МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ
ФАКУЛЬТЕТ АЕРОНАВІГАЦІЇ,
ЕЛЕКТРОНІКИ ТА ТЕЛЕКОМУНІКАЦІЙ
КАФЕДРА ТЕЛЕКОМУНІКАЦІЙНИХ ТА РАДІОЕЛЕКТРОННИХ СИСТЕМ

ДОПУСТИТИ ДО ЗАХИСТУ
Завідувач кафедри

_____ Роман ОДАРЧЕНКО
“_____”_____2023 р.

КВАЛІФІКАЦІЙНА РОБОТА

(ПОЯСНЮВАЛЬНА ЗАПИСКА)

ВИПУСКНИКА ОСВІТНЬОГО СТУПЕНЯ БАКАЛАВР

Тема: «Інформаційна система моніторингу та аналізу інтернет трафіку на базі протоколу Netflow»

Виконавець: _____ Станіслав ТАТАРИНЦЕВ
(підпис)

Керівник: _____ Володимир ЧУПРИН
(підпис)

Нормоконтролер: _____ Денис БАХТІЯРОВ
(підпис)

Київ 2023

НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ

Факультет аеронавігації, електроніки та телекомунікацій
Кафедра телекомунікаційних та радіоелектронних систем
Спеціальність 172 «Телекомунікації та радіотехніка»
Освітньо-професійна програма «Телекомунікаційні системи та мережі»

ЗАТВЕРДЖУЮ
Завідувач кафедри

Роман ОДАРЧЕНКО
“_____”_____2023 р.

ЗАВДАННЯ
на виконання кваліфікаційної роботи

Татаринцева Станіслава Валентиновича
(прізвище, ім'я, по батькові випускника в родовому відмінку)

1. Тема кваліфікаційної роботи: «Інформаційна система моніторингу та аналізу інтернет трафіку на базі протоколу Netflow»
затверджена наказом ректора від «29» березня 2023 р. № 421/ст
2. Термін виконання роботи: з 22.05.2023 р. по 25.06.2023 р.
3. Вихідні дані до роботи: комп'ютер або сервер для встановлення програмного забезпечення; програмне забезпечення для збору та аналізу даних Netflow; мережне обладнання, що підтримує протокол Netflow; критерії та параметри для оцінки ефективності мережі, що будуть використовуватися для аналізу даних; методології та підходи до аналізу даних про трафік, наприклад, статистичний аналіз, моделювання.
4. Зміст пояснювальної записки: аналіз протоколів обліку мережевого трафіку; інформаційна система моніторингу та аналізу інтернет трафіку на базі протоколу NetFlow; розробка програмного забезпечення для збору та сортування інформації про мережу.
5. Перелік обов'язкового графічного (ілюстративного) матеріалу: слайди презентації в програмному пакеті Microsoft PowerPoint, код розробленого програмного забезпечення в середовищі Microsoft Visual Studio 2022.

6. Календарний план-графік

№ пор.	Завдання	Термін виконання	Відмітка про виконання
1	Розробити деталізований зміст розділів кваліфікаційної роботи	22.05.2023-24.05.2023	Виконано
2	Вступ	25.05.2023	Виконано
3	Аналіз протоколів обліку мережевого трафіку	26.05.2023-29.05.2023	Виконано
4	Інформаційна система моніторингу та аналізу інтернет трафіку на базі протоколу NetFlow	30.05.2023-07.06.2023	Виконано
5	Розробка програмного забезпечення для збору та сортування інформації про мережу	08.06.2023-14.06.2023	Виконано
6	Усунення недоліків та захист кваліфікаційної роботи	15.06.2023-25.06.2023	Виконано

7. Дата видачі завдання: “19" травня 2023 р.

Керівник кваліфікаційної роботи _____ Володимир ЧУПРИН
(підпис керівника) (П.І.Б.)

Завдання прийняв до виконання _____ Станіслав ТАТАРИНЦЕВ
(підпис випускника) (П.І.Б.)

РЕФЕРАТ

Кваліфікаційна робота «Інформаційна система моніторингу та аналізу інтернет трафіку на базі протоколу Netflow» містить 60 сторінок, 23 рисунки, 0 таблиць, 11 використаних джерел.

ІНФОРМАЦІЙНА СИСТЕМА, МОНІТОРИНГ, АНАЛІЗ ІНТЕРНЕТ ТРАФІКУ, ПРОТОКОЛ NETFLOW, ЕФЕКТИВНІСТЬ МЕРЕЖІ, КІЛЬКІСНІ ДОСЛІДЖЕННЯ, МОДЕЛЮВАННЯ ПОВЕДІНКИ МЕРЕЖІ, АНОМАЛЬНА ПОВЕДІНКА В МЕРЕЖІ, АВТОМАТИЗОВАНА ОБРОБКА ДАНИХ, ОПТИМІЗАЦІЯ МЕРЕЖІ.

Мета кваліфікаційної роботи – розробка інформаційної системи моніторингу та аналізу інтернет трафіку на основі протоколу NetFlow.

Об'єктом кваліфікаційної роботи є інформаційна система моніторингу та аналізу інтернет-трафіку на базі протоколу Netflow. Ця система є складним технічним об'єктом, який складається з компонентів апаратного та програмного забезпечення, мережевих пристроїв, баз даних та інших компонентів.

Предметом кваліфікаційної роботи є сам процес розробки, налаштування та впровадження інформаційної системи моніторингу та аналізу інтернет-трафіку на базі протоколу Netflow. В рамках роботи можуть досліджуватимуться такі аспекти, як аналіз вимог до системи, вибір компонентів системи, розробка та налаштування програмного забезпечення, налаштування мережевих пристроїв та збір інформації про трафік, а також аналіз та візуалізація зібраної інформації.

Методи досліджень, що були використані в роботі включають:

- Аналіз літературних джерел та вивчення наукових робіт на тему моніторингу та аналізу інтернет-трафіку, а також вивчення документації та рекомендацій по використанню протоколу Netflow.

- Експериментальні дослідження, що проведені на основі тестування різних програмних засобів та дозволяють збирати і аналізувати дані про трафік, на різних обладнаннях та в різних мережевих умовах.
- Створення прототипу системи моніторингу та аналізу інтернет-трафіку, що базується на протоколі Netflow. Цей прототип може бути використаний для оцінки функціональності та ефективності системи.
- Аналіз даних, зібраних від користувачів, що користуються мережею, та створення відповідних моделей для покращення якості обслуговування мережі.
- Статистичний аналіз даних про трафік для визначення найбільш активних пристроїв та додатків, що споживають більшість ресурсів мережі.
- Вивчення різних аспектів експлуатації системи моніторингу та аналізу інтернет-трафіку, зокрема, проблеми сумісності з різними мережевими пристроями, шляхи вирішення проблем зі збором та аналізом даних, а також оцінка можливостей по використанню отриманих результатів для підвищення ефективності мережі.

Загалом, для досягнення мети роботи були використані різноманітні методи досліджень, включаючи аналітичний огляд літератури, експериментальні дослідження, створення прототипів, аналіз даних, моделювання та використання інтелектуальних алгоритмів.

Практичне значення отриманих результатів. При написанні кваліфікаційної роботи протестовано рішення для аналізу трафіку, яке зберігає інформацію про пакети в базі даних MySQL та за допомогою C# вивантажує дані у найбільш читальному вигляді, що може бути використане реальними мережевими адміністраторами для аналізу мережевого трафіку.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ ……..…….. 6
ВСТУП ……...…….…….... 8
РОЗДІЛ 1. АНАЛІЗ ПРОТОКОЛІВ ОБЛІКУ МЕРЕЖЕВОГО ТРАФІКУ …….. 12
1.1. Область застосування системи …….…… 12
1.2. Види протоколу NetFlow …….….. 26
1.3. Структура роботи протоколу ……..…….. 31
РОЗДІЛ 2. ІНФОРМАЦІЙНА СИСТЕМА МОНІТОРИНГУ ТА АНАЛІЗУ ІНТЕРНЕТ ТРАФІКУ НА БАЗІ ПРОТОКОЛУ NETFLOW ……...…….. 33
2.1. Огляд мережі та її окремих компонентів ……..... 33
2.2. Налаштування мережного обладнання ……..….. 38
2.3. Конфігурування сервера ……... 39
РОЗДІЛ 3. РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ ЗБОРУ ТА СОРТУВАННЯ ІНФОРМАЦІЇ ПРО МЕРЕЖУ …….. 44
3.1. Робота у середовищі Visual Studio ……...…. 44
3.2. Інтерфейс програми ……...…… 51
ВИСНОВКИ ……..……..... 57
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ ……..……...… 59

ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАЧЕНЬ

ARPAnet (англ. Advanced Research Projects Agency Network) – мережа агентства передових досліджень у США.

CLNS (англ. Connection Less Network Protocol) – мережевий протокол передачі даних без встановлення з'єдання.

DDP (англ. Distributed Data Protocol) – клієнт-серверний протокол для запитів та оновлення серверної бази даних, а також синхронізації тих оновлень поміж усіма клієнтами.

ICMP (англ. Internet Control Message Protocol) – міжмережевий протокол керуючих повідомлень.

IETF (англ. Internet Engineering Task Force) відкрита міжнародна організація проєктувальників, вчених, мережевих операторів та провайдерів.

IPX (англ. Internetwork Packet Exchange) – протокол мережевого рівня моделі OSI, призначений для передачі дейтаграм у системах, неорієнтованих на з'єднання.

MIB (англ. Management Information Base) – віртуальна база даних, яка використовується для управління об'єктами в мережі зв'язку.

MySQL – вільна реляційна система керування базами даних.

OSI (англ. Open Systems Interconnection Basic Reference Model) - абстрактна мережева модель для комунікацій і розробки мережевих протоколів.

PDU (англ. Protocol Data Unit) – одиниця інформації яка транспортована через всю комп'ютерну мережу.

QoS (англ. Quality of Service) – набір методів для керування ресурсами мережі.

SMTP (англ. Simple Mail Transfer Protocol) – простий протокол передачі, призначений для передачі електронної пошти в мережі.

SNMP (англ. Simple Network Management Protocol) – протокол керування мережами зв'язку на основі архітектури TCP/IP.

SSL (англ. Secure Sockets Layer) – криптографічний протокол, який має хороший рівень безпеки.

TCP/IP (англ. Transmission Control Protocol, англ. Internet Protocol) – набір або стек протоколів мережі Інтернет.

UDP (англ. User Datagram Protocol) – протокол датаграм користувачів, один з протоколів стеку TCP/IP.

UNIX – сімейство багатозадачних і багатокористувацьких операційних систем, які розроблені в 1970-х роках.

VoIP (англ. Voice over IP) – технологія передачі медіа-даних у реальному часі за допомогою сімейства протоколів TCP/IP.

WMI (англ. Windows Management Instrumentation) – базова технологія для централізованого керування і стеження за роботою інфраструктури на Windows.

WREN (англ. Watching Resources from the Edge of the Network) – перегляд ресурсів на кінцях мережі.

ВСТУП

У минулому, коли мережі переважно будувалися на шинній топології, моніторинг трафіку був простим завданням. Це пояснювалося тим, що середовище передачі мереж було простим, що спрощувало спостереження за ними в рази. На той час, моніторинг трафіку міг бути здійснений одним пристроєм, який підключався до мережі. Проте, з часом зросла потреба у високій пропускній здатності мережі, а також у розвитку технології комутації пакетів. Ці фактори призвели до зміни середовища передачі на високосегментовані топології. Тепер загальний трафік не можна прослуховувати з однієї точки, тому для повної картини потрібно моніторити кожний порт. Через велику кількість з'єднань типу "точка-точка", встановлення приладів для моніторингу є незручним, і велика кількість таких приладів ускладнює задачу моніторингу. Крім того, самі комутатори та маршрутизатори мають складну архітектуру, і швидкість обробки та передачі пакетів є важливим фактором, що визначає продуктивність мережі. Отже, єдиним реальним рішенням для моніторингу трафіку є здійснення його безпосередньо всередині активного мережного обладнання, зокрема маршрутизаторів.

Актуальність теми. Протокол NetFlow є досить актуальним для моніторингу мережі в сучасному середовищі, де високосегментовані топології та велика кількість точка-точка з'єднань ускладнюють завдання моніторингу трафіку. Використання NetFlow дозволяє отримати повну картину розподілу трафіку в мережі та забезпечити аналіз його характеристик, таких як джерело трафіку, цільовий адрес, протокол, порти, використані ресурси мережі та інші параметри.

Завдяки протоколу NetFlow можна виявляти та усувати проблеми з мережевим трафіком, такі як затори, збої в роботі мережі, атаки та інші проблеми, що можуть негативно впливати на продуктивність мережі та безпеку інформації. Крім того, аналіз даних NetFlow може допомогти забезпечити оптимальну роботу мережі, налаштувати правила мережевої безпеки та вирішувати інші проблеми, що стосуються мережі.

Протокол NetFlow є досить актуальним в контексті моніторингу трафіку в мережах. З його допомогою можна збирати детальну інформацію про трафік, яка дозволяє аналізувати його характеристики та використовувати для вирішення різноманітних завдань, таких як:

- ***Відлагодження мережевих проблем:*** збираючи докладну інформацію про трафік, NetFlow дозволяє виявляти проблеми в мережі, такі як затори, низьку пропускну здатність, підозрілий трафік, і т.д.
- ***Планування ємності мережі:*** інформація, зібрана з NetFlow, допомагає зрозуміти, як використовується пропускна здатність мережі в реальному часі та прогнозувати, які ємності необхідні для підтримки потреб мережі в майбутньому.
- ***Забезпечення безпеки мережі:*** NetFlow допомагає виявляти підозрілий трафік, що може свідчити про атаки або злочинну діяльність, та забезпечує збір детальної інформації про трафік для подальшого аналізу та виявлення загроз.
- ***Аналіз трафіку:*** збір детальної інформації про трафік дозволяє проводити його докладний аналіз, включаючи визначення часу відповіді, використання ресурсів та інші характеристики, що можуть бути важливими для підтримки продуктивності мережі.

Отже, використання протоколу NetFlow є актуальним в сучасних мережевих середовищах, де важливо мати докладну інформацію про трафік для вирішення різних завдань, пов'язаних з моніторингом [1-11].

Мета і завдання дослідження.

Мета кваліфікаційної роботи – розробка інформаційної системи моніторингу та аналізу інтернет трафіку на основі протоколу NetFlow.

Для досягнення поставленої мети вирішуються такі наукові завдання.

- Вивчення основ протоколу NetFlow.
- Принцип роботи та функціонування протоколу NetFlow.
- Класифікація протоколу NetFlow.
- Реалізація моніторингу за допомогою протоколу NetFlow.
- Аналіз трафіку за допомогою програми.

Об'єктом кваліфікаційної роботи є інформаційна система моніторингу та аналізу інтернет-трафіку на базі протоколу Netflow. Ця система є складним технічним об'єктом, який складається з компонентів апаратного та програмного забезпечення, мережевих пристроїв, баз даних та інших компонентів.

Предметом кваліфікаційної роботи є сам процес розробки, налаштування та впровадження інформаційної системи моніторингу та аналізу інтернет-трафіку на базі протоколу Netflow. В рамках роботи можуть досліджуватимуться такі аспекти, як аналіз вимог до системи, вибір компонентів системи, розробка та налаштування програмного забезпечення, налаштування мережевих пристроїв та збір інформації про трафік, а також аналіз та візуалізація зібраної інформації.

Методи досліджень, що були використані в роботі включають:

- Аналіз літературних джерел та вивчення наукових робіт на тему моніторингу та аналізу інтернет-трафіку, а також вивчення документації та рекомендацій по використанню протоколу Netflow.
- Експериментальні дослідження, що проведені на основі тестування різних програмних засобів та дозволяють збирати і аналізувати дані про трафік, на різних обладнаннях та в різних мережевих умовах.
- Створення прототипу системи моніторингу та аналізу інтернет-трафіку, що базується на протоколі Netflow. Цей прототип може бути використаний для оцінки функціональності та ефективності системи.
- Аналіз даних, зібраних від користувачів, що користуються мережею, та створення відповідних моделей для покращення якості обслуговування мережі.
- Статистичний аналіз даних про трафік для визначення найбільш активних пристроїв та додатків, що споживають більшість ресурсів мережі.
- Вивчення різних аспектів експлуатації системи моніторингу та аналізу інтернет-трафіку, зокрема, проблеми сумісності з різними мережевими пристроями, шляхи вирішення проблем зі збором та аналізом даних, а також оцінка можливостей по використанню отриманих результатів для підвищення ефективності мережі.

Загалом, для досягнення мети роботи були використані різноманітні методи досліджень, включаючи аналітичний огляд літератури, експериментальні дослідження, створення прототипів, аналіз даних, моделювання та використання інтелектуальних алгоритмів.

Практичне значення отриманих результатів.

При написанні кваліфікаційної роботи протестовано рішення для аналізу трафіку, яке зберігає інформацію про пакети в базі даних MySQL та за допомогою C# вивантажує дані у найбільш читальному вигляді, що може бути використане реальними мережевими адміністраторами для аналізу мережевого трафіку.

Апробація отриманих результатів. Основні положення роботи доповідалися та обговорювалися на таких конференціях:

- Науково-практична конференція «Проблеми експлуатації та захисту інформаційно-комунікаційних систем», м. Київ, 2023 р.

