**Міністерство освіти і науки України**
**Тернопільський національний технічний університет імені Івана Пулюя**

*Факультет комп'ютерно-інформаційних систем і програмної інженерії*
(повна назва факультету )
*Кафедра комп'ютерних систем та мереж*
(повна назва кафедри)

# КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

*бакалавр*
(назва освітнього ступеня)

на тему: *Комп'ютеризована система управління якістю обслуговування корпоративних мереж*

Виконав: студент *IV курсу,* групи *СІ-42*
спеціальності *123 Комп'ютерна інженерія*
(шифр і назва спеціальності)

| | | |
|---|---|---|
| | (підпис) | *Шуб'як Д.І.* (прізвище та ініціали) |
| Керівник | (підпис) | *Баран І.О.*. (прізвище та ініціали) |
| Нормоконтроль | (підпис) | *Тиш Є.В.* (прізвище та ініціали) |
| Завідувач кафедри | (підпис) | *Осухівська Г.М.* (прізвище та ініціали) |
| Рецензент | (підпис) | (прізвище та ініціали) |

Тернопіль -2025

Міністерство освіти і науки України
**Тернопільський національний технічний університет імені Івана Пулюя**

Факультет *комп'ютерно-інформаційних систем і програмної інженерії*
(повна назва факультету)

Кафедра *комп'ютерних систем та мереж*
(повна назва кафедри)

ЗАТВЕРДЖУЮ
Завідувач кафедри
_____ *Осухівська Г.М.*
(підпис) (прізвище та ініціали)
«__» _____ 2025 р.

# ЗАВДАННЯ
## НА КВАЛІФІКАЦІЙНУ РОБОТУ

на здобуття освітнього ступеня *Бакалавр*
(назва освітнього ступеня)

за спеціальністю *123 Комп'ютерна інженерія*
(шифр і назва спеціальності)

Студенту *Шуб'яку Денису Ігоровчиу*
(прізвище, ім'я, по батькові)

1. Тема роботи *Комп'ютеризована система управління якістю обслуговування корпоративних мереж*

Керівник роботи *Баран Ігор Олегович., к.т.н., доц.*
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом ректора від *« 27 » 01 2025 року № 4/7-53*

2. Термін подання студентом завершеної роботи *20.06. 2025 р.*

3. Вихідні дані до роботи *Технічне завдання*

4. Зміст роботи (перелік питань, які потрібно розробити)

*Вступ.*
*1. Огляд предметної області.*
*2. Проектна частина.*
*3. Практична частина.*
*4. Безпека життєдіяльності, основи охорони праці.*
*Висновки*

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, слайдів)

*1. Структурна схема мережі для демонстрації роботи*
*2. Структурна схема стенду*
*3. Топологія мережі на тестовому стенді*
*4. Результати виконання команди ping між клієнтами*

6. Консультанти розділів роботи

| Розділ | Прізвище, ініціали та посада консультанта | Підпис, дата | |
|---|---|---|---|
| | | завдання видав | завдання прийняв |
| *Безпека життєдіяльності, основи охорони праці* | | | |

7. Дата видачі завдання *27.01.2025 р.*

КАЛЕНДАРНИЙ ПЛАН

| № з/п | Назва етапів роботи | Термін виконання етапів роботи | Примітка |
|---|---|---|---|
| *1.* | *Ознайомлення з завданням до кваліфікаційної роботи* | *27.01 – 30.01* | *Виконано* |
| *2.* | *Розробка технічного завдання* | *31.01 – 10.02* | *Виконано* |
| *3.* | *Підбір джерел про управління якістю обслуговування* | | |
| | *корпоративних мереж* | *11.02 – 08.03* | *Виконано* |
| *4.* | *Опрацювання літературних джерел* | *09.03 – 26.03* | *Виконано* |
| *5.* | *Виконання дослідження щодо розробки системи* | | |
| | *управління якістю обслуговування корпоративних* | | |
| | *мереж* | *27.03 – 16.04* | *Виконано* |
| *6.* | *Написання програмного коду і створення стенду* | *17.04 – 28.04* | *Виконано* |
| *7.* | *Оформлення розділу «Огляд предметної області»* | *29.04 – 02.05* | *Виконано* |
| *8.* | *Оформлення розділу «Проектна частина»* | *03.05 – 14.05* | *Виконано* |
| *9.* | *Оформлення розділу «Практична частина»* | *15.05 – 25.05* | *Виконано* |
| *10.* | *Виконання завдання до підрозділу «Безпека* | | |
| | *життєдіяльності, основи охорони праці»* | *26.05 – 31.05* | *Виконано* |
| *11.* | *Оформлення кваліфікаційної роботи* | *01.06 – 11.06* | *Виконано* |
| *12.* | *Нормоконтроль* | *09.06 – 12.06* | *Виконано* |
| *13.* | *Перевірка на плагіат* | *10.06 – 14.06* | *Виконано* |
| *14.* | *Попередній захист кваліфікаційної роботи* | *14.06 – 20.06* | *Виконано* |
| *15.* | *Захист кваліфікаційної роботи* | *27.06* | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

Студент _____ *Шуб'як Д.І.*
(підпис) (прізвище та ініціали)

Керівник роботи _____ *Баран І.О.*
(підпис) (прізвище та ініціали)

# АНОТАЦІЯ

Шуб’як Д.І. Комп’ютеризована система управління якістю обслуговування корпоративних мереж: робота на здобуття кваліфікаційного ступеня бакалавра: спец. 123 — комп’ютерна інженерія. Тернопіль: Тернопільський національний технічний університет імені Івана Пулюя, 2025. — 51 с.

Ключові слова: віртуальна машина, машина стану, метрики якості, протокол, скрипт, управління якістю

Кваліфікаційна робота присвячена розробці спеціалізованої комп’ютеризованої системи для керування якістю обслуговування (QoS) корпоративних мереж.

Описано QoS-метрики та моделі. Проведено порівняльний аналіз існуючих методів керування QoS корпоративних мереж (ручне налагодження, SDN, автоматизовані системи). Описано їх позитивні та негативні сторони.

Розроблено структуру корпоративної мережі, виконано налагодження її обладнання. Наведено опис розробленого протоколу, котрий дає змогу автоматизувати налаштування QoS без застосування SDN та автоматизованих систем. Описано типи повідомлень протоколу (Hello, Update, Notification) та машину стану протоколу.

Для втілення заявлених функцій протоколу було побудовано алгоритм та написано код на мові Python. Створений скрипт було запущений на стенді із віртуальних машин,. За результатами виконаних тестів було доведено, що розроблений протокол успішно виконує заявлені функції та дає змогу автоматизувати налагодження QoS для повного маршруту руху трафіку.

# ANNOTATION

Shchubiak D. Computerized system for managing the quality of service of corporate networks: Bachelor's Graduation Thesis: speciality 123 — computer engineering. Ternopil: Ternopil Ivan Puluj National Technical University, 2025. — 51 p.

Keywords: virtual machine, state machine, quality metrics, protocol, script, quality management

The bachelor thesis deals with the development of a specialized computerized system for managing the quality of service (QoS) of corporate networks.

QoS metrics and models are described. A comparative analysis of existing methods of managing QoS of corporate networks (manual tuning, SDN, automated systems) is carried out. Their positive and negative sides are described.

The structure of the corporate network is developed, its equipment is tuned. A description of the developed protocol is given, which allows you to automate QoS settings without using SDN and automated systems. The types of protocol messages (Hello, Update, Notification) and the protocol state machine are described.

To implement the stated functions of the protocol, an algorithm was built and code was written in Python. The created script was run on a stand from virtual machines. According to the results of the tests performed, it was proven that the developed protocol successfully performs the stated functions and allows you to automate QoS tuning for the entire traffic route.

ЗМІСТ

# ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ СКОРОЧЕНЬ І ТЕРМІНІВ

ACL — Access Control List

API - Application Programming Interface

DiffServ - Differentiated services

DIP - Destination Internet Protocol

FSM - Finite-State Machine

GNU — GNU's Not UNIX

ICMP — Internet Control Message Protocol

ID - Identifier

IntServ — Integrated services

IOS - Internetwork Operating System

IP - Internet Protocol

MAC — Media Access Control

Netns — Network namespaces

QoS - Quality of Service

RFC — Request for Comments

RTT - Round Trip Time

SDN — Software-Defined Networking

SIP - Source Internet Protocol

TCP — Transmission Control Protocol

ToS - Type of Service

UDP — User Datagram Protocol

VLAN — Virtual Local Area Network

VoIP — Voice over IP

ВМ – віртуальна машина

ОС – операційна система

ПЗ – програмне забезпечення

# ВСТУП

Корпоративні мережі все частіше стають тим фактором, котрий стримує зростання обчислювальної інфраструктури установи. Причинами є обсяг тафіку, котрий постійно зростає, а також його видова багатоманітність.

В цьому випадку актуальною задачею є відшукання ймовірних засобів збільшення якості функціонування мережі завдяки використанню внутрішніх можливостей. Один із таких варіантів - оптимізувати керування трафіком, що здатне забезпечити ефективне функціонування систем різного роду. Негативні наслідки незадовільного обслуговування можуть призвести до фінансових проблем, неякісної комунікації та збільшення затримок у виробництві. Тому виникла потреба гарантування часу реакції, пропускної здатності мережі і інших параметрів. Така технологія була розроблена та отримала назву «Якість обслуговування» (QoS). Вона застосовує розподілення за категоріями і надання пріоритетів трафікам. Це, зі свого боку, дає змогу забезпечити для даних із більшим пріоритетом значно якісніший режим мережевого передавання, який не залежить від умов щодо пропускної здатності потоків тих застосунків, котрі є не такими важливими [1].

Використовуються декілька методів управління технологією QoS. У даній роботі розглядатимуться, зокрема, ручне і автоматизоване налаштування, а також SDN.

Мета роботи – створення нового методу управління якістю обслуговування корпоративних мереж в межах комп’ютеризованої системи.

Завдання, необхідні для досягнення даної мети:

– обґрунтувати актуальність теми роботи;

– дослідити наявні методи і моделі керування QoS мережі із виявленням їх недоліків;

– створити та описати новий метод керування QoS мережі;

– запропонувати можливі напрямки розвитку роботи надалі.

# РОЗДІЛ 1 ОГЛЯД ПРЕДМЕТНОЇ ОБЛАСТІ

## 1.1 Метрики якості обслуговування

Багато сучасних додатків вимагають високу якість мережі, яка є сукупністю показників, званих метриками мережі. До основних метрик якості мережі належать:

– втрати пакетів;

– затримки;

– джиттер.

Втрати пакетів свідчать, яка кількість з надісланих джерелом пакетів отримано адресатом [2]. Здебільшого, це залежить від застосунку як дає раду із втратами. Для веб-застосунків, котрі застосовують протокол TCP, пакет надсилається знову, а у випадку телефонної розмови (із застосуванням UDP) — пакет ігнорується.

Затримка є часом, котрий необхідний даним, щоб дістатися від джерела до одержувача [2]. Затримка створює незручність при веденні діалогу, призводить до перекриття розмов та виникнення відлуння [3]. Здебільшого для оцінки зайнятості каналу застосовують RTT, котра є часовим діапазоном між надсиланням пакета та завершенням його опрацювання приймачем (рис. 1.1). Найчастіше цю метрику моніторять із використанням протоколу ICMP.

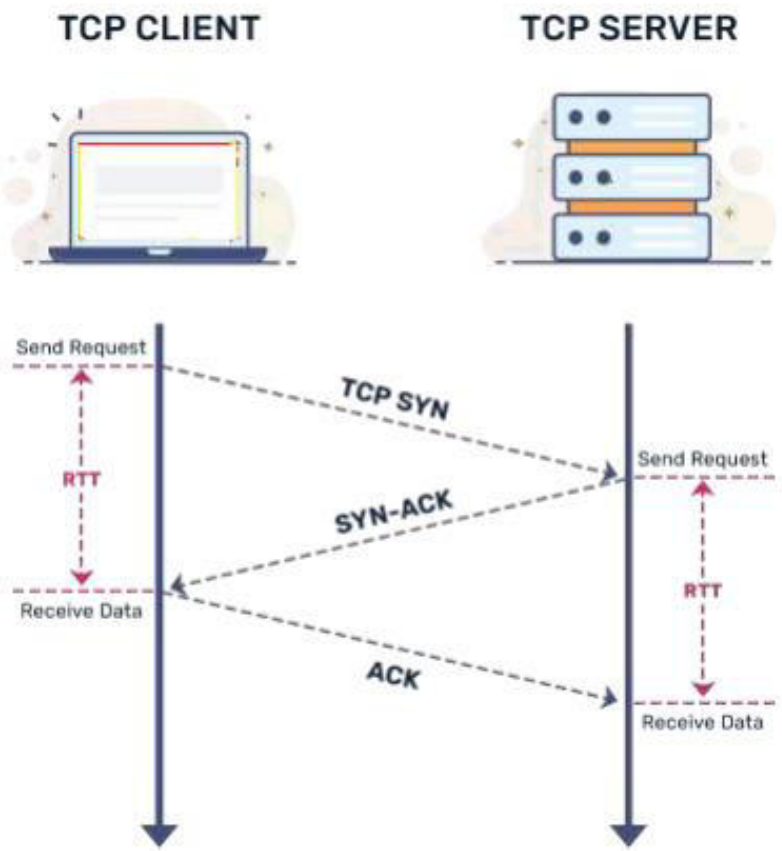


Рисунок 1.1 – RTT у мережі

Джитер є коливанням затримок у процесі передавання послідовних пакетів. Додатково до того сутність даної метрики перебуває у залежності від застосунку. Більшості застосунків достатньо, щоб пакет був просто доставлений і байдуже на наявність затримки. Проте для IP -телефонії це є важливим. Джитер веде до особливих порушень передавання мовлення, котрі відчуваються як тріщання та клацання [3].

