Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет Кібербезпеки
(повна назва)

Кафедра Інформаційно-мережної інженерії
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

рівень вищої освіти перший (бакалаврський)

Порівняльне дослідження засобів інжинірингу трафіку
(тема)

Виконав:
Здобувач першого року навчання,
групи ТРІМІу-23-1
Бризгалов Максим
(власне ім'я, прізвище)

Спеціальність 172 Електронні комунікації та радіотехніка
(код і повна назва спеціальності)

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

Освітня програма Інформаційно-мережна інженерія
(повна назва освітньої програми)

Керівник *ст.викл.Олександр Малінін*
(посада, власне ім'я, прізвище)

Допускається до захисту

Завідувач кафедри _____ _____ *Микола Москалець*
(підпис) (власне ім'я, прізвище)

2026

Харківський національний університет радіоелектроніки

Факультет Кібербезпеки

Кафедра Інформаційно-мережної інженерії

Рівень вищої освіти перший (бакалаврський)

Спеціальність 172 Електронні комунікації та радіотехніка
(код і повна назва)

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

Освітня програма Інформаційно-мережна інженерія
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____
(підпис)
«25» 05 2026 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

здобувачеві Бризгалову Максиму Олеговичу
(прізвище, ім'я, по батькові)

1. Тема роботи Порівняльне дослідження засобів інжинірингу трафіку
затверджена наказом університету від 25 05 2026 р. № 415Ст
2. Термін подання здобувачем роботи до екзаменаційної комісії 19 06 2026 р.
3. Вихідні дані до роботи *Процеси моніторингу, аналізу та інжинірингу мережного трафіку в сучасних телекомунікаційних системах. Програмне забезпечення, що підлягає дослідженню: Zabbix, Observium, Nagios, Network Olympus, Wireshark, tcpdump, NetworkMiner, Cain and Abel.*

4. Зміст пояснювальної записки (перелік питань, які потрібно розробити)
Вступ
1. Теоретичні основи та методологія інжинірингу і моніторингу мережного трафіку
2. Практичний аналіз та порівняльне дослідження функціональних засобів інжинірингу трафіку
Висновки

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (п.5 включається до завдання за рішенням випускової кафедри) _____
Слайди у форматі Power Point (назва, мета і актуальність кваліфікаційної роботи, вступ, Класифікація сучасних програмних засобів моніторингу та аналізу трафіку, Первинна класифікація та архітектурні характеристики досліджуваних засобів інжинірингу трафіку, Система критеріїв оцінки, Моніторинг інфраструктури (NMS), Консольний та пакетний аналіз (DPI), Спеціалізований аналіз та форензика, Гібридний контур моніторингу, Матриця порівняльного аналізу засобів інжинірингу та моніторингу трафіку, висновки та результати роботи)

6. Консультанти розділів роботи (п.6 включається до завдання 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
Основна частина	ст.викл. Малінін О.П.		22.06.2026

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Строк / термін виконання етапів роботи	Примітка
1	Ознайомлення із завданням. Уточнення ТЗ	25.05-27.05.2026	виконано
2	Підбір літератури за темою роботи	28.05-01.06.2026	виконано
3	Виконання розділу 1	02.06-10.06.2026	виконано
4	Виконання розділу 2	11.06-17.06.2026	виконано
5	Оформлення презентаційного матеріалу	18.06-19.06.2026	виконано
6	Підготовка до захисту у ЕК	20.06-22.06.2026	виконано

Дата видачі завдання 25 05 2026 р.

Здобувач _____
(підпис)

Керівник роботи _____ *ст.викл. Олександр Малінін*
(підпис) (посада, власне ім'я, прізвище)

РЕФЕРАТ

Пояснювальна записка: 80 стор. формату А4, 20 рис., 2 табл., 17 посилань.

Об'єкт дослідження – процеси моніторингу, аналізу та інжинірингу мережевого трафіку в сучасних телекомунікаційних системах.

Мета роботи – комплексний системно-аналітичний аналіз сучасних програмних засобів моніторингу та інжинірингу трафіку, їх класифікація за техніко-економічними критеріями та обгрунтування концепції побудови двоконтурних гібридних систем спостереження для підвищення надійності, безпеки та продуктивності телекомунікаційних мереж.

Проведено системний теоретичний аналіз програм інжинірингу трафіку, деталізовано ключові мережні метрики. Розроблено комплексну систему технічних, архітектурних, економічних критеріїв та критеріїв безпеки для оцінки програмного забезпечення.

ІНЖИНІРИНГ ТРАФІКУ, МОНІТОРІНГ МЕРЕЖІ, ТЕОРІЯ МАСОВОГО ОБСЛУГОВУВАННЯ, СНІФІНГ, ГЛИБОКИЙ АНАЛІЗ ПАКЕТІВ (DPI), ГІБРИДНИЙ КОНТУР, ZABBIX, WIRESHARK.

ABSTRACT

Explanatory note: 80 pp. format A4, 20 Fig., 17 reference, 2 tab.

The object of the research is processes for monitoring, analyzing, and engineering network traffic in modern telecommunications systems.

The purpose of the work is a comprehensive systematic and analytical review of modern software tools for traffic monitoring and engineering, their classification based on technical and economic criteria, and a justification of the concept for building dual-loop hybrid monitoring systems to enhance the reliability, security, and performance of telecommunications networks.

A systematic theoretical analysis of traffic engineering software has been conducted, and key network metrics have been detailed. A comprehensive system of technical, architectural, economic, and security criteria for evaluating software has been developed.

TRAFFIC ENGINEERING, NETWORK MONITORING, QUEUEING THEORY, SNIFFING, DEEP PACKET INSPECTION (DPI), HYBRID CIRCUIT, ZABBIX, WIRESHARK.

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ..... 9
ВСТУП..... 12
1 ТЕОРЕТИЧНІ ОСНОВИ ТА МЕТОДОЛОГІЯ ІНЖИНІРИНГУ І МОНІТОРИНГУ МЕРЕЖНОГО ТРАФІКУ..... 14
1.1 Еволюція, сутність та завдання інжинірингу мережного трафіку..... 14
1.2 Класифікація сучасних програмних засобів моніторингу та аналізу трафіку 15
1.2.1 Системи управління мережею (Network Management Systems, NMS)..... 16
1.2.2 Аналізатори мережних пакетів (Сніфери / Packet Analyzers).....16
1.2.3 Засоби мережної форензики та аудиту безпеки.....17
1.3 Архітектурні методи, протоколи та технології збору мережних даних..... 19
1.3.1 Протокольна взаємодія на основі SNMP та WMI..... 19
1.3.2 Агентний та безагентний (Agentless) моніторинг..... 21
1.3.3 Технології перехоплення та аналізу сирого трафіку (Packet Capture).....21
1.4 Математичні та теоретичні основи моделювання мережних потоків.....23
1.4.1 Метрики якості обслуговування (QoS / QoE)..... 23
1.4.2 Застосування теорії масового обслуговування та моделей розподілу потоків.....24
1.5 Проблематика безпеки, аномалій та несанкціонованого аналізу в мережному інжинірингу.....26

1.5.1 Класифікація мережних аномалій та деструктивних потоків даних.....26
1.5.2 Легітимний моніторинг проти несанкціонованого перехоплення (Сніфінгу).....27
1.6 Постановка задачі порівняльного дослідження та критерії оцінювання.....28
ВИСНОВКИ ДО РОЗДІЛУ 1..... 31
2 ПРАКТИЧНИЙ АНАЛІЗ ТА ПОРІВНЯЛЬНЕ ДОСЛІДЖЕННЯ ФУНКЦІОНАЛЬНИХ ЗАСОБІВ ІНЖИНІРИНГУ ТРАФІКУ.....33
2.1 Аналіз систем комплексного моніторингу мережної інфраструктури (NMS)..... 33
2.1.1 Програмний комплекс Observium.....33
2.1.2 Програмний комплекс Nagios..... 36
2.1.3 Консольний аналізатор мережних пакетів tcpdump.....38
2.1.4 Засіб мережної криміналістики та форензики NetworkMiner.....43
2.1.5 Аналізатор мережних пакетів Wireshark.....47
2.1.6 Багатофункціональний комплекс аудиту безпеки Cain and Abel.50
2.1.7 Розподілена система інфраструктурного моніторингу Zabbix.....53
2.1.8 Автоматизована система моніторингу Network Olympus.....57
2.2 Концепція побудови гібридних контурів моніторингу телекомунікаційних систем 60
2.2.1 Перший контур: Макроскопічний інфраструктурний моніторинг.....61
2.2.2 Другий контур: Мікроскопічний аналіз та пакетна діагностика.62
2.2.3 Синергетичний ефект та техніко-економічне обгрунтування

гібридизації.....63
2.3 Комплексна порівняльна оцінка досліджуваних засобів інжинірингу трафіку.....64
ВИСНОВКИ ДО РОЗДІЛУ 2..... 66
ВИСНОВКИ..... 68
ПЕРЕЛІК ПОСИЛАНЬ..... 71
ДОДАТОК А. Слайди презентації.....73

ПЕРЕЛІК СКОРОЧЕНЬ

API (Application Programming Interface) – інтерфейс прикладного програмування;

APR (ARP Poison Routing) – компрометація таблиці маршрутизації протоколу ARP (ARP-спофінг);

ARP (Address Resolution Protocol) – протокол визначення мережної адреси за фізичною;

BPF (Berkeley Packet Filter) – механізм фільтрації пакетів Берклі;

CLI (Command Line Interface) – консольний інтерфейс (інтерфейс командного рядка);

CPU (Central Processing Unit) – центральний процесор;

DHCP (Dynamic Host Configuration Protocol) – протокол динамічного налаштування вузла;

DPI (Deep Packet Inspection) – технологія глибокого аналізу пакетів;

FTP (File Transfer Protocol) – протокол передачі файлів;

GUI (Graphical User Interface) – графічний інтерфейс користувача;

HTTP (HyperText Transfer Protocol) – протокол передачі гіпертексту;

HTTPS (HyperText Transfer Protocol Secure) – захищений протокол передачі гіпертексту;

ICMP (Internet Control Message Protocol) – протокол міжмережних керуючих повідомлень;

IP (Internet Protocol) – міжмережний протокол;

IPsec (IP Security) – набір протоколів для забезпечення безпеки протоколу IP;

MAC (Media Access Control) – унікальний фізичний ідентифікатор мережевого пристрою;

MITM (Man-in-the-Middle) – тип кібератаки «Людина посередині»;

NMS (Network Management System) – система управління та моніторингу мережі;

OSI (Open Systems Interconnection) – базова еталонна модель взаємодії відкритих систем;

PCAP (Packet Capture) – стандартизований прикладний інтерфейс та формат файлів для фіксації мережевих пакетів;

QoS (Quality of Service) – якість мережного обслуговування;

RAM (Random Access Memory) – оперативна пам'ять (оперативний запам'ятовувальний пристрій) ;

SMB (Server Message Block) – мережний протокол прикладного рівня для спільного доступу до файлів;

SNMP (Simple Network Management Protocol) – простий протокол управління мережею;

SSH (Secure Shell) – мережний протокол сеансового рівня для безпечного віддаленого керування;

STP (Spanning Tree Protocol) – мережний протокол канального рівня для усунення петель у топології;

TCP (Transmission Control Protocol) – протокол керування передачею даних із гарантією доставки;

TE (Traffic Engineering) – інжинірингу трафіку;

TLS/SSL (Transport Layer Security / Secure Sockets Layer) – криптографічні протоколи безпеки транспортного рівня;

UDP (User Datagram Protocol) – протокол користувацьких датаграм без встановлення з'єднання;

VoIP (Voice over IP) – технологія передачі голосу через протокол IP;

WLAN (Wireless Local Area Network) – бездротова локальна комп'ютерна мережа;

WMI (Windows Management Instrumentation) – інструментарій керування Windows;

ІТ – інформаційні технології;

ІТС – інформаційно-телекомунікаційна система;

ТМО – теорія масового обслуговування.

ВСТУП

Розвиток сучасних інформаційно-комунікаційних технологій (ІКТ) характеризується експоненційним зростанням обсягів даних, що передаються через глобальні та локальні мережі. Сучасні корпоративні та відомчі комп'ютерні мережі перетворилися на складні гетерогенні інфраструктури, які об'єднують тисячі пристроїв, різноманітні операційні системи, хмарні сервіси та концепції інтернет-речей (IoT). За таких умов забезпечення стабільності, доступності, продуктивності та безпеки мережної архітектури стає першочерговим завданням для профільних фахівців – системних адміністраторів та інженерів з кібербезпеки.

Інжиніринг трафіку (Traffic Engineering, TE) є фундаментальною складовою управління сучасними телекомунікаційними системами. Він охоплює сукупність методів, алгоритмів та програмно-апаратних засобів, спрямованих на оптимізацію продуктивності робочих мереж. Головна мета інжинірингу трафіку полягає в забезпеченні динамічного та раціонального розподілу інформаційних потоків усередині мережі з метою мінімізації затримок, уникнення перевантаження окремих каналів зв'язку та максимізації загальної пропускної здатності.

Проте успішна реалізація концепцій інжинірингу трафіку неможлива без наявності об'єктивних, точних та оперативних даних про поточний стан мережного середовища. Саме цю функцію виконують програмні засоби моніторингу та аналізу трафіку. Вони виступають у ролі «органів чуття» для мережної інфраструктури, дозволяючи в реальному часі або ретроспективно фіксувати будь-які зміни, оцінювати метрики якості обслуговування (QoS), виявляти технічні несправності та оперативно реагувати на аномальну діяльність, викликану внутрішніми збоями або зовнішніми кібератаками.

Актуальність детального дослідження та порівняльного аналізу засобів інжинірингу і моніторингу зумовлена тим, що на сучасному ринку програмного забезпечення (як комерційного, так і з відкритим вихідним кодом – Open Source) представлена велика кількість інструментів. Вони суттєво відрізняються за своєю архітектурою, методами збору даних, рівнем деталізації аналізу та системними вимогами. Відтак, вибір оптимального інструментарію під конкретні задачі підприємства чи специфіку мережі є складною багатокритеріальною задачею, яка потребує глибокого теоретичного обгрунтування та систематизації.

1 ТЕОРЕТИЧНІ ОСНОВИ ТА МЕТОДОЛОГІЯ ІНЖИНІРИНГУ І МОНІТОРИНГУ МЕРЕЖНОГО ТРАФІКУ

1.1 Еволюція, сутність та завдання інжинірингу мережного трафіку

Історично концепція управління мережами базувалася на класичних протоколах маршрутизації (таких як OSPF або RIP), які використовують статичні метрики для визначення найкоротшого шляху (Shortest Path First). Проте практика показала, що такий підхід є неефективним у масштабованих мережах із високою динамікою навантаження. Маршрутизація за найкоротшим шляхом часто призводить до ситуації, коли окремі (центральні) канали мережі виявляються критично перевантаженими, тоді як альтернативні маршрути залишаються незадіяними. Це явище отримало назву «незбалансованого розподілу трафіку».

Для вирішення цієї проблеми наприкінці 1990-х років почала формуватися концепція інжинірингу трафіку (TE). На відміну від традиційної маршрутизації, інжиніринг трафіку розглядає мережу не просто як сукупність вузлів та ліній зв'язку, а як динамічну систему розподілу ресурсів, де потоки даних орієнтовані на задоволення конкретних вимог щодо якості (Performance-Oriented) [1].

Основними завданнями, які вирішує інжиніринг трафіку у поєднанні із системами моніторингу, є:

– Управління місткістю (Capacity Management): оптимальне розміщення інформаційних потоків у межах існуючої топології мережі з метою недопущення ситуацій, коли обсяг трафіку перевищує фізичну пропускну здатність інтерфейсів.

– Мінімізація затримок та пакетних втрат (Delay and Loss Minimization): забезпечення пріоритетності для трафіку, чутливого до затримок (наприклад, VoIP, відеоконференції, потокові трансляції в реальному часі).

– Забезпечення відмовостійкості (Reliability and Resilience): автоматичне перенаправлення трафіку на резервні маршрути у разі виходу з ладу окремих ліній зв'язку або маршрутизаторів (Fast Reroute).

– Економічна оптимізація: Максимально ефективне використання вже наявної дороговартісної інфраструктури, що дозволяє відтермінувати витрати на закупівлю нового обладнання.

Ефективне управління трафіком базується на циклічному процесі, який складається з трьох ключових етапів:

– Вимірювання та моніторинг (Measurement): збір первинних даних про стан мережі, обсяги переданих пакетів, завантаженість інтерфейсів та структуру трафіку.

– Аналіз та моделювання (Analysis): обробка зібраної інформації, виявлення трендів, прогнозування поведінки мережі та ідентифікація «вузьких місць» (bottlenecks).

– Оптимізація та модифікація (Control): зміна параметрів маршрутизації, конфігурація політик QoS, перерозподіл смуги пропускання на основі результатів аналізу [1].

Програмні засоби, що аналізуються в даній роботі, покривають перші два етапи цього циклу, забезпечуючи аналітичний фундамент для прийняття рішень щодо реконфігурації мережі.

1.2 Класифікація сучасних програмних засобів моніторингу та аналізу трафіку

Для систематизації інструментарію, що розглядається у кваліфікаційній роботі, необхідно розробити комплексну класифікацію. Залежно від функціонального призначення, архітектури побудови та методів взаємодії з

мережною інфраструктурою, всі програмні засоби можна розділити на кілька функціональних класів [2].

1.2.1 Системи управління мережею (Network Management Systems, NMS)

Цей клас програмного забезпечення призначений для глобального, стратегічного контролю стану всієї IT-інфраструктури. Системи NMS фокусуються на зборі агрегованих метрик продуктивності та доступності пристроїв (серверів, комутаторів, маршрутизаторів). Вони зазвичай працюють за архітектурою «клієнт-сервер» або «менеджер-агент».

Головними інструментами збору даних тут виступають протоколи SNMP (Simple Network Management Protocol), ICMP (Internet Control Message Protocol), а також технології WMI (Windows Management Instrumentation) та SSH для Unix-систем. Системи NMS не аналізують кожен окремий пакет даних, але вони надають чітку картину утилізації ресурсів у часі, будують графіки навантаження та генерують сповіщення (alerts) у разі виходу параметрів за встановлені межі [2].

