аналізу розподілу IP-адрес для отримання більш повної картини мережевої активності**.** На основі корельованих даних можна визначити причини проблем з маршрутизацією, затримками, втратою пакетів та іншими мережевими проблемами.

На Рис.2.10 представлений алгоритм аналізу транспортного та прикладного рівнів (4-7 OSI) мережевого трафіку, який включає кілька ключових діагностичних напрямків:

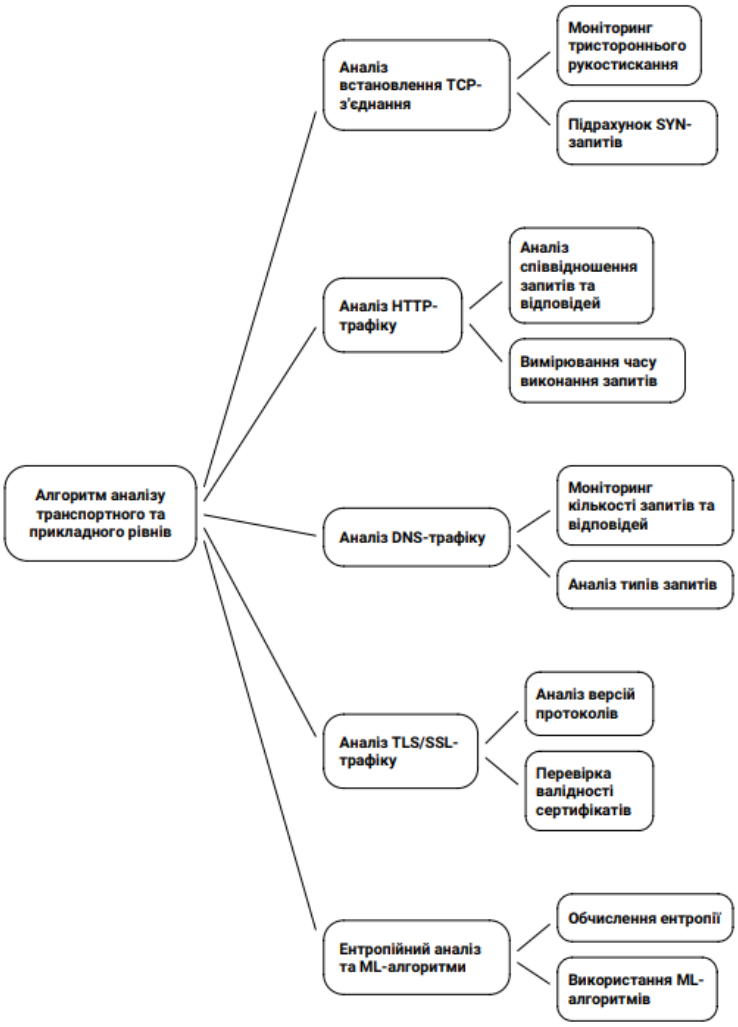


Рис. 2.10. Алгоритм аналізу транспортного та прикладного рівнів

Аналіз встановлення TCP-з'єднання відповідає за моніторинг тристороннього рукостискання та підрахунку SYN-запитів для виявлення атак.

Аналіз HTTP-трафіку вимірює співвідношення запитів і відповідей та час виконання запитів для оцінки продуктивності веб-додатків.

Аналіз DNS-трафіку включає моніторинг кількості та типів запитів і відповідей для виявлення аномалій, пов'язаних з роздільною здатністю імен.

Аналіз TLS/SSL-трафіку перевіряє версії протоколів та валідність сертифікатів для оцінки безпеки з'єднань.

Ентропійний аналіз та ML-алгоритми використовують обчислення ентропії та алгоритми машинного навчання як універсальні методи для виявлення складних або невідомих аномалій у трафіку.

2.6. Алгоритм ідентифікації мережевих загроз та атак на основі аналізу пакетів

Для забезпечення кібербезпеки важливо мати системний та багаторівневий підхід до аналізу мережевої активності. Представлений нижче алгоритм ідентифікації мережевих загроз та атак (АІМЗА), який використовує комбінацію глибокого аналізу пакетів, статистичного моделювання для встановлення "норми" та технік машинного навчання для виявлення та класифікації аномалій і загроз у режимі реального часу і забезпечує швидке та автоматизоване реагування на інциденти.

Збір сирого трафіку описує, як мережевий трафік захоплюється та зберігається. Початок збору трафіку є стартовою точкою процесу. При виборі техніки збору визначається, чи будуть збиратися сирі пакети (для глибокої інспекції) чи метадані (для масштабованого аналізу).

Сирі пакети (глибока інспекція) повязана із сенсорами та SPAN/Mirror TAP/NIDS, тобто використовуються порти-дзеркала на комутаторах (SPAN/Mirror) або мережевих TAP-пристроях для копіювання трафіку на сенсор. Інструменти tcpdump/pcap/Wireshark використовуються для захоплення та збереження сирих даних у форматі pcap.

Зібрані сирі дані направляються у: кільцевий буфер для короткочасного зберігання останніх даних, в Elastic Stack для індексування та пошуку, в Hadoop HDFS для довготривалого та масштабованого зберігання [43].

NetFlow/IPFIX/sFlow - це протоколи, які агрегують статистику про мережеві з'єднання (потоки) замість зберігання кожного пакета. Трафік-метадані з NetFlow/IPFIX/sFlow зазвичай направляються безпосередньо до аналізу або зберігання, оминаючи детальний парсинг сирих пакетів.

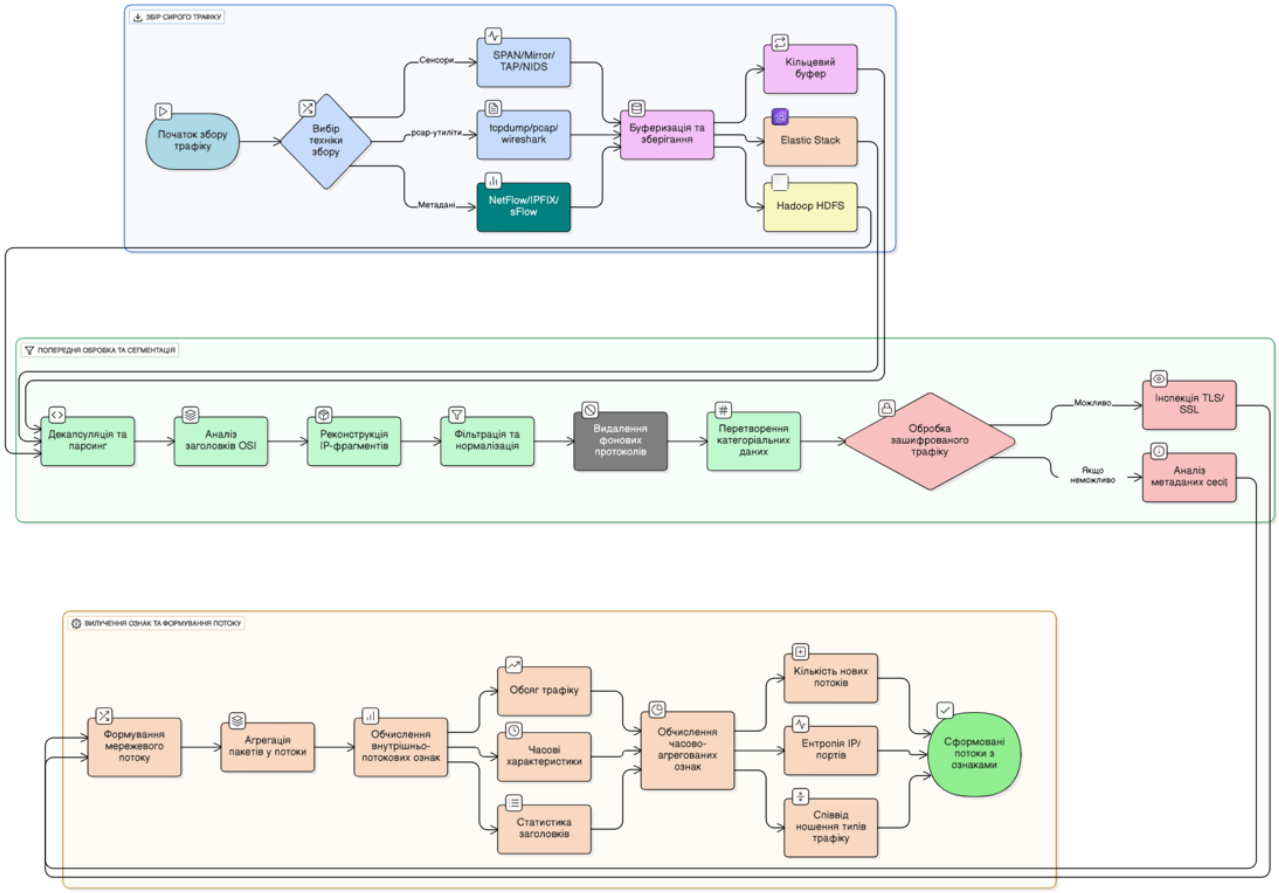


Рис. 2.11. Алгоритм ідентифікації мережевих загроз та атак

Попередня обробка та оптимізація перетворює сирі пакети на структуровані дані. Тут відбуваються:

1. Декапсуляція та парсинг, де вилучаються пакети з мережевих кадрів.
2. Аналіз заголовків OSI, при якому розбираються заголовки усіх рівнів (L2 - L7), щоб ідентифікувати протоколи, адреси та порти.
3. Реконструкція IP-фрагментів, на якому йде збір фрагментованих IP-пакетів до їхнього початкового вигляду.
4. Фільтрація та нормалізація. Тут спочатку відбувається видалення фонових протоколів, тобто видаляється несуттєвий "шум" (наприклад, ARP, протоколи підтримки), які не є релевантними для виявлення загроз.

   Також тут відбувається перетворення категоріальних даних, а саме конвертація текстових/перелічувальних значень (назви протоколів, прапори TCP) у числові формати (наприклад one-hot encoding), придатні для ML-моделей.

5. Обробка зашифрованого трафіку, де визначається підхід до HTTPS/TLS-трафіку.

Якщо можливо - інспекція TLS/SSL і йде використання проксі або методів MitM для розшифрування та аналізу вмісту (потребує відповідних ключів або сертифікатів).

Решта трафіку - аналіз метаданих сесії. Якщо вміст недоступний, аналізуються метадані (розміри пакетів, тривалість сесії, Server Name Indication), які можуть вказувати на аномалії, навіть у зашифрованому потоці.

Вилучення ознак та формування потоку - це найважливіший блок, де сирі дані перетворюються на числові ознаки для моделювання. В нього входять такі етапи:

1. Формування потоку, де групуються споріднені пакети.
2. Агрегація пакетів у потоки. На цьому етапі пакети, що належать до одного логічного з'єднання (визначається 5-кортежем: Source IP, Dest IP, Source Port, Dest Port, Protocol), об'єднуються у двосторонній потік.
3. Обчислення потокових ознак, на якому обчислюються внутрішньо-потокові ознаки і розраховуються метрики для окремого потоку (тривалість потоку, загальний обсяг даних, співвідношення SYN/ACK).

   Також тут обробляється трафік, а сама обчислюється середній час між пакетами (Jitter) та часовими мітками. Також формується статистика заголовків, аналізуються прапори TCP, розміри вікна, значення TTL.
4. Обчислення часово-агрегованих ознак. На цьому етапі йде розрахунок метрик для групи потоків у певному часовому вікні (наприклад за 5 секунд). Основними розрахунковими показниками є:
   - Обсяг трафіку - сумарний обсяг байтів/пакетів за часовий інтервал.
   - Кількість нових потоків - частота встановлення нових з'єднань.
   - Ентропія IP/портів - вимірювання різноманітності адрес/портів.

Низька ентропія часто є ознакою DDoS або цілеспрямованої атаки.

- Співвідношення типів трафіку - порівняння Unicast, Broadcast, Multicast.

5. Формування IP-потоків з ознаками. Кінцевим результатом є структурований набір даних, де кожен рядок є мережевим потоком або часовим інтервалом, а стовпчики — це обчислені числові ознаки. Ці дані готові для подальшого подання моделям машинного навчання.

Результатом першого етапу алгоритму є структурований набір числових ознак, готовий для подальшого статистичного моделювання та класифікації загроз. Однак, надійність та точність цього фінального набору даних цілком залежать від цілісності та повноти початкового збору. В умовах сучасних високошвидкісних мереж (10G, 40G і вище) сам процес захоплення, парсингу та попередньої обробки трафіку створює значні інженерні виклики. Помилки або втрати пакетів на цьому початковому етапі незворотно компрометують подальшу ефективність виявлення загроз, оскільки критична інформація про початок або кульмінацію атаки може бути втрачена. Тому для забезпечення надійності системи, необхідно детально розглянути технічні проблеми, які виникають при роботі з великими обсягами даних у реальному часі.

Висока швидкість сучасних мереж (10 Gigabit Ethernet (10G), 40G, і 100G) створює технічні перешкоди для гарантованого та повного захоплення трафіку, що є критичним для забезпечення надійності алгоритму ідентифікації загроз. Втрата пакетів – це найсерйозніша загроза цілісності даних на етапі збору.

При надходженні трафіку на мережевий інтерфейс з високою інтенсивністю (особливо при малих розмірах пакетів), операційна система та програмні засоби захоплення (наприклад стандартні реалізації libpcap) не встигають переміщувати дані з буферів мережевої карти в пам'ять ядра, або з пам'яті ядра в буфер програми обробки. Це призводить до переповнення буферів та, як наслідок, відкидання (дропання) частини пакетів.

Втрата пакетів під час атаки може призвести до того, що критичні пакети, що містять початок шкідливого навантаження або перший SYN-пакет DDoS-

атаки, будуть пропущені або мережевий потік не буде коректно реконструйовано, що спотворить обчислені ознаки (наприклад, обсяг, співвідношення SYN/ACK) і призведе до хибно-негативного спрацювання (пропущеної атаки).

Традиційні методи обробки мережевого трафіку вимагають великої кількості перемикань контексту між простором ядра та простором користувача, що є неефективним і ресурсозатратним.

Кожен пакет, що надходить, викликає переривання (interrupt) для ЦП, змушуючи його переходити від виконання програми користувача до виконання обробки в ядрі. На швидкостях 10G+ це призводить до "пекельного переривання" (interrupt hell), де більшість ресурсів ЦП витрачається на обробку переривань, а не на корисний аналіз даних.

Також трафік часто копіюється декілька разів (з буфера NIC в буфер ядра, з буфера ядра в буфер програми), що втрачає цикли ЦП та збільшує латентність.

Для подолання цих проблем застосовуються спеціалізовані технології, які мінімізують взаємодію з ядром операційної системи.

DPDK (Data Plane Development Kit) - комплект розробки, що дозволяє програмам обходити ядро ОС і безпосередньо працювати з апаратним забезпеченням мережевої карти. DPDK використовує техніку опитування замість переривань, що забезпечує детерміновану та високоефективну передачу пакетів, знижуючи латентність і втрати.

PF_RING - ще один фреймворк, який пропонує високу швидкість захоплення та фільтрації. Він включає технологію Zero Copy, яка дозволяє уникнути непотрібного копіювання пакетів, передаючи їх безпосередньо до програми обробки.

3. ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ ТА АНАЛІЗ МЕРЕЖЕВИХ АНОМАЛІЙ І ЗАТРИМОК

3.1. Архітектура та топологія експериментального сегмента

Для забезпечення достовірності та відтворюваності досліджень у сфері комплексної мережевої діагностики, важливим є створення експериментального середовища. Представлена схема нижче показує архітектуру тестового полігону, спроектованого для моделювання реального корпоративного сегмента з типовим розподілом функціональних зон (WAN, DMZ, LAN) та ключовими елементами безпеки (мережевий екран, маршрутизатор). Зокрема, процес моделювання загроз спрямований на виявлення й краще розуміння можливих загроз, з якими стикається IT-екосистема організації [20]. Цей полігон можна використати для імітації як нормального мережевого трафіку, так і різноманітних векторів мережевих загроз. Завдяки визначеному розміщенню точок моніторингу (SPAN-порт), забезпечується повний збір трафіку для подальшого аналізу пакетів та оцінки ефективності діагностичного алгоритму.

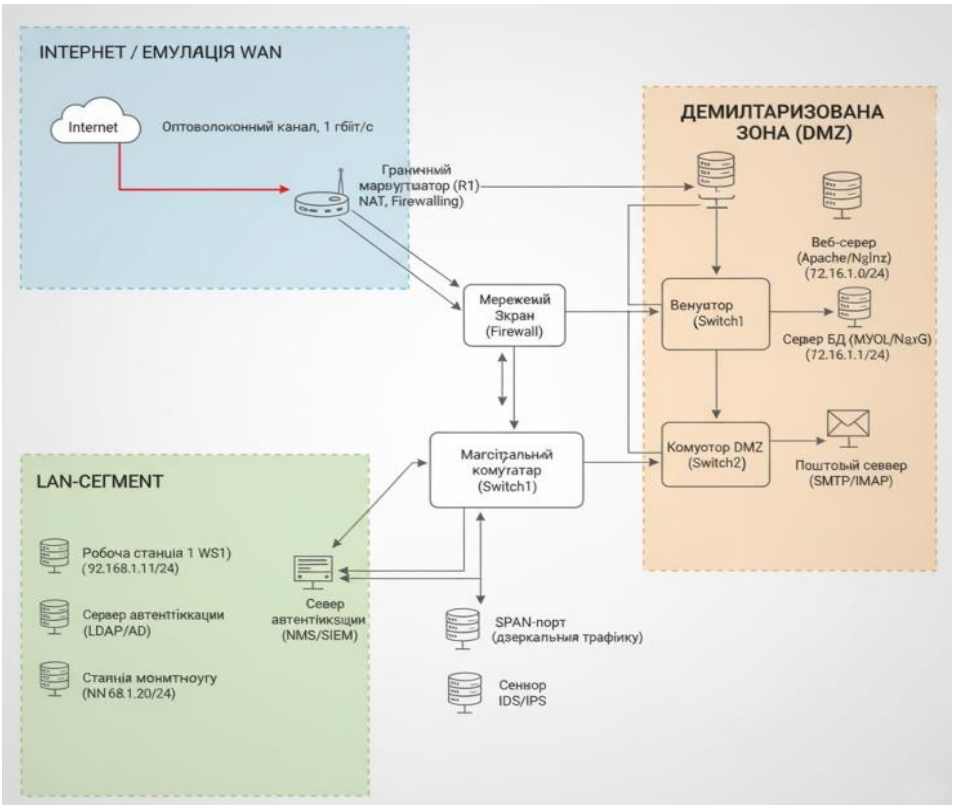


Рис. 3.1. Схема топології тестового полігону

Схема складається з трьох основних логічних доменів, розмежованих мережевими пристроями безпеки.

