Міністерство освіти і науки України

Харківський національний університет радіоелектроніки

Факультет навчально-науковий центр заочної форми навчання
(повна назва)

Кафедра електронних обчислювальних машин
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА

Пояснювальна записка

Рівень вищої освіти перший (бакалаврський)

Програмні засоби моніторингу мережного трафіка

(тема)

Виконав:
здобувач 4 року навчання,
групи КІУКІз-21-1
Олександр КІЛЮШИК
(власне ім'я, прізвище)

Спеціальність
123 «Комп'ютерна інженерія»
(код і повна назва спеціальності)

Тип програми освітньо-професійна
(освітньо-професійна або освітньо-наукова)

Освітня програма
Комп'ютерна інженерія
(повна назва освітньої програми)

Керівник: ас. Дмитро ДЯЧЕНКО
(посада, власне ім'я, прізвище)

Допускається до захисту

Завідувач кафедри ЕОМ _____ Андрій КОВАЛЕНКО
(підпис) (власне ім'я, прізвище)

2025 р.

Харківський національний університет радіоелектроніки

Факультет навчально-науковий центр заочної форми навчання

Кафедра електронних обчислювальних машин

Рівень вищої освіти перший (бакалаврський)

Спеціальність 123 «Комп'ютерна інженерія»
(код і повна назва)

Тип програми освітньо-професійна
(освітньо-професійна або освітньо-наукова)

Освітня програма Комп'ютерна інженерія
(повна назва)

ЗАТВЕРДЖУЮ:
Зав. кафедри _____
(підпис)
"____" _____ 20__ р.

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ

здобувачеві Кілюшику Олександру Олександровичу
(прізвище, ім'я, по батькові)

1. Тема роботи Програмні засоби моніторингу мережного трафіка

затверджена наказом по університету від " 05 " травня 2025 р. № 72 СТз

2. Термін подання здобувачем роботи до екзаменаційної комісії 16 червня 2025 р.

3. Вхідні дані до роботи
трафік
моніторинг
програмні засоби
PyQt5

4. Перелік питань, що потрібно опрацювати у роботі
Теоретичні основи моніторингу мережного трафіка
Огляд сучасних програмних засобів моніторингу мережного трафіка
Порівняльний аналіз програмних засобів моніторингу
Розробка програмних засобів моніторингу

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій 11 слайдів

6. Консультанти розділів роботи (заповнюється за наявності консультантів згідно з наказом, зазначеним у п.1)

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Строк / терміни виконання етапів роботи	Примітка
1	Отримання завдання та аналіз літератури	05.05.2025–07.05.2025	
2	Огляд існуючих аналогів	08.05.2025–13.05.2025	
3	Вибір алгоритмів	13.05.2025–17.05.2025	
4	Вибір програмних засобів	18.05.2025–23.05.2025	
5	Програмна реалізація	24.05.2025–26.05.2025	
6	Аналіз отриманих результатів	27.05.2025–28.05.2025	
7	Оформлення записки	28.05.2025–01.06.2025	

Дата видачі завдання " 05 " травня 2025 р.

Здобувач _____
(підпис)

Керівник роботи _____ ас. Дмитро ДЯЧЕНКО
(підпис) (посада, власне ім'я, прізвище)

РЕФЕРАТ

Пояснювальна записка кваліфікаційної роботи: 59 с., 8 рис., 1 дод., 10 джерел.

SNMP, МЕРЕЖЕВИЙ МОНІТОРИНГ, АНАЛІЗ ПАКЕТІВ, АРХІТЕКТУРА МЕРЕЖІ, МЕТРИКИ ПРОДУКТИВНОСТІ, TTL, ПРОПУСКНА ЗДАТНІСТЬ, ВТРАТИ ПАКЕТІВ, ЛАТЕНТНІСТЬ, ДЖИТТЕР, МІВ, МЕРЕЖЕВЕ АДМІНІСТРУВАННЯ, ВІЗУАЛІЗАЦІЯ ДАНИХ, РЕАЛЬНО-ЧАСОВИЙ МОНІТОРИНГ, ETHERNET, GET-ЗАПИТИ, TRAP-ПОВІДОМЛЕННЯ, ВЕБ-ІНТЕРФЕЙС, ДІАГНОСТИКА МЕРЕЖІ, МЕРЕЖЕВИЙ ТРАФІК.

Метою кваліфікаційної роботи розробка програмних засобів моніторингу трафіку корпоративної мережі з використанням PyQt5.

У ході виконання кваліфікаційної роботи проаналізовано архітектуру SNMP, яка відображає класичну тришарову модель взаємодії між менеджером, агентами та базами даних, при цьому особливу увагу приділено візуалізації різних типів комунікації між компонентами системи. Практична реалізація інтерфейсу мережевого аналізатора демонструє успішне поєднання функціональності професійних інструментів моніторингу з сучасними веб-технологіями, забезпечуючи реально-часову візуалізацію мережевого трафіку через інтерактивні графіки TTL, динамічне оновлення метрик пропускної здатності, детальне логування пакетів з можливістю аналізу Ethernet-фреймів та HEX-даних, а також зручні елементи керування процесом моніторингу. Створений інтерфейс характеризується високою функціональністю, інтуїтивною зрозумілістю та адаптивністю до різних сценаріїв використання, що робить його придатним як для навчальних цілей, так і для практичного застосування в реальних мережевих середовищах.

ABSTRACT

Bachelor’s thesis: 59 pages, 8 figures, 1 appendices, 10 sources.

SNMP, NETWORK MONITORING, PACKET ANALYSIS, NETWORK ARCHITECTURE, PERFORMANCE METRICS, TTL, BANDWIDTH, PACKET LOSS, LATENCY, JITTER, MIB, NETWORK ADMINISTRATION, DATA VISUALIZATION, REAL-TIME MONITORING, ETHERNET, GET REQUESTS, TRAP MESSAGES, WEB INTERFACE, NETWORK DIAGNOSTICS, NETWORK TRAFFIC.

The major goal of this thesis is the development of software tools for monitoring corporate network traffic using PyQt5.

In order to the SNMP architecture was thoroughly analyzed, reflecting the classical three-tier interaction model involving the manager, agents, and databases. Special emphasis was placed on the visualization of various types of communication between system components.

The practical implementation of the network analyzer interface demonstrates the successful integration of professional monitoring functionality with modern web technologies. It enables real-time visualization of network traffic through interactive TTL-based charts, dynamic updates of bandwidth metrics, detailed packet logging with support for Ethernet frame and HEX data analysis, and user-friendly controls for managing the monitoring process.

The developed interface is distinguished by its high functionality, intuitive design, and adaptability to diverse usage scenarios, making it suitable both for educational purposes and for practical deployment in real-world network environments.

ЗМІСТ

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ 7
1 ТЕОРЕТИЧНІ ОСНОВИ МОНІТОРИНГУ МЕРЕЖНОГО ТРАФІКА 9
1.1 Концептуальні засади мережного моніторингу 9
1.2 Архітектурні підходи до моніторингу мережевої інфраструктури 12
1.3 Методології аналізу мережного трафіка 16
2 ОГЛЯД СУЧАСНИХ ПРОГРАМНИХ ЗАСОБІВ МОНІТОРИНГУ МЕРЕЖНОГО ТРАФІКА 21
2.1 Класифікація програмних рішень для мережного моніторингу 21
2.2 Функціональний аналіз провідних програмних платформ 24
2.3 Технічні характеристики та вимоги до впровадження 28
3 ПОРІВНЯЛЬНИЙ АНАЛІЗ ПРОГРАМНИХ ЗАСОБІВ МОНІТОРИНГУ 32
3.1 Методологія порівняльного аналізу 32
3.2 Аналіз функціональних можливостей 35
3.3 Оцінка ефективності та зручності використання 38
4 РОЗРОБКА ПРОГРАМНИХ ЗАСОБІВ МОНІТОРИНГУ 42
4.1 Структурна схема ПЗ моніторингу трафіку 42
4.2 Реалізація ПЗ 48
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ 52
ДОДАТОК А Графічний матеріал кваліфікаційної роботи 53

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

ПЗ – програмне забезпечення

CoS – клас сервісу

CQ – модель обслуговування, налаштованого користувачем

DNS – служба домених імен

DSCP – поле коду диференційованої послуги

ECN –Explicit Congestion Notification

EIGRP – Enhanced Interior Gateway Routing Protocol

ETSI – Європейський інститут стандартизації телекомунікацій

FIFO – модель перший прийшов - перший на обслуговування

FTP – протокол передачі файлів

HSRP – Hot Standby Router Protocol

HTTP – протокол передачі гіпертексту

ВСТУП

Забезпечення ефективного моніторингу та управління якістю передавання даних у комп’ютерних мережах становить складне завдання, яке потребує ретельного технічного підходу навіть для досвідчених фахівців. Для досягнення коректної роботи подібних систем необхідним є застосування спеціалізованих платформ, які оснащені відповідними інструментами. Сучасний розвиток інформаційних технологій сприяв появі багатофункціональних рішень, що здатні адаптуватися до різноманітних типів мережевих середовищ. Такі системи характеризуються високим рівнем автоматизації: вони самостійно ідентифікують більшість компонентів мережі та надають інформацію про їх функціонування в різних візуальних формах, зручних для аналізу користувачем. У контексті загального розуміння проблематики якості обслуговування мереж, виникає потреба в розробці інструменту, який здійснює моніторинг мережевого трафіку на рівні робочої станції, з метою збору релевантних даних для їх подальшого представлення у придатному для сприйняття вигляді.

Метою кваліфікаційної роботи є розробка програмних засобів моніторингу трафіку корпоративної мережі з використанням PyQt5.

Завдання:

- проаналізувати методи та інструменти для моніторингу та управління якістю обслуговування комп’ютерної мережі;
- провести огляд існуючих комп’ютерних систем моніторингу мережі;
- провести огляд протоколів для моніторингу трафіку в комп’ютерній мережі;
- розробити програмні засоби.

1 ТЕОРЕТИЧНІ ОСНОВИ МОНІТОРИНГУ МЕРЕЖНОГО ТРАФІКА

1.1 Концептуальні засади мережного моніторингу

Концептуальні засади мережного моніторингу формують теоретичну основу для розуміння принципів функціонування сучасних систем контролю мережевого трафіка. Фундаментальним поняттям у цій сфері є власне мережний моніторинг, який визначається як систематичний процес збору, аналізу та інтерпретації даних про стан та функціонування мережевої інфраструктури з метою забезпечення її оптимальної роботи та своєчасного виявлення проблем.

Історична еволюція підходів до мережного моніторингу демонструє поступальний розвиток від примітивних засобів контролю доступності мережевих вузлів до складних інтелектуальних систем комплексного управління мережевою інфраструктурою. Початкові етапи розвитку мережного моніторингу характеризувалися використанням простих утиліт командного рядка та базових протоколів моніторингу, що дозволяли отримувати лише обмежену інформацію про стан мережі. Поступово, з ускладненням мережевих архітектур та зростанням вимог до якості мережевих сервісів, виникла необхідність у створенні більш досконалих засобів моніторингу.

Сучасний етап розвитку мережного моніторингу характеризується інтеграцією передових технологій штучного інтелекту, машинного навчання та великих даних у процеси аналізу мережевого трафіка. Це дозволяє не лише реактивно реагувати на проблеми, але й проактивно прогнозувати потенційні інциденти та оптимізувати мережеву інфраструктуру на основі глибокого аналізу історичних даних та поточних трендів.

Основні цілі мережного моніторингу охоплюють широкий спектр завдань, від забезпечення базової доступності мережевих ресурсів до

складних процесів оптимізації продуктивності та планування розвитку інфраструктури. Первинною метою є забезпечення безперервної доступності мережевих сервісів для кінцевих користувачів, що досягається через постійний контроль стану мережевого обладнання, каналів зв'язку та критично важливих додатків.

Забезпечення якості обслуговування становить другу ключову мету мережного моніторингу, яка передбачає контроль дотримання встановлених параметрів обслуговування користувачів та додатків. Це включає моніторинг показників затримки, пропускної здатності, доступності та надійності мережевих сервісів, а також забезпечення дотримання угод про рівень сервісу.