До цього класу серед досліджуваних у роботі засобів належать Zabbix, Nagios, Observium та Network Olympus. Вони дозволяють автоматизувати рутинну діяльність адміністратора шляхом створення централізованих панелей керування (dashboards), де відображається здоров'я всієї мережі підприємства.

1.2.2 Аналізатори мережних пакетів (Сніфери / Packet Analyzers)

Аналізатори пакетів (сніфери) представляють інший рівень дослідження мережі – тактичний або мікроскопічний. На відміну від NMS, які бачать картину «зверху», сніфери працюють безпосередньо на рівні мережних інтерфейсів,

перехоплюючи та декодуючи кожен біт інформації, що передається по кабелю або через бездротове середовище.

В основі роботи цих засобів лежить переведення мережної карти в так званий «змішаний режим» (promiscuous mode), що дозволяє пристрою приймати всі пакети, незалежно від того, кому вони адресовані в межах одного домену колізій чи широкомовного домену. Сніфери використовують технологію глибокого аналізу пакетів (Deep Packet Inspection, DPI), розбираючи заголовки протоколів усіх рівнів моделі OSI (від канального до прикладного) [3].

У даній роботі цей напрям представлений утилітами Wireshark та tcpdump. Вони є незамінними для локалізації складних плаваючих помилок, аналізу нестандартних протоколів, розбору логіки роботи мережних додатків та проведення детального інженерного аналізу трафіку під час збоїв.

1.2.3 Засоби мережної форензики та аудиту безпеки

Окрему нішу займають інструменти, орієнтовані на інформаційну безпеку та ретроспективний аналіз інцидентів (Network Forensics). На відміну від класичних сніферів, які просто відображають структуру пакетів, засоби форензики націлені на агрегацію даних та вилучення з них корисних артефактів. Вони здатні «збирати» з сирого потоку пакетів реальні файли, зображення, електронні листи, сесії чатів, а також ідентифікувати облікові дані (логіни та паролі), що передаються в незашифрованому вигляді.

Крім того, деякі засоби цього класу мають функціонал для активного аудиту, що включає перевірку стійкості мережі до атак типу MITM (Man-in-the-Middle), сканування вразливостей, тестування бездротових мереж та підбір паролів.

У даному дослідженні до цієї категорії відносяться NetworkMiner (чистий пасивний інструмент форензики) та Cain and Abel (багатофункціональна утиліта для аудиту безпеки та відновлення паролів) [4].

Для наочності та структурування теоретичного матеріалу зведено базову класифікацію досліджуваних засобів у порівняльну таблицю.

Таблиця 1.1. Первинна класифікація та архітектурні характеристики досліджуваних засобів інжинірингу трафіку

Програмний засіб	Функціональний клас	Основний метод збору даних	Рівень деталізації аналізу	Спрямованість використання
Zabbix	Засіб аудиту безпеки та відновлення	Активний/Пасивний (SNMP, Агенти, ICMP, WMI)	Макрорівень (метрики, Графіки, Статус пристроїв)	Моніторинг інфраструктури, бізнес-моніторинг
Nagios	Моніторинг стану та сервісів (NMS)	Активний (опитування через плагіни, NRPE)	Макрорівень (стан сервісів: Up/Down/Critical)	Контроль доступності сервісів та хостів
Observium	Автоматизований моніторинг (NMS)	Пасивний/Активний (Автовиявлення через SNMP)	Макрорівень (утилізація портів, мережна топологія)	Моніторинг мережного обладнання (Cisco, Juniper тощо)
Network Olympus	Моніторинг малого/середнього бізнесу	Активний (WMI, SNMP, ICMP без агентів)	Макрорівень (логи, сценарії автоматизації)	Управління IT-інфраструктурою ОС Windows/Linux
Wireshark	Аналізатор пакетів (сніфер)	Пасивний (перехоплення через libpcap/Npcap)	Мікрорівень (повний розбір структури пакетів)	Глибокий траблшутінг, аналіз протоколів
tcpdump	Консольний аналізатор пакетів	Пасивний (перехоплення через libpcap)	Мікрорівень (текстовий дамп заголовків та тіла пакетів)	Швидкий аналіз на серверах без графічного інтерфейсу

Продовження таблиці 1.1

Network Miner	Інструмент мережної криміналістики	Пасивний (аналіз файлів PCAP або дзеркального трафіку)	Рівень артефактів (вилучення файлів, хостів, сертифікатів)	Мережна форензика, розслідування інцидентів
Cain and Abel	Засіб аудиту безпеки та відновлення	Активний/Пасивний (ARP-spoofing, сніфінг)	Рівень безпеки (перехоплення паролів, дешифрація хешів)	Аудит захищеності, тестування на проникнення

1.3 Архітектурні методи, протоколи та технології збору мережних даних

Для детального розуміння принципів функціонування систем інжинірингу трафіку необхідно дослідити механізми, за допомогою яких програмне забезпечення отримує первинну інформацію з мережі. Вибір конкретного методу збору даних безпосередньо впливає на навантаження на обчислювальні ресурси досліджуваних об'єктів та пропускну здатність каналів зв'язку. У сучасних телекомунікаційних системах архітектура збору даних поділяється на два фундаментальні напрямки: опитування метрик (інфраструктурний моніторинг) та пряме перехоплення пакетів (аналіз потоків даних) [4].

1.3.1 Протокольна взаємодія на основі SNMP та WMI

Для систем класу NMS (Zabbix, Nagios, Observium, Network Olympus) базовим інструментом збору макростатистики є сімейство протоколів прикладного рівня моделі OSI.

Simple Network Management Protocol (SNMP) є галузевим стандартом для моніторингу мережних пристроїв. Архітектура SNMP базується на концепції «менеджер-агент». Агент – це програмний модуль, вбудований в операційну

систему маршрутизатора, комутатора або сервера. Менеджер (сервер моніторингу) періодично надсилає запити агенту для отримання значень конкретних параметрів.

Вся інформація про пристрій структурується у вигляді бази керованої інформації (MIB – Management Information Base), яка має ієрархічну деревоподібну структуру. Кожен об'єкт у цьому дереві ідентифікується унікальним цифровим кодом – OID (Object Identifier). Наприклад, OID для отримання системного часу пристрою або кількості байт, що пройшли через конкретний порт комутатора, чітко регламентовані стандартами RFC [2,3].

У дослідженні важливо враховувати еволюцію та безпеку версій SNMP:

– SNMPv1 та SNMPv2c: Використовують аутентифікацію за допомогою текстового рядка (Community String), який передається через мережу у відкритому (незашифрованому) вигляді. Це створює серйозні вразливості, оскільки зловмисник за допомогою сніферів (Wireshark чи tcpdump) може перехопити цей рядок і отримати доступ до конфігурації пристрою.

– SNMPv3: Сучасна версія, що підтримує криптографічний захист. Вона забезпечує аутентифікацію користувачів (моделі USM) та шифрування пакетів за алгоритмами AES/DES, що робить її безпечною для використання у відкритих або корпоративних мережах з високими вимогами до захисту інформації.

Windows Management Instrumentation (WMI) – це альтернативна пропрієтарна технологія від компанії Microsoft, яка активно використовується такими системами, як Network Olympus та Zabbix, для моніторингу серверної інфраструктури на базі ОС Windows. WMI надає значно ширший спектр даних про внутрішній стан операційної системи, ніж стандартний SNMP: від списку запущених процесів та служб до детальної статистики використання дискових масивів та логів подій (Event Logs). Проте WMI створює вище навантаження на процесор та використовує складнішу систему динамічних портів RPC, що потребує додаткового налаштування міжмережних екранів.

1.3.2 Агентний та безагентний (Agentless) моніторинг

При розгортанні комплексних систем моніторингу архітектурний вибір між агентним та безагентним підходами є критично важливим для оптимізації трафіку інжинірингу.

– Агентний моніторинг (на прикладі Zabbix Agent): Передбачає встановлення спеціалізованого програмного забезпечення безпосередньо на цільовий об'єкт (сервер). Агент працює локально, збирає метрики системи з мінімальними затримками і може функціонувати у двох режимах: пасивний режим (сервер моніторингу підключається до агента і запитує дані), активний режим (фгент самостійно збирає метрики та за визначеним графіком відправляє їх на сервер. Це суттєво знижує навантаження на сервер моніторингу та дозволяє обходити обмеження NAT та міжмережних екранів).

– Безагентний моніторинг (Agentless – Nagios, Network Olympus): збір даних здійснюється за допомогою стандартних системних протоколів (ICMP, SSH, Telnet, WMI, SNMP). Цей підхід не потребує встановлення стороннього ПЗ на кінцеві хости, що спрощує адміністрування, проте обмежує глибину збору специфічних метрик операційної системи та збільшує мережний трафік за рахунок постійного опитування (polling) з боку сервера [4].

1.3.3 Технології перехоплення та аналізу сирого трафіку (Packet Capture)

Для засобів мікрорівня (Wireshark, tcpdump, NetworkMiner), завдання полягає не в опитуванні лічильників, а в отриманні копії реальних бінарних даних, що передаються по фізичних чи віртуальних каналах зв'язку.

Основу архітектури перехоплення пакетів на рівні операційної системи складають спеціалізовані бібліотеки драйверів: libpcap (для UNIX-подібних систем)

та Npcap/WinPcap (для ОС Windows). Коли мережна карта працює у звичайному режимі, її апаратний контролер фільтрує пакети на канальному рівні моделі OSI: він приймає лише ті фрейми, MAC-адреса призначення яких збігається з MAC-адресою цієї карти, або є широкомовними (Broadcast).

Для повного аналізу трафіку бібліотеки перехоплення переводять мережний адаптер у змішаний режим (Promiscuous mode). У цьому режимі фільтрація за MAC-адресою вимикається, і всі фрейми, які фізично досягають інтерфейсу, копіюються в буфер операційної системи, звідки їх зчитує аналізатор (наприклад, tcpdump).

У масштабах комутованої мережі (де використовуються комутатори – Switches) простого переведення карти в змішаний режим недостатньо, оскільки комутатор спрямовує трафік лише на той порт, де знаходиться адресат. Для вирішення цієї проблеми та забезпечення легітимного інжинірингу трафіку застосовують три основні методи [4]:

– Дзеркалювання портів (SPAN — Switched Port Analyzer): функція комутатора, яка дозволяє копіювати весь трафік з одного або кількох портів (Ports-джерела) та перенаправляти цю копію на виділений порт (Port-призначення), до якого підключено станцію моніторингу з Wireshark або Zabbix. Існує також технологія RSPAN (Remote SPAN), яка дозволяє передавати копію трафіку через декілька комутаторів у межах однієї віртуальної мережі (VLAN).

– Мережні TAP-пристрої (Test Access Point): Апаратні розгалужувачі, що інтегруються безпосередньо в розрив кабелю зв'язку. Вони забезпечують постійне копіювання сигналу на фізичному рівні. Перевага TAP полягає в тому, що вони не навантажують процесор комутатора (на відміну від SPAN) і гарантують 100% перехоплення трафіку навіть за умови критичного перевантаження каналу.

– ARP-Spoofing (Перехоплення шляхом атаки): Метод, який використовується утилітами для аудиту безпеки (наприклад, Cain and Abel). Він базується на недоліках безпеки протоколу ARP (Address Resolution Protocol).

Програма надсилає неправдиві ARP-відповіді в локальну мережу, переконуючи шлюз та цільовий комп'ютер у тому, що її MAC-адреса належить легітимному одержувачу. В результаті весь трафік між ними починає йти через комп'ютер зловмисника чи аудитора (атака Man-in-the-Middle). Цей метод є активним, деструктивним і використовується виключно під час тестування мережі на проникнення або аналізу її захищеності.

1.4 Математичні та теоретичні основи моделювання мережних потоків

Ефективний інжиніринг трафіку неможливий без математичного апарату, який дозволяє формалізувати процеси передачі даних, оцінити ймовірність втрати пакетів та розрахувати необхідну пропускну здатність. Програмні аналізатори надають сирі метрики, які згодом інтерпретуються через призму фундаментальних мережних моделей [4].

1.4.1 Метрики якості обслуговування (QoS / QoE)

В інжинірингу трафіку ключовими критеріями оцінки ефективності мережі є параметри QoS (Quality of Service). Системи моніторингу безперервно вимірюють наступні математичні величини:

– Пропускна здатність (Throughput): Кількість інформації, що успішно передається через мережу за одиницю часу. Вимірюється в бітах за секунду (bps), мегабітах ($Mbps$) тощо. На відміну від теоретичної місткості каналу (Bandwidth), Throughput враховує накладні витрати протоколів та пакетні втрати.

– Затримка (Latency / Network Delay): Час, необхідний для проходження пакета від джерела до призначення. Вона складається з кількох компонентів:

$$\text{Delay} = d_{\text{proc}} + d_{\text{queue}} + d_{\text{trans}} + d_{\text{prop}}$$ (1.1)

де d_{proc} – затримка обробки (маршрутизатором), d_{queue} – затримка в черзі,
d_{trans} – затримка передачі (залежить від швидкості інтерфейсу), d_{prop} – затримка поширення сигналу у фізичному середовищі (обмежена швидкістю світла).

– Джитер (Jitter): Статистичне коливання затримки між послідовно отриманими пакетами. Математично джитер визначається як середнє абсолютне відхилення затримки:

$$\text{Jitter} = E[|D_i - D_{i-1}|]$$ (1.2)

де D_i – затримка i-го пакета. Високий рівень джитера є критичним для мультимедійного трафіку (VoIP, відеопотоки), що фіксується аналізаторами типу Wireshark через аналіз RTP-потоків.

– Коефіцієнт втрати пакетів (Packet Loss Rate): Відношення кількості втрачених пакетів до загальної кількості відправлених. Виникає через переповнення буферів комутуючих пристроїв під час пікових навантажень [1-3].

1.4.2 Застосування теорії масового обслуговування та моделей розподілу потоків

Для того щоб засоби інжинірингу трафіку (наприклад, графіки завантаження у Zabbix або аналітичні звіти Observium) могли прогнозувати поведінку мережі, використовується апарат теорії масового обслуговування (ТМО). У межах цієї

теорії будь-який мережний пристрій (маршрутизатор чи комутатор) розглядається як система масового обслуговування (СМО), де:

Вхідний потік – це пакети даних, що надходять на вхідний інтерфейс пристрою;

Черга – це буферна пам'ять (пакетний буфер) пристрою, де пакети очікують на обробку, якщо швидкість надходження вища за швидкість передачі;

Канал обслуговування – це обчислювальний процесор пристрою або вихідний інтерфейс із певною пропускною здатністю.

Класичною моделлю, що описує поведінку найпростішого мережевого трафіку, є модель $M/M/1$ за нотацією Кендалла. Вона передбачає, що вхідний потік пакетів є Пуассонівським (має розподіл Пуассона), час обслуговування пакетів розподілений за експоненціальним законом, а в системі наявний один канал обслуговування з нескінченною чергою [5,6].

Інтенсивність завантаження каналу зв'язку (ρ) визначається як відношення інтенсивності надходження пакетів (λ) до інтенсивності їх обслуговування (μ):

$$\rho = \frac{\lambda}{\mu}$$

Якщо коефіцієнт завантаження $\rho \ge 1$, черга починає зростати до нескінченності, що в реальних пристроях призводить до переповнення буфера та миттєвого зростання метрики Packet Loss Rate (коефіцієнту втрати пакетів). Середня кількість пакетів у системі (N) та середній час перебування пакета в мережі (T) відповідно до моделі $M/M/1$ розраховуються за формулами:

$$N = \frac{\rho}{1 - \rho}, \quad T = \frac{1}{\mu - \lambda}$$

Проте сучасні дослідження в галузі інжинірингу трафіку доводять, що реальний телекомунікаційний трафік у корпоративних мережах не є найпростішим (Пуассонівським). Йому притаманна властивість самоподібності (фрактальності). Це означає, що трафік зберігає свій імпульсний, нерівномірний характер (наявність

«вибухових» сплесків – traffic bursts) незалежно від масштабів часу (секунди, хвилини, години).

Для математичного опису такого самоподібного трафіку використовується коефіцієнт Херста (H), який змінюється в межах $0.5 < H < 1$. Чим ближче значення H до одиниці, тим вищим є рівень самоподібності трафіку та тим складніше передбачити поведінку мережі класичними методами ТМО. Саме тому виникає гостра потреба у безперервному пасивному моніторингу за допомогою систем типу Wireshark та tcpdump, які дозволяють зафіксувати реальний профіль навантаження та виявити приховані фрактальні сплески, що здатні паралізувати роботу мережевих вузлів [7].

1.5 Проблематика безпеки, аномалій та несанкціонованого аналізу в мережному інжинірингу

Інжиніринг трафіку тісно пов'язаний із забезпеченням інформаційної безпеки. З одного боку, засоби моніторингу є головним інструментом адміністратора для виявлення загрози, з іншого – самі ці засоби можуть стати об'єктом атаки або інструментом зловмисника.

1.5.1 Класифікація мережних аномалій та деструктивних потоків даних

Під час експлуатації телекомунікаційних систем інженери постійно стикаються з відхиленнями від нормального профілю трафіку (базової лінії — Baseline). Усі аномалії, які фіксуються програмними засобами, можна розділити на дві великі категорії:

– Технічні (ненавмисні) аномалії:

Мережеві петлі (Routing/Broadcasting Loops): Виникають через помилки конфігурації протоколів динамічної маршрутизації або протоколу STP (Spanning

Tree Protocol). Приводять до виникнення «широкомовного шторму», коли пакети нескінченно циркулюють у мережі, повністю утилізуючи смугу пропускання. Такі аномалії миттєво фіксуються в Zabbix або Observium через стрибок завантаження процесорів комутаторів до 100%.

Апаратні збої: Вихід з ладу мережевого інтерфейсу або пошкодження кабелю, що генерує велику кількість помилкових фреймів (FCS Errors, Alignment Errors), які детально детектуються в Wireshark.

– Навмисні (зловмисні) аномалії:

Атаки на відмову в обслуговуванні (DoS/DDoS): Спрямовані на виснаження ресурсів сервера або каналу зв'язку. Це може бути як банальне затоплення сміттєвим трафіком (SYN Flood, UDP Flood), так і складні атаки прикладного рівня (HTTP Flood).

Сканування та розвідка (Network Scanning): Масове опитування портів та IP-адрес для пошуку вразливостей. Програми моніторингу бачать це як різке зростання кількості унікальних сесій з однієї IP-адреси [4].