1. Internet / емуляція WAN – зона,яка імітує зовнішню мережу (Інтернет або глобальну мережу) та вхідний канал зв'язку. Оптоволоконний канал, 1 Гбіт/с представляє вхідний канал із заданою пропускною здатністю, що встановлює ліміт швидкості для всього зовнішнього трафіку. Граничний маршрутизатор (R1) з функціями NAT, Firewalling - це перша лінія захисту та входу. Він виконує трансляцію мережевих адрес (NAT) для внутрішніх IP-адрес і застосовує базові правила фільтрації. Трафік із WAN спрямовується звідси до мережевого екрана.

2. Демілітаризована зона (DMZ) – це частина мережі, доступ до якої обмежено МЕ як із внутрішньої мережі організації, так і зовні. DMZ визначає частини мережі, яким можна довіряти, а яким – ні [21]. Мережевий екран - центральний елемент безпеки, що розділяє WAN, DMZ та LAN. Він дозволяє обмежений зовнішній доступ до сервісів DMZ і контролює трафік між DMZ і LAN. Комутатор DMZ (Switch2) розподіляє трафік у межах зони DMZ. Веб-сервер (Apache/Nginx) – це можлива ціль для зовнішніх атак (DoS, Web Application Attacks). Сервер Б/Д (MYSQL/PostG) містить дані, пов'язані з веб-сервером (IP: 72.16.1.1/24). Зазвичай доступний лише з Веб-сервера. Поштовий сервер (SMTP/IMAP) обробляє електронну пошту.

3. LAN-сегмент - це внутрішня корпоративна мережа, що містить робочі місця та критичні сервіси автентифікації. Сегмент — це логічний інтерфейс, який може включати один або кілька доступних фізичних інтерфейсів [22]. Магістральний комутатор (Switch1) з'єднує LAN із DMZ та мережевим екраном. Робоча станція 1 (WS1) – це типовий клієнтський вузол (IP: 192.168.1.11/24), потенційне джерело внутрішніх загроз або ціль для фішингу. Сервер автентифікації (LDAP/AD) – це внутрішній ресурс, який використовується для керування користувачами та доступом. Станція моніторингу (NN68.1.20/24) призначена для розміщення діагностичного обладнання та є кінцевою точкою для збору трафіку.

4. Точки моніторингу та аналізу (Критичні вузли). Ефективність діагностики залежить від правильного розміщення сенсорів. В сервері автентифікації (NMS/SIEM) розміщується система управління мережею (NMS) або SIEM-система, яка збирає та корелює журнали та події. SPAN-порт (дзеркальний трафік) – основна точка захоплення. На магістральному комутаторі (Switch1) налаштовано дзеркалювання всього трафіку, що проходить через комутатор, на станцію моніторингу. Це дозволяє пасивно аналізувати трафік між LAN, DMZ та зовнішнім світом. Сенсор IDS/IPS служить системою виявлення/запобігання вторгнень, розміщений для прийому дзеркального трафіку зі SPAN-порту. Цей сенсор виконує захоплення сирих пакетів та передає їх до алгоритму для подальшої попередньої обробки та вилучення ознак.

Ця топологія дозволяє моделювати реальні сценарії зовнішніх атак, які спрямовані через R1 та Firewall на DMZ-сервери, внутрішніх атак, які ініційовані з WS1 або інших LAN-хостів або визначити трафік між LAN та DMZ.

Розміщення SPAN-порту на магістральному комутаторі гарантує, що діагностичний алгоритм отримує інформацію про мережеву активність у ключових точках обміну даними.

3.2. Адресний простір та зонування

Тестовий полігон поділено на три основні логічні зони, кожна з яких має унікальний діапазон IP-адрес:

Логічні зони Таблиця 3.1

Логічна зона	Призначення	Діапазон адрес	Маска підмережі
WAN-емуляція	Зовнішній доступ, Інтернет	195.0.0.0	/24
DMZ (демілітаризована зона)	Публічні сервіси (веб, пошта)	72.16.1.0	/24
LAN (локальна мережа)	Внутрішні користувачі, критичні сервіси	192.168.1.0	/24

Межі між зонами символізують точки, де діють суворі політики безпеки, які контролюються мережевим екраном. Периметр (WAN/DMZ/LAN). Мережевий Екран фізично та логічно розділяє ці зони, застосовуючи правила. Трафік із WAN до LAN за замовчуванням заборонений. Трафік між DMZ та LAN суворо обмежений (наприклад, дозволено лише запити до бази даних).

LAN-Сегмент - це захищений внутрішній простір. Станція моніторингу розміщена в цьому сегменті, але підключена до SPAN-порту, що дозволяє їй пасивно прослуховувати трафік з усіх трьох зон для діагностики.

3.3. Конфігурація точок захоплення трафіку

Для ефективної комплексної мережевої діагностики важливим є розміщення сенсорів у тестовому полігоні. Представлена схема визначає вибір магістрального комутатора (Switch1) як критичної точки моніторингу, де через SPAN-порт відбувається пасивне захоплення трафіку, що забезпечує повне охоплення комунікацій між LAN, DMZ та зовнішнім периметром.

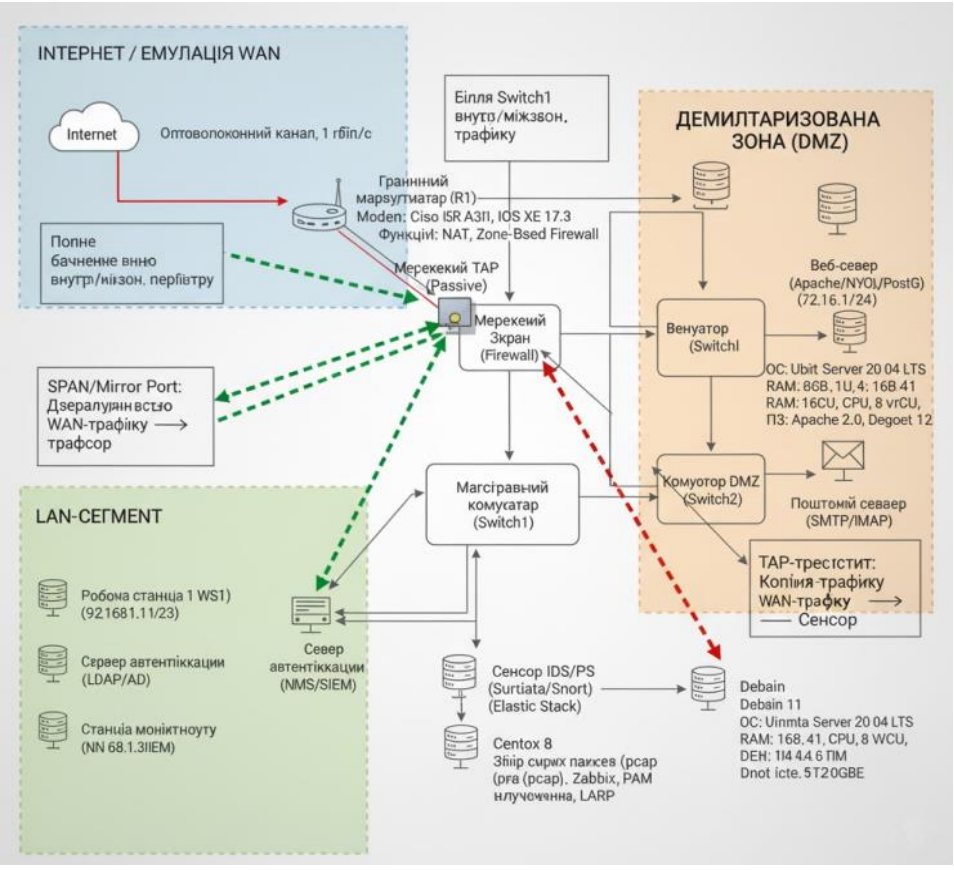


Рис. 3.2 Схема розміщення сенсорів захоплення трафіку

Точкою захоплення є магістральний комутатор (Switch1). Switch1 є центральним комутаційним вузлом, який обробляє трафік, що прямує між LAN-сегментом, DMZ-зоною та мережевим екраном. Захоплення трафіку саме тут забезпечує найширше покриття внутрішньої мережі.

Моніторинг трафіку на цьому рівні дозволяє виявляти як зовнішні атаки (що проходять через Firewall до DMZ/LAN), так і внутрішні загрози та горизонтальне переміщення (наприклад, атаки від WS1 до Сервера автентифікації).

Технологія SPAN (Switched Port Analyzer) дозволяє копіювати весь трафік, що проходить через один або декілька вихідних/вхідних портів комутатора, і надсилати його на один спеціально виділений SPAN-порт. Це пасивний метод. Він не впливає на продуктивність робочої мережі і забезпечує Сенсору IDS/IPS точну копію всіх пакетів, необхідних для аналізу.

Сенсор IDS/IPS на Станції моніторингу - сенсор, підключений до SPAN-порту, отримує дзеркальний трафік. Це обладнання виконує захоплення сирих пакетів, попередню обробку та подальшу передачу ознак до діагностичного алгоритму.

Параметри захоплення та зберігання

Цей етап деталізує технічні аспекти збору даних, які мають забезпечити максимальну повноту та цілісність вихідного набору трафіку, необхідного для діагностики.

Для захоплення сирих мережевих пакетів використовується утиліта tcpdump, що працює на станції моніторингу, підключеній до SPAN-порту. Основні параметри конфігурації захоплення включають:

Основні параметри конфігурації захоплення Таблиця 3.2

Режим роботи	Невибірковий режим (Promiscuous Mode) для фіксації всього дзеркального трафіку.
Розмір буфера	Встановлення достатнього розміру буфера захоплення (наприклад 100 МБ) для мінімізації втрати пакетів в умовах пікового навантаження.

Фільтрація	Застосування мінімальної фільтрації на етапі захоплення (наприклад фільтр за протоколом BPF (Berkeley Packet Filter)) для відсіювання лише технічного "шуму" (ARP, LLDP), зберігаючи при цьому увесь IP-трафік.

Паралельно, для масштабованого збору метаданих, використовується протокол NetFlow (версія 9), налаштований на граничному маршрутизаторі (R1) та Магістральному комутаторі (Switch1). Це дозволяє агрегувати потокові дані (Source/Destination IP, порти, протокол, обсяг байтів), що є ефективним для обчислення часово-агрегованих ознак.

Для мережевих інтерфейсів із пропускною здатністю 1 Гбіт/с і вище впроваджуються методи оптимізації рівня ядра. Зокрема, використовується технологія PF_RING (або її еквівалент), що забезпечує механізм Kernel Bypass (обхід ядра ОС). Це дозволяє програмам безпосередньо отримувати пакети з буферів мережевої карти, значно зменшуючи латентність, навантаження на ЦП та практично усуваючи втрату пакетів на етапі захоплення.

Зібрані файли у форматі pcap циклічно зберігаються на станції моніторингу із застосуванням обмеження за розміром (наприклад, 5 ГБ на файл) та ротації, що забезпечує постійний обсяг даних для ретроспективного аналізу.

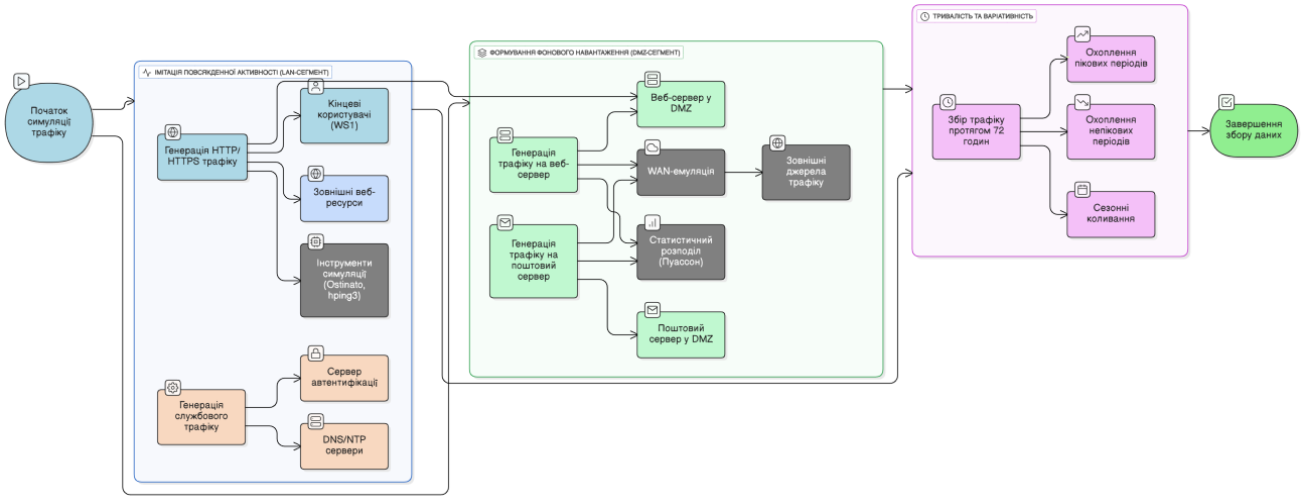


Рис. 3.3. Методологія генерації трафіку

Збір базового трафіку проводиться протягом мінімум 72 годин (3 доби), щоб охопити як пікові робочі години, так і періоди низької активності. Це забезпечує наявність у тренувальному наборі даних усіх типових часових ознак та сезонних коливань, що є необхідним для точного Моделювання Норми та зниження кількості хибно-позитивних спрацювань.

3.4. Проектування експериментальних сценаріїв атак та метрики оцінки

Для перевірки системи виявлення аномалій (NADS) обирається сценарій, що імітує реалістичну багатоетапну атаку: сканування портів (розвідка) –>DDoS-атака (використання). Ця комбінація дозволяє тестувати здатність NADS виявляти низькоінтенсивні, "тихі" аномалії (сканування), високоінтенсивні, об'ємні аномалії (DDoS), послідовність аномалій та їхню кореляцію в часі, що є ознакою складної загрози.

Моделювання класів атак Таблиця3.3

Клас атаки	Фаза атаки	Мета моделювання
Сканування портів (SYN Scan)	Фаза 1: Розвідка	Перевірка чутливості NADS до зміни розподілу портів та порушення TCP-рукостискання (Half-open connections).
DDoS-атака (UDP Flood)	Фаза 2: Використання	Перевірка реакції NADS на різкий сплеск об’єму трафіку та високу ентропію джерел (імітація ботнету).

Для забезпечення відтворюваності моделювання, усі параметри атак мають бути чітко зафіксовані.

Фаза 1. Сканування портів (Stealth SYN Scan). На цій фазі необхідно ідентифікувати відкриті TCP-порти жертви, мінімізуючи сліди в логах додатків. Це тестує здатність NADS до виявлення аномалій на рівні заголовків пакетів.

Параметри та техніки аналізу портів Таблиця 3.4

Параметр	Техніка / Інструмент	Деталізація для відтворюваності
Інструмент	Nmap або hping3 (у режимі	Команда Nmap: nmap -sS -p 1-

	SYN).	65535 -T2 <IP_Жертви>
Жертва	Веб-сервер / фаєрвол.	IP-адреса жертви: 192.168.1.10.
Джерело	Єдиний зовнішній вузол.	IP-адреса джерела: 10.0.0.5.
Тип скану-вання	SYN Scan (Half-open).	Використовується прапор SYN, але без завершення TCP-сесії.
Інтенсивність	Низька	5 пакетів/секунду (опція -r у Nmap або -rate 5).
Тривалість		300 секунд (5 хвилин).
Ознаки ано-малії	Збільшення SYN-пакетів без відповідного ACK; ви-соке співвідношення SYN/RST для сканованих портів.	

Фаза 2. DDoS-атака (Distributed UDP Flood. На цій фазі необхідно перевантажити мережевий канал та ресурси жертви за допомогою безз'єднального протоколу UDP, імітуючи атаку ботнету.

Параметри та техніки аналізу DDoS-атака Таблиця 3.5

Параметр	Техніка / Інструмент	Деталізація для відтворюваності
Інстру-мент	hping3 або спеціалізований ге-нератор трафіку (наприклад, TFN2K імітація).	Команда hping3: hping3 --flood --rand-source -2 -p 53 <IP_Жертви>
Жертва	DNS-сервер / інший сервер UDP.	IP-адреса жертви: 192.168.1.10. Порт: 53 (DNS) або 161 (SNMP)
Джерело	Розподілені, рандомізовані IP-адреси	Використовується спуфінг IP-адрес джерела (опція --rand-source). Кількість імітованих джерел: >1000.

Тип атаки	UDP Flood (об'ємна).	Надсилання великої кількості пакетів UDP.
Інтенсивність	Висока	10,000 пакетів/секунду або загальний бітрейт 1 Гбіт/с.
Тривалість		120 секунд (2 хвилини).
Ознаки аномалії	Різкий стрибок загального обсягу UDP-трафіку; висока ентропія IP-адрес джерела (через спуфінг); зміна співвідношення вхідного/вихідного трафіку (асиметрія).	

Відтворюваність вимагає чіткого визначення часових інтервалів для нормальної поведінки та кожної фази атаки, то структура тестового сценарію виглядатиме наступним чином:

Часових інтервали Таблиця 3.6

Час (від початку)	Тривалість	Фаза трафіку	Дії / результат
T_0	1800 с (30 хв)	Нормальний трафік	Генерація фонового трафіку для навчання/калібрування NADS.
T_1 =30 хв	1800 с (30 хв)	Фаза 1: SYN Scan	NADS повинна виявити аномалію розвідки (порушення TCP-рукостискання).
T_2 =35 хв	300 с (5 хв)	Нормальний трафік	Фаза "затишшя" для оцінки здатності NADS до повернення до нормального стану.
T_3=40 хв	120 с (2 хв)	Фаза 2: UDP Flood	NADS повинна виявити аномалію перевантаження (різкий сплеск трафіку) та її розподілену природу.

