Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя
(повне найменування вищого навчального закладу)
Факультет комп'ютерно-інформаційних систем і програмної інженерії
Охорона (назва факультету)
Кафедра комп'ютерних систем та мереж
(повна назва кафедри)

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

Магістр
(назва освітнього ступеня)

на тему: **Методи управління багатоадресною передачею даних в комп’ютерних мережах**

Виконав: студент 6 курсу групи СІм-61
спеціальності
123 «Комп’ютерна інженерія»
(шифр і назва спеціальності (напряму підготовки))

Сабат Р.М.
(підпис) (прізвище та ініціали)

Керівник Баран І.О.
(підпис) (прізвище та ініціали)

Нормоконтроль Тиш Є.В.
(підпис) (прізвище та ініціали))

Завідувач кафедри Осухівська Г.М.
(підпис) (прізвище та ініціали)

Рецензент Цуприк Г.Б.
(підпис) (прізвище та ініціали)

м. Тернопіль – 2023

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
_____ *доц. Осухівська Г.М.*
(підпис) (прізвище та ініціали)
« » 20__ р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

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

за спеціальністю *123 Комп'ютерна інженерія*
студенту *Сабату Роману Миколайовичу*
(прізвище, ім'я, по батькові)

1. Тема роботи *Методи управління багатоадресною передачею даних в комп'ютерних мережах*

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

Затверджені наказом по університету від *« 01 » грудня 2023 року № 4/7-1132*

2. Термін подання студентом роботи *26.12.2023*

3. Вихідні дані до роботи *наукові літературні джерела*

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)
1 Аналіз предметної області дослідження. 2 Теоретична частина. 3. Практична частина. 4 Охорона праці та безпека в надзвичайних ситуаціях

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, слайдів)
1. Тема, мета, задачі, об'єкт, предмет, новизна дослідження. 2. Актуальність дослідження. 3. Існуючі методи передачі даних. 4. Порівняльний аналіз методів передачі даних. 5. Модель мережі_Глобальне багатоадресне дерево. 6. Об'єкти протоколу. 7. Тестування віртуального стенду 8. Висновки.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Охорона праці	*Осухівська Г.М., доцент*		
Безпека в НС	*Стадник. І.Я., проф. каф. ОХ*		

7. Дата видачі завдання _____ 2023 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Термін виконання етапів роботи	Примітка
1	*Затвердження теми кваліфікаційної роботи*	*01.12.23*	*Виконано*
2	*Аналіз літературних джерел*	*02.12-06.12.23*	*Виконано*
3	*Обґрунтування актуальності дослідження*	*07.12-09.12.23*	*Виконано*
4	*Аналіз предмету дослідження та предметної області*	*10.12-11.12.23*	*Виконано*
5	*Проведення дослідження методів та засобів аналітичного опрацювання даних*	*11.12-12.12.23*	*Виконано*
6	*Оформлення розділу «Аналіз предметної області дослідження»*	*12.12-13.12.23*	*Виконано*
7	*Оформлення розділу «Теоретична частина»*	*13.12-14.12.23*	*Виконано*
8	*Оформлення розділу «Практична частина»*	*14.12-15.12.23*	*Виконано*
9	*Оформлення розділу «Охорона праці та безпека в надзвичайних ситуаціях»*	*07.12-12.12.23*	*Виконано*
10	*Нормоконтроль*	*11.12-15.12.23*	
11	*Попередній захист роботи*	*19.12.23*	*Виконано*
12	*Захист кваліфікаційної роботи*	*27.12.23*	

Студент _____ *Сабат Р.М.*
(підпис) (прізвище та ініціали)

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

АНОТАЦІЯ

Методи управління багатоадресною передачею даних в комп’ютерних мережах // Кваліфікаційна робота за освітнім рівнем «магістр» // Сабат Роман Миколайович // Тернопільський національний технічний університет імені Івана Пулюя, факультет комп’ютерно–інформаційних систем і програмної інженерії, кафедра комп’ютерних систем та мереж, група СІм–62 // Тернопіль, 2023 // с. – 70, рис. – 24, табл. – 4, аркушів А1 – 8 , бібліогр. – 26.