Виявлення та діагностика проблем продуктивності складає третю важливу мету, що передбачає своєчасне ідентифікування вузьких місць у мережевій інфраструктурі, аналіз причин погіршення продуктивності та надання рекомендацій щодо їх усунення. Сучасні системи моніторингу здатні не лише виявляти існуючі проблеми, але й прогнозувати потенційні інциденти на основі аналізу трендів та аномалій у поведінці мережевого трафіка.

Забезпечення інформаційної безпеки є четвертою критично важливою метою мережного моніторингу, що включає виявлення аномальної активності, потенційних кіберзагроз та порушень політик безпеки. Сучасні системи мережного моніторингу інтегруються з рішеннями інформаційної безпеки для забезпечення комплексного захисту мережевої інфраструктури.

Типологія мережного трафіка представляє складну ієрархічну структуру, що включає різноманітні категорії даних, які передаються через мережеву інфраструктуру. Основною класифікацією є поділ трафіка за протоколами передачі даних, що охоплює трафік рівня додатків, транспортного рівня, мережевого рівня та канального рівня згідно з моделлю OSI.

Трафік рівня додатків характеризується специфічними особливостями різних типів додатків, від простих веб-сторінок до складних мультимедійних потоків та критично важливих бізнес-додатків. Кожен тип додатків має унікальні вимоги до мережевих ресурсів та специфічні патерни поведінки, що необхідно враховувати при налаштуванні моніторингу.

Характеристики мережного трафіка включають широкий спектр параметрів, що визначають якість та ефективність передачі даних. Пропускна здатність представляє максимальний обсяг даних, що може бути переданий через мережевий канал за одиницю часу, та є одним з ключових показників продуктивності мережі. Затримка характеризує час, необхідний для передачі пакета даних від джерела до призначення, та критично впливає на якість інтерактивних додатків.

Втрати пакетів відображають відсоток пакетів, що не досягли призначення через перевантаження мережі, збої обладнання або інші проблеми. Джиттер характеризує варіацію затримки пакетів та особливо важливий для додатків реального часу, таких як голосовий та відеозв'язок.

Особливості поведінки різних протоколів передачі даних визначають специфічні вимоги до моніторингу та аналізу трафіка. Протокол TCP характеризується надійністю передачі даних та механізмами контролю потоку, що створює специфічні патерни поведінки при перевантаженнях мережі. Протокол UDP забезпечує швидку передачу даних без гарантій доставки, що використовується в додатках реального часу та створює інші види навантаження на мережу.

Методи класифікації мережного трафіка включають статистичні підходи, базовані на аналізі характеристик пакетів, та семантичні методи, що враховують зміст та контекст передачі даних. Сучасні системи моніторингу використовують комбінацію різних методів класифікації для досягнення максимальної точності ідентифікації типів трафіка.

1.2 Архітектурні підходи до моніторингу мережевої інфраструктури

Архітектурні підходи до моніторингу мережевої інфраструктури визначають фундаментальні принципи організації та функціонування систем збору, обробки та аналізу мережевих даних. Вибір архітектурного підходу суттєво впливає на ефективність, масштабованість, надійність та вартість системи моніторингу, а також визначає можливості її інтеграції з існуючою інфраструктурою та адаптації до змінних вимог організації.

Централізована архітектура моніторингу представляє традиційний підхід, за якого всі функції збору, обробки та аналізу мережевих даних зосереджені в єдиному центральному вузлі або кластері серверів. Цей підхід характеризується простотою управління та конфігурації, оскільки всі налаштування та політики моніторингу централізовано контролюються з єдиної точки. Централізована архітектура забезпечує консолідований погляд на стан всієї мережевої інфраструктури та спрощує процеси звітності та аналізу.

Однак централізована архітектура має суттєві обмеження щодо масштабованості, особливо у великих розподілених мережах. Центральний вузол може стати вузьким місцем системи при зростанні обсягів мережевого трафіка та кількості моніторованих пристроїв. Крім того, централізована архітектура створює єдину точку відмови, збій якої може призвести до повної втрати можливостей моніторингу.

Розподілена архітектура моніторингу передбачає декомпозицію функцій системи між множиною вузлів, розташованих у різних частинах мережевої інфраструктури. Кожен вузол відповідає за моніторинг певного сегмента мережі та може виконувати локальну обробку та аналіз даних. Результати локального аналізу агрегуються та передаються до центрального вузла для консолідованого аналізу та звітності.

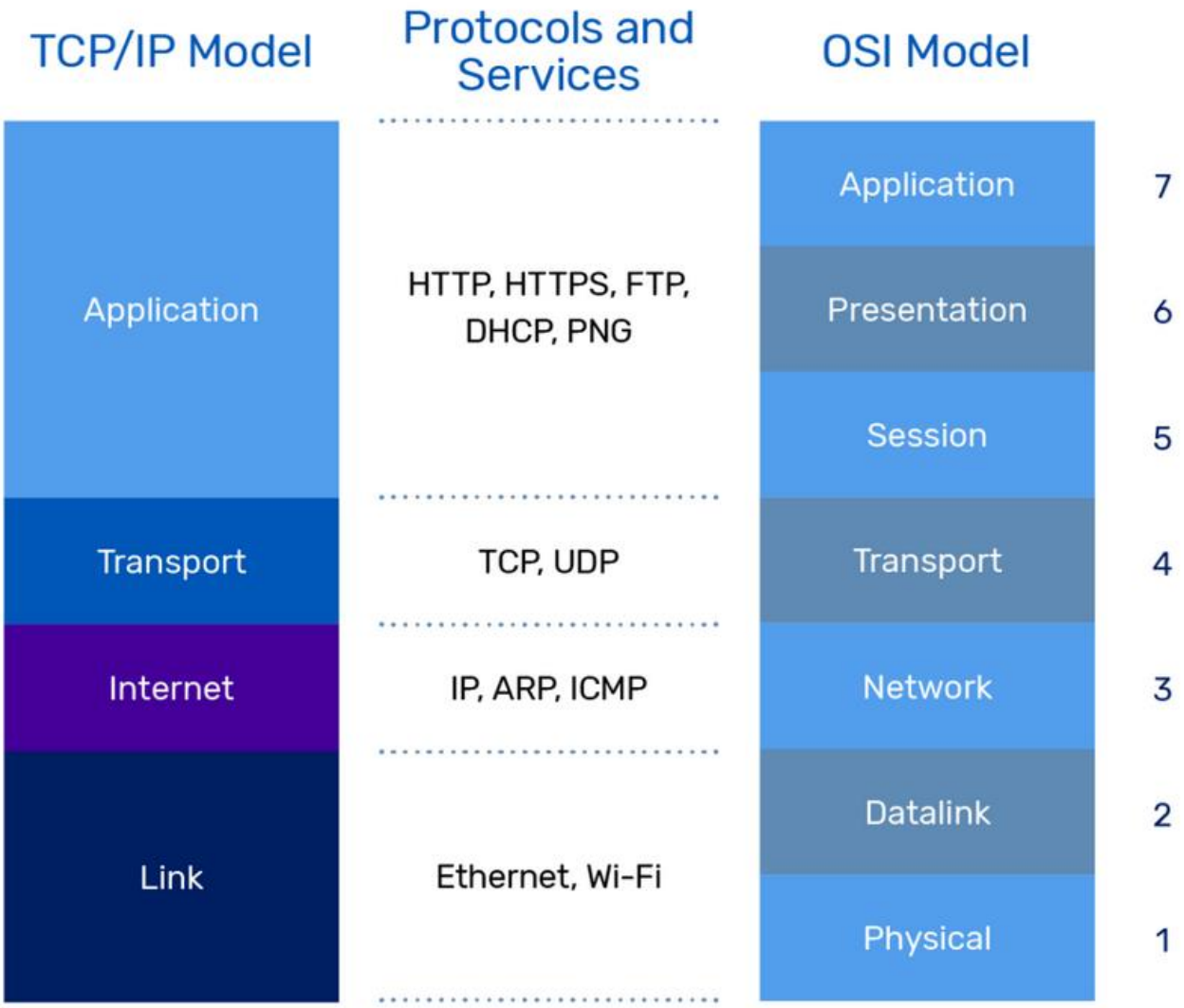

Рисунок 1.1 – Модель OSI

Розподілена архітектура забезпечує кращу масштабованість та відмовостійкість системи, оскільки навантаження розподіляється між множиною вузлів, а збій окремого вузла не призводить до повної втрати функціональності системи. Локальна обробка даних дозволяє зменшити навантаження на мережеві канали та забезпечити швидше реагування на локальні інциденти.

Гібридні архітектурні підходи поєднують переваги централізованого та розподіленого підходів, забезпечуючи гнучкість у виборі оптимальної стратегії для різних компонентів системи моніторингу. Гібридна архітектура може включати централізовану систему управління та конфігурації з розподіленими вузлами збору та первинної обробки даних.

Ієрархічна архітектура моніторингу представляє спеціалізований випадок розподіленої архітектури, що організована у вигляді багаторівневої структури. Нижній рівень ієрархії включає агентів збору даних, розташованих безпосередньо на мережевому обладнанні або в безпосередній близькості до моніторованих ресурсів. Середній рівень складають регіональні або сегментні сервери агрегації, що збирають та попередньо обробляють дані від множини агентів. Верхній рівень представлений центральними серверами управління та аналізу, що забезпечують консолідований погляд на стан всієї інфраструктури.

Ієрархічна архітектура забезпечує оптимальний баланс між централізованим управлінням та розподіленою обробкою даних, дозволяючи ефективно масштабувати систему моніторингу для великих та складних мережевих інфраструктур. Кожен рівень ієрархії може бути оптимізований для виконання специфічних завдань, що підвищує загальну ефективність системи.

Принципи побудови багаторівневих систем моніторингу базуються на концепції поділу функціональності між різними рівнями абстракції відповідно до моделі OSI або інших стандартизованих моделей мережевої архітектури. Фізичний рівень моніторингу включає контроль стану мережевого обладнання, якості сигналу, температурних режимів та інших фізичних параметрів інфраструктури.

Канальний рівень моніторингу забезпечує контроль передачі кадрів даних, виявлення колізій та помилок передачі, моніторинг завантаженості портів комутаторів та інших параметрів локальних мережевих сегментів. Мережевий рівень фокусується на моніторингу маршрутизації пакетів, аналізі топології мережі, контролі доступності мережевих вузлів та оптимізації маршрутів передачі даних.

Транспортний рівень моніторингу включає аналіз встановлення та завершення з'єднань, контроль потоків даних, виявлення втрат пакетів та моніторинг продуктивності транспортних протоколів. Рівень сесій забезпечує

моніторинг встановлення та управління сесіями між додатками, контроль часу життя сесій та аналіз патернів використання.

Рівень представлення включає моніторинг процесів кодування та декодування даних, контроль ефективності компресії та шифрування, аналіз форматів даних. Рівень додатків забезпечує моніторинг продуктивності бізнес-додатків, аналіз користувацького досвіду, контроль дотримання угод про рівень сервісу та моніторинг специфічних для додатків метрик.

Особливості інтеграції засобів моніторингу з існуючою мережевою інфраструктурою визначаються необхідністю мінімізації впливу на продуктивність мережі при забезпеченні максимальної повноти та точності збору даних. Пасивні методи інтеграції базуються на аналізі копій мережевого трафіка, отриманих через дзеркалювання портів, мережеві відгалужувачі або спеціалізовані пристрої збору трафіка.

Активні методи інтеграції передбачають генерацію синтетичного трафіка для тестування доступності та продуктивності мережевих сервісів. Гібридні підходи поєднують пасивний моніторинг реального трафіка з активним тестуванням критично важливих сервісів для забезпечення комплексного контролю стану мережі.

Вимоги до масштабованості систем моніторингу визначаються необхідністю адаптації до зростання мережевої інфраструктури без суттєвого погіршення продуктивності або збільшення складності управління. Горизонтальна масштабованість передбачає можливість додавання нових вузлів обробки для розподілення навантаження. Вертикальна масштабованість включає можливість нарощування потужності існуючих вузлів системи.