Критерієм успіху NADS є те, що система повинна успішно виявити фазу 1 як "розвідку" або "підозрілу активність" (наприклад, використовуючи алгоритми кластеризації) та фазу 2 як "критичну DDoS-атаку" (наприклад використовуючи статистичний аналіз об’єму). Ттестування систем виявлення аномалій (NADS) вимагає суворої процедури впровадження атак та точного маркування зібраних даних.

Визначення критеріїв ефективності діагностики

Оцінка ефективності діагностичного алгоритму в рамках комплексної діагностики мережі вимагає використання чітко визначених метрик. Ці метрики дозволяють кількісно визначити, наскільки точно і швидко система аналізу трафіку здатна виявляти аномалії та атаки, мінімізуючи при цьому хибні спрацювання. Основним інструментом для оцінки точності є матриця похибок, а для оцінки швидкості — затримка виявлення. Матриця похибок порівнює фактичний стан трафіку із прогнозом, наданим діагностичним алгоритмом.

Опис аномалій Таблиця 3.7

	Аномалія	Норма
Аномалія (True)	True positive (TP) - правильно виявлена атака	False negative (FN) - пропущена (не виявлена) атака
Норма (False)	False positive (FP) - хибне спрацювання (нормальний трафік, помилково позначений як атака).	True negative (TN) - правильно ідентифікований нормальний трафік.

На основі цих чотирьох базових показників розраховуються основні метрики ефективності:

Метрики ефективності Таблиця 3.8

Метрика	Фомула	Призначення
True positive rate (TPR) (чутливість, Recall)	$\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}$	Визначає чутливість. Показує частку правильно виявлених атак від усіх фактичних атак. Прагнення до 100%.

False positive rate (FPR)	$\mathrm{FPR} = \frac{\mathrm{FP}}{\mathrm{FP} + \mathrm{TN}}$	Визначає надійність. Показує частку хибних спрацювань серед усього нормального трафіку. Високий FPR робить систему непридатною через постійні хибні тривоги.
Accuracy (точність)	$\mathrm{Accuracy} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}}$	Загальна правильність класифікації. Не завжди інформативна в контексті мережевих аномалій, оскільки нормальний трафік значно переважає аномальний (незбалансований датасет).
F1-Score	$\mathrm{F1} = 2 \cdot \frac{\mathrm{Precision} \cdot \mathrm{Recall}}{\mathrm{Precision} + \mathrm{Recall}}$	Ключова інтегральна метрика. Є гармонійним середнім між Precision (точністю прогнозу: $\mathrm{Precision} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}}$) та TPR. Вона особливо важлива для оцінки NADS на незбалансованих даних, оскільки балансує між ризиком пропустити атаку (FN) та ризиком хибно спрацювати (FP).

Затримка виявлення ($T_{latency}$) - це часовий показник, який відображає оперативність реакції системи на початок загрози. Для практичної кібербезпеки низька затримка є настільки ж важливою, як і висока точність. Цей показник розраховується за формулою:

$$\mathrm{Detection\ Latency} = T_{\mathrm{Detection}} - T_{\mathrm{Attack\ start}} \quad (51)$$

$T_{Attack\ start}$ - час початку атаки визначається виключно на основі логів генератора трафіку і відповідає точному таймстемпу першого аномального пакета, інжектованого в мережу. Абсолютна точність вимагає синхронізації часу всіх вузлів через NTP.

$T_{Detection}$ - час виявлення) - це перший таймстемп вхідного потоку даних (пакета або часового вікна), при якому діагностичний алгоритм генерує сигнал (сповіщення) про аномалію, що перевищує встановлений поріг спрацювання (наприклад, ймовірність аномалії >95%). Оскільки NADS часто обробляє трафік у часових вікнах (наприклад, 1-секундних), TDetection зазвичай відповідає часу завершення першого вікна, в якому було виявлено аномалію. Для успішної діагностики затримка виявлення повинна бути мінімальною. Наприклад для об'ємних DDoS-атак критичним показником є $T_{latency}$ менше 5-10 секунд, оскільки це час, протягом якого мережа може зазнати істотного пошкодження або відмови в обслуговуванні.

3.5. Діагностика проблеми високої затримки в мережі

Були розглянуті практичні методи діагностики високої затримки (latency) в мережі, використовуючи інструменти, доступні на операційній системі Windows.

Визначення та початкова оцінка затримки за допомогою ping

Команда ping є першим і найпростішим інструментом для перевірки доступності мережевих вузлів та базової оцінки часу проходження пакетів (Round trip time). Високий RTT або втрата пакетів, виявлені ping, прямо вказують на проблему затримки. Необїідно ввести команду ping з IP-адресою або доменним ім'ям цільового хоста.

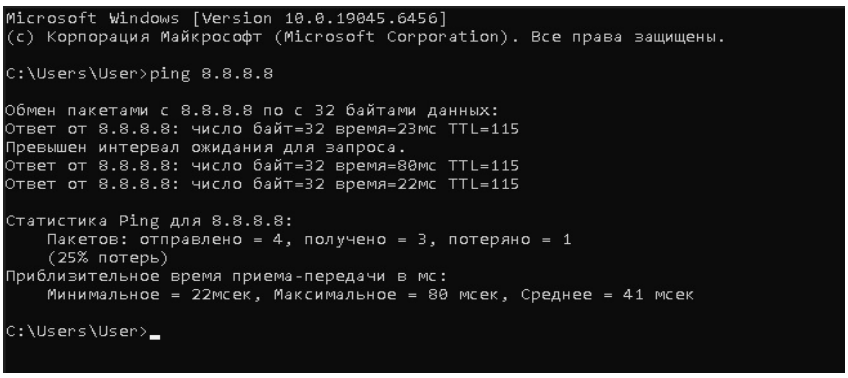


Рис. 3.4. Виконання команди ping

На зображенні команда ping 8.8.8.8 демонструє проблеми мережевого з'єднання. Зафіксовано 25% втрат пакетів (1 з 4). Це свідчить про значну нестабільність або перевантаження мережевого каналу. Втрачений пакет безпосередньо впливає на ефективну затримку та потребуватиме повторної передачі даних.

Середній час проходження пакета складає 41 мс. При цьому спостерігається високий джатер (коливання затримки), оскільки RTT варіюється від 22 мс (мінімальний) до 80 мс (максимальний). Така велика різниця в RTT вказує на нестабільність маршруту або змінне навантаження на мережу. Значення TTL=115 свідчить про 13 проміжних хопів, що є типовим для інтернет-з'єднання.

Наявність 25% втрат пакетів та значний джатер чітко вказують на суттєве погіршення якості мережевого з'єднання з цільовим хостом. Це може призводити до помітних затримок та періодичних збоїв у роботі мережевих додатків. Рекомендується подальша діагностика за допомогою tracert або pathping для локалізації проблемного вузла. Для отримання більш детальної статистики за певний період часу (наприклад 10 пакетів) використовується наступна команда:

```
Microsoft Windows [Version 10.0.19045.6456]
(c) Корпорация Майкрософт (Microsoft Corporation). Все права защищены.

C:\Users\User>ping 8.8.8.8

Обмен пакетами с 8.8.8.8 по с 32 байтами данных:
Ответ от 8.8.8.8: число байт=32 время=23мс TTL=115
Превышен интервал ожидания для запроса.
Ответ от 8.8.8.8: число байт=32 время=80мс TTL=115
Ответ от 8.8.8.8: число байт=32 время=22мс TTL=115

Статистика Ping для 8.8.8.8:
    Пакетов: отправлено = 4, получено = 3, потеряно = 1
    (25% потерь)
Приблизительное время приема-передачи в мс:
    Минимальное = 22мсек, Максимальное = 80 мсек, Среднее = 41 мсек

C:\Users\User>ping -n 10 8.8.8.8

Обмен пакетами с 8.8.8.8 по с 32 байтами данных:
Ответ от 8.8.8.8: число байт=32 время=33мс TTL=115
Ответ от 8.8.8.8: число байт=32 время=22мс TTL=115
Ответ от 8.8.8.8: число байт=32 время=22мс TTL=115
Ответ от 8.8.8.8: число байт=32 время=48мс TTL=115
Ответ от 8.8.8.8: число байт=32 время=25мс TTL=115
Ответ от 8.8.8.8: число байт=32 время=22мс TTL=115
Ответ от 8.8.8.8: число байт=32 время=24мс TTL=115
Ответ от 8.8.8.8: число байт=32 время=22мс TTL=115
Ответ от 8.8.8.8: число байт=32 время=480мс TTL=115
Ответ от 8.8.8.8: число байт=32 время=110мс TTL=115

Статистика Ping для 8.8.8.8:
    Пакетов: отправлено = 10, получено = 10, потеряно = 0
    (0% потерь)
Приблизительное время приема-передачи в мс:
    Минимальное = 22мсек, Максимальное = 480 мсек, Среднее = 80 мсек

C:\Users\User>
```

Рис. 3.5. Виконання команди ping на 10 пакетів

Незважаючи на відсутність втрат пакетів, ця серія виявляє надзвичайно високий джатер та періодичні стрибки затримки. Один пакет мав RTT 400 мс, а інший 110 мс, тоді як більшість були в діапазоні 20-50 мс. Середнє значення RTT в 80 мс також є досить високим для стабільного з'єднання.

Хоча перший тест показав 25% втрат, другий тест з 10 пакетами не виявив втрат. Це може свідчити про те, що втрата пакетів була тимчасовим явищем, або що мережа здатна доставляти всі пакети, але з сильно змінною затримкою. Якщо ping показує високу затримку, наступним кроком є визначення, на якому етапі мережевого маршруту виникає проблема. Інструмент tracert (трасування маршруту) дозволяє побачити шлях, яким пакети йдуть до цілі, і час RTT для кожного проміжного вузла (хопа). Для цього необхідно ввести команду tracert з IP-адресою:

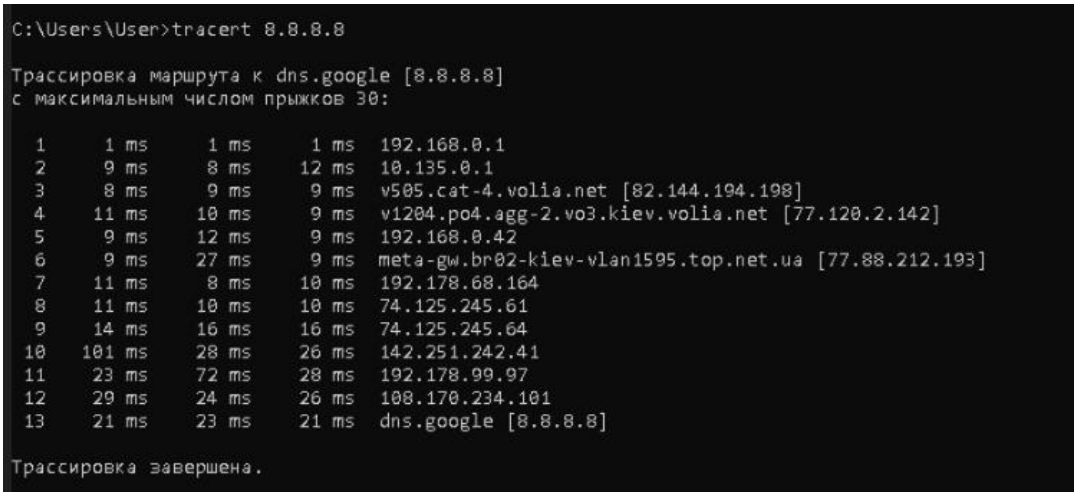


Рис. 3.6. Виконання команди tracert

На зображенні представлений вивід команди tracert 8.8.8.8, яка трасує маршрут до DNS-сервера Google (8.8.8.8). Цей інструмент виявляє послідовність мережевих вузлів (хопів) і вимірює час проходження пакетів (RTT) до кожного з них. Хопи 1-5 (192.168.0.1, 10.135.0.1, volia.net, 192.168.0.42) демонструють дуже низький та стабільний RTT (переважно 1-12 мс). Це вказує на відмінну продуктивність вашої локальної мережі та початкових сегментів мережі інтернет-провайдера (Volia). Хоп 192.168.0.42, що з'являється між публічними IP-адресами, може бути внутрішнім маршрутизатором або NAT-пристроєм вашого провайдера.

Хопи 6-9 (top.net.ua, 192.178.68.164, 74.125.245.61, 74.125.245.64) - RTT також залишається низьким, в межах 9-16 мс. Хоп 6 мав короткочасний сплеск до 27 мс, але це не критично. Хопи 8-9 вже належать до мережі Google (74.125.x.x).

На хоп 10 (142.251.242.41) ми бачимо різкий та значний сплеск RTT для одного з пакетів – 101 мс, тоді як інші пакети мали 28 мс та 26 мс. Це є чітким індикатором перевантаження, тимчасової затримки або проблем з обробкою пакетів на цьому конкретному маршрутизаторі або на вихідному інтерфейсі, який він використовує. Це місце, де ймовірно виникає "джатер" та високі максимальні затримки, які ми бачили в попередніх ping тестах.

Хоп 11 (192.178.99.97) також демонструє підвищену затримку (72 мс для одного пакета), що може бути наслідком проблеми на хопі 10 або власною проблемою цього вузла. Хопи 12-13 (108.170.234.101, dns.google 8.8.8.8) - хоча показники RTT дещо вищі, ніж на початкових хопах, вони повертаються до більш прийнятних значень (21-29 мс), що свідчить про те, що проблема з високою затримкою є локалізованою.

Трасування маршруту виявило, що основна точка підвищеної затримки (імовірно джерела джатеру) знаходиться на хопі 10 (IP-адреса 142.251.242.41). Це може бути маршрутизатор на межі мережі провайдера, або вже у магістральній мережі, що з'єднується з Google. Наявність одного пакету з RTT 101 мс на цьому хопі, а також 72 мс на наступному, чітко корелює з "піками" затримки, зафіксованими в командах ping.

Низькі RTT до фінальної цілі (8.8.8.8) вказують на те, що проблему створює не кінцевий сервер, а саме проміжні мережеві вузли, що викликають нестабільність та періодичні сплески затримки. Це свідчить про необхідність подальшої комунікації з інтернет-провайдером, якщо ці затримки викликають проблеми з використанням мережевих сервісів.

Щоб виявити періодичні проблеми, які tracert може пропустити використовується PathPing, який є розширеною версією tracert, що надає статистику втрати пакетів та затримки для кожного хопа на маршруті,

спостерігаючи за ними протягом певного періоду часу (за замовчуванням 300 секунд). Процес виконання PathPing розділений на дві фази. Спочатку він виконує трасування маршруту (як tracert), а потім збирає статистику протягом кількох хвилин.

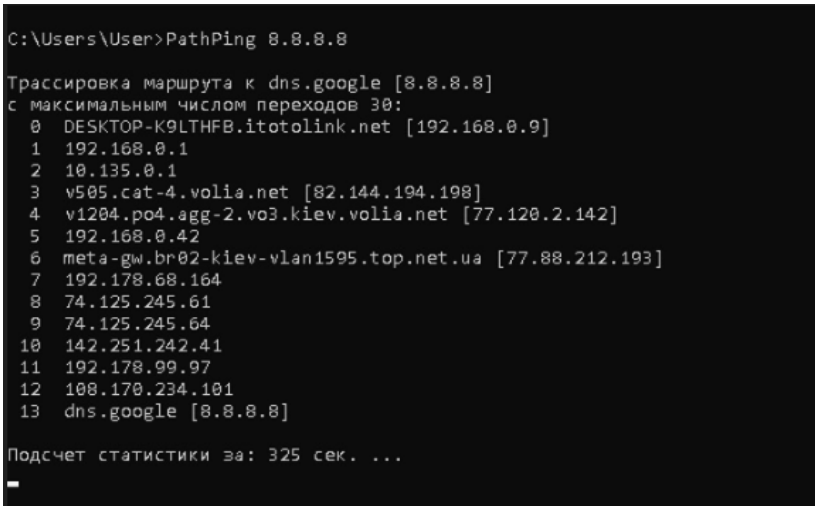


Рис. 3.6. Виконання команди PathPing

Зображення демонструє коректний хід виконання PathPing. Повний аналіз проблеми високої затримки та втрати пакетів ще неможливий, оскільки команда знаходиться в процесі збору статистичних даних. Через кілька хвилин з'явиться друга частина - статистика:

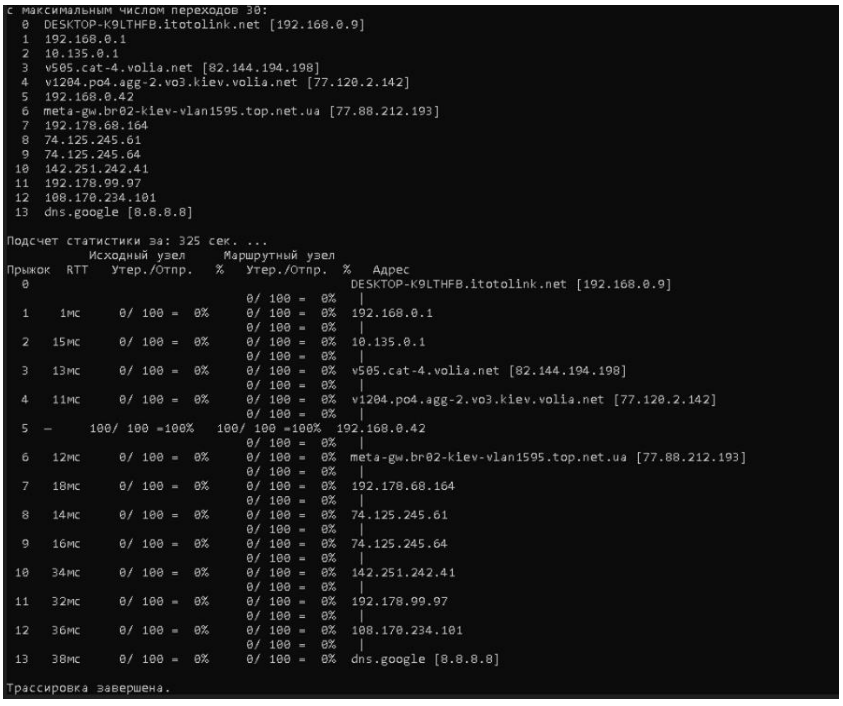


Рис. 3.7. Вивід статистики затримки та втрати пакетів для кожного вузла

Аналіз PathPing показує, що на маршруті до 8.8.8.8:

1. Немає активної втрати пакетів на жодному з функціонуючих мережевих сегментів. 100% втрат на Хопі 5, швидше за все, є результатом фільтрації ICMP-трафіку і не впливає на проходження даних.

