МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ПОЛІСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет інформаційних технологій,
обліку та фінансів
Кафедра комп'ютерних технологій
і моделювання систем

Кваліфікаційна робота
на правах рукопису

Яковенко Валерій Борисович

УДК 004.056:336.71

КВАЛІФІКАЦІЙНА РОБОТА

Програмна система кібербезпекового аналізу мережевого трафіку комерційного банку «Credit Agricole»

Спеціальність – 125 «Кібербезпека»

Подається на здобуття освітнього ступеня магістр

кваліфікаційна робота містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело
_____ В. Б. Яковенко

Керівник роботи
Корченко Анна Олександрівна
доктор технічних наук, професор

Житомир – 2023

Висновок кафедри

_____

за результатами попереднього захисту:

_____

Протокол засідання кафедри

_____

№ _____від «_____» _____20____р.

Завідувач кафедри

_____

_____ _____ _____

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

«_____» _____20____р.

Результати захисту кваліфікаційної роботи

Здобувач вищої освіти _____захистив (ла)

(прізвище ,ім'я, по батькові)

кваліфікаційну роботу з оцінкою:

сума балів за 100-бальною шкалою _____

за шкалою ECTS _____

за національною шкалою _____

Секретар ЕК

_____ _____ _____

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

АНОТАЦІЯ

Яковенко В. Б. Програмна система кібербезпекового аналізу мережевого трафіку комерційного банку «Credit Agricole». – Кваліфікаційна робота на правах рукопису.

Кваліфікаційна робота на здобуття освітнього ступеня магістр за спеціальністю 125 – «Кібербезпека». – Поліський національний університет, Житомир, 2023.

Мета: розробка програмної системи аналізу мережевого трафіку для виявлення підозрілого трафіку.

Завдання:

провести аналіз програмних та програмно-апаратних систем, що аналізують мережевий трафік; розробити програмну систему кібербезпекового аналізу мережевого трафіку; провести експериментальне дослідження запропонованого рішення.

Використана методика дослідження: теоретичний аналіз, опис, проектування, програмування.

Ключові слова:

Аналізатор мережевого трафіку, програмна система, порівняння програм-аналізаторів, структурна схема, блок-схема алгоритму функціонування, Visual Studio, C#, Windows Forms, WinPcap, SharpPcap, PacketDotNet, LibPcap, графічний інтерфейс, клас, метод, шифрування даних, дослідження функціональних можливостей.

SUMMARY

Yakovenko V. B. Software System for Cybersecurity Analysis of Network Traffic in «Credit Agricole» Commercial Bank. – Qualification work on the manuscript rights.

Qualification work for the acquisition of the second (master's) level of higher education in the specialty 125 «Cybersecurity». – Polissia National University, Zhytomyr, 2023.

Objective: The development of a software system for analyzing network traffic to detect suspicious activities.

Tasks: conduct an analysis of software and hardware systems that analyze network traffic; develop a software system for cybersecurity analysis of network traffic; conduct experimental research on the proposed solution.

Research Methodology: Theoretical analysis, description, design, programming.

Keywords:

Network traffic analyzer, software system, comparison of analyzers, structural diagram, algorithm functioning block diagram, Visual Studio, C#, Windows Forms, WinPcap, SharpPcap, PacketDotNet, LibPcap, graphical interface, class, method, data encryption, investigation of functional capabilities.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

INTA – Iris Network Traffic Analyzer;

ITCE – Iris Traffic Capture EngineTM;

EIS – Ethernet Internet traffic Statistic;

WPF – Windows Presentation Foundation;

API – Application Programming Interface.

ВСТУП

У сучасному світі безперервний прогрес новітніх технологій нерозривно пов’язаний з появою все досконаліших загроз, які спрямовані на компрометацію дуже цінної інформації. Забезпечення кібербезпеки та захисту мережевої інфраструктури банку стало актуальним як ніколи раніше і потреби у створенні відповідних програмних систем неперервно зростають. Необхідність захисту інформації банку, що становить банківську таємницю є важливим аспектом ефективного функціонування, тому потреба і актуальність в захисті мережевої інфраструктури банку є цілком обґрунтованою.

Призначення проекту: розроблена програмна система аналізу мережевого трафіку дозволить аналізувати мережевий трафік відділення банку кібербезпеки та вчасно виявляти потенційно-небезпечні підключення.

Метою і завданням кваліфікаційної роботи є розробка програмної системи аналізу мережевого трафіку.

Об’єктом дослідження є процес аналізу мережевого трафіку та можливості виявлення потенційно-небезпечного трафіку.

Предметом дослідження є методи, системи та програмні засоби, які використовуються для аналізу мережевого трафіку.

Методи дослідження: теоретичний аналіз, опис, проектування, програмування.

Перелік публікацій автора за темою дослідження: аналіз стану кібербезпеки та захисту інформації філії комерційного банку, аналіз особливостей програмних систем для аналізу мережевого трафіку, розробка програмної системи аналізу мережевого трафіку для вдосконалення захисту інформації.

Наукова новизна: вперше розроблена програмна система кібербезпекового аналізу мережевого трафіку комерційного банку «Credit Agricole» за рахунок здатності перехоплювати мережевий трафік в режимі підвищеного захисту, можливості вибору між звичайним та захищеним режимом, а також у відкритому або зашифрованому вигляді відображати отримані дані, що дає можливість

виявляти потенційно-небезпечний трафік та виконувати моніторинг трафіку з урахуванням безпеки мережі.

Практичне значення отриманих результатів: розроблене алгоритмічне та програмне забезпечення аналізу мережевого трафіку, що виявляє потенційно-небезпечний трафік, який може зашкодити стабільній роботі мережевої інфраструктури відділення банку. В залежності від режиму роботи застосунок може шифрувати інформацію та створювати файли інформації, які містять зібрані дані.

Розділ 1. ПЕРЕГЛЯД АНАЛОГІЧНИХ ІСНУЮЧИХ СИСТЕМ. ПРИЗНАЧЕННЯ ТА ОБЛАСТЬ ВИКОРИСТАННЯ ПРОГРАМНОЇ СИСТЕМИ

1.1 Аналіз програмних (програмно-апаратних) систем, які аналізують мережевий трафік

Перед створенням програмної системи важливо визначити наскільки вона є актуальною на даний проміжок часу. Найефективнішим методом визначення актуальності (новизни) є аналіз існуючих аналогів та визначення їх можливостей.

Аналізатор трафіку, або сніфер – мережевий аналізатор трафіку, програма або програмно-апаратний пристрій, призначений для перехоплення і подальшого аналізу, або тільки аналізу мережевого трафіку, призначеного для інших вузлів [1] – [4].

Wireshark (Ethereal) – це програмний аналізатор, який дозволяє користувачу спостерігати за потоком даних, що проходить через мережу в реальному часі.