1.5.2 Легітимний моніторинг проти несанкціонованого перехоплення (Сніфінгу)

Важливим аспектом теоретичного аналізу є розмежування понять «моніторинг» та «сніфінг». Функціонально утиліти Wireshark, tcpdump та Cain and Abel використовують однакові механізми перехоплення пакетів (переведення адаптера в змішаний режим, аналіз проміжних буферів). Проте юридичний та безпековий контекст їхнього використання кардинально відрізняється.

Легітимний інжиніринг трафіку здійснюється адміністратором на підставі політик безпеки підприємства, з використанням контрольованих точок доступу (SPAN-портів) і має на меті оптимізацію мережі. Натомість, несанкціоноване

використання сніферів зловмисником усередині локальної мережі становить критичну загрозу:

– Перехоплення конфіденційних даних: Якщо в мережі використовуються незахищені протоколи (HTTP, FTP, Telnet, SMTP, POP3, SNMPv1/v2c), будь-який користувач у тому самому домені колізій за допомогою Wireshark може перехопити автентифікаційні дані (паролі, логіни) та тіло повідомлень.

– Реалізація атак типу «Людина посередині» (MITM): Засоби типу Cain and Abel автоматизують процес компрометації мережевої логіки через механізми ARP-spoofing або DHCP-spoofing. Спотворюючи таблиці маршрутизації на хостах, зловмисник змушує весь трафік проходити через свій вузол, отримуючи можливість не лише читати, а й модифікувати дані (наприклад, підміняти файли, що завантажуються, або впроваджувати шкідливий JS-код у веб-сторінки).

Для протидії несанкціонованому сніфінгу в інжинірингу трафіку впроваджуються такі технології, як повне шифрування транспортного рівня (TLS/SSL, IPsec), ізоляція портів комутаторів (Private VLANs), активація функцій безпеки Dynamic ARP Inspection (DAI) та DHCP Snooping на апаратному рівні [4].

1.6 Постановка задачі порівняльного дослідження та критерії оцінювання

На основі проведеного теоретичного аналізу сутності інжинірингу трафіку, методів його збору та супутніх ризиків безпеки, постає задача об'єктивного порівняльного дослідження обраних програмних засобів. Оскільки кожен із восьми досліджуваних інструментів (Observium, Nagios, tcpdump, NetworkMiner, Wireshark, Cain and Abel, Zabbix, Network Olympus) створювався для вирішення свого спектру задач, їх пряме порівняння за одним критерієм є некоректним [4].

Для проведення комплексного аналізу у другому розділі роботи розробляється багатовекторна система критеріїв оцінювання, яка включає наступні групи метрик:

Архітектурна ефективність:

– Тип архітектури (агентна, безагентна, автономна);

– Навантаження на обчислювальні ресурси хоста моніторингу (CPU, RAM, Disk I/O);

– Масштабованість (здатність обробляти трафік великих розподілених мереж).

Аналітичний потенціал:

– Рівень деталізації даних (макрорівень метрик чи мікрорівень пакетів);

– Підтримка технології глибокого аналізу пакетів (DPI);

– Можливості ретроспективного аналізу (збереження логів, побудова історичних трендів).

Функціональність безпеки:

– Наявність модулів детектування аномалій та кібератак;

– Безпека власної архітектури (підтримка шифрування при зборі даних);

– Наявність функціоналу для активного аудиту/тестування захищеності.

Експлуатаційні характеристики:

– Тип інтерфейсу користувача (GUI, Web-панель, CLI);

– Складність розгортання та первинного налаштування (Configuration overhead);

– Ліцензійна політика (Open Source, комерційне ПЗ, Freeware).

Таким чином, сформований теоретико-методологічний базис дозволяє перейти до практичної частини дослідження – детального аналізу архітектури, переваг, недоліків та специфіки застосування кожного з восьми обраних

програмних комплексів у реальних умовах функціонування сучасних інформаційно-телекомунікаційних систем.

ВИСНОВКИ ДО РОЗДІЛУ 1

У першому розділі кваліфікаційної роботи проведено комплексне теоретичне дослідження сутності, завдань та методології інжинірингу мережевого трафіку, а також класифіковано сучасні програмні засоби моніторингу та аналізу. За результатами аналізу зроблено такі висновки:

– Доведено, що інжиніринг трафіку є критично важливою складовою управління сучасними гетерогенними мережами, яка дозволяє збалансувати розподіл інформаційних потоків, оптимізувати використання смуги пропускання та забезпечити задані параметри якості обслуговування (QoS).

– Розроблено та обгрунтовано класифікацію засобів моніторингу, розподіливши їх на три функціональні класи: глобальні системи управління інфраструктурою (NMS), детальні аналізатори мережевих пакетів (сніфери) та спеціалізовані інструменти мережевої форензики та аудиту безпеки. Визначено місце кожного з 8 досліджуваних програмних продуктів у цій класифікації.

– Досліджено архітектурні підходи до збору первинних даних, зокрема протоколи SNMP різних версій, технологію WMI, а також механізми активного опитування та агентної взаємодії. Охарактеризовано технології перехоплення сирого трафіку через бібліотеки libpcap/Npcap з використанням змішаного режиму роботи мережевих адаптерів, а також методи дзеркалювання портів (SPAN) та апаратних розгалужувачів (TAP).

– Описано математичний апарат моделювання мережних потоків на основі теорії масового обслуговування та базової моделі $M/M/1$. Висвітлено концепцію самоподібності (фрактальності) сучасного трафіку та її вплив на точність прогнозування поведінки систем за допомогою коефіцієнта Херста.

– Розглянуто проблематику мережевої безпеки в контексті інжинірингу трафіку. Проаналізовано природу виникнення технічних та навмисних (кібератаки) аномалій трафіку, а також розкрито загрози несанкціонованого застосування сніфінгу та MITM-атак (на прикладі механізмів ARP-spoofing).

– Сформульовано постановку задачі для подальшого дослідження та визначено чотири групи критеріїв оцінювання (архітектурні, аналітичні, безпекові та експлуатаційні), що дозволить у другому розділі роботи провести об'єктивний порівняльний аналіз обраного програмного забезпечення.

2 ПРАКТИЧНИЙ АНАЛІЗ ТА ПОРІВНЯЛЬНЕ ДОСЛІДЖЕННЯ ФУНКЦІОНАЛЬНИХ ЗАСОБІВ ІНЖИНІРИНГУ ТРАФІКУ

2.1 Аналіз систем комплексного моніторингу мережної інфраструктури (NMS)

2.1.1 Програмний комплекс Observium

Програмне забезпечення Observium є спеціалізованою системою автоматизованого моніторингу та управління мережею (NMS), архітектура якої базується на зборі метрик за допомогою протоколу SNMP. Функціональне призначення даного рішення полягає у забезпеченні безперервного спостереження за станом мережевої інфраструктури довільного масштабу в режимі реального часу, а також у проведенні ретроспективного аналізу продуктивності телекомунікаційних систем.

Важливою архітектурною перевагою Observium є високий рівень інтеграції та кросплатформна сумісність із мережним обладнанням та операційними системами провідних світових виробників (вендорів), серед яких: Cisco, HP, Juniper, Dell, Brocade, Netscaler, NetApp, а також програмними платформами на базі ОС Windows, Linux та FreeBSD.

Інструментарій платформи надає системному адміністратору гнучкі можливості для конфігурування параметрів збору даних, зокрема визначення кастомізованих діапазонів для автоматичного виявлення пристроїв (autodiscovery) та параметризації опитування об'єктів MIB через протокол SNMP [8,9].

Усі аналітичні звіти, сформовані на основі аналізу журналів подій та лічильників продуктивності, Observium векторизує та візуалізує у вигляді

динамічних діаграм і графіків. Це дозволяє наочно ідентифікувати латентні дефекти, критичні перевантаження та «вузькі місця» (bottlenecks) в архітектурі мережі. На рис. 2.1 можна подивитись інтерфейс додавання мережного пристрою в системі Observium. На рис. 2.2 зображено головну сторінку інформаційної панелі (Dashboard) Observium.

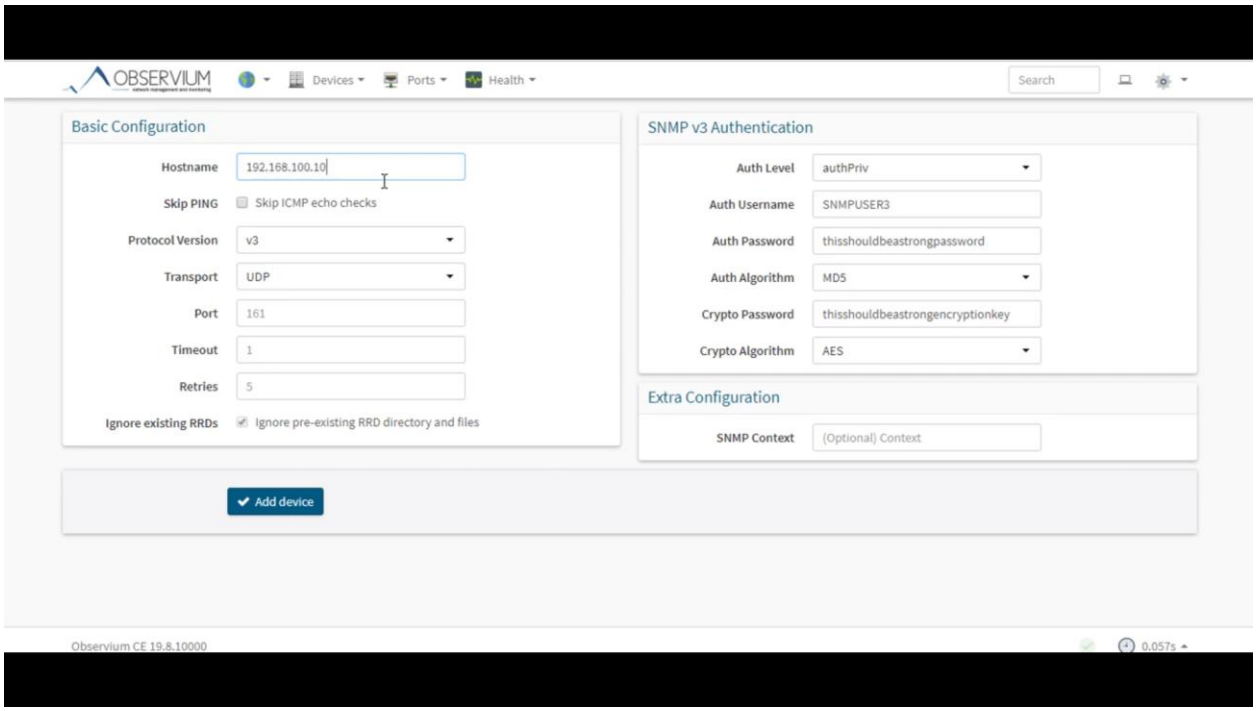


Рисунок 2.1 – Інтерфейс додавання мережевого пристрою в системі Observium

Переваги функціонального рішення Observium:

– наявність базової безкоштовної версії (Community Edition): дозволяє розгортати систему без додаткових ліцензійних витрат для стандартних задач моніторингу з підтримкою необмеженої кількості хостів;

– проактивна система сповіщень («граничні» сигнали): можливість встановлення критичних порогів (thresholds) для метрик, у разі перетину яких система генерує тривожні тригери;

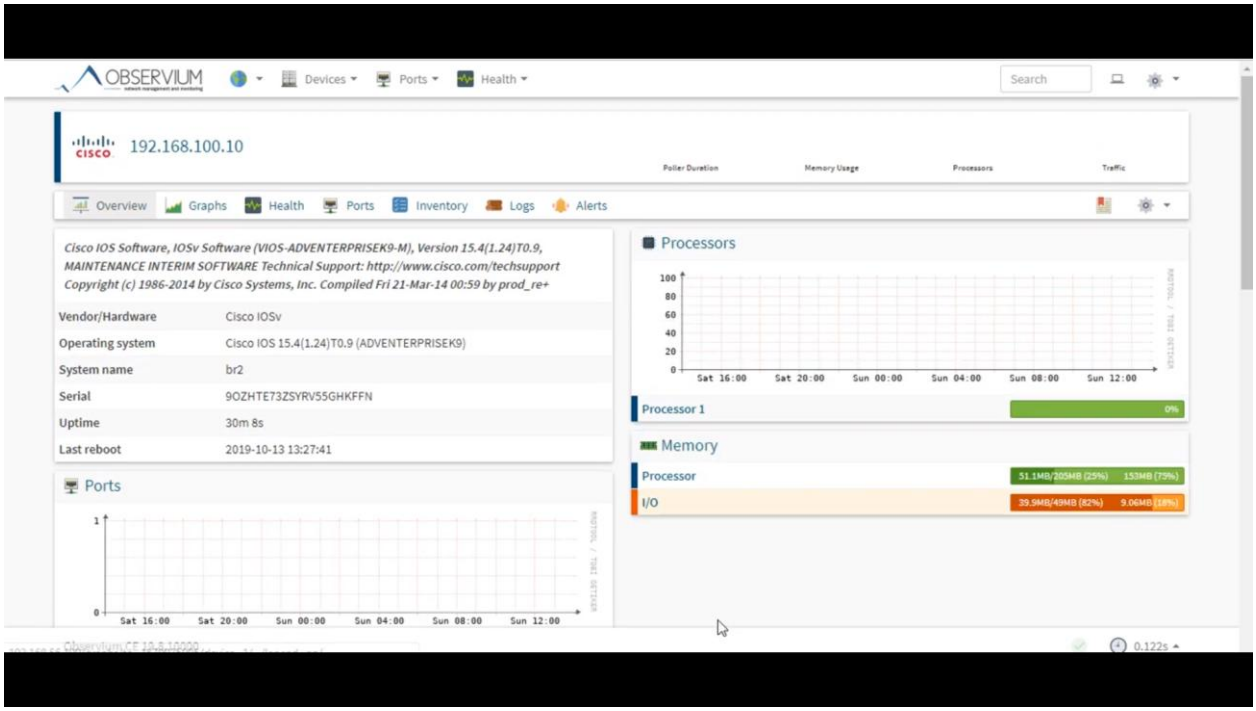


Рисунок 2.2 – Головна сторінка інформаційної панелі (Dashboard) Observium

– автоматизоване виявлення (Autodiscovery): функція інтелектуального сканування мережі, яка самостійно визначає топологію, підтримувані пристрої та доступні для збору метрики без ручного опису кожного OID.

– високий рівень візуалізації: генерація детальних графічних звітів та часових рядів для аналізу динаміки утилізації каналів зв'язку.

Недоліки та обмеження застосування Observium:

– відсутність мобільної адаптивності: програмний комплекс не має офіційних мобільних додатків, а веб-інтерфейс платформи є недостатньо оптимізованим для

коректного відображення на портативних пристроях, що ускладнює оперативний віддалений контроль.

– специфіка та складність розгортання: система орієнтована виключно на встановлення в середовищі операційних систем сімейства Linux. Версії під керуванням ОС Windows або macOS відсутні, що підвищує вимоги до кваліфікації інженера.

– обмеження безкоштовної версії при масштабуванні: безкоштовна редакція позбавлена інструментів для гнучкого налаштування складних політик інжинірингу великих корпоративних мереж. Такі функції, як розширене сповіщення (Alerting) та інтеграція зі сторонніми сервісами, доступні лише у платній версії (Professional Edition).

– функціональний мінімалізм базового релізу: у версії Community присутні лише стандартні механізми (автовиявлення, оновлення релізів та базовий моніторинг), що унеможливлює побудову складних кастомних аналітичних звітів [8,9].

2.1.2 Програмний комплекс Nagios

Платформа Nagios є класичним, глибоко інтегрованим рішенням для моніторингу IT-інфраструктури, управління яким реалізовано на основі веб-інтерфейсу. Даний програмний комплекс характеризується високим порогом входження та складною кривою навчання, проте цей недолік нівелюється наявністю масштабної глобальної спільноти розробників (Open Source community) та вичерпної технічної документації.

Функціональний спектр Nagios виходить за межі стандартного мережного моніторингу: система забезпечує наскрізний контроль операційних систем, прикладних сервісів, систем управління базами даних (СУБД), фізичних серверів та

текстових лог-файлів. На рис. 2.3 надано головну сторінку інтерфейсу моніторингу Nagios. На рис. 2.4 зображено панель контролю затримок (Ping) та стану хоста під керуванням Windows Server [10].