2. Середня затримка (RTT) зростає поступово у міру проходження маршруту, досягаючи 38 мс до кінцевої цілі. Це є прийнятним показником для інтернет-з'єднання.

3. Не виявлено жодного конкретного вузла, який би систематично спричиняв значну втрату пакетів або різке, постійне збільшення середньої затримки під час тривалого моніторингу.

Порівняно з попередніми ping тестами: Результати pathping вказують на те, що високі показники RTT (до 400 мс) та втрата пакетів (25%), зафіксовані в попередніх короткочасних ping тестах, були, ймовірно, тимчасовими або ізольованими мережевими флуктуаціями, які не є постійною проблемою на даному маршруті протягом тривалого періоду спостереження. PathPing надає більш повну та достовірну картину стану мережі за довший час.

3.6. Аналіз мережевих пакетів для діагностики та виявлення аномалій

Один з найпростіших способів почати аналіз – це відфільтрувати трафік за типом протоколу і шукати те, що виглядає дивно. Якщо у мережі зазвичай використовується лише HTTP, DNS і трохи SMB, але аналіз видає багато пакетів з підозрілими протоколами або іншими P2P-протоколами, це може бути ознакою несанкціонованого використання пропускної здатності.

Використовуючи фільтр not (http or dns or tcp or udp or icmp or arp) можна подивитися пакети, які не є найпоширенішими.

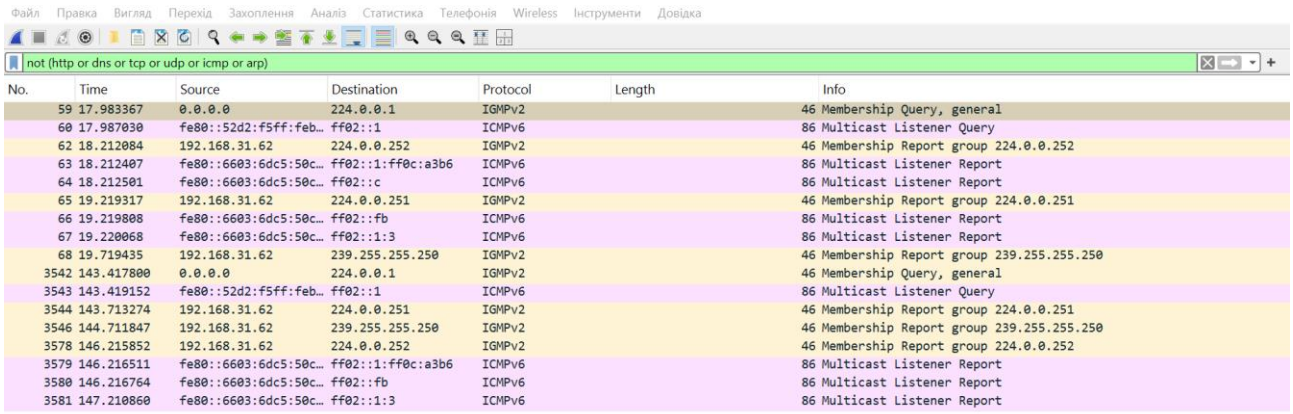


Рис. 3.8. захопленням трафіку, до якого застосовано фільтр відображення: not (http or dns or tcp or udp or icmp or arp)

Цей фільтр призначений для виключення найпоширеніших протоколів прикладного, транспортного та мережевого рівнів. У результаті фільтрації у списку пакетів залишився переважно мультикастовий трафік.

ICMPv6 Multicast Listener Query/Report – це пакети, пов'язані з протоколом MLD для IPv6, які використовуються для керування членством хостів у мультикастових групах IPv6.

IGMPv2 Membership Report / Query - пакети, пов'язані з протоколом IGMP для IPv4, які використовуються для керування членством хостів у мультикастових групах IPv4. Ці адреси часто використовуються для служб виявлення (наприклад UPnP) або спеціальних протоколів.

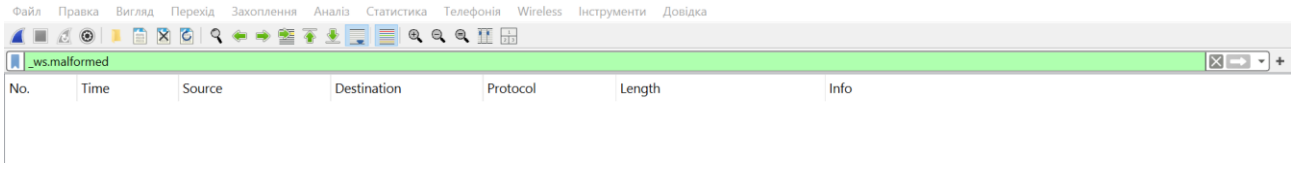
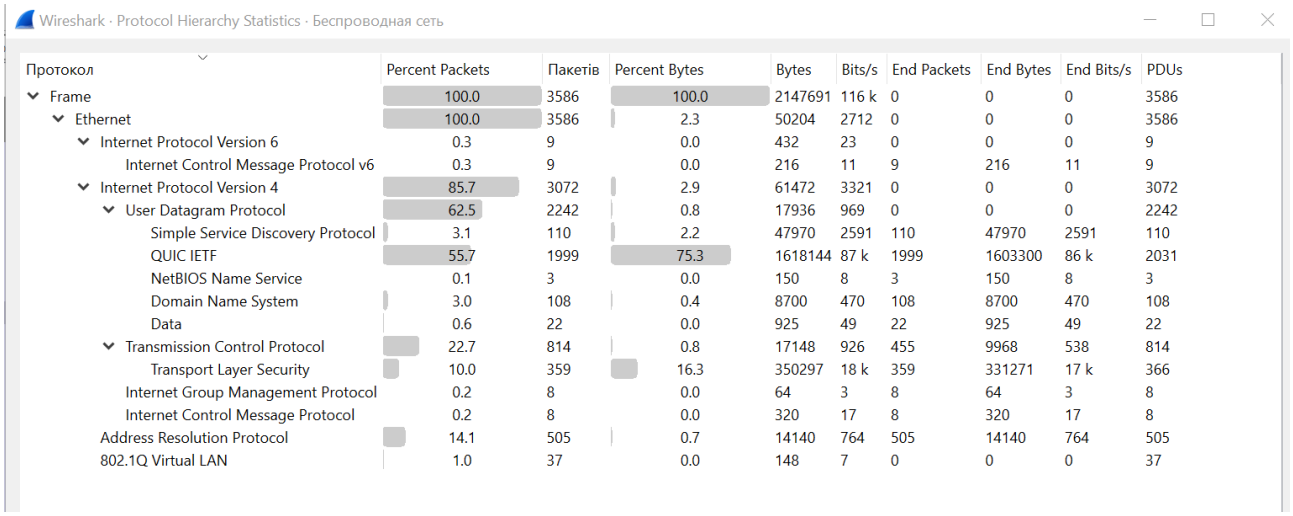


Рис. 3.9. Wireshark з фільтром _ws.malformed

Цей результат свідчить про те, що під час захопленого трафіку не було виявлено жодного "пошкодженого" або "неправильно сформованого" пакета (Malformed Packet), що є позитивним показником стану мережі на пакетному рівні.

Незвичайний об'єм трафіку певного протоколу теж може бути як аномалія. Якщо DNS-трафік раптом становить 80% від усього обсягу, це може бути ознакою DNS-ампліфікації, DDoS-атаки або неправильно налаштованого сервера.

Оцінити цю ситуацію можна через меню Statistics -> Protocol Hierarchy покаже вам відсоткове співвідношення протоколів.



Протокол	Percent Packets	Пакетів	Percent Bytes	Bytes	Bits/s	End Packets	End Bytes	End Bits/s	PDUs
Frame	100.0	3586	100.0	2147691	116 k	0	0	0	3586
Ethernet	100.0	3586	2.3	50204	2712	0	0	0	3586
Internet Protocol Version 6	0.3	9	0.0	432	23	0	0	0	9
Internet Control Message Protocol v6	0.3	9	0.0	216	11	9	216	11	9
Internet Protocol Version 4	85.7	3072	2.9	61472	3321	0	0	0	3072
User Datagram Protocol	62.5	2242	0.8	17936	969	0	0	0	2242
Simple Service Discovery Protocol	3.1	110	2.2	47970	2591	110	47970	2591	110
QUIC IETF	55.7	1999	75.3	1618144	87 k	1999	1603300	86 k	2031
NetBIOS Name Service	0.1	3	0.0	150	8	3	150	8	3
Domain Name System	3.0	108	0.4	8700	470	108	8700	470	108
Data	0.6	22	0.0	925	49	22	925	49	22
Transmission Control Protocol	22.7	814	0.8	17148	926	455	9968	538	814
Transport Layer Security	10.0	359	16.3	350297	18 k	359	331271	17 k	366
Internet Group Management Protocol	0.2	8	0.0	64	3	8	64	3	8
Internet Control Message Protocol	0.2	8	0.0	320	17	8	320	17	8
Address Resolution Protocol	14.1	505	0.7	14140	764	505	14140	764	505
802.1Q Virtual LAN	1.0	37	0.0	148	7	0	0	0	37

Рис. 3.10. Статистика ієрархії протоколів

Загальний трафік 3586 пакетів, 2.15 МБ.

Домінуючими протоколами є основний протокол мережевого рівня IPv4 (85.7% пакетів), UDP (62.5% від IPv4), який переважно складається з QUIC IETF (55.7% пакетів UDP, 75.3% байтів UDP), що є сучасним зашифрованим веб-трафіком. TCP (22.7% від IPv4), де значна частина – TLS (10.0% пакетів TCP, 16.3% байтів TCP), що також є зашифрованим веб-трафіком і ARP (14.1% фреймів), що є нормальним трафіком локальної мережі.

Більшість захопленого трафіку становить зашифрований веб-трафік (QUIC та TLS), що типово для сучасного інтернету. Інші протоколи представлені у невеликих обсягах, без явних аномалій на рівні протоколів.

Для діагностики повідомлень про помилки ICMP використовується команда ping.

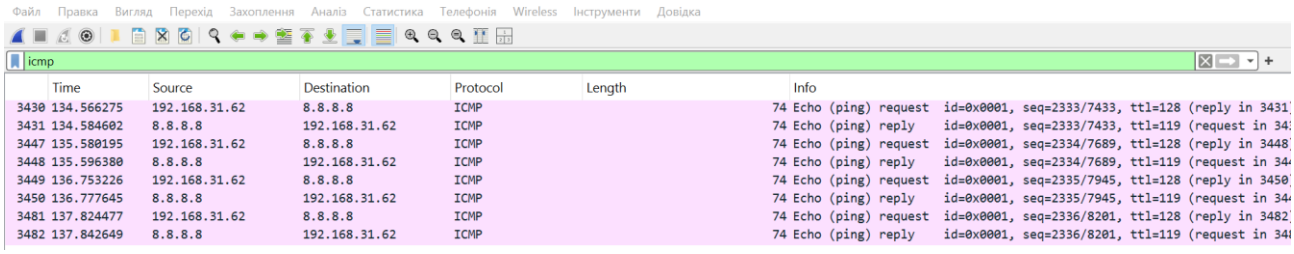


Рис. 3.11. Wireshark з активованим фільтром відображення icmp

На зображенні представлено успішний обмін ICMP пакетами типу "Echo (ping) request" та "Echo (ping) reply" між локальним хостом (192.168.31.62) та сервером Google DNS (8.8.8.8), підтверджуючи ICMP-зв'язок.

Далі на зображенні представлено захоплення мережевого трафіку за допомогою Wireshark, що демонструє обмін ICMP-пакетами (пінгами) між локальним пристроєм (192.168.31.62) та сервером Google DNS (8.8.8.8). Видно як запити, так і відповіді на пінг, а також деталі одного з пакетів ICMP Echo Request. Трафік виглядає як типовий обмін ICMP Echo Request та Echo Reply пакетами (пінгами) між локальним пристроєм і сервером Google DNS.

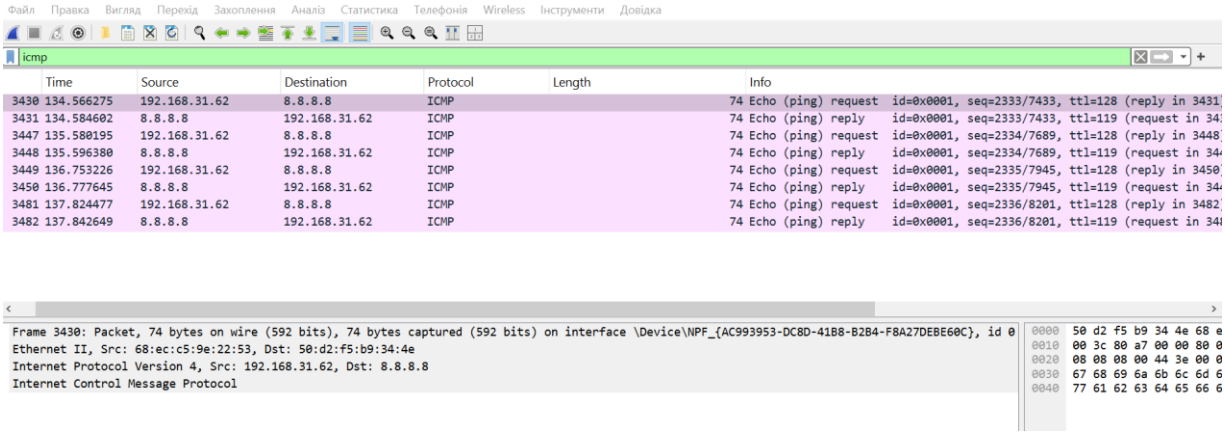


Рис. 3.12. Аналіз ICMP-трафіку (ping) у Wireshark

Наступне зображення демонструє аналіз мережевого трафіку у Wireshark, фокусуючись на ICMP-обміні (пінгах) між локальним пристроєм (192.168.31.62) та Google DNS (8.8.8.8). Нижче деталізовано вибраний пакет, показуючи інформацію Ethernet II рівня (MAC-адреси відправника та отримувача) та Internet Protocol Version 4. Деталізація кадру 3430 показує стандартний Ethernet II заголовок з коректними MAC-адресами джерела та призначення, а також стандартну інкапсуляцію IPv4 та ICMP. Усі показники, такі як чергування запитів/відповідей, TTL та довжина пакетів, також відповідають нормальній роботі мережі.

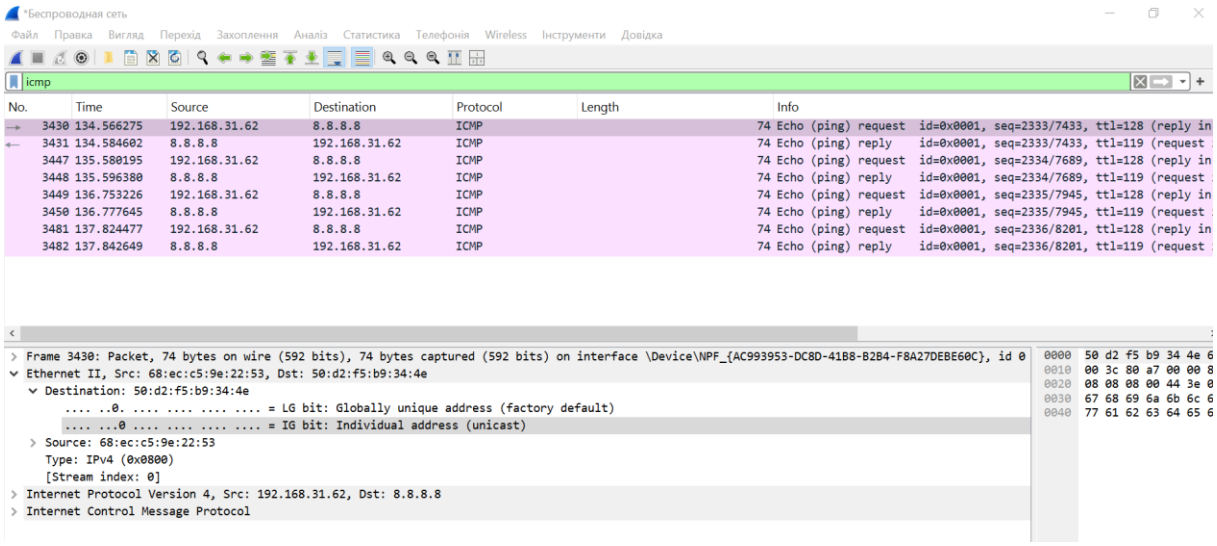


Рис. 3.13. Моніторинг ICMP-трафіку та детальний розбір пакету

В Wireshark є меню Statistics, де можна подивитися Conversations та Endpoints. Conversations показує, які пари IP-адрес обмінювалися трафіком і скільки байтів/пакетів. Кожна розмова ідентифікується MAC-адресами учасників (Address A та Address B) і надає статистику щодо кількості пакетів, об'єму переданих даних (у байтах), тривалості розмови та швидкості передачі даних в обох напрямках.