Вимоги до надійності систем моніторингу включають забезпечення високої доступності, відмовостійкості та відновлюваності після збоїв. Це досягається через резервування критично важливих компонентів, реплікацію даних, автоматичне переключення на резервні системи та регулярне створення резервних копій конфігурацій та історичних даних.

1.3 Методології аналізу мережного трафіка

Методології аналізу мережного трафіка становлять комплексну систему підходів, методів та алгоритмів, призначених для вилучення корисної інформації з великих обсягів мережевих даних з метою забезпечення ефективного управління мережевою інфраструктурою. Сучасні методології аналізу характеризуються інтеграцією традиційних статистичних методів з передовими технологіями машинного навчання та штучного інтелекту, що дозволяє досягти нового рівня автоматизації та точності аналізу.

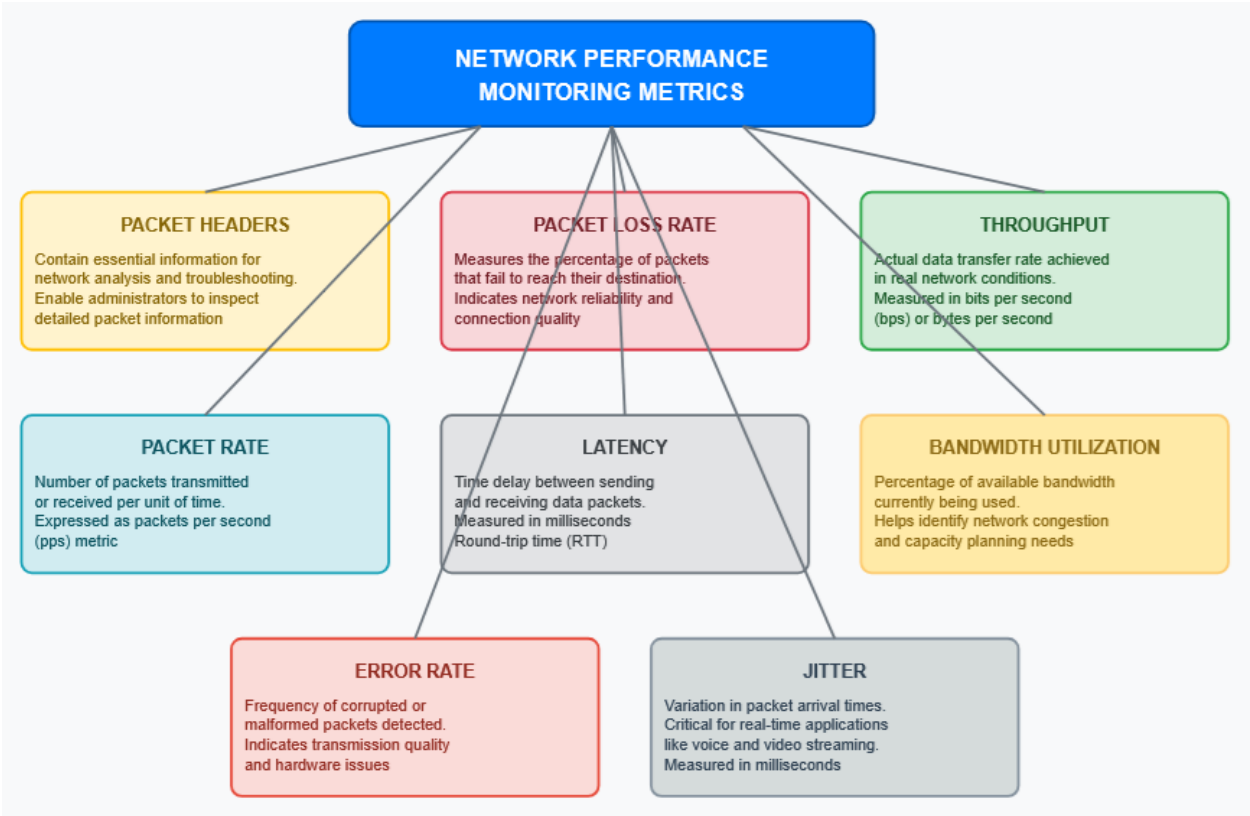

Рисунок 1.2 – Метрики моніторингу мережі

Статистичні методи обробки мережевих даних формують фундаментальну основу для аналізу трафіка та включають широкий спектр математичних підходів до характеризації, класифікації та прогнозування поведінки мережевих потоків. Дескриптивна статистика забезпечує базове

розуміння характеристик мережевого трафіка через обчислення основних статистичних показників, таких як середні значення, медіана, мода, стандартне відхилення, асиметрія та ексцес розподілів різних мережевих метрик.

Аналіз розподілів мережевих даних дозволяє ідентифікувати типові патерни поведінки трафіка та виявляти аномальні ситуації. Нормальний розподіл часто спостерігається у метриках, пов'язаних з агрегованим навантаженням мережі, тоді як експоненційні та степеневі розподіли характерні для розмірів файлів, тривалості сесій та інтервалів між подіями.

Кореляційний аналіз мережевих даних забезпечує виявлення взаємозв'язків між різними мережевими метриками та дозволяє ідентифікувати причинно-наслідкові зв'язки між подіями в мережі. Автокореляційний аналіз часових рядів мережевих даних виявляє періодичні патерни та тренди, що є критично важливими для прогнозування майбутнього навантаження та планування ресурсів.

Аналіз трендів у мережевих даних базується на методах виявлення довгострокових тенденцій зміни ключових мережевих метрик. Лінійні тренди можуть свідчити про постійне зростання або зменшення навантаження на мережу, тоді як нелінійні тренди можуть вказувати на складніші залежності, пов'язані з циклічними змінами у використанні мережевих ресурсів.

Сезонний аналіз мережевого трафіка дозволяє ідентифікувати регулярні патерни змін навантаження, пов'язані з робочими циклами організації, добовими ритмами активності користувачів або іншими періодичними факторами. Розуміння сезонних патернів є критично важливим для планування ресурсів та оптимізації продуктивності мережі.

Методи виявлення аномалій у мережевому трафіку включають статистичні підходи, базовані на аналізі відхилень від нормальної поведінки, та алгоритмічні методи, що використовують машинне навчання для автоматичного виявлення незвичайних патернів. Статистичні методи виявлення аномалій базуються на встановленні порогових значень для

ключових метрик на основі історичних даних та статистичних розподілів.

Метод стандартних відхилень використовує статистичні властивості нормального розподілу для ідентифікації значень, що суттєво відхиляються від середніх показників. Метод інтерквартильного розмаху базується на аналізі розподілу даних через квартилі та є більш стійким до викидів порівняно з методами, базованими на середніх значеннях.

Прогнозування навантаження на мережу представляє складне завдання, що вимагає врахування множини факторів, включаючи історичні тренди, сезонні коливання, планові зміни в інфраструктурі та непередбачувані події. Методи короткострокового прогнозування фокусуються на передбаченні навантаження на період від кількох хвилин до кількох годин та базуються на аналізі поточних трендів та недавньої історії.

Довгострокове прогнозування охоплює періоди від кількох днів до кількох місяців і вимагає врахування складніших факторів, таких як ріст організації, зміни в бізнес-процесах, сезонні коливання та технологічні зміни. Економетричні моделі, такі як ARIMA, експоненційне згладжування та сезонна декомпозиція, широко використовуються для довгострокового прогнозування мережевого навантаження.

Методи класифікації мережного трафіка забезпечують автоматичне розпізнавання типів додатків та протоколів на основі аналізу характеристик мережевих потоків. Класифікація на основі портів використовує стандартні номери портів для ідентифікації додатків, але має обмеження через використання динамічних портів та тунелювання трафіка.

Глибока інспекція пакетів забезпечує більш точну класифікацію через аналіз змісту пакетів та виявлення специфічних сигнатур протоколів та додатків. Статистична класифікація базується на аналізі статистичних характеристик мережевих потоків, таких як розподіл розмірів пакетів, інтервали між пакетами та тривалість потоків.

Методи машинного навчання революціонізували підходи до аналізу мережного трафіка, забезпечивши можливість автоматичного виявлення

складних патернів та залежностей у великих обсягах мережевих даних. Наглядове навчання використовується для задач класифікації трафіка, прогнозування навантаження та виявлення аномалій, коли доступні мічені навчальні дані.

Алгоритми дерев рішень забезпечують інтерпретовані моделі класифікації трафіка, що дозволяють зрозуміти логіку прийняття рішень. Методи опорних векторів ефективні для класифікації в просторах високої розмірності та забезпечують хорошу генералізацію на нових даних.

Ненаглядове навчання застосовується для виявлення прихованих структур у мережевих даних, кластеризації схожих типів трафіка та виявлення аномалій без попереднього знання про нормальну поведінку. Алгоритми кластеризації, такі як k-means, DBSCAN та ієрархічна кластеризація, використовуються для групування схожих мережевих потоків та ідентифікації різних класів трафіка.

Методи зменшення розмірності, такі як аналіз головних компонент та t-SNE, дозволяють візуалізувати та аналізувати складні багатовимірні мережеві дані в просторах меншої розмірності.

Глибоке навчання та нейронні мережі представляють найбільш передові підходи до аналізу мережного трафіка, що дозволяють виявляти надзвичайно складні патерни та залежності. Згорткові нейронні мережі ефективно аналізують просторові структури в мережевих даних, тоді як рекурентні нейронні мережі спеціалізуються на аналізі часових послідовностей.

Автоенкодери використовуються для виявлення аномалій через навчання на нормальних даних та ідентифікацію випадків, що погано реконструюються моделлю. Генеративно-змагальні мережі можуть застосовуватися для генерації синтетичного мережевого трафіка для тестування та валідації систем моніторингу.

Ансамблеві методи поєднують множину різних алгоритмів для досягнення кращої точності та стабільності результатів аналізу. Bagging,

boosting та stacking є основними підходами до створення ансамблевих моделей для аналізу мережного трафіка.

Онлайн-навчання дозволяє моделям адаптуватися до змін у характеристиках мережного трафіка в режимі реального часу, що є критично важливим для динамічних мережевих середовищ. Інкрементальне навчання забезпечує оновлення моделей при надходженні нових даних без необхідності повторного навчання на всьому наборі даних.

2 ОГЛЯД СУЧАСНИХ ПРОГРАМНИХ ЗАСОБІВ МОНІТОРИНГУ МЕРЕЖНОГО ТРАФІКА

2.1 Класифікація програмних рішень для мережного моніторингу

Сучасний ринок програмних засобів моніторингу мережного трафіка характеризується надзвичайною різноманітністю рішень, що охоплюють широкий спектр функціональних можливостей, архітектурних підходів та бізнес-моделей. Систематизація цих рішень потребує комплексного підходу до класифікації, що враховує множину критеріїв та дозволяє організаціям обрати оптимальне рішення відповідно до їх специфічних потреб та вимог.

Основною категоризацією програмних засобів моніторингу є поділ на комерційні, відкриті та гібридні рішення, кожне з яких має унікальні характеристики, переваги та обмеження. Комерційні рішення представлені продуктами провідних технологічних компаній та спеціалізованих постачальників програмного забезпечення, що пропонують комплексні платформи з повним циклом технічної підтримки, документації та професійних сервісів.

Комерційні програмні засоби характеризуються високим рівнем інтеграції різних функцій моніторингу в єдиній платформі, розвиненими можливостями візуалізації та звітності, професійною технічною підтримкою та регулярними оновленнями функціональності. Ці рішення зазвичай пропонують розширені можливості масштабування, інтеграції з корпоративними системами та відповідність галузевим стандартам безпеки та аудиту.

Вартісна модель комерційних рішень може базуватися на різних принципах, включаючи ліцензування за кількістю моніторованих пристроїв, обсягом трафіка, функціональними можливостями або комбінацією цих факторів. Багато постачальників пропонують гнучкі моделі ціноутворення,

включаючи підписки, оплату за використання та гібридні моделі.