Ключові слова: MULTICAST, ACK, PGM, M/TCP, НАДІЙНА ДОСТАВКА ДАНИХ, БАГАТОАДРЕСНЕ ДЕРЕВО

Кваліфікаційна робота присвячена дослідженню методів багатоадресної передачі та розробці алгоритму функціонування надійного багатоадресного транспортного протоколу.

Виконано аналітичний огляду існуючих багатоадресних транспортних протоколів, та визначено критерії, що висуваються до нового протоколу.

Наведено результати розробки алгоритму роботи розроблюваного протоколу, що відповідає визначеним критеріям. Сформовано глобальне багатоадресне дерево. Було описано основні механізми функціонування протоколу та алгоритм його роботи. Приведено опис застосування багаторівневої ієрархії.

Описано демонстраційну реалізацію протоколу, описано віртуальний стенд, на якому проводиться тестування реалізації, а також наведено результати тестування.

В результаті роботи розроблено алгоритм функціонування надійного багатоадресного транспортного протоколу та проведено перевірку його працездатності на віртуальному стенді.

ANNOTATION

Methods of managing multicast data transmission in computer networks // Master thesis // Sabat Roman // Ternopil Ivan Pul'uj National Technical University, Faculty of Computer Information Systems and Software Engineering, Department of Computer Systems and Nets, group CIm - 61 // Ternopil, 2023 // p. – 70, fig. – 24 , table. – 4, Sheets A1 - 8 , Ref. - 26.

Keywords: MULTICAST, ACK, PGM, M/TCP, RELIABLE DATA DELIVERY, MULTI-ADDRESS TREE

The thesis deals with the the research of multicast transmission methods and the development of an algorithm for the operation of a reliable multicast transport protocol.

An analytical review of the existing multicast transport protocols was performed, and the criteria put forward for the new protocol were determined.

The results of the development of the working algorithm of the developed protocol, which meets the specified criteria, are presented. A global multicast tree is formed. The main mechanisms of the protocol's functioning and the algorithm of its work were described. A description of the application of a multi-level hierarchy is given.

The demonstration implementation of the protocol is described, the virtual stand on which the implementation is tested is described, and the test results are also given.

As a result of the work, an algorithm for the operation of a reliable multicast transport protocol was developed and its performance was tested on a virtual stand.

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ СКОРОЧЕНЬ І ТЕРМІНІВ

ACK (Acknowledge) – підтвердження

ACK-based mechanisms – механізм на основі підтвердження пакету даних

ADU (Application Data Unim) – блок даних програми

DR (Designated Receiver) – виділений приймач / призначений одержувач

FEC (Forward Error Correction) – механізм на основі прямої корекції помилок

M/TCP (Multicast-extension to TCP) – багатоадресне розширення для TCP

NAK / NACK (Negative Acknowledge) – негативне підтвердження

NACK-based mechanisms – механізм на основі негативних підтверджень

NCF (NAK Confirmation) – підтвердження про отримання NAK

PGM – Pragmatic General Multicast

RM (Reliable Multicast) – надійна багатоадресна передача

TCP(Transmission Control Protocol) –протокол управління передачею

Tree-based ACK mechanisms – механізм на основі дерева підтверджень

TTL (Time to Live) – час життя пакету даних в IP-протоколі

UDP (User Datagram Protocol, UDP) – один із протоколів в стеку TCP/IP

НБТП – надійний багатоадресний транспортний протокол

ПЗ – програмне забезпечення

ЗМІСТ

ВСТУП

Актуальність теми. Завдання гарантованої передачі даних групі одержувачів часто виникає в різних програмах, таких як текстові та мультимедійні чати, розповсюдження файлів (наприклад, оновлення ПЗ), аудіо-відео трансляції, і таке інше. Для її вирішення нині застосовуються три основні підходи.

Клієнт-серверний підхід передбачає створення виділеного сервера, який надсилає дані від відправника всім одержувачам. Недоліками цього підходу є наявність єдиної точки відмови, а також високі вимоги до апаратного забезпечення та зв'язку [1].