РОЗДІЛ 1
АНАЛІЗ ПРОТОКОЛІВ ОБЛІКУ МЕРЕЖЕВОГО ТРАФІКУ

1.1. Область застосування системи

Моніторинг і управління комп'ютерною мережею є важливою і обов'язковою задачею як для мережевого інженера, так і для системного адміністратора, оскільки на даний момент 9 з 10 компаній мають свою локальну установку. При першому перегляді рядового співробітника може показатися, що в роботі мережевого інженера немає нічого складного: адміністратор мережі отримує повідомлення від користувачів мережі про несправності та проводить роботу за їх усуненням. Такі роботи прийнято називати реактивною підтримкою [1].

Реактивний підхід недостатній для обслуговування крупних мереж, оскільки кількість повідомлень про несправності може збільшитися лавиноподібно, а це може призвести до відмови критичних мережевих сервісів. Корпоративна мережа в даний час є бізнес-критичним інструментом. Любий, навіть 5-ти хвилинний простій мережі може негативно вплинути на доходи компанії, тому мережевим інженерам варто уникнути подібних випадків [1].

Для запобігання непередбаченої відмови мережі інженер повинен застосовувати комплексний і структурований підхід до задачі управління мережею, що включає, в тому числі, виконання проактивних дій.

Так, у загальному випадку, завдання управління комп'ютерною мережею може бути розділене на два компоненти [1-3]:

- структуровані роботи – заплановані роботи з підтримки мережі;
- реактивні роботи – роботи з усунення виявлених неполадок.

Безумовно, в реальній мережевій інфраструктурі звести до нуля реактивні роботи не є можливим, однак структурований підхід при управлінні мережею дозволяє мінімізувати такі роботи. Крім того, структурований підхід дозволяє більш ефективно виявити і усунути вразливість в комп'ютерних мережах.

Існують визначені моделі (ITIL, FCAPS, TMN, Cisco Lifecycle Services), описуючі завдання підтримки IT-інфраструктури в усій або сегментованій мережевій інфраструктурі зокрема. Використання таких моделей як Основа при розробці власної структурованої схеми управління мережею не є мовитоном. Світовий досвід показує, що процедури, описані в стандартизованих моделях, у багатьох випадках вибиті [2].

Можна виділити основні пункти, які потрібні в рамках рішення задачі управління мережею, і яку роль в управлінні мережею грає кожен із запропонованих пунктів.

Забезпечення доступу до мережних пристроїв для керування ними. Безумовно, мережевий інженер повинен мати можливість будь-якої миті часу підключитися до мережного пристрою, щоб внести зміни в налаштування або подивитися телеметричну інформацію. Основний інструмент управління мережевим пристроєм - командний рядок (CLI), проте практично всі сучасні мережеві пристрої надають зручний графічний інтерфейс (GUI).

Існує широкий спектр програмного забезпечення, за допомогою якого можна одночасно керувати відразу кількома пристроями та/або технологіями.

Слід зазначити, що доступ до мережевих пристроїв необхідний не тільки для мережевого інженера, але і для системи моніторингу. Крім того, при вирішенні завдання забезпечення доступу до мережних пристроїв дуже важливо дотримуватися належного рівня безпеки.

Управління мережею у великих територіально – розподілених мережевих інфраструктурах може бути організовано як у тих каналах, якими передається користувальницький трафік, і спеціально виділених каналів, що певною мірою зменшує ризик сильного навантаження на основний канал передачі.

Моніторинг мережевої інфраструктури можна розбити на такі компоненти [1-4]:

- моніторинг мережевих пристроїв;
- моніторинг каналів передачі даних.

У свою чергу, завдання моніторингу мережевих пристроїв може бути розбите на такі підпункти [3]:

- збирання системних повідомлень – логів пристрою;
- моніторинг доступності та телеметрії мережного пристрою;
- оповіщення інженера про зміни в мережі.

Постійний моніторинг мережної інфраструктури дозволяє мережному інженеру отримувати всю необхідну інформацію про мережу в режимі реального часу Крім того, моніторинг допомагає визначити рівень (*baseline*) для параметрів мережевих пристроїв та каналів передачі даних. Для різних мережних інфраструктур межа допустимих значень параметрів може змінюватись. Так, наприклад, для однієї мережевої інфраструктури завантаження процесора маршрутизатора на периметрі мережі у нормальному режимі роботи не перевищує 10%. В іншій мережі, аналогічний маршрутизатор працює постійно та стабільно з навантаженням 30 – 40%. При цьому, хоча завантаженість маршрутизатора у другому випадку відносно висока, всі сервіси та бізнес-критичні програми працюють нормально. У такому разі мережевий інженер може прийняти запропоновані значення за показник нормального режиму роботи мережного пристрою. При перевищенні даних значень, мережевий інженер бачить, що у мережі відбувається аномалія, можливі виникнення проблем у роботі сервісів та додатків, отже, потрібне оперативне втручання: пошук причини перевищення допустимих показників та усунення джерела проблеми [3-4].

Світовий досвід інженерів показує, що моніторинг у рамках описаних вище завдань є необхідним та достатнім для підтримки роботи мережної інфраструктури на належному рівні. При акуратному дотриманні викладених вище пунктів мережевий інженер зможе впоратися практично з будь-якою мережевою проблемою в стислі терміни. Багато несправностей мережі можуть бути усунені проактивно. Якщо врахо-вані не всі запропоновані пункти, мережевий інженер в якийсь момент часу обов'язково зіткнеться з ситуацією нестачі інформації для вирішення чергової проблеми, а це в рази збільшує її час вирішення, оскільки проходиться в екстреному порядку додавати засоби моніторингу, що бракують. А в деяких випадках, навіть після маніпу-

ляцій є ризик, що мережевий інженер після встановлення нового інструменту моніторингу не завжди зможе коректно інтерпретувати нові дані, адже у нього немає інформації про те, як мережа працювала раніше, які показники вважати нормальними, а які аномальними [1-6].

Заміна застарілого або обладнання, що вийшло з ладу. З часом надійність мережевих пристроїв падає. Застаріле обладнання схильне до нових видів атак, тому представляє вразливість для мережної інфраструктури в цілому. Крім того, застаріле обладнання з певного моменту часу перестає відповідати постійно зростаючим вимогам щодо функціональності продуктивності. Таким чином, мережевий інженер повинен мати вичерпну інформацію про моделі використовуваних мережевих пристроїв та проводити заміну обладнання на більш сучасні пристрої за потреби.

Оновлення програмного забезпечення мережевих пристроїв. Виробники мережного обладнання постійно розробляють нові операційні системи та нові версії систем для пристроїв. Як правило, вихід нової версії програмного забезпечення обумовлений виправленням уразливостей операційної системи, виправленням багів у коді, збільшенням продуктивності, а також додаванням нового функціоналу.

Резервне копіювання конфігурацій мережевих пристроїв. Конфігурація пристрою має таку неприємну особливість, як скидання при раптовій відмові пристрою. Мережевий інженер повинен замінити пристрій, що вийшов з ладу, аналогічним, що має такі ж налаштування. Безумовно, завдання заміни пристрою, що відмовив, значно спрощується і прискорюється, якщо мережевий інженер має резервну копію конфігурації. Всі сучасні пристрої мають можливість швидко «злити» файл із конфігурацією на будь-який локальний сервер. Конфігураційний файл може знадобитися так і при відмові будь-якої функції пристрою після переналаштування обладнання. Така ситуація може статися як з вини мережного інженера (людський фактор), так і з вини виробника програмного забезпечення (проявляється баг). Тим не менш, працездатність сервісу має бути відновлена у стислий термін. За наявності резервної копії робочої конфігурації мережевий інженер може швидко провести процедуру відкату [5].

Підтримка документації мережної інфраструктури. Як правило, первісна документація мережевої інфраструктури створюється на етапах проєктування та впровадження. Однак у процесі експлуатації вносяться численні зміни у налаштування мережевих пристроїв, топологію мережі і т.д. Маючи під рукою такий документ багато в чому спрощує життя мережевих інженерів і практично мінімізує людський фактор, коли дані та інформація про мережу може загубитися через те, що людина, яка її знала, забула її.

При організації керування мережними пристроями необхідно вибрати тип побудови схеми керування, а також забезпечити правильне та безпечне налаштування самих пристроїв для забезпечення підключення до них з метою налаштування.

На даний момент можна виділити кілька типів керування мережевим обладнанням. До найчастіше описуваних методів відносяться управління всередині смуги передачі трафіку (in-band) і поза смуги передачі трафіку (out-of-band). Крім класичного підходу, існує кілька альтернативних варіантів управління мережевим обладнанням [2-4].

Перший тип управління in-band передбачає передачу трафіку управління обладнанням (Telnet, SSH, HTTPs тощо) та трафіку моніторингу (Syslog, SNMP, Netflow та ін.) за тими ж фізичними каналами та портами на мережному обладнанні, де передається і звичайний користувальницький трафік тобто, та сама мережа забезпечує передачу всіх даних. Безумовно при налаштування обладнання необхідно на логічному рівні сегментувати трафік управління та інший трафік. Це можна зробити, використовуючи віртуальні мережі (VLAN), списки доступу (ACL), міжмережні екрани та інше. Але, як було зазначено раніше, «фізика» залишається однією. Основний полюс такого рішення – простота. Але за простоту, як це зазвичай буває, доводиться платити. Головний мінус цього варіанта полягає в тому, що якщо мережа повністю паралізована, наприклад, паразитним трафіком, доступ до пристроїв теж губиться і локалізувати проблему віддалено стає практично неможливо. Це ускладнить діагностику та виправлення ситуації у роботі мережі. Щоб цього не сталося, необхідно на мережевому обладнанні налаштувати різні рівні якості обслуговування трафіку (QoS). Для

трафіку управління та моніторингу потрібно виділити мінімально необхідну смугу пропускання з пріоритетом, що дозволяє його передавати, навіть якщо мережа перевантажена, попередньо її промаркувавши. Однак такий підхід не дає 100% гарантії та ускладнює налаштування обладнання. Також є ймовірність, що пристрій за якоюсь причини автоматично заблокує порт, і віддалений доступ до нього пропаде (наприклад, комутатор Cisco може перевести порт у стан errdisabled). Також можна просто помилково заблокувати порт, через який йде трафік управління, тим самим втративши доступ на сам пристрій [4-6].

Другий тип управління out-of-band передбачає передачу трафіку управління з окремих фізичних каналів зв'язку тобто, будується друга мережа, яка обслуговує передачу трафіку керування. На кожному мережевому рівні Ваш пристрій виділяє окремий порт для підключення до мережі. Зазвичай використовується виділений комутатор, до якого підключається вся інфраструктура управління (машина адміністратора, Syslog-сервер, Netflow-колектор та інше). За такої організації, адміністратор мережі практично завжди матиме доступ на мережеве обладнання, навіть якщо основна мережа повністю відмовила. Безперечно основними мінусами є необхідність використання окремого обладнання, а також додаткові настройки. Ще варто звернути увагу, що при організації out-of-band управління завжди потрібні окремі канали зв'язку. Коли обладнання стоїть в одній серверній, це не є великою проблемою. Завжди можна знайти додаткові патч-корди. Але якщо обладнання рознесене по будівлі або знаходиться на територіально розподілених майданчиках, питання з виділеними каналами ставати критичним. Як це часто буває, додаткові мідні або оптичні траси прокласти досить проблематично, якщо вони не були спочатку передбачені. Також варто звернути увагу на необхідність наявності на мережному обладнанні додаткових портів для підключення до мережі управління. Дуже часто в маршрутизаторах присутні лише два фізичних інтерфейсів, які зазвичай вже зайняті [4-6].

Якщо у нас територіально розподілена мережа, організувати управління out-of-band у віддаленому офісі, особливо якщо його топологія досить проста, стає складним і не виправданим з точки зору ціни. Тому з'являється гібридний варі-

ант управління мережею: out-of-band для центрального офісу та in-band для віддалених офісів. Точно така ж схема може бути використана, якщо мережа розподілена по будівлі та через відсутність додаткових міжповерхових з'єднань, організувати скрізь outof-band управління стає складно.

Хотілося б відзначити ще один варіант управління мережею, який більшою мірою можна зарахувати до out-of-band – використання консольного сервера. У цьому випадку кожен мережний пристрій підключається до консольного сервера, через який керується обладнанням. Найбільшою перевагою цього варіанта є те, що можливість прямого підключення до пристрою є завжди, навіть коли він не коректно завантажився. Але такий варіант управління не завжди зручний і прийнятний, тому що пристрій повинен знаходитися тільки на невеликій відстані від консольного сервера, наприклад, в одній серверній кімнаті, а ця умова не завжди реалізується на підприємстві. Також слід зазначити, що таке підключення до пристроїв не можна використовувати для моніторингу мережного обладнання. До мінусів також можна віднести швидкість передачі консольного кабелю, через низьку швидкість виведення великої кількості інформації може тривати досить багато часу. Відомі такі випадки, коли виведення інформації проводилося протягом 30 хвилин, а для оперативного вирішення проблеми це занадто багато.

Крім консольного сервера, як альтернативний варіант, можна мати спеціально виділений ноутбук із консольним кабелем тобто мінімальний комплект, що дозволяє підключитися до мережного обладнання по консолі. Такий варіант дозволить за необхідності досить оперативно дійти до пристрою та підключиться до нього. Зазвичай у потрібний момент, немає або ноутбука під рукою або консольного кабелю з перехідником.

Такий тип управління, як управління з хмари є в деякому роді гібридним типом. Одним із прикладів управління мережевим обладнання з хмари є концепція, реалізована в лінійці продуктів Cisco Meraki. Усі пристрої цієї марки (а туди входять і комутатори, та пристрої безпеки, а також рішення щодо побудови безпроводової мережі) автоматично після встановлення підключаються до хмари Cisco. Для управління цією

структурою мережному інженеру необхідно підключитись до хмарного порталу. Головним недоліком цієї схеми є той факт, що якщо зникає зв'язок з хмарою, керувати пристроями не вийде. Це суттєво підвищує вимоги до надійності та кількості Інтернет-каналів [7].

Концепція програмно-конфігурованих мереж є однією з найбільш розвиваються на ринку мережевих технологій. Вона передбачає повне відділення функцій управління пристроями та контролю трафіку від функцій передачі. Тобто за керування всіма мережевими пристроями та логіку контролю за трафіком (наприклад, протоколи маршрутизації, службові протоколи, vlan) відповідає деякий централізований програмний пристрій (контролер), а мережні пристрої займаються лише передачею трафіку. З одного боку, плюси підходу очевидні: це зручне управління всією мережею, з дуже гнучким функціоналом (додаткові функції реалізуються програмно).

В ідеалі налаштування мережного обладнання для забезпечення функцій управління та моніторингу має рекомендувати рекомендація виробника обладнання. Паролі повинні мати достатню міру надійності. Для віддаленого підключення повинні такі протоколи як SSH і HTTPs. Доступ до пристрою має бути обмеженим. Список таких рекомендацій можна продовжувати досить довго. При цьому є ще рекомендації щодо коректної настоянки кожного протоколу управління, рекомендації щодо налаштування якості обслуговування (QoS), рекомендації щодо зняття телеметрії з обладнання тощо. Якщо відкрити хоч одну рекомендацію щодо налаштування перелічених вище параметрів, можна побачити, що вона займає не одну сторінку, але це налаштування важливе.

Короткий перелік рекомендацій, який можна розширити або доповнити, звернувшись до документації на обладнання [4-7]:

- Для віддаленого підключення рекомендується використовувати такі захищені протоколи, як SSH та HTTPs. Вже практично кожному відомо, що в трафіку, який передається через протокол telnet логін і пароль, передаються у відкритому вигляді, що не є добре.

- Ступінь надійності облікових записів, призначених для входу на активні мережні пристрої, має бути вищим за середній. Дефолтні облікові записи краще видалити. Усі паролі повинні зберігатися у конфігурації у захищеному та зашифрованому вигляді.
- Весь трафік моніторингу та управління, що передається по відкритих каналах, повинен бути зашифрований: або за допомогою VPN з'єднання, або з використанням захищених протоколів, таких як SNMPv3, SFTP, SCP.
- Доступ до пристрою повинен бути обмежений лише для певного кола користувачів. Наприклад, на устаткуванні Cisco це можна зробити за допомогою списків доступу (ACL).
- На пристроях має бути налаштована синхронізація часу. Час краще не прописувати вручну, а налаштувати зв'язок пристрою з NTP-сервером. Це дозволить точніше визначати монети, коли здійснювалася спроба легітимного чи легітимного підключення.
- Усі підключення та команди, що вводяться, рекомендується логувати. Особливо це актуально у разі керування пристроєм кількома людьми.
- Кожен користувач повинен підключатися, використовуючи свій унікальний обліковий запис.

Налаштування рівнів доступу на пристрої також є важливою частиною конфігурування. При підключенні повинно виводитись повідомлення про те, що доступ дозволено лише авторизованим користувачам.

Як було зазначено раніше, крім рекомендацій, що стосуються безпосереднього керування пристроями, існують такі ж рекомендації щодо безпечного налаштування різних протоколів (наприклад, EIGRP, OSPF тощо), що забезпечують роботу мережі. Наприклад, рекомендується завжди включати логування змін стану протоколу (додавання/видалення маршрутів тощо), а також автентифікацію між пристроями під час встановлення сусідніх відносин. Також є цілий набір рекомендацій щодо налаштування якості обслуговування (QoS) трафіку управління та моніторингу. Туди ж можна віднести питання фільтрації та обмеження швидкості для того чи іншого протоколу з

метою запобігання DDoS-атаки типу «відмова в обслуговуванні». До моніторингу мережних пристроїв можна віднести збір системних повідомлень, моніторинг доступності та телеметрії мережного пристрою, а також оповіщення інженера про зміни в мережі. Для збору з пристроїв системних повідомлень практично на всіх пристроях є syslog повідомлення. Збір телеметричних даних провадиться з використанням протоколу SNMP. Також можна налаштувати систему повідомлень для інженера, яка надсилатиме йому повідомлення про зміни в мережі. Моніторинг каналів зв'язку організується за допомогою протоколу Netflow або його аналогів. Далі кожен етап буде розглянуто докладніше [7].