Відкриті рішення представляють альтернативний підхід, базований на принципах відкритого програмного коду та спільнотного розвитку. Ці рішення характеризуються відсутністю ліцензійних витрат, повною прозорістю коду, можливістю кастомізації та адаптації під специфічні потреби організації. Відкриті рішення часто розробляються та підтримуються активними спільнотами розробників, що забезпечує швидке виправлення помилок та впровадження нових функцій.

Однак відкриті рішення можуть потребувати значних внутрішніх ресурсів для впровадження, налаштування та підтримки, а також можуть мати обмежені можливості технічної підтримки порівняно з комерційними альтернативами. Деякі організації компенсують ці обмеження через залучення спеціалізованих консультантів або постачальників комерційної підтримки для відкритих рішень.

Гібридні рішення поєднують елементи комерційних та відкритих підходів, пропонуючи базову функціональність у відкритому коді з додатковими комерційними модулями або сервісами. Цей підхід дозволяє організаціям початково використовувати безкоштовні версії продуктів з можливістю поступового розширення функціональності через придбання комерційних доповнень.

Позиціонування програмних засобів на ринку визначається їх цільовими сегментами, функціональними можливостями та стратегіями конкуренції. Рішення рівня підприємства орієнтовані на великі корпорації з складними мережевими інфраструктурами та високими вимогами до функціональності, масштабованості та надійності. Ці рішення зазвичай включають розширені можливості інтеграції, автоматизації та управління через API.

Рішення для середнього бізнесу фокусуються на балансі між функціональністю та вартістю, пропонуючи основні можливості моніторингу з спрощеними процесами впровадження та управління. Ці продукти часто

включають попередньо налаштовані шаблони та мастери конфігурації для прискорення початкового розгортання.

Спеціалізовані рішення орієнтовані на конкретні технології, галузі або типи мережевої інфраструктури. Наприклад, рішення для моніторингу хмарних середовищ, програмно-визначених мереж, промислових мереж або телекомунікаційних інфраструктур мають специфічні функції та можливості, адаптовані до особливостей цих сегментів.

Цільові сегменти застосування визначаються специфічними потребами різних типів організацій та галузей. Телекомунікаційні компанії потребують рішень з розширеними можливостями моніторингу якості сервісу, аналізу продуктивності мереж та управління SLA. Фінансові установи зосереджуються на рішеннях з високими вимогами до безпеки, аудиту та дотримання регуляторних вимог.

Виробничі підприємства потребують рішень, адаптованих для промислових протоколів та технологій, з можливостями інтеграції з системами автоматизації виробництва. Освітні установи зосереджуються на рішеннях з можливостями управління доступом користувачів, моніторингу використання ресурсів та забезпечення безпеки мережі.

Тенденції розвитку ринку програмних засобів мережного моніторингу відображають загальні тренди цифрової трансформації та еволюції мережевих технологій. Перехід до хмарних рішень стає домінуючою тенденцією, оскільки організації прагнуть зменшити витрати на інфраструктуру, підвищити гнучкість та отримати доступ до передових функцій без необхідності значних капітальних інвестицій

Хмарні рішення для мережного моніторингу пропонують переваги швидкого розгортання, автоматичного масштабування, регулярних оновлень та доступу до передових аналітичних можливостей. Software-as-a-Service модель дозволяє організаціям зосередитися на основних бізнес-процесах, делегуючи управління інфраструктурою моніторингу спеціалізованим постачальникам.

Інтеграція з DevOps-процесами стає критично важливою вимогою для сучасних програмних засобів моніторингу, оскільки організації впроваджують практики безперервної інтеграції та доставки. Рішення моніторингу повинні підтримувати автоматизацію розгортання, конфігурацію через код, інтеграцію з системами управління версіями та підтримку контейнерних технологій.

Впровадження технологій штучного інтелекту та машинного навчання кардинально змінює можливості програмних засобів моніторингу, дозволяючи автоматизувати складні процеси аналізу, прогнозування та прийняття рішень. Інтелектуальні системи моніторингу здатні самостійно виявляти аномалії, прогнозувати потенційні проблеми та рекомендувати оптимальні дії для їх вирішення.

2.2 Функціональний аналіз провідних програмних платформ

Функціональний аналіз сучасних програмних платформ моніторингу мережного трафіка вимагає комплексного дослідження їх архітектурних особливостей, технологічних рішень та користувацьких можливостей. Провідні платформи на ринку демонструють різноманітні підходи до вирішення завдань мережного моніторингу, кожен з яких має унікальні переваги та сфери оптимального застосування.

Можливості збору та агрегації мережевих даних становлять фундаментальну основу будь-якої платформи моніторингу та визначають її здатність забезпечувати повний та точний огляд стану мережевої інфраструктури. Сучасні платформи підтримують множину методів збору даних, включаючи SNMP, NetFlow, sFlow, IPFIX, WMI, API та спеціалізовані агенти.

SNMP (рисунок 2.1) залишається одним з найпоширеніших протоколів для збору даних з мережевого обладнання, забезпечуючи доступ до широкого спектру метрик продуктивності та стану пристроїв. Сучасні платформи

підтримують всі версії SNMP, включаючи SNMPv3 з розширеними можливостями безпеки, та пропонують інтелектуальні механізми оптимізації запитів для мінімізації навантаження на мережу.

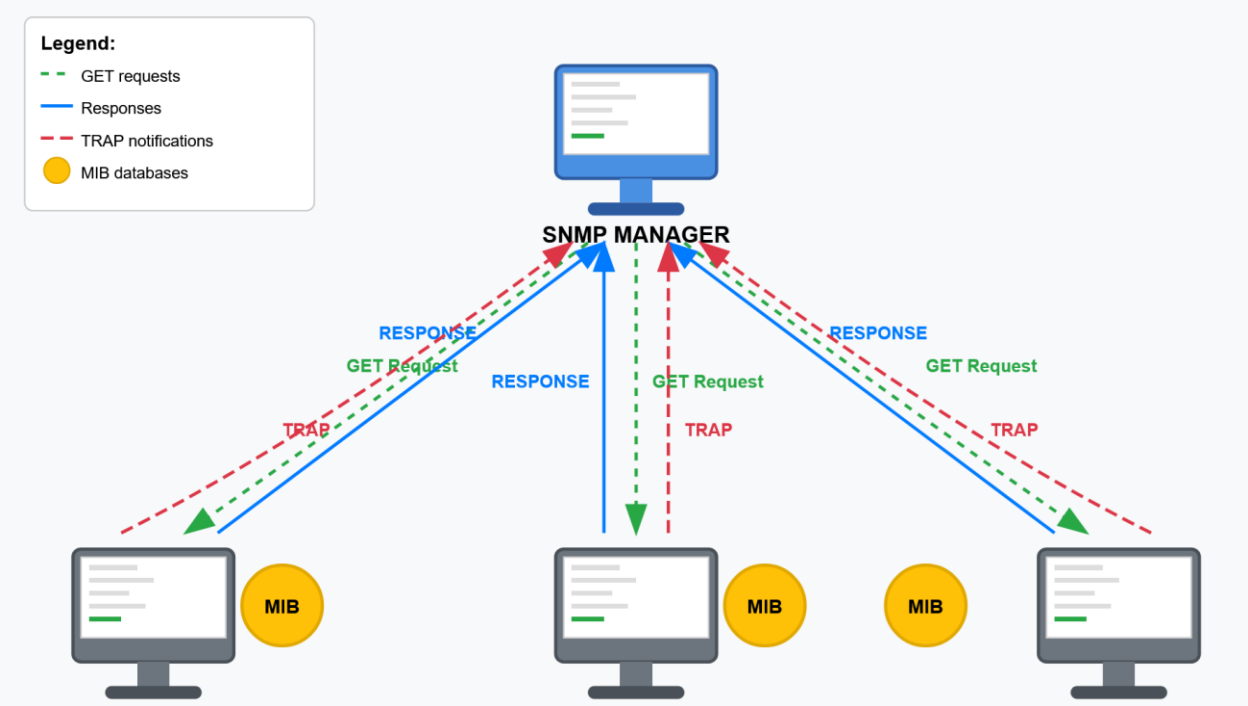

Рисунок 2.1 – Протокол SNMP

Технології потокового аналізу, такі як NetFlow, sFlow та IPFIX, забезпечують детальну інформацію про мережні потоки без необхідності глибокої інспекції кожного пакета. Ці технології дозволяють аналізувати патерни використання мережевих ресурсів, ідентифікувати додатки та протоколи, виявляти аномальну активність та оптимізувати мережеву продуктивність.

Агентські технології збору даних забезпечують більш детальний моніторинг серверів, робочих станцій та спеціалізованого обладнання. Сучасні агенти характеризуються мінімальним впливом на продуктивність моніторованих систем, здатністю працювати в автономному режимі та можливостями локальної обробки даних.

API-інтеграції дозволяють збирати дані з хмарних сервісів, додатків та систем, що не підтримують традиційні протоколи моніторингу. REST API,

GraphQL та інші сучасні інтерфейси забезпечують гнучкі можливості інтеграції з різноманітними джерелами даних.

Методи візуалізації та представлення даних є критично важливими для ефективного використання систем моніторингу різними категоріями користувачів. Сучасні платформи пропонують широкий спектр візуалізацій, від простих графіків та таблиць до складних інтерактивних дашбордів та тривимірних представлень мережевої топології.

Dashboard-технології забезпечують консолідоване представлення ключової інформації про стан мережевої інфраструктури, дозволяючи користувачам швидко оцінити загальну ситуацію та ідентифікувати області, що потребують уваги. Сучасні дашборди підтримують персоналізацію, рольовий доступ, інтерактивні елементи та можливості детального аналізу.

Топологічна візуалізація надає графічне представлення мережевої інфраструктури з можливістю відображення стану, навантаження та взаємозв'язків між різними компонентами. Інтелектуальні алгоритми автоматичного виявлення топології спрощують процес створення та підтримки актуальних схем мережі.

Геоінформаційна візуалізація дозволяє відобразити розподілену мережевую інфраструктуру на географічних картах, що особливо корисно для організацій з множиною географічно розподілених локацій. Інтеграція з картографічними сервісами забезпечує актуальні та детальні карти з можливостями масштабування та персоналізації.

Засоби конфігурації моніторингу визначають складність та гнучкість налаштування системи під специфічні потреби організації. Сучасні платформи пропонують різні рівні конфігурації, від простих майстрів налаштування до складних скриптових мов та API програмування.

Автоматичне виявлення пристроїв та сервісів значно спрощує початкове налаштування моніторингу, автоматично ідентифікуючи мережеві компоненти та застосовуючи відповідні шаблони моніторингу. Інтелектуальні алгоритми виявлення використовують комбінацію різних

технік, включаючи сканування мережі, аналіз DNS записів, SNMP discovery та інтеграцію з системами управління конфігурацією.

Шаблонні системи дозволяють стандартизувати налаштування моніторингу для однотипних пристроїв та сервісів, забезпечуючи консистентність конфігурації та спрощуючи управління великими інфраструктурами. Розвинені шаблонні системи підтримують наслідування, параметризацію та умовну логіку.

Налаштування алертів та сповіщень є критично важливою функціональністю, що забезпечує своєчасне реагування на проблеми та інциденти. Сучасні системи алертів підтримують складну логіку тригерів, ескалації, групування подібних алертів та інтеграцію з різноманітними каналами сповіщень.

Інтелектуальні алгоритми фільтрації алертів допомагають зменшити кількість false positive сповіщень та забезпечити фокусування уваги на критично важливих проблемах. Базові фільтри, кореляційний аналіз та машинне навчання використовуються для підвищення релевантності алертів.

Інтеграція з зовнішніми системами сповіщень включає підтримку електронної пошти, SMS, месенджерів, систем тікетів та інших корпоративних платформ комунікації. Webhook та API інтеграції забезпечують можливості автоматизації реагування на інциденти.

Можливості інтеграції з системами управління IT-інфраструктурою визначають здатність платформи моніторингу функціонувати як частина більшої екосистеми корпоративних систем. ITSM інтеграції дозволяють автоматично створювати тікети для виявлених проблем, відстежувати статус їх вирішення та забезпечувати зворотний зв'язок між системами моніторингу та управління сервісами.