Децентралізований підхід (peer-to-peer) передбачає створення накладеної мережі безпосередньо між одержувачами. Недоліком цього методу є залежність швидкості передачі інформації від кількості клієнтів, що знаходяться онлайн, а також проблема пошуку вузлів накладеної мережі (бенкетів) [3].

Третій підхід - використання багатоадресної передачі на мережному рівні (multicast). Є механізмом економії смуги пропускання, котра робить трафік коротшим під дією доставляння одного потоку даних відразу множині абонентів. Багатоадресна передача дає змогу кільком одержувачам отримати повідомлення, не передаючи його кожному окремому вузлу. Перевагою підходу є максимально ефективне використання каналів зв'язку. Недоліками є необхідність підтримки мережевого рівня, а також несумісність TCP з multicast. В силу зазначених недоліків цей підхід застосовується, в основному, для аудіо-відео трансляцій у межах корпоративної або провайдерської мережі [4].

Рішенням міг би стати транспортний протокол із гарантованою доставкою потоку даних, який використовує багатоадресну розсилку. Зараз немає універсального протоколу такого типу. Тому актуальним є дослідження методів передачі даних та створення протоколу, що відповідає переліку критеріїв, визначених у результаті порівняння існуючих реалізацій.

Мета дослідження: розробка НБТП.

В роботі поставлено та розв’язано **наступні задачі**:

– дослідити основні критерії до керування передачею даних;

– провести дослідження та порівняльний аналіз існуючих багатоадресних транспортних протоколів;

– визначити вимоги до нового транспортного протоколу;

– розробити алгоритм функціонування нового транспортного протоколу;

– виконати реалізацію НБТП.

Об’єкт дослідження: механізми керування багатоадресним передаванням даних.

Предмет дослідження: управління багатоадресною передачею пакетних даних в мережі.

Методи дослідження: Методологічною основою є інформаційно-аналітичне дослідження, порівняння характеристик наявних способів, вибір оптимальних та аналіз підходів, проведення тестування віртуального стенда.

Наукова новизна отриманих результатів:

– досліджено особливості групування одержувачі в локальні області або домени із застосуванням DR;

– запропоновано відновлювати втрачені пакети локальними повторними передачами, а не повторними передачами від початкового відправника, що дає змогу значно зменшити наскрізну затримку і покращити загальна пропускну здатність;

– впроваджено єдине підтвердження доставки, котре генерується для локальної області, що запобігає розриву підтвердження.

Практичне значення одержаних результатів. Розроблені способи можуть бути використані в реальних комп’ютерних мережах для забезпечення надійного керування багатоадресним розсиланням пакетних даних.

Публікації. Результати дослідження апробовано на XI науково-технічній конференції «Інформаційні моделі, системи та технології» у вигляді опублікованих тез [9].

Структура роботи. Робота складається з пояснювальної записки та графічної частини. Пояснювальна записка складається з вступу, 4 розділів, висновків, списку використаної літератури та додатків. Обсяг роботи: пояснювальна записка – 70 арк. формату А4, графічна частина – 8 аркушів формату А1.

РОЗДІЛ 1
АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

На даний момент немає транспортного протоколу загального призначення з гарантованою доставкою, що використовує технологію RM. Завдання його реалізації є вкрай трудомістким.

Різні додатки мають різні вимоги до транспортного протоколу, цей факт обмежує простір розробки. Так, наприклад, до двох додатків, що мають різні вимоги, не може бути застосовано одне рішення. Однак, існують успішні вузькопрофільні реалізації, призначені для додатків певного типу.

1.1. Основні підходи до управління передачею даних

1.1.1. Обмеження з боку програми

Вимоги до додатків для надійної багатоадресної передачі настільки ж широкі та різноманітні, як і самі додатки. Проте існує низка вимог, які суттєво впливають на розробку протоколу з технологією RM. Короткий список включає необхідність наявності / потреби:

- програмної інформація про те, що всі приймачі отримали дані?
- додатку працювати з великою кількістю приймачів?
- програмі забезпечувати повністю надійну доставку даних?
- додатку здійснювати обмежену за часом доставку даних?