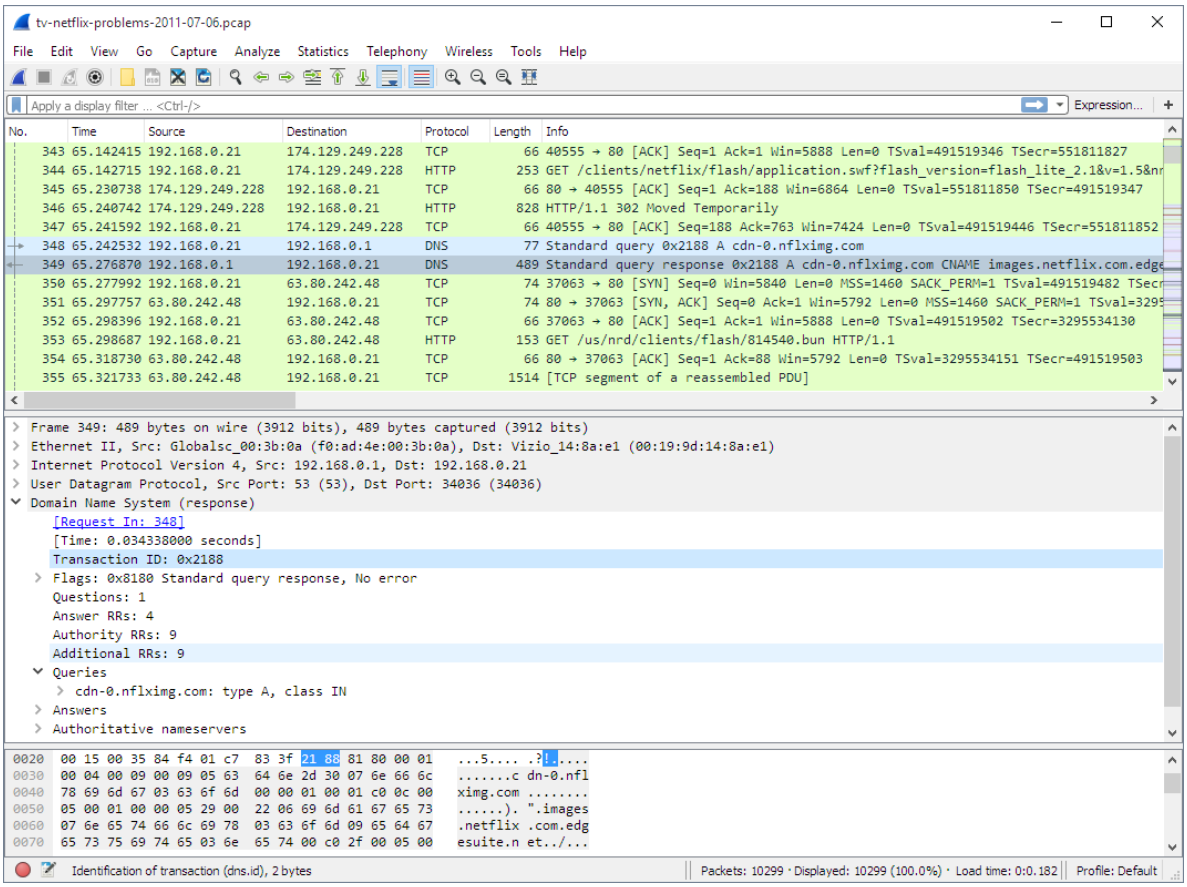

Рис. 1.1 – Wireshark

Wireshark має глибоке розуміння різних мережевих протоколів, що дозволяє розшифровувати мережеві пакети і виводити значення кожного поля протоколу на будь-якому рівні. Ця програма підтримує різноманітні формати вхідних даних і може аналізувати файли даних, захоплені іншими програмами, розширюючи можливості моніторингу мережі [5] – [8].

Можливості Wireshark включають: сортування та фільтрацію інформації, можливість спостереження за мережевим трафіком в режимі реального часу, розпізнавання структури мережевих протоколів, виявлення та вирішення проблем в мережі, а також можливість детального перегляду вмісту мережевого пакета на всіх рівнях мережі [9] – [12].

CommTraffic – це програмний застосунок для моніторингу та аналізу мережевого трафіку, призначений для використання як у локальних, так і у комутованих комп’ютерних мережах. Під час спостереження за локальною мережею, CommTraffic надає інформацію про обсяг трафіку та використання мережі для кожного комп’ютера в цьому сегменті.

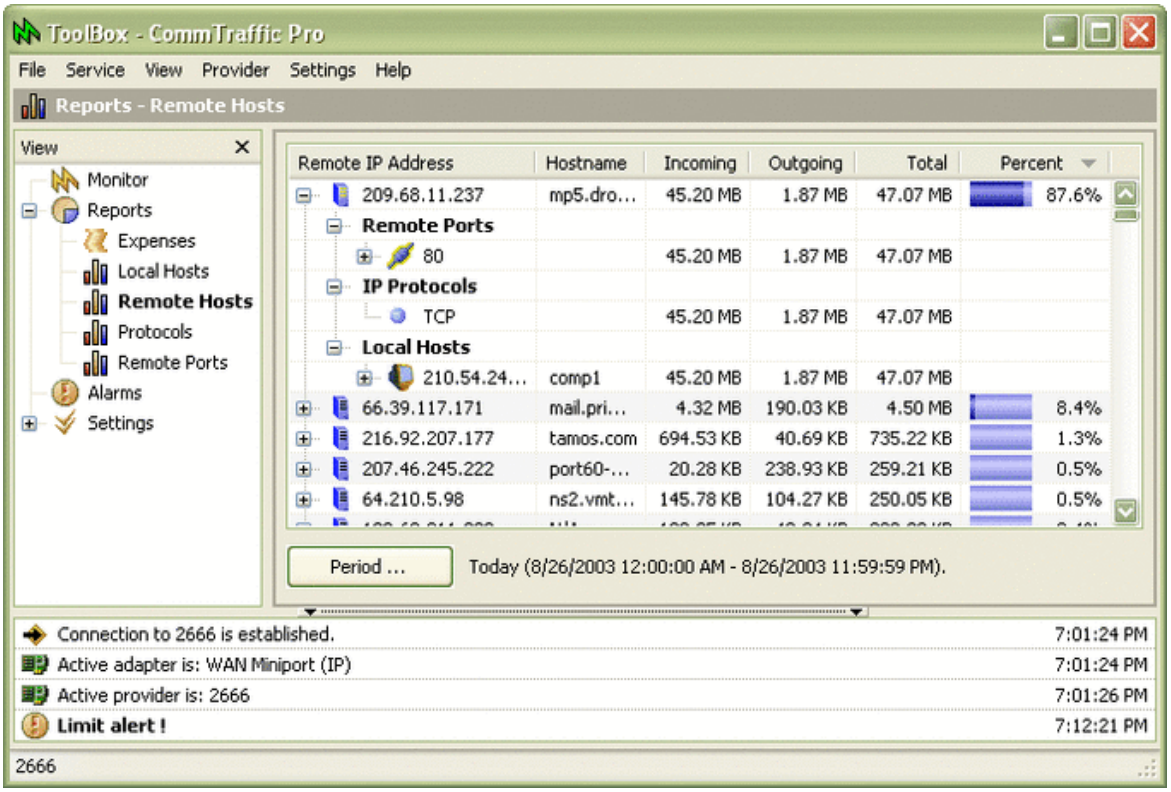

Рис. 1.2 – CommTrafic

CommTraffic має інтуїтивний інтерфейс, який легко налаштовується та надає статистику у графічному та числовому вигляді. Головне вікно програми можна налаштувати для відображення вхідної, інформації про вхідну, вихідну та загальну статистику трафіку в різних стилях.

Функціональні можливості CommTraffic: встановлення обмежень часу та трафіку відповідно до тарифного плану, нагадування про можливі перевищення витрат, створення звітів, які відображають обсяг мережевого трафіку та витрати на інтернет-підключення, а також аналіз статистики трафіку для віддалених та локальних хостів, IP-протоколів і локальних / віддалених TCP/UDP-портів [13] – [16].

Iris Network Traffic Analyzer – надійний інструмент, орієнтований на моніторинг та аналіз мережевого трафіку з виявленням аномалій. В першу чергу програма призначена для системних адміністраторів і фахівців з мережевої безпеки, дозволяючи їм проводити комплексну оцінку своїх мереж і виявляти потенційні вразливості.