## 1.2 Якість обслуговування (QoS)

Для покращення продуктивності мережі без, властиво, підвищення пропускної здатності самої транспортної мережі створили технологію QoS.

Застосовується три моделі гарантування QoS:

– Best Effort (не забезпечує якість передачі трафіку, властиво усі потоки є рівними);

– IntServ (забезпечує якість власне для кожного окремого потоку, попередньо запасає ресурси від того, хто надсилає, до власне отримувача);

– DiffServ (за встановлення якості відповідає будь-який вузол, через який проходить передача, резервування немає).

Здебільшого в даний час у мережах застосовується механізми DiffServ. Це пов’язано із тим, що модель IntServ потребує значних ресурсів від обчислювальних процесорів тих пристроїв, котрі видають дані, через те, що потрібне постійне попереднє резервування ресурсів.

DiffServ- модель ґарантує дотримання QoS на базі виразно визначених складових, котрі поєднуються для забезпечення виділення потрібних послуг. Служби DiffServ передбачають існування систематизовувачів і формування трафіку на границі мережі. На додачу повинна бути забезпечена підтримка функції розподілення ресурсів у ядрі мережі для ґарантування потрібної політики. Здійснюється поділ трафіку на класи, із застосуванням кількох рівнів власне QoS [4].

Така модель забезпечує, що трафік більшість часу одержуватиме адекватне обслуговування, проте ймовірно, із якимось ступенем градації. При перевантаженні потоки здатні до адаптовування свого трафіку згідно із наявними ресурсами і обслуговування триватиме, навіть за ймовірної нижчою якості. Плюс такого рішення – вища загальна ефективність, оскільки воно дає змогу передавати більше число потоків значно простіше, властиво із мінімальним підтриманням сигналізації і при допомозі простих способів структури шляхів передавання даних [4].

## 1.3 Методи втілення QoS

### 1.3.1 Налаштування в ручному режимі

Цей підхід полягає у поступовому одноманітному налаштуванні властиво адміністратором мережі в ручну правил обслуговування трафіку для кожного пристрою.

Переважно, мережеві адміністратори формують у електронних таблицях правила QoS на підставі аналізу вимог QoS клієнтів. Після цього на підставі цих правил формують типові конфігурації для обладнання мережі.

Такий метод є тривалим та неефективним, що зв'язано із помилками, котрі трапляються через людський фактор, котрий здатен призвести до деградації сервісів та зайвих грошових витрат. Проте він є умовно дешевим, оскільки не потребує закупівлі, застосування та функціонування нового коштовного обладнання як у випадку використання SDN, або ПЗ для автоматизованого налаштування обладнання мережі.

### 1.3.2 SDN

Основна ідея SDN – відокремлення функцій передавання трафіку від функцій керування (включно із контролем як самого трафіку, так і тих пристроїв, котрі виконують його передавання) [5].

На контролер покладаються функції аналізування пакетів, внесення змін до даних в пакетах і правил пересилання. Власне самі ж пристрої додержуються вказівок контролера, значно зменшується навантаження на обчислювальні ресурси. Як підсумок досягається централізація власне логіки керування мережею, а це дає змогу виконати конфігурування мережі як єдиного цілого та відчутно спрощує її експлуатацію.

Open- архітектура SDN ґарантує сумісність із декількома доставщиками. API -інтерфейси забезпечують можливість використання широкого спектру програм, зокрема хмарне оркестрування і необхідні для бізнесу мережеві програми. На додачу, інтелектуальне ПЗ здатне управляти обладнанням різних фірм при допомозі програмних інтерфейсів, наприклад OpenFlow. Врешті решт, з SDK інтелектуальні сервіси та різноманітні додатки властиво здатні функціонувати у загальному для них програмному середовищі [6].

Наявні рівні мереж SDN:

– застосунків;

– контролю;

– передавання.

Ресурси третього рівня здійснюють мережеві функції, зокрема передавання та опрацювання даних, та поведінка їх керується на другому рівні. Взаємодія контролера із складовими третього рівня ймовірна при допомозі інтерфейсів і спеціальних протоколів (Open Flow, як варіант), котрі ґарантують взаємодію із обладнанням мережі. З другої сторони контролер забезпечує стандартизовані API, наявність яких дозволяє створювати програми управління мережею [5]. Ці програми здатні, для прикладу, управляти пропускною здатністю, забезпечувати динамічне призначення пріоритетів різноманітних видів трафіку, перевіряти наявність доступу до ресурсів мережі.

На рис. 1.2 показана архітектура SDN.

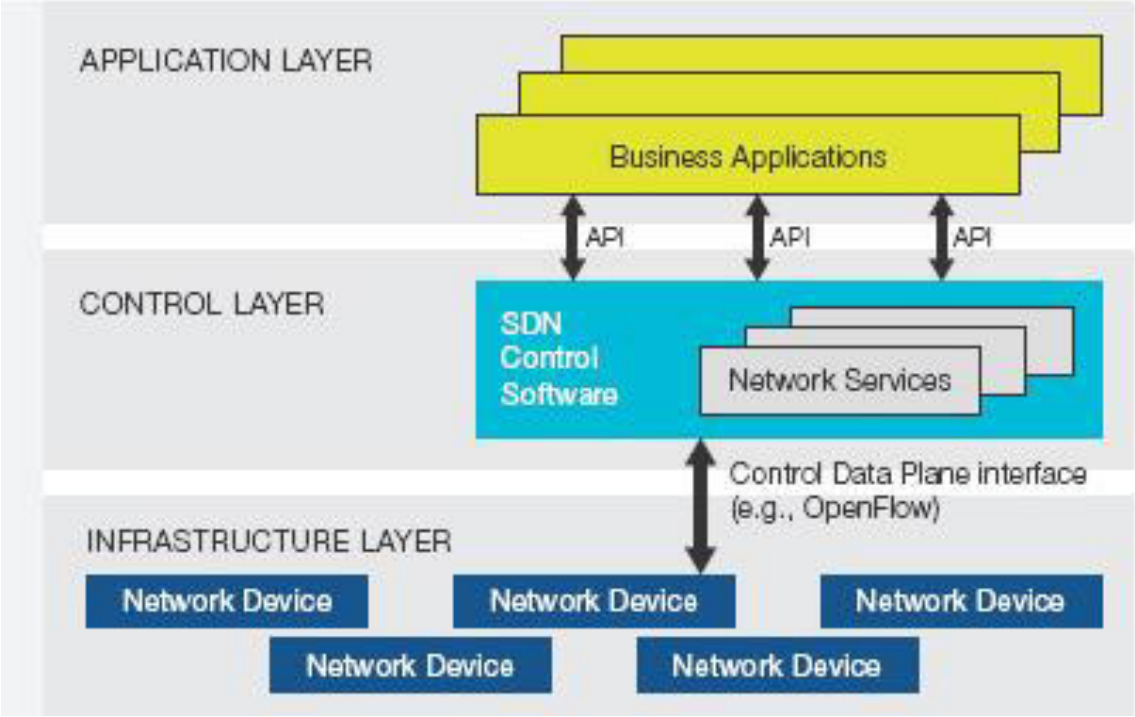


Рисунок 1.2 – Архітектура SDN

Основним недоліком концепції є значна вартість втілення. Обладнання, котре підтримує цю технологію і повністю управляється із центрального контролера, здатна купити собі не кожна фірма. Окрім того наразі наявні тільки кілька виробників згаданого устаткування.

### 1.3.3 Автоматизовані системи керування QoS

Є складними програмно-апаратними комплексами, котрі централізовано обраховують параметри топології мережі, опісля автоматично під'єднуються до обладнання мережі і здійснюють його налаштування.

Основні їх складові:

– управляюча система (ПК з інстальованим спеціальним ПЗ, котре налаштовує устаткування мережі в автоматичному режимі);

– база даних, котра містить мережну інформацію (топологія, політики і т.п.);

– кероване мережеве устаткування, на якому властиво потрібно налаштувати QoS.

Варто розглянути автоматизовані системи керування QoS на прикладі AutoQoS від Cisco Systems.

AutoQoS дає змогу виконати автоматизацію налагодження функції QoS, а фактично пріорітизації при доставленні мережних пакетів. Система ввела цю особливість в ОС IOS і Catalyst OS, а також спростила втілення IP QoS в мережі (як локальні, так і глобальні).

Саме це і забезпечило властиво автоматизацію налагодження інфраструктури IP при втіленні технології VoIP повністю по мережі — від розподільного обладнання до корпоративної IP –магістралі (рис. 1.3). На додачу це значно спростило налаштування послуг, котрі власне і забезпечує сервіс-провайдер [8].

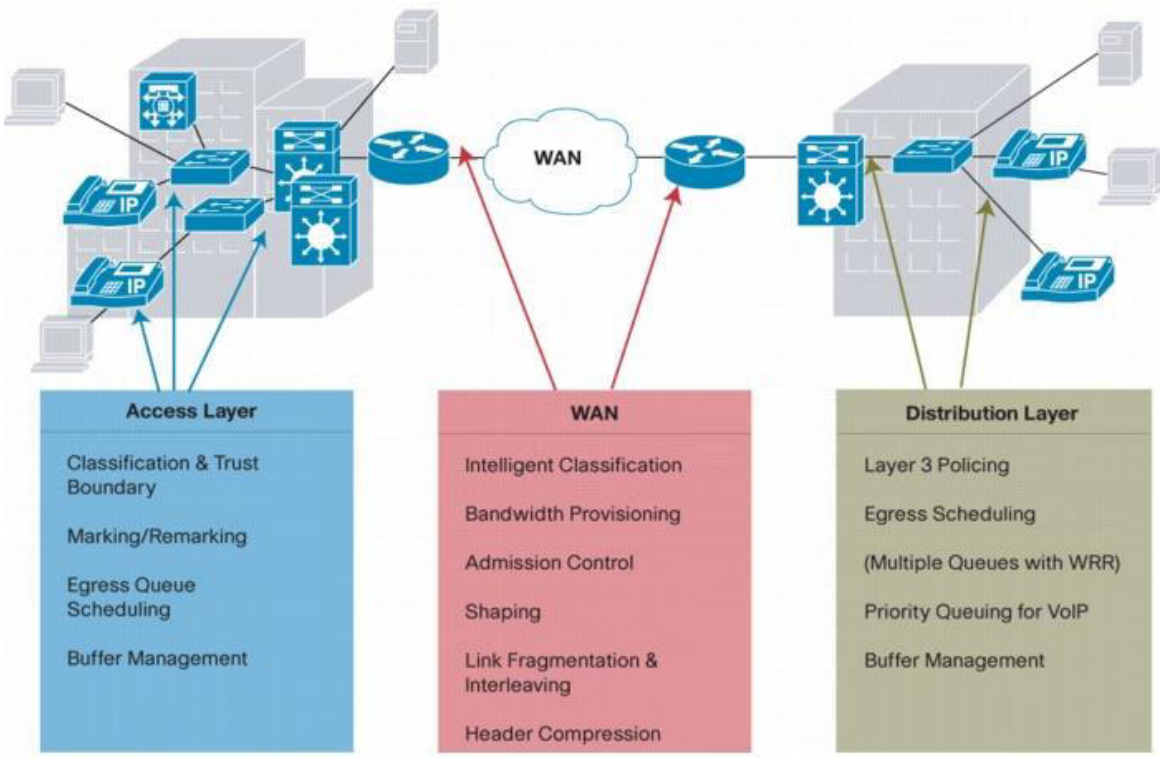


Рисунок 1.3 – Приклад розгортання QoS для VoIP від AutoQoS

Необхідно відмітити переваги таких систем:

– забезпечення автоматизації і оптимізації власне процесу налаштування конфігурації обчислювальної інфраструктури установи;

– втілення автоматизованих процесів класифікування трафіку, формування політик його опрацювання, налагодження конфігурації тощо;

– поліпшення надійності мережі через зниження (а у окремих випадках власне навіть виключення) помилок оператора та конфігурації;

– втілення централізованого керування для відстеження трафіку і одержання деталізованої інформації щодо стану операцій у мережі;

– здатність економічно керувати власне функціями QoS у великих IP-мережах, через поглиблений аналіз, інтелектуальний дизайн QoS і масштабовану реалізацію.

Мінусами таких систем є відсутність можливості переналагодження QoS за зміни топології мережі і великі фінансові видатки для організації при їх покупці, втіленні та функціонуванні.