CMDB інтеграції забезпечують синхронізацію інформації про конфігурацію інфраструктури між системами моніторингу та базами даних управління конфігурацією. Це дозволяє автоматично оновлювати параметри моніторингу при змінах в інфраструктурі та забезпечувати консистентність

даних.

Інтеграція з системами автоматизації дозволяє реалізувати сценарії автоматичного реагування на виявлені проблеми, включаючи перезапуск сервісів, перерозподіл навантаження, активацію резервних систем та інші відновлювальні дії.

2.3 Технічні характеристики та вимоги до впровадження

Технічні характеристики програмних засобів моніторингу мережного трафіка та вимоги до їх впровадження визначають практичну можливість та ефективність використання цих рішень в конкретних операційних середовищах. Детальний аналіз технічних аспектів є критично важливим для прийняття обґрунтованих рішень щодо вибору та планування впровадження систем моніторингу.

Системні вимоги програмних продуктів варіюються від мінімальних конфігурацій для простих рішень до потужних кластерних систем для комплексних платформ корпоративного рівня. Вимоги до процесорних ресурсів залежать від обсягів оброблюваних даних, складності аналітичних алгоритмів та кількості одночасних користувачів системи.

Сучасні системи моніторингу активно використовують багатопоточність та паралельну обробку для ефективного використання багатоядерних процесорів. Оптимізовані алгоритми обробки даних та ефективні структури даних дозволяють досягти високої продуктивності навіть на відносно скромних апаратних конфігураціях.

Вимоги до оперативної пам'яті визначаються необхідністю зберігання буферів даних, кешування часто використовуваної інформації та роботи аналітичних алгоритмів. Системи реального часу потребують достатніх обсягів пам'яті для забезпечення безперервної обробки потоків даних без втрат інформації.

Дискова підсистема відіграє критично важливу роль у системах

моніторингу, оскільки вони генерують великі обсяги історичних даних. Вимоги до дискового простору залежать від кількості моніторованих об'єктів, частоти збору даних та періоду збереження історичної інформації. Високопродуктивні дискові підсистеми з SSD накопичувачами рекомендуються для забезпечення швидкого доступу до даних та ефективної роботи баз даних.

Мережеві вимоги включають достатню пропускну здатність для збору даних з розподілених компонентів інфраструктури та надійні мережеві з'єднання для забезпечення безперервності моніторингу. Системи моніторингу повинні бути спроектовані з урахуванням можливих мережевих збоїв та включати механізми буферизації та відновлення з'єднань.

Особливості установки та конфігурації в різних операційних середовищах відображають різноманітність IT-інфраструктур сучасних організацій. Підтримка множини операційних систем, включаючи різні дистрибутиви Linux, Windows Server, Unix-системи та віртуалізовані середовища, є важливим фактором для багатьох організацій.

Контейнеризація стає все більш популярним підходом до розгортання систем моніторингу, забезпечуючи консистентність середовища виконання, спрощення процесів розгортання та можливості горизонтального масштабування. Docker контейнери та Kubernetes оркестрація дозволяють ефективно управляти життєвим циклом компонентів системи моніторингу.

Хмарні розгортання пропонують альтернативу традиційним on-premises інсталяціям, забезпечуючи гнучкість масштабування, зменшення операційних витрат та доступ до керованих сервісів інфраструктури. Гібридні моделі розгортання дозволяють поєднувати переваги хмарних та локальних рішень.

Продуктивність та масштабованість є критично важливими характеристиками, що визначають здатність системи моніторингу справлятися з зростаючими обсягами даних та розширенням інфраструктури. Горизонтальна масштабованість дозволяє додавати нові вузли обробки для

розподілення навантаження, тоді як вертикальна масштабованість включає нарощування ресурсів окремих компонентів.

Архітектурні рішення для забезпечення масштабованості включають розподілені бази даних, мікросервісну архітектуру, асинхронну обробку подій та ефективні алгоритми балансування навантаження. Сучасні системи моніторингу використовують технології Big Data для обробки величезних обсягів мережевих даних.

Здатність обробляти великі обсяги мережевого трафіка в режимі реального часу вимагає оптимізованих алгоритмів обробки потоків даних, ефективних структур даних та розумного кешування. Stream processing технології дозволяють аналізувати дані "на льоту" без необхідності попереднього збереження.

Особливості інтеграції з мережевим обладнанням різних виробників відображають гетерогенність сучасних мережевих інфраструктур. Підтримка стандартних протоколів моніторингу забезпечує сумісність з широким спектром пристроїв, але різні виробники можуть мати специфічні особливості реалізації стандартів.

Vendor-specific розширення та приватні MIB дозволяють отримати доступ до унікальних функцій та метрик конкретного обладнання, але можуть ускладнити міграцію на обладнання інших виробників. Абстракція від специфічних особливостей обладнання через уніфіковані інтерфейси дозволяє створювати більш гнучкі та портабельні рішення моніторингу.

Підтримка стандартних протоколів моніторингу включає SNMP всіх версій, Telnet, SSH, WMI, REST API та інші поширені інтерфейси. Сучасні тенденції включають підтримку YANG моделей, NETCONF протоколу та інших технологій програмно-визначених мереж.

Можливості кастомізації під специфічні потреби організації визначають гнучкість системи моніторингу та її здатність адаптуватися до унікальних вимог бізнесу. Скриптові інтерфейси дозволяють створювати власні перевірки, збирачі даних та алгоритми обробки інформації.

Plugin архітектури забезпечують можливості розширення функціональності системи через сторонні модулі або власні розробки. Відкриті API дозволяють інтегрувати системи моніторингу з власними додатками та автоматизувати різноманітні операційні процеси.

Системи налаштувань та конфігурації повинні бути достатньо гнучкими для адаптації до специфічних організаційних політик, процедур та вимог безпеки. Рольова модель доступу дозволяє налаштувати різні рівні доступу для різних категорій користувачів.

3 ПОРІВНЯЛЬНИЙ АНАЛІЗ ПРОГРАМНИХ ЗАСОБІВ МОНІТОРИНГУ

3.1 Методологія порівняльного аналізу

Розробка обґрунтованої методології порівняльного аналізу програмних засобів моніторингу мережного трафіка є фундаментальним завданням, що визначає об'єктивність та практичну цінність дослідження. Методологічний підхід повинен забезпечувати системний, комплексний та збалансований аналіз різноманітних аспектів програмних рішень, враховуючи специфічні потреби різних категорій користувачів та типів організацій.

Формування критеріїв оцінки становить першочергове завдання методологічного дослідження та вимагає глибокого розуміння потреб ринку, технологічних тенденцій та практичних вимог до систем моніторингу. Функціональні критерії охоплюють широкий спектр можливостей програмних засобів, включаючи базові функції збору та аналізу даних, розширені можливості візуалізації та звітності, засоби автоматизації та інтеграції з зовнішніми системами.

Глибина функціонального покриття визначається здатністю системи моніторингу забезпечувати комплексний контроль всіх рівнів мережевої інфраструктури, від фізичного рівня до рівня додатків. Оцінюється підтримка різноманітних протоколів моніторингу, можливості збору даних з гетерогенного обладнання та здатність адаптуватися до специфічних вимог різних типів мережевих архітектур.

Гнучкість конфігурації та кастомізації є критично важливим аспектом, що визначає здатність системи адаптуватися до унікальних потреб організації. Оцінюються можливості створення власних метрик, алгоритмів аналізу, шаблонів моніторингу та інтеграційних модулів. Якість API та документації визначає можливості розширення функціональності системи власними розробками.

Технічні критерії включають оцінку архітектурних рішень, продуктивності, масштабованості, надійності та вимог до інфраструктури. Архітектурна зрілість оцінюється через аналіз використання сучасних технологічних підходів, таких як мікросервісна архітектура, контейнеризація, хмарна готовність та підтримка розподілених систем.

Продуктивність системи моніторингу визначається її здатністю обробляти великі обсяги даних в режимі реального часу без суттєвого впливу на моніторовану інфраструктуру. Оцінюються показники пропускної здатності, затримки обробки, ефективності використання ресурсів та здатності до горизонтального масштабування.

Масштабованість включає аналіз можливостей системи адаптуватися до зростання кількості моніторованих об'єктів, обсягів трафіка та числа користувачів. Вертикальна масштабованість характеризує можливості нарощування потужності окремих компонентів, тоді як горизонтальна масштабованість визначає здатність до розподіленого розгортання.

Критерії зручності використання охоплюють якість користувацького інтерфейсу, інтуїтивність навігації, ефективність робочих процесів та можливості персоналізації для різних ролей користувачів. Оцінюється крива навчання для нових користувачів, доступність документації та навчальних матеріалів, якість технічної підтримки.

Ергономіка інтерфейсу включає аналіз візуального дизайну, логічності організації функцій, ефективності навігації та відповідності сучасним стандартам користувацького досвіду. Мобільна підтримка та адаптивність інтерфейсу стають все більш важливими в контексті сучасних робочих процесів.

Критерії вартості володіння включають не лише початкові ліцензійні витрати, але й усі пов'язані витрати протягом життєвого циклу системи. Ліцензійні моделі варіюються від одноразових платежів до підписок, від оплати за кількість моніторованих об'єктів до необмежених ліцензій.

Витрати на впровадження включають професійні сервіси, навчання

персоналу, міграцію даних та інтеграцію з існуючими системами. Операційні витрати охоплюють технічну підтримку, оновлення, адміністрування системи та витрати на інфраструктуру.

Критерії підтримки з боку розробника включають якість документації, доступність технічної підтримки, частоту оновлень, активність спільноти користувачів та довгострокову стратегію розвитку продукту. Комерційні рішення зазвичай пропонують професійну підтримку з гарантованим часом відповіді, тоді як відкриті рішення покладаються на спільнотну підтримку.

Система показників для кількісної та якісної оцінки повинна забезпечувати об'єктивне порівняння різних програмних рішень. Кількісні показники включають метрики продуктивності, такі як кількість оброблюваних пакетів на секунду, час відгуку системи, обсяг споживаних ресурсів та максимальна кількість підтримуваних об'єктів моніторингу.

Якісні показники оцінюються через експертні оцінки, опитування користувачів та аналіз функціональних можливостей. Використовуються шкали оцінок, рейтингові системи та порівняльні таблиці для систематизації результатів аналізу.

Методи збору даних для порівняльного аналізу включають лабораторне тестування, аналіз документації, інтерв'ю з постачальниками та користувачами, аналіз відгуків спільноти та експертні оцінки. Комбінація різних методів забезпечує всебічність та достовірність результатів дослідження.

Обґрунтування вибору програмних засобів для детального дослідження базується на аналізі ринкових позицій, функціональних можливостей, представленості в різних сегментах ринку та доступності для тестування. Вибірка повинна репрезентувати різні типи рішень, архітектурні підходи та бізнес-моделі.

Встановлення рамок порівняння включає визначення області дослідження, обмежень аналізу, критеріїв включення та виключення продуктів з дослідження. Рамки порівняння повинні бути чітко визначені та

послідовно застосовані до всіх досліджуваних рішень.

3.2 Аналіз функціональних можливостей

Детальний аналіз функціональних можливостей обраних програмних засобів моніторингу мережного трафіка передбачає систематичне дослідження всіх аспектів їх функціональності з метою виявлення переваг, недоліків та специфічних особливостей кожного рішення. Цей аналіз формує основу для об'єктивного порівняння та обґрунтованого вибору оптимального рішення для конкретних потреб організації.

Можливості збору мережевих даних становлять фундаментальну основу будь-якої системи моніторингу та визначають її здатність забезпечувати повне та точне уявлення про стан мережевої інфраструктури. Аналіз включає оцінку підтримуваних протоколів збору даних, їх ефективності, надійності та впливу на продуктивність моніторованої інфраструктури.

SNMP підтримка аналізується з точки зору сумісності з різними версіями протоколу, ефективності обробки великих MIB дерев, можливостей bulk операцій та оптимізації запитів. Оцінюється здатність системи автоматично виявляти та класифікувати SNMP об'єкти, обробляти custom MIB та адаптуватися до специфічних реалізацій різних виробників обладнання.