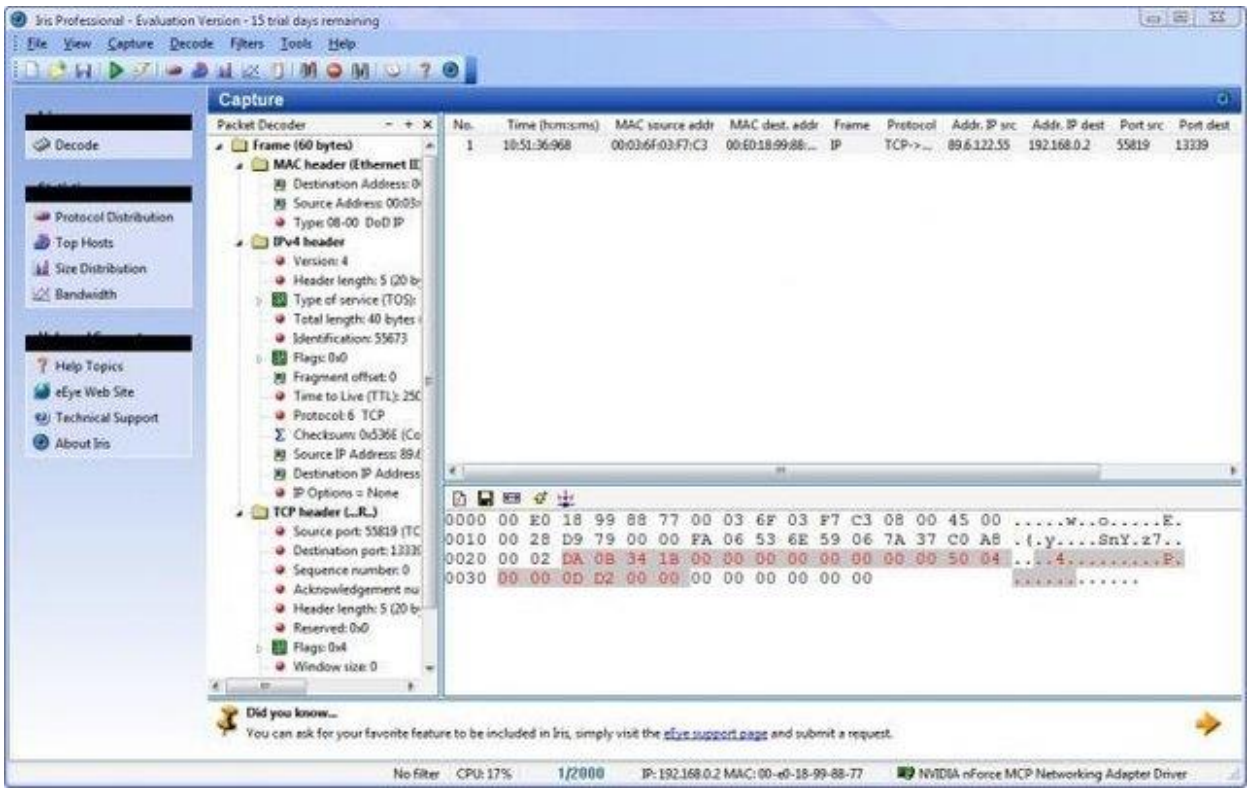

Рис. 1.3 – Iris Network Traffic Analyzer

Крім того, після проведення ретельного аналізу програма надає професійні рекомендації, адаптовані до конкретної ситуації.

Можливості INTA:

- Неперервне захоплення трафіку: Iris Traffic Capture EngineTM (ITCE) працює у фоновому режимі для збору інформації, незалежно від операційної системи або сесії користувача, що дозволяє захоплювати увесь цільовий трафік.
- Повна реконструкція прийнятих і переданих пакетів – має можливість відновлювати прийняті та відправлені пакети, включаючи файли та веб-сторінки, переглянуті під час сесії, у їхньому первісному форматі. Це насамперед дозволяє точно відновити всі події, включаючи натискання клавіш, які відбулися під час сесії.
- Перехоплення пакетів і Spoofing – програма дозволяє виявляти спуфінг пакетів, проводити тестування конфігурації брандмауера і реєструвати свідоцтва мережевих вторгнень. Інструмент також збирає в реальному часі повні журнали всіх тестів та будь-якої шкідливої активності.
- Моніторинг електронної пошти та месенджерів миттєвих повідомлень – здійснює моніторинг нешифрованого трафіку електронної пошти та миттєвих повідомлень, розширюючи можливості звичайної електронної пошти та месенджерів [17] – [21].

Ethernet Internet Traffic Statistic - безкоштовна програма для моніторингу інтернет-трафіку, сумісна з ADSL, LAN, Wi-Fi і Bluetooth з’єднаннями. Програма не потребує інсталяції і має інтерфейс, що доступний англійською та російською мовами.

Можливості EIS: відображення кількості отриманих і відправлених даних, моніторинг швидкості передачі, збір даних в режимі реального часу і створення графіків на основі зібраної інформації [22, 23].

Кожна з цих програм має різноманітний функціонал та можливості для використання. Проведемо порівняння даних аналогів для визначення їх переваг та недоліків в табл. 1.1.

Короткий аналіз було здійснено і щодо інших програм-аналогів, що виконують аналіз мережевого трафіку, проте їхній функціонал досить подібний і не є настільки різноманітним, тому здійснювати їх порівняння недоречно [24] – [29].

Табл. 1.1 – Порівняння програм-аналізаторів мережевого трафіку

		Параметри					
		Побудова графіку швидкості передачі даних	Побудова графіку переміщення трафіку	Імпорт / Експорт даних	Запуск моніторингу за вимогою	Зміна відображення даних у відкритому або зашифрованому вигляді	Можливість зміни мінімального кроку між звітами даних
Програми	Wireshark	+	-	- / +	-	-	+
	CommTraffic	-	+	- / -	-	-	-
	INTA	+	-	- / -	-	-	+
	EIS	-	+	- / -	-	-	-

Порівняння було здійснено за такими параметрами: побудова графіку швидкості передачі даних, побудова графіку переміщення трафіку, імпорт / експорт даних, запуск моніторингу за вимогою, можливість змінювати поріг доступу до сервера та можливість зміни мінімального кроку між звітами даних.

Побудова графіку швидкості передачі даних – можливість створити графік, що показує швидкість передачі даних.

Побудова графіку переміщення трафіку – графік, який показує переміщення даних по мережі в конкретний проміжок часу.

Імпорт / Експорт даних – це процес автоматичного або напівавтоматичного введення та виведення наборів даних між різними програмами. Цей процес включає

«перетворення» даних з формату, який використовується в одній програмі, в формат, який використовується в іншій програмі. Таке перетворення часто виконується автоматично за допомогою машинних процесів, таких як транскодування або конвертація даних [30] – [34].

Моніторинг мережі – це ключовий елемент управління мережею, який включає в себе контроль та аналіз продуктивності та функціонування мережевих компонентів з метою забезпечення оптимальної ефективності роботи мережі [35] – [39]. Запуск моніторингу за вимогою виконує моніторинг мережі в потрібний користувачу момент часу.

Зміна відображення даних у відкритому або зашифрованому вигляді – режими роботи програми, за яких перехоплені мережеві дані відображаються у списку або у відкритому, або у зашифрованому вигляді.

Можливість зміни мінімального кроку між звітами даних – обмеження, яке визначає інтервал між створенням звітів даних.

1.2 Призначення та область використання програмної системи

Програмна система, яку буде створено, призначена для аналізу мережевого трафіку серверу, областю використання якого є банківська діяльність. Даний застосунок матиме можливість аналізувати мережевий трафік банківського серверу, проводити моніторинг мережевого трафіку, а також особливою його складовою стануть режими роботи програми, в звичайному режимі дані будуть відображатися у відкритому вигляді, а в захищеному режимі – шифруватися та відображатись у зашифрованому вигляді.

Висновки до першого розділу

Проведено аналіз програмних систем, що аналізують мережевий трафік, в результаті чого стало зрозуміло, що переглянуті системи мають різноманітний функціонал та можливості для використання. Використовуючи просту таблицю та загальні параметри для порівняння, вдалося ідентифікувати переваги та недоліки кожного застосунку.

Розділ 2. РОЗРОБКА ПРОГРАМНОЇ СИСТЕМИ АНАЛІЗУ МЕРЕЖЕВОГО ТРАФІКУ ДЛЯ ПОКРАЩЕННЯ СИСТЕМИ ЗАХИСТУ ІНФОРМАЦІЇ

2.1 Розроблення структурної і блок-схеми алгоритму функціонування програмної системи

Проведений аналіз існуючих програмних систем аналізу мережевого трафіку та визначення їх недоліків дає змогу розробити програмну систему, яка в своїй реалізації матиме власну відмінну характеристику від вже існуючих програмних систем.

Перш за все перед створенням будь-якої програмної системи необхідно сформулювати власну стратегію (послідовність) того, як саме програма повинна функціонувати і з чим вона має взаємодіяти. Для створення такої послідовності виконання роботи програмою, будується структурна схема роботи програмного застосунку та блок-схема алгоритму функціонування.

Структурна схема є графічним зображенням структури програмного застосунку, що відображає послідовність виконання операцій, логічні зв’язки та взаємодію частин коду програми.

Створена структурна схема програмної системи зображена на рис. 2.1.