## 1.4 Висновки до першого розділу

Провівши порівняльний аналіз та дослідження наявних методів QoS-керування обслуговування корпоративних мереж вдалося отримати наступні висновки:

– ручне налагодження QoS є дуже трудомістким та мало ефективним методом, котрий пов'язаний із помилками, які трапляються завдяки людському фактору, котрий здатен до зниження рівня якості служб і фінансових затрат, на додачу;

– мінусом SDN є потреба у заміні наявного парку мережного устаткування на нове при допомозі SDN, що веде до вагомих фінансових вкладень організації. На додачу якщо вийде з ладу контролер мережі, тоді вона перестає повністю функціонувати, цей факт заставляє резервувати контролер, що також веде до властиво додаткових фінансових затрат;

– мінусами автоматизованих систем керування QoS є відсутність можливості переналаштування QoS, якщо змінюється топологія мережі і , знову ж таки, необхідність вкладання додаткових коштів для компанії за умови їх купівлі та експлуатації.

За результатом огляду можна зробити висновок, що на сьогоднішній день в мережах застосовуються всі три моделі QoS - Best Effort Service, Integrated Service і DiffServ. Оскільки в Best Effort задіяні всі вільні на даний час ресурси мережі без спеціального конфігурування мережного обладнання, то сенс моделі полягає в своєчасному розширенні пропускної спроможності каналів зв'язку. Для Integrated Service характерна необхідність планування та прогнозування потреб ресурсів, що часто проблематично, або взагалі неможливо. У більшості мереж використовується DiffServ, тому в роботі буде розглядатися тільки модель DiffServ.

У розглянутих методах управління QoS корпоративних мереж (ручному, автоматизованих системах, SDN) виявлено та описано вище значні недоліки. Тому існують такі ситуації, коли застосування жодного з цих методів є

недоцільним. Наприклад, необхідно автоматизувати управління QoS за недостатнього фінансування або заміна мережного обладнання неможлива через недоступність (віддаленість розташування, відсутність допуску на вузли зв'язку). Тому необхідно запропонувати новий метод.

Подальша робота присвячена дослідженню створення нового методу управління QoS корпоративних мереж. Буде розроблено новий алгоритм управління якістю обслуговування. Також планується створити ПЗ, яке цей підхід дозволить реалізувати та протестувати на життєздатність.

В результаті застосування методу будуть вирішені проблеми, що виникають при використанні існуючих методів, такі як трудомісткість та помилки конфігурації через людський фактор, значні фінансові витрати по впровадженню SDN та автоматизованих систем, неможливість переналаштування QoS при зміні топології мережі у автоматизованих систем. Ці проблеми будуть вирішені за рахунок використання створеного в ході роботи ПЗ, функціонал якого дозволить автоматизувати та оптимізувати процес налаштування мережного обладнання.

# РОЗДІЛ 2 ПРОЕКТНА ЧАСТИНА

## 2.1 Розробка структури корпоративної мережі

На прикладі мережі, схема якої представлена на рис. 2.1, опишемо механізм роботи нового методу управління QoS.

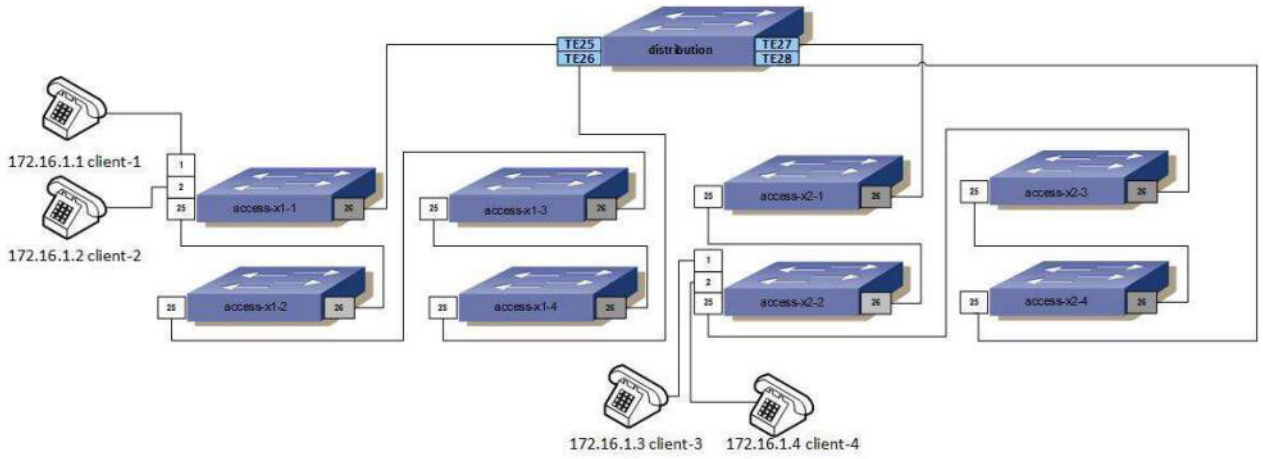


Рисунок 2.1 – Структурна схема мережі для демонстрації роботи методу

Перед адміністратором мережі стоїть завдання надати певний клас QoS для Ip-телефонів, позначених на схемі, як client-1, client-2, client-3, client-4.

Для виконання поставленого завдання необхідно пройти такі етапи:

– забезпечити механізм налаштування комутатора доступу. На даному етапі необхідно визначити, які саме налаштування потрібно вказати на кінцевому мережевому обладнанні та здійснити його конфігурування;

– комутатор доступу повинен здійснити пошук сусідніх комутаторів та спільно з ними побудувати топологію мережі;

– комутатори повинні здійснити побудову найкоротшого маршруту між Ip-телефонами у різних кінцях мережі;

– комутатори обчислюють та автоматично виставляють політику обслуговування до транзитних портів, якими буде передаватися трафік, а також до портів доступу для Ip-телефонів.

## 2.2 Налаштування кінцевого мережевого обладнання

Розглянемо приклад налаштування QoS у IOS Cisco на комутаторах серії Catalyst 6500/6000.

Налаштування політики в IOS включає такі кроки:

– визначення диспетчерів політик – нормальна швидкість трафіку, максимальна швидкість (якщо використовується), блокування та дія політики;

– створення QoS ACL для вибору контрольованого трафіку та приєднання диспетчера політик до цього ACL;

– застосування QoS ACL для всіх потрібних портів або VLAN [9].

У нашому випадку Ip -телефон client- 1 під'єднаний до комутатора access-xl-1 у порт ge1 і надсилає трафік UDP зі швидкістю 1 Мбіт/сек. Необхідно обмежити цей потік трафіку UDP до 50 кбіт/сек та відкинути зайвий трафік.

Налаштуємо функції застосування політик на кінцевому комутаторі access-xl-1. Менеджер політик приєднується до VLAN. Порт ge1 належить VLAN 10.

Потрібно налаштувати QoS на порту gel у режимі vlan-based. Використовуємо команду set port qos ge1 vlan-based для нашого завдання.

Конфігурація кінцевого комутатора access-xl-1 показана на рис. 2.2.

```
set qos enable #Увімкнення QoS [9]

set qos policer aggregate udp_50kbps rate 50 burst 10
drop #Визначення диспетчера політик [9]

set qos acl ip udp_qos_vid10 dscp 0 aggregate udp_50kbps
udp any any eq 5060 #Створення QoS ACL для вибору трафіку
та приєднання диспетчера політик до QoS ACL [9]

commit qos acl all #Компіляція цього QoS ACL [9]

set port qos ge1 vlan-based #Налаштування порту для режиму
QoS на основі віртуальної локальної мережі [9]

set qos acl map udp_qos_vid10 10 #Зіставлення QoS ACL
з віртуальною локальною мережею 10 [9]
```

Рисунок 2.2 – Конфігурація кінцевого комутатора access-xl-1

## 2.3 Опис роботи протоколу

Після включення протоколу комутатори роблять розсилку hello - повідомлень на групову адресу 01:80: C2:00:00:60 і очікують у відповідь такі ж hello - повідомлення , комутатори встановлюють сусідські відносини, додаючи кожен одне одного у свою локальну таблицю сусідів.

Для вставки заголовка, який містить в собі метрику і поле ToS, були виділені 4 байти між тегом 802.1Q і PayLoad. Перші два байти займає EtherType, котрий дає змогу відрізнити Ethernet-кадри проектованого протоколу від решти. Для перевірки працездатності алгоритму і моделювання був обраний EtherType 8999, оскільки він є вільним. Наступні два байти містять у собі безпосередньо метрику, яка впливає на те, як власне і будуть побудовані маршрути, а також поле ToS, яке характеризує конкретний клас сервісу у даному DiffServ домені. Таким чином вдається зберегти і використовувати всі вхідні EtherType та

заголовки, які були вставлені раніше.

Підсумковий вид кадру представлений на рис. 2.3.

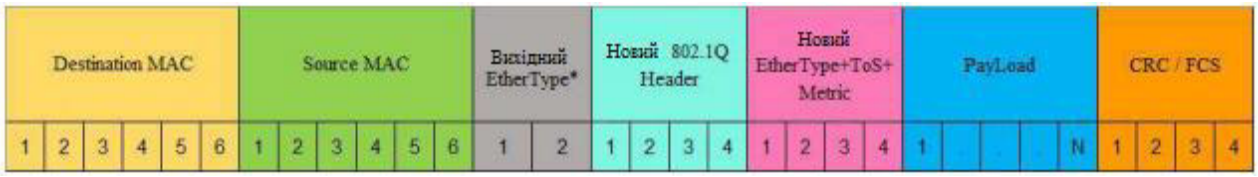


Рисунок 2.3 – Формат Ethernet кадру для роботи протоколу

Для розробки алгоритму було введено ряд термінів, що описує деякі нові функції та стани комутаторів та його інтерфейсів, зокрема:

– Edge інтерфейси, до них під’єднуються кінцеві пристрої, які не входять до протоколу;

– Core інтерфейс, вони під’єднуються до інших комутаторів із мережі протоколу;

– Metric - числове значення в таблиці MAC -адрес, що впливає на вибір маршруту для трафіку, що відправляється;

– заголовок кадру - набір байтів, які містять будь-яку інформацію. У даній ситуації це інформація про вартість маршруту до пристрою, з якого прийшов кадр [10].

У ході роботи протокол аналізує кадри та зберігає найкоротші маршрути між мережевими пристроями. Для цього створено таблицю, схожу на стандартну таблицю MAC -адрес, але вона має додаткове поле Metrics, а також можливість зберігати для однієї адреси кілька інтерфейсів. Зберігання кількох інтерфейсів обумовлено можливістю балансувати трафік, якщо дозволяє топологія.

Приклад таблиці представлений у табл. 2.1.

Таблиця 2.1 – Приклад таблиці проектованого протоколу

| MAC | Interface | Age | Metrics |
|---|---|---|---|
| A | e0/0 | 20 | 10 |
| B | e0/1 | 20 | 40 |
| C | e0/2 | 20 | 20 |
| .... | .... | .... | .... |

Після обміну повідомленнями кожен комутатор знає про всі з'єднання, на підставі пар будується повна топологія мережі, що містить усі комутатори і всі зв'язки між ними.

Потім спираючись на інформацію про повну топологію мережі, комутатори вибудовують найкоротший маршрут між мережевим обладнанням, що знаходиться в різних кінцях мережі.

Комутатори обмінюються інформацією з таблиць, що містить зв'язки MAC-адрес та відповідних їм значень Metrics. Знаючи за яким інтерфейсом знаходиться мережеве обладнання з даною MAC -адресою, комутатори надсилають дані через потрібні інтерфейси сусідам. Саме на цих інтерфейсах алгоритм налаштовує необхідну QoS. Спочатку налаштування QoS задає адміністратор на кінцевому комутаторі доступу, до якого безпосередньо під’єднаний клієнт, який потребує певної QoS. Після цього комутатор конвертує параметри QoS в бінарний код і поміщає його в полі QoS повідомлення Update. Сформоване повідомлення Update надсилається сусіднім комутаторам. Сусіди беруть повідомлення Update. Вилучають бінарний код з поля QoS і конвертують його в зрозумілу для цього комутатора конфігурацію. Замінивши інтерфейси на потрібні (інтерфейс, з якого отримано повідомлення Update, і інтерфейс, за яким знаходиться наступний сусід), протокол застосовує налаштування QoS на даному комутаторі. Потім формує нове повідомлення Update з бінарним кодом налаштувань у полі QoS і передає своїм сусідам далі за маршрутом трафіку.

Якщо в одного з комутаторів зникає зв'язок із сусідом, тоді:

– він розсилає всім нові повідомлення;

– всі комутатори наново будують топологію мережі;

– наново обраховують найкоротші маршрути між мережевим обладнанням;

– оновлюють таблицю даних всіх станів каналів домену.

## 2.4 Типи повідомлень протоколу

### 2.4.1 Hello

Відправляються регулярно через фіксовані інтервали часу (за замовчуванням, 10 секунд) на групову адресу 01:80:C2:00:00:60, яка входить до діапазону вільних групових MAC- адрес 01:80:C2:00:00:50 — 01:80:C2:00:00

Повідомлення Hello виконують такі завдання:

– пошук сусідів та встановлення сусідських відносин;

– здійснюють функцію keepalive для підтримки встановлених сусідських відносин.