Підтримка протоколів потокового аналізу, таких як NetFlow, sFlow, IPFIX та J-Flow, досліджується з точки зору повноти реалізації стандартів, можливостей обробки різних версій протоколів та ефективності аналізу великих обсягів потокових даних. Аналізуються можливості кореляції потокових даних з іншими джерелами інформації та алгоритми агрегації потоків.

Агентські технології збору даних оцінюються через аналіз їх архітектури, впливу на продуктивність моніторованих систем, можливостей

локальної обробки даних та здатності працювати в автономному режимі. Досліджуються механізми автентифікації та шифрування агентських з'єднань, можливості централізованого управління конфігурацією агентів.

Синтетичний моніторинг включає аналіз можливостей генерації тестового трафіка, симуляції користувацької активності та проведення активних перевірок доступності та продуктивності сервісів. Оцінюються типи підтримуваних синтетичних тестів, можливості географічно розподіленого тестування та інтеграція результатів з пасивним моніторингом.

Методи обробки та аналізу зібраних даних визначають здатність системи перетворювати сирі мережеві дані на корисну інформацію для прийняття рішень. Аналізуються алгоритми агрегації даних, статистичної обробки, виявлення трендів та аномалій, прогнозування та класифікації трафіка.

Системи базелайнування дозволяють встановлювати нормальні рівні продуктивності та поведінки мережевих компонентів, що є критично важливим для ефективного виявлення проблем. Досліджуються алгоритми автоматичного базелайнування, адаптивні пороги та машинне навчання для покращення точності детекції аномалій.

Кореляційний аналіз подій дозволяє виявляти зв'язки між різними мережевими подіями та ідентифікувати кореневі причини проблем. Аналізуються можливості часової кореляції, топологічної кореляції та кореляції на основі бізнес-сервісів.

Засоби візуалізації та звітності є критично важливими для ефективного використання системи моніторингу різними категоріями користувачів. Дослідження включає аналіз типів підтримуваних візуалізацій, можливостей персоналізації дашбордів, якості графічних інтерфейсів та ефективності представлення інформації.

Dashboard функціональність аналізується з точки зору гнучкості компонування, різноманітності віджетів, можливостей інтерактивної взаємодії та адаптивності до різних розмірів екранів. Оцінюються

можливості створення різних типів дашбордів для різних ролей користувачів та автоматичного оновлення інформації.

Топологічна візуалізація досліджується через аналіз алгоритмів автоматичного виявлення топології, якості графічного представлення, можливостей інтерактивної навігації та відображення динамічної інформації про стан мережевих компонентів.

Системи звітності аналізуються з точки зору різноманітності підтримуваних форматів звітів, можливостей планування та автоматичної генерації звітів, гнучкості налаштування змісту та дизайну звітів. Досліджуються можливості створення executive-level звітів, технічних звітів та compliance звітів.

Можливості налаштування моніторингу для різних типів мережевого обладнання та протоколів визначають універсальність та адаптивність системи. Аналізується підтримка обладнання різних виробників, специфічних протоколів та технологій, можливості автоматичного виявлення та класифікації пристроїв.

Шаблонні системи моніторингу досліджуються з точки зору їх гнучкості, можливостей наслідування та параметризації, простоти створення та підтримки шаблонів. Аналізуються готові бібліотеки шаблонів для популярного обладнання та можливості створення власних шаблонів.

Гнучкість конфігурації оцінюється через аналіз можливостей створення власних метрик, алгоритмів аналізу, правил алертів та інтеграційних модулів. Досліджуються скриптові інтерфейси, API для розширення функціональності та можливості інтеграції з зовнішніми системами.

Засоби автоматизації включають можливості автоматичного виявлення та конфігурування моніторингу нових пристроїв, автоматичного реагування на події, планування завдань та інтеграції з системами автоматизації інфраструктури. Аналізуються workflow движки, системи правил та можливості інтеграції з DevOps інструментами.

Інтеграційні можливості досліджуються через аналіз підтримуваних

API, протоколів інтеграції, готових коннекторів до популярних систем та можливостей двостороннього обміну даними. Особлива увага приділяється інтеграції з ITSM системами, SIEM рішеннями, хмарними платформами та системами управління конфігурацією.

3.3 Оцінка ефективності та зручності використання

Комплексна оцінка ефективності та зручності використання програмних засобів моніторингу мережного трафіка вимагає багатоаспектного підходу, що включає аналіз впливу на продуктивність мережевої інфраструктури, якості користувацького досвіду та операційної ефективності. Ця оцінка є критично важливою для розуміння практичної цінності рішень та їх придатності для конкретних операційних сценаріїв.

Аналіз ефективності програмних засобів з точки зору їх впливу на продуктивність мережевої інфраструктури включає дослідження ресурсоспоживання, мережевого навантаження, генерованого процесами моніторингу, та загального впливу на стабільність роботи моніторованих систем. Цей аспект є особливо критичним для високонавантажених продуктивних середовищ, де навіть незначний негативний вплив може мати серйозні наслідки.

Споживання мережевих ресурсів аналізується через вимірювання обсягу трафіка, генерованого різними методами збору даних, частоту запитів до мережевого обладнання та ефективність протоколів моніторингу. SNMP polling оптимізація включає аналіз bulk операцій, intelligent polling intervals та адаптивних алгоритмів, що зменшують навантаження на мережу при збереженні якості моніторингу.

Протоколи потокового аналізу досліджуються з точки зору їх впливу на продуктивність мережевого обладнання, обсягу генерованих потокових даних та ефективності їх обробки. Аналізуються можливості налаштування sampling rates, фільтрації потоків та оптимізації експорту даних.

Системні ресурси, споживані компонентами моніторингу, включають процесорне навантаження, використання оперативної пам'яті, дискового простору та мережевої пропускної здатності. Дослідження проводиться в різних конфігураціях навантаження для визначення масштабованості та граничних можливостей систем.

Алгоритми обробки даних аналізуються з точки зору їх обчислювальної складності, ефективності використання ресурсів та здатності до паралелізації. Особлива увага приділяється real-time обробці великих обсягів даних та можливостям горизонтального масштабування.

Зручність використання для різних категорій користувачів є критично важливим фактором успішного впровадження та експлуатації систем моніторингу. Аналіз включає дослідження інтерфейсів для мережевих адміністраторів, системних інженерів, аналітиків безпеки та керівників IT-підрозділів, кожен з яких має специфічні потреби та рівень технічної експертизи.

Мережеві адміністратори потребують детальної технічної інформації, можливостей швидкої діагностики проблем, доступу до raw даних та інструментів для глибокого аналізу. Оцінюється зручність навігації між різними рівнями деталізації, ефективність пошуку та фільтрації даних, якість діагностичних інструментів.

Системні інженери фокусуються на інтеграційних можливостях, автоматизації процесів, можливостях скриптування та API доступу. Аналізується якість документації API, простота створення власних інтеграцій, можливості bulk операцій та автоматизації рутинних завдань.

Керівники IT-підрозділів потребують high-level оглядів, трендового аналізу, compliance звітності та інформації для прийняття стратегічних рішень. Досліджуються можливості створення executive dashboards, якість аналітичних звітів, можливості прогнозування та планування ресурсів.

Інтерфейси користувача аналізуються з точки зору їх інтуїтивності, консистентності, ефективності робочих процесів та відповідності сучасним

стандартам UX дизайну. Оцінюється логічність організації функцій, якість візуального дизайну, швидкість завантаження сторінок та відзивність інтерфейсу.

Навігаційна структура досліджується через аналіз ієрархії меню, breadcrumb навігації, пошукових можливостей та shortcut функцій. Важливим аспектом є можливість швидкого доступу до часто використовуваних функцій та персоналізації робочого простору.

Мобільна підтримка стає все більш важливою в контексті сучасних робочих процесів та включає аналіз адаптивності інтерфейсу, функціональності мобільних додатків та можливостей отримання критичних сповіщень на мобільні пристрої.

Якість документації та навчальних матеріалів є критично важливою для ефективного використання системи та зменшення часу навчання нових користувачів. Аналіз включає оцінку повноти документації, її актуальності, структурованості та доступності для різних рівнів користувачів.

Технічна документація досліджується з точки зору детальності опису функцій, якості прикладів використання, покриття різних сценаріїв використання та регулярності оновлень. Важливим аспектом є наявність troubleshooting guides, best practices та архітектурних рекомендацій.

Навчальні матеріали включають онлайн курси, відеоуроки, вебінари та практичні лабораторні роботи. Оцінюється якість педагогічного дизайну, актуальність контенту, різноманітність форматів навчання та можливості сертифікації.

Спільнота користувачів та екосистема навколо продукту досліджуються через аналіз активності форумів, якості спільнотної підтримки, наявності третісторонніх ресурсів та інтеграцій. Активна спільнота може значно полегшити процес навчання та вирішення проблем.

Доступність технічної підтримки аналізується з точки зору каналів зв'язку, часу відповіді, якості підтримки та покриття різних рівнів проблем. For комерційних рішень оцінюються SLA підтримки, можливості escalation

та доступність експертної допомоги.

Можливості кастомізації інтерфейсу під потреби різних ролей користувачів включають персоналізацію дашбордів, налаштування алертів, створення власних звітів та адаптацію робочих процесів. Досліджуються можливості створення role-based інтерфейсів та делегування адміністративних функцій.

Засоби забезпечення безперервної роботи системи моніторингу включають механізми резервування, failover процедури, backup та recovery можливості. Аналізуються архітектурні рішення для забезпечення високої доступності, процедури відновлення після збоїв та можливості disaster recovery.

Кластеризація та load balancing досліджуються з точки зору їх простоти налаштування, ефективності розподілення навантаження та прозорості для кінцевих користувачів. Важливим аспектом є можливість seamless failover без втрати функціональності моніторингу.

Моніторинг самої системи моніторингу включає можливості самодіагностики, health checks, performance metrics та алерти про стан системи. Цей мета-моніторинг є критично важливим для забезпечення надійності системи моніторингу.

4 РОЗРОБКА ПРОГРАМНИХ ЗАСОБІВ МОНІТОРИНГУ

4.1 Структурна схема ПЗ моніторингу трафіку

Архітектурний аналіз розробленої системи моніторингу мережного трафіка свідчить про її багатокомпонентну структуру, що забезпечує комплексний підхід до збору, обробки та аналізу мережевих даних. Програмний продукт являє собою інтегровану систему, побудовану на принципах модульної архітектури, що дозволяє забезпечити ефективну взаємодію між різними функціональними компонентами.

Центральним елементом користувацької взаємодії з системою виступає модуль графічного інтерфейсу, який реалізує багатовіконну архітектуру для забезпечення зручного доступу до різних аспектів моніторингу мережевого трафіка. Цей модуль включає головне меню навігації, інтерфейс конфігурації загальних параметрів системи, спеціалізовані інтерфейси для аналізу TCP та UDP трафіка, окремий інтерфейс для моніторингу HTTP протоколу, а також додаткові засоби відображення супутньої мережевої інформації.

Фундаментальною основою функціонування системи є перехоплювач пакетів, який забезпечує збір мережевого трафіка в режимі реального часу. Цей компонент реалізує низькорівневі механізми захоплення мережевих пакетів та їх буферизації для подальшої обробки аналітичними модулями системи.

Процес декомпозиції захоплених мережевих даних здійснюється спеціалізованим модулем розбору пакетів, який забезпечує детальний аналіз структури мережевих пакетів та виділення їх складових компонентів. Цей модуль реалізує алгоритми парсингу різних мережевих протоколів та забезпечує структуроване представлення отриманої інформації.

Завершальним елементом архітектури виступає модуль збору інформації для моніторингу якості мережі, який агрегує дані від інших

компонентів системи та формує комплексну картину стану мережевої інфраструктури. Цей модуль реалізує алгоритми обчислення ключових метрик продуктивності мережі та забезпечує їх систематизацію для подальшого аналізу.

Взаємодія між описаними компонентами системи ілюструється схематичним представленням архітектури, наведеним на рисунку 4.1, що демонструє потоки даних та функціональні зв'язки між модулями розробленого програмного рішення.