Підтвердження доставки даних. У багатьох додатках логічно визначені одиниці даних повинні доставлятися кільком клієнтам, наприклад, файл чи набір файлів, пакет ПЗ тощо. ADU визначається як логічно відокремлений блок даних, який корисний для програми. У деяких випадках одиниця даних програми може бути досить короткою, щоб поміщатися в один пакет, тоді як в інших випадках може бути набагато довшою за нього [7].

НБТП може гарантувати надійну доставку за допомогою механізму підтвердження, що інформує відправника про факт доставки даних. Існує два типи підтверджень, на рівні:

– ADU;
– пакета.

Перший використовується, щоб, наприклад, визначити, коли необхідно припинити відправлення пакетів конкретного ADU. Другий використовується, наприклад, для визначення моменту, коли можна звільнити буферний простір, що використовується для зберігання пакетів, доставка яких була підтверджена [7].

Деяким додаткам суворо необхідно отримувати підтвердження доставки ADU від усіх приймачів, або інформацію про те, які конкретно приймачі не отримали ADU.

Масштабованість. Масштабування набору одержувачів є одним із найважливіших обмежень при виборі механізму підтвердження доставки. Так, наприклад, протокол використовує механізм, при якому кожен відправник відправляє джерелу пакет ACK після кожного отриманого пакета даних погано масштабується, так як джерелу необхідно обробляти ACK пакети від кожного одержувача тому, при величезній кількості одержувачів можуть виникати навантаження. Для вирішення цієї проблеми можуть бути використані різні механізми підтвердження доставки. Ці механізми описані в пункті 1.1.3.

Повністю надійна доставка. Багато додатків вимагають повністю надійної доставки ADU. У таких додатках втрата якогось сегмента ADU веде до того, що інші сегменти даного блоку перестають бути корисними. Наприклад, програми передачі файлів вимагають повністю надійної доставки.

Однак деякі програми не потребують повної надійності доставки. Прикладом є трансляція аудіо, де відсутні пакети лише знижують якість аудіо, що приймається.

Обмежена за часом доставка даних. Деякі додатки мають жорсткі обмеження часу доставки - якщо дані не надходять до одержувача до певного часу,

немає сенсу їх доставляти взагалі [7]. Така обмеженість може виникати при передачі даних у реальному часі, наприклад, при передачі потокового аудіо або відео (рис. 1.1). У цьому разі надзвичайно важливо, щоб додаток мав контроль у тому, що він передає у конкретний час.

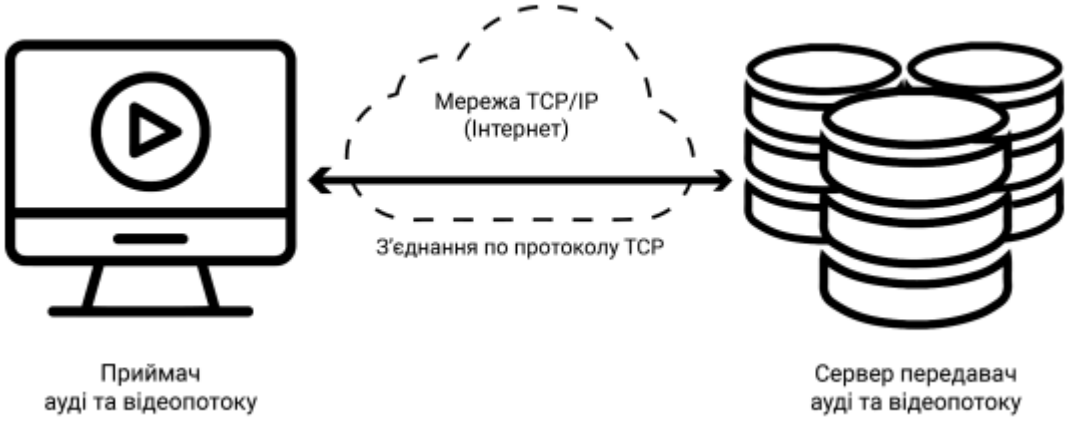

Рисунок 1.1 – Застосування протоколу TCP для потокової передачі пакетів даних

1.1.2. Обмеження з боку мережі

Властивості мережі, в якій розгортається програма, також можуть накладати певні обмеження на розробку протоколу.