Формат повідомлення Hello показано на рис. 2.4.

| EtherType = 8999 | | Type | ID Комутатора | | | | | | Hello interval | | Dead interval | | ID сусідного комутатора | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 2 | 1 | 1 | 2 | 3 | 4 | 5 | 6 | 1 | 2 | 1 | 2 | 1 | 2 | 3 | 4 | 5 | 6 |

Рисунок 2.4 – Формат повідомлення Hello

Повідомлення Hello містить такі поля:

– Type - тип повідомлення (для повідомлення Hello значення рівне 1);

– ID Комутатора – ідентифікує комутатор у межах мережі. У ролі ідентифікатора використовується MAC -адреса комутатора;

– Hello interval - задає часовий період розсилки hello - повідомлень (за

замовчуванням, 10 секунд);

– Dead interval – час очікування відповіді сусідів;

– ID сусіднього комутатора – ідентифікатор сусіднього комутатора.

Список сусідів формується з ідентифікаторів сусідів, яких комутатор отримав повідомлення Hello протягом часу, заданого у полі Dead interval [11].

### 2.4.2 Update

Повідомлення Update використовується для обміну конфігурацією, яка керує QoS, між комутаторами із сусідськими відносинами. Дана конфігурація міститься у двійковому форматі у полі QoS. Значення поля Type дорівнює 2.

На рис. 2.5 наведено формат повідомлення Update.

| EtherType = 8999 | | Type | ID Комутатора | | | | | | MAC-адреса | | | | | | QoS |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 2 | 1 | 1 | 2 | 3 | 4 | 5 | 6 | 1 | 2 | 3 | 4 | 5 | 6 | 300 |

Рисунок 2.5 – Формат повідомлення Update

На комутаторі доступу, до якого під’єднано кінцевого клієнта, адміністратор виконує налаштування, що керує QoS для даного клієнта. Необхідну частину створеної конфігурації комутатор, використовуючи серіалізацію, кодує в двійковому форматі, розміщує отриманий код у полі QoS повідомлення Update і передає сусідньому комутатору, який і приймає повідомлення Update. Вилучає двійковий код з поля QoS і десеріалізує його, приводячи до текстового вигляду. Потім комутатор застосовує отриману конфігурацію до висхідного порту, з якого отримано повідомлення Update. За таблицею сусідів комутатор визначає низхідний порт і також застосовує цю конфігурацію. Таким чином, комутатор змінює номери портів в отриманій від сусіднього комутатора конфігурації і виконує автоматизоване налаштування транзитних Core портів. Змінену конфігурацію аналогічним способом комутатор серіалізує в двійковий код, поміщає його в полі QoS повідомлення Update і

передає наступному сусідові за маршрутом.

Такий алгоритм повторюється доти, доки повідомлення Update не досягає комутатора доступу, до якого під'єднаний другий кінцевий клієнт. Цей комутатор доступу також десеріалізує двійковий код із повідомлення Update у текстовий вигляд, застосовує отриману конфігурацію на висхідний порт. Потім по таблиці MAC –адрес визначає порт, до якого під'єднано адресата повідомлення, та застосовує на нього необхідну конфігурацію.

В результаті, всі порти на маршруті передачі між кінцевими клієнтами налаштовані на певний рівень QoS за допомогою автоматизованого налаштування.

### 2.4.3 Notification

Це інформаційне повідомлення. Використовуються, коли виникають помилки та/або змінюється топологія мережі. Відправляються сусіднім комутаторам. Значення поля Type дорівнює 3.

На рис. 2.6 представлено формат повідомлення Notification.

| EtherType = 8999 | | Type | ID Комутатора | | | | | | Error code |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 2 | 1 | 1 | 2 | 3 | 4 | 5 | 6 | 1 |

Рисунок 2.6 – формат повідомлення Notification

Коди помилок:

1. - Помилка обробки повідомлення.
2. - Помилка у сусідських відносинах.
3. - Помилка обміну інформацією про стан каналів, синхронізацію баз даних.
4. - Зміна топології мережі.

2.5 Машина стану протоколу

Комутатор з протоколом прийняття рішень в діях з іншими комутаторами використовує кінцевий автомат (finite-state machine (FSM)), який включає чотири стани: Idle, HelloSent, HelloConfirm, Established (рис. 2.7). Для кожного сеансу протокол підтримує змінну стану, яка відстежує, у якому з цих чотирьох станів перебуває сеанс. Протокол визначає повідомлення, якими кожен комутатор повинен обмінюватися, щоб переключити сеанс із одного стану до іншого.

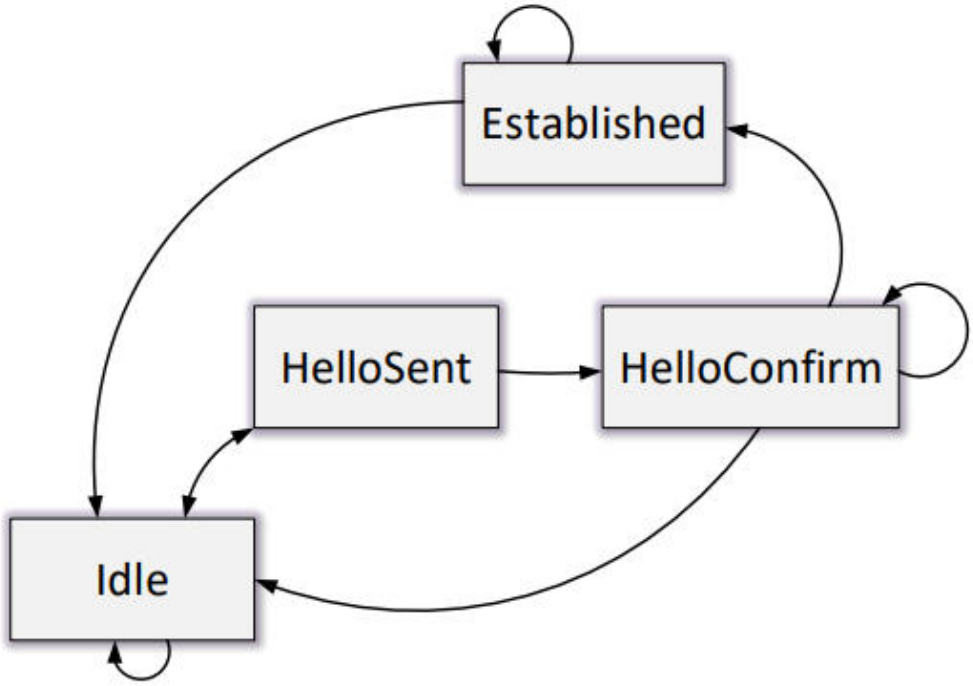


Рисунок 2.7 – Машина стану протоколу

Перший стан (Idle) – це стан очікування. У стані очікування протокол ініціалізує всі ресурси, відхиляє всі спроби з'єднання. Другий стан – HelloSent. У стані HelloSent комутатор відправляє повідомлення Hello по всіх Core інтерфейсів і очікує таке ж у відповідь, щоб перейти в стан HelloConfirm. Обмін повідомленнями перевірки активності виконується, і після успішного отримання комутатор перетворюється на стан Established. Це означає, що запущено правильну версію протоколу і всі налаштування правильні. Комутатори-сусіди переходять до обміну інформацією зі своїх баз даних, що містять бінарні

значення, що визначають якість обслуговування для мережного обладнання з конкретною MAC- адресою. Для цього використовуються повідомлення Update. У стані Established комутатор може надсилати та отримувати: Hello, Update, Notification повідомлення до свого сусіда та від нього.

## 2.6 Висновки до другого розділу

За результатами аналізу предметної галузі та пропозицій на ринку мережевого обладнання та ПЗ, було прийнято рішення розробити таке ПЗ, котре реалізує можливості та функції досліджуваного протоколу для управління QoS корпоративних мереж. Цей протокол дозволить автоматизувати та оптимізувати налаштування якості обслуговування для конкретних клієнтів. Наступним етапом у розробці протоколу буде його реалізація та тестування на спеціально підготовленому стенді, який має підтвердити заявлені функції.

.

# РОЗДІЛ 3 ПРАКТИЧНА ЧАСТИНА

## 3.1 Python

Python є універсальною сучасною мовою програмування високого рівня, перевагам якої є висока продуктивність програмних втілень та чітка структура коду, котрий добре читається (рис. 3.1). Синтаксис Python надзвичайно полегшений, це, зів свого боку, дає змогу оволодіти ним за відносно незначний проміжок часу. Його ядро володіє напрочуд зручною структурою, а значна кількість вбудованих бібліотек дає змогу використовувати широкий перелік вигідних функцій та можливостей. Властиво Python може застосовуватися для створення прикладних скриптів, програм, та й розроблення WEB-сервісів.


Рисунок 3.1 – Логотип мови Python

Мова здатна забезпечити підтримку значного списку стилів створення застосунків різного призначення та роду, зокрема є дуже зручною для роботи з ООП та функціонального програмування.

CPython є один із найбільш популярних інтерпретаторів мови, властиво написаний на Сі. Розповсюджується це середовище безкоштовно із Free-ліцензією.

Саме завдяки простоті написання та налагодження коду для реалізації протоколу вибрано мову Python.

## 3.2 Структура коду протоколу

При написанні протоколу за основу було взято блоки коду вивчення кадру та управління пересиланням кадрів із протоколу, розробленого в [10] Для додавання заявленого функціоналу було створено блоки управління службовими повідомленнями та обробки конфігурації QoS.

Таким чином, код реалізованого протоколу можна розділити на чотири частини:

- блок вивчення кадру;
- блок керування службовими повідомленнями;
- блок управління пересиланням кадрів;
- блок обробки конфігурації QoS.

Блок вивчення кадру можна поділити на такі складові:

- блок вивчення метрик та заповнення таблиць;
- блок балансування трафіку.

Основне завдання блоку вивчення кадрів є заповнення таблиць маршрутизації. Кожен кадр, який проходить через комутатор, вивчається та інформація поміщається у спеціальну таблицю, що містить MAC-адресу джерела, інтерфейс, з якого прийшов кадр, «вік» запису (за замовчуванням 30 секунд) та метрику. Записи з однаковою MAC-адресою джерела потрапляють у таблицю із заміною, яка відпрацьовує за алгоритмом «менше метрика – кращий маршрут». Якщо в таких записів метрика однакова, але відрізняються вхідні інтерфейси, то в таблицю додаються обидва записи. Це дозволяє надалі використовувати балансування трафіку.

Блок балансування трафіку необхідний для поділу трафіку доступними каналами передачі даних. Головною особливістю є балансування трафіку за рівноцінними маршрутами. Ця функція дозволяє розподілити навантаження

рівномірно по всіх доступних каналах зв'язку в мережі.

Балансування може здійснюватися двома способами:

– за пакетами;

– за потоками.

Балансування за пакетами працює за принципом «один канал – один пакет». Даний підхід є неоптимальним, тому що може виникнути втрата пакетів, що негативно позначиться на роботі мережі.

Балансування каналами використовує алгоритм «один канал – один потік». Під потоком розуміється сукупність параметрів, які можуть однозначно визначити пакети, що генеруються тими чи іншими програмами, протоколами, сервісами тощо.

Вибір каналу для потоку здійснюється за допомоги хеш-функції за формулою (3.1):

$$X = (SIP + DIP + Proto + SPort + DPort) \% N, \quad (3.1)$$

де X – номер отриманого каналу;

SIP – IP адреса джерела;

DIP – IP адреса призначення;

Proto – ідентифікатор протоколу;

SPort - TCP\UDP порт джерела;

DPort - TCP\UDP порт призначення;

N – кількість доступних каналів.

Перед відправкою пакета алгоритм обраховує хеш-функцію з полів заголовків пакета. Хеш-сума, отримана шляхом сумування полів заголовків кадру, ділиться на кількість доступних інтерфейсів, за якими знаходиться пристрій із MAC -адресою призначення. За отриманим числом визначається канал і до нього відправляється пакет. Таким чином вдається добитися розподілу потоку трафіку на всі доступні канали.

Після вивчення кадру та отримання номера інтерфейсу спрацьовує блок

пересилання кадру. Якщо кадр був отриманий від пристроїв, котрі не входять до мережі комутаторів, де власне і працює реалізований протокол, то на інтерфейсі, звідки прийшов даний кадр у таблиці ставиться метрика 0, а сам кадр вставляється EtherType 8999. Даний EtherType необхідний для спрощення пошуку метрики в кадрі і для можливості відрізнити пакет. Якщо пакет прийшов з іншого комутатора мережі протоколу, то відбувається перевірка запису в таблиці на можливість зміни і збільшивши метрику на задану в налаштуваннях одиницю. При виході пакета з мережі протоколу EtherType і метрика з кадру знімаються [10].

Завдяки блоку управління службовими повідомленнями здійснюється пошук та встановлення сусідських відносин із мережевим обладнанням за допомогою повідомлень Hello, передача налаштувань якості обслуговування (повідомлення Update), сповіщення сусідів про зміни та помилки в мережі (повідомлення Notification).

Блок обробки конфігурації QoS відповідає за прийом налаштувань з поля QoS у повідомленні Update, отриманому від сусіднього мережного обладнання або адміністратора мережі, у разі ручного налаштування. Даний блок інтерпретує отримані налаштування у зрозумілий для використовуваного обладнання лексикон, змінює найменування інтерфейсів для налаштування на них QoS політик. Потім передає ці налаштування далі формування нового повідомлення Update, який буде відправлений наступному сусідові з топології мережі.

## 3.3 Опис стенду для тестування

Для проведення тестів на перевірку функціоналу розробленого протоколу був зібраний віртуальний стенд, який являє собою тринадцять ВМ із встановленою ОС GNU/Linux Ubuntu 12.04.5. ВМ мають по два процесори Intel Xeon E5320 з тактовою частотою 1 Гб оперативної пам’яті кожен.