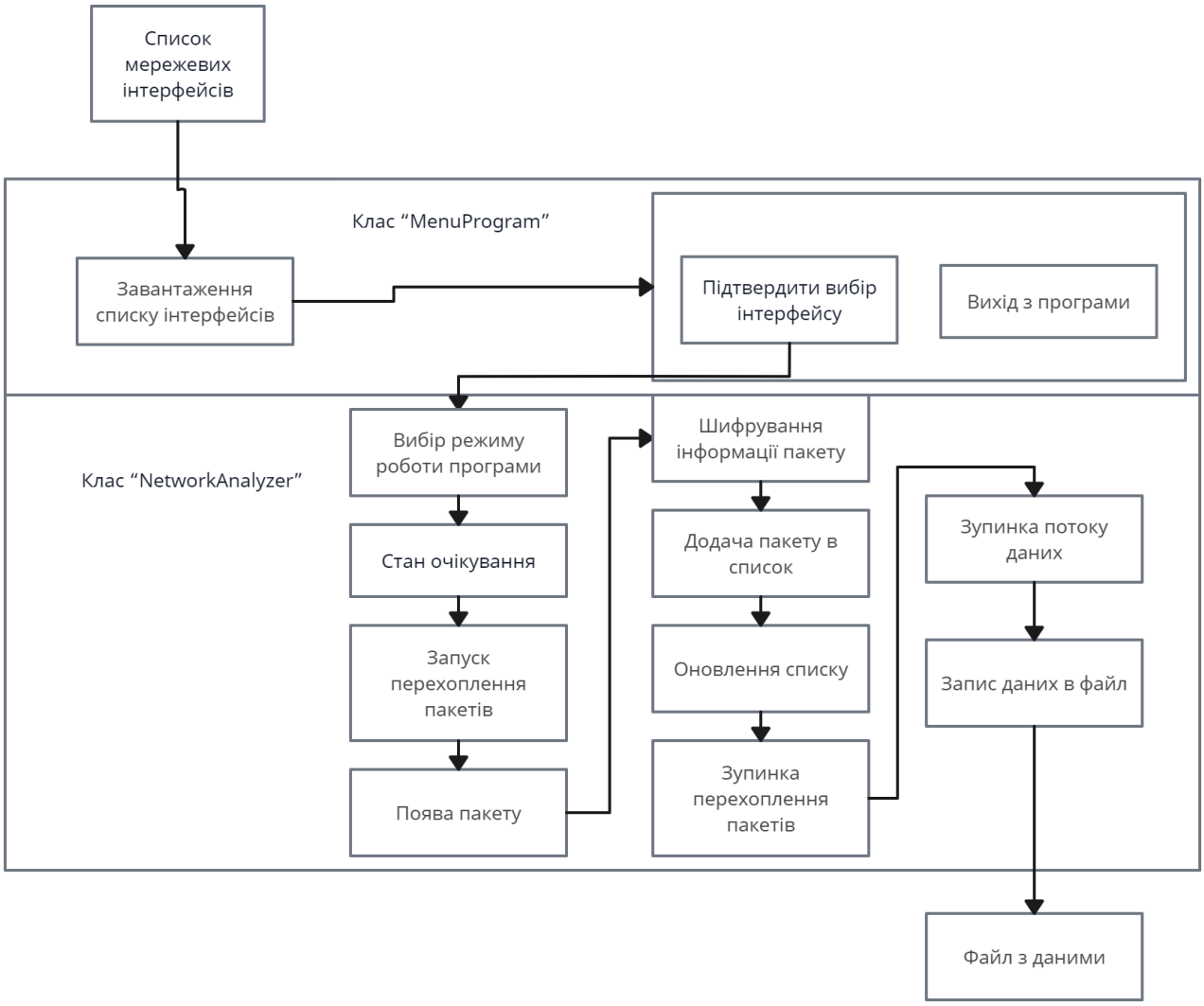

Рис. 2.1 – структурна схема програмного застосунку

На початку після запуску застосунку ідентифікується список мережевих інтерфейсів пристрою, після чого розпочинається робота класу «MenuProgram», який завантажує ідентифікований список мережевих інтерфейсів.

Далі залежно від вибору або відбувається вихід із програми, або підтверджується вибір одного із мережевих інтерфейсів для роботи з ним.

Після вибору інтерфейсу програми, виконується робота класу «NetworkAnalyzer». Спочатку потрібно вибрати режим роботи програми. Коли режим вибрано, програма переходить в стан очікування, коли нічого не відбувається. Якщо натиснути на кнопку «Запуск», то відбудеться процес запуску перехоплення пакетів.

Тепер, коли перехоплення пакетів розпочалось, відбуваються послідовні процеси появи пакету, додачі пакету в список, оновлення списку.

Якщо було вибрано захищений режим роботи програми, то інформація буде шифруватися в момент появи пакетів.

Ці послідовні процеси можуть виконуватись безперервно, доки не буде натиснута кнопка «Зупинити», що реалізує процес зупинки перехоплення пакетів.

Результатом зупинки перехоплення пакетів є послідовні процеси зупинки потоку даних, запису даних в файл та в кінцевому результаті отримання файлу з даними.

Блок-схема алгоритму функціонування є процесом розв’язання поставленої мети в графічному вигляді з використанням блоків позначення потоків, даних та операцій.

Побудована блок-схема алгоритму аналізатора мережевого трафіку зображена на рис. 2.2.

Рис. 2.2 – блок-схема алгоритму програмного застосунку

Блок-схема програмного застосунку чітко відображає як функціонує сам застосунок.

Перше, що необхідно зробити після запуску застосунку – вибрати мережевий інтерфейс для аналізу трафіку, тому існує умова того, що якщо інтерфейс не було обрано, то його обов’язково потрібно вибрати, до того моменту застосунок не буде виконувати свої функції.

Якщо ж інтерфейс було обрано, то розпочинається перехоплення пакетів даних вибраного інтерфейсу. В нашому випадку таким інтерфейсом буде сервер банку. Після початку захоплення пакетів перевіряється умова на те, чи вперше було запущено процес, чи ні. Якщо процес було запущено не вперше, то програма очистить вже наявний список даних перед тим, як почати новий запис, а вже потім

запустить процеси перехоплення пакетів. В випадку, коли процес було запущено вперше і список пустий, то запускаються наступні процеси: перехоплення пакетів даних, додавання пакетів даних в список, після чого список оновлюється.

В випадку, якщо відбулася зупинка перехоплення пакетів, то відбувається наступне: перевіряється чи ще є інформація в потоці. Якщо інформація ще присутня, тоді спочатку потрібно очистити потік, а вже потім зупинити його. В випадку відсутності інформації в потоці, відбувається тільки зупинка потоку, після чого незалежно від того, який випадок відбувся, програма виконує запис отриманих даних в файл.

2.2 Розроблення опису програмного застосунку та його реалізація

Заздалегідь створена структурна схема та блок-схема алгоритму відкрили можливість створити програмну систему. Для виконання задачі було вирішено використовувати Microsoft Visual Studio як основне середовище розробки. Microsoft Visual Studio – комплексний спеціалізований інструмент з підтримкою різноманітних технологій, що дозволяє використовувати для розробки безліч мов програмування таких як C, C++, C#, Python, JavaScript та інші, створювати графічні інтерфейси, підключати існуючі бібліотеки для більш легкого досягнення поставленої мети.

Розуміння того, що застосунок повинен аналізувати мережевий трафік та працювати з мережевими протоколами, інтерфейсами і мережею в цілому спонукало використовувати керовану мову програмування C#. Ця мова є максимально об’єктно-орієнтованою, оптимально адаптованою до роботи з мережею, що дозволяє використовувати бібліотеки та реалізовувати такі графічні інтерфейси як Windows Forms або Windows Presentation Foundation. Навіть за умови того, що C# є високорівневою мовою програмування, її збалансована позиція між мовами низького (мовами ядра) та високого (прикладного) рівня дозволяє забезпечити ефективний зв’язок з ядром операційної системи та графічним інтерфейсом, що стало ключовим аспектом при виборі мови програмування.

Побудова застосунку аналізу мережевого трафіку з використанням мови програмування C# дозволяє створити власний графічний інтерфейс, оскільки варіант консольного відображення є недоцільним в такому випадку через обмежені функціональні можливості.

Для реалізації програмної системи було вирішено використовувати графічний інтерфейс Windows Forms, який, незважаючи на власні функціональні недоліки перед новим інтерфейсом WPF, має переваги в швидкій побудові інтерфейсу і більш простому налаштуванні.

Після обрання необхідних компонентів робота над розробкою застосунку перейшла на етап реалізації. Розпочалось створення проекту застосунку на основі C# з графічним інтерфейсом Windows Forms.

Інтерфейс в даному випадку є графічним вікном, в якому розміщені робочі області, елементи управління та поля для виведення інформації (див. рис. 2.1).