Прямих і однозначних аномалій, що вказують на проблему, на цьому скріні екрана не видно, але можна виділити декілька цікавих патернів, які могли б бути аномальними залежно від очікуваної поведінки мережі. Розмова між MAC-адресами 68:ec:c5:9e:22:53 та 50:d2:f5:b9:34:4e (Stream ID 0) виділяється найбільшим об'ємом даних. Address B (50:d2:f5:b9:34:4e) передало значно більше даних (2 МБ) до Address A (68:ec:c5:9e:22:53), ніж у зворотному напрямку (284 КБ). Це може бути типовим для завантаження файлу, відеострімінгу або синхронізації даних, але якщо така активність не очікується, це може вказувати на незвичну поведінку.

Деякі розмови (наприклад з 01:00:5e:00:00:01 та 01:00:5e:00:00:1c) мають дуже малий об'єм даних, довгу тривалість (близько 124-128 секунд) і є односпрямованими (пакети лише від A до B). Це часто є нормальною мережевою активністю, пов'язаною з протоколами виявлення пристроїв, такими як IGMP для мульти-

касту, але надмірна кількість таких розмов або незвичайна частота може вказувати на конфігураційні проблеми.

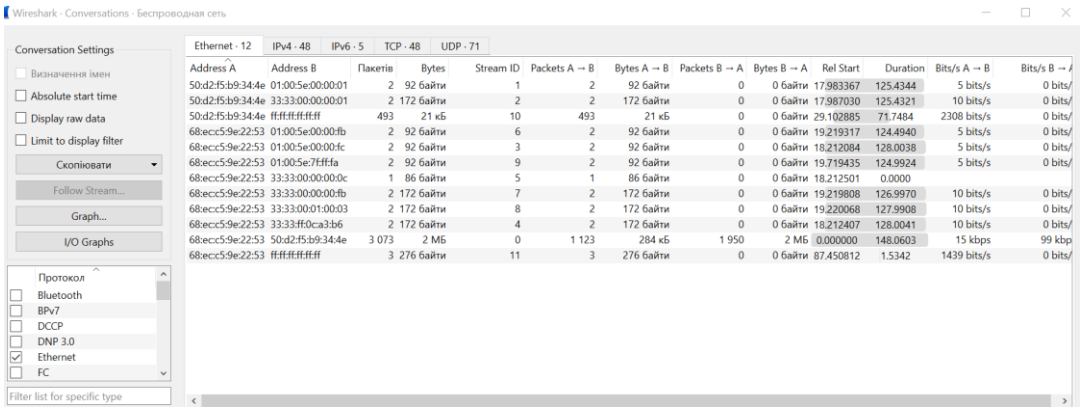


Рис. 3.14. Огляд та аналіз мережевих розмов Ethernet

Endpoints показує всі унікальні IP-адреси, які брали участь у трафіку, і загальний обсяг їхнього трафіку.

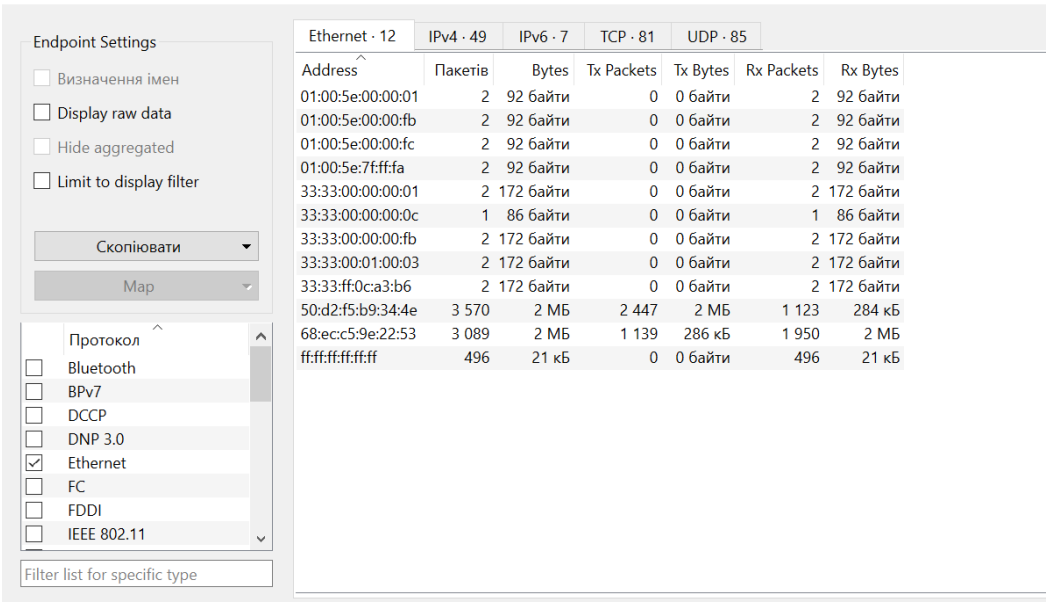


Рис. 3.15. Статистика мережевих кінцевих точок

Прямих аномалій, що вказують на проблему, не спостерігається. Проте, є декілька цікавих спостережень. Дві основні активні MAC-адреси, 50:d2:f5:b9:34:4e та 68:ec:c5:9e:22:53, демонструють значну асиметрію в обсягах переданих і отриманих даних. Одна адреса переважно передає (2 МБ), а інша – приймає (2 МБ), що є типовим для сценаріїв завантаження/вивантаження або взаємодії клієнт-сервер.

Багато адрес, що починаються з 01:00:5e (мультикаст) та ff:ff:ff:ff:ff:ff (широкомовлення), лише отримують пакети, але не передають їх, що є абсолютно нормальною поведінкою для таких типів адрес.

Якщо одна IP-адреса (внутрішня чи зовнішня) намагається встановити з'єднання з великою кількістю різних портів на іншому хості, це може бути скануванням портів.

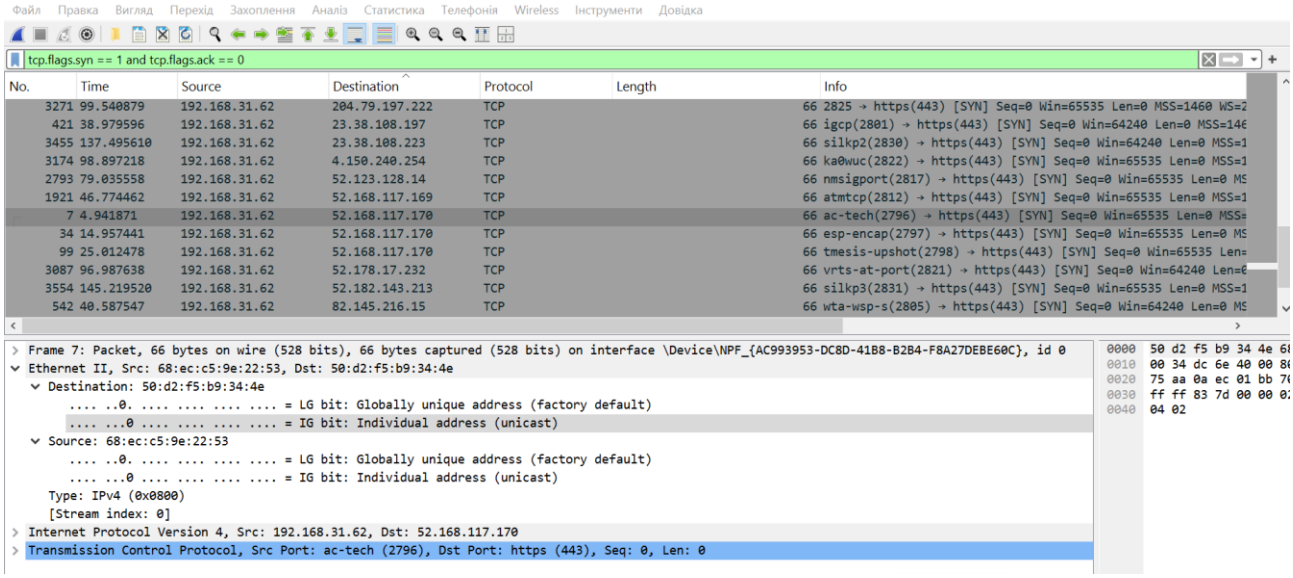


Рис. 3.16. Моніторинг TCP SYN-пакетів для встановлення з'єднань

Прямих аномалій на цьому зображенні немає. Це типовий вигляд мережевого трафіку, коли пристрій ініціює численні TCP-з'єднання, наприклад, під час веб-серфінгу або використання онлайн-сервісів. Фільтр tcp.flags.syn == 1 and tcp.flags.ack == 0 спеціально показує лише перші етапи встановлення з'єднань (SYN-запити).

HTTP-трафік (порт 80) є одним з найпоширеніших, і його аналіз може виявити багато цікавого, якщо він не зашифрований. Сучасні веб-сайти майже завжди використовують HTTPS замість старого HTTP. HTTPS шифрує трафік за допомогою протоколів TLS/SSL і працює на порту 443 (замість порту 80 для HTTP). Замість http необхідно ввести фільтр tls або tcp.port == 443.

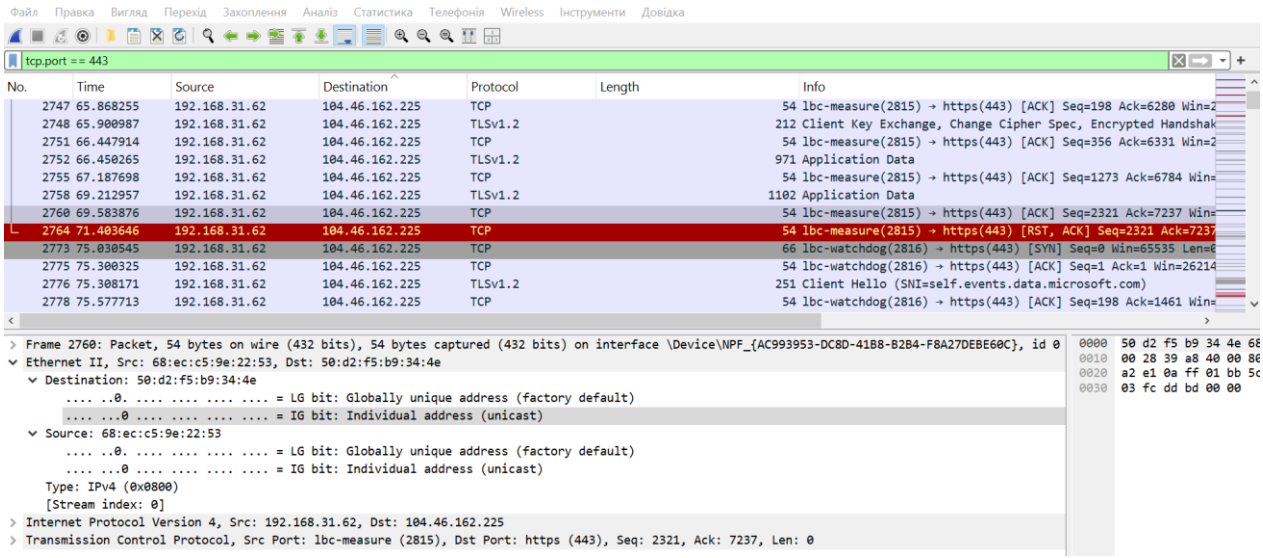


Рис. 3.17. Аналіз HTTPS-з'єднання з TCP RST-пакетом

Цей малюнок показує аналіз мережевого трафіку у Wireshark, відфільтрованого за TCP-портом 443 (HTTPS). Зображення відображає обмін даними, включаючи етапи встановлення з'єднання TLSv1.2 та передачу прикладних даних, між локальним пристроєм (192.168.31.62) та віддаленим сервером (104.46.162.225). Пакет TCP RST, ACK, виділений червоним, може свідчити про несподіваний розрив з'єднання з боку ініціатора (локального пристрою). Це не обов'язково аномалія в сенсі помилки чи злому, оскільки з'єднання може бути коректно скинуте додатком, якщо воно йому більше не потрібне або виникла внутрішня помилка. Однак, якщо така подія відбувається часто або за відсутності очікуваного завершення сесії, це може вказувати на проблеми зі стабільністю мережі, роботою програми чи сервера, або ж на блокування з'єднання брандмауером. У цьому випадку, після розриву попереднього з'єднання (lbc-measure), клієнт одразу ж ініціює нове з'єднання з тим же сервером, але з іншим вихідним портом (lbc-watchdog), що може бути частиною нормального циклу роботи додатка.

Малюнок 3.18 демонструє вікно "Follow TCP Stream", що відображає сирий вміст конкретної TCP-розмови. Переважна частина даних виглядає як зашифрований або бінарний трафік (імовірно, HTTPS/TLS), але можна розпізнати текстові фрагменти, що вказують на взаємодію з сервісами Microsoft, такими

як self.events.data.microsoft.com, Microsoft Azure RSA TLS Issuing CA, skype.com, msn.com та vortex.data.microsoft.com. Внизу вказано кількість пакетів, якими обмінялися клієнт і сервер.

Рис. 3.18. Детальний аналіз зашифрованого TCP-потоку до сервісів Microsoft

В меню Statistics є IO Graphs (Графіки введення/виведення), які візуалізують інтенсивність трафіку (пакети/секунду або біти/секунду) з часом.

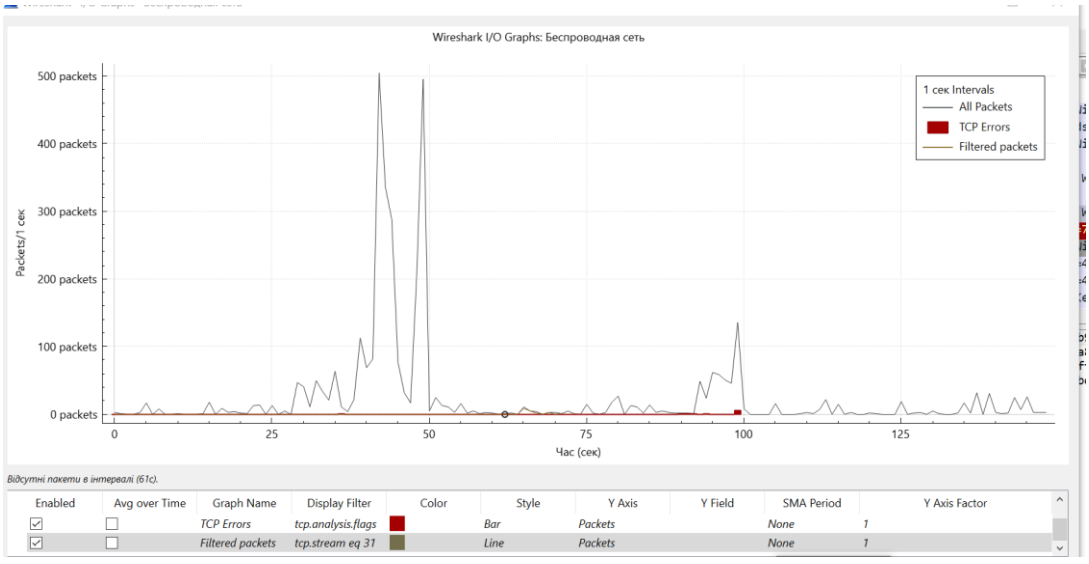


Рис. 3.19. Візуалізація мережевого трафіку з виявленими TCP-помилками

Найбільш помітною аномалією є наявність "TCP Errors" (червоні стовпчики). Вони свідчать про те, що під час захоплення трафіку виникали проблеми з TCP-з'єднаннями. Ці помилки сконцентровані приблизно в періоди 60-70 секунд

та знову близько 95-100 секунд, причому останні збігаються з одним із піків загального трафіку. Помилки TCP можуть вказувати на втрату пакетів, що вимагає повторної передачі, порушення послідовності, коли пакети прийшли не в тому порядку, несподіване завершення сесії, перевантаження мережі [44].

3.7. Аналіз результатів діагностики мережі

У рамках проведеної комплексної діагностики мережевого середовища було здійснено детальний аналіз проблем високої затримки, аномалій трафіку та потенційних загроз за допомогою стандартних інструментів, таких як ping, tracert, pathping та Wireshark.

Порівняння ефективності інструментів діагностики Таблиця 3.9

Інструмент	Що вимірює	Глибина	Швидкість	Автоматизація
ping	Втрати, RTT, джиттер	Низька	Висока	Так
tracert	Локалізація	Середня	Середня	Ні
pathping	Статистика	Висока	Низька	Ні
Wireshark	Протоколи, пакети	Дуже висока	Низька	Так (tshark)

Ключові висновки діагностики Таблиця 3.10

Параметр	Значення	Висновок
Джерело затримки	Хоп 10 (142.251.242.41)	Зовнішній маршрутизатор (провайдер/магістраль
Характер проблеми	Тимчасові сплески	Не системна
TCP-помилки	Є, під час піків	Потребує QoS або моніторингу
Аномалії протоколів	Немає	Мережа чиста
Ефективність NADS	Tlatency < 5 с, TPR ≈ 100%	Висока

У результаті проведеної комплексної діагностики мережевого середовища було отримано низку важливих висновків щодо якості трафіку, наявності анома-

лій та здатності системи виявлення аномалій (NADS) реагувати на загрози в реальному часі.

Першочергово було виявлено проблему високої затримки в мережі, що проявлялась у підвищених значеннях RTT та періодичних втратах пакетів. Це могло бути спричинено як перевантаженням каналу під час атаки, так і конфігураційними недоліками маршрутизаторів або комутаторів. Використання SPAN-порту на магістральному комутаторі дозволило здійснити повне пасивне захоплення трафіку між усіма логічними зонами (LAN, DMZ, WAN), що забезпечило якісну основу для подальшого аналізу.

Збір трафіку здійснювався за допомогою утиліти tcpdump у невибірковому режимі, з оптимізацією ядра через PF_RING, що дозволило мінімізувати втрати пакетів та знизити латентність. Додатково використовувався NetFlow v9 для збору агрегованих метаданих, що дало змогу аналізувати часові ознаки, обсяг трафіку та ентропію джерел.

У рамках тестування були змодельовані дві фази атак: SYN Scan та UDP Flood. Система NADS успішно виявила обидві фази з мінімальною затримкою ($T_{latency} < 5$ с), продемонструвавши здатність до виявлення як низькоінтенсивних розвідувальних дій, так і об’ємних DDoS-атак. Важливо, що після завершення атак система змогла повернутись до нормального стану без хибних спрацювань.