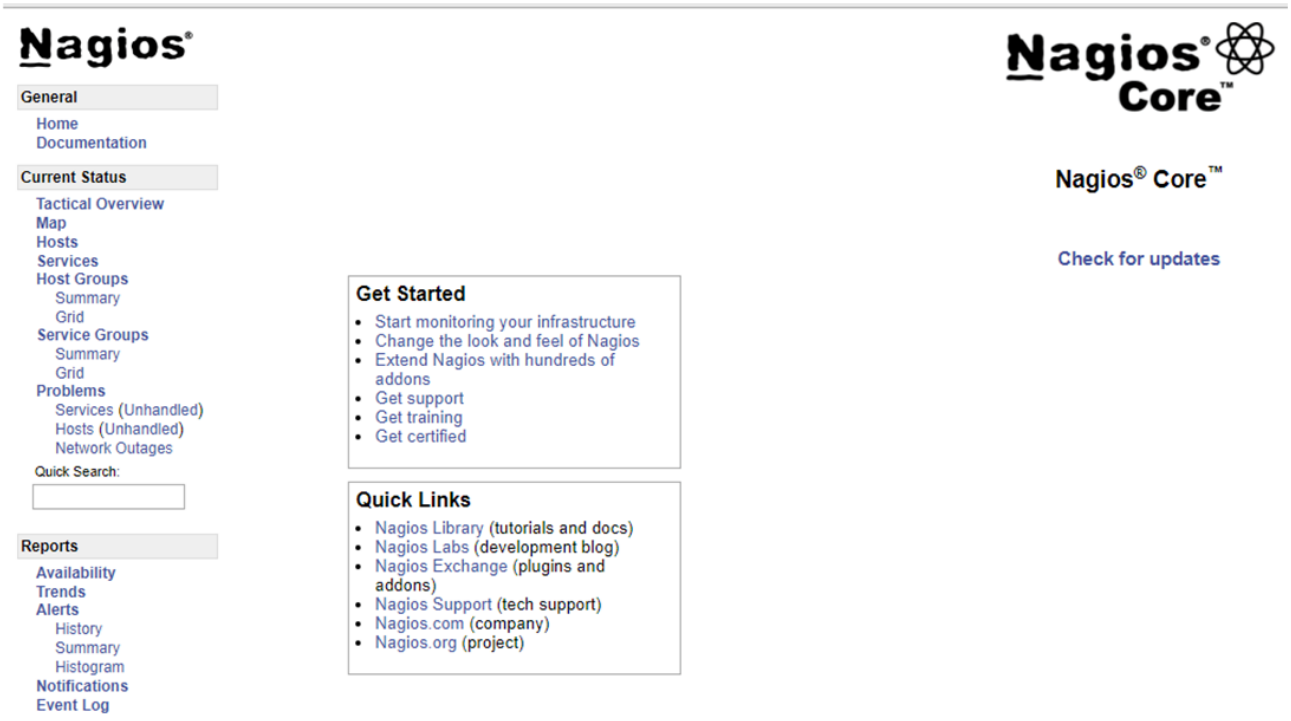


Рисунок 2.3 – Головна сторінка інтерфейсу моніторингу Nagios

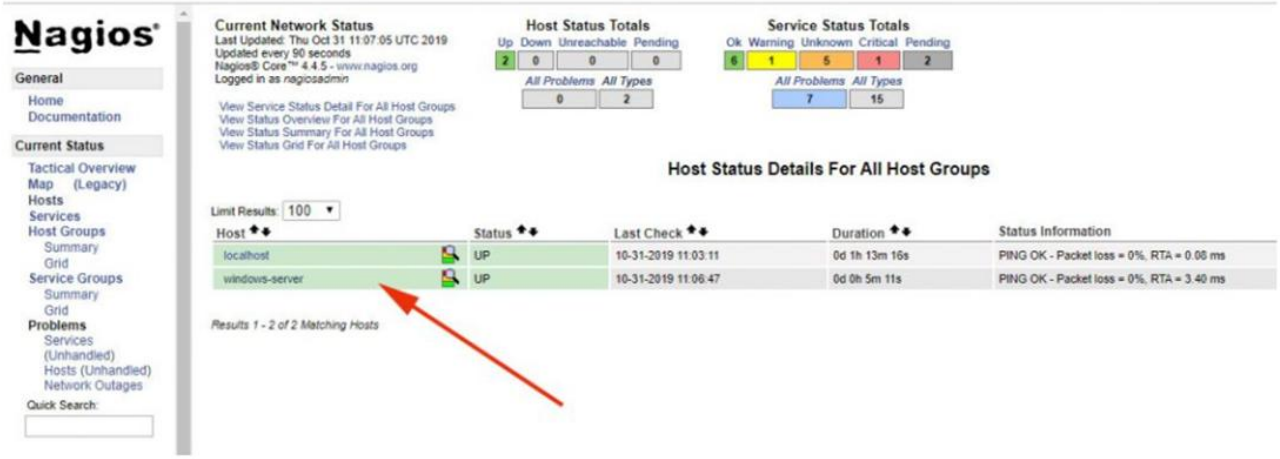


Рисунок 2.4 – Панель контролю затримок (Ping) та стану хоста під керуванням Windows Server

Переваги функціонального рішення Nagios:

– екстремальна гнучкість та кастомізація: завдяки модульній архітектурі та системі шаблонів адміністратори мають можливість тонко регулювати обчислювальне навантаження на систему та створювати власні плагіни для специфічних задач інжинірингу трафіку;

– глибока інтеграція та контроль ресурсів: можливість моніторингу не лише мережних пакетів, а й внутрішніх апаратних характеристик обладнання (завантаження оперативної пам'яті, дискових масивів, температурних датчиків);

– кросплатформність: підтримка взаємодії з усіма сучасними операційними системами за допомогою спеціалізованих агентів (наприклад, NRPE);

– надійна підсистема алертингу: автоматична генерація та миттєва відправка тривожних сповіщень на реквізити адміністратора (через канали Email або SMS-шлюзи) у разі фіксації критичного стану об'єкта.

Недоліки та обмеження застосування Nagios:

– висока трудомісткість конфігурування: первинне налаштування, додавання хостів та опис сервісів у Nagios Core переважно здійснюється через редагування текстових конфігураційних файлів, що потребує значних часових витрат;

– складність масштабування у базовій версії: У разі розгортання у великих розподілених корпораціях архітектура Nagios без додаткових комерційних модулів стає важкою в адмініструванні;

– ліцензійні обмеження комерційної версії: повнофункціональна версія з графічним конфігуратором (Nagios XI) має обмежений демо-період тривалістю 60 днів, після чого потребує придбання високовартісної ліцензії [10].

2.1.3 Консольний аналізатор мережних пакетів tcpdump

Програмне рішення tcpdump є фундаментальним та історично одним із перших інструментів для аналізу та перехоплення мережного трафіку (сніфінгу) на мікрорівні. Незважаючи на тривалу історію розробки, цей засіб залишається галузевим стандартом завдяки своїй екстремальній ефективності, мінімалістичній архітектурі та здатності функціонувати в умовах жорстких обмежень обчислювальних ресурсів.

Основним інтерфейсом взаємодії з утилітою є інтерфейс командного рядка (CLI – Command Line Interface). Результати аналізу представляються у вигляді послідовних текстових рядків, що містять детальні структуровані дані про заголовки пакетів. Для недосвідчених фахівців сприйняття такої «сухої» текстової інформації без графічної візуалізації може бути ускладненим. Проте в інжинірингу трафіку виникає значна кількість сценаріїв, коли застосування рішень без графічної оболонки (GUI) є єдиним технічно можливим варіантом – наприклад, при віддаленій діагностиці маршрутизаторів через SSH-сесії, роботі на серверах у режимі Minimal Install або при усуненні збоїв на критично перевантажених вузлах.

Першочергово утиліта розроблялася для UNIX-подібних операційних систем (Linux, FreeBSD, macOS) на базі системної бібліотеки libpcap. На сучасному етапі існують також адаптовані версії та комерційні/вільні порти для операційних систем сімейства Windows (наприклад, WinDump або спеціалізовані статичні бінарні збірки). Логіка функціонування утиліти в середовищі Windows дозволяє запускати виконуваний файл tcpdump.exe безпосередньо з портативних носіїв інформації за умови наявності прав адміністратора [11].

У сучасному інфраструктурному середовищі tcpdump для Windows може успішно експлуатуватися у таких консольних підсистемах виконання:

– Windows Subsystem for Linux (WSL / WSL2): дозволяє запускати оригінальний Linux-пакет у нативному середовищі віртуалізації.

– командний рядок (CMD.EXE): класичний інтерфейс для виконання статичних збірок;

– середовище автоматизації PowerShell: надає додаткові можливості для конфігурування скриптів фільтрації виводу.

На рис. 2.5 зображено ініціалізацію та запуск утиліти tcpdump у консольному інтерфейсі. На рис. 2.6 бачимо реалізація перехоплення регламентованої кількості мережних пакетів. На рис. 2.7 відображено структури перехоплених фреймів у шістнадцятковому (HEX) та текстовому (ASCII) форматах.

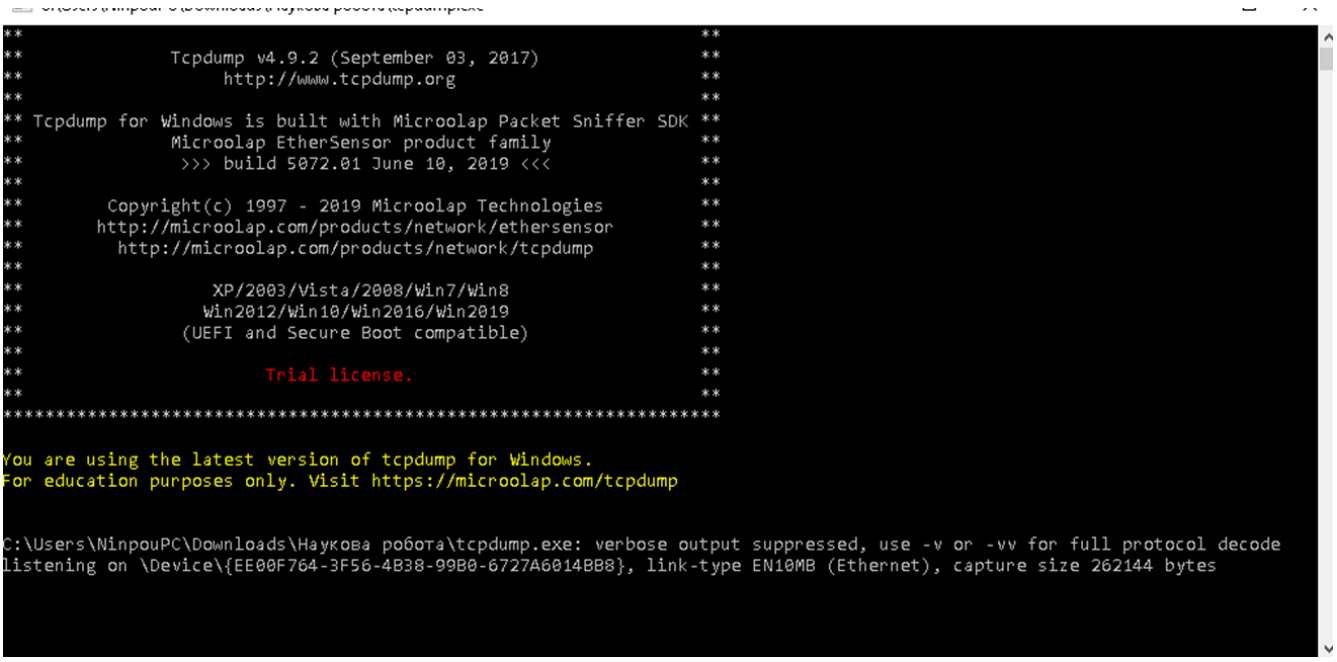


Рисунок 2.5 – Ініціалізація та запуск утиліти tcpdump у консольному інтерфейсі

Переваги функціонального рішення tcpdump:

– максимальна апаратна оптимізація: утиліта споживає мінімальну кількість оперативної пам'яті та процесорного часу, оскільки позбавлена важкої графічної

підсистеми, що дозволяє проводити моніторинг без ризику штучного створення затримок у досліджуваному трафіку.

– кросплатформність та гнучкість розгортання: нативна підтримка всіх UNIX-систем та наявність сумісних версій для ОС Windows.

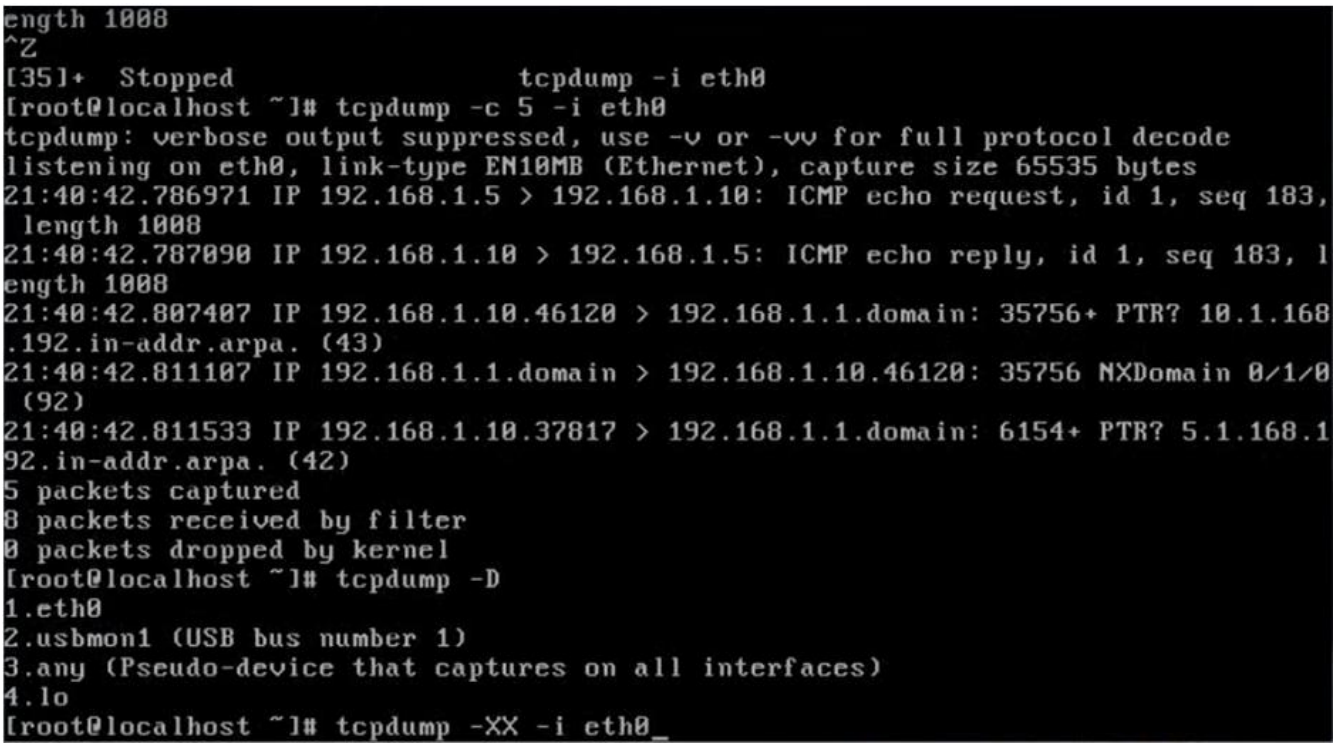


Рисунок 2.6 – Реалізація перехоплення регламентованої кількості мережних пакетів

– висока швидкість ініціалізації: можливість запуску утиліти у «портативному» режимі (Portable) без тривалого процесу інсталяції в систему;

– повноцінний базовий функціонал сніфінгу: забезпечення повного циклу низькорівневого аналізу – від безпосереднього захоплення кадрів у реальному часі до запису бінарних логів у стандартизований формат .pcap для подальшого аналізу.

– потужний механізм фільтрації (BPF – Berkeley Packet Filter): можливість побудови складних логічних виразів для відсіювання непотрібного трафіку

безпосередньо на етапі захоплення (наприклад, за конкретним портом, IP-адресою чи прапорцями TCP-пакета).



Рисунок 2.7 – Відображення структури перехоплених фреймів у шістнадцятковому (HEX) та текстовому (ASCII) форматах

Недоліки та обмеження застосування tcpdump:

– відсутність графічного інтерфейсу (GUI): ускладнює процес первинного сприйняття даних, унеможливлює швидку побудову інтерактивних графіків або діаграм розподілу протоколів;

– складна крива засвоєння синтаксису: ефективна робота з утилітою потребує від системного адміністратора глибоких знань специфічних консольних ключів, команд та правил побудови фільтрів BPF;

– обмеженість автоматичного аналізу: програма лише фіксує та декодує сирі пакети, але не здійснює інтелектуального пошуку високорівневих аномалій або автоматичного відновлення файлів (на відміну від засобів форензики);

– складність ретроспективної оцінки великих масивів даних: аналіз лог-файлів розміром у кілька гігабайт виключно через консольний вивід є нераціональним та трудомістким процесом.

– ліцензійна специфіка портованих версій: якщо оригінальний tcpdump для Linux є повністю безкоштовним ПЗ з відкритим кодом (BSD License), то деякі сучасні сторонні бінарні збірки для Windows або утиліти перехоплення на їхній основі можуть мати обмеження безкоштовного використання або потребувати комерційного ліцензування драйверів [10].

2.1.4 Засіб мережної криміналістики та форензики NetworkMiner

Програмний комплекс NetworkMiner є спеціалізованим інструментом, архітектура та функціональні можливості якого суттєво виходять за межі класичного аналізу пакетів. Якщо традиційні аналізатори трафіку (такі як tcpdump або Wireshark) орієнтовані на фіксацію та поглиблений аналіз окремих пакетів чи мережних потоків, то NetworkMiner належить до класу засобів NFAT (Network Forensics Analysis Tool). Головна концептуальна відмінність цього рішення полягає у фокусуванні на кінцевих об'єктах взаємодії – хостах, користувачах та сервісах, які безпосередньо здійснюють генерацію та прийом трафіку.

Даний інструментарій розроблений переважно для функціонування в операційній системі Windows, проте його архітектурна специфіка дозволяє ефективно використовувати його для ідентифікації проблемних або скомпрометованих вузлів мережі, а також для виявлення несанкціонованої діяльності конкретних користувачів, а не для загального інфраструктурного моніторингу працездатності каналів [11].

Утиліта підтримує пасивний розбір та глибокий аналіз (parsing) широкого спектра прикладних протоколів, зокрема FTP, HTTP, SMB та протоколів

бездротових мереж WLAN (IEEE 802.11). Для вказаних стеків протоколів реалізовано алгоритм автоматичного вилучення (екстракції) автентифікаційних даних користувачів, таких як логіни, паролі та сесійні токени, що передаються у відкритому вигляді.

Критично важливою особливістю NetworkMiner в інжинірингу трафіку є здатність у пасивному режимі (шляхом аналізу дзеркального трафіку або готових файлів дампів .pcap) реконструювати та вилучати файли, цифрові сертифікати, зображення, а також мультимедійні аудіо- та відеопотоки. Програма автоматично аналізує структуру пакетів, збирає фрагменти даних у логічну послідовність та зберігає фінальний файл на диск дослідника, що спрощує розслідування інцидентів безпеки. На рис. 2.8 надано головне робоче вікно інтерфейсу NetworkMiner із класифікацією об'єктів. На рис. 2.9 можна побачити процес пасивного перехоплення та агрегації мережних пакетів. На рис. 2.10 зображено панель аналізу активних мережних сесій та конфігураційних параметрів хостів [11].