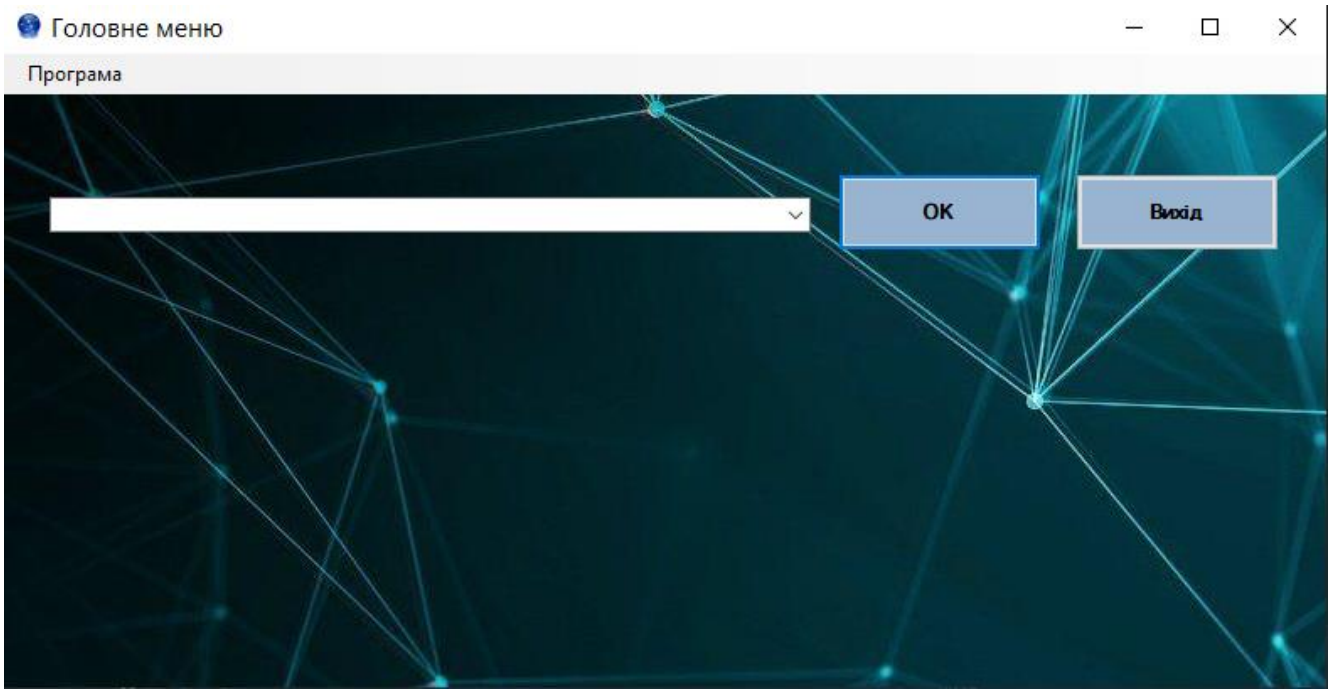

Рис. 2.1 – перша вкладка графічного інтерфейсу програми

Застосунок містить як логічну так і графічну складову, а також інтегровані в проект бібліотеки прискорення роботи з мережею.

Логічна складова застосунку містить процес контролю мережевого трафіку, що поділяється на два етапи: моніторинг і аналіз. Під час моніторингу відбувається збір первинних даних, що надають інформацію про кількість пакетів, час захоплення, адреси джерел з яких пакети були відправлені, адреси місць призначення пакетів, використовувані протоколи захоплених пакетів та об’єм (довжина) пакетів даних. Етап аналізу включає в себе вибірковий перегляд детальної інформації того чи іншого пакету даних.

На рис. 2.2 зображена графічна складова застосунку, що містить кнопки, поля відображення та робочі області.

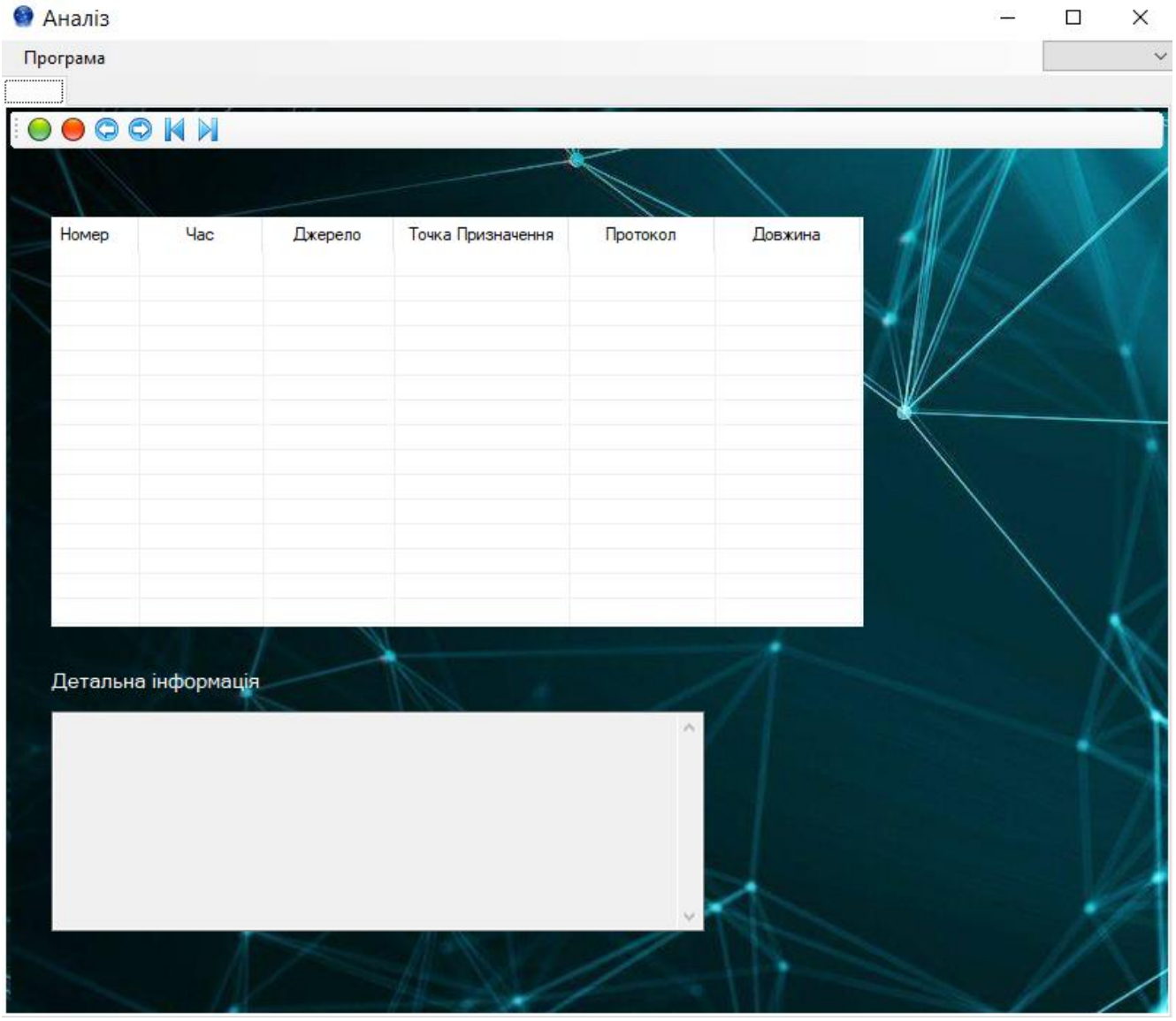

Рис. 2.2 – друга вкладка графічного інтерфейсу

Інтегровані бібліотеки SharpPcap та PacketDotNet є невід’ємною та важливою складовою для забезпечення дієздатності додатку, вони допомагають налаштувати зв’язки з мережевим інтерфейсом та спростити захоплення мережевих пакетів прямо з мережі. Ці бібліотеки потребують також використання технології захоплення пакетів WinPcap, яка встановлюється окремо.

WinPcap – це технологія, призначена для перехоплення мережевих пакетів перед їх отриманням мережевою картою. У наборі інструментів WinPcap включені аналізатори протоколів, мережеві монітори, системи виявлення мережевих вторгнень, генератори трафіку, утиліти для тестування мережі та інші засоби. Виконує фільтрацію пакетів на рівні ядра, обладнана функціоналом для збору статистики мережі і підтримує можливість віддаленого захоплення пакетів. WinPcap включає в себе драйвери для зчитування пакетів, що надходять на мережеву карту, а також низькорівневі бібліотеки, які дозволяють взаємодіяти з мережевими драйверами.

SharpPcap – це бібліотека перехоплення пакетів для середовища .NET, яка підтримує операційні системи Windows, Mac і Linux. Він має API для виконання операцій з перехоплення, ін’єкції, аналізу та збору пакетів і може використовуватися з будь-якою мовою платформи .NET, такою як C# або VB.NET.

SharpPcap є комплексною бібліотекою, що має як власний функціонал, так і містить в собі підкласи, які потрібно підключати окремо. Однією з таких під класових бібліотек, яка використовується для реалізації додатку є LibPcap.

LibPcap – це бібліотека, яка надає API для отримання пакетів безпосередньо з мережі. Подібно до бібліотеки SharpPcap, LibPcap спеціалізується на роботі з мережею, але безпосередньо виявляє перелік існуючих мережевих пристроїв для взаємодії з ними, в той час як SharpPcap зосереджується на виконанні інших завдань.

Після створення логічної та графічної складової застосунку, варто зосередитись на класах та методах, які виконують необхідні для роботи функції.

Всього застосунок містить 2 класи: MenuProgram та NetworkAnalyzer. Розглянемо їх відмінності нижче в таблиці 2.1.