Односпрямовані канали. У мережах певних типів може бути відсутня будь-яка форма зворотнього зв'язку. Основним прикладом є супутникове мовлення, де зворотний канал може бути дуже вузьким або взагалі відсутнім. Для таких мереж простір рішень дуже обмежений, оскільки зникає можливість реалізувати більшість механізмів підтвердження доставки. Єдиним рішенням є FEC.

Рівні підтримки з боку мережі. НБТП передбачає механізми, що працюють на кінцевих вузлах та маршрутизатори, які пересилають багатоадресні пакети. Однак, можливі реалізації вдаються до деякого додаткового ступеня підтримки з боку вузлів мережі. Усі можливі реалізації можна розбити на чотири класи

залежно від ступеня підтримки з боку мережі [8]:

– жодної додаткової підтримки. Маршрутизатор просто пересилають пакети, протокол працює тільки на кінцевих вузлах мережі;

– багаторівневий підхід . Цей підхід передбачає, що дані розбиті на кілька груп багатоадресної розсилки, а одержувачі приєднуються до відповідних груп, щоб отримувати лише необхідний трафік. Реалізація цього підходу вимагає додаткової підтримки від маршрутизаторів;

– серверний підхід. У цьому підході застосовуються додаткові вузли, які перебирають на себе частину функцій з доставки даних чи агрегування інформації про це;

– підхід на основі підтримки з боку маршрутизаторів. При використанні даного підходу маршрутизатори беруть на себе весь функціонал з доставки даних одержувачам, який включає і механізми контролю над доставкою і можливе агрегування інформації про їх доставку. Оскільки маршрутизатори можуть безпосередньо впливати на багатоадресну маршрутизацію, вони мають більший контроль над тим, який трафік спрямовується членам групи, ніж вузли на основі серверного підходу. Однак маршрутизатори зазвичай не мають великого обсягу вільної пам'яті або обчислювальної потужності, що обмежує функціонал, який можна реалізувати в них.

1.1.3. Основні механізми підтвердження доставки

Дві основні проблеми, що виникають при розробці протоколу з технологією RM:

– контролю навантаження;

– забезпечення гарної пропускної спроможності.

Втрата пакетів відіграє головну роль щодо цих проблем і є основною перешкодою, яку необхідно подолати для досягнення хорошої пропускної спроможності та забезпечення роботи мережі без перевантажень. Таким чином,

реєстрація втрати пакета та реагування на неї має вирішальне значення для вирішення цих проблем. Механізми контролю за доставкою можуть використовувати один або кілька методів [7, 9]:

– ACK-based mechanism**;**

– негативне підтвердження відсутніх пакетів даних;

– надмірність, що дозволяє приймати не всі пакети.

ACK-based mechanism є найпростішим. Кожен одержувач відправляє відправнику пакет із підтвердженням після кожного прийнятого пакета даних. Якщо підтвердження не надходить, відправник здійснює повторне відправлення пакетів. Такі механізми обмежені дуже невеликою кількістю одержувачів через нездатність відправника обробляти велику кількість підтверджень.

Tree-based ACK Mechanisms організований за структурою зв'язкового ациклічного графа , в якому одержувачі є кінцевими вершинами, а коренем є відправник. Одержувачі генерують ACK -пакет для батьківського вузла, який, у свою чергу, агрегує ці ACK -пакети та відправляє своєму батьківському вузлу, таким чином усі підтвердження доходять до відправника, цей механізм відноситься лише до передачі підтверджень про доставку, в той час як дані передаються як зазвичай від відправника до одержувача. Даний механізм є добре масштабованим і має хорошу стійкість до відмов, але вимагає певного ступеня підтримки з боку мережі.