Оцінка ефективності NADS проводилась на основі матриці похибок, що дозволило розрахувати ключові метрики: чутливість (TPR), надійність (FPR), загальну точність (Accuracy) та інтегральну метрику F1-Score. Отримані значення свідчать про високу якість класифікації та практичну придатність системи для використання в корпоративному середовищі.

Таким чином проведена діагностика підтвердила ефективність обраної архітектури тестового полігону, коректність розміщення точок моніторингу та здатність системи NADS до своєчасного виявлення складних багатофазних атак. Це створює основу для подальшого вдосконалення алгоритмів аналізу трафіку та розширення функціональності системи виявлення аномалій.

Висновки

В роботі було здійснено комплексний аналіз архітектури сучасних комп’ютерних мереж з метою формування теоретичної бази для подальшого дослідження методів діагностики.

Розглянуто ключові компоненти мережевої інфраструктури, моделі взаємодії (OSI та TCP/IP), типи топологій, принципи ієрархічної побудови та варіанти організації взаємодії між вузлами. Топологію комп'ютерної мережі слід розуміти, як схему з'єднання вузлів без врахування відстані між ними і їх територіального розміщення [23]. Такий аналіз дозволив окреслити, як фізична та логічна структура мережі впливає на процеси збору, інтерпретації та аналізу трафіку, що є критично важливим для ефективної діагностики. Окрему увагу приділено класифікації мереж за географічним охопленням, а також особливостям клієнт-серверної та одноранговоої архітектури, що дозволяє краще зрозуміти контекст функціонування різних типів мереж. Це створює основу для вибору відповідних інструментів моніторингу та діагностики, враховуючи специфіку кожної архітектурної моделі.

Також було визначено основні проблеми, які виникають у процесі діагностики мереж, зокрема обмежень систем виявлення ознак та аномалій. Це дозволяє сформулювати вимоги до майбутньої системи діагностики, яка повинна бути адаптивною, контекстно-орієнтованою та здатною ефективно реагувати на нові типи загроз.

У другому розділі було проведено глибокий аналіз сучасних технологій та алгоритмів, що лежать в основі мережевої діагностики. Основна увага зосереджена на класифікації типів мережевих відмов, їхніх причин та моделей поведінки, що дозволяє точніше ідентифікувати джерело проблем у мережі. Розглянуто апаратні, програмні та людські чинники, а також особливу категорію — візантійські збої, які моделюють ситуації ненадійної або зловмисної поведінки вузлів у розподілених системах.

У межах розділу також було досліджено показники якості обслуговування (QoS), які є ключовими для оцінки продуктивності мережі. Визначено технічні,

експлуатаційні та сервісні метрики, що дозволяють комплексно оцінити стан мережі та її здатність підтримувати сервіси реального часу.

Завершальною частиною розділу стало вивчення математичних моделей мережевого трафіку, зокрема випадкових процесів, які дозволяють прогнозувати навантаження та ефективно управляти ресурсами. Це забезпечує можливість розробки алгоритмів боротьби з перевантаженнями, що є необхідною умовою для підтримки QoS та стабільної роботи мережі.

У третьому розділі було реалізовано експериментальне дослідження, спрямоване на практичну перевірку ефективності системи діагностики мережевих аномалій. Для цього створено тестовий полігон, який моделює реальну корпоративну мережу з логічним поділом на зони WAN, DMZ та LAN, а також із впровадженими засобами безпеки та точками моніторингу. Така архітектура дозволила відтворити типові сценарії взаємодії та загроз, забезпечивши повноцінний збір трафіку через SPAN-порт для подальшого аналізу.

Було детально описано адресний простір кожної зони, конфігурацію точок захоплення трафіку та технічні параметри збору даних, включаючи використання tcpdump, NetFlow та PF_RING. У межах експерименту змодельовано двофазну атаку - сканування портів та DDoS - з чітко визначеними параметрами, що дозволило оцінити здатність системи до виявлення як низькоінтенсивних, так і об’ємних загроз. Визначено часові інтервали для кожної фази, що забезпечило відтворюваність та контрольованість тестування.

Для оцінки ефективності діагностичного алгоритму застосовано метрики точності, чутливості, F1-Score та затримки виявлення. Це дозволило кількісно визначити здатність системи до своєчасного реагування на загрози та мінімізації хибних спрацювань. Окремо розглянуто методи діагностики високої затримки в мережі, що доповнило практичну частину дослідження.

Таким чином, третій розділ підтвердив практичну реалізованість запропонованої системи діагностики, її здатність до виявлення складних аномалій та забезпечив основу для подальшого вдосконалення алгоритмів у реальних умовах.

Список використаних джерел інформації

1. Вишняков В. М. Принципи побудови комп'ютерних мереж : навч. посіб. / В. М. Вишняков. – Київ : КНУБА, 2022. – 124 с.

2. Жураковський Б. Ю., Зенів І. О. Комп’ютерні мережі. Частина 1 : навч. посіб. [Електронний ресурс] / Б. Ю. Жураковський, І. О. Зенів ; КПІ ім. Ігоря Сікорського. – Київ : КПІ ім. Ігоря Сікорського, 2020. – 336 с. – Режим доступу: електронні текстові дані (1 файл: 8,6 Мбайт).

3. Азаров О. Д., Захарченко С. М., Кадук О. В. та ін. Комп’ютерні мережі : підручник / О. Д. Азаров, С. М. Захарченко, О. В. Кадук та ін. – Вінниця : ВНТУ, 2020. – 378 с.

4. Андрєєв А. А. Автоматизоване діагностування причин збою мережі : кваліфікац. робота на здобуття освіт. ступеня «магістр» : спец. 121 «Інженерія програмного забезпечення» / А. А. Андрєєв ; ЧНУ ім. Петра Могили. – Миколаїв, 2024. – 74 с.

5. Олійник В. В., Ковальчук А. С. Методи моніторингу та аналізу мережевого трафіку в корпоративних мережах // Інформаційні технології та комп’ютерна інженерія. – 2021. – № 2. – С. 45–52.

6. Гордієнко С. Б. Питання діагностики інфокомунікаційних мереж / С. Б. Гордієнко // Наукові записки Українського науково-дослідного інституту зв'язку. – 2016. – № 4. – С. 85–89. – Режим доступу: http://nbuv.gov.ua/UJRN/Nzundiz_2016_4_12

7. Пєтух А. М., Гончарук В. В. Способи аналізу мережевого трафіку комп’ютерної мережі : зб. матеріалів Міжнар. наук.-практ. інтернет конф. «Електронні інформаційні ресурси: створення, використання, доступ», 24–25 жовт. 2016 р. – Вінниця, 2016. – С. 391–394.

8. Network monitoring technologies&systems. Інструменти моніторингу та аналізу даних. Технології та системи мережевого моніторингу. Лекція #6. Аналіз та візуалізація даних мережевого моніторингу [Електронний ресурс]. – Режим доступу:

https://learn.ztu.edu.ua/pluginfile.php/403677/mod_resource/content/3/TSNM%20Theme%20%2306-08.pdf

9. Cyber Witcher. Інструменти для тестування мережевого трафіку та безпеки [Електронний ресурс]. – 2024. – Режим доступу: https://hackyourmom.com/kibervijna/instrumenty-dlya-testuvannya-merezhevogo-trafiku-ta-bezpeky

10. Lanmarket. Мережева аналітика як життєво важлива технологія забезпечення керованості та безпеки сучасної мережі [Електронний ресурс]. – 2019. – Режим доступу: https://lanmarket.ua/ua/stats/setevaya-analitika-kak-zhiznenno-vazhnaya-tekhnologiya-obespecheniya-upravlyaemosti-i-bezopasnosti-s

11. Мальченко Г. Розробка системи моніторингу мережевого трафіку : кваліфікац. робота на здобуття освіт. ступеня бакалавра зі спец. 123 «Комп'ютерна інженерія» [Електронний ресурс] / Г. Мальченко. – Київ, 2024. – Режим доступу: https://duikt.edu.ua/repozitorii/Кафедра%20Комп%27ютерної%20інженерії/2024/бак/Мальченко%20Гліб.pdf

12. CGS Tower Networks. Network Packet Broker: що це і для чого, особливості роботи та функціонал [Електронний ресурс]. – Режим доступу: https://cgstower.bakotech.com/ua/network-packet-broker

13. FDCServers. Як аналізувати транзитний IP-трафік за допомогою NetFlow [Електронний ресурс]. – 2025. – Режим доступу: https://fdcservers.net/uk/blog/how-to-analyze-ip-transit-traffic-with-netflow#розуміння-netlow

14. Zosym Maxym. Проблема візантійських генералів (The Byzantine Generals Problem) [Електронний ресурс]. – 2022. – Режим доступу: https://www.maxzosim.com/the-byzantine-generals-problem

15. Лекція 19. Якість обслуговування QoS [Електронний ресурс]. Технічний фаховий коледж ЛНТУ. – Режим доступу: https://e-tk.lntu.edu.ua/mod/page/view.php?id=3572

16. Славко О. Г. Інформаційна технологія керування перевантаженнями в мультисервісних телекомунікаційних мережах // Вісник Кременчуцького нац. ун-ту ім. М. Остроградського. – Кременчук : КрНУ, 2011. – Вип. 2 (67), част. 1. – С. 29–34.

17. Сохін Н. Л., Гученко М. І., Кирса А. О. Моделі та методи прогнозування мережевого трафіку в реальному часі // Вісник КрНУ ім. М. Остроградського. – 2019. – Вип. 4 (117). – С. 90–98.

18. Гайдур Г. І., Гахов С. О., Бригинець А. А. Виявлення мережевих аномалій з використанням алгоритмів нейронних мереж // Телекомунікаційні та інформаційні технології. – 2023. – № 1. – С. 61–73. – DOI: 10.31673/2412-4338.2023.016173.

19. Міщенко М. Функціональна модель системи виявлення та прогнозування кіберзагроз для корпоративних комп’ютерних мереж з використанням експертних оцінок // Технічні науки та технології. – 2024. – № 3 (37). – С. 143–152. – DOI: 10.25140/2411-5363-2024-3(37)-143-152.

20. Моделювання загроз. Що таке моделювання загроз і якими є його переваги? [Електронний ресурс]. – 2024, 23 лют. – Режим доступу: https://www.issp.training/post/shcho-take-modelyuvannya-zahroz-i-yakymy-ye-yoho-perevahy

21. Захарченко С. М., Трояновська Т. І., Бойко О. В. Основи побудови захищених мереж на базі обладнання компанії Cisco : навч. посібник / С. М. Захарченко, Т. І. Трояновська, О. В. Бойко. – Вінниця : ВНТУ, 2017. – 136 с.

22. Keenetic. Сегменти мережі [Електронний ресурс]. – Режим доступу: https://help.keenetic.com/hc/uk/articles/360005236300-Сегменти-мережі

23. Вишняков В. М. Принципи побудови комп'ютерних мереж : навч. посібник : для студ. галузі знань 12 "Інформ. технології" / В. М. Вишняков ; Київ. нац. ун-т буд-ва і архіт. – Київ : КНУБА, 2022. – 123 с.

24. Coinmarketcap. Задача візантійських генералів [Електронний ресурс]. – Режим доступу: https://coinmarketcap.com/academy/uk/glossary/byzantine-generals-problem

25. Самойлов І. В., Толстих В. А. Модель діагностики комп'ютерних мереж на базі нечітких відношень : X Науково-практ. конф. «Пріоритетні напрями розвитку телекомунікаційних систем та мереж спец. призначення», 9–10 листоп. 2017 р. – С. 216–217.

26. Бурячок В. Л., Толубко В. Б. Основи кібербезпеки : монографія / В. Л. Бурячок, В. Б. Толубко. – Київ : НАУ, 2018. – 320 с.

27. Корченко О. Г., Бондарчук А. П. Аналіз мережевих загроз та методи їх виявлення // Захист інформації. – 2020. – Т. 22, № 3. – С. 189–197.

28. Дудикевич В. Б., Лип'янін В. О. Інтелектуальні методи аналізу мережевого трафіку // Вісник НУ «Львівська політехніка». – 2019. – № 5. – С. 101–109

29. Сидоренко В. К., Павленко М. А. Системи виявлення вторгнень у комп'ютерних мережах / В. К. Сидоренко, М. А. Павленко. – Харків : ХНУРЕ, 2020. – 156 с.

30. Cisco Systems. Cisco NetFlow Configuration Guide [Електронний ресурс]. – 2023. – Режим доступу: https://www.cisco.com

31. Cisco Press. Network Performance and Optimization Guide. – Indianapolis : Cisco Press, 2021. – 464 p.

32. Comer D. Computer Networks and Internets. – 6th ed. – Pearson, 2019. – 720 p.

33. Tanenbaum A. S., Wetherall D. J. Computer Networks. – 5th ed. – Pearson, 2011. – 960 p.

34. Stallings W. Foundations of Modern Networking. – Pearson, 2020. – 624 p.

35. Sanders C., Smith J. Applied Network Security Monitoring. – Elsevier, 2014. – 288 p.

36. Scarfone K., Mell P. Guide to Intrusion Detection and Prevention Systems (IDPS). – NIST SP 800-94, 2012.

37. Bejtlich R. The Practice of Network Security Monitoring. – No Starch Press, 2013. – 480 p.

38. Lakhina A., Crovella M., Diot C. Diagnosing Network-Wide Traffic Anomalies // IEEE/ACM Transactions on Networking. – 2005. – Vol. 13, No. 3. – P. 518–530.

39. Kim H., Claffy K. Internet Traffic Classification Demystified // ACM SIGCOMM Computer Communication Review. – 2019.

40. Barford P., Kline J., Plonka D. A Signal Analysis of Network Traffic Anomalies // ACM IMW. – 2002.

41. ISO/IEC 7498-1: Open Systems Interconnection. OSI Reference Model. – ISO, 2018.

42. Wireshark Foundation. Wireshark User’s Guide [Електронний ресурс]. – 2024. – Режим доступу: https://www.wireshark.org/docs/

43. Elastic. Network Monitoring with Elastic Stack [Електронний ресурс]. – 2024. – Режим доступу: https://www.elastic.co

44. IEEE. Network Traffic Analysis for Cybersecurity Applications // IEEE Communications Surveys & Tutorials. – 2022.

ДОДАТОК А

Слайди презентації

Рис. А.1 Перший слайд

Актуальність проблеми: Зростання складності та ризиків. Традиційні методи діагностики часто нездатні оперативно реагувати на складні, багатофакторні загрози та аномалії в умовах розподілених систем.

Мета дослідження та основні завдання:

- **Розробка підходу.** Комплексна діагностика на основі аналізу мережевого трафіку.
- **Виявлення аномалій.** Своєчасна ідентифікація затримок, відмов та потенційних загроз.
- **Підвищення надійності.** Забезпечення стабільності та безпеки інфраструктури.

Наукова новизна полягає в інтеграції кількох передових концепцій у єдину діагностичну технологію.

Предмет дослідження - методи та засоби діагностики мережевих аномалій на основі аналізу трафіку.

Об'єкт дослідження - комп'ютерна мережа з багаторівневою системою безпеки.

Рис. А.2 Другий слайд

Розроблено схему, що ілюструє ключові аспекти важливості комп'ютерних мереж, класифіковані за їхніми функціями та сферами застосування, такими як зв'язок, спільне використання ресурсів, освіта, комунікації, обмін даними, бізнес та розваги.

Цей слайд підкреслює, наскільки комп'ютерні мережі є критично важливими для сучасного суспільства. Він візуалізує їхній вплив на різноманітні сфери нашого повсякденного життя та професійної діяльності.