Табл. 2.1 – Класи застосунку та їх опис

Клас	Опис
MenuProgram	Відповідає за головне меню застосунку та вибір мережевого пристрою, з яким буде виконуватись робота.
NetworkAnalyzer	Виконує основну частину роботи застосунку, а саме: запуск та зупинку перехоплення пакетів, створення переліку перехоплених пакетів у вигляді списку, перегляд детальної інформації кожного перехопленого пакету та можливість вибрати пакет із наявного списку.

У таблицях 2.2 та 2.3 наведено перелік основних методів, що виконують важливі функції в існуючих класах.

Табл. 2.2 – Основні методи класу MenuProgram та їх опис

Метод	Опис
MenuProgram_Load	Відповідає за відображення списку мережевих інтерфейсів та виборі одного з них.
get_Interface	Працює як кнопка, при натисканні якої підтверджується вибір мережевого пристрою для подальшої роботи з ним.

Табл. 2.3 – Основні методи класу NetworkAnalyzer та їх опис

Метод	Опис
StartCapture	Розпочинає процес перехоплення пакетів.
ListView_ItemSelectionChanged	Змінює стан списку, заповнюючи його отриманими даними.
StopCapture	Завершує процес перехоплення пакетів.
PreviousPacket	Переходить до попереднього пакету в списку.
NextPacket	Переходить до наступного пакету в списку.
ToTheTop	Переходить на початковий перехоплений пакет даних в списку.

ToTheEnd	Переходить на кінцевий перехоплений пакет даних в списку.
Capture_Proccess	Виконує сам процес перехоплення пакетів.
Encryptor	Виконує шифрування даних перехопленого пакету.
Device_OnPacketArrival	Отримує та надає параметри для формування списку даних.

Висновки до другого розділу

Розроблена програмна система аналізу мережевого трафіку комерційного банку «Credit Agricole», що дає можливість виявляти потенційно-небезпечний трафік, який впливає на роботу мережевої інфраструктури відділення банку та отримувати зібрані дані мережевого трафіку у вигляді файлу.

Розділ 3. ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ ПРОГРАМНОЇ СИСТЕМИ АНАЛІЗУ МЕРЕЖЕВОГО ТРАФІКУ

3.1 Дослідження функціональних можливостей програмного застосунку

Після реалізації програмного застосунку варто обов’язково дослідити його функціональні можливості. При запуску застосунку відображається створений графічний інтерфейс класу MenuProgram, що відповідає формі головного меню.

Головне меню – це зображена на рис. 3.1 вкладка інтерфейсу, на якій розміщено поле для вибору мережевого інтерфейсу, реалізовані кнопки для підтвердження вибору або виходу з програми. Також форма містить меню «Програма», через яке можна вийти з програми.

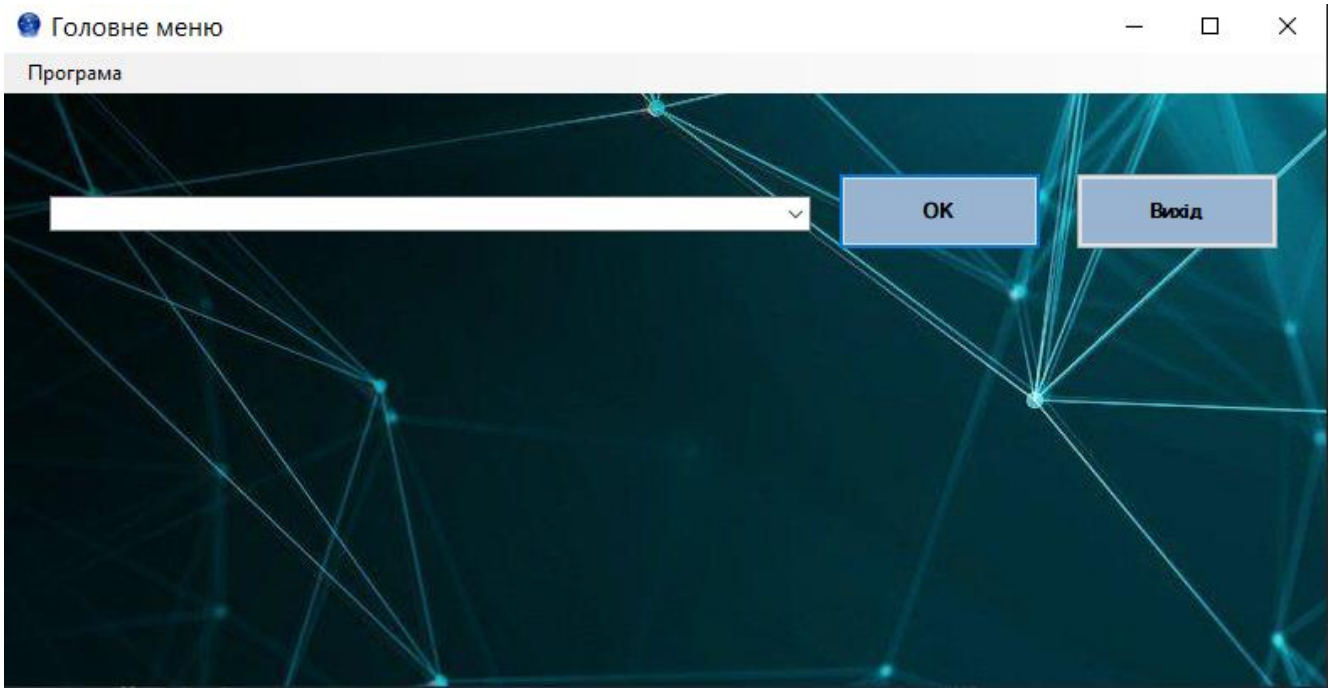

Рис. 3.1 – головна форма інтерфейсу

Після вибору мережевого інтерфейсу, програма почне відображати вкладку графічного інтерфейсу «Аналіз», що безпосередньо є частиною класу NetworkAnalyzer, в якій і знаходяться основні функціональні можливості створеного застосунку (див. рис. 3.2).

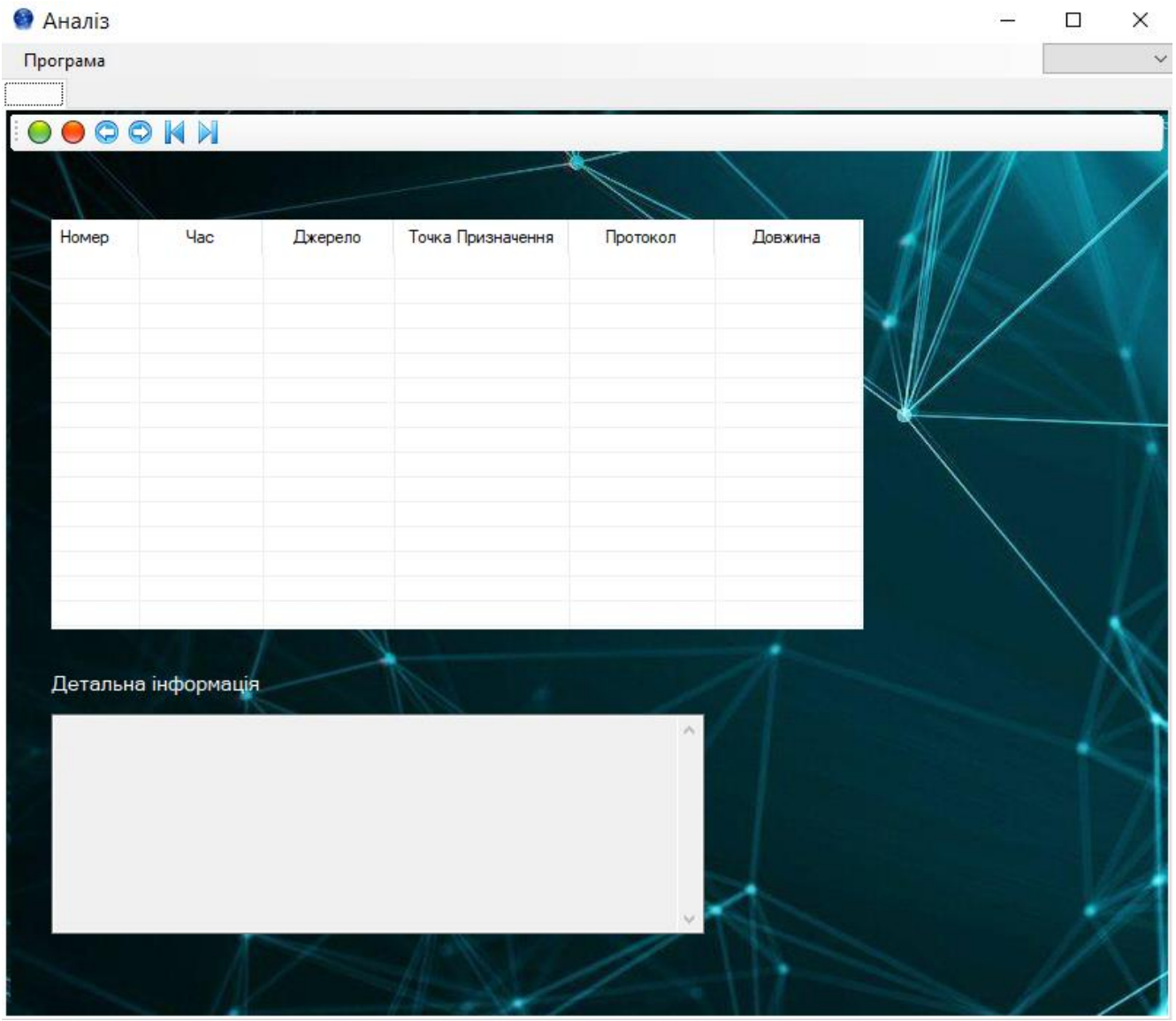

Рис. 3.2 – графічний інтерфейс вкладки «Аналіз»