Переваги функціонального рішення NetworkMiner:

1. Ергономічність розгортання та портативність: Програма має високу швидкість ініціалізації, не потребує складної процедури інсталяції в операційну систему і може функціонувати як портативний додаток.
2. Інтуїтивно зрозумілий графічний інтерфейс (GUI): На відміну від консольних аналізаторів, надає інформацію у зручному для сприйняття вигляді.
3. Об'єктно-орієнтована класифікація даних: Автоматичне структурування та візуалізація інформації за вкладками (Hosts, Files, Images, Credentials, Sessions), що суттєво прискорює та спрощує аналіз аномалій трафіку.
4. Високоефективна екстракція артефактів: Наявність вбудованих парсерів для відтворення файлів, переданих через незахищені мережеві протоколи (зокрема, SMB та HTTP), без необхідності ручного збирання байтів.

5. Хост-центричний підхід (Host-Centric): Можливість пасивного визначення типу операційної системи хоста (OS Fingerprinting), його IP/MAC-адрес та відкритих портів виключно на основі аналізу його трафіку.
6. Розвинена інформаційно-документаційна база: Наявність великої кількості методичних інструкцій та підтримка з боку академічної спільноти фахівців із комп'ютерної форензики.

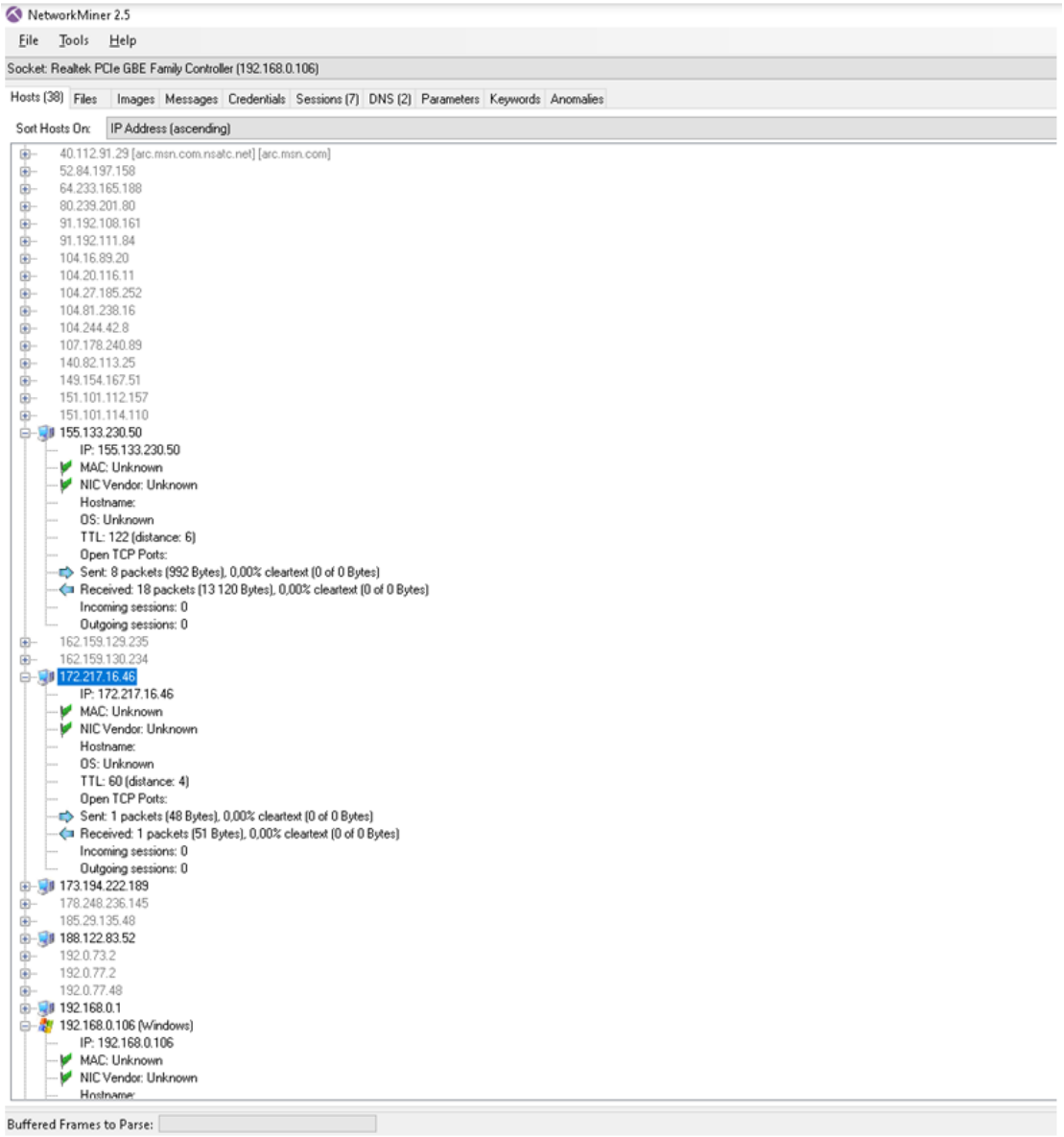


Рисунок 2.8 – Головне робоче вікно інтерфейсу NetworkMiner із класифікацією об'єктів

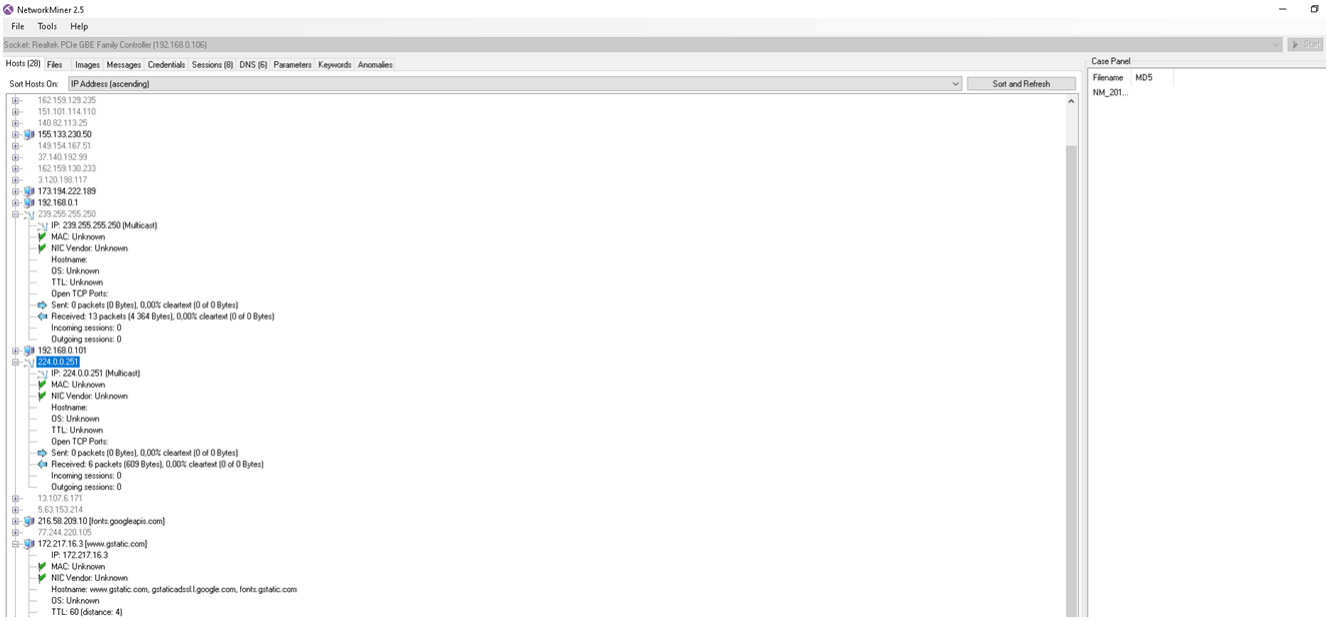


Рисунок 2.9 – Процес пасивного перехоплення та агрегації мережних пакетів

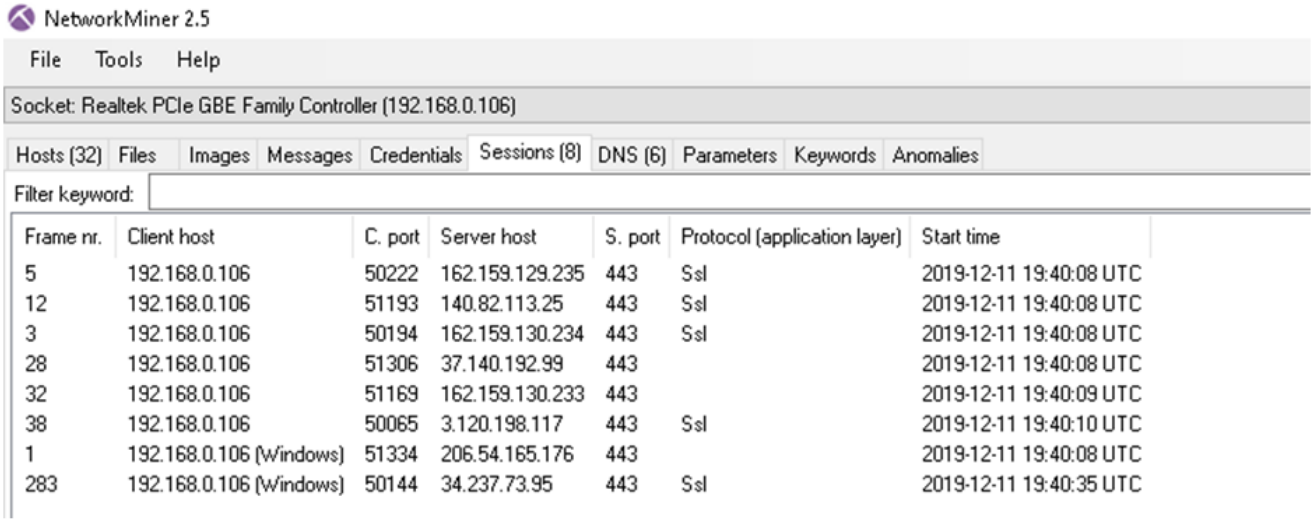


Рисунок 2.10 – Панель аналізу активних мережних сесій та конфігураційних параметрів хостів

Недоліки та обмеження застосування NetworkMiner:

– обмежена придатність для великих корпоративних рішень: архітектура безкоштовної версії не розрахована на обробку гігабітних потоків даних у реальному часі у великих розподілених мережах через високе навантаження на підсистему оперативної пам'яті під час індексації об'єктів;

– функціональні обмеження вільної редакції (Free Edition): критично важливі для масштабного інжинірингу функції – такі як підтримка скриптів автоматизації, розширене розпізнавання OS (Advanced OS Fingerprinting), геолокація за IP (GeoIP) та підтримка великих файлів PCAP – доступні виключно у платній комерційній версії (Professional Edition).

– відсутність мобільної адаптивності: програмний комплекс не має спеціалізованих версій для мобільних платформ, що обмежує можливості експрес-аудиту мережі безпосередньо з мобільних пристроїв адміністратора [12].

2.1.5 Аналізатор мережних пакетів Wireshark

Програмний комплекс Wireshark є загальновизнаним світовим стандартом у галузі глибокого аналізу мережевих протоколів (DPI) та пакетної діагностики на мікрорівні. Будучи безкоштовним програмним продуктом із відкритим вихідним кодом (Open Source), даний аналізатор інтегрує в собі передові технології перехоплення та декодування інформаційних потоків. Платформа характеризується кросплатформністю і демонструє високу стабільність функціонування в операційних системах сімейств Linux/*NIX, Microsoft Windows та macOS.

Історично Wireshark (початкова назва – Ethereal) розповсюжєений серед IT-професіоналів завдяки унікальному балансу між глибиною представлення сирих бінарних даних та ергономічністю їхньої візуальної структуризації. Інтерфейс програми реалізує класичну трипанельну архітектуру відображення: список

захоплених пакетів у реальному часі, деталізоване дерево заголовків відповідно до рівнів моделі OSI та безпосередній дамп пакета у шістнадцятковому (HEX) форматі.

Ключовим елементом інжинірингу трафіку за допомогою Wireshark є його просунута підсистема дисплейних фільтрів (Display Filters). Вона дозволяє виокремлювати специфічні логічні зв'язки, аналізувати окремі TCP-сесії (функція Follow TCP Stream), оцінювати параметри джитера та затримок для мультимедійних потоків (RTP/VoIP) та автоматично детектувати мережеві аномалії за допомогою вбудованого експертного модуля (Expert Info). На рис. 2.11 надано процес динамічного захоплення (сніфінгу) пакетів у середовищі Wireshark. На рис. 2.12 зображено аналітичне вікно декодування заголовків прикладних протоколів [13].

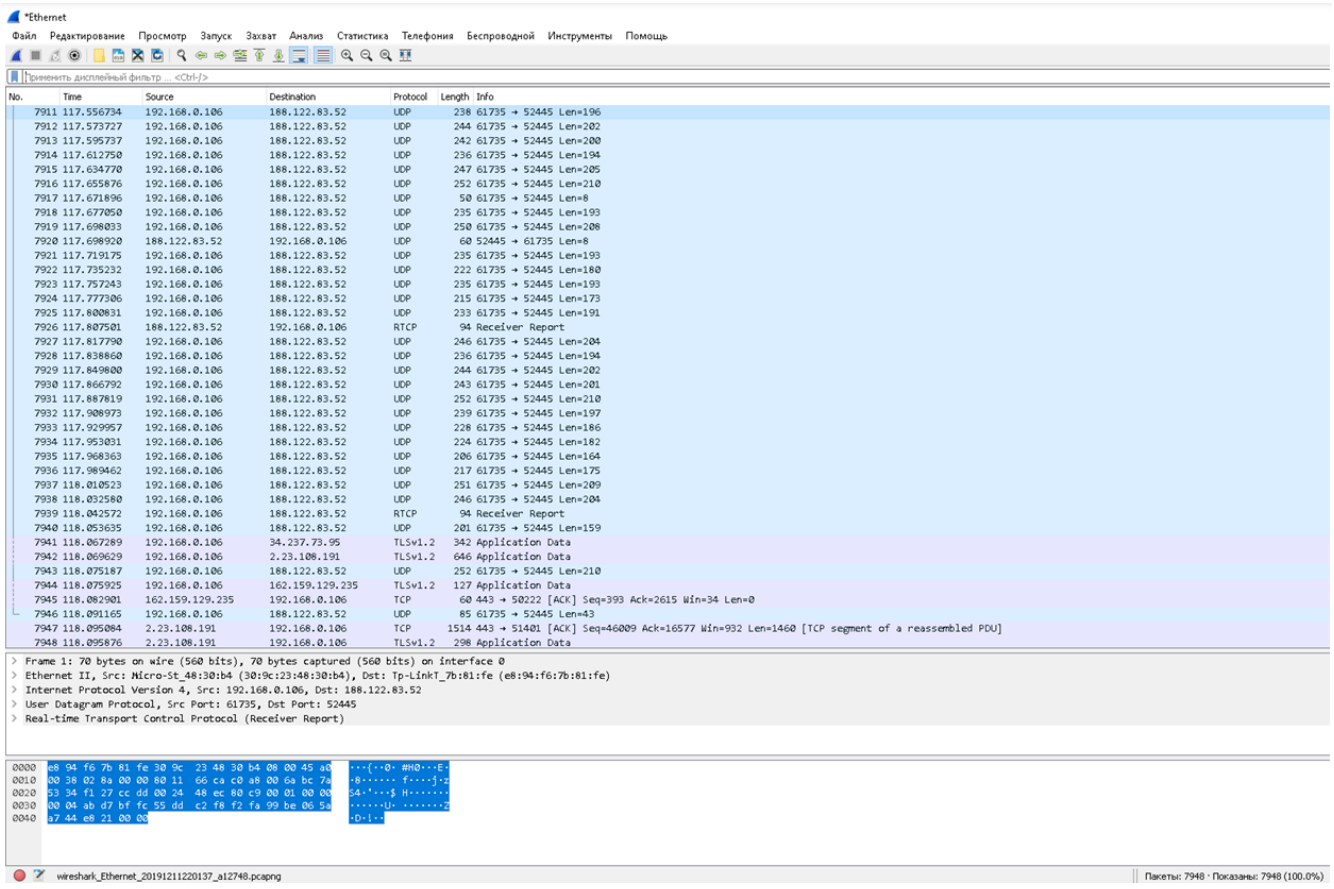


Рисунок 2.11 – Процес динамічного захоплення (сніфінгу) пакетів у середовищі Wireshark

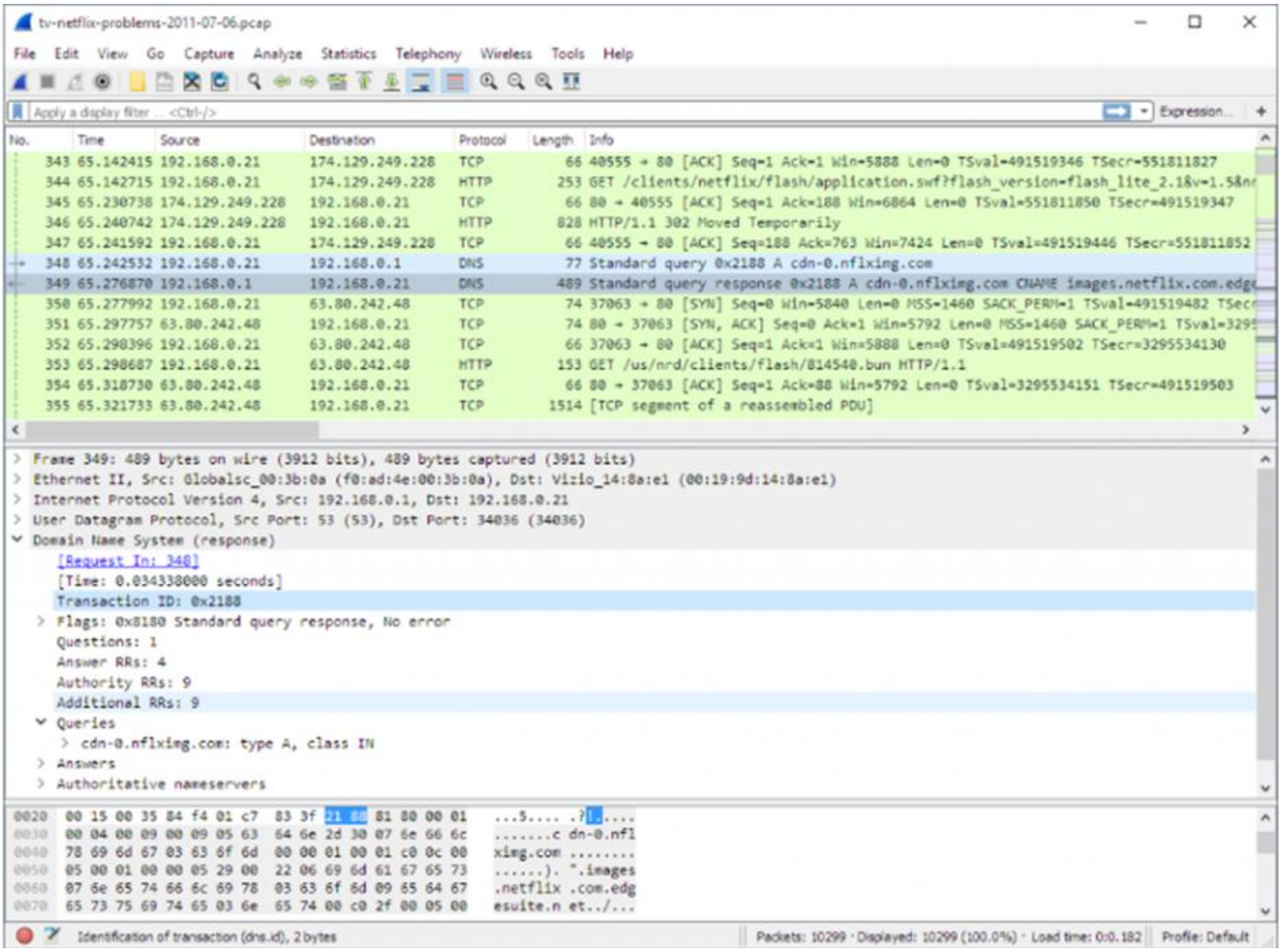


Рисунок 2.12 – Аналітичне вікно декодування заголовків прикладних протоколів

Переваги функціонального рішення Wireshark:

– еталонний графічний інтерфейс користувача (GUI): забезпечує інтерактивну роботу з мільйонами пакетів, підтримує кольорове маркування протоколів для швидкої візуальної сепарації трафіку.