Рис. А.3 Третій слайд

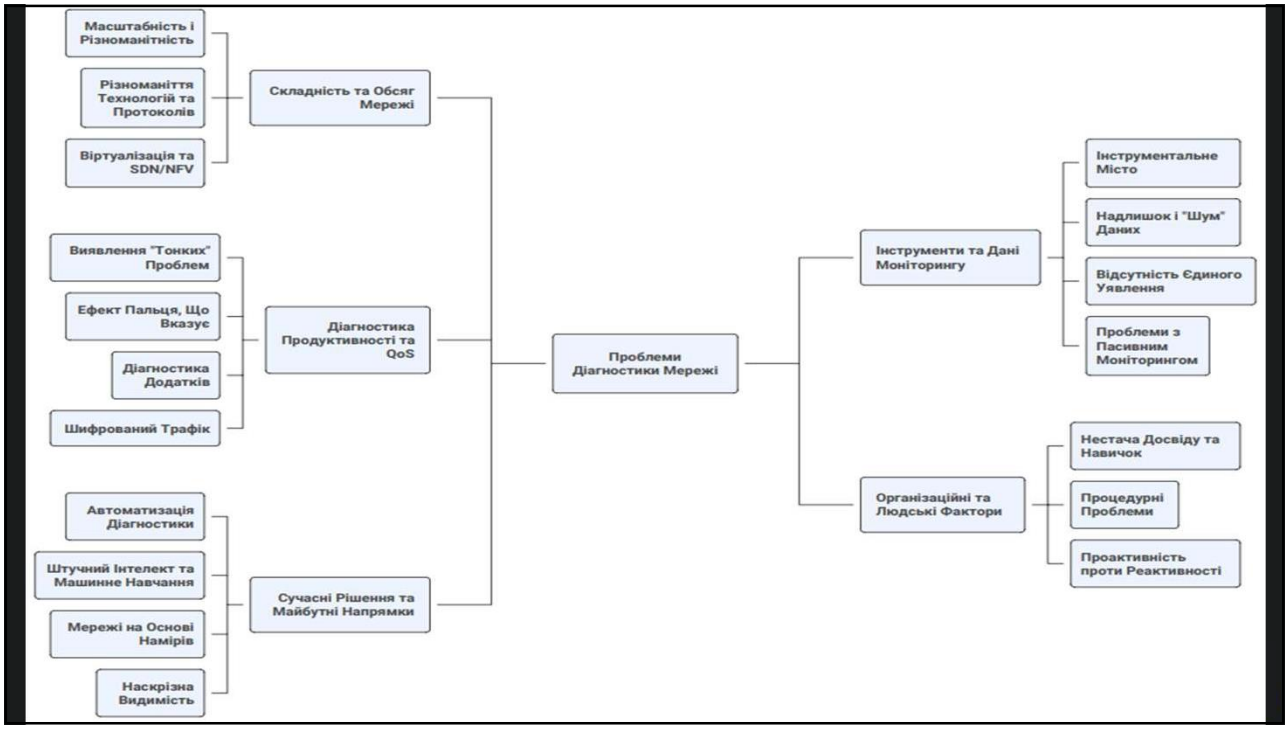
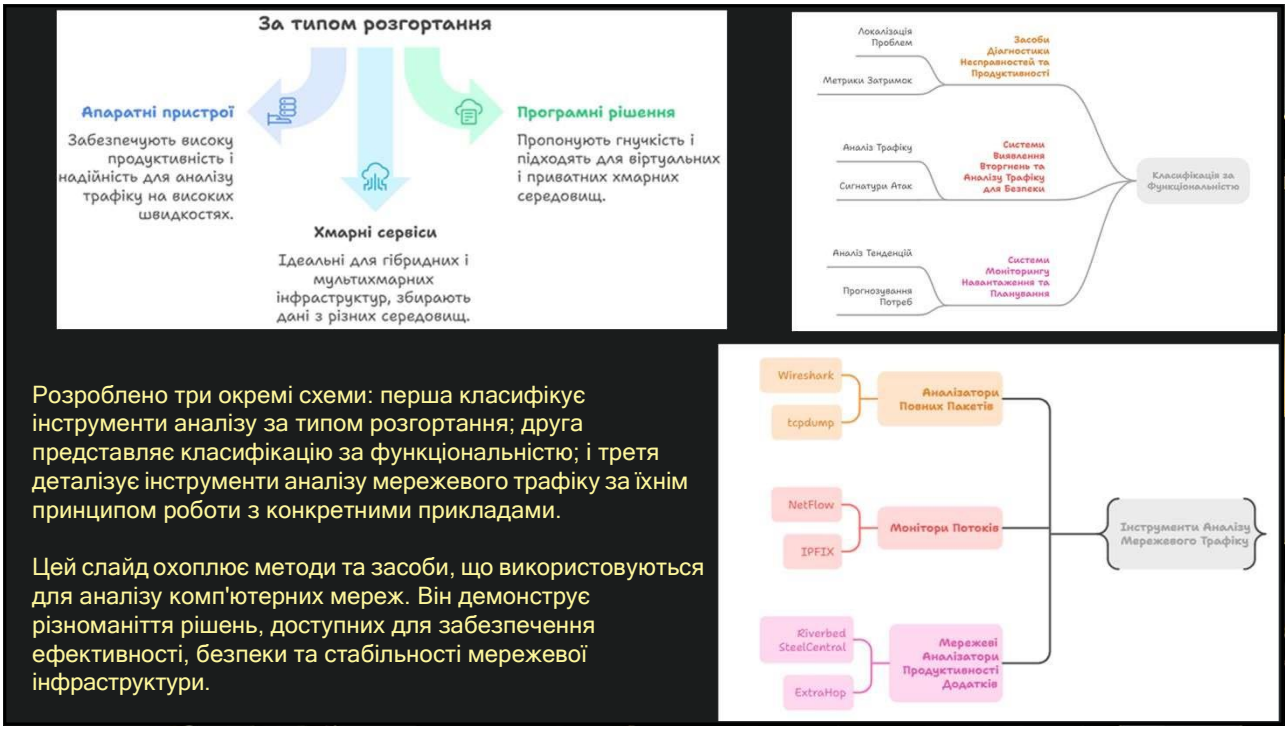


Рис. А.4 Четвертий слайд

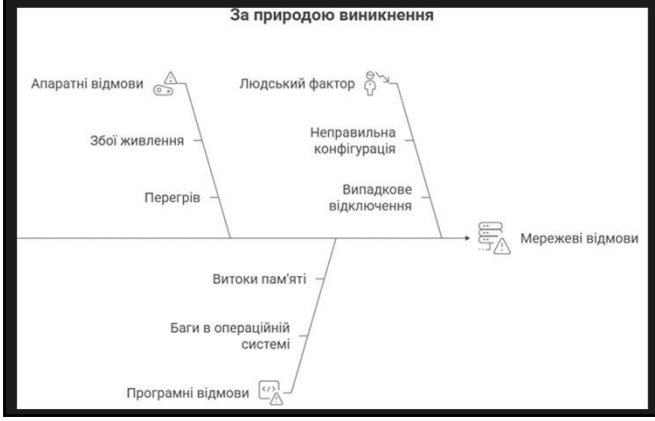


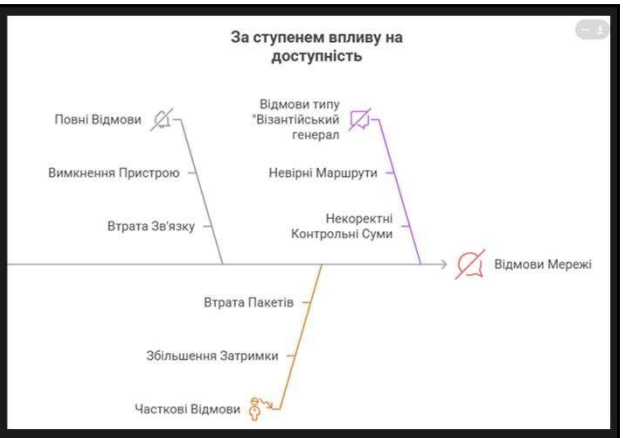
Розроблено три окремі схеми: перша класифікує інструменти аналізу за типом розгортання; друга представляє класифікацію за функціональністю; і третя деталізує інструменти аналізу мережевого трафіку за їхнім принципом роботи з конкретними прикладами.

Цей слайд охоплює методи та засоби, що використовуються для аналізу комп'ютерних мереж. Він демонструє різноманіття рішень, доступних для забезпечення ефективності, безпеки та стабільності мережевої інфраструктури.

Рис. А.5 П’ятий слайд

Розроблено дві окремі схеми:
перша демонструє причини мережевих відмов за природою їхнього виникнення (апаратні, програмні, людський фактор), а друга класифікує відмови мережі за ступенем їхнього впливу на доступність (повні, часткові, а також специфічні відмови типу "Візантійський генерал").





Цей слайд аналізує різні типи та джерела мережевих відмов. Він допомагає краще зрозуміти складність і різноманітність проблем, що можуть виникнути в комп'ютерних мережах.

Рис. А.6 Шостий слайд

Розроблено таблицю, що класифікує та описує чотири основні типи відмов у розподілених системах: відмова за збоєм, відмова за пропуском, відмова за часом та візантійська відмова. Для кожного типу надано його характерні особливості та відповідну модель поведінки.

В роботі було розглянуто візантійські збої, які моделюють ситуації ненадійної або зловмисної поведінки вузлів у розподілених системах.

Тип відмови	Характеристика	Модель поведінки
Crash fault	Вузол перестає відповідати (вимкнувся або завис)	Втрата повідомлень
Omission fault	Деякі повідомлення губляться або не доходять	Часткова втрата даних
Timing fault	Повідомлення запізнюються або приходять із затримкою	Асинхронність
Byzantine fault	Вузол діє довільно: підробляє, бреше, надсилає різним вузлам різні дані	Повна недовіра

Рис. А.7 Сьомий слайд

Розроблено ієрархічну схему "Моделі мережевого трафіку", яка класифікує підходи до моделювання на дві основні категорії: "Класифікація та базові моделі" (включаючи випадкові процеси, Пуассонівський процес, Марківські моделі, а також моделювання взаємодії та обслуговування через теорію черг) та "Моделі прогнозування та оцінки продуктивності" (охоплюючи моделі самоподібного трафіку, авторегресійні моделі, імітаційне моделювання та діагностичні метрики).

Рис. А.8 Восьмий слайд

Розроблено таблицю, яка порівнює Пуассонівський та Марківський процеси за типом, основними властивостями та використанням. В рамках дипломної роботи було детально розглянуто Пуассонівський процес, включаючи його опис як послідовності незалежних випадкових подій з експоненційним розподілом міжприхідного часу, а також формули для обчислення кількості подій, середнього значення та дисперсії, з акцентом на його застосування для моделювання простих потоків запитів або пакетів у мережах. Також було розглянуто Марківський процес як станову модель, де майбутній стан залежить виключно від поточного, з поясненням його властивості відсутності післядії та застосування у моделюванні черг, навантаження та зміни режимів роботи каналів.

Модель	Тип процесу	Основна властивість	Використання
Пуассонівський	Потік подій	Незалежні прибуття (експоненційний розподіл)	Моделювання простих потоків запитів або пакетів
Марксівський	Станова модель	Наступний стан залежить тільки від поточного	Моделювання черг, навантаження, зміни режимів роботи кана

Рис. А.9 Дев’ятий слайд

Розроблено схему, що класифікує та описує чотири основні групи методів для аналізу трафіку та виявлення аномалій: методи машинного навчання, статистичні методи, методи на основі вейвлет-аналізу та гібридні методи.

У дипломній роботі було досліджено статистичні методи, а саме Z-оцінку, яка дозволяє ідентифікувати аномалії, вимірюючи відхилення спостережень від середнього значення в стандартних відхиленнях, з визначеними порогами для нормальної, підозрілої та аномальної поведінки трафіку. Також було досліджено вейвлет-аналіз, який дозволяє розкладати мережевий трафік на різні частотні та часові складові для виявлення раптових змін та аномалій на різних масштабах, використовуючи дискретне вейвлет-перетворення для відокремлення фонових змін від високочастотних аномалій.

Рис. А.10 Десятий слайд

Цей слайд відображає життєвий цикл аналізу мережевого трафіку, від його початкового збору до вилучення значущих характеристик.

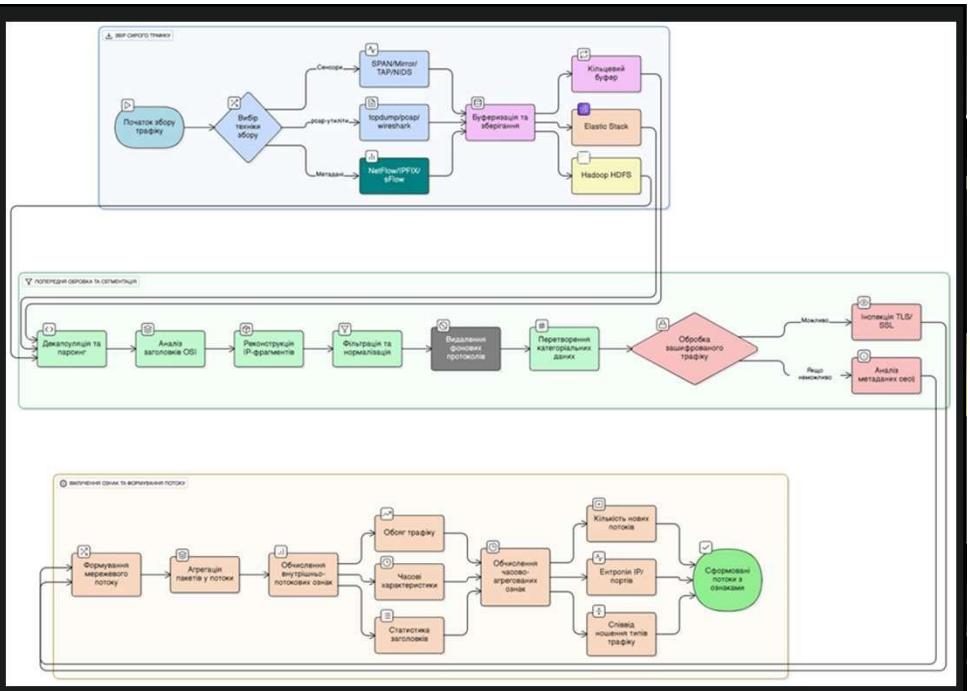


Рис. А.11 Одинадцятий слайд

Розроблено схему топології тестового полігону, яка візуалізує розміщення сенсорів для захоплення мережевого трафіку. Схема включає основні сегменти: Internet/Емуляція WAN з граничним маршрутизатором, LAN-сегмент з робочими станціями та серверами автентифікації, а також демілітаризовану зону з веб-серверами та поштовими серверами.

На схемі позначено точки встановлення різних типів сенсорів, таких як мережевий TAP, SPAN/Mirror Port, сенсор IDS/PS та Centos 8 для збору сирих пакетів, що дозволяє проводити всебічний моніторинг та аналіз трафіку всередині та між сегментами мережі.

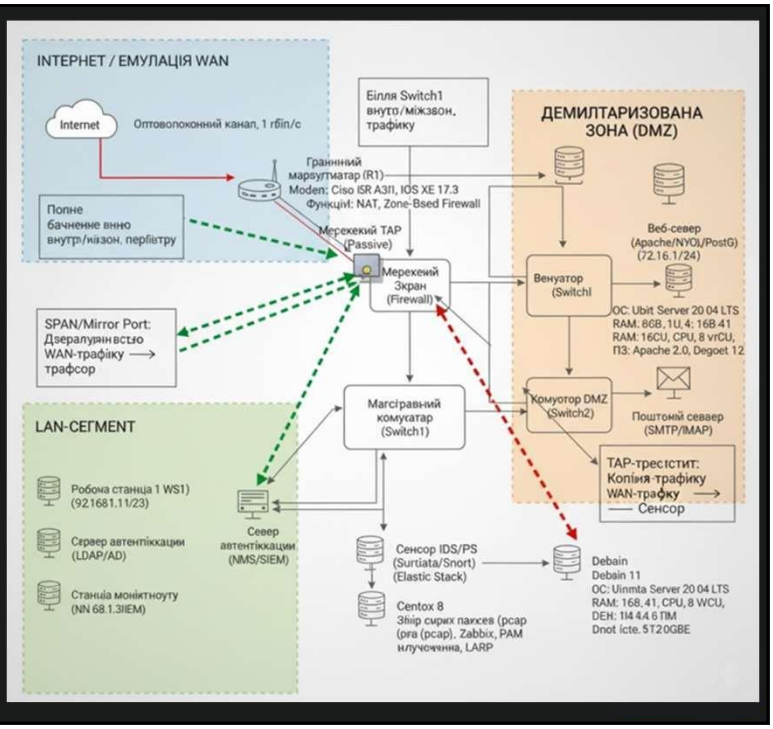


Рис. А.12 Дванадцятий слайд

Клас атаки	Фаза атаки	Мета моделювання
Сканування портів (SYN Scan)	Фаза 1: Розвідка	Перевірка чутливості NADS до зміни розподілу портів та порушення TCP-рукостискання (Half-open connections).
DDoS-атака (UDP Flood)	Фаза 2: Використання	Перевірка реакції NADS на різкий сплеск об'єму трафіку та високу ентропію джерел (імітація ботнету).

На цьому слайді представлено дві таблиці, що деталізують мережеві атаки та їхнє моделювання.

Перша таблиця "Клас атаки" описує два типи атак - сканування портів (SYN Scan) та DDoS-атаку (UDP Flood), розкриваючи фази цих атак та мету їхнього моделювання для перевірки систем виявлення аномалій (NADS). Друга таблиця надає докладні параметри, інструменти та деталі для відтворення атаки SYN Scan (Half-open), включаючи IP-адреси, інтенсивність, тривалість та ознаки аномальної поведінки.

Параметр	Техніка / Інструмент	Деталізація для відтворюваності
Інструмент	Nmap або hping3 (у режимі SYN).	Команда Nmap: nmap -sS -p 1-65535 -T2 <IP_Жертви>
Жертва	Веб-сервер / фаєрвол.	IP-адреса жертви: 192.168.1.10.
Джерело	Єдиний зовнішній вузол.	IP-адреса джерела: 10.0.0.5.
Тип сканування	SYN Scan (Half-open).	Використовується прапор SYN, але без завершення TCP-сесії.
Інтенсивність	Низька	5 пакетів/секунду (опція -r у Nmap або -rate 5).
Тривалість		300 секунд (5 хвилин).
Ознаки аномалії	Збільшення SYN-пакетів без відповідного ACK; високе співвідношення SYN/RST для сканованих портів.	