На даній вкладці розміщений весь основний функціонал: є можливість перейти в меню «Програма», де можна як вийти з застосунку, так і повернутись на попередню вкладку, щоб змінити мережевий інтерфейс (див. рис. 3.3).

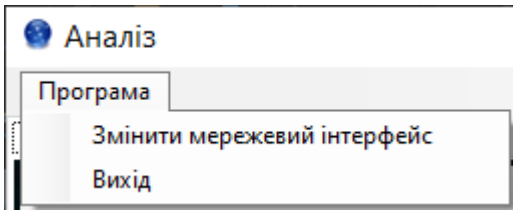

Рис. 3.3 – меню «Програма»

Не менш важливим є поле вибору режиму роботи програми (див. рис. 3.4), в залежності від того, який вибрано режим, програма відображатиме дані в відкритому або зашифрованому вигляді.

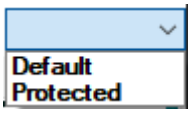

Рис. 3.4 – поле вибору режиму роботи програми

Крім цього головною частиною функціоналу застосунку є поле з кнопками, що виконують основні дії застосунку (див. рис. 3.5).

Рис. 3.5 – функціональне поле кнопок застосунку

Кожна з кнопок відповідає функції свого методу, що був описаний в підпункті 2.2.

Зелена кнопка (метод StartCapture) – це запуск перехоплення пакетів даних.

Червона кнопка (метод StopCapture) – здійснює завершення перехоплення.

Кнопка «вліво» (метод PreviousPacket) – виконує переміщення на попередній пакет в визначеному списку даних, що був сформований в результаті перехоплення пакетів даних.

Кнопка «вправо» (метод NextPacket) – подібно до попереднього виконує переміщення по списку, але на наступний пакет.

Кнопка «стрілка наліво впритул» (метод ToTheTop) – виконує переміщення до початкового пакету даних в списку.

Кнопка «стрілка направо впритул» (метод ToTheEnd) – виконує переміщення до кінцевого пакету даних в списку.

Після опису функціональних можливостей цього поля з кнопками, проведемо тестовий запуск перехоплення пакетів даних.

На рис. 3.6 можна побачити, що перехоплення пакетів в звичайному режимі роботи відбулося вдало і ми маємо створений відкритий список даних, який заповнений перехопленими пакетами даних з комп’ютерної мережі.

Номер	Час	Джерело	Точка Призначення	Протокол	Довжина
1	15:2:10:487	172.20.10.3	239.255.255.250	Udp	209
2	15:2:10:859	172.20.10.3	20.42.73.27	Tcp	266
3	15:2:11:370	172.20.10.3	20.42.73.27	Tcp	1355
4	15:2:11:495	172.20.10.3	239.255.255.250	Udp	209
5	15:2:11:868	172.20.10.3	20.42.73.27	Tcp	1355
6	15:2:12:261	20.42.73.27	172.20.10.3	Tcp	120
7	15:2:12:321	172.20.10.3	20.42.73.27	Tcp	66
8	15:2:12:333	20.42.73.27	172.20.10.3	Tcp	78
9	15:2:12:360	20.42.73.27	172.20.10.3	Tcp	78
10	15:2:12:509	172.20.10.3	239.255.255.250	Udp	209
11	15:2:13:453	20.42.73.27	172.20.10.3	Tcp	484
12	15:2:13:458	172.20.10.3	20.42.73.27	Tcp	266
13	15:2:13:496	172.20.10.3	172.20.10.1	Udp	72
14	15:2:13:524	172.20.10.3	239.255.255.250	Udp	209

Рис. 3.6 – тестовий запуск перехоплення пакетів даних в звичайному режимі роботи

Використання кнопок для переміщення по списку було заздалегідь обмежено умовами, щоб застосунок не мав шпарин для потенційних вразливостей і раптового вимкнення.

В полі «Детальна інформація» можна отримати подробиці по даним щодо вибраного пакету даних в списку, що зображено на рис. 3.7.

Рис. 3.7 – детальна інформація про вибраний пакет даних

Обов’язково варто перевірити і дієздатність роботи застосунку в захищеному режимі. Після проведення перехоплення мережевих пакетів маємо список даних в зашифрованому вигляді (див. рис. 3.8). Детальна інформація про пакети даних під час роботи програми в даному режимі не передбачена.

Номер	Час	Джерело	Точка Призначення	Протокол	Довжина
1	cbhl4s0noxn...	8Jlitf0lCJh7xx...	HlxdrlZuZp2fhODU+...	DVcowRkMpUc=	RX0KXCn45m0
2	cbhl4s0noxn...	8Jlitf0lCJh7xx...	HlxdrlZuZp2fhODU+...	DVcowRkMpUc=	+rBdCZug26Y=
3	cbhl4s0noxn...	HlxdrlZuZp2fh...	8Jlitf0lCJh7xxFPlqS...	DVcowRkMpUc=	JsiTFrOH/UU=
4	s+aacgbZpal...	HlxdrlZuZp2fh...	by4Eaf534F/oDrL9g...	DVcowRkMpUc=	8oDxJGzCO6Y=
5	s+aacgbZpal...	HlxdrlZuZp2fh...	by4Eaf534F/oDrL9g...	DVcowRkMpUc=	8oDxJGzCO6Y=
6	/HZ0vwGzM...	by4Eaf534F/o...	HlxdrlZuZp2fhODU+...	DVcowRkMpUc=	JsiTFrOH/UU=
7	/HZ0vwGzM...	by4Eaf534F/o...	HlxdrlZuZp2fhODU+...	DVcowRkMpUc=	JsiTFrOH/UU=
8	/HZ0vwGzM...	HlxdrlZuZp2fh...	u7wLT+GWT9aWk...	9wuYyfM5cY0=	MxjPFLwxEqM=
9	/HZ0vwGzM...	HlxdrlZuZp2fh...	u7wLT+GWT9aWk...	9wuYyfM5cY0=	3yQRVhNsEpk=
10	/HZ0vwGzM...	by4Eaf534F/o...	HlxdrlZuZp2fhODU+...	DVcowRkMpUc=	cSCNC7a3Cdl=
11	/HZ0vwGzM...	by4Eaf534F/o...	HlxdrlZuZp2fhODU+...	DVcowRkMpUc=	cSCNC7a3Cdl=
12	MSQzt39ZNs...	8Jlitf0lCJh7xx...	HlxdrlZuZp2fhODU+...	DVcowRkMpUc=	v8RXEAXFit8=
13	jdvj60sWpYV...	HlxdrlZuZp2fh...	by4Eaf534F/oDrL9g...	DVcowRkMpUc=	JsiTFrOH/UU=
14	jdvj60sWpYV...	HlxdrlZuZp2fh...	by4Eaf534F/oDrL9g...	DVcowRkMpUc=	JsiTFrOH/UU=

Рис. 3.8 – запуск перехоплення мережевих пакетів в захищеному режимі роботи

Висновки до третього розділу

Проведено дослідження програмної системи аналізу мережевого трафіку, що дало змогу впевнитись в унікальності наявних функціональних можливостей застосунку. Крім того, експериментальне дослідження підтвердило роботу програмного застосунку, що, як наслідок, дає можливість використовувати розроблене програмне забезпечення в повному обсязі.

ВИСНОВКИ

Проведено аналіз програмних систем, що аналізують мережевий трафік, в результаті чого стало зрозуміло, що переглянуті системи мають різноманітний функціонал та можливості для використання. Використовуючи просту таблицю та загальні параметри для порівняння, вдалося ідентифікувати переваги та недоліки кожного застосунку.