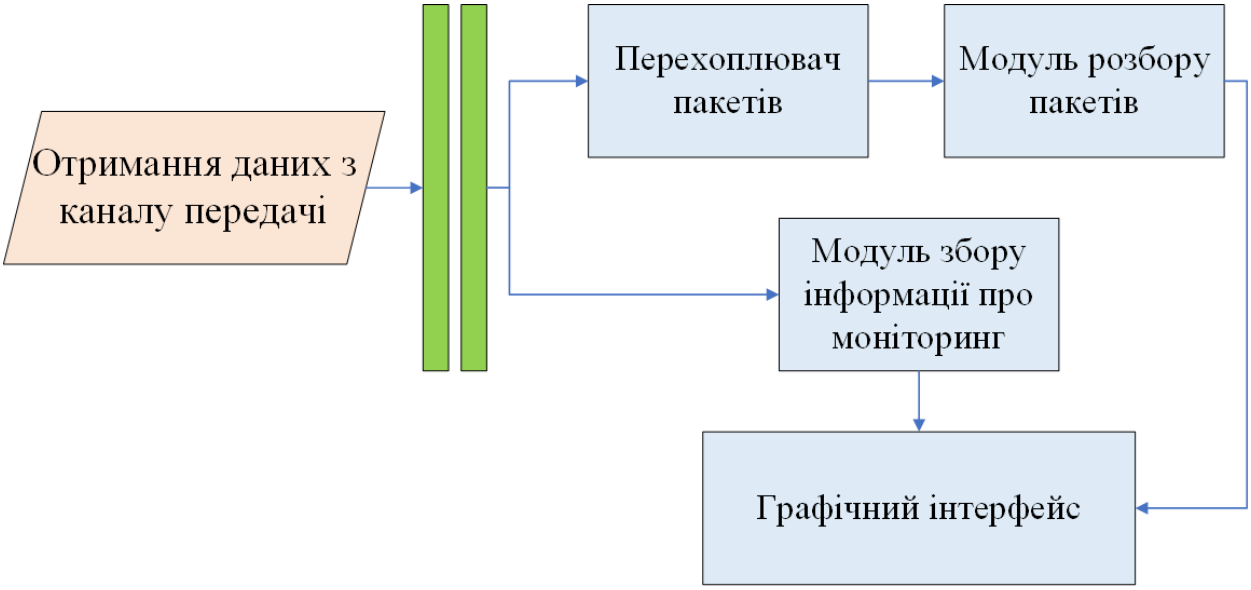

Рисунок 4.1 – Загальна структура системи

Функціональний спектр розробленої системи охоплює комплексний набір операцій з обробки мережевого трафіка, що включає отримання даних безпосередньо з каналу передачі інформації та детальний аналіз структурних компонентів мережевих пакетів. Система забезпечує витяг заголовків на різних рівнях мережевої моделі, включаючи фреймовий рівень, IP-рівень та транспортні протоколи TCP, UDP і ICMP. Процес декодування даних реалізується для незашифрованого контенту, що дозволяє проводити глибокий аналіз змісту мережевих пакетів.

Аналітичні можливості системи включають генерацію ключових

метрик моніторингу мережевої якості, таких як пропускна спроможність каналів зв'язку, інтенсивність пакетного трафіка в одиницю часу, статистика втрачених пакетів та параметри часу життя пакетів. Результати аналізу представляються користувачу через спеціалізовані графічні інтерфейси з можливістю візуалізації певних типів інформації у формі динамічних графіків.

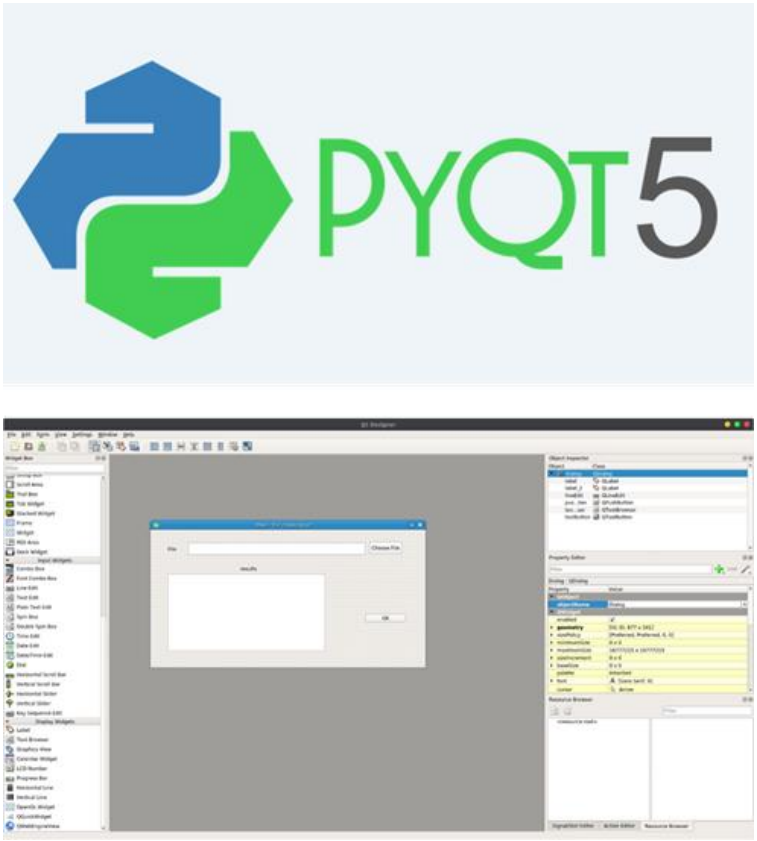

Рисунок 4.2 – Вибір програмних засобів

Архітектурною основою користувацького інтерфейсу виступає фреймворк PyQt5 (рисунок 4.2), який являє собою комплексний набір прив'язок графічного фреймворку Qt до мови програмування Python. Цей інструментарій забезпечує широкий спектр можливостей, включаючи розвинену систему віджетів графічного інтерфейсу, гнучкі механізми стилізації, засоби інтеграції з базами даних SQL та підтримку різноманітних медіаформатів. Інтегроване середовище Qt Designer дозволяє здійснювати крос-платформну розробку дизайну інтерфейсних елементів.

Реалізація графічного модуля базується на використанні ключових компонентів PyQt5, зокрема модуля QtCore, який містить фундаментальні неграфічні класи, систему сигналів та слотів, абстракції для роботи з потоками та спільною пам'яттю, а також модуля QtGui, що забезпечує функціональність графічних компонентів та елементів управління.

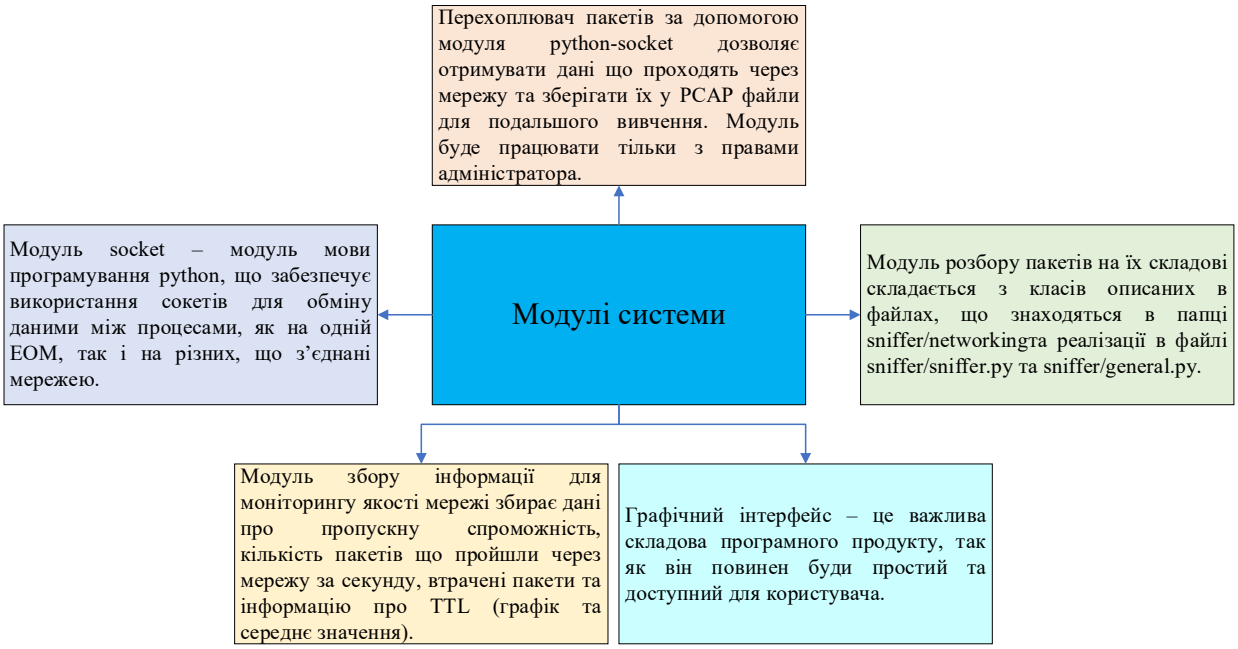

Рисунок 4.3 – Модулі системи

Структурна організація модуля графічного інтерфейсу передбачає розділення на дві основні категорії компонентів. Перша категорія включає імплементації вікон, що представляють собою базові оболонки інтерфейсних елементів, створені засобами фреймворку PyQt5 (рисунок 3.3). До цієї категорії належать головне вікно програми, вікно загальної інформації та спеціалізовані вікна для відображення даних різних мережевих протоколів.

Друга категорія компонентів представлена віджетами вікон, які реалізують повноцінну логіку отримання даних від інших модулів системи та їх відображення в інтерфейсі. Ці компоненти використовують механізми інтервального переривання для періодичного оновлення відображуваної інформації та мьютекси для забезпечення синхронізації доступу до даних, що

можуть змінюватися іншими модулями системи.

Функціональне призначення основних класів системи визначається специфікою оброблюваних даних. Клас головного вікна забезпечує координацію роботи інших віджетів та управління навігацією між різними розділами програми. Спеціалізовані класи відповідають за відображення детальної інформації про конкретні типи мережевого трафіка, включаючи TCP, UDP та HTTP пакети, а також забезпечують візуалізацію відповідних метрик у вигляді графіків часу життя пакетів.

Інтеграція з мережевою підсистемою операційної системи здійснюється через модуль socket мови програмування Python, який забезпечує використання сокетного інтерфейсу для міжпроцесної комунікації як в рамках однієї системи, так і між мережевими вузлами. Модуль збору інформації для моніторингу якості мережі агрегує дані про пропускну спроможність, інтенсивність пакетного потоку, статистику втрачених пакетів та параметри часу життя пакетів з їх подальшою візуалізацією.

Процес розробки графічного інтерфейсу здійснювався з використанням спеціалізованого інструменту PyQt5 Designer, що дозволив створити дизайн інтерфейсу та зберегти його в форматі UI-файлів. Подальша конвертація здійснювалася засобами пакету pyuic5, який трансформує UI-файли в Python-класи, що забезпечують базову структуру інтерфейсних елементів.

Архітектура класів вікон передбачає поєднання візуальних компонентів з програмною логікою взаємодії користувача з інтерфейсом. Правильна ініціалізація віджетів вимагає виклику конструктора базового класу QtWidgets.QMainWindow, створення екземпляра UI-класу та виконання методу налаштування інтерфейсу.

Візуалізація графічної інформації реалізується через компонент QGraphicsView, налаштування якого включає встановлення відповідного вирівнювання, створення графічної сцени та ініціалізацію об'єктів для малювання. Періодичне оновлення текстової інформації забезпечується механізмом QTimer, який дозволяє встановлювати регулярні інтервали

виконання функцій оновлення накопиченої перехоплювачем пакетів інформації.

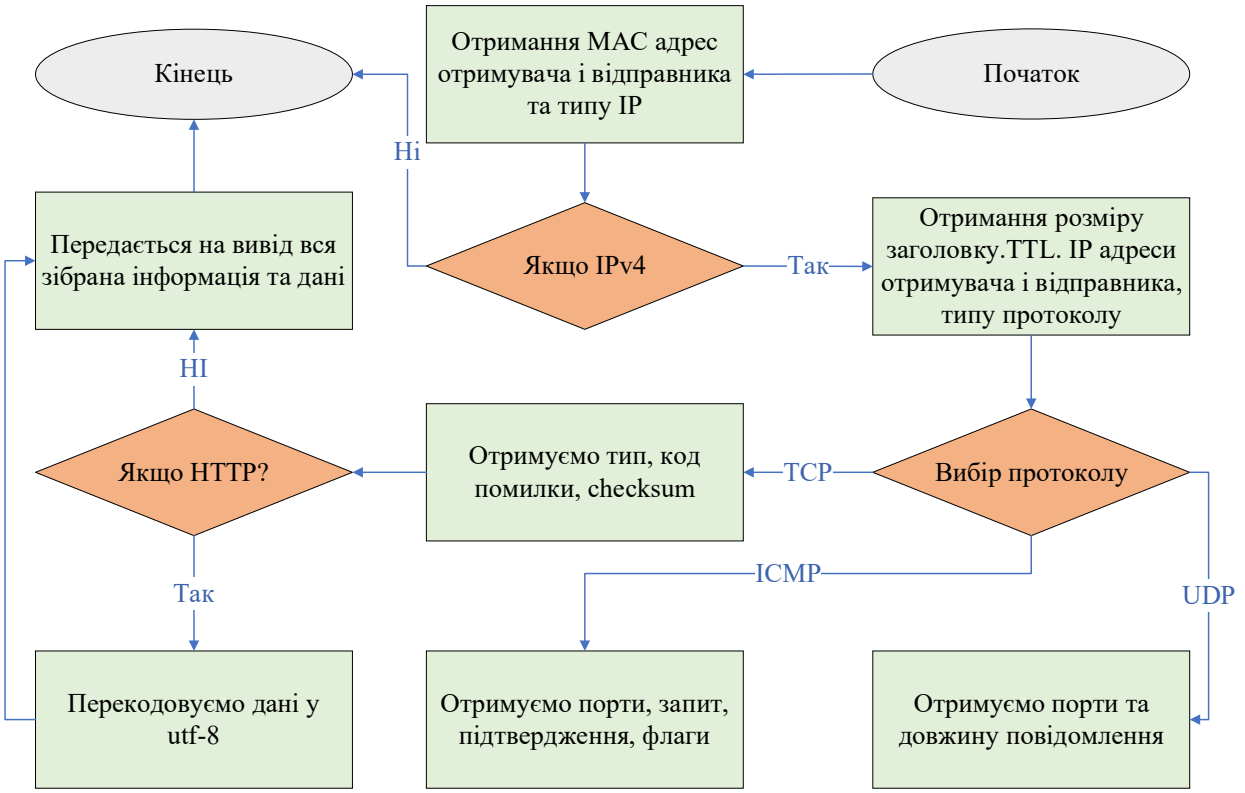

Рисунок 4.4 – Алгоритм модуля розбору пакетів

Модулі перехоплення та розбору пакетів (рисунок 4.4) функціонують в окремому процесі для запобігання блокуванню користувацького інтерфейсу. Перехоплювач отримує дані в первинному форматі та передає їх модулю розбору, який здійснює аналіз заголовків та формує структуровані дані про пакети. Технічна реалізація перехоплювача базується на використанні сокетів типу AF_PACKET, що забезпечує доступ до пакетів через ядро операційної системи з можливістю контролю кожного біта заголовка.

Система моніторингу якості мережі здійснює збір та обробку ключових показників продуктивності, включаючи пропускну спроможність в одиницю часу, максимальні значення пропускної спроможності, інтенсивність пакетного трафіка та статистику втрачених фреймів. Отримання даних про обсяги переданої та отриманої інформації здійснюється через читання

системних файлів операційної системи Linux, що містять статистику мережевих адаптерів. Алгоритм обчислення базується на періодичному зчитуванні значень та обчисленні різниць для визначення поточної пропускної спроможності та оновлення максимальних показників.

4.2 Реалізація ПЗ

Експлуатація розробленої системи моніторингу мережного трафіка вимагає попереднього встановлення інтерпретатора Python версії 3.7.3, який може бути отриманий з офіційного репозиторію розробника. Ця специфічна версія забезпечує сумісність з усіма використаними в проекті бібліотеками та гарантує стабільну роботу всіх компонентів системи.

Ініціалізація програмного комплексу здійснюється через виконання головного виконуваного файлу getPackets.py, що призводить до відображення основного навігаційного інтерфейсу системи, структура якого представлена на рисунку 4.5. Цей інтерфейс служить центральною точкою доступу до всіх функціональних модулів програми та забезпечує інтуїтивну навігацію між різними режимами моніторингу.

Активація основного режиму моніторингу здійснюється через інтерфейсний елемент General, взаємодія з яким призводить до відкриття спеціалізованого робочого вікна, зображеного на рисунку 4.5, та автоматичного запуску процесів збору та аналізу мережевого трафіка. Цей режим забезпечує комплексний моніторинг мережевої активності з відображенням ключових показників продуктивності та якості з'єднання.

Кожен функціональний розділ програми оснащений уніфікованою системою управління, що включає елементи контролю за станом модулів перехоплення та розбору пакетів. Користувач має можливість динамічно керувати процесами збору даних, ініціюючи або припиняючи роботу аналітичних компонентів відповідно до поточних потреб моніторингу.

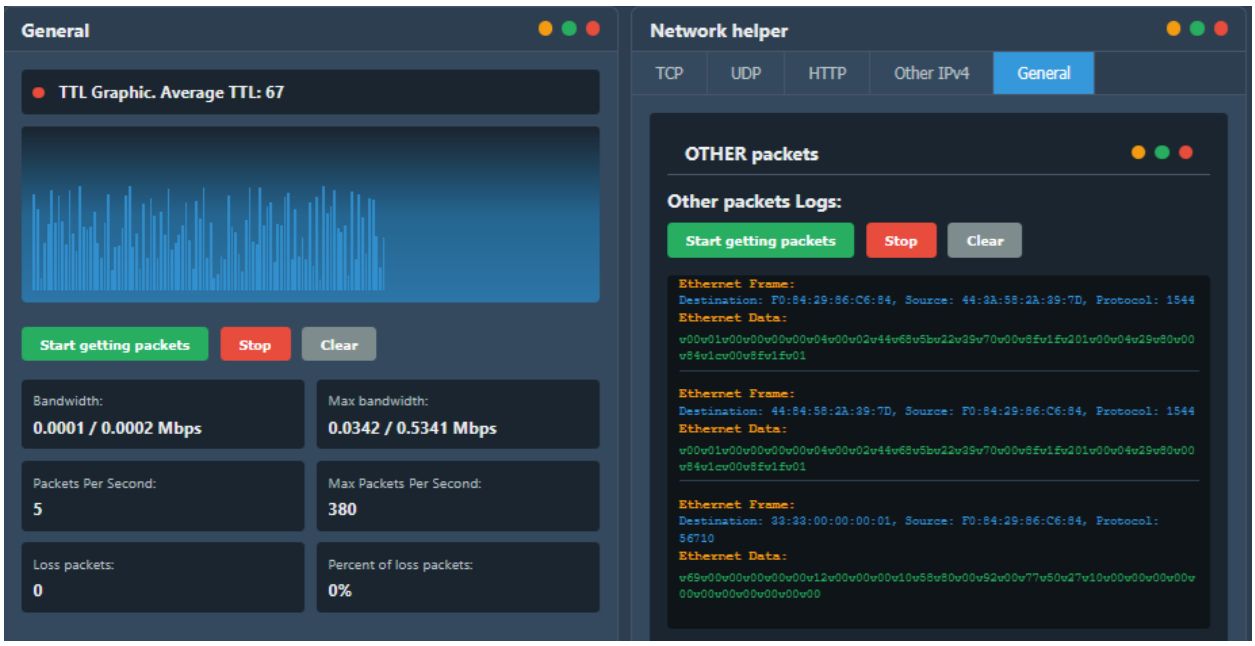

Рисунок 4.5 – Результати роботи

Результати роботи представлені на рисунку 4.5.

Додатковою функціональністю системи є можливість очищення накопичених даних через спеціальний елемент управління Clear, який забезпечує видалення всієї збереженої в пам'яті програми інформації. Ця функція дозволяє користувачу розпочати новий сеанс моніторингу з чистого стану системи, що є особливо корисним при переході до аналізу нових мережевих сценаріїв або при необхідності скидання накопичених статистичних даних.

ВИСНОВКИ

У результаті проведеної роботи було здійснено розробку програмних засобів моніторингу трафіка комп'ютерної мережі, архітектурною основою якої виступає графічний інтерфейс, реалізований засобами фреймворку PyQt5. Створена система являє собою комплексне програмне рішення, що забезпечує ефективний контроль та аналіз мережевого трафіка в режимі реального часу.

Теоретичну основу дослідження складає детальний аналіз сучасних методологій моніторингу мережевого трафіка та підходів до управління якістю мережевих сервісів. Проведено комплексне дослідження існуючих програмних засобів моніторингу трафіка, що застосовуються в корпоративних мережевих інфраструктурах, з метою виявлення їх функціональних можливостей, переваг та обмежень.

Фундаментальною складовою роботи є ретельний аналіз семирівневої моделі OSI як концептуальної основи для розуміння процесів передачі даних у комп'ютерних мережах. Цей аналіз дозволив сформувати цілісне уявлення про взаємодію різних рівнів мережевого стеку та їх вплив на якість передачі інформації.

Особливу увагу приділено дослідженню протоколів, що використовуються для моніторингу мережевого трафіка, включаючи аналіз їх технічних характеристик, сфер застосування та можливостей інтеграції в комплексні системи мережевого моніторингу. Проведено систематизацію та класифікацію основних протокольних рішень відповідно до їх функціонального призначення та технічних особливостей.

Комплексний аналіз параметрів якості обслуговування мережевих сервісів дозволив ідентифікувати ключові метрики, що визначають ефективність функціонування мережевої інфраструктури. Ці параметри становлять основу для формування системи показників якості, що

реалізована в розробленій програмній системі.

Технологічну основу реалізації системи складає обґрунтований вибір інструментальних засобів розробки, що базується на аналізі їх функціональних можливостей, технічних характеристик та сумісності з поставленими завданнями. Детально описано основні метрики моніторингу якості обслуговування комп'ютерної мережі, що забезпечують комплексну оцінку стану мережевої інфраструктури та ефективності її функціонування.

Результатом дослідження є створення функціонального програмного продукту, що демонструє практичну реалізацію теоретичних концепцій мережевого моніторингу та забезпечує ефективні засоби контролю якості мережевих сервісів у корпоративному середовищі.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Stallings, W. SNMP, SNMPv2, SNMPv3, and RMON 1 and 2. 4th edition. Addison-Wesley Professional, 2021. – 672 с.

2. Mauro, D., Schmidt, K. Essential SNMP: Help for System and Network Administrators. 2nd edition. O'Reilly Media, 2020. – 456 с.

3. Perkins, D., McGinnis, E. Understanding SNMP MIBs. Prentice Hall, 2019. – 512 с.

4. Leinwand, A., Conroy, K. Network Management: A Practical Perspective. 3rd edition. Addison-Wesley Professional, 2018. – 624 с.

5. Cisco Systems. Internetwork Troubleshooting Handbook. 2nd edition. Cisco Press, 2019. – 832 с.

6. Rose, M., McCloghrie, K. How to Manage Your Network Using SNMP. Prentice Hall, 2020. – 384 с.

7. Subramanian, M. Network Management: Principles and Practice. 2nd edition. Addison-Wesley Professional, 2018. – 896 с.

8. Morris, S. Network Management and Monitoring. McGraw-Hill Education, 2021. – 752 с.

9. Clemm, A. Network Management Fundamentals. Cisco Press, 2019. – 544 с.

10. Ballew, S. Managing IP Networks with Cisco Routers. O'Reilly Media, 2020. – 416 с.