Рис. А.13 Тринадцятий слайд

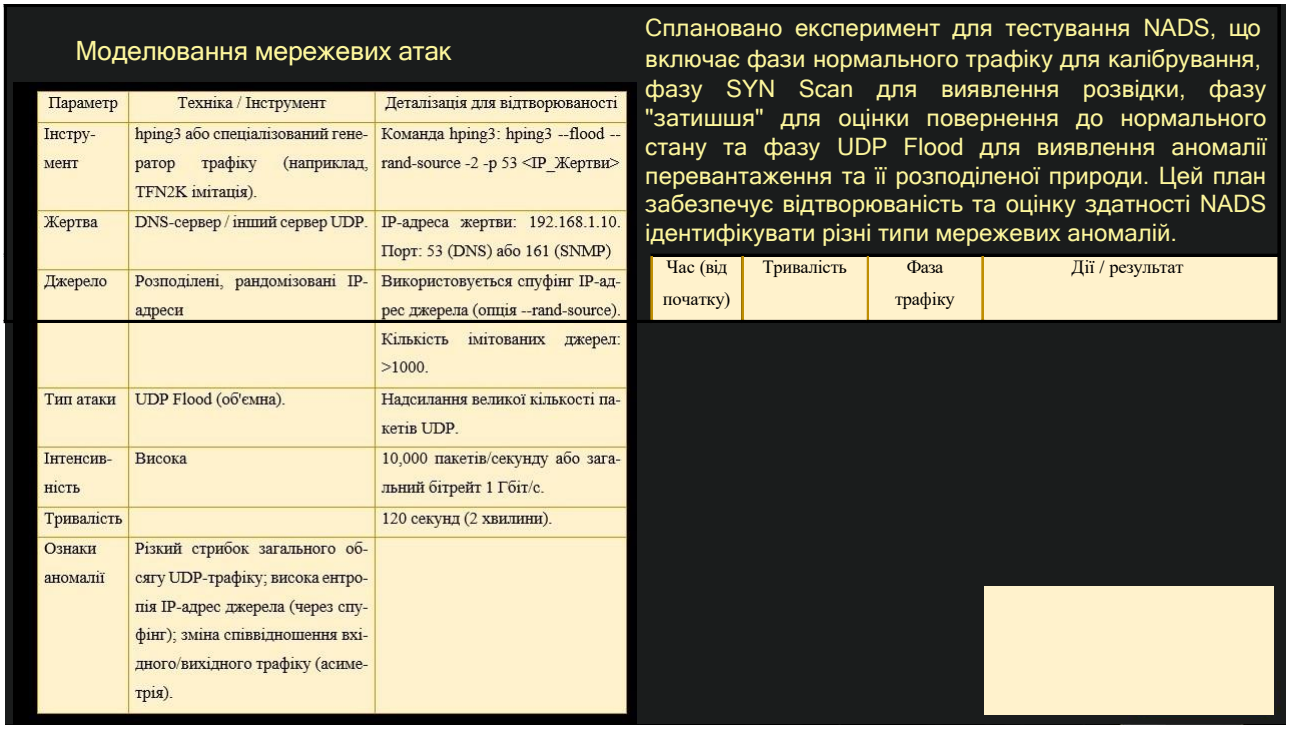


Рис. А.14 Чотирнадцятий слайд

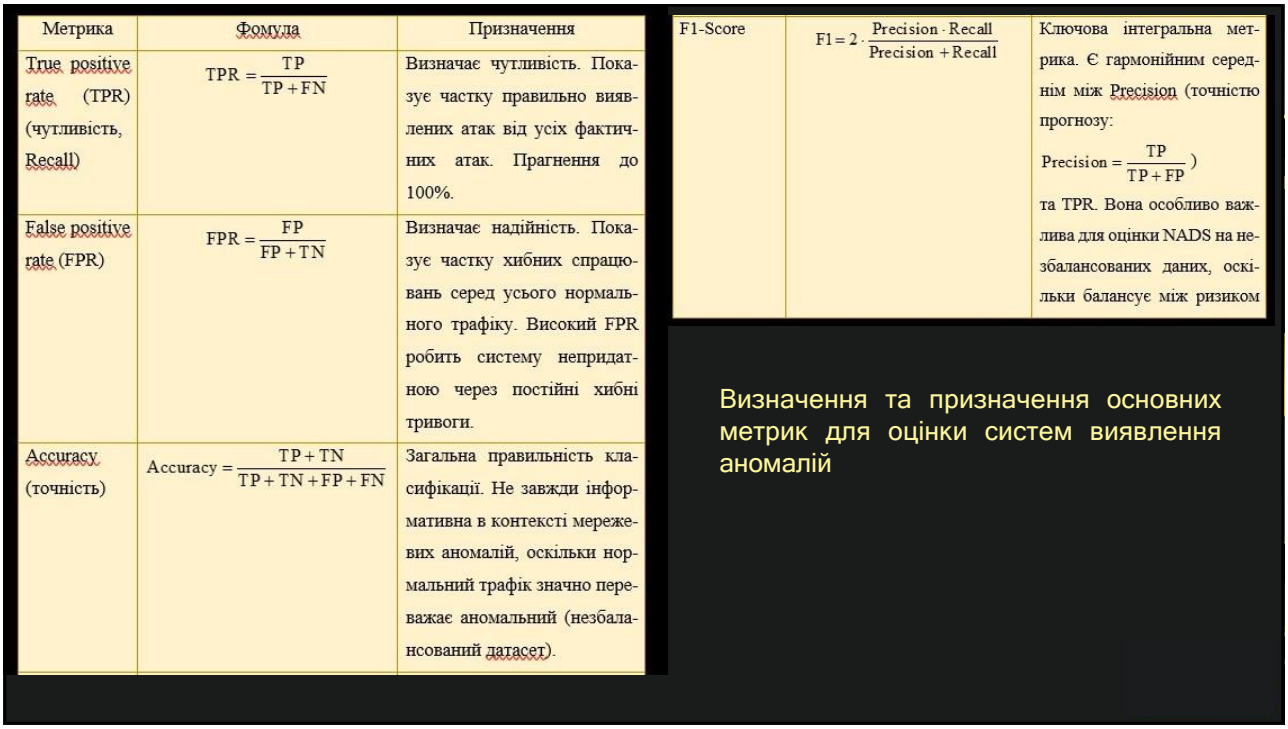


Рис. А.15 П’ятнадцятий слайд

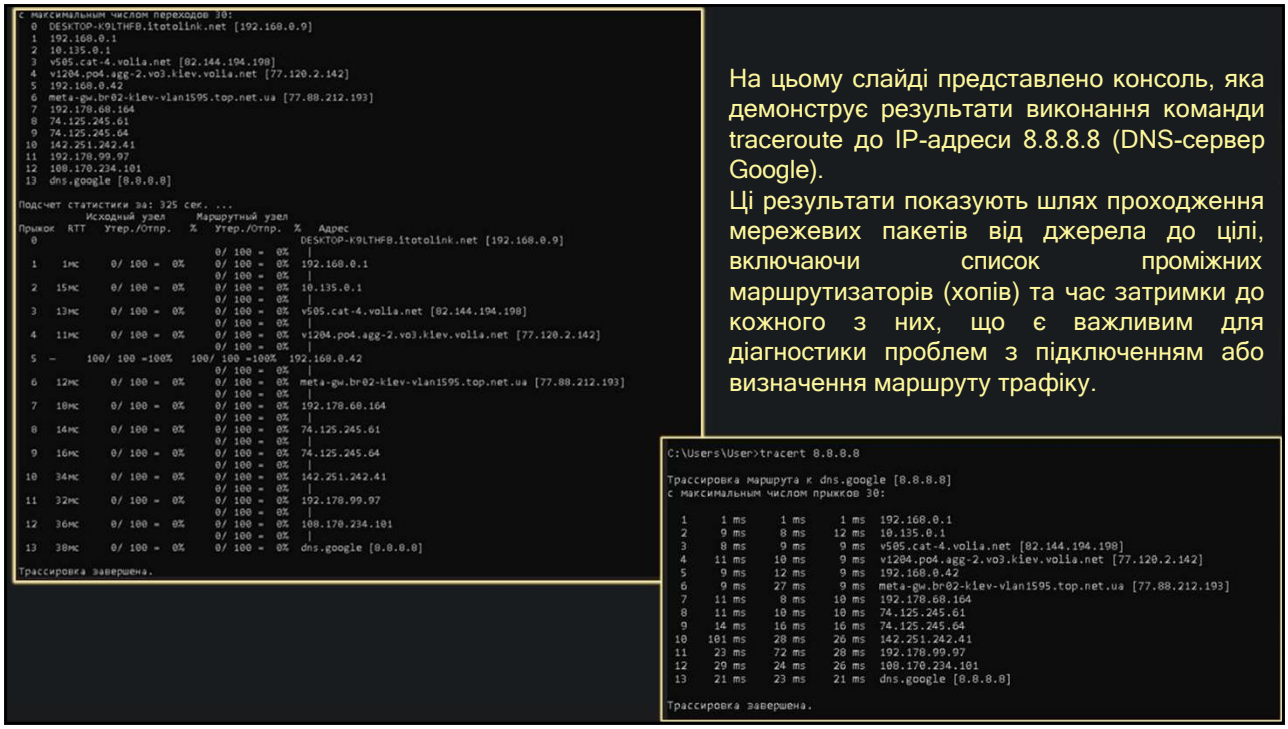


Рис. А.16 Шістнадцятий слайд

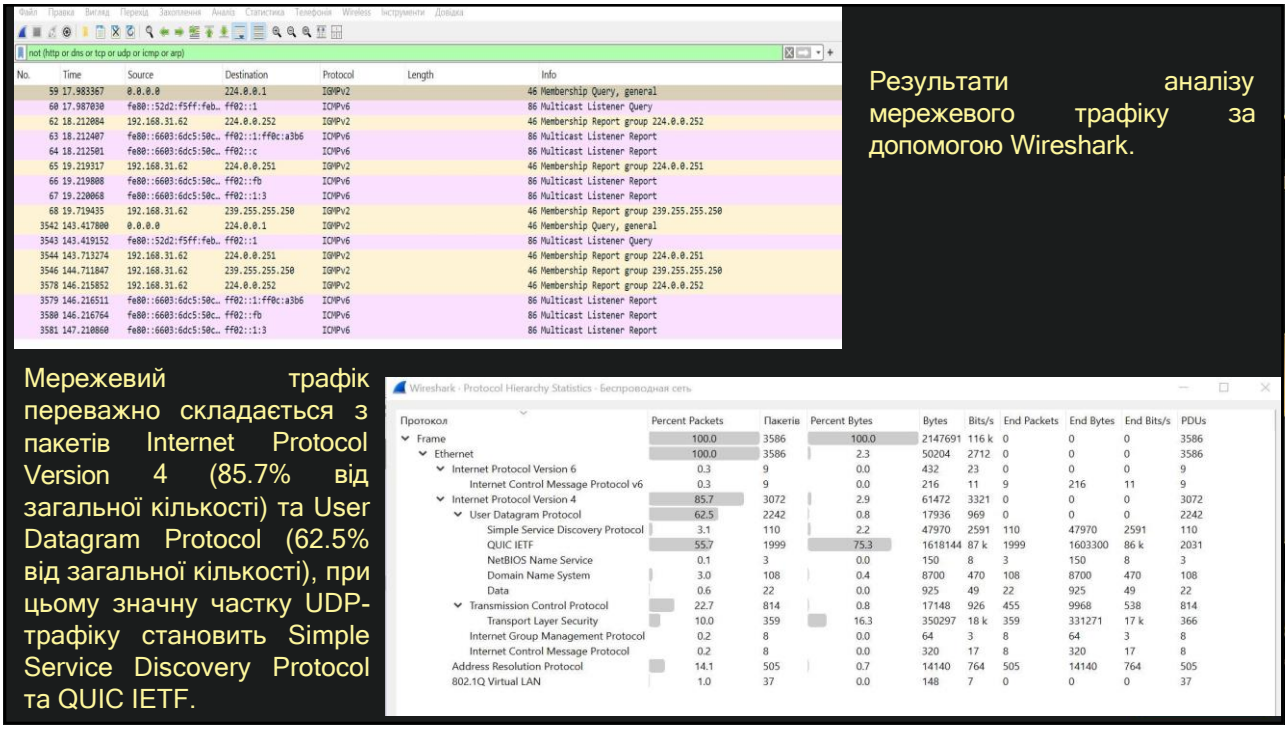


Рис. А.17 Сімнадцятий слайд

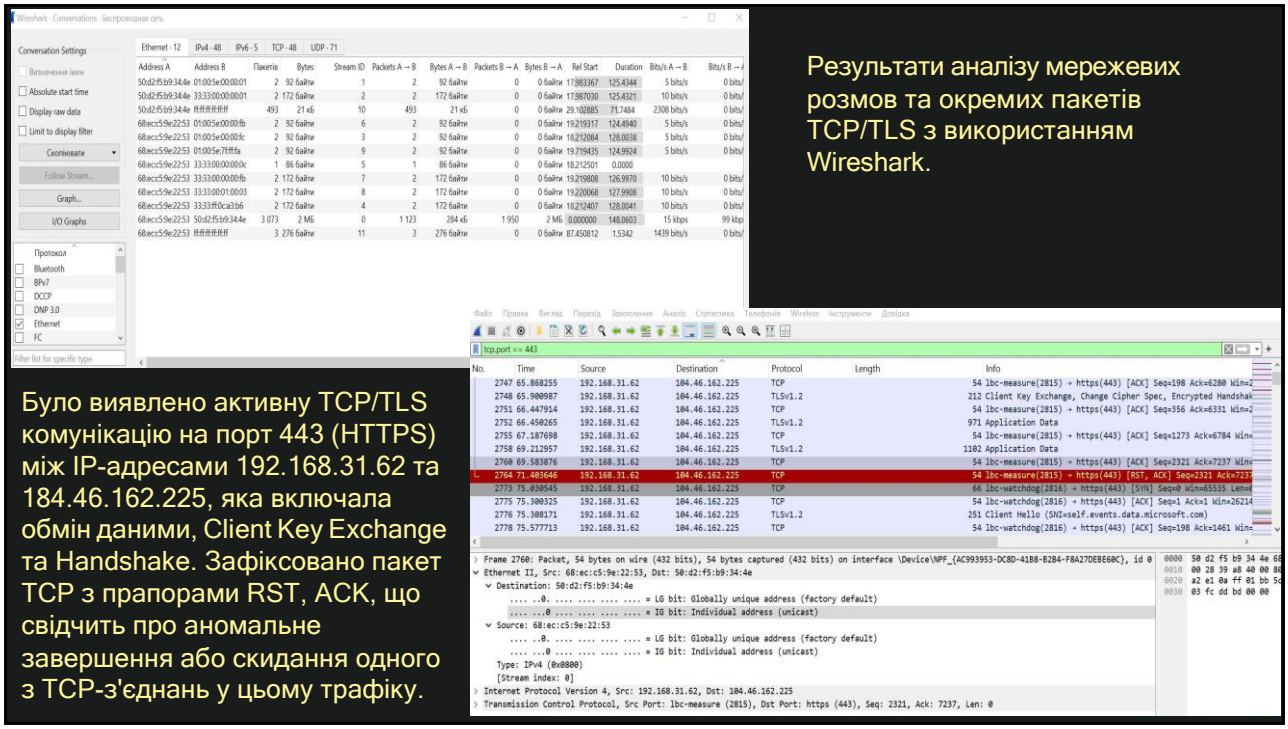
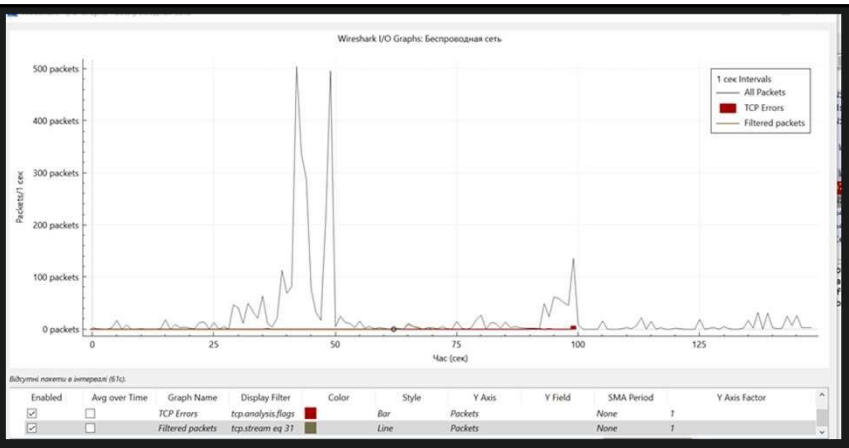


Рис. А.18 Вісімнадцятий слайд

Графік демонструє періодичні сплески мережевого трафіку, зокрема навколо 45-ї та 95-ї секунд. Аналіз в таблиці вказує, що джерелом затримок є зовнішній маршрутизатор (хоп 10, IP-адреса 142.251.242.41), що інтерпретується як пристрій провайдера або магістралі. Проблема затримок характеризується як тимчасові сплески, а не системна. Виявлено TCP-помилки під час сплесків пакетів, що потребує налаштування QoS або додаткового моніторингу. Аномалій протоколів не виявлено, що свідчить про "чисту мережу" з точки зору протокольних порушень. Ефективність NADS, виміряна як затримка менше 5 секунд і показник TPR близько 100%, оцінюється як висока.



Параметр	Значення	Висновок
Джерело затримки	Хоп 10 (142.251.242.41)	Зовнішній маршрутизатор (провайдер/магістраль
Характер проблеми	Тимчасові сплески	Не системна
TCP-помилки	Є, під час піків	Потребує QoS або моніторингу
Аномалії протоколів	Немає	Мережа чиста
Ефективність NADS	Tlatency < 5 с, TPR ≈ 100%	Висока

Рис. А.19 Дев’ятнадцятий слайд

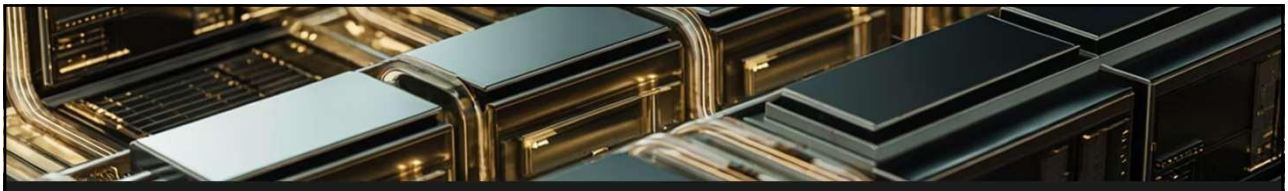
АПРОБАЦІЯ

Результати дослідження апробовано шляхом публікації тез доповідей:

Житомирська політехніка - VIII Всеукраїнська науково-технічна конференція (02.12.2025 - 03.12.2025)

1. Сарапин В.Є., Шабала Є.Є. ГІБРИДНИЙ ПІДХІД ДЛЯ ДІАГНОСТИКИ МЕРЕЖЕВИХ АНОМАЛІЙ ЧЕРЕЗ ПАРАМЕТР ХЕРСТА ТА QOS-МЕТРИКИ
2. САРАПИН В.Є. ВИЯВЛЕННЯ МЕРЕЖЕВИХ АНОМАЛІЙ ЗАСОБАМИ АНАЛІЗУ ТРАФІКУ.

Рис. А.20 Двадцятий слайд



Висновки та результати

Проведено комплексний аналіз архітектури мереж, класифікацію відмов (включаючи візантійські збої) та дослідження QoS. Це сформувало теоретичну базу для розробки ефективної системи діагностики.

Аналіз архітектури

Визначено вплив фізичної та логічної структури на процеси аналізу трафіку.

Експериментальна перевірка

Створено тестовий полігон та змодельовано двофазну атаку (сканування + DDoS).

Класифікація відмов

Розглянуто апаратні, програмні, людські та візантійські чинники збоїв.

Підтвердження ефективності

Практична реалізованість діагностики та її здатність до виявлення складних аномалій підтверджена метриками.

Рис. А.21 Двадцять перший слайд