Розроблена програмна система аналізу мережевого трафіку комерційного банку «Credit Agricole», що дає можливість виявляти потенційно-небезпечний трафік, який впливає на роботу мережевої інфраструктури відділення банку та отримувати зібрані дані мережевого трафіку у вигляді файлу.

Проведено дослідження програмної системи аналізу мережевого трафіку, що дало змогу впевнитись в унікальності наявних функціональних можливостей застосунку. Крім того, експериментальне дослідження підтвердило роботу програмного застосунку, що, як наслідок, дає можливість використовувати розроблене програмне забезпечення в повному обсязі.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Аналізатор трафіку. *ЛНТУ* : веб-сайт. URL: https://elib.lntu.edu.ua/sites/default/files/elib_upload/12/samr/sam4.htm (дата звернення: 21.10.2023).

2. Аналізатор трафіку. *NiNa.Az* : веб-сайт. URL: https://www.wikidata.uk-ua.nina.az/Аналізатор_трафіку.html (дата звернення: 21.10.2023).

3. Аналізатор трафіку. *SnooPeR* : веб-сайт. URL: https://snooper.in.ua/administruvannia-pk/os-windows/71-2010-06-14-10-55-16.html (дата звернення: 21.10.2023).

4. Аналізатор трафіку. *Вікіпедія* : веб-сайт. URL: https://uk.wikipedia.org/wiki/Аналізатор_трафіку (дата звернення: 21.10.2023).

5. WIRESHARK (ETHEREAL) - АНАЛІЗАТОР МЕРЕЖЕВОГО ТРАФІКУ. *UALinux* : веб-сайт. URL: https://ualinux.com/uk/ubuntu-apps-internet/wireshark (дата звернення: 21.10.2023).

6. Wireshark Wiki. *Wireshark* : веб-сайт. URL: https://wiki.wireshark.org/Home (дата звернення: 21.10.2023).

7. Wireshark. *Вікіпедія* : веб-сайт. URL: https://uk.wikipedia.org/wiki/Wireshark (дата звернення: 21.10.2023).

8. Wireshark – подробное руководство по началу использования. *Habr* : веб-сайт. URL: https://habr.com/en/articles/735866/ (дата звернення: 21.10.2023).

9. Підручник Wireshark: аналізатор мережі та паролів. *HackYourMom* : веб-сайт. URL: https://hackyourmom.com/kibervijna/pidruchnyk-wireshark-analizator-merezhi-ta-paroliv/ (дата звернення: 21.10.2023).

10. Wireshark — основні можливості та як ним користуватись. *DOU.ua* : веб-сайт. URL: https://dou.ua/forums/topic/44274/ (дата звернення: 21.10.2023).

11. Wireshark, програма для збору та аналізу пакетів у мережі. *ubunlog* : веб-сайт. URL: https://ubunlog.com/uk/wireshark-una-aplicacion-para-la-captura-y-analisis-de-paquetes-en-la-red/ (дата звернення: 21.10.2023).

12. Wireshark. *WINSOFT.COM.UA* : веб-сайт. URL: https://winsoft.com.ua/windows/internet/kontrol-trafiku/wireshark (дата звернення: 21.10.2023).

13. CommTraffic. *softonic* : веб-сайт. URL: https://commtraffic.en.softonic.com (дата звернення: 22.10.2023).

14. CommTraffic. *Instalki.pl* : веб-сайт. URL: https://www.instalki.pl/download/programy/windows/narzedzia/sieciowe/commtraffic/ (дата звернення: 22.10.2023).

15. CommTraffic 3.1. *Informer Technologies, Inc.* : веб-сайт. URL: https://commtraffic.informer.com (дата звернення: 22.10.2023).

16. CommTraffic 3.1. *UpdateStar* : веб-сайт. URL: https://commtraffic.updatestar.com (дата звернення: 22.10.2023).

17. Iris Network Traffic Analyzer. *mydiv* : веб-сайт. URL: https://soft.mydiv.net/win/download-iris-network-traffic-analyzer.html (дата звернення: 22.10.2023).

18. Iris Network Traffic Analyzer. *softonic* : веб-сайт. URL: https://iris-network-traffic-analyzer.en.softonic.com (дата звернення: 23.10.2023).

19. Iris Network Traffic Analyzer. *Informer Technologies, Inc.* : веб-сайт. URL: https://iris-network-traffic-analyzer.software.informer.com (дата звернення: 23.10.2023).

20. Iris Network Traffic Analyzer. *Malavida* : веб-сайт. URL: https://iris-network-traffic-analyzer.en.malavida.com/windows/ (дата звернення: 23.10.2023).

21. Iris™ The Network Traffic Analyzer. *StudFiles* : веб-сайт. URL: https://studfile.net/preview/2817614/ (дата звернення: 23.10.2023).

22. Ethernet Internet traffic Statistic 1.02.37. *Download.BG* : веб-сайт. URL: https://www.download.bg/index.php?cls=program&id=385868 (дата звернення: 23.10.2023).

23. Internet traffic. *Wikipedia* : веб-сайт. URL: https://en.wikipedia.org/wiki/Internet_traffic (дата звернення: 24.10.2023).

24. SolarWinds NetFlow Traffic Analyzer *I.T.PRO* : веб-сайт. URL: https://itpro.ua/product/solarwinds-netflow-traffic-analyzer-4/?tab=description (дата звернення: 24.10.2023).

25. Техопедія пояснює Tcpdump *uk.theastrologypage.com* : веб-сайт. URL: https://uk.theastrologypage.com/tcpdump (дата звернення: 24.10.2023).

26. Kismet (програма) *NiNa.Az* : веб-сайт. URL: https://www.wikidata.uk-ua.nina.az/Kismet_(програма).html (дата звернення: 24.10.2023).

27. EtherApe *SourceForge* : веб-сайт. URL: https://etherape.sourceforge.io (дата звернення: 24.10.2023).

28. NetworkMiner - Захоплення файлів у мережевому трафіку та пакетах даних Sniff *genuinelamps.com* : веб-сайт. URL: https://genuine-lamps.com/uk/windows/12255-networkminer-8211-capture-files-on-network-traffic-and-sniff-data-packets.html (дата звернення: 24.10.2023).

29. NetworkTrafficView *NirSoft* : веб-сайт. URL: https://www.nirsoft.net/utils/network_traffic_view.html (дата звернення: 24.10.2023).

30. Імпорт і експорт даних. *NiNa.Az* : веб-сайт. URL: https://www.wikidata.uk-ua.nina.az/Імпорт_і_експорт_даних.html (дата звернення: 25.10.2023).

31. Імпорт і експорт даних. *Вікіпедія* : веб-сайт. URL: https://uk.wikipedia.org/wiki/Імпорт_і_експорт_даних (дата звернення: 25.10.2023).

32. About Data Import. *Google Support* : веб-сайт. URL: https://support.google.com/analytics/answer/3191589 (дата звернення: 25.10.2023).

33. What is data export? *G2.com, Inc.* : веб-сайт. URL: https://www.g2.com/glossary/data-export-definition (дата звернення: 25.10.2023).

34. Data import and export. *IBM* : веб-сайт. URL: https://www.ibm.com/docs/en/sig-and-i/10.0.2?topic=configuring-data-import-export (дата звернення: 25.10.2023).

35. Основні функції моніторингу мережі. *Businessyield.com* : веб-сайт. URL: https://businessyield.com/uk/technology/network-monitoring/ (дата звернення: 26.10.2023).

36. What Is Network Monitoring? *Cisco Systems, Inc.* : веб-сайт. URL: https://www.cisco.com/c/en/us/solutions/automation/what-is-network-monitoring.html (дата звернення: 26.10.2023).

37. What is Network Monitoring? *ManageEngine* : веб-сайт. URL: https://www.manageengine.com/network-monitoring/basics-of-network-monitoring.html (дата звернення: 26.10.2023).

38. What is Network Monitoring? *TechTarget* : веб-сайт. URL: https://www.techtarget.com/searchnetworking/definition/network-monitoring (дата звернення: 26.10.2023).

39. What is Network Monitoring? *VMware* : веб-сайт. URL: https://www.vmware.com/topics/glossary/content/network-monitoring.html (дата звернення: 26.10.2023).

40. Методичні рекомендації до виконання та захисту кваліфікаційної роботи здобувачами другого (магістерського) рівня вищої освіти спеціальності 125 «Кібербезпека» / Молодецька К. В., Євсеєв С. П., Тимонін Ю. О., Веретюк С. М. Житомир : ПНУ, 2023. 42с.