Детальна інформація про технічні характеристики ВМ наведена на рис. 3.2 та 3.3.

```
client-1@switch:~$ free
             total       used       free     shared    buffers     cached
Mem:       1025644     272764     752880          0      43896     162740
-/+ buffers/cache:      66128     959516
Swap:      1046524          0    1046524
```

Рисунок 3.2 – Інформація про оперативну пам'ять (вивід команди free)

```
client-1@switch:~$ cat /proc/cpuinfo
processor       : 0
vendor_id       : GenuineIntel
cpu family      : 6
model           : 15
model name      : Intel(R) Xeon(R) CPU           E5320  @ 1.86GHz
stepping        : 11
microcode       : 0xbc
cpu MHz         : 1861.914
cache size      : 4096 KB
physical id     : 0
siblings        : 1
core id         : 0
cpu cores       : 1
apicid          : 0
initial apicid  : 0
fdiv_bug        : no
f00f_bug        : no
coma_bug        : no
fpu             : yes
fpu_exception   : yes
cpuid level     : 10
wp              : yes
flags           : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr
aperfmperf pni ssse3 cx16 x2apic tsc_deadline_timer hypervisor lah
bogomips        : 3723.82
clflush size    : 64
cache_alignment : 64
address sizes   : 42 bits physical, 48 bits virtual
power management:

processor       : 1
vendor_id       : GenuineIntel
cpu family      : 6
model           : 15
model name      : Intel(R) Xeon(R) CPU           E5320  @ 1.86GHz
stepping        : 11
microcode       : 0xbc
cpu MHz         : 1861.914
cache size      : 4096 KB
physical id     : 2
siblings        : 1
core id         : 0
cpu cores       : 1
apicid          : 2
initial apicid  : 2
fdiv_bug        : no
f00f_bug        : no
coma_bug        : no
fpu             : yes
fpu_exception   : yes
cpuid level     : 10
wp              : yes
flags           : fpu vme de pse tsc msr pae mce cx8 apic sep mtr
aperfmperf pni ssse3 cx16 x2apic tsc_deadline_timer hypervisor la
bogomips        : 3723.82
clflush size    : 64
cache_alignment : 64
address sizes   : 42 bits physical, 48 bits virtual
power management:
```

Рисунок 3.3 – Інформація про процесор (вивід команди cat /proc/cpuinfo)

Дев'ять ВМ виконують функції комутаторів - на них запущено розроблений протокол. Чотири ВМ, що залишилися, є клієнтами. Вони працюють у спрощеному режимі і запускають застосунки, які діагностують мережу (ping, tcpdump). Всі ВМ з’єднані за допомогою віртуальних каналів передачі даних.

Структурна схема віртуального стенду для тестування представлена на рис. 3.4.

Зібраний віртуальний стенд значно спрощує запуск і налагодження протоколу, що тестується, порівняно зі стендом з фізичних комутаторів та ip - телефонів.

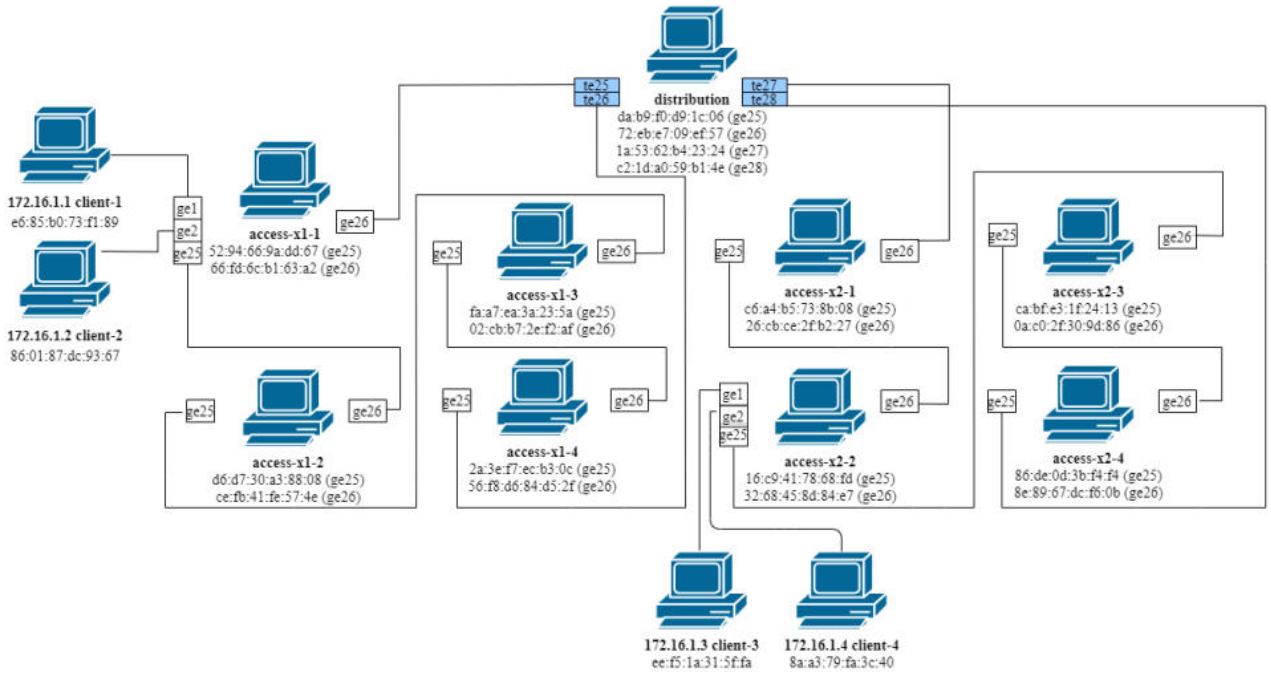


Рисунок 3.4 – Структурна схема стенду

Всі ВМ зі своїми інтерфейсами створені та налаштовані за допомогою скрипта setup.sh. Покажемо його роботу з прикладу комутатора access-x1-1. Для цього зробимо вивід скрипта setup.sh для комутатора access-x1-1 (рис. 3.5).

З наведеного видно, що скрипт setup.sh створює network namespaces з ім'ям access-x1-1. Network namespaces (netns) - це логічна копія мережевого стека зі своїми власними маршрутами, правилами брандмауера та мережевими пристроями [12]. Потім для access-x1-1 створюються та запускаються інтерфейси з іменами "ge26", "ge25", "ge1", "ge2". Для інтерфейсу "ge1" вказується сусідній інтерфейс "eth0", котрий належить netns під назвою client-1. А для інтерфейсу "ge2" – інтерфейс "eth0", який належить netns з ім'ям client-2.

Таким чином, створені та налаштовані ВМ, які виконують функції комутатора доступу access-x1-1 та під’єднаних до нього ip -телефонів client -1 та client-2.

```
user@switch:~$ cat setup.sh | grep access-x1-1
/sbin/ip netns add access-x1-1
/sbin/ip netns exec distribution /sbin/ip link set dev "ge26"
netns access-x1-1
/sbin/ip netns exec access-x1-1 /sbin/ip addr flush dev "ge26"
/sbin/ip netns exec access-x1-1 /sbin/ip link set dev "ge26"
up
/sbin/ip netns exec access-x1-2 /sbin/ip link set dev "ge25"
netns access-x1-1
/sbin/ip netns exec access-x1-1 /sbin/ip addr flush dev "ge25"
/sbin/ip netns exec access-x1-1 /sbin/ip link set dev "ge25"
up
/sbin/ip netns exec access-x1-1 /sbin/ip link add name "ge1"
type veth peer name "eth0"
/sbin/ip netns exec access-x1-1 /sbin/ip addr flush dev "ge1"
/sbin/ip netns exec access-x1-1 /sbin/ip link set dev "ge1"
up
/sbin/ip netns exec access-x1-1 /sbin/ip link set dev "eth0"
netns client-1
/sbin/ip netns exec access-x1-1 /sbin/ip link add name "ge2"
type veth peer name "eth0"
/sbin/ip netns exec access-x1-1 /sbin/ip addr flush dev "ge2"
/sbin/ip netns exec access-x1-1 /sbin/ip link set dev "ge2"
up
/sbin/ip netns exec access-x1-1 /sbin/ip link set dev "eth0"
netns client-2
```

Рисунок 3.5 – Вивід скрипта setup.sh для комутатора access-x1-1

Щоб запустити робочий протокол на визначеній ВМ, необхідно авторизуватися на стенді під її ім'ям. Для прикладу запустимо протокол на комутаторі доступу access-x1-1 (рис. 3.6).

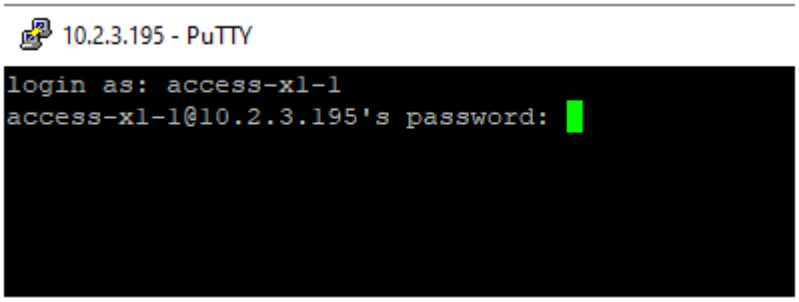


Рисунок 3.6 – Авторизація на комутаторі access-x1-1

Після авторизації бачимо вітальне повідомлення із системною інформацією ВМ (рис. 3.7).

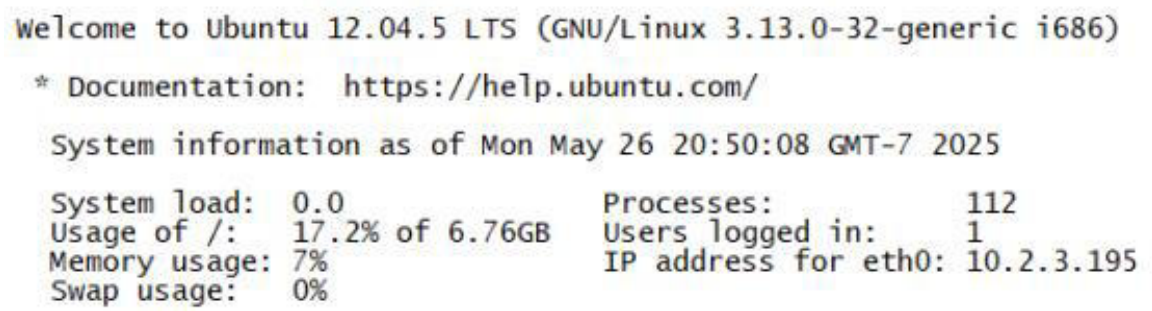


Рисунок 3.7 – Вітальне повідомлення ВМ access-x1-1

Після цього автоматично запускається скрипт access-x1-1.sh, який для netns з ім'ям access-x1-1 запускає скрипт switch.py із параметром access (рис. 3.8).

```
user@switch:~$ cat access-x1-1.sh
#!/bin/sh
/usr/bin/sudo /sbin/ip netns exec access-x1-1 /usr/bin/sudo
/usr/bin/python /home/user/switch.py access
```

Рисунок 3.8 – Виведення скрипта access-x1-1.sh

Параметр access видається лише тим ВМ, до яких безпосередньо під’єднані клієнти. У скрипті switch.py міститься вихідний код розробленого протоколу, написаний високорівневою мовою програмування Python версії 2.7. Після запуску цього скрипта починає роботу протокол.

Аналогічно відбувається запуск для інших ВМ із стенду для тестування.

## 3.4 Результати тестування розробленого протоколу

При тестуванні розробленого протоколу на відповідність заявленим функціям використовувалися утиліти ping і tcpdump.

Утиліта ping генерує ICMP пакети, які дозволяють дізнатися про доступність того чи іншого хоста у мережі.

Утиліта tcpdump - це аналізатор, який перехоплює мережевий трафік, що проходить через пристрій, на якому запущено утиліту.

На віртуальному стенді було проведено тестування розробленого протоколу, що показало наступні результати. На рис. 3.9 зображено топологію з працюючими на час тестування пристроями.