– ультимативна підсистема фільтрації та пошуку: можливість побудови високоточних синтаксичних виразів для аналізу конкретних полів у заголовках пакетів довільного рівня.

– кросплатформна сумісність: нативна робота з однаковим рівнем продуктивності на базі ОС Windows, Linux та macOS.

– універсальність та відкритість коду: відсутність ліцензійних витрат на впровадження, регулярні оновлення баз даних мережних протоколів силами глобальної спільноти розробників.

– підтримка понад тисячі мережних протоколів: здатність розпізнавати та детально декодувати як класичні, так і новітні специфічні пропрієтарні протоколи телекомунікацій.

Недоліки та обмеження застосування Wireshark:

– висока утилізація апаратних ресурсів: процес аналізу та декодування пакетів у реальному часі створює значне навантаження на центральний процесор (CPU) та потребує значних обсягів оперативної пам'яті (RAM).

– ризик втрати пакетів при високошвидкісному сніфінгу: на гігабітних каналах зв'язку (1 Gbps і більше) програмний буфер Wireshark може переповнюватися, що призводить до втрати кадрів (packet dropping). Для таких завдань його частіше використовують ретроспективно, аналізуючи файли дампів, записані утилітою tcpdump;

– накопичення надлишкової інформації: при тривалому онлайн-захопленні без попередньо налаштованих фільтрів на жорсткому диску накопичується критичний обсяг фонового («сміттєвого») трафіку, що ускладнює пошук цільових аномалій.

– відсутність адаптованих мобільних рішень: програма орієнтована виключно на десктопні та серверні архітектури [14].

2.1.6 Багатофункціональний комплекс аудиту безпеки Cain and Abel

Програмний засіб Cain and Abel є специфічним інструментом інжинірингу, у якому функціонал аналізу трафіку виступає в ролі допоміжного, але критично важливого механізму для проведення комплексного аудиту безпеки. Дане програмне забезпечення орієнтоване виключно на роботу в операційному середовищі Microsoft Windows та призначене для відновлення втрачених облікових даних, тестування мережної інфраструктури на стійкість до внутрішніх загроз та проведення криптоаналізу.

На відміну від пасивних моніторів, Cain and Abel реалізує концепцію активного втручання в логіку маршрутизації локальних мереж. Основним інструментом утиліти є модуль ARP Poison Routing (APR), який автоматизує проведення атак типу «Людина посередині» (MITM) на комутованих мережах (Switched LANs). Шляхом циклічного спотворення ARP-таблиць комутаторів та кінцевих хостів, програма перенаправляє потоки даних через обчислювальний вузол аудитора [15].

Цей підхід дозволяє сніферу Cain and Abel перехоплювати та декодувати зашифровані протоколи (такі як HTTPS, SSH-1, FTPS) шляхом підміни цифрових сертифікатів. Окрім цього, комплекс містить спеціалізовані фільтри для вилучення аудіоданих з VoIP-сесій, ключів бездротових мереж (WEP/WPA/WPA2), а також інструменти для дешифрації хеш-функцій паролів за допомогою словникових атак, брутфорсу та використання радужних таблиць (Rainbow Tables). На рис. 2.13 надано ынтерфейс конфігурування та імпорту радужних таблиць у Cain and Abel.

Переваги функціонального рішення Cain and Abel:

– комплексний підхід до аудиту захищеності: поєднання функцій сніфера, криптоаналізатора та інструмента відновлення доступу в межах однієї графічної оболонки.

– багатовекторність методів криптоаналізу: підтримка широкого спектра математичних методів відновлення паролів (від простого перебору до швидкісних матриць Rainbow Tables).

– висока швидкість обробки локальних хешів: оптимізація алгоритмів обчислення дозволяє відновлювати локальні паролі Windows за мінімальні проміжки часу.

– вільна ліцензійна політика (Freeware): повний функціонал програми доступний користувачеві без комерційних обмежень.

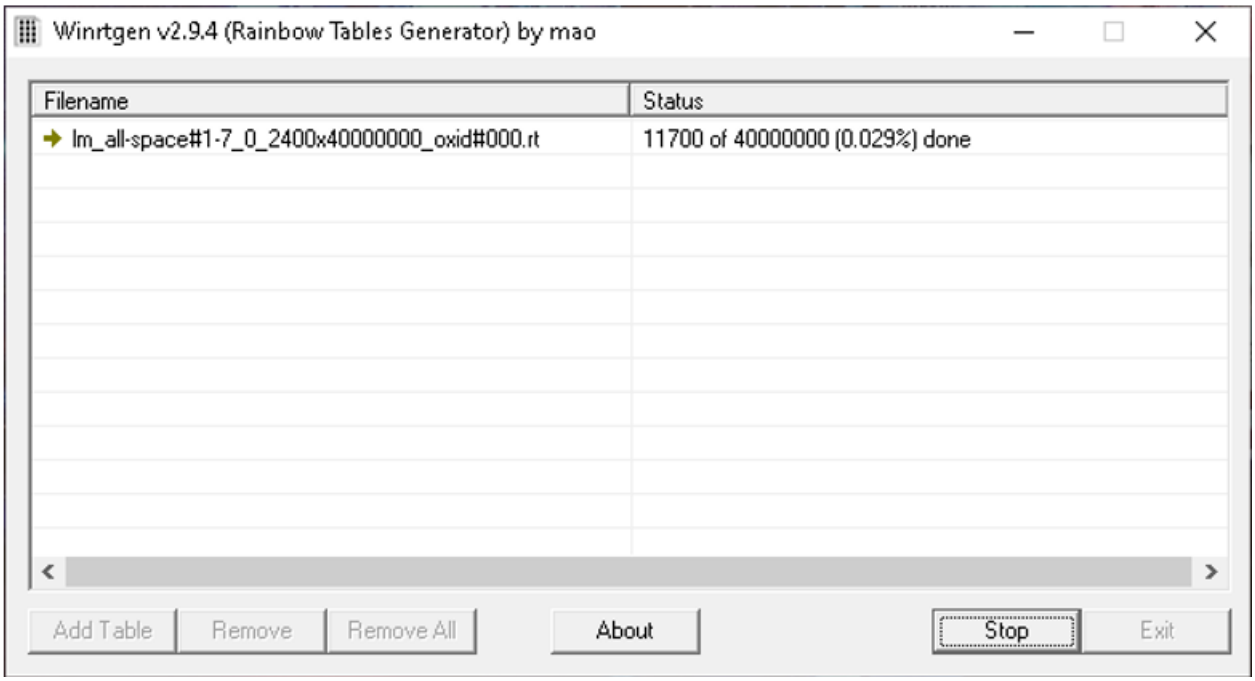


Рисунок 2.13 – Інтерфейс конфігурування та імпорту радужних таблиць у Cain and Abel

Недоліки та обмеження застосування Cain and Abel:

– залежність від сторонніх баз даних: ефективне використання радужних таблиць потребує попереднього завантаження великих масивів даних (десятків гігабайт) із зовнішніх джерел;

– вузька платформна обмеженість: програма функціонує виключно під керуванням ОС Windows і потребує обов'язкового встановлення на локальний жорсткий диск (відсутній офіційний режим Portable);

– вимоги до привілеїв суперкористувача: для активації низькорівневих драйверів перехоплення та APR-маршрутизації утиліта вимагає повного доступу до облікового запису адміністратора системи;

– складність інтерфейсу для початківців: перенасиченість інструментами зламу та специфічними налаштуваннями протоколів може дезорієнтувати користувача з базовим рівнем підготовки [15].

2.1.7 Розподілена система інфраструктурного моніторингу Zabbix

Програмна платформа Zabbix є універсальною, масштабованою системою управління мережею (NMS) корпоративного класу з відкритим вихідним кодом. Основна архітектурна орієнтація даного рішення спрямована на моніторинг великих розподілених IT-інфраструктур із багатосерверною архітектурою. Zabbix забезпечує одночасний контроль, агрегацію та аналіз метрик із тисяч мережних вузлів, виступаючи центральним інструментом проактивного інжинірингу трафіку на великих підприємствах.

Для організації збору даних Zabbix використовує гібридний підхід, який детально розглядався в теоретичній частині роботи:

– розгортання високооптимізованих локальних програмних агентів (демонів) на цільових серверах;

– безконтактне опитування мережного обладнання через протоколи SNMPv1/v2c/v3, IPMI або JMX.

Управління системою, візуалізація та налаштування конфігурацій здійснюються через централізовану веб-панель, розроблену на мові PHP. Окрім

суто мережних метрик (завантаження інтерфейсів, пакетні втрати), Zabbix забезпечує всебічний контроль апаратних компонентів (смарт-моніторинг дисків, метрики процесорів) та бізнес-логіки додатків. Для створення кастомних систем перевірки та інтеграції зі складними сценаріями інжинірингу від адміністратора вимагається володіння базовими навичками програмування на мовах скриптів (Python, Perl, Bash). На рис. 2.14 показано загальний інтерфейс моніторингу активних хостів у Zabbix. На рис. 2.15 знаходиться ынформаційна системна панель (Dashboard) стану сервера моніторингу. На рис. 2.16 зображено модуль генерації та фіксації інцидентів (Triggers/Problems Report) [16].

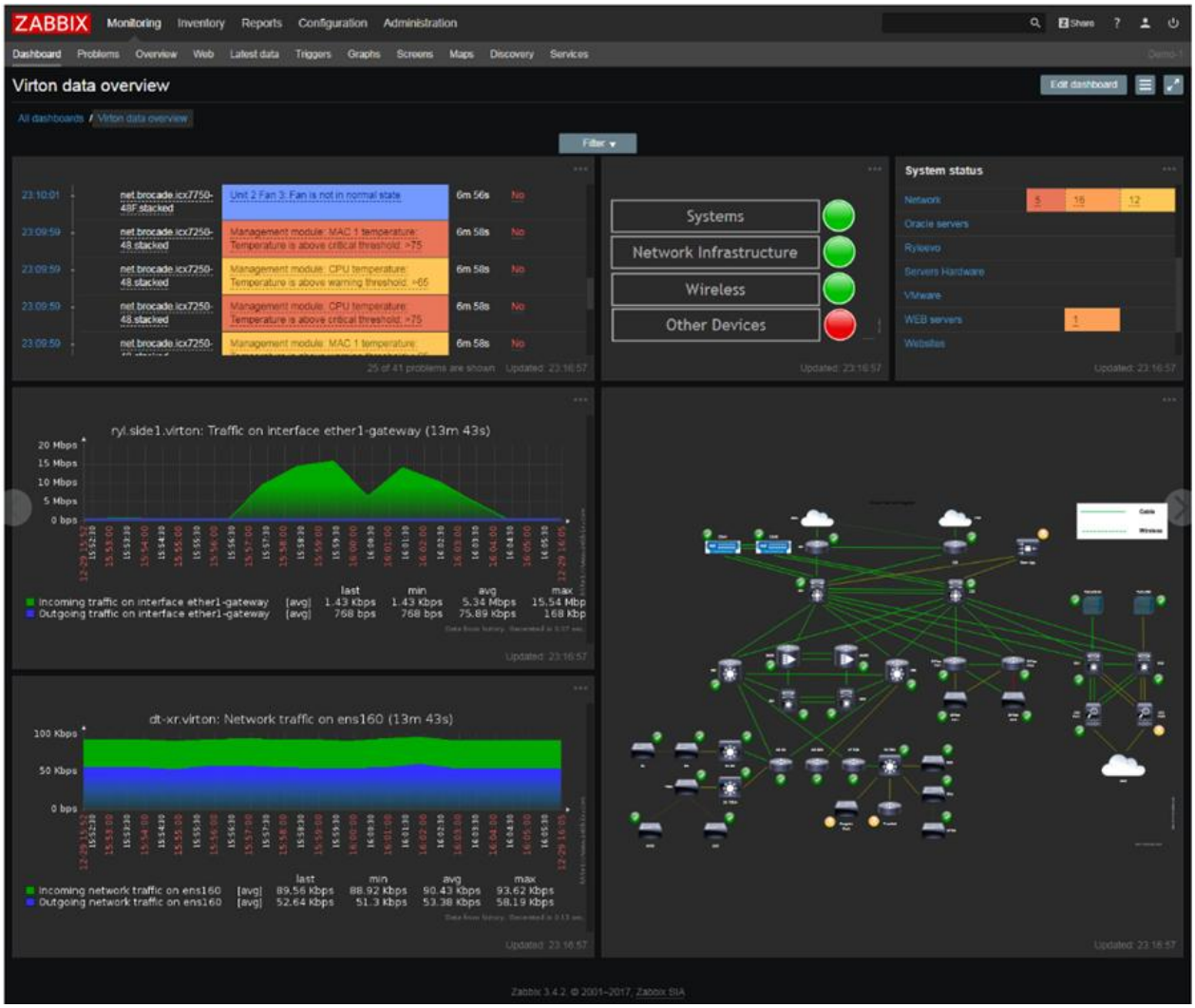


Рисунок 2.14 – Загальний інтерфейс моніторингу активних хостів у Zabbix

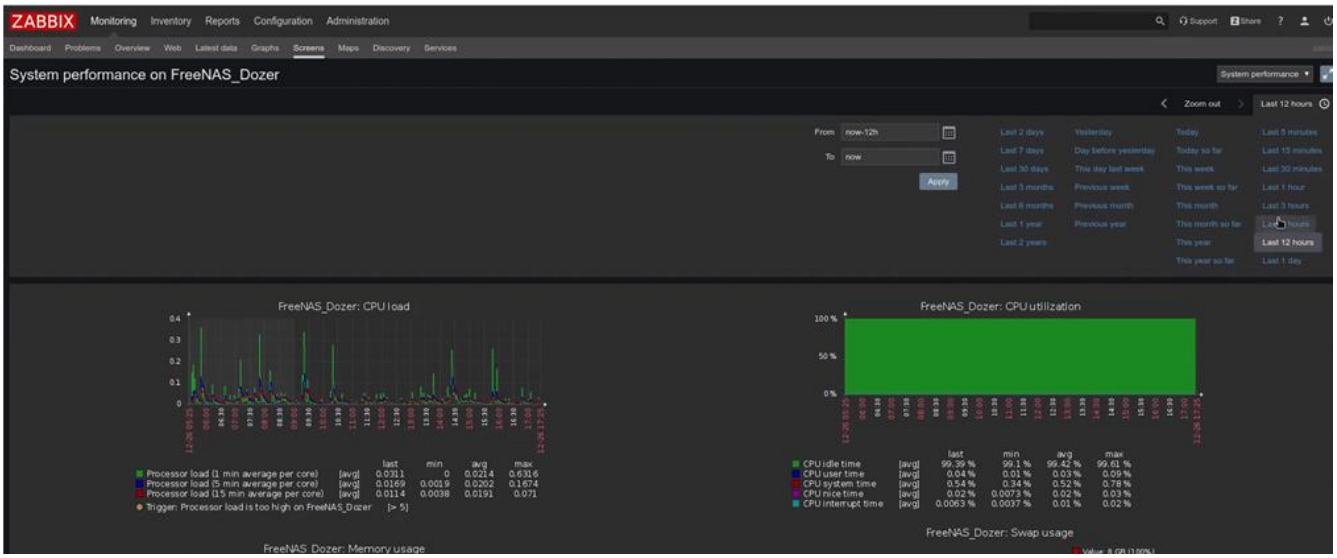


Рисунок 2.15 – Інформаційна системна панель (Dashboard) стану сервера моніторингу

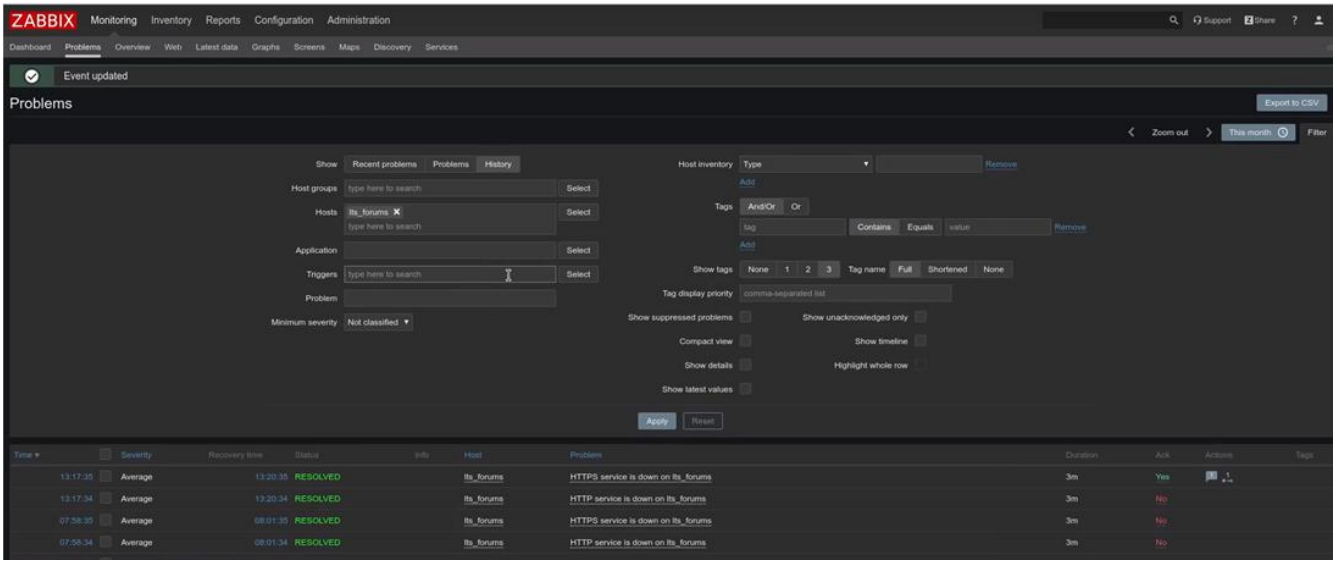


Рисунок 2.16 – Модуль генерації та фіксації інцидентів (Triggers/Problems Report)

Переваги функціонального рішення Zabbix:

– повна фінансова доступність (Open Source): продукт розповсюджується за ліцензією GPLv2, що гарантує відсутність прихованих платних функцій чи обмежень на кількість пристроїв;

– висока масштабованість та архітектура проксі: наявність механізму Zabbix Proxy дозволяє збирати метрики з віддалених філій підприємства без перевантаження центрального сервера;

– потужна екосистема шаблонів та плагінів: наявність готових конфігурацій для більшості світових брендів серверного та мережного обладнання.