Логи візуалізують процес роботи мережного пристрою, відображають його стан. Виконання системою будь-якої дії відбивається відповідним системним повідомленням – логом. Існують різні рівні деталізації системних повідомлень. Як правило, в залежності від рівня деталізації інформації в логах на обладнанні існують різні рівні логування. Для обладнання Cisco Systems представлено 8 рівнів: від рівня 0 (Emergencies – повідомлення про непрацездатність системи) до рівня 7 (Debugging – повідомлення про налагодження).

Устаткування надає такі можливості щодо виведення логів: на консоль пристрою (console logging), локальний буфер пристрою (buffer logging), термінальну лінію (monitor logging) і зовнішній мережевий накопичувач – виділений Syslog-сервер. У ролі останнього може бути централізована система моніторингу.

При включенні логування необхідно бути дуже уважним. Існують деякі нюанси, невиконання яких може призвести до відмови пристрою та/або необхідності його перезапуску. Перший нюанс стосується виведення в консоль або термінальну лінію логів високого рівня деталізації, зокрема – рівень 7. Особливо цей пункт стосується ситуації, у яких інженер активує додаткову трасування будь-якого сервісу командою debug. Мережевий пристрій може генерувати занадто багато повідомлень в одиницю часу, весь процесорний час буде витрачено на виведення даних повідомлень на екран. Пристрій може «зависнути» і перестати виконувати свою головну функцію – маршрутизувати та комутувати мережевий трафік. При необхідності перегляду логів високого рівня деталізації рекомендуємо виводити повідомлення до локального буфера.

Другий аспект стосується виведення лог-повідомлень в локальний буфер. На мережних пристроях Cisco локальний буфер виділяється із загального пулу оперативної пам'яті. Якщо ми запитаємо занадто великий обсяг пам'яті під лог-буффер, пристрій може відчувати нестачу оперативної пам'яті, що в свою чергу може призвести до «зависання» пристрою, незапланованого перезавантаження. Окремо варто виділити налаштування логування на міжмережевих екранах Cisco ASA. Розподіл лог-повідомлень за рівнями для багатьох випадків не є оптимальним. Наприклад, повідомлення, пов'язані з роботою списків доступу (ACL) на пристрої або з правилами трансляції IP-адрес (NAT) можуть мати рівні від 2 до 6 [2-5].

Можна з упевненістю сказати, що і при стабільному та нормальному режимі роботи пристрою не можна виключати виведення лог-повідомлень високого рівня критичності. У зв'язку з цим, деякі розробники операційних систем для мережевих пристроїв ввели розширені можливості щодо оптимізації та налаштування системи логування на пристрої. Інженер має можливість змінити рівень будь-якого лог-повідомлення. Крім того, адміністратор може створювати списки системних повідомлень, об'єднуючи в групи подій. Наприклад, для налагодження та моніторингу роботи сервісу VPN, мережевий інженер може створити список, до якого будуть потрапляти тільки лог-повідомлення, пов'язані з роботою даного сервісу, та надсилати на виділений Syslog-сервер лише налаштований список. Крім того, пристрій Cisco ASA може надсилати списки лог-повідомлень по електронній пошті. Для будь-якого мережного пристрою системні лог- повідомлення є головним, а в деяких випадках і єдино доступним інструментом пошуку проблем та несправностей. При проведенні діагностики мережевої проблеми, інженер після перевірки коректності Конфігурації, першим кроком повинен переглянути логи мережевих пристроїв. Імовірність виявлення причини проблеми системного логу вкрай велика.

Системні повідомлення незамінні під час розслідування мережевих проблем, моменти прояву яких непередбачувані. Крім вирішення мережевих проблем, хотілося б відзначити ще одну сферу застосування системних повідомлень. Лог-повідомлення можна використовувати разом із вбудованим в операційну систему пристроїв Cisco

редактором автоматичних сценаріїв Cisco EEM (Embedded Event Manager). Цей функціонал дозволяє створювати скрипти для автоматичної зміни конфігурацій пристроїв. Лог-повідомлення може виступати тригером запуску скрипта. Протокол SNMP – Network Management Protocol – є стандартом для обміну керуючою інформацією між мережевими пристроями та системою управління мережею (NMS – Network Management System). З точки зору моніторингу мережі протокол SNMP є незамінним засобом збирання телеметричної інформації з мережевих пристроїв. Збір телеметричних показників мережевих пристроїв невід'ємним компонентом керування комп'ютерною мережею [4-6].

Телеметрична інформація дозволяє мережному інженеру шукати «вузькі місця» у мережевій топології, запобігати можливим відмовам, відстежувати причини мережевих проблем, визначати робочі рівні для показників мережевих пристроїв, виявляти аномалії у роботі мережі. До найважливіших датчиків телеметрії пристроїв належать такі показники, як завантаження процесора пристрою, завантаження оперативної пам'яті, робота систем живлення, охолодження, температура пристрою.

Перераховані показники телеметрії рекомендується відстежувати будь-якому мережевому пристрої. Крім того, на пристроях, залежно від покладеного функціоналу, рекомендується включати моніторинг додаткові параметри. Так, для пристроїв, що термінують VPN-рекомендується опитувати відповідні SNMP OID. Для багатьох мережевих пристроїв важливо знати поточне завантаження мережевих інтерфейсів. Ця інформація допоможе досить точно оцінити завантаження каналів передачі даних у мережній інфраструктурі. Є навіть ряд проблем, що виникають у мережі, пошук причин та усунення яких просто нездійсненні без попереднього збору та аналізу телеметрії пристроїв, зібраної за протоколом SNMP. Наприклад, ситуації, коли зв'язок пропадає не повністю, а частково, або якість зв'язку настільки погіршується, що стає неприйнятним для певних мережевих додатків.

У більшості випадків при виникненні подібних проблем мережевий інженер не може отримати практично жодної інформації за допомогою перевірки конфігурацій мережевих пристроїв або переглядом лог-повідомлень. У внаслідок таких

первинних перевірок складається враження, що вся мережева інфраструктура працює коректно та безвідмовно. Але при цьому постійно надходять скарги від кінцевих користувачів про те, що «відео не вантажиться», «Телефонія працює погано і якість голосу неприйнятна». Найбільш імовірною причиною виникнення подібних проблем є зростання навантаження на мережні пристрої та/або на канали передачі даних. За допомогою NMS, що збирає телеметрію мережевих пристроїв SNMP легко оцінити динаміку зміни навантаження на мережеві пристрої. Цілком ймовірно, що на шляху проходження проблемного потоку даних, знаходяться одне або кілька мережних пристроїв, завантажених надмірно [8].

Після виявлення таких пристроїв інженер зможе зробити висновок про те, чи є дані вузли «вузьким місцем» мережевої топології, чи дані пристрої схильні до якогось аномального небажаного впливу (Вірусна активність, нелегітимний трафік великих обсягів і т.д.). Якщо пристрій виявився «вузьким місцем», мережевий інженер повинен підняти питання заміні устаткування більш продуктивну модель. Якщо Високе завантаження є аномалією – потрібне подальше розслідування. Наприклад, за допомогою даних, отриманих за SNMP, ми можемо локалізувати "паразитний трафік", що завантажує мережеве обладнання. Засобами SNMP можна опитувати OID пристроїв, які відповідають за завантаження мережевих інтерфейсів Якщо «перевантаженим» пристроєм є комутатор, велика ймовірність, що на парі його інтерфейсів ми зможемо побачити аномально високий рівень завантаження. У багатьох випадках такі міркування актуальні й у маршрутизаторів. Після виявлення пари інтерфейсів інженер може уточнити, які пристрої підключені до даним портам. Можливо, для підтвердження висунутої гіпотези також відключити ці інтерфейси на якийсь час і подивитися, чи знизиться завантаження мережевого пристрою і чи поліпшиться зрештою якість зв'язку. Таким чином, збір інформації з SNMP допомагає виявити причину проблеми, розслідувати яку іншими засобами неможливо [3-8].

Необхідно відзначити, що проблема з погіршенням якості зв'язку може виявлятися не тільки внаслідок надмірного завантаження мережного обладнання, але і в результаті високого рівня утилізації каналів зв'язку. Знову ж таки, опитування SNMP

OID пристрою, що містить інформацію про завантаження мережевих інтерфейсів, допомагає виявити і опосередковано визначити завантаження підключених до інтерфейсів каналів. Найпростіший приклад. До інтерфейсу високопродуктивного маршрутизатора Cisco підключено WAN-канал. Пропускна спроможність каналу за договором із провайдером становить 10 Мбіт/с. Користувачі, знову ж таки, періодично мають проблеми з якістю зв'язку по цьому каналу. За допомогою збору інформації з SNMP ми визначили, що протягом усього часу використання каналу завантаження інтерфейсу маршрутизатора, до якого підключено канал, не перевищувала 3 Мбіт/с. Однак з деякого моменту часу спостерігається завантаження 10-12 Мбіт/с, що перевищує пропускну здатність каналу. Очевидно, проблема у надмірному рівні утилізації WAN-каналу. У цій ситуації подальше розслідування проблеми засобами SNMP важко: потрібно визначити якісний склад трафіку в каналі, тобто IP-адреси відправників/одержувачів протоколи та порти, що використовуються. Описане завдання вирішується за допомогою протоколу NetFlow.

Протокол NetFlow розроблено компанією Cisco Systems. З точки зору проблеми управління мережею, цей протокол є незамінним інструментом для моніторингу завантаження каналів передачі даних. Звичайно, протокол NetFlow не може отримувати інформацію безпосередньо з каналу (витої пари або оптичної лінії) – дані знімаються з пристроїв, підключених безпосередньо до сегменту, що цікавить. NetFlow підтримується багатьма мережевими пристроями. По NetFlow можуть надсилати інформацію маршрутизатори, комутатори та міжмережеві екрани Cisco. NetFlow є пропрієтарним протоколом Cisco Systems, однак, слід зазначити, що існує і відкритий аналог даного протоколу - sFlow. SFlow реалізований у сучасних моделях мережевих пристроїв багатьох виробників мережного обладнання – HP, Zyxel тощо [1-3].

Архітектура NetFlow вкрай проста і складається з двох компонентів: мережевий пристрій, що відправляє інформацію про трафік, що проходить через нього, і NetFlow-колектор. Останній є збирачем та аналізатором інформації, отриманої за NetFlow. Принцип дії протоколу ось у чому. На мережному устаткуванні при відкритті чергової сесії передачі формується інформація про даної сесії, звана потік (flow). Потік містить у собі таку інформацію, як кількість байт, що передаються, вхідний і вихідний

інтерфейс для сесії, IP-адреси джерела/приймача, порти джерела/приймача, номер протоколу IP, параметри QoS і т.д. Потоки акумулюються на мережному пристрої та відправляються у бік NetFlow-колектора в UDP-датаграмах. NetFlow-колектор агрегує отриману інформацію, проводить аналіз та формує зручні звіти та графіки. Один із популярних NetFlow-колекторів – NetFlow Analyzer, але існують колектори та інших виробників. За допомогою протоколу NetFlow мережевий інженер отримує повну картину трафіку каналами. Інженер може переглядати якісний склад трафіку (IP-адреси, порти, програми) у будь-якому сегменті мережі, а також оцінювати, який відсоток пропускної спроможності каналу займає той чи інший потік [1-3].

1.2. Види протоколу Netflow

NetFlow – відкритий приватний протокол, розроблений Cisco для відстеження трафіку мережі. Netflow дозволяє аналізувати мережевий трафік, базуючись на сеансах і робить запис про кожну операцію TCP/IP [1].

Дана система побудована за допомогою сенсора, колектора та аналізатора [1-3]:

- за допомогою ***сенсора*** збираються дані щодо трафіку, що передається. Зазвичай їх ставлять у вузлах мережі, що розташовані на граничних маршрутизаторах сегментів аналізованої мережі.
- ***колектор*** отримує інформацію від сенсорів та збирає її. Далі дані надсилаються до файлу для подальшої обробки. Варто зауважити, що різні колектори надсилають дані та зберігають їх у різних форматах.
- ***аналізатор***, або як його прийнято називати, система обробки, читає дані файли та формує їх у звіти, які більш зрозумілі та зручні для людського сприйняття, але ця система має бути сумісна з тим форматом даних, у яких її надав колектор для обробки. Але зазвичай колектор та аналізатор у новіших системах об'єднані в одну.

Зазвичай колектор та аналізатор є частинами одного програмного комплексу, що працює на сервері.

Необхідно одразу з'ясувати одну річ – колектор та його відповідний аналізатор є пасивними елементами системи. Сенсор відправляє дані про трафік колектору, той приймає, аналізатор фільтрує та аналізує отриману інформацію та заповнює базу даних на сервері. При правильно працюючому сервері нам немає необхідності власноруч підключати пристрої, які слід відстежувати на сервері. У той час як сенсор шле звіти інші елементи системи займаються своїми завданнями: колектор приймає та надсилає звіти, а аналізатор аналізує отримані дані. Поки сенсор надсилає звіти, колектор їх приймає, аналізатор реєструє. Під час вимкненого сенсора він просто ховається з режиму «онлайн» поточної статистики.

NetFlow використовує UDP або SCTP для передачі даних про трафік колектору. Відомо те, що колектор слухає порт 2055, 9555 або 9995 (або порт, який був вказаний безпосередньо при налаштуванні).

Сенсор виділяє з трафіку, що надходить потоки, що характеризуються такими параметрами [3]:

- адреса джерела;
- адресу призначення;
- порт джерела для UDP та TCP;
- порт призначення для UDP та TCP;
- тип та код повідомлення для ICMP;
- номер протоколу IP;
- мережевий інтерфейс (параметр ifindex SNMP);
- IP Type of Service.

Потоком вважається набір пакетів, які прямують у одному напрямі. Коли сенсор визначає, що потік закінчився (зміни параметрів пакетів, або скидання TCP - сесії), він відправляє інформацію в колектор. Залежно від налаштувань він також може періодично відправляти в колектор інформацію про потоки, що все ще йдуть. Це дуже важливий момент – при налаштуванні сенсора можна самостійно вирішити, за якими параметрами надіслана на колектор інформація буде об'єднана у звітах.

Для різнобічного аналізу трафіку традиційно застосовують кілька основних технологій – класичний мережевий аналіз, SNMP і NetFlow. Всі вони демонструють різний підхід до питання обліку трафіку, використовують різні способи отримання інформації. У результаті різні результати, що відрізняються повнотою та характером інформації.

Класичний мережевий аналіз для розгортання на мережі вимагає деяких витрат, апаратних та економічних, і виправданий не завжди. Особливо це стосується корпоративних мереж (навіть щодо великих), доцільніше застосовувати такий аналіз у мережах масштабу оператора зв'язку, провайдера, оскільки він здатний дати вичерпну інформацію про потоки інформації не лише від користувачів, а й від магістральних вузлів. Для корпоративних мереж така інформація виявиться надмірною, і знадобиться хороший фахівець для її аналізу. Тому у великих мережах зазвичай застосовують протокол ***SNMP*** – простий протокол керування мережами [3].

Принцип і основна ідея полягає в тому, що надсилаються короткі запити на інтерфейси мережевих пристроїв, а потім вже на підставі відповідей отримують інформацію про мережу. SNMP також надає інформацію та дані про завантаження процесора та пам'яті, він не спрямований суто на аналіз трафіку, а широко орієнтований.

NetFlow у свою чергу ідеально підходить для збору всієї можливої інформації саме про трафік, у розрізі користувачів, підмереж, потоків від конкретних додатків тощо. Сенсор апаратно вбудований в обладнання маршрутизації, і доступні всі процеси обробки інформації для аналізу сенсор NetFlow видає в значному обсязі, в ній міститься буквально все - від адресації джерела та одержувача до протоколів та інтерфейсів, що використовуються, мітки часу і т.д. Тому NetFlow найефективніший для обліку трафіку, зокрема пошуку вузьких місць у мережі, виявлення основних споживачів тощо.

Flexible NetFlow – це новітня технологія NetFlow. Flexible NetFlow розширює можливості початкового протоколу NetFlow, дозволяючи налаштовувати параметри аналізу трафіку відповідно до конкретних вимог адміністратора мережі. Технологія Flexible NetFlow дозволяє створювати складніші налаштування для

аналізу трафіку та експорту даних за допомогою компонентів налаштування, які можна використовувати повторно [2].

Flexible NetFlow працює з форматом експорту 9 версії, його особливістю є робота на шаблонах, які забезпечують можливість розширити формат потрібного запису та оновити NetFlow без зміни вже існуючого формату. Важливо відзначити, що багато корисних команд Flexible NetFlow було введено разом із версією Cisco IOS 15.1. На рис. 1.1 зображено приклад мережі з аналізатором трафіку.