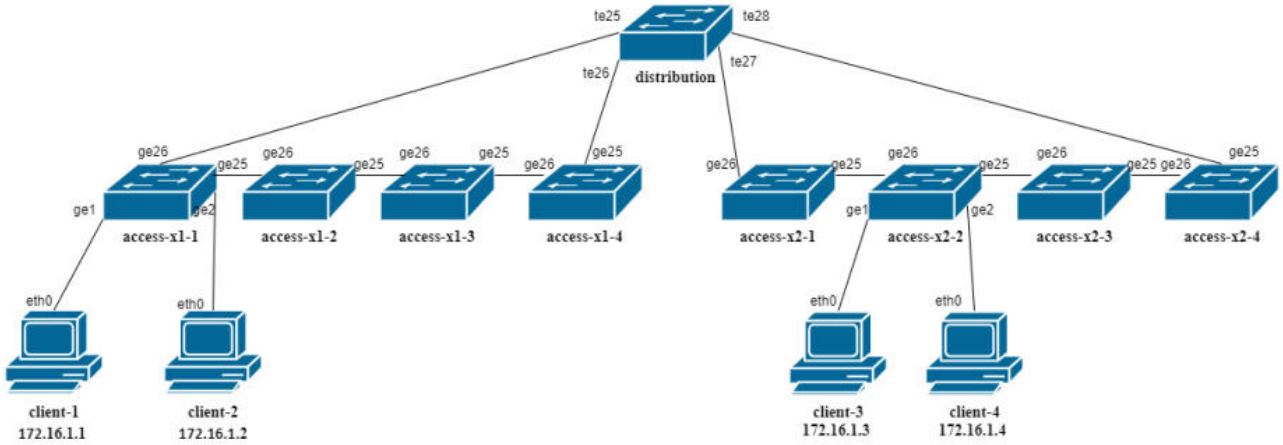


Рисунок 3.9 – Топологія мережі на тестовому стенді

При запуску утиліти ping із зазначенням кількості відправлених пакетів 1000 шт. з client -1 на client-4, з client-2 на client-3, з client-3 на client -1, з client-4 на client-2 одночасно були отримані такі результати (рис. 3.10):

– відповідь на ping була отримана від усіх кінцевих хостів;

– подвоєння та втрати пакетів не спостерігалися;

– хости-комутатори відпрацювали алгоритм протоколу без помилок та коректно заповнили таблиці маршрутизації.

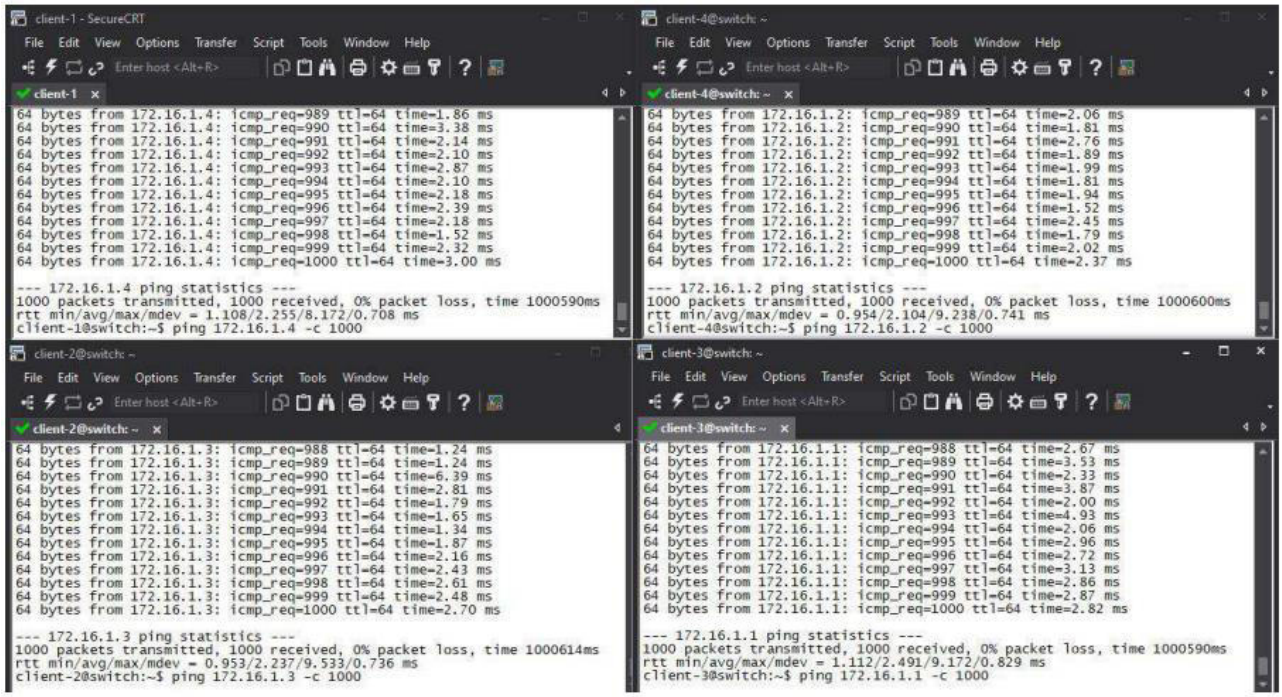


Рисунок 3.10 – Результати виконання команди ping між клієнтами

На рис. 3.11 представлена інформація з client-1 та всіх комутаторів при виконанні команди ping з client-1 на client-4.

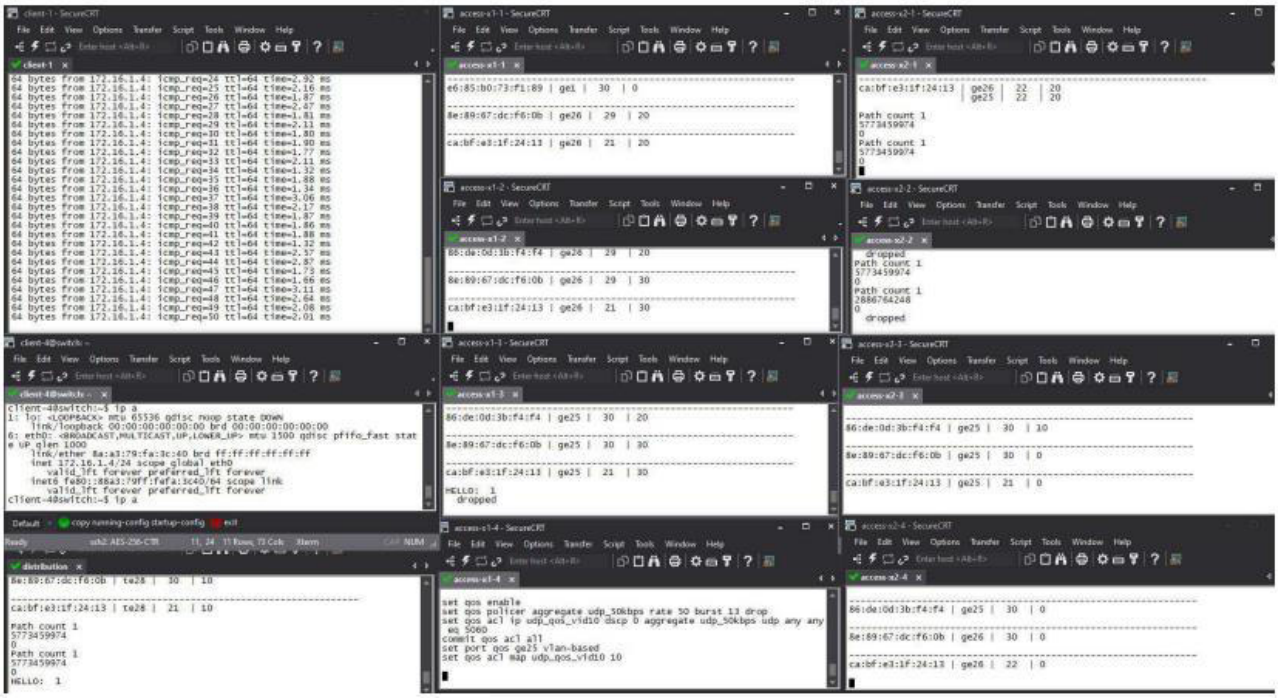

Рисунок 3.11 – Інформація з комутаторів при виконанні команди ping між client-1 та client-4

За допомогою інформації з рис. 3.10 можна простежити отриманий при роботі протоколу маршрут для клієнтів client -1 і client-4: client -1 ↔ access-x1-1 ↔ distribution ↔ access-x2-1 ↔ access-x2-2 ↔ client-4.

Отриманий результат дозволяє зробити висновок, що алгоритм протоколу працює правильно і вибудовує найоптимальніший маршрут для трафіку.

На рис. 3.12 зображені повідомлення Update з комутаторів access- x1-1 та access-x2-2, до яких безпосередньо під'єднані клієнти client-1 та client-4 відповідно. Видно, що розроблений протокол здійснює передачу налаштувань QoS, введених вручну адміністратором мережі, за маршрутом проходження трафіку між комутаторами access-x1-1 та access-x2-2. Отже, розроблений протокол виконує заявлену функцію.

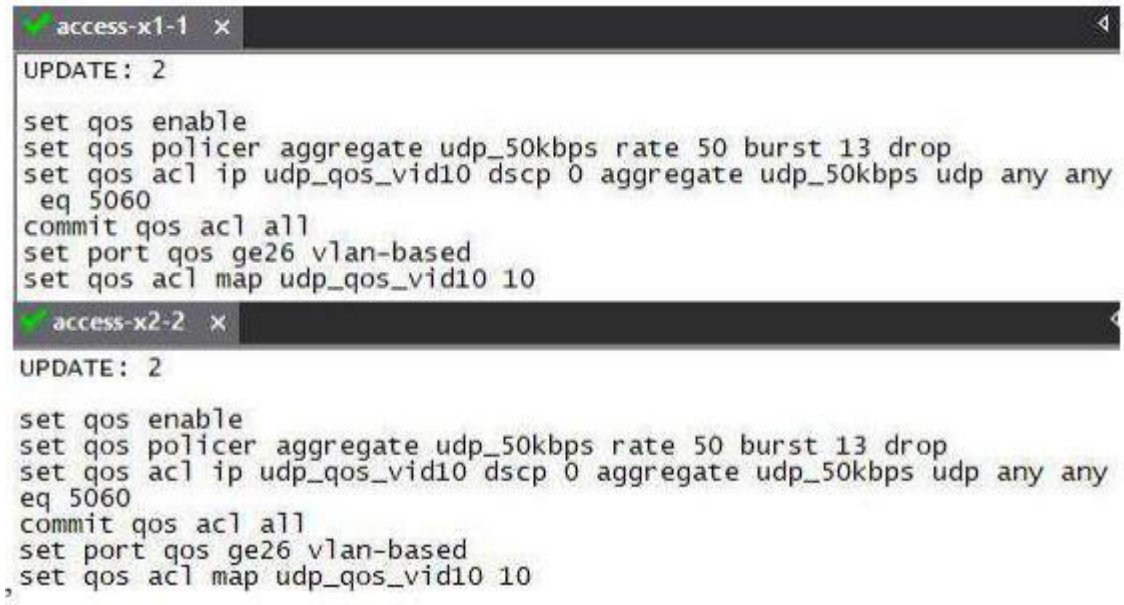


Рисунок 3.12 – Повідомлення Update з налаштуваннями якості обслуговування

Для додаткової перевірки було знято дамп за допомогою утиліти tcpdump на віртуальному каналі між access-x1-1 та access-x1-2 при включеному client -1 (рис. 3.13).

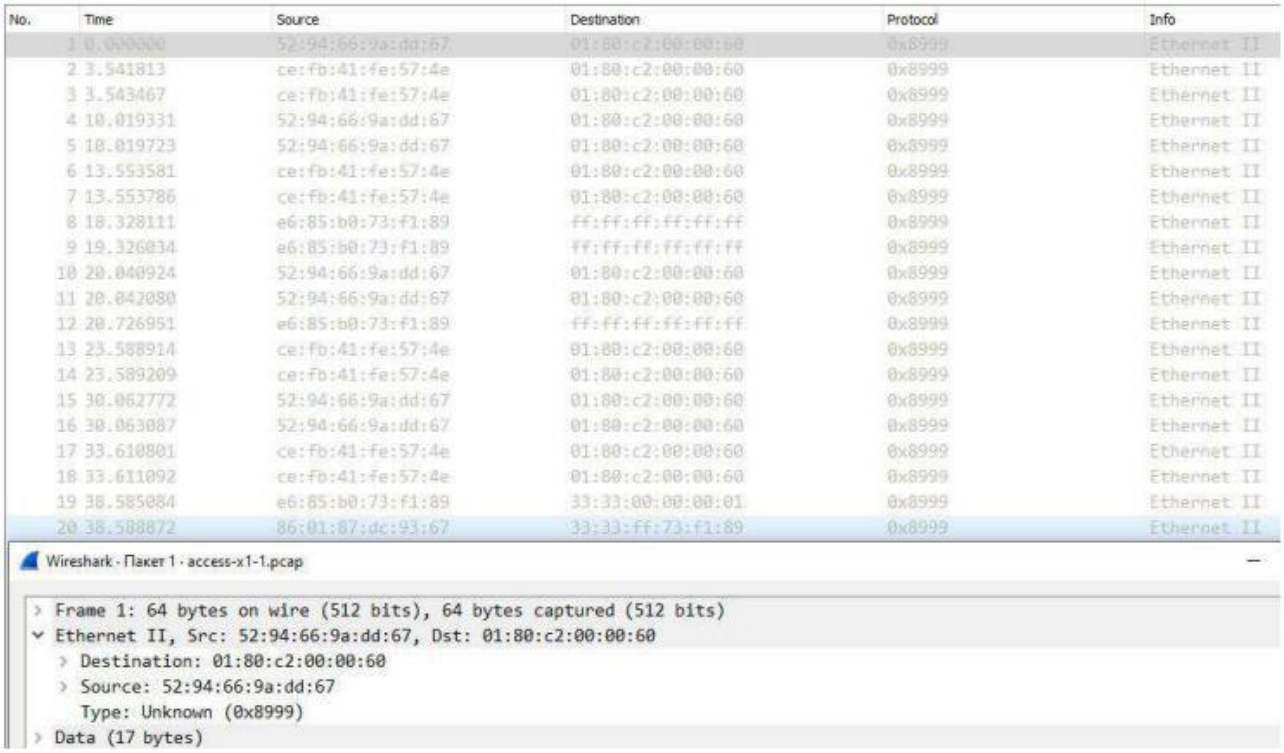


Рисунок 3.13 – Дамп на каналі між access-x1-1 та access-x1-2

Також із використанням графічної утиліти для аналізування мережевих пакетів Wireshark, за допомогою інформації, зображеної на рис. 3.12, можна простежити за роботою протоколу.