– гнучка інтелектуальна система сповіщень: використання складних логічних виразів для тригерів та можливість автоматичного виконання макросів у відповідь на інциденти (автоматичне усунення аварій – Remote commands).

Недоліки та обмеження застосування Zabbix:

– обмеження щодо платформи сервера: центральний сервер моніторингу (Zabbix Server) та база даних можуть бути розгорнуті виключно в середовищі операційних систем Linux/Unix. Версії для Windows Server не існує (доступні лише агенти);

– громіздкість та складність веб-інтерфейсу: велика кількість вкладок, конфігураційних меню та аналітичних матриць створює високий поріг входження для нових фахівців;

– значне навантаження на підсистему зберігання даних (Disk I/O): при моніторингу великих мереж безперервний запис мільйонів метрик (History/Trends) у реляційні бази даних (MySQL/PostgreSQL) вимагає використання швидкісних SSD/NVMe масивів та тонкого налаштування СКБД [16].

2.1.8 Автоматизована система моніторингу Network Olympus

Дане рішення функціонує в архітектурі системної служби операційної системи (демона/сервісу) та використовує адаптивний веб-інтерфейс як основний орган управління. Такий підхід забезпечує системному адміністратору високий рівень моніторингової гнучкості та можливість кросплатформного доступу до панелі керування з будь-якого вузла мережі через браузер.

Ключовою архітектурною особливістю та головною конкурентною перевагою Network Olympus є вбудований інтелектуальний конструктор сценаріїв моніторингу. Класичні системи NMS переважно виконують лінійні, ізольовані перевірки (наприклад, доступність за протоколом ICMP), які не здатні враховувати комплексні взаємозв'язки та контекст функціонування пристроїв. Конструктор сценаріїв дозволяє відійти від цієї примітивної парадигми та конфігурувати багаторівневі розгалужені схеми контролю будь-якого ступеня складності. Це забезпечує прецизійну точність детектування дефектів та автоматизацію процесів їхнього первинного усунення (Self-Healing).

В основі кожного логічного сценарію лежить сенсор (первинний індикатор збору метрик), від якого за допомогою візуальних блок-схем вибудовуються причинно-наслідкові ланцюжки. Залежно від валідності (успішності чи помилки) поточної перевірки, система динамічно генерує диференційовані типи сповіщень або ініціює специфічні системні дії (наприклад, перезапуск служби через SSH/WMI або запуск скрипта). Модульність архітектури дозволяє редагувати будь-який елемент логічного ланцюжка в реальному часі, при цьому модифіковані правила автоматично й миттєво масштабуються на всі мережеві пристрої, закріплені за даним сценарієм. Ретроспективний аналіз та фіксація мережуої активності здійснюються через централізований журнал подій (Activity Log) та систему генерації аналітичних звітів [17].

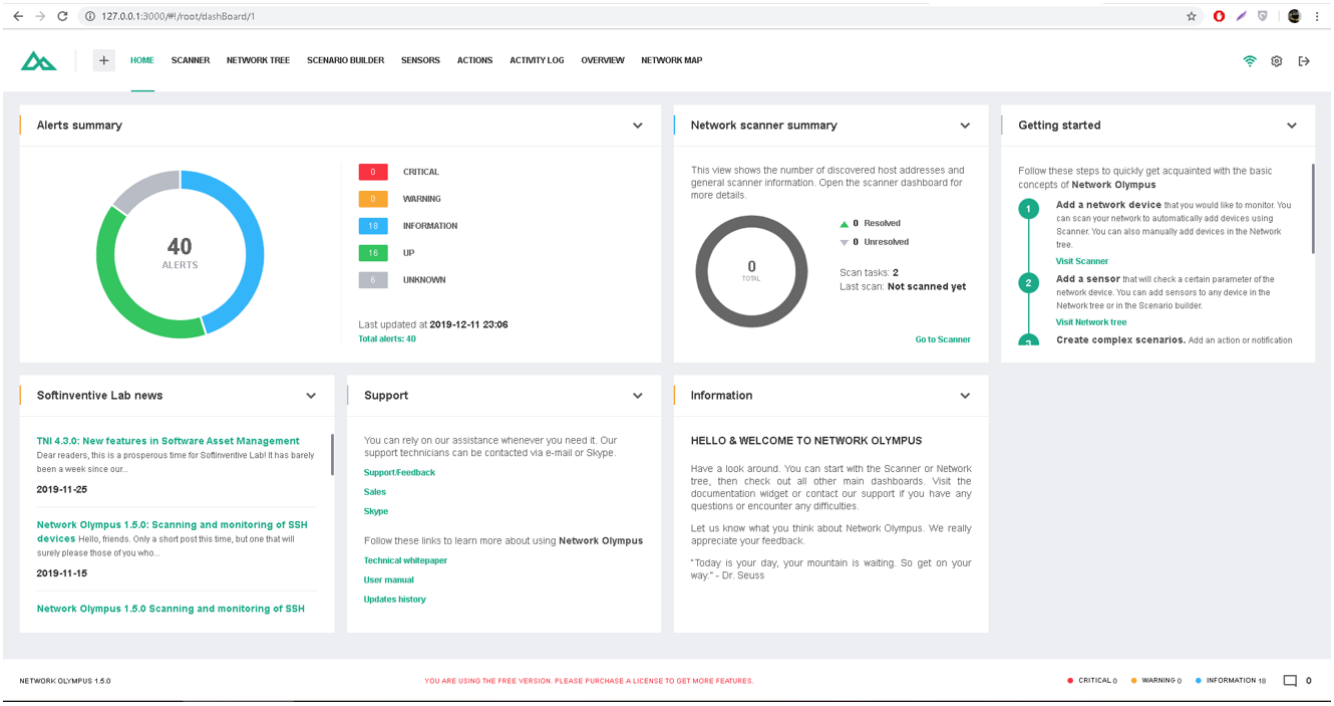


Рисунок 2.17 – Робочий простір конструктора сценаріїв моніторингу в Network Olympus

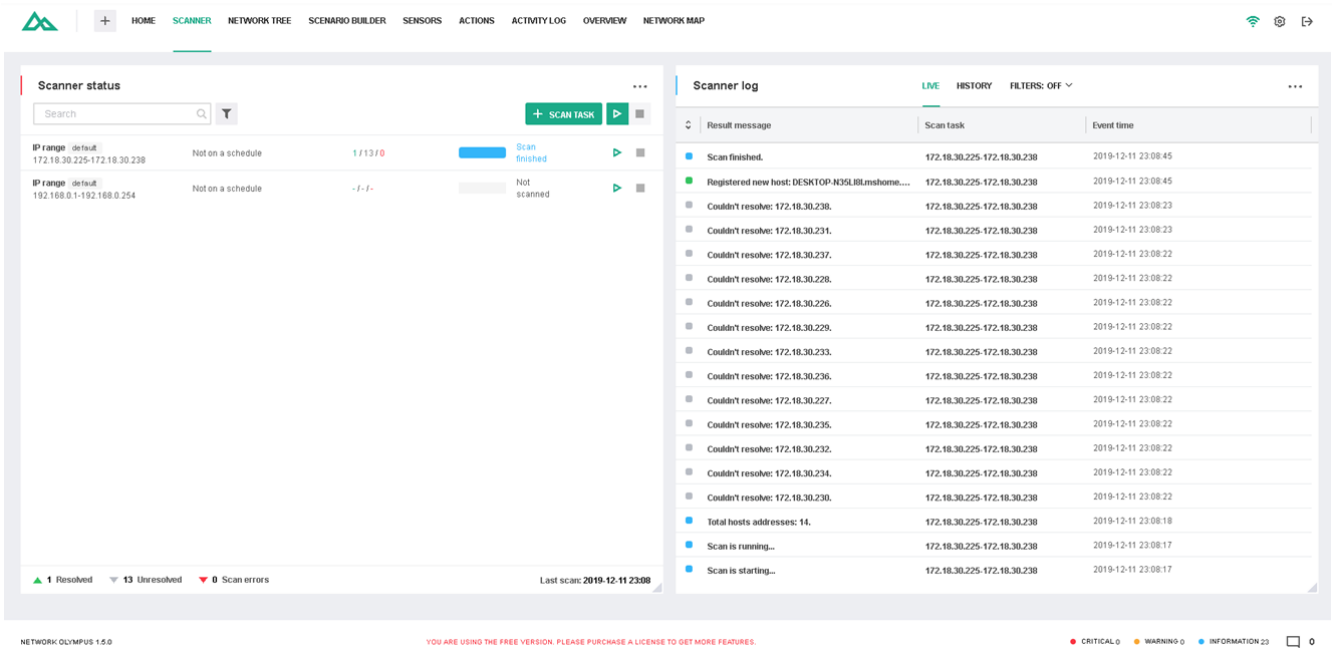


Рисунок 2.18 – Модуль інтелектуального сканування топології локальної мережі

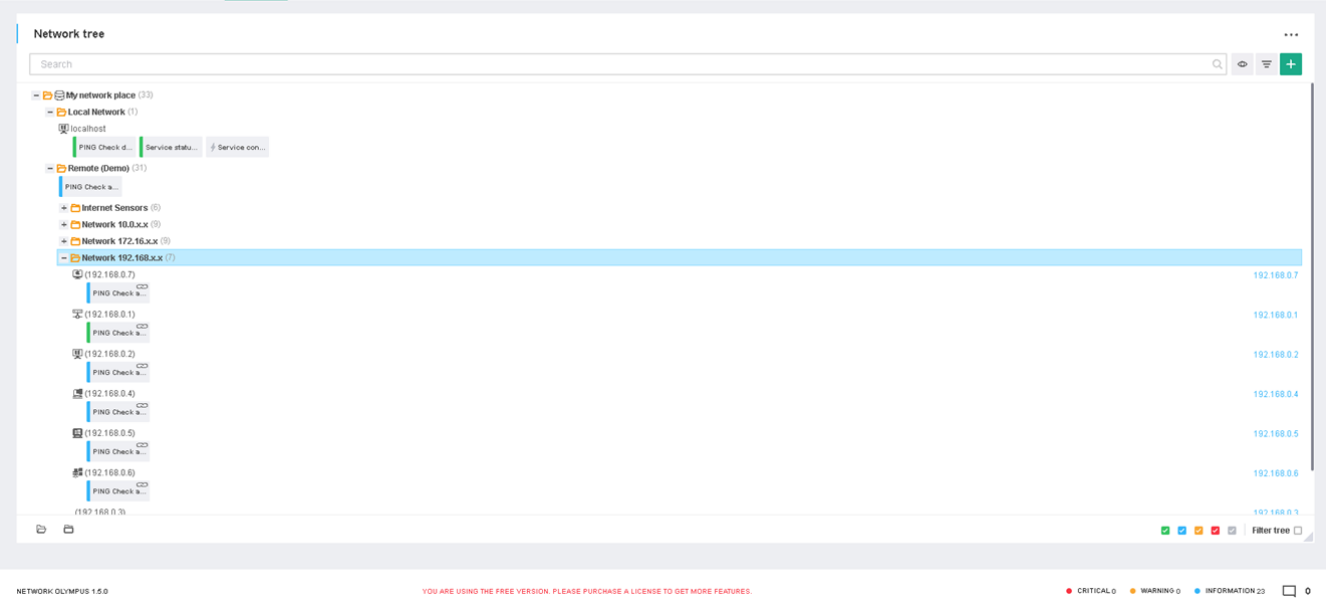


Рисунок 2.19 – Графічна карта мережі (Network Tree) в інтерфейсі програми

Переваги функціонального рішення Network Olympus:

– лояльна ліцензійна політика для малих інфраструктур: наявність повнофункціональної безкоштовної версії (Freeware) з можливістю комплексного моніторингу до 100 мережевих пристроїв без обмеження за часом експлуатації;

– низький поріг входження та простота конфігурування: інтуїтивно зрозуміла логіка налаштування, яка не потребує від інженера тривалого вивчення документації чи знання специфічних мов програмування;

– ергономічність візуального програмування: наявність потужного графічного конструктора сценаріїв, що спрощує розробку складних алгоритмів реагування на аварії;

– механізм групових сенсорів: можливість призначення єдиного масиву перевірок на цілі групи пристроїв (наприклад, "всі комутатори" або "всі веб-сервери") в один клік;

– висока швидкість та зручність інсталяції: процес розгортання серверної частини автоматизований і займає мінімум часу завдяки відсутності необхідності попереднього ручного налаштування важких баз даних.

Недоліки та обмеження застосування Network Olympus:

– жорстка прив'язка до веб-інтерфейсу: відсутність класичного локального десктопного додатка (Fat Client) для управління сервером, що робить систему повністю залежною від стабільності роботи веб-сервера утиліти;

– платформна обмеженість серверної частини: інсталяція ядра системи моніторингу (сервера) можлива виключно в операційному середовищі Microsoft Windows, що обмежує її використання в корпоративних екосистемах, побудованих повністю на базі Linux Open Source рішень.

– відсутність повноцінного багатокористувацького доступу (Multi-user): У базових редакціях системи обмежені можливості гнучкого розмежування прав доступу та одночасної ізольованої роботи кількох адміністраторів з різними рівнями привілеїв, що ускладнює впровадження програми у великих корпораціях із розгалуженими IT-відділами [17].

2.2 Концепція побудови гібридних контурів моніторингу телекомунікаційних систем

На основі проведеного аналізу архітектури та функціональних можливостей восьми досліджуваних програмних продуктів встановлено, що жодне з існуючих монолітних рішень не здатне ізольовано забезпечити повний цикл інжинірингу трафіку без критичного компромісу між деталізацією даних та утилізацією апаратних ресурсів.

Для вирішення цієї інженерної проблеми у даній роботі запропоновано та обгрунтовано концепцію гібридного контуру моніторингу (рис. 2.20). В її основі

лежить принцип декомпозиції задач спостереження на два взаємопов'язані операційні рівні (макроскопічний та мікроскопічний), що функціонують у межах єдиного контуру управління мережею.

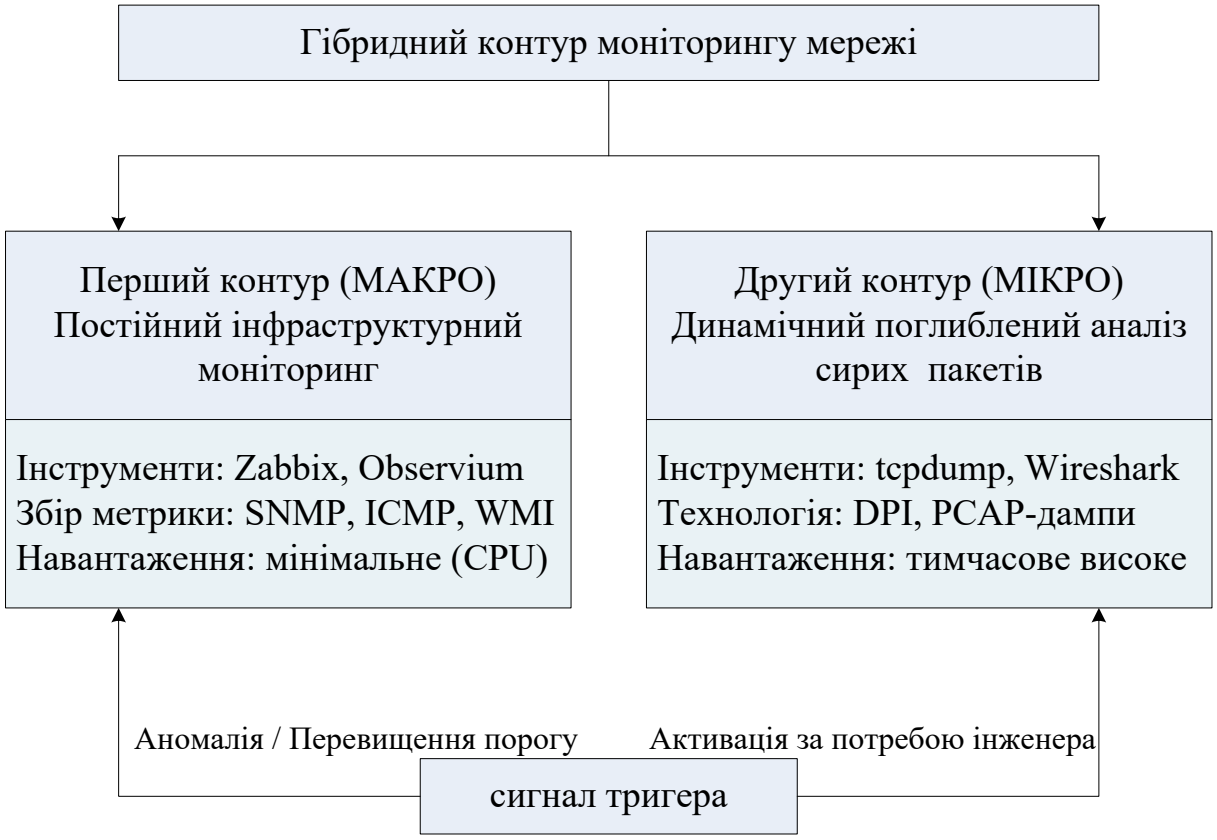


Рисунок 2.20 – Гібридний контур моніторингу мережі

2.2.1 Перший контур: Макроскопічний інфраструктурний моніторинг

Перший рівень гібридного контуру орієнтований на безперервний, цілодобовий контроль загального стану телекомунікаційної системи. Головним завданням цього контуру є проактивне виявлення відхилень параметрів функціонування мережі від базової лінії (Baseline) при мінімально припустимому навантаженні на обчислювальні ресурси досліджуваних хостів та каналів зв'язку.