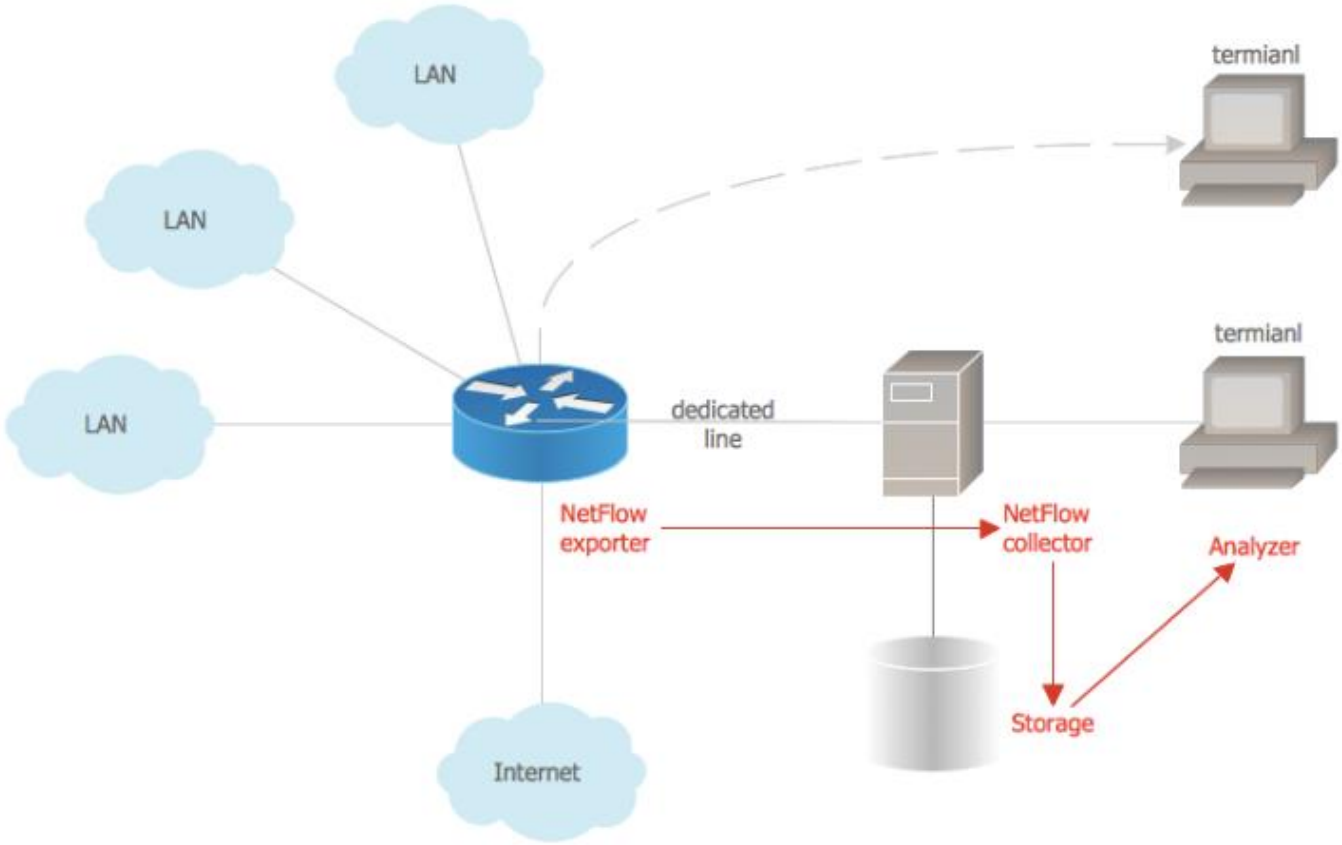

Рис. 1.1. Приклад реалізації мережі з аналізатором трафіку

Flexible NetFlow підтримує більше різних параметрів записів даних про потоки. Flexible NetFlow дозволяє адміністратору визначати записи для кешу контролю потоку Flexible NetFlow, вибираючи додаткові та обов'язкові поля, що визначаються користувачем, для налаштування збору даних відповідно до конкретних вимог. Записи для кешу контролю потоку Flexible NetFlow називаються записами, що визначаються

користувачем. Значення додаткових полів додаються до потоків для надання додаткової інформації про трафік у потоках (рис. 1.2). У разі зміни значення додаткового поля новий потік не створюється.

Якщо NetFlow для отримання даних про потік обробляє кожен пакет, зберігає його параметри в NetFlow Cache, а потім експортує їх до NetFlow Collector, протокол sFlow базується на аналізі статистичної вибірки пакетів [3].

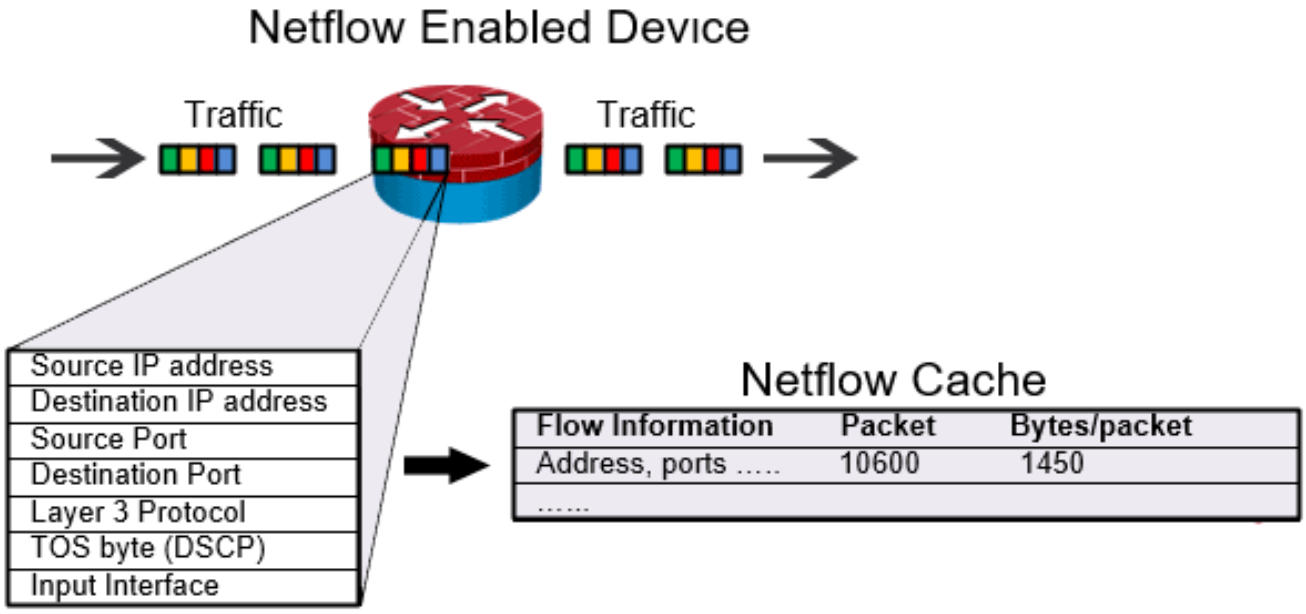

Рис. 1.2. Маршрутизатор Netflow і потік даних

Вперше цей метод продемонстрував HP на виставці Telecom'91. Однак широкого поширення дана технологія набула лише останніми роками, що було викликано появою високошвидкісних мереж та переходом до комутації пакетів. Розглянемо суть методу оцінки трафіку щодо вибірки пакетів на конкретному прикладі [9].

Нехай через мережу передається 1 млн пакетів. Випадкова вибірка 25% трафіку дає 2500 пакетів. Якщо 1000 з них представляє певний клас трафіку *V*, наприклад, голосовий трафік, то доречне питання про оцінку кількості пакетів цього класу в загальному трафіку. Він міститиме щонайменше 1000 пакетів класу *V*, оскільки вони потрапили у вибірку. Максимально можливе їх число – 998 500, оскільки у вибірці є 1500 неголосових пакетів. Взагалі, такі значення можливі, проте вкрай малоймовірні.

Найбільш правдоподібним буде припущення, що частина голосових пакетів у загальному трафіку близька до такої у вибірці, тобто приблизно 40%. Похибка оцінки обчислюється за допомогою статистичних методів.

Програмний модуль, що реалізує функції агента sFlow, працює як частина управляючого програмного забезпечення, що розташовується на мережному пристрої. Коли вибирається пакет, його заголовок витягується і поміщається в дейтаграму або деталізовану карту передачі пакета по мережі, яка включає заголовок, адреси та порти відправника та одержувача, статистику інтерфейсу та іншу інформацію, необхідну для аналізу трафіку на рівнях OSI від другого до сьомого. Сформована дейтаграма відразу ж подається по мережі в sFlow Collector. Один колектор може обробляти дані більш як від 20 тис. портів.

Вибір пакетів у типовому випадку виконується за допомогою комутувальних або маршрутизуючих замовних спеціалізованих мікросхем (**ASIC**) зі швидкістю, яку дозволяють фізичні з'єднання. Записується також стан елементів комутаційних та маршрутних таблиць, пов'язаних із кожним пакетом вибірки.

Слід зазначити, що sFlow, в порівнянні з іншими перерахованими варіантами, споживає мало ресурсів процесора, пам'яті та смуги пропускання, що є важливим фактором при виборі технології, за якою проводитиметься моніторинг трафіку у високошвидкісних мережах.

1.3. Структура роботи протоколу

Принцип дії протоколу ось у чому. На мережному устаткуванні, при відкритті будь-якого сеансу передачі, складається інформація про наданому сеансі, що називається потоком (flow).

Відомості про потік включають різну інформацію, таку як, кількість байтів, що передаються, вхідний і вихідний інтерфейси для сеансу, тип і код повідомлення для ICMP, IP-адреси відправника / отримувача, порти (UDP і TCP) відправника / отриму-

вача, номер протоколу IP, параметри QoS і т. д. Вище перелічені відомості про потоки накопичуються на мережному пристрої та надсилаються колектору NetFlow у вигляді датаграм UDP [1-6].

Після отримання інформації колектор обробляє її, аналізує та створює зручні для сприйняття звіти та графіки.

Існують різні колектори NetFlow від різних фірм, таких як NetFlow Analyzer, Scrutinizer, PRTG Network Monitor та інші. Кожна програма має свої плюси та мінуси.

Протокол NetFlow надає повну картину трафіку в каналах, що дозволяє побачити якісний його склад (IP-адреси, порти, програми) у будь-якому сегменті мережі. NetFlow пропонує нам проаналізувати, яку частку пропускної спроможності каналу (у відсотковому співвідношенні) займає якийсь потік [10].

NetFlow застосовується в різних областях, але три ключових: моніторинг трафіку, аналіз поведінки мережі, проведення аудитів мережевої інфраструктури. Провайдери зв'язку компанії особливо зацікавлені в обліку трафіку. Під час проведення аудиту NetFlow допомагає збирати повну та детальну інформацію про реальний трафік та завантаження мережі.

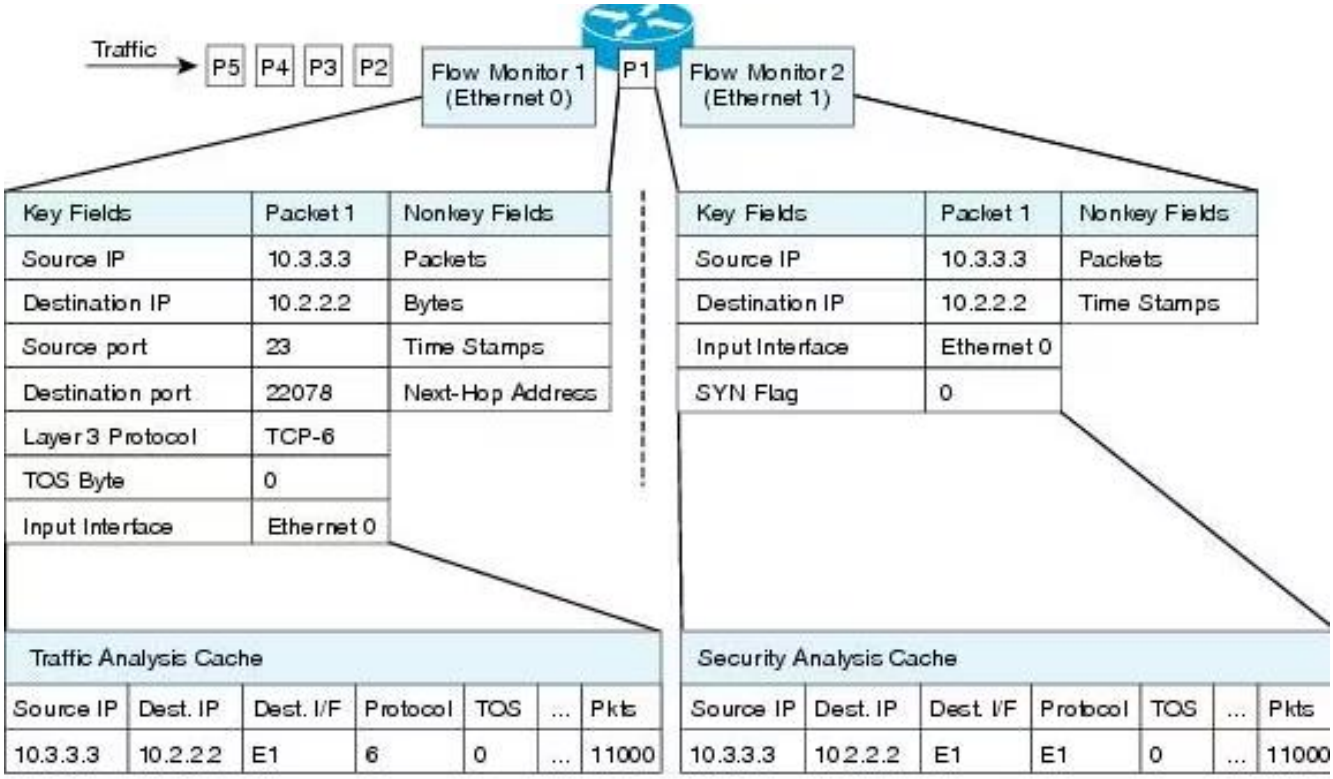

Рис. 1.3. Запис Netflow

РОЗДІЛ 2
ІНФОРМАЦІЙНА СИСТЕМА МОНІТОРИНГУ ТА АНАЛІЗУ ІНТЕРНЕТ ТРАФІКУ НА БАЗІ ПРОТОКОЛУ NETFLOW

2.1. Огляд мережі та її окремих компонентів

Кожна компанія має певний характер розв'язуваних завдань, який найчастіше формулює вимоги до конфігурації локальної мережі. Від числа працівників залежить кількість робочих станцій. Важливу роль грає внутрішня ієрархія підприємства. Для фірми з горизонтальною структурою оптимальним рішенням є проста однорангова мережа, тому що всі співробітники повинні мати доступ до даних. Навпаки, фірмі з вертикальною структурою більше підійде варіант мережі з виділеним сервером, тому що в цьому випадку має бути достеменно відомо, який співробітник до якої інформації повинен мати доступ, а до якої ні. У такому варіанті мережі існує можливість адміністрування та визначення прав доступу.

В даному випадку розглядається невелике підприємство, що має локальну мережу з 19 робочих станцій та файловий сервер з розмежованим доступом до інформації. Залежно від типу мережі під час проєктування зазвичай виникає питання обмеження довжини кабельного сегмента. Якщо розглядати ситуацію, коли мережа охоплює кілька поверхів будівлі, постає питання використання репітерів і комутарів. У ситуації з нашою організацією вся мережна інфраструктура розташовується на одному поверсі будівлі, і відстань між сегментами мережі не така велика, щоб використовувати репітери. Логічну схема вихідної мережі підприємства наведено рис. 2.1.

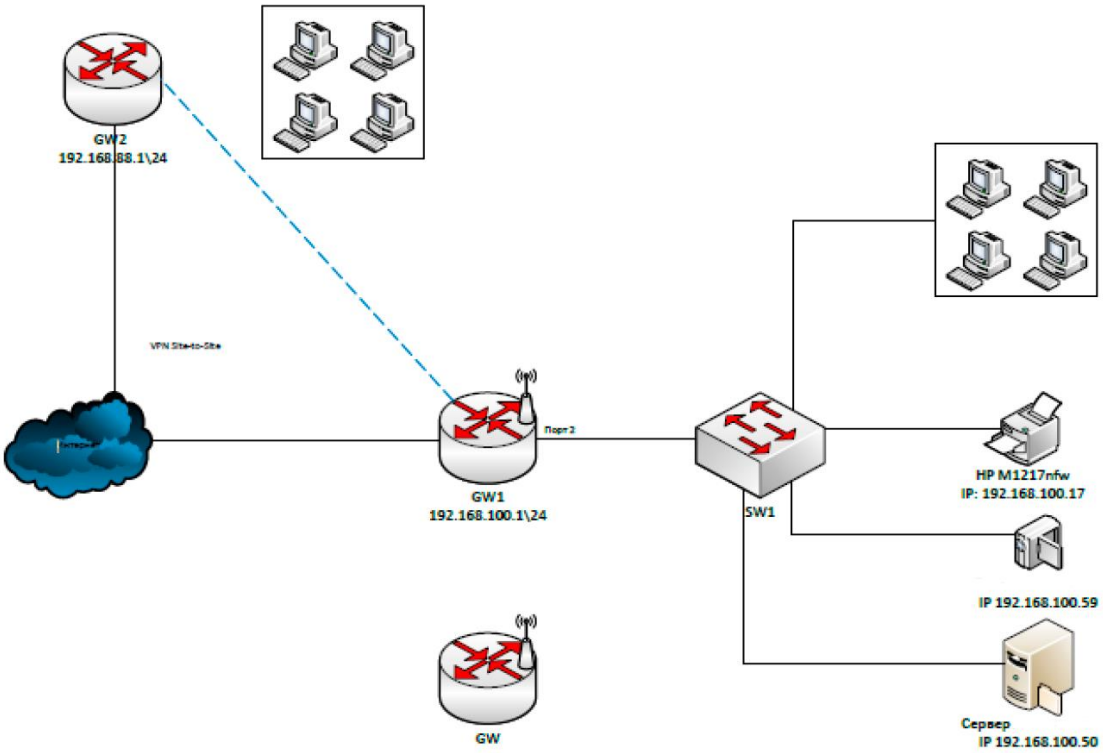

Рис. 2.1. Схема мережі

Для стабільної роботи роль маршрутизатора виконує Mikrotik RB951Ui-2HnD (рис.2.2). Це високопродуктивний роутер Wi-Fi як для офісу, так і для дому. Як процесор використовується Atheros AR9344 з частотою 600 МГц, а разом з 128 Мб оперативної пам'яті маршрутизатор показує себе вдвічі потужніше, ніж пристрої цієї серії, що випущені раніше. Роутер обладнаний 5-ма Ethernet портами. Ще є можливість отримувати живлення за технологією PoE та надати електроживлення за цією ж технологією на інший пристрій вже через п'ятий порт.

Рис. 2.2. Зовнішній вигляд маршрутизатора

Маршрутизатор поставляється в комплекті з встановленим програмним комплексом Mikrotik RouterOS з ліцензією четвертого рівня, що робить роботу з пристроєм зручнішим та швидшим. RouterOS може мати як консольний вигляд (рис. 2.3), так і графічний (рис. 2.4). За функціоналом ці два інтерфейси ідентичні, тому який використовувати – справа лише смаку.

Цей маршрутизатор повністю відповідає вимогам мережі та забезпечує її безперебійну та якісну роботу. У своєму ціновому сегменті цей варіант також є кращим.

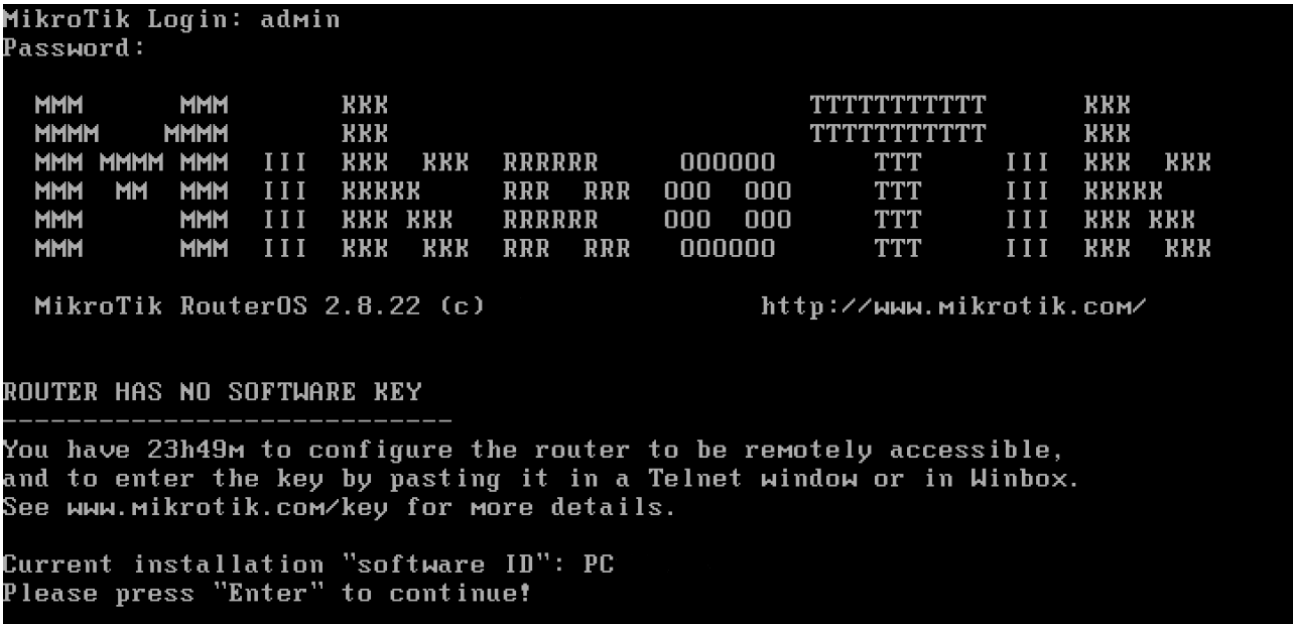

Рис. 2.3. Консольный варіант RouterOS

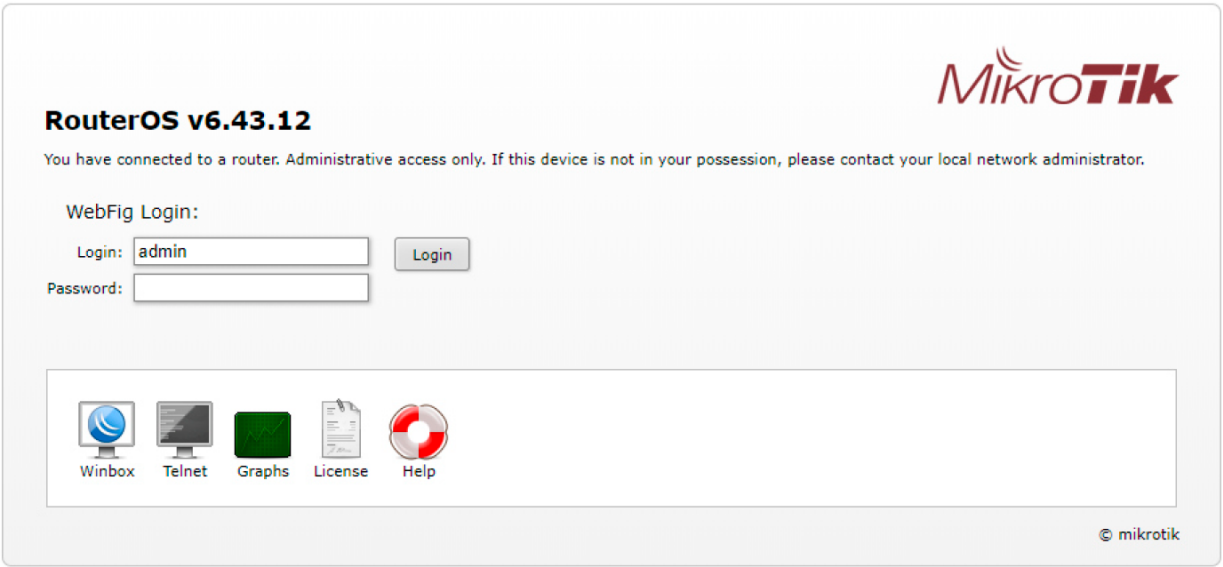

Рис. 2.4. Графічний варіант RouterOS

Крім маршрутизатора, до мережних пристроїв цієї мережі належить ще й комутатор (свіч). Свіч – це мережний пристрій, який забезпечує з'єднання вузлів мережі для організації єдиної системи доступу користувачів зі своїх робочих станцій до мережних ресурсів. Для комутації в мережі компанії використовується свічка марки Cisco Catalyst 2960 (рис. 2.5).

Cisco Catalyst 2960 - нове сімейство комутаторів другого рівня з фіксованою конфігурацією, що дозволяє підключати робочі станції до мереж Fast Ethernet та Gigabit Ethernet на швидкості середовища передачі, задовольняючи зростаючі потреби у пропускній здатності на периферії мережі. Для агрегації застосовуються комбіновані гігабітні uplink-порти, які можуть поєднуватися в єдиний канал за технологією GigabitEtherChannel.

Ця серія комутаторів орієнтована насамперед на підприємства малого та середнього бізнесу. У порівнянні з популярною серією комутаторів Catalyst 2950 моделі сімейства 2960 забезпечують ширший набір функцій забезпечення безпеки та якості обслуговування, а також керування смугою пропускання. У комутаторах серії Catalyst 2960 для полегшення процесу конфігурування була передбачена функція Smartports, яка дозволяє виконати основні налаштування порту комутаторів, ґрунтуючись на його призначенні. Cisco Catalyst 2960 забезпечують потребу передачі даних зі швидкістю 100 Мбіт/сек і 1 Гбіт/сек, дозволяють використовувати LAN сервіси, наприклад, для мереж передачі даних, побудованих у філіях корпорацій. Сімейство Catalyst 2960 дозволяє забезпечити високу безпеку даних за рахунок вбудованого NAC, підтримки QoS та високого рівня стійкості системи.

Рис. 2.5. Комутатор Cisco Catalyst 2960

До основних характеристик цієї моделі можна віднести:

- високий рівень безпеки, удосконалені списки контролю доступу (ACL);
- вбудовані порти подвійного призначення, що функціонують як для міді, так і оптоволокна. Кожен порт має вбудований порт 10/100/1000 Ethernet і порт SFP Gigabit Ethernet порт. При цьому одночасно активним може бути лише один із портів;
- організація контролю мережі та оптимізація ширини каналу з використанням QoS, диференційованого обмеження швидкості та ACL.

Щоб забезпечити безпеку мережі, комутатор використовує кілька методів автентифікації користувача, технології шифрації даних, організацію розмежування доступу до мережевих ресурсів на підставі ідентифікатора користувача, порту або MAC-адреси.

Cisco Catalyst 2960 є лідером у співвідношенні ціна / продуктивність. У даного комутатора високий рівень безпеки, вбудована енергозберігаюча технологія, що важливо для офісної будівлі. Мережним адміністраторам швидко і легко знаходити та усувати проблеми в мережі завдяки підтримуваній функції Loopback Detection та діагностики кабелю.

Локальна мережа забезпечує фізичне з'єднання робочих станцій з файловим сервером, віддалених один від одного, і складається з:

- серверу;
- 19 персональних комп'ютерів;
- мережного кабелю;
- мережевих адаптерів;
- 1 комутатора;
- 1 маршрутизатора.

Лінія зв'язку між сегментами мережі призначена для передачі текстової інформації, а саме різних документів для роботи співробітників, тому вимоги до пропускної спроможності лінії зв'язку не є критичними і повинні становити не менше 50 Мбіт/с.

Локальна мережа повинна забезпечувати:

- обмін інформацією між сегментами мережі;
- роботу програмного забезпечення у мережевому режимі;
- спільне використання доступу до Інтернету.

При побудові локальної мережі було враховано, що мережа не складна у модифікації і не залежить від роботи однієї з робочих станцій.

2.2. Налаштування мережного обладнання

Для того, щоб реалізувати на практиці систему контролю інтернет-трафіку, необхідно виконати такі завдання:

- увімкнути на зовнішньому інтерфейсі Mikrotik захоплення NetFlow - статистики;
- надіслати отриману статистику до NAS (наприклад, у службу `flow-tools`, через `flow-capture`);

У Mikrotik є своя реалізація протоколу Netflow, яка повністю сумісна зі стандартом Cisco і називається **Mikrotik Traffic Flow**. За допомогою Traffic Flow мережевий адміністратор може виявляти проблеми, які можуть виникнути у мережі, аналізувати та оптимізувати загальні характеристики мережі. Оскільки Traffic Flow сумісний повністю з Cisco Netflow, то він може також використовуватися утилітами, розробленими для Netflow.

Traffic Flow підтримує такі версії Netflow [10-11]:

- ***Version 1*** – перша версія протоколу, рекомендується використовувати лише тому випадку, коли відсутні альтернативи;
- ***Version 5*** – покращена версія першої версії Netflow;
- ***Version 9*** – нова версія, яка дозволяє додавати нові поля та типи записів завдяки шаблонному виконанню.

Для того, щоб почати збір статистики про трафік необхідно спочатку включити Traffic Flow і визначитися з інтерфейсом, з якого, власне, і проводитиметься збір. Робиться це за допомогою комбінації наступних команд:

```
/ip traffic-flow set enabled=yes interfaces=WAN
```

Після включення Traffic Flow та визначення з інтерфейсом, з якого потрібно отримувати інформацію про потоки, необхідно налаштувати хост призначення (колектор), який отримуватиме цю інформацію. Налаштовується наступною командою:

```
/ip traffic-flow target add dst-address=<192.168.50.100>
    port=9091 v9-template-timeout=1m version=5
```

Можна збирати трафік як з одного інтерфейсу, так і з декількох (у цьому випадку назви інтерфейсів слід вказати через кому). В даному випадку йде збір з інтерфейсу WAN, через який йде вихід до локальної мережі та до інтернету.

Як порт був обраний – 9091. При роботі використано версію v5 протоколу Netflow. IP-адреса 192.168.50.100 є адресою сервера-колектора. Можна вказати кілька колекторів, використовуючи різні версії протоколу та номери портів UDP.

Після цього маршрутизатор почне надсилати дані про потоки колектору.

2.3. Конфігурування сервера

Далі, як операційну систему, на якій слід встановити `Flow-Tools`, була обрана FreeBSD (рис. 2.6).

FreeBSD – це сучасна операційна система для настільних комп'ютерів, ноутбуків, серверів та систем з підтримкою великої кількості платформ.

В основі FreeBSD лежить операційна система 4.4.BSD-Lite Каліфорнійського Університету з деякими удосконаленнями з 4.4.BSD-Lite2. Також вона побічно базується на 386 BSD (BSD Net/2, перенесеної на платформу i386 TM Вільямом Джолітцем (William Jolitz)), хоча від початкового коду залишилося дуже мало.

FreeBSD використовується компаніями, інтернет-провайдерами, науковцями, професіоналами в обчислювальній техніці, студентами та рядовими користувачами по всьому світу для роботи, освіти та відпочинку.

Мета проєкту FreeBSD – надати швидку та стабільну операційну систему загального призначення, яку можна використовувати з будь-якою метою без будь-яких обмежень.

Налаштування `Flow-Tools` на FreeBSD виконується такими командами:

Встановлення NetFlow сенсора:

```
pkg install flow-tools
```

Налаштування запуску:

```
  echo 'flow_capture_enable="YES"' >> /etc/rc.conf.local
echo 'flow_capture_flags="-N-2"' >> /etc/rc.conf.local
```

Запуск:

```
service flow_capture start
```

Результат виконання команд можна побачити на рис. 2.7. На рисунку видно, що служба запущена без помилок і їй присвоєно PID = 799. Це означає, що `flow-tools` успішно встановлені, і система готова приймати пакети даних від маршрутизатора.

Рис. 2.6. Вікно операційної системи FreeBSD

```
root@osboxes:~ # pkg install flow-tools
Updating FreeBSD repository catalogue...
FreeBSD repository is up to date.
All repositories are up to date.
Checking integrity... done (0 conflicting)
The most recent version of packages are already installed
root@osboxes:~ # echo 'flow_capture_enable="YES"' >> /etc/rc.conf.local
root@osboxes:~ # echo 'flow_capture_flags="-N-2"' >> /etc/rc.conf.local
root@osboxes:~ #
root@osboxes:~ #
root@osboxes:~ # service flow_capture start
flow_capture already running?  (pid=799).
root@osboxes:~ #
root@osboxes:~ #
root@osboxes:~ #
```

Рис. 2.7. Результати введення команд

Як базу даних, у якій зберігатиметься інформація, зібрана колектором, було обрано MySQL.

MySQL – це одна з найпопулярніших та найпоширеніших СУБД (система управління базами даних) в інтернеті. Вона не призначена для роботи з великими обсягами інформації, але її застосування ідеально, наприклад, для інтернет сайтів, як невеликих, так і великих.

MySQL відзначається гарною швидкістю роботи, надійністю, гнучкістю. Робота з нею, як правило, не викликає великих труднощів. Підтримка сервера MySQL автоматично включається до постачання PHP.

Важливим фактором є її безкоштовність. MySQL поширюється за умов загальної ліцензії GNU (GPL, GNU Public License).

Завдання тривалого зберігання інформації часто зустрічається у програмуванні Web-додатків: підрахунок відвідувачів у лічильнику, зберігання повідомлень у форумі, віддалене управління змістом інформації на сайті і т.д.

Додаток на PHP, що використовує для зберігання інформації базу даних (зокрема MySQL) завжди працює швидше за програму, побудовану на файлах. Справа в тому, що бази даних написані мовою C++, і написати на PHP програму, яка працювала б з жорстким диском, ефективніша за базу даних - завдання нерозв'язне за визначенням, оскільки програми на PHP в принципі працюють повільніше, ніж програми на C++, оскільки PHP - Інтерпретатор, а C++ - компілятор.

Таким чином, основна перевага бази даних полягає в тому, що вона бере на себе всю роботу з жорстким диском і робить це дуже ефективно.

В операційній системі FreeBSD встановлений пакетний менеджер pkg, за допомогою нього і здійснюється встановлення нових компонентів у систему

Встановлення та підготовка СУБД MySQL для імпорту NetFlow даних:

Встановимо, запустимо та налаштуємо MySQL сервер:

```
pkg install mysql56-server
```

Запуск служби

```
echo 'mysql_enable="YES"' >> /etc/rc.conf service mysql
start
```

Початкове налаштування СУБД:

```
mysql_secure_installation
```

Встановимо Perl-модулі для роботи скрипта із СУБД:

```
pkg install p5-DBI p5-DBD-mysql
```

Входимо в СУБД і створюємо базу та користувача для роботи з нею:

```
mysql -u root -p
```

Далі необхідно створити саму базу даних та користувача для неї:

```
mysql> create database data_netflow;
mysql> grant insert, create, update, select, delete on
data_netflow.
mysql> flush privileges; mysql> exit;
```

Після успішного виконання вищевикладених команд можна впевнено сказати, що СУБД встановлена на операційній системі FreeBSD. Висновок команди, що відображає список створених інформаційних баз, можна побачити на рис. 2.8.

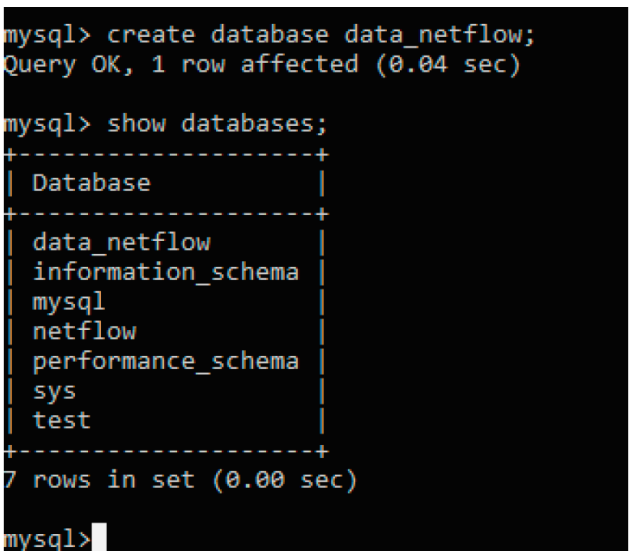

Рис. 2.8. Лістинг списку встановлених баз даних

РОЗДІЛ 3
РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ ЗБОРУ ТА СОРТУВАННЯ ІНФОРМАЦІЇ ПРО МЕРЕЖУ

3.1. Робота у середовищі Visual Studio

Microsoft Visual Studio – лінійка продуктів компанії Microsoft, що включають інтегроване середовище розробки програмного забезпечення та низку інших інструментальних засобів. Дані продукти дозволяють розробляти як консольні програми, так і програми з графічним інтерфейсом, у тому числі з підтримкою технології Windows Forms, а також веб-сайти, веб-програми, веб-служби як в рідному, так і в керованому кодах для всіх платформ, підтримуваних Windows, Windows Mobile, Windows CE, .NET Framework, Xbox, Windows Phone .NET Compact Framework та Silverlight.

Visual Studio включає редактор вихідного коду з підтримкою технології IntelliSense і можливістю найпростішого рефакторингу коду. Вбудований налагоджувач може працювати як відладчик рівня вихідного коду, так і відладчик машинного рівня. Інші вбудовані інструменти включають редактор форм для спрощення створення графічного інтерфейсу програми, веб-редактор, дизайнер класів і дизайнер схеми бази даних. Visual Studio дозволяє створювати та підключати сторонні доповнення (плагіни) для розширення функціональності практично на кожному рівні, включаючи додавання підтримки систем контролю версій вихідного коду (як, наприклад, Subversion та Visual SourceSafe), додавання нових наборів інструментів (наприклад, для редагування та візуального проєктування) коду предметно-орієнтованими мовами програмування) або інструментами для інших аспектів процесу розробки програмного забезпечення (наприклад, клієнт Team Explorer для роботи з Team Foundation Server).

12 листопада 2014 року було оголошено, що «Visual Studio 2015» прийнято як остаточний варіант імені продукту. В даній роботі використовується найсучасніша версія Visual Studio під назвою «Visual Studio 2022» з номером версії 17.5 (LTSC), що була представлена 11 квітня 2023 року.

Visual Studio 2022 надається в трьох редакціях: безкоштовної Community Edition, що поєднує всі Express-версії, і платних Professional Edition для невеликих проєктів та Enterprise Edition для великих проєктів.

Рис. 3.1. Логотип інтегрованого середовища розробки програмного забезпечення Visual Studio 2022

Visual Studio надає доступ до цілого ряду інших утиліт, які дозволяють переглядати та змінювати різні аспекти комп'ютера або мережі, не залишаючи середовища розробки. Завдяки цим інструментам можна переглядати служби, що виконуються, та активні з'єднання з базами даних, заглядати в таблиці на сервері SQL Server і навіть відвідувати веб-сайти з використанням вікна Microsoft Edge.

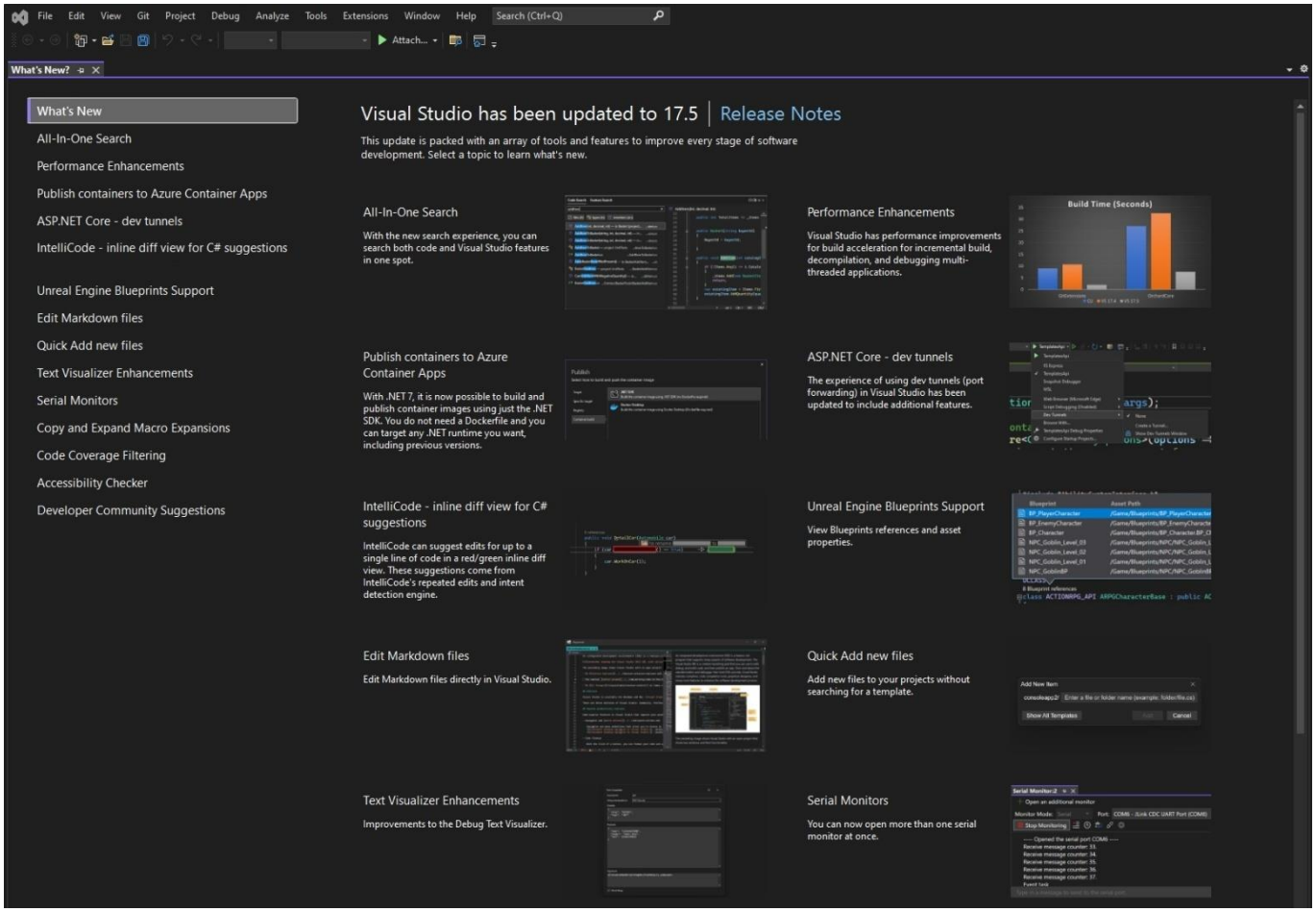

Рис. 3.2. Перелік основних змін та можливостей Visual Studio 2022 ver. 17.5

Visual Studio дозволяє отримувати доступ до документації MSDN прямо із середовища IDE. У разі, наприклад, виникнення сумнівів щодо призначення того чи іншого ключового слова під час роботи з текстовим редактором, можна виділити це ключове слово і натиснути <F1>, в результаті чого Visual Studio автоматично підключиться до MSDN і відобразить відповідні розділи довідки. Аналогічно, якщо потрібно подивитися, що означає та чи інша помилка компіляції, потрібно виділяти повідомлення з помилкою та натиснути <F1>.

Також Visual Studio містить графічні редактори та конструктори XML, що забезпечує підтримку розробки програм Windows, орієнтованих на мобільні пристрої, підтримку розробки програм Microsoft Office та Windows Workflow Foundation, містить вбудовану підтримку рефакторингу коду та інструменти візуального конструювання класів.

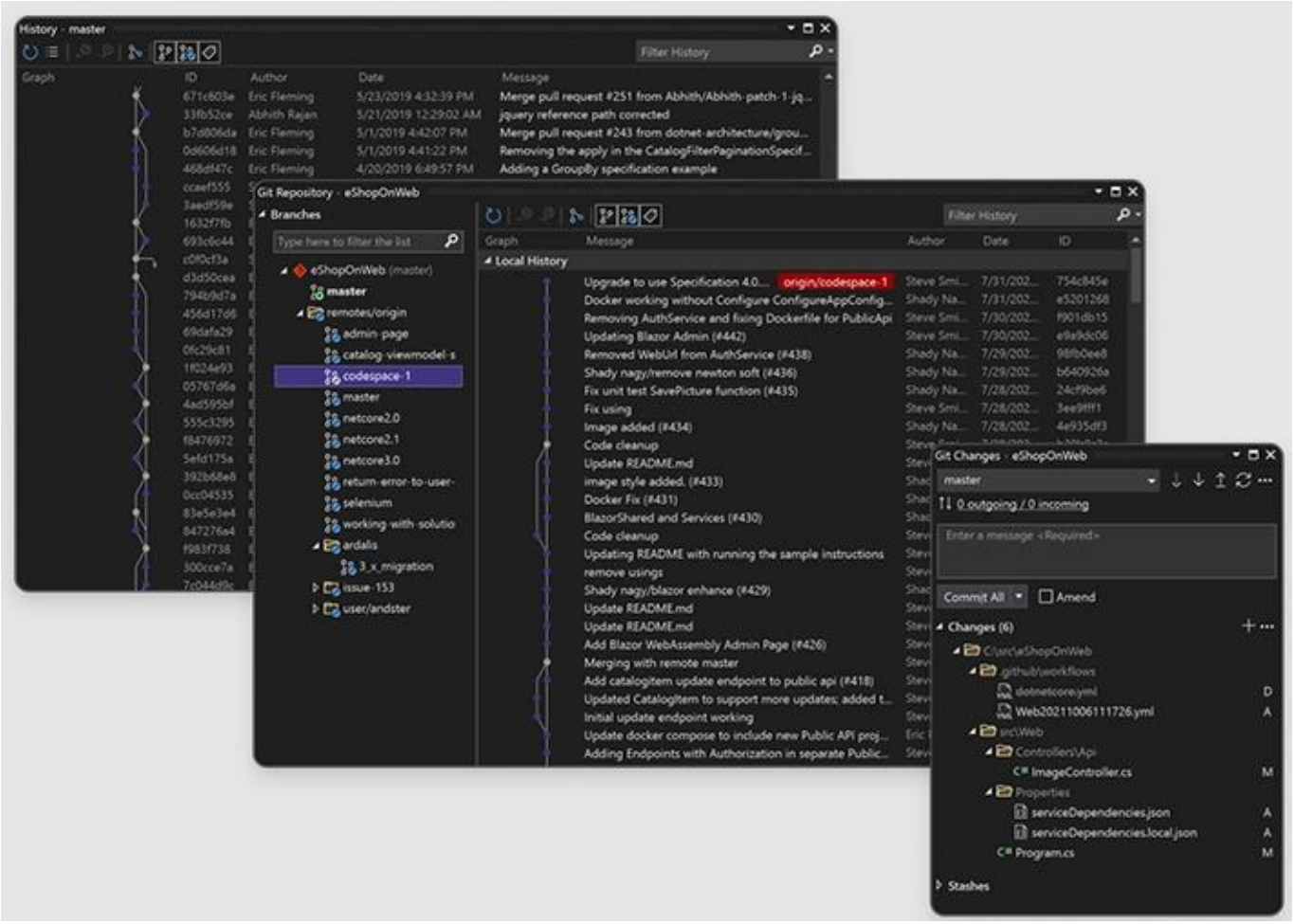

Рис. 3.4. Початкове вікно програми

Після інсталяції Visual Studio можна приступати до створення першого проєкту. У Visual Studio рідко коли потрібно починати з порожнього файлу і додавання коду C#. Зрозуміло, що можливість створення порожнього проєкту програми існує. Це потрібно, якщо справді виникла потреба в написанні коду з нуля, або при створенні рішення, яке має містити декілька проєктів.

Натомість, необхідно просто вказати Visual Studio, проєкт якого типу має бути створений, і середовище автоматично згенерує файли та код C#, що утворюють відповідний зазначеному типу проєкту каркас. Далі залишиться додати до цього каркасу власний код.

При розробці програми з інтерфейсом користувача, на зразок програми Windows, бібліотеки елементів керування Windows або програми ASP.NET, потрібно використовувати вікно Design View (Конструктор).

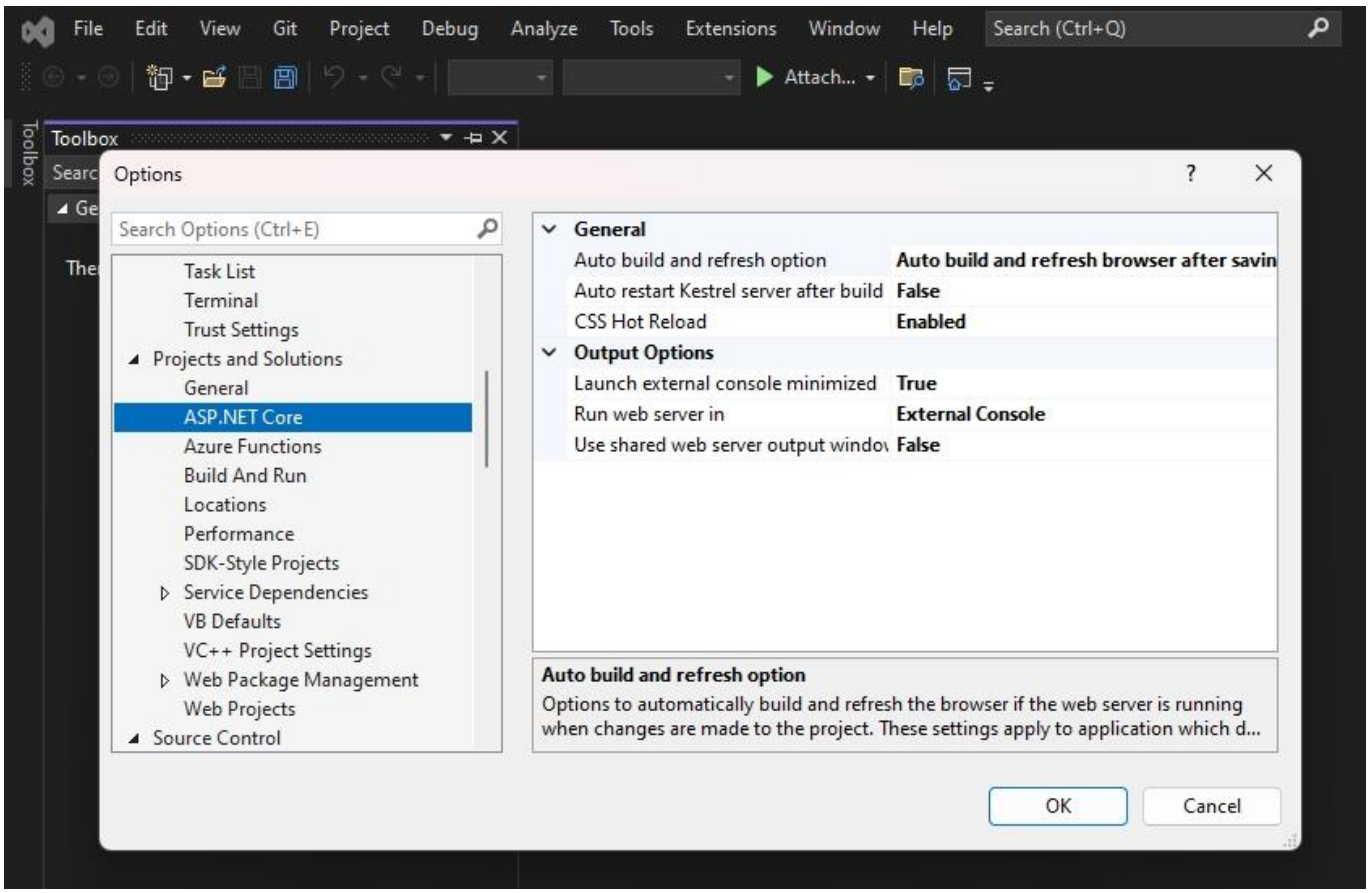

Рис. 3.4. Приклад налаштування бібліотеки ASP.NET

У цьому вікні пропонується візуальне уявлення про те, як виглядатиме форма. Зазвичай вікно Design View застосовується разом із ще одним вікном, яке називається Toolbox (Панель інструментів). В цьому вікні відображається величезна кількість компонентів .NET, які можна перетягувати в програму, що розробляється, як показано на малюнку 3.5.

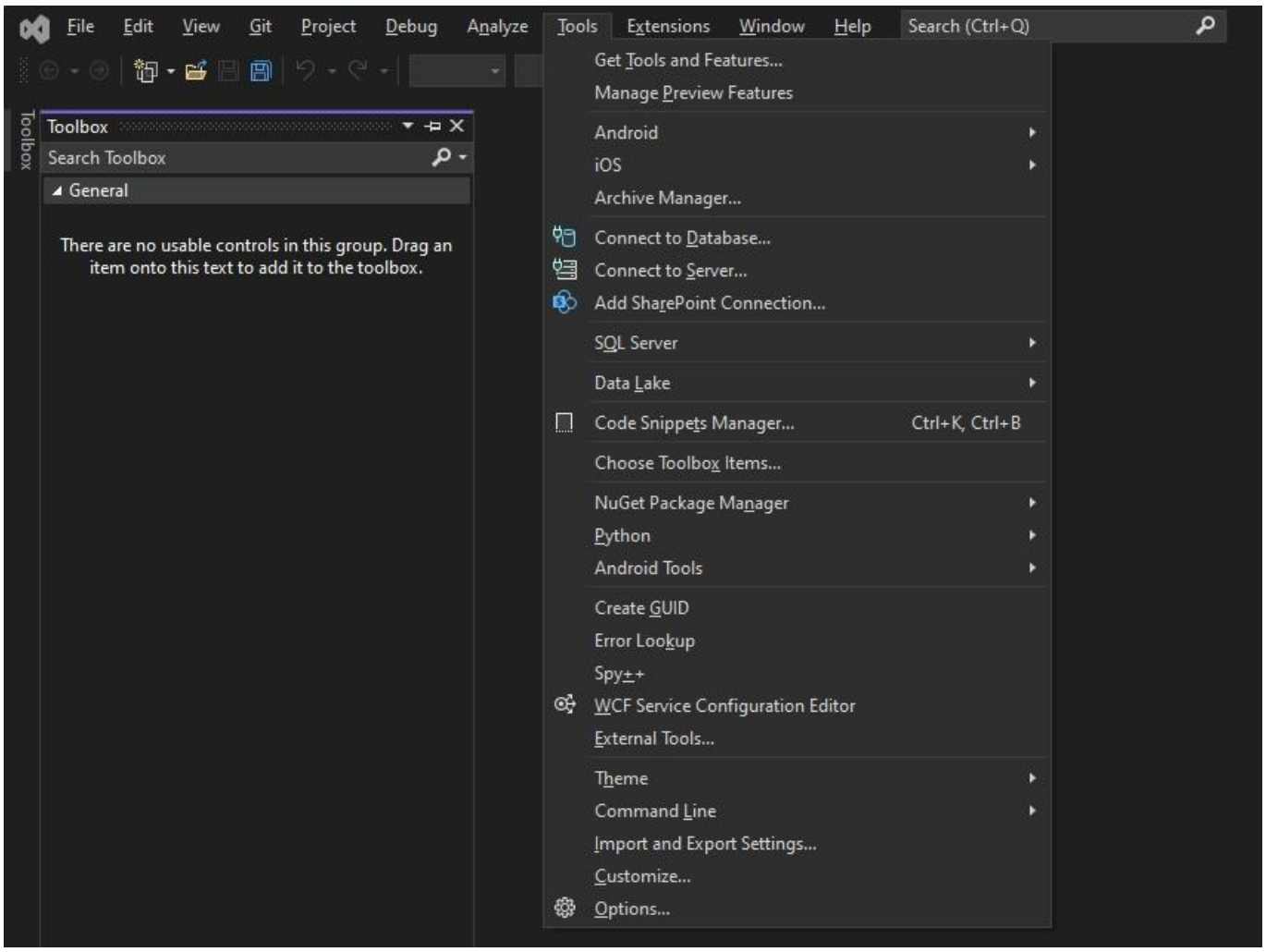

Рис. 3.5. Панель управління

Вікно Properties (рис.3.6) теж бере свій початок ще зі старої версії IDE-середовища Visual Basic. Насправді базові класи .NET, які представляють форми та елементи управління, мають безліч властивостей. Ці властивості визначають зовнішній вигляд і поведінку, наприклад, Width, Height, Enabled (що вказує, чи дозволено введення в даному елементі керування) або Text (текст, що відображається в даному елементі керування), і Visual Studio відомо про багато з них. Вікно Properties дозволяє редагувати початкові значення більшості таких властивостей тих елементів управління, які Visual Studio виявляє під час читання вихідного коду.

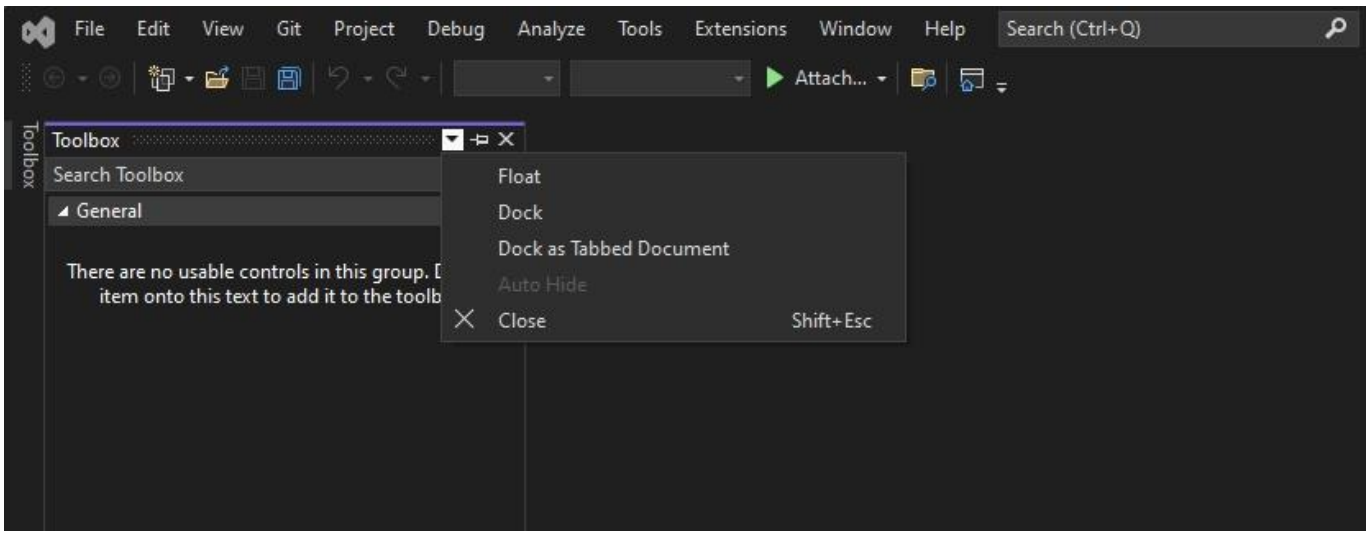

Рис. 3.7. Вкладка Search Toolbox

Як мову програмування було обрано мову C#. **C#** - це витончена об'єктно-орієнтована мова зі строгою типізацією, що дозволяє розробникам створювати різні безпечні та надійні програми, що працюють на платформі .NET Framework. C# можна використовувати для створення клієнтських додатків Windows, XML-веб-служб, розподілених компонентів, додатків клієнт-сервер, додатків баз даних і т. д. Visual C# надає розвинений редактор коду, зручні конструктори інтерфейсу користувача, інтегрований налагоджувач та багато інших засобів, які спрощують розробку додатків мовою C# для платформи .NET Framework.

Синтаксис C# дуже багатий, але при цьому простий та зручний у вивченні. Характерні фігурні дужки C# миттєво впізнаються всіма, хто знайомий із C, C++ чи Java. Розробники, які знають будь-яку з цих мов, зазвичай дуже швидко починають ефективно працювати в C#. Синтаксис C# спрощує багато складнощів C++, але при цьому надає відсутні в Java потужні функції, наприклад типи значень, що обнулюються, перерахування, делегати, лямбда-вираження і прямий доступ до пам'яті. C# підтримує універсальні методи та типи, які забезпечують більш високий рівень безпеки та продуктивності, а також ітератори, що дозволяють визначати в класах колекцій власну поведінку ітерації, яка може легко застосувати у коді клієнта. Вирази LINQ створюють дуже зручну мовну конструкцію для строго типізованих запитів.

Процес побудови в C# простіше проти C чи C++, але гнучкіший, ніж Java. Окремі файли заголовка не використовуються, і немає необхідності оголошувати методи

та типи у визначеному порядку. Вихідний файл C# може визначити будь-яку кількість класів, структур, інтерфейсів та подій.

3.2. Інтерфейс програми

Було поставлено завдання написати зручний графічний інтерфейс для виведення даних з бази даних MySQL. Системному адміністратору має бути зручно шукати необхідну мережеву інформацію, відображати за фільтрами. Зрештою було написано 2 форми:

- Форма з авторизацією;
- Форма програми.

Форма авторизації (рис. 3.8) складається з двох елементів TextBox, трьох елементів Label та однієї кнопки Button. Після натискання кнопки відбувається звіряння введених аутентифікаційних даних з тими, що прописані у файлі `enter.cs`. Під час введення неправильних облікових даних виходить повідомлення «Неправильний логін або пароль».

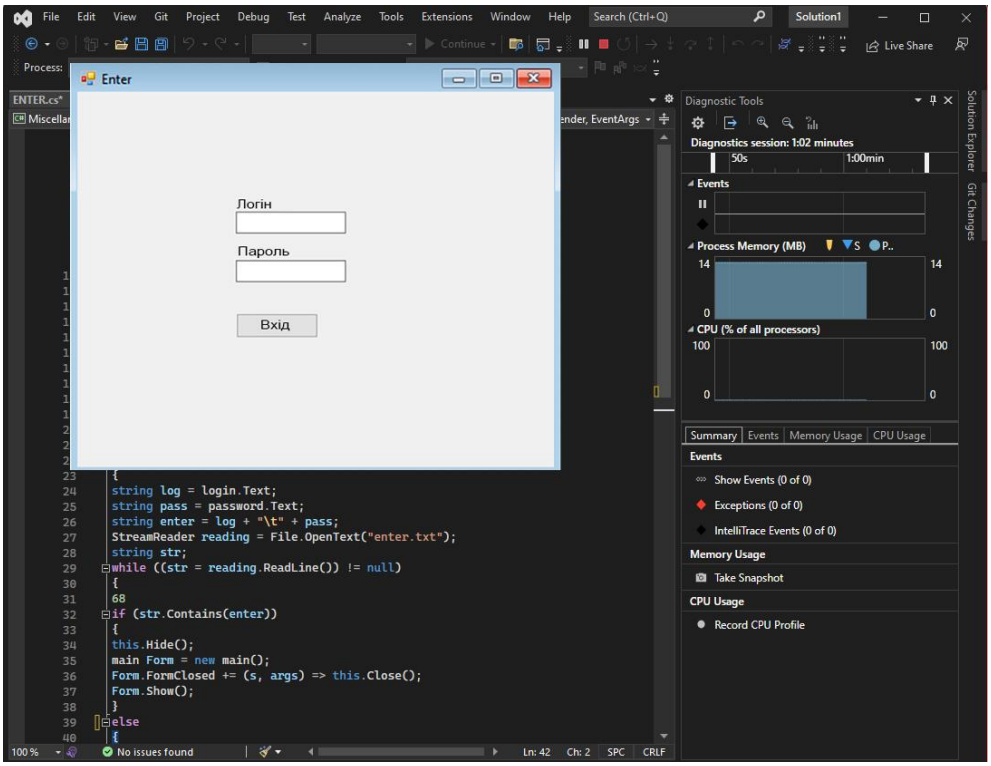

Рис. 3.8. Реалізація форми ENTER.cs у програмному середовищі Visual Studio 2022

При введенні пароля включена функція приховування пароля від сторонніх очей, тому у стані, що працює, дана форма виглядає так, як на рис. 3.8. Лістинг коду в програмному середовищі Visual Studio 2022 ver. 17.5 представлений на рис. 3.9.

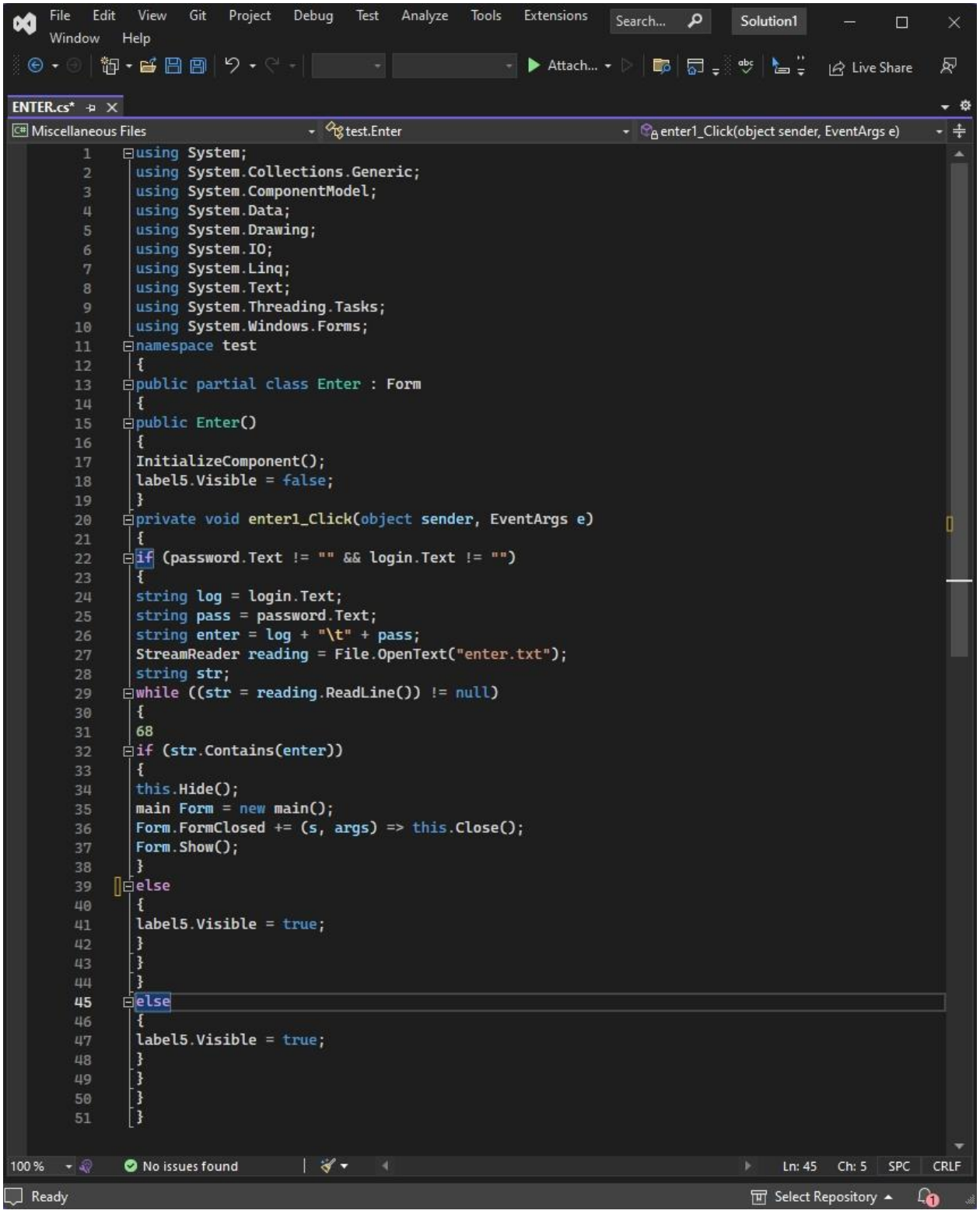

Рис. 3.9. Лістинг коду в програмному середовищі Visual Studio 2022 ver. 17.5

Крім форми з авторизацією є основна частина програми під назвою FORMULA.cs, код якої представлено на рис. 3.10-3.12.

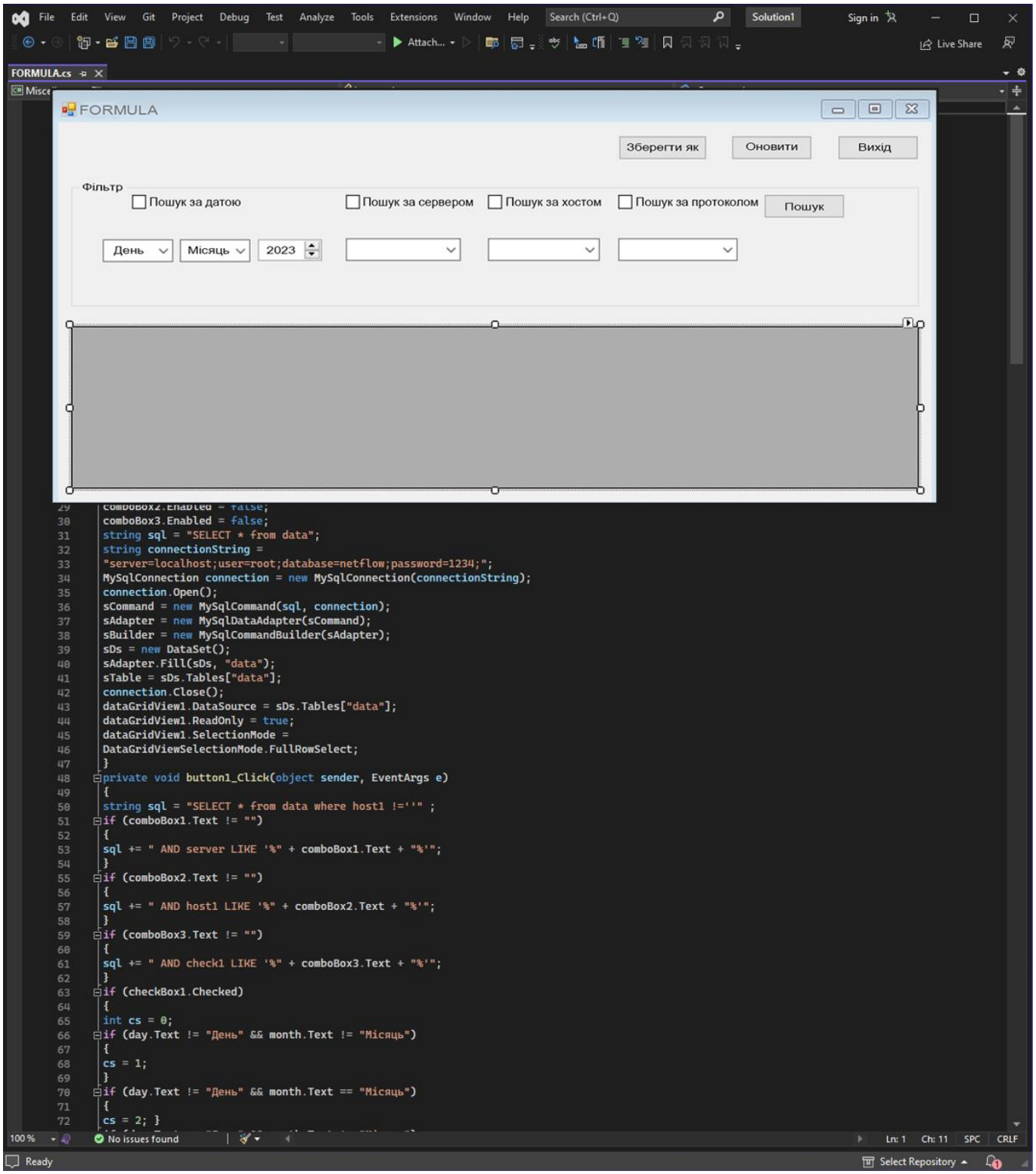

Рис. 3.10. Вікно програми FORMULA.cs

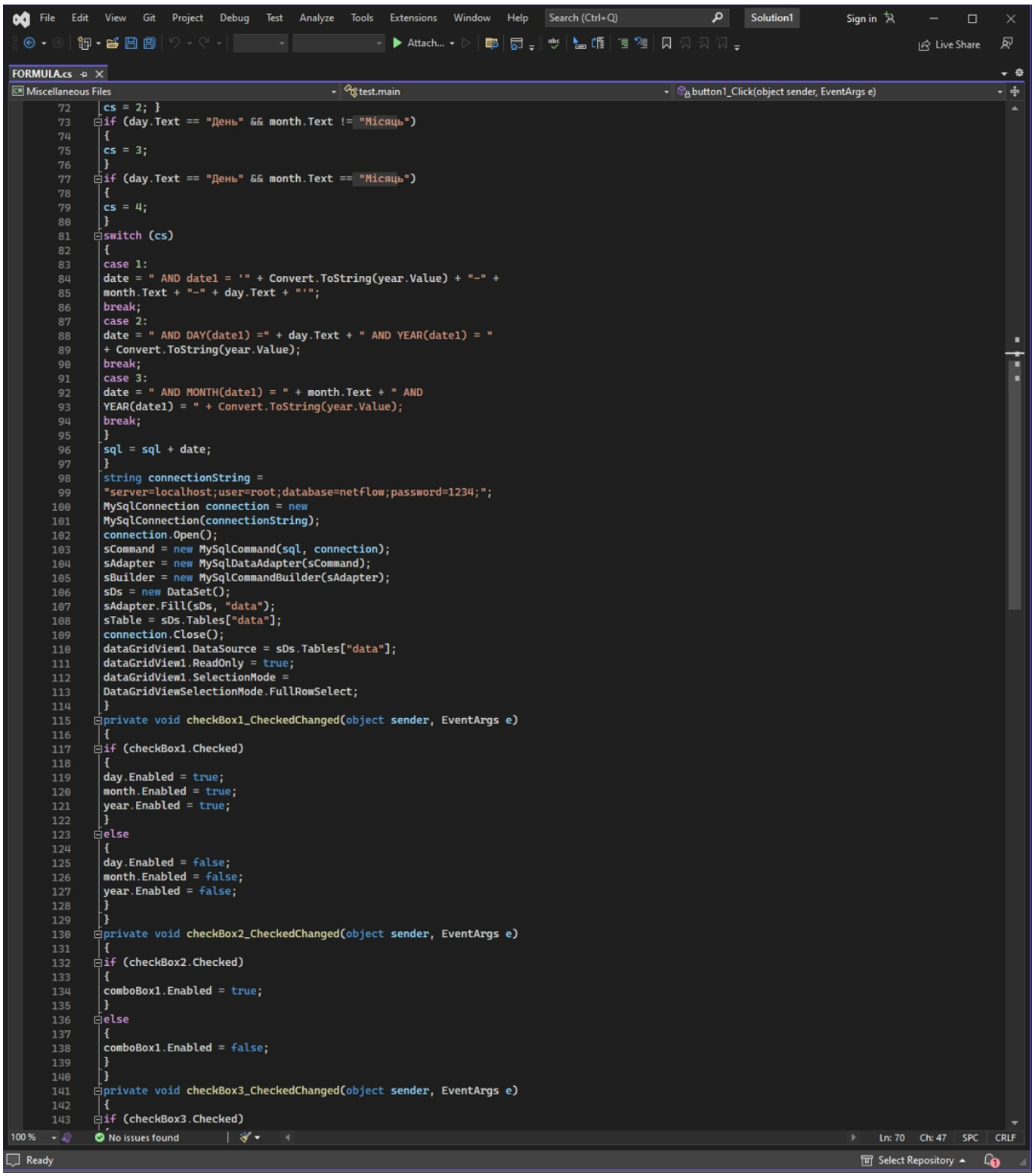

Рис. 3.11. Код програми FORMULA.cs у програмному середовищі Visual Studio 2022

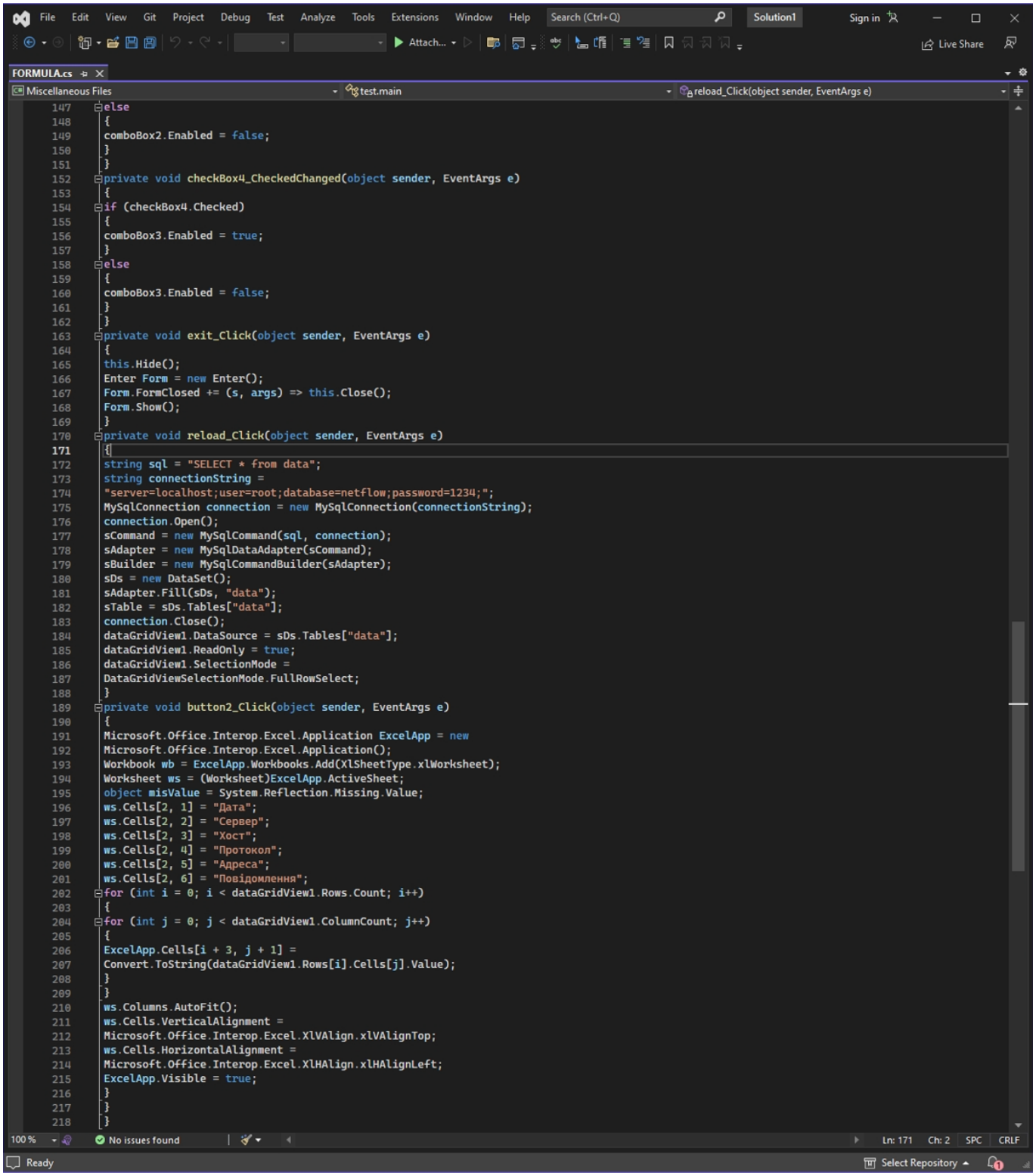

Рис. 3.12. Код програми FORMULA.cs у програмному середовищі Visual Studio 2022

DataGridView призначений для виведення таблиці з бази даних MySQL. Цей елемент управління має безліч налаштувань.

Після успішного входу в програму друга форма автоматично відкриється. Пошук із зазначенням фільтрів допомагає швидше знайти потрібну інформацію. Можна вказати як дату, і сервер, якщо він відомий, і протокол передачі.

Кнопка «Оновити» оновлює інформацію та підвантажує нову базу даних. Якщо натиснути кнопку «Зберегти як», можна отримати вивантаження поточного звіту до excel-файл (рис. 3.13). Кнопка «Вихід» повертає до попередньої форми з авторизацією.

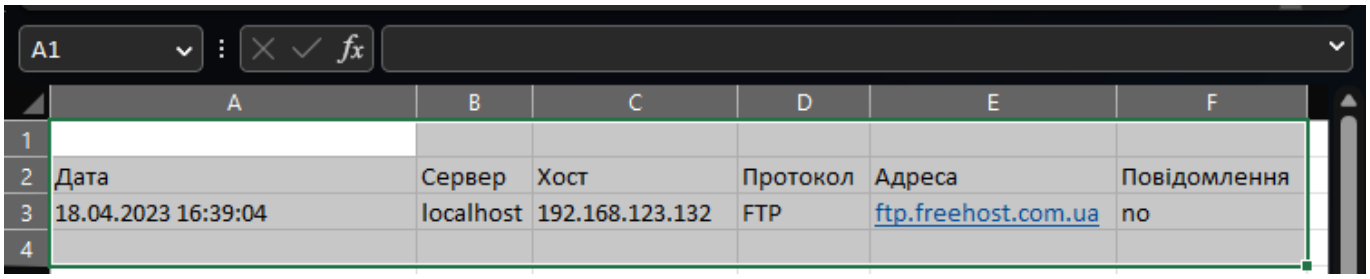

Рис. 3.13. Вигляд вивантаженого поточного звіту в програмному пакеті Microsoft Excel 2021

ВИСНОВКИ

Правильно налаштована система моніторингу мережі дозволяє істотно зменшити час реагування на проблеми, що виникли в мережі, контролювати вхідний і вихідний трафік і мінімізувати паразитний трафік в локальній мережі.

При побудові системи необхідно враховувати специфікацію мережі, пропускну здатність каналів зв'язку, оскільки неправильно налаштована система може не тільки не виконувати своїх прямих обов'язків, а ще й зіпсувати зв'язок у існуючій мережі.

Використання протоколу NetFlow в локальній мережі підприємства може бути корисним з багатьох причин. Спочатку, протокол NetFlow дозволяє збирати детальну статистику щодо трафіку в мережі. Це може допомогти адміністраторам мережі отримати більш глибоке розуміння того, як використовується їхня мережа. Наприклад, вони можуть виявити надмірні обсяги трафіку в певні години дня або трафік, що генерується певними додатками або пристроями.

Крім того, використання NetFlow може допомогти виявити можливі загрози безпеці мережі, такі як атаки або шкідливе програмне забезпечення. Аналіз NetFlow може допомогти виявити незвичайні мережеві активності, які можуть бути зумовлені зловмисниками, що намагаються зламати мережу.

Загалом, використання протоколу NetFlow може допомогти покращити продуктивність та безпеку локальної мережі підприємства, забезпечуючи адміністраторам мережі засоби для отримання детальної інформації про трафік та можливі загрози.

Після виконаних робіт можна зробити такі висновки:

- маршрутизатор виробника Mikrotik є оптимальним варіантом для компанії, що входить до сегменту малого бізнесу;
- сервер, на якому встановлена FreeBSD, швидко обробляє запити, тому що дана операційна система оперативно взаємодіє апаратним забезпеченням комп'ютера;

- рішення на основі NetFlow є найбюджетнішим засобом виявлення вторгнення, тому що не передбачають використання відводів трафіку та будь-яких методів виявлення аномалій шляхом включення обладнання у розрив;
- розроблений додаток для виведення даних працює стабільно та дозволяє фільтрувати дані залежно від потреб мережевого інженера.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. D. González-Sánchez et al., "Model-Driven Network Monitoring Using NetFlow Applied to Threat Detection," 2022 IEEE 8th International Conference on Network Softwarization (NetSoft), Milan, Italy, 2022, pp. 450-455.

2. M. H. Haghighat, Z. A. Foroushani and J. Li, "SAWANT: Smart Window Based Anomaly Detection Using Netflow Traffic," 2019 IEEE 19th International Conference on Communication Technology (ICCT), Xi'an, China, 2019, pp. 1396-1402.

3. X. Liu, Z. Tang and B. Yang, "Predicting Network Attacks with CNN by Constructing Images from NetFlow Data," 2019 IEEE 5th Intl Conference on Big Data Security on Cloud (BigDataSecurity), IEEE Intl Conference on High Performance and Smart Computing, (HPSC) and IEEE Intl Conference on Intelligent Data and Security (IDS), Washington, DC, USA, 2019, pp. 61-66.

4. J. Z. i. Muñoz, J. Suárez-Varela and P. Barlet-Ros, "Detecting cryptocurrency miners with NetFlow/IPFIX network measurements," 2019 IEEE International Symposium on Measurements & Networking (M&N), Catania, Italy, 2019, pp. 1-6.

5. D. S. Terzi, R. Terzi and S. Sagiroglu, "Big data analytics for network anomaly detection from netflow data," 2017 International Conference on Computer Science and Engineering (UBMK), Antalya, Turkey, 2017, pp. 592-597.

6. R. Wang, Z. Jia and L. Ju, "An Entropy-Based Distributed DDoS Detection Mechanism in Software-Defined Networking," 2015 IEEE Trustcom/BigDataSE/ISPA, Helsinki, Finland, 2015, pp. 310-317.

7. T. A. Tang, L. Mhamdi, D. McLernon, S. A. R. Zaidi and M. Ghogho, "Deep learning approach for Network Intrusion Detection in Software Defined Networking," 2016 International Conference on Wireless Networks and Mobile Communications (WINCOM), Fez, Morocco, 2016, pp. 258-263.

8. Z. M. Fadlullah et al., "State-of-the-Art Deep Learning: Evolving Machine Intelligence Toward Tomorrow’s Intelligent Network Traffic Control Systems," in IEEE Communications Surveys & Tutorials, vol. 19, no. 4, pp. 2432-2455.

9. C. Yin, Y. Zhu, J. Fei and X. He, "A Deep Learning Approach for Intrusion Detection Using Recurrent Neural Networks," in IEEE Access, vol. 5, pp. 21954-21961.

10. M. E. Ahmed, H. Kim and M. Park, "Mitigating DNS query-based DDoS attacks with machine learning on software-defined networking," MILCOM 2017 - 2017 IEEE Military Communications Conference (MILCOM), Baltimore, MD, USA, 2017, pp. 11-16.

11. J. Hou, P. Fu, Z. Cao and A. Xu, "Machine Learning Based DDos Detection Through NetFlow Analysis," MILCOM 2018 - 2018 IEEE Military Communications Conference (MILCOM), Los Angeles, CA, USA, 2018, pp. 1-6.