Перш за все відбувається пошук сусідів, розсилання повідомлень Hello — кадри № 1 і 2. Далі, після встановлення сусідських відносин, комутатори обмінюються повідомленнями Update — кадри № 5 та 7. На кадрах № 8 та 9 client-1 надсилає широкомовний запит для побудови ARP -таблиці. На кадрах № 19 та 20 відбувається передача корисних даних від client -1.

## 3.5 Висновки до третього розділу

В результаті проведеного тестування були отримані результати, завдяки яким можна зробити висновок, що розроблений протокол успішно виконує заявлені функції:

– дозволяє на рівні комутації використовувати всі доступні канали передачі даних;

– ризик виходу з ладу мережі відсутній;

– забезпечується розсилання службових повідомлень.

За допомогою розробленого протоколу вдалося автоматизувати передачу параметрів якості обслуговування по всьому маршруту руху трафіку від відправника до одержувача.

# РОЗДІЛ 4 БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ

## 4.1 Стихійні лиха та їх класифікація

Стихійні дії сил природи, поки що не повною мірою підвладні людині та щорічно завдають державі і населенню величезних збитків. Стихійні лиха - це такі явища природи, що викликають екстремальні ситуації, порушують нормальну життєдіяльність населення, роботу безлічі об'єктів. Стихійні лиха є трагедією для будь-якої держави. Через стихійні лиха страждає економіка країни, бо при цьому руйнуються виробничі підприємства, знищуються матеріальні цінності, гинуть люди.

Стихійні лиха – небезпечні природні явища, як правило раптового походження, хоча іноді і прогнозовані за допомогою метеорології, але на інтенсивність яких люди впливати не можуть. Їх можна класифікувати: за швидкістю переміщення - землетруси, зсуви, цунамі, снігопади, ожеледі - швидкі; підвищення рівня води в ріках через інтенсивні опади або танення снігу, льоду (повіні), звільнення внутрішньої енергії Землі, виверження вулканів - повільні. Часто виникають потужні, високошвидкісні потоки повітря через швидкий перепад значень атмосферного тиску (урагани, смерчі і т.п.). Стихійні лиха речовинного характеру можуть ініціювати виникнення різноманітних полів, які негативно впливають на здоров'я, самопочуття людини [26].

Стихійні явища часто виникають в комплексі, що значно посилює їх негативний вплив. Небезпечні природні явища визначаються трьома основними групами процесів - ендогенні, екзогенні та гідрометеорологічні.

Стихійні лиха, які характерні для України, за структурою можна поділити на прості, що включають один елемент - наприклад, сильний вітер, зсув або землетрус та складні. Вони складаються з декількох процесів однієї групи або

Деякі стихійні лиха (пожежі, обвали, зсуви і навіть землетруси) можуть виникати в результаті дій самих людей, тобто мають антропогенне походження, але наслідки їх завжди є діями сил природи. Для кожного стихійного лиха характерна наявність властивих йому вражаючих чинників, що несприятливо впливають на стан здоров'я, життя людини [26].

Причинами стихійних лих можуть бути:

– швидке переміщення речовини (землетрусу, зсуви);
– вивільнення внутріземної енергії (вулканічна діяльність, землетруси);
– підвищення рівня вод річок, ставків і морів (повені, цунамі);
– вплив надзвичайно сильного вітру (урагани, торнадо, циклони).

Важливо своєчасно провести роботи, спрямовані на локалізацію природного лиха, щоб зменшити зони руйнувань, звести до мінімуму кількість загиблих та постраждалих.

В Україні найчастіше спостерігаються такі надзвичайні ситуації природного характеру:

– небезпечні геологічні явища (зсуви, обвали, осипки, просадки земної поверхні);
– небезпечні метеорологічні явища (зливи, урагани, сильні снігопади, сильний град, ожеледь);
– небезпечні гідрологічні явища (повені, паводки);
– природні пожежі лісових та торф’яних масивів;
– масові інфекції та хвороби людей, тварин, рослин.

В останні роки кількість стихійних лих в Україні та в світі в цілому значно збільшилася. Найчастіше в Україні виникають такі природні катастрофи як землетруси, повені, посухи (на Півдні України), лісові пожежі в літню пору року, снігові замети, зсуви поверхні.

Є серйозні підстави вважати, що масштабність впливу лиха й катастроф на соціальні, економічні, політичні та інших процесів сучасного нашого суспільства та їх драматизм вже перевищили такий рівень, який дозволяв ставитися до них

як до локальних збоїв у розміреному функціонуванні державних та громадських структур [26].

Отже, перед людиною та громадськістю в XXI в. вимальовується нова мета - глобальна безпека. Досягти цього можна, в першу чергу, за допомогою зміни світогляду людини, а також покращення системи профілактичних заходів у боротьбі зі стихійними лихами, а саме: вдосконалення рятувальних служб та рятувальної техніки, проведення попереджувальних заходів та пропагандистської роботи з громадянами щодо правил поведінки та дій під час стихійних лих. Це допоможе в майбутньому зменшити кількість загиблих та постраждалих від природних катастроф, а також зменшить матеріальні збитки, що були завдані стихійним лихом.

Природні лиха з часом нікуди не зникнуть. Будуть виникати землетруси в геологічно активних районах, будуть виникати повені, а штормові припливи стануть, раз у раз затопляти морські узбережжя, не обійдеться і пожеж. Людина безсила запобігти природним процесам, але тільки в наших силах зменшити кількість жертв і матеріальних втрат.

## 4.2 Соціальне значення охорони праці

Соціальне значення охорони праці полягає в сприянні росту ефективності суспільного виробництва шляхом безперервного вдосконалення і поліпшення умов праці, підвищення їх безпеки, зниження виробничого травматизму і профзахворювань [27]. Соціальне значення охорони праці проявляється в зростанні продуктивності праці, збереженні трудових ресурсів і збільшенні сукупного національного продукту.

Охорона праці полягає в сприянні росту ефективності виробництва, яке досягається шляхом безперервного вдосконалення і поліпшення умов праці, підвищення їх безпеки, зниження виробничого травматизму і профзахворювань.

Зростання продуктивності праці відбувається в результаті збільшення фонду робочого часу завдяки скороченню внутрішньо-змінних простоїв шляхом

ліквідації мікротравм або зниження їх кількості, а також завдяки запобіганню передчасного стомлення шляхом раціоналізації і покращення умов праці та введенню оптимальних режимів праці і відпочинку та інших заходів, які сприяють підвищенню ефективності використання робочого часу.

Важливим питанням є зростання продуктивності праці, яка відбувається в результаті збільшення фонду робочого часу завдяки скороченню внутрішньозмінних простоїв шляхом ліквідації мікротравм або зниження їх кількості, а також завдяки запобіганню передчасного стомлення шляхом раціоналізації і покращення умов праці та введенню оптимальних режимів праці і відпочинку та інших заходів, які сприяють підвищенню ефективності використання робочого часу [27].

Особливої уваги заслуговує те, що збереження трудових ресурсів і підвищення професійної активності працюючих відбувається завдяки покращенню стану здоров’я і подовженню середньої тривалості життя шляхом покращення умов праці, що супроводжується високою трудовою активністю і підвищенням виробничого стажу. Підвищується професійний рівень також завдяки зростанню кваліфікації і майстерності. Відповідно і збільшення сукупного національного продукту відбувається завдяки покращенню вищеперелічених показників та їх складових компонентів. Збереження трудових ресурсів і підвищення професійної активності працюючих відбувається завдяки покращенню стану здоров'я і подовженню середньої тривалості життя шляхом покращення умов праці, що супроводжується високою трудовою активністю і підвищенням виробничого стажу. Підвищується професійний рівень також завдяки зростанню кваліфікації і майстерності. Збільшення сукупного національного продукту відбувається завдяки покращенню вищеперелічених показників та їх складових компонентів. Крім того, соціальне значення охорони праці проявляється в зростанні продуктивності праці, збереженні трудових ресурсів.

Комплекс заходів з поліпшення умов праці може забезпечити приріст продуктивності праці на 15-20%. Так, нормалізація освітлення робочих місць

збільшує продуктивність на 6-13% та скорочує брак на 25%. Раціональна організація робочого місця підвищує продуктивність праці на 21%, раціональне фарбування робочих приміщень – на 25% [27]. Збільшення ефективного фонду робочого часу може бути досягнуто за рахунок скорочення тимчасової непрацездатності працівників внаслідок хвороб та виробничого травматизму.

## 4.3 Висновки до четвертого розділу

В цьому розділі розглянуто важливі питання безпеки життєдіяльності та основи охорони праці, а саме: описано стихійні лиха і їх класифікація та соціальне значення охорони праці.

# ВИСНОВКИ

У ході виконання роботи здійснено дослідження наявних методів та моделей управління QoS. Інформація про них, включаючи переваги та недоліки, міститься у огляді предметної області. На основі цієї інформації було ухвалено рішення про розробку нового протоколу, який дозволяє автоматизувати налаштування якості обслуговування без впровадження SDN та автоматизованих систем, представлених на ринку.

Для реалізації заявлених функцій нового протоколу був розроблений алгоритм і написаний код високорівневою мовою програмування Python. Створений скрипт був запущений на стенді із ВМ, що працюють ОС GNU/Linux Ubuntu. Дев'ять ВМ виконували функції комутаторів, а чотири кінцевих пристроїв. За результатами проведених тестів було встановлено, що розроблений протокол виконує заявлені функції та дозволяє автоматизувати налаштування QoS по всьому маршруту руху трафіку.

# СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ


1. Микитишин А. Г., Митник М. М., Стухляк П. Д. Телекомунікаційні системи та мережі. Тернопіль: Тернопільський національний технічний університет імені Івана Пулюя, 2017. 384 с.
2. Микитишин А. Г., Митник М. М., Стухляк П. Д., Пасічник В. В. Комп'ютерні мережі. Книга 2. [навчальний посібник]. Львів : "Магнолія 2006", 2014. 312 с.
3. Jonathan Davidson, James F. Peters, Manoj Bhatia, Satish Kalidindi, Sudipto Mukherjee. Voice over IP Fundamentals, 2nd Edition. Cisco Press. 2006. 394 p.
4. ISO 8402-94 Управління якістю та забезпечення якості. Словник
5. Software-defined networking (SDN): визначення й особливості програмно-визначених мереж. - URL: https://netwave.ua/blog/software-defined-networking-sdn-viznachennya-j-osoblivosti-programno-viznachenih-merezh/ (дата звертання: 12.05.2025)
6. Гніденко М.П., Вишнівський В.В., Ільїн О.О.. «Побудова SDN мереж». К: ДУТ, 2019. 190 с.
7. Основи QoS. - URL: https://edu-cisco.org/uk/courses/quality-of-service/ (дата звертання: 13.05.2025).
8. Cisco AutoQoS. - URL: https://www.cisco.com/c/en/us/products/ios-nx-os-software/autoqos/index.html (дата звертання: 14.05.2025).
9. QoS Policing on Catalyst 6500/6000 Series Switches. - URL: https://www.cisco.com/c/en/us/support/docs/switches/catalyst-6000-series-switches/12493-102.html (дата звертання: 14.05.2025).
10. Єфіменко А. А. Основи побудови локальних комп'ютерних мереж Ethernet на базі керованих комутаторів компанії Cisco : навчальний посібник. Житомир : Житомирська політехніка, 2021. 116 с.
11. OSPF. - URL: https://www.vpnunlimited.com/ua/help/cybersecurity/ospf?srsltid=AfmBOorhkLyz7TMuo9qc_YGTfu4EQW5-7COUbgS7XQkCV

4X47h674IOy (дата звертання: 14.05.2025).

12. Resource management: Linux kernel Namespaces and cgroups. - URL: http://docplayer.net/14284170-Resource-management-linux-kernel- namespaces-and-cgroups.html (дата звертання: 16.05.2025).

13. Микитишин А. Г., Митник М. М., Стухляк П. Д., Пасічник В. В. Комп'ютерні мережі. Книга 1 [навчальний посібник]. Львів : «Магнолія 2006», 2013. 256 с.

14. Жаровський Р.О., Луцик Н.С., Осухівська Г.М., Паламар А.М., Тиш Є.В. Методичні вказівки до виконання кваліфікаційної роботи бакалавра для здобувачів першого (бакалаврського) рівня вищої освіти за спеціальністю 123 «Комп'ютерна інженерія» усіх форм навчання. Тернопіль: ТНТУ, 2024. 39 с.

15. Голь В.Д., Ірха М.С. Телекомунікаційні та інформаційні мережі: навчальний посібник. Київ : ІСЗЗІ КПІ ім. Ігоря Сікорського, 2021. 250 с.