NAK-based mechanisms, замість надсилання підтверджень для кожного отриманого пакета даних одержувачі надсилають так звані NAK для кожного пакета даних, який вони не отримали. Це має низку переваг перед механізмами на основі ACK [9]:

– відправнику не потрібно знати точну кількість одержувачів. Це усуває етап побудови топології, необхідний алгоритмів з урахуванням ACK;

– підвищення відмовостійкості;

– потрібно лише одне NAK від будь-якого з одержувачів, щоб донести до відправника інформацію про те який пакет втрачено і в якого числа одержувачів.

Недоліками є те, що відправнику важче визначити той момент, коли можна звільнити буфер передачі, також необхідні додаткові механізми, щоб визначити чи дійсно всі пакети даних дійшли до одержувача.

FEC - це добре відомий метод захисту даних від ушкоджень. Найпростіша форма FEC на рівні пакета полягає в тому, щоб застосувати до групи пакетів бітову операцію порозрядного поділу (XOR), внаслідок цієї операції формується новий пакет, який відправляється разом із рештою.

Якщо, наприклад, було відправлено три вихідні пакети плюс пакет XOR, то, якщо одержувач пропустив якийсь із вихідних пакетів даних, але отримав пакет XOR, він може відтворити відсутній вихідний пакет.

1.2. Огляд існуючих методів передачі даних

1.2.1. Протокол PGM

Працює з IP-multicast. PGM гарантує, що одержувач у групі або отримає всі пакети даних, або зможе виявити непоновлену втрату пакетів даних. Масштабованість досягається за рахунок побудови ієрархії та використання наступних механізмів: FEC, NAK та придушення NAK. PGM є експериментальним протоколом [11].

Його використання передбачає наявність у мережі вузлів, які підтримують PGM. Тим не менш, він також призначений для роботи, хоч і з меншою ефективністю, в мережах, де деякі або всі вузли не підтримують PGM [12].

PGM дозволяє приймачам приєднуватися та від'єднуватись у будь-який час, забезпечуючи надійність тільки в межах поточного вікна передачі. Отже, він найкраще підходить для додатків, які здатні відновити дані на своєму рівні, у разі невідновної втрати на рівні протоколу, таким чином PGM кращий для додатків, яким важливо отримувати дані в реальному часі.

Приклади додатків – це поширення котирувань акцій та створення образу

диска. У першому випадку надійність дуже важлива, але якщо на момент скасування котирування передача не відбулася, то від повторної передачі слід відмовитися. При створенні образу диска більш ефективно продовжувати передачу нових даних, а не уповільнювати роботу для кількох «повільних» одержувачів, які пізніше можуть вилучити дані, що бракують, за допомогою стандартних методів клієнт-сервер.

Архітектура. Дерево PGM будується з використанням вузлів, що підтримують PGM у існуючому багатоадресному дереві. Відправник багатоадресно передає послідовні пакети даних (ODATA) одержувачам. Коли одержувачі виявляють відсутні пакети, вони направляють NAK своєму батькові. Батько підтверджує прийом NAK, багатоадресно відправляючи у відповідь NCF. Пакет відновлення (RDATA) генерується або хостом-джерелом, або локальним вузлом, виділеним для процедур відновлення (DLR) у відповідь NAK. RDATA — це залежно від параметрів сеансу або знову відправлений втрачений пакет або пакет FEC. Перед відправкою NAK, одержувачі вводять певний таймер, і якщо до закінчення часу буде отримано відповідний NCF, то відправлення NAK скасовується, таким чином забезпечується отримання лише однієї копії втраченого пакета групи одержувачів [11]. Більш наочно цей процес описаний на рис. 1.2.

Позначення на рис. 1.2: (1) Одержувачі відправляють NAK про втрачений пакет. (2) Батьківський маршрутизатор здійснює багатоадресну передачу NCF. (3) Маршрутизатор направляє NAK своєму батькові, який (4) здійснює багатоадресну передачу NCF. (5) NAK передається відправнику, який відповідає багатоадресним NCF (6). Пізніше інший одержувач виявляє втрату того ж пакета і передає одноадресно NAK (7) своєму батькові, який багатоадресно передає NCF (8). Батько не відправляє висхідний NAK, оскільки він уже наголосив на прийомі відповідного NCF [11].