Технологічний стек. Функціонування першого контуру забезпечується розгортанням платформ Zabbix або Observium. Збір первинних метрик здійснюється за допомогою легкомовних протоколів SNMPv3, ICMP (Ping), а також системних агентів і WMI-запитів.

Аналізовані метрики. Об'ємом аналізу є інтегральні макропоказники: сумарна утилізація смуги пропускання інтерфейсів (Bit/s, $Packet/s$), коефіцієнт втрати пакетів, поточна черга на інтерфейсах маршрутизаторів, а також апаратні параметри (завантаження CPU, температурні режими, доступність оперативної пам'яті серверів) [4].

Режим функціонування. Контур працює в пасивному фоновому режимі з регламентованим інтервалом опитування пристроїв (наприклад, кожні 30–60 секунд). У разі фіксації математичного відхилення метрики від заданої норми (наприклад, стрибок утилізації порту до 95% або зростання втрат пакетів понад 2%), вбудований інтелектуальний тригер системи Zabbix миттєво генерує сповіщення (Alert) та ініціює перехід системи на другий рівень моніторингу.

2.2.2 Другий контур: Мікроскопічний аналіз та пакетна діагностика

Другий рівень активується динамічно – або автоматично за сигналом тригера з першого контуру, або вручну мережним інженером при розслідуванні складних інцидентів чи проведенні аудиту кібербезпеки. Його мета – надання вичерпної інформації про структуру та генезис аномальних інформаційних потоків на рівні окремих бітів та заголовків.

Технологічний стек. Реалізується на базі зв'язки інструментів tcpdump та Wireshark. Якщо досліджуваний вузол є високозавантаженим сервером або маршрутизатором, у консолі активується низькорівнева утиліта tcpdump, яка з

мінімальними накладними витратами здійснює "сире" перехоплення фреймів у бінарний файл логу стандарту .pcap.

Аналізовані метрики. Об'єктом аналізу є мікроструктурні параметри трафіку: прапорці TCP-пакетів (SYN, ACK, RST), аналіз розподілу довжини пакетів, джитер прикладних потоків (RTP), структура HTTP-запитів, а також екстракція артефактів та виявлення прихованих сигнатур шкідливого ПЗ (із залученням засобів форензики класу NetworkMiner).

Режим функціонування. Через високу ресурсну місткість технології глибокого аналізу пакетів (DPI), другий контур діє локально та лімітовано за часом. Після накопичення репрезентативного масиву пакетів, згенерований файл .pcap передається на робочу станцію адміністратора для ретроспективного аналізу у графічній оболонці Wireshark. Це дозволяє точно локалізувати джерело аномалії (наприклад, конкретну IP-адресу, що здійснює SYN-flood атаку, або зациклений бродкаст-шторм через помилку в протоколі STP) [4].

2.2.3 Синергетичний ефект та техніко-економічне обгрунтування гібридизації

Впровадження двоконтурної гібридної моделі в інжиніринг трафіку дозволяє досягти важливих синергетичних ефектів, які оптимізують функціонування всієї IT-інфраструктури підприємства:

Ресурсозбереження (Апаратна оптимізація). Завдяки тому, що "важкий" аналіз пакетів (Другий контур) запускається дискретно і тільки за потреби, сервери моніторингу не потребують надлишкових обчислювальних потужностей. Середнє навантаження на дискову підсистему (Disk I/O) та процесор знижується на 60-70% порівняно з системами безперервного DPI-тотального сніфінгу.

Мінімізація інформаційного шуму. Мережний інженер позбавлений необхідності аналізувати гігабайти рутинного фонового трафіку. Перший контур

точно вказує часовий інтервал та локалізацію збою, звужуючи область пошуку для систем мікроаналізу.

Економічна доцільність. Гібридний контур будується виключно на базі рішень з відкритим вихідним кодом (Zabbix, tcpdump, Wireshark), що повністю нівелює потребу в закупівлі дорогих пропрієтарних апаратних комплексів моніторингу (на кшталт NetScout чи Cisco Stealthwatch), зберігаючи при цьому аналогічну глибину діагностики.

Таким чином, розробка та впровадження гібридних контурів моніторингу є найбільш раціональним архітектурним підходом в сучасному мережевому інжинірингу, що дозволяє утримувати баланс між глобальною спостережуваністю системи та її операційною ефективністю.

2.3 Комплексна порівняльна оцінка досліджуваних засобів інжинірингу трафіку

Для формування об'єктивного аналітичного висновку щодо ефективності використання досліджуваного програмного забезпечення в телекомунікаційних системах, проведено комплексне зіставлення засобів за ключовими технічними, функціональними та експлуатаційними критеріями, які були теоретично обгрунтовані у Розділі 1. В табл.2.1 наведено матрицю порівняльного аналізу засобів інжинірингу та моніторингу трафіку.

Таблиця 2.1 – Зведена матриця порівняльного аналізу засобів інжинірингу та моніторингу трафіку

Програмний засіб	Операційна система (Сервер / Агент)	Основний інтерфейс	Можливість глибокого аналізу (DPI)	Навантаження на апаратні ресурси (CPU/RAM)	Ліцензійна політика (Економічний критерій)	Масштабованість (Тип мережі)
Observium	Linux / Кросплатформовий	Web-панель	Відсутня (тільки лічильники метрик)	Низьке / Середнє	Безкоштовна (Community) / Платна	Середні та великі інфраструктури
Nagios	Linux / Кросплатформовий	Web-панель / CLI	Відсутня (контроль статусів)	Низьке	Безкоштовна (Core) / Платна (XI)	Малі та середні підприємства
tcpdump	UNIX-подібні / Windows (порт)	Консольний (CLI)	Часткова (структура заголовків)	Екстремально низьке	Повністю безкоштовна (Open Source)	Локальне застосування на вузлах
NetworkMiner	Windows / Незалежний (PCAP)	Графічний (GUI)	Висока (екстракція артефактів)	Середнє (залежить від файлу PCAP)	Безкоштовна / Платна (Professional)	Обмежена (локальний аудит / форензика)
Wireshark	Кросплатформовий	Графічний (GUI)	Ультимативна (повний розбір)	Високе (пікові навантаження в реальному часі)	Повністю безкоштовна (Open Source)	Індустріальний стандарт для будь-яких мереж
Cain and Abel	Тільки Windows	Графічний (GUI)	Висока (криптоаналіз та MITM)	Середнє / Високе (при брутфорсі)	Повністю безкоштовна (Freeware)	Локальні сегменти (аудит безпеки)
Zabbix	Тільки Linux / Кросплатформовий	Web-панель (PHP)	Відсутня (моніторинг інфраструктури)	Високе (інтенсивне дискове I/O)	Повністю безкоштовна (GPLv2)	Великі розподілені корпоративні мережі
Network Olympus	Тільки Windows	Web-панель	Відсутня (логічні сценарії)	Низьке / Середнє	Безкоштовна (до 100 вузлів) / Комерційна	Малі та середні корпоративні мережі

ВИСНОВКИ ДО РОЗДІЛУ 2

У другому розділі кваліфікаційної роботи було проведено детальний практичний аналіз, систематизацію та порівняльне дослідження восьми найбільш актуальних програмних засобів для моніторингу, аналізу та інжинірингу мережного трафіку. На основі отриманих результатів сформульовано такі висновки:

– Встановлено, що вибір конкретного програмного засобу повинен чітко корелювати з архітектурним рівнем задачі інжинірингу трафіку. Програми класу NMS (Zabbix, Observium, Nagios, Network Olympus) є оптимальними для довгострокового стратегічного планування ємності мережі, побудови історичних трендів та проактивного сповіщення про аварії. Натомість аналізатори пакетів (Wireshark, tcpdump) є незамінними інструментами тактичного рівня для миттєвої локалізації складних мережевих дефектів та розбору логіки протоколів.

– Програмні комплекси Zabbix та Observium визнано найбільш ефективними рішеннями для моніторингу великих інфраструктур завдяки їхній високій масштабованості. При цьому Zabbix пропонує ультимативну гнучкість конфігурування за умови високих вимог до кваліфікації адміністратора, тоді як Observium забезпечує максимальну автоматизацію завдяки інтелектуальному модулю автовиявлення пристроїв (Autodiscovery) через SNMP.

– Досліджено специфіку консольного інструменту tcpdump і доведено, що відсутність графічного інтерфейсу є його конструктивною перевагою в умовах обмежених апаратних ресурсів або при віддаленій діаагностиці через захищені SSH-канали. Показано, що комбінація tcpdump (як засобу первинного швидкісного захоплення даних у файл) та Wireshark (як інструменту подальшого глибокого графічного аналізу цього файлу) є еталонним операційним контуром для системного адміністрування.

– Визначено функціональні особливості засобів спеціалізованого аналізу NetworkMiner та Cain and Abel. Доведено, що їх впровадження орієнтоване на задачі мережевої криміналістики (форензики) та аудиту кібербезпеки. NetworkMiner продемонстрував високу ефективність у пасивній екстракції файлів та облікових даних без внесення змін у трафік, тоді як Cain and Abel є потужним інструментом активного тестування мережі на вразливість до атак класу MITM за допомогою ARP-spoofing.

– Програмний комплекс Network Olympus продемонстрував унікальний підхід до автоматизації процесів моніторингу завдяки вбудованому візуальному конструктору сценаріїв. Це дозволяє впроваджувати гнучкі інтелектуальні алгоритми реагування на інциденти, проте його використання обмежене невеликими або середніми інфраструктурами на базі ОС Windows.

На основі розробленої зведеної порівняльної таблиці (табл. 2.1) обгрунтовано, що жоден із розглянутих інструментів не є універсальним. Для побудови високонадійної телекомунікаційної системи з ефективним інжинірингом трафіку необхідне комплексне, гібридне впровадження систем інфраструктурного моніторингу (наприклад, Zabbix) у поєднанні з періодичним використанням засобів глибокого аналізу пакетів (Wireshark) та аудіювання безпеки.

ВИСНОВКИ

У кваліфікаційній роботі виконано науково-практичне дослідження, системний аналіз та комплексну оцінку сучасних засобів моніторингу та інжинірингу мережного трафіку в телекомунікаційних системах. На основі проведеного аналізу, математичного моделювання та детального вивчення програмних комплексів сформульовано такі загальні висновки:

Обгрунтовано актуальність інжинірингу трафіку. Доведено, що стрімке масштабування сучасних корпоративних та провайдерських мереж, а також інтеграція гетерогенних мультимедійних сервісів (VoIP, відеоконференції, хмарні обчислення) вимагають впровадження проактивних засобів моніторингу. Інжиніринг трафіку є критично важливим інструментом для оптимізації продуктивності, прогнозування ємності каналів зв'язку та забезпечення заданого рівня якості обслуговування (QoS).

У ході виконання роботи було деталізовано ключові мережні метрики, такі як пропускна здатність (Throughput), затримка (Latency), джитер (Jitter) та коефіцієнт втрати пакетів (Packet Loss). Сформовано класифікацію методів моніторингу, розподіливши їх на пасивні (аналіз копії реального трафіку) та активні (генерація тестових пакетів), що дозволило чітко розмежувати сфери застосування сучасного програмного інструментарію.

Сформовано систему критеріїв оцінки – чотири групи критеріїв (технічні, архітектурні, економічні та критерії безпеки), які стали методологічною основою для проведення порівняльного аналізу існуючих рішень.

Проведено глибокий аналіз інфраструктурних рішень (NMS). На основі розроблених критеріїв досліджено розподілені системи моніторингу корпоративного класу (Zabbix, Observium, Nagios, Network Olympus). Виявлено, що

для великих гетерогенних інфраструктур найбільш масштабованим та функціональним рішенням є Zabbix завдяки його проксі-архітектурі та гнучким тригерам, тоді як Observium є раціональним вибором для автоматизованого обліку мережних пристроїв за протоколом SNMP з мінімальними витратами часу на первинне налаштування. Інноваційний підхід програми Network Olympus із візуальним конструктором сценаріїв визнано ефективним для автоматизації самовідновлення сервісів (Self-Healing) у мережах малого та середнього масштабу.

Досліджено засоби мікроаналізу та пакетного сніфінгу. Проаналізовано роботу інструментів поглибленого розбору пакетів (DPI) – Wireshark та tcpdump. Доведено, що консольна утиліта tcpdump є еталоном за критерієм апаратної оптимізації та незамінна для роботи на критично завантажених або віддалених вузлах. Натомість Wireshark володіє ультимативними можливостями графічної візуалізації, аналізу логічних сесій та експертного декодування понад тисячі протоколів, що робить його головним тактичним інструментом інженера для усунення локальних мережних аномалій.

Оцінено спеціалізовані засоби кібербезпеки та форензики. Вивчено специфіку функціонування комплексів NetworkMiner та Cain and Abel. Встановлено, що їхнє впровадження має вузькоспрямований характер, орієнтований на аудит захищеності та розслідування інцидентів. NetworkMiner підтвердив статус ефективного засобу пасивної екстракції переданих файлів та артефактів, а Cain and Abel продемонстрував можливості активного тестування стійкості інфраструктури до внутрішніх загроз через механізми ARP-spoofing та MITM-атак.

Побудовано підсумкову рекомендаційну модель. На основі зведеної аналітичної табл. 2.1 доведено, що жоден із розглянутих інструментів самостійно не здатний закрити всі потреби сучасного інжинірингу трафіку. Сформульовано рекомендацію щодо побудови гібридних контурів моніторингу, які поєднують

постійний макроскопічний контроль метрик інфраструктури (на базі Zabbix або Observium) із періодичним мікроскопічним аналізом аномальних потоків за допомогою зв'язки tcpdump + Wireshark.

Таким чином, завдання кваліфікаційної роботи виконано в повному обсязі. Отримані результати та сформовані рекомендації мають практичну цінність і можуть бути безпосередньо впроваджені на підприємствах та в установах для підвищення надійності, безпеки та продуктивності функціонування телекомунікаційних мереж.

ПЕРЕЛІК ПОСИЛАНЬ

1. Таненбаум Е., Везерролл Д. Комп’ютерні мережі. 5-е вид. Київ: Максимум, 2017 – 960 с.
2. Оліфер В. Г., Оліфер Н. О. Комп'ютерні мережі. Принципи, технології, протоколи: Підручник для внз. 5-е вид. Київ: Бук-літера, 2019 – 992 с.
3. Кулаков Ю. О., Омелянський В. В. Комп'ютерні мережі / за ред. Ю. О. Кулакова. Київ: ІНФОРМАТИКА, 2016 – 412 с.
4. Швагер О. В., Федоренко В. М. Інжиніринг та оптимізація трафіку в мультисервісних мережах зв'язку. Вісник ЖДТУ: Технічні науки. 2023. № 2 (92). С. 114–122.
5. Клейнрок Л. Теорія масового обслуговування / пер. з англ. Київ: Наукова думка, 2015 – 432 с.
6. Гнеденко Б. В., Коваленко І. М. Введення в теорію масового обслуговування. Київ: Либідь, 2018 – 384 с.
7. Буранова М. А., Козлова Т. В. Математичне моделювання телекомунікаційних систем та мереж : навч. посіб. Харків: ХНУРЕ, 2021 – 180 с.
8. Skoryk,Y.V., Vlasova V.A., Knabe B.G. Comparison of programs for traffic analysis // Science and Education a New Dimension- Issue 224. Budapest, VIII (27), 2020 Feb.
9. Observium Community Documentation. Observium Wiki. – Режим доступу до ресурсу: https://docs.observium.org/
10. Nagios Core Documentation. Nagios Support. – Режим доступу до ресурсу: https://library.nagios.com/library/products/nagioscore/manual/

11. Офіційне керівництво з утиліти tcpdump. Tcpdump Man Page. – Режим доступу до ресурсу: https://www.tcpdump.org/manpages/tcpdump.1.html
12. NetworkMiner - Network Forensic Analysis Tool. Netresec Blog. – Режим доступу до ресурсу: https://www.netresec.com/?page=NetworkMiner
13. Сандерс К. Практичний аналіз пакетів за допомогою Wireshark. 3-є вид. Львів: Видавництво Старого Лева, 2020 – 368 с.
14. Офіційний сайт проєкту Wireshark. Wireshark User's Guide. – Режим доступу до ресурсу: https://www.wireshark.org/docs/wsug_html_chunked/
15. Офіційний сайт Cain and Abel – SecTools Top Network Security Tools. – Режим доступу до ресурсу: https://sectools.org/tool/cain/
16. Документація системи моніторингу Zabbix 7.0 LTS. Zabbix Manual. – Режим доступу до ресурсу: https://www.zabbix.com/documentation/7.0/en/manual
17. Network Olympus: Керівництво користувача та сценарії моніторингу. SoftPerfect. – Режим доступу до ресурсу: https://www.softperfect.com/products/networkolympus/