16. P. Anakhov, V. Zhebka, G. Grynkevych and A. Makarenko. Protection of telecommunication network from natural hazards of global warming. Eastern Europian Journal of Enterprise Technologies. 2020. No. 3/10, pp. 26 –37.

17. Побудова системи IP телефонії на базі Cisco. - URL: https://techexpert.ua/solutions-it/building-an-ip-telephony-system-based-on-cisco/ (дата звертання: 16.05.2025).

18. Li, D. etc. A survey of network update in SDN // Frontiers of computer science. 2017. № 1. pp. 4-12.

19. Y.1540. Служба передачі даних по міжмережевому протоколу (IP) – Параметри робочих характеристик перенесення та доступності IP-пакетів. 2016.

20. Лупенко С. А., Пасічник В. В., Тиш Є. В. Комп'ютерна логіка. Львів: Видавництво «Магнолія - 2006». 2015. 354 с.

21. Чайковський А.В., Жаровський Р.О., Лещишин Ю.З Конспект лекцій з дисципліни «Дослідження і проєктування комп'ютерних систем та мереж» для студентів спеціальності 123 – Комп'ютерна інженерія. Тернопіль, 2021. 148 с.

22. Quality of Service (QoS) в High-Priority Applications. - URL: https://www.transition.com/wp-content/uploads/2016/05/qos_wp.pdf (дата

звертання: 19.05.2025).

23. Огляд якості обслуговування (QoS). - URL: https://technet.microsoft.com/library/hh831679.aspx (дата звертання: 19.05.2025).

24. Cisco Collaboration System Solution Reference Network Designs. Available at: - URL: http://www.cisco.com/c/en/us/td/docs/voice_ip_comnycucm/smd/collabl 1 col lab 11/ cac.html (дата звертання: 20.05.2025).

25. Луцків А., Лупенко С., Пасічник В. Паралельні та розподільнені обчислення. Навчальний посібник. Львів: Видавництво «Магнолія 2006», 2024. 566 с.

26. Толок А.О. Крюковська О.А. Безпека життєдіяльності: Навч. посібник, 2011. 215 с.

27. Основи охорони праці: Підручник.; 3-те видання / За ред. К. Н Ткачука. К.: Основа, 2011. 480 с.

Додаток А.

Технічне завдання

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп’ютерно-інформаційних систем і програмної інженерії

Кафедра комп’ютерних систем та мереж

**“Затверджую”**

Завідувач кафедри КС

_____ Осухівська Г.М.

“___” _____2025 р

# КОМП’ЮТЕРИЗОВАНА СИСТЕМА УПРАВЛІННЯ ЯКІСТЮ ОБСЛУГОВУВАННЯ КОРПОРАТИВНИХ МЕРЕЖ

## ТЕХНІЧНЕ ЗАВДАННЯ

на _8_ листках

**Вид робіт:** Кваліфікаційна робота

**На здобуття освітнього ступеня «Бакалавр»**

**Спеціальність 123 «Комп’ютерна інженерія»**

| «УЗГОДЖЕНО» | «ВИКОНАВЕЦЬ» |
|---|---|
| Керівник кваліфікаційної роботи | Студент групи СІ-42 |
| _____ к.т.н., доц. Баран І.О. | _____ Шуб’як Д.І. |
| «____» _____ 2025 р. | «____» _____ 2025 р. |

**Тернопіль 2025**

# 1 Загальні відомості

## 1.1 Повна назва та її умовне позначення

Повна назва теми кваліфікаційної роботи: «Комп’ютеризована система управління якістю обслуговування корпоративних мереж».

Умовне позначення кваліфікаційної роботи: КС КРБ 123.246.00.00

## 1.2 Виконавець

Студент групи СІ-42, факультету комп’ютерно-інформаційних систем і програмної інженерії, кафедри комп’ютерної інженерії, Тернопільського національного технічного університету імені Івана Пулюя, Шуб’як Денис Ігорович.

## 1.3 Підстава для виконання роботи

Підставою для виконання кваліфікаційної роботи є наказ по університету (№ 4/7-53 від 27.01.2025 р.)

## 1.4 Планові терміни початку та завершення роботи

Плановий термін початку виконання кваліфікаційної роботи – 28.01.2025 р.

Плановий термін завершення виконання кваліфікаційної роботи – 25.06.2025 р.

## 1.5 Порядок оформлення та пред’явлення результатів роботи

Порядок оформлення пояснювальної записки та графічного матеріалу здійснюється у відповідності до чинних норм та правил ІСО, ЕСКД, ЕСПД та ДСТУ.

Пред’явлення проміжних результатів роботи з виконання кваліфікаційної роботи здійснюється у відповідності до графіку, затвердженого керівником роботи.

Попередній захист кваліфікаційної роботи відбувається при готовності роботи на 90% , наявності пояснювальної записки та графічного матеріалу.

Пред’явлення результатів кваліфікаційної роботи відбувається шляхом захисту на відповідному засіданні ЕК, ілюстрацією основних досягнень за допомогою графічного матеріалу.

# 2 Призначення і цілі створення системи

## 2.1 Призначення системи

Комп’ютеризована система управління якістю обслуговування корпоративних мереж призначена для оптимізації керування трафіком (розподілення за категоріями і надання пріоритетів), що здатне забезпечити ефективне функціонування систем різного роду.

До складу системи повинні входити як апаратна складова, так і програмна.

Доцільність створення системи зумовлена тим, що існує потреба в корпоратвиних мережах забезпечити для даних із більшим пріоритетом значно якісніший режим мережевого передавання, який не залежить від умов щодо пропускної здатності потоків тих застосунків, котрі є не такими важливими.

## 2.2 Мета створення системи

Основна метою є створення нового методу управління якістю обслуговування корпоративних мереж в межах комп’ютеризованої системи.

Для того, щоб досягти поставленої мети роботи, необхідно розв’язати наступні задачі:

– обґрунтувати актуальність теми роботи;

– дослідити наявні методи і моделі керування QoS мережі із виявленням їх недоліків;

– створити та описати новий метод керування QoS мережі;

– запропонувати можливі напрямки розвитку роботи надалі.

В результаті застосування системи будуть вирішені проблеми, що виникають при використанні існуючих методів, зокрема трудомісткість та помилки конфігурації через людський фактор, значні фінансові витрати по впровадженню SDN та

автоматизованих систем, неможливість переналаштування QoS при зміні топології мережі у автоматизованих систем. Ці проблеми будуть вирішені за рахунок використання створеного ПЗ, функціонал якого дозволить автоматизувати та оптимізувати процес налаштування мережного обладнання.

## 2.3 Характеристика системи

### 2.3.1 Основні задачі та функції системи

Розробка дозволятиме автоматизувати налаштування якості обслуговування без впровадження SDN та автоматизованих систем, представлених на ринку.

Основні функції, що вимагають реалізації в комп’ютеризованій системі управління якістю обслуговування корпоративних мереж:

– розподіл трафіку за категоріями і надання йому пріоритетів;

– забезпечення для даних із більшим пріоритетом значно якіснішого режиму мережевої передачі пакетів;

– аналіз пакетів, внесення змін до даних в пакетах і правил пересилання;

– застосування політик на кінцевих пристроях;

– підтримка встановлених сусідських відносин між пристроями мережі.

# 3 Вимоги до системи

## 3.1 Вимоги до системи в цілому

Система повинна бути  спроектована так, щоб до її складу особливих зусиль можна інтегрувати різні елементи, не порушуючи при цьому структуру системи.

У проектованій системі повинні бути забезпечені:

– проведення аналіз структури мережевого трафіку;

– можливість оцінювання ефективності використання комп'ютерних мереж;

– розподіл навантаження рівномірно за усіма доступними каналами зв'язку;

– продуктивність роботи програмного забезпечення;

– гарантування часу реакції та пропускної здатності мережі;

– часова ефективність та ефективність використання ресурсів системи.

### 3.1.1 Вимоги до структури та функціонування системи

До структури та функціонування комп’ютеризованої системи управління якістю обслуговування корпоративних мереж:

- локальна комп’ютерна мережа;
- віртуальні машини;
- комутатори;
- сховище зібраних даних;
- програмне середовище для аналізу трафіку.

### 3.1.2 Вимоги до способів та засобів зв’язку між компонентами системи

Після включення протоколу комутатори роблять розсилку hello - повідомлень на групову адресу 01:80: C2:00:00:60 і очікують у відповідь такі ж hello - повідомлення , комутатори встановлюють сусідські відносини, додаючи кожен одне одного у свою локальну таблицю сусідів. У ході роботи протокол аналізує кадри та зберігає найкоротші маршрути між мережевими пристроями. Для цього створено таблицю, схожу на стандартну таблицю MAC -адрес, але вона має додаткове поле Metrics, а також можливість зберігати для однієї адреси кілька інтерфейсів

### 3.1.3 Вимоги по діагностуванню системи

Діагностика комп’ютеризованої система управління якістю обслуговування корпоративних мереж відбувається у відповідності до затвердженого розкладу профілактичних заходів.

### 3.1.4 Перспективи розвитку, модернізація системи

Перспективами розвитку та модернізації комп’ютеризованої системи управління якістю обслуговування корпоративних мереж є розширення функціонального наповнення для аналізу трафіку на усіх мережевих ділянках.

### 3.1.5 Вимоги до надійності системи

Комп’ютеризована система управління якістю обслуговування корпоративних мереж повинна бути захищена на рівнях моделі OSI.

Фізичний рівень захисту повинен забезпечувати надійність щодо доступу до апаратного забезпечення. Програмний рівень захисту повинен передбачати захист від сторонніх втручань і впливів.

### 3.1.6 Вимоги до функцій та задач, які виконує система

Перед адміністратором мережі стоїть завдання надати певний клас QoS для Ip-телефонів. Для виконання поставленого завдання необхідно пройти такі етапи:

– забезпечити механізм налаштування комутатора доступу. На даному етапі необхідно визначити, які саме налаштування потрібно вказати на кінцевому мережевому обладнанні та здійснити його конфігурування;

– комутатор доступу повинен здійснити пошук сусідніх комутаторів та спільно з ними побудувати топологію мережі;

– комутатори повинні здійснити побудову найкоротшого маршруту між Ip-телефонами у різних кінцях мережі;

– комутатори обчислюють та автоматично виставляють політику обслуговування до транзитних портів, якими буде передаватися трафік, а також до портів доступу для Ip-телефонів

### 3.1.7 Вимоги до апаратного забезпечення

– Віртуальна машина з двома процесорами Intel Xeon E5320 з тактовою частотою 1 Гб – 4 шт;

– Комутатор Cisco Catalyst 6500/6000.

### 3.1.8 Вимоги до програмного забезпечення

Мова програмування для розробки - Python

Операційна система – GNU/Linux Ubuntu 12.04.5

Застосунки, які діагностують мережу - ping, tcpdump.

## 4 Вимоги до документації

Документація повинна відповідати вимогам ЄСКД та ДСТУ

Комплект документації повинен складатись з:

– пояснювальної записки;

– графічного матеріалу:

1 Структурна схема мережі для демонстрації роботи

2 Структурна схема стенду

3 Топологія мережі на тестовому стенді

4 Результати виконання команди ping між клієнтами

*Примітка: У комплект документації можуть вноситися міни та доповнення в процесі розробки.

## 5 Техніко-економічні показники

Планована собівартість комп'ютеризованої системи управління якістю обслуговування корпоративних мереж повинна становити не більше 20 000 грн.

*Примітка: собівартість системи може змінюватись під час розрахунку в процесі розробки.

## 6 Стадії та етапи проектування

Таблиця 1 – Стадії та етапи виконання кваліфікаційної роботи бакалавра

| № етапу | Назва етапу виконання кваліфікаційної роботи | Термін виконання |
|---|---|---|
| 1 | 2 | 3 |
| 1. | Ознайомлення з завданням до кваліфікаційної роботи | 24.04 – 25.04 |
| 2. | Розробка технічного завдання | 26.04 – 29.04 |
| 3 | Підбір джерел про управління якістю обслуговування | |
| | корпоративних мереж | 22.02 – 08.03 |
| 4 | Опрацювання літературних джерел | 09.03 – 26.03 |
| 5 | Виконання дослідження щодо розробки системи | |
| | управління якістю обслуговування корпоративних | |
| | мереж | 27.03 – 16.04 |
| 6 | Написання програмного коду і створення стенду | 17.04 – 28.04 |
| 7 | Оформлення розділу «Огляд предметної області» | 29.04 – 02.05 |
| 8 | Оформлення розділу «Проектна частина» | 03.05 – 14.05 |
| 9 | Оформлення розділу «Практична частина» | 15.05 – 25.05 |

| 1 | 2 | 3 |
|---|---|---|
| 10 | Виконання завдання до підрозділу «Безпека | |
| | життєдіяльності, основи охорони праці» | 26.05 – 31.05 |
| 11 | Оформлення кваліфікаційної роботи | 01.06 – 11.06 |
| 12 | Нормоконтроль | 09.06 – 12.06 |
| 13 | Перевірка на плагіат | 11.06 – 14.06 |
| 14 | Попередній захист кваліфікаційної роботи | 14.06 – 17.06 |
| 15 | Захист кваліфікаційної роботи | 24.06 |

## 7 Додаткові умови виконання кваліфікаційної роботи

Під час виконання кваліфікаційної роботи у дане технічне завдання можуть вноситися зміни та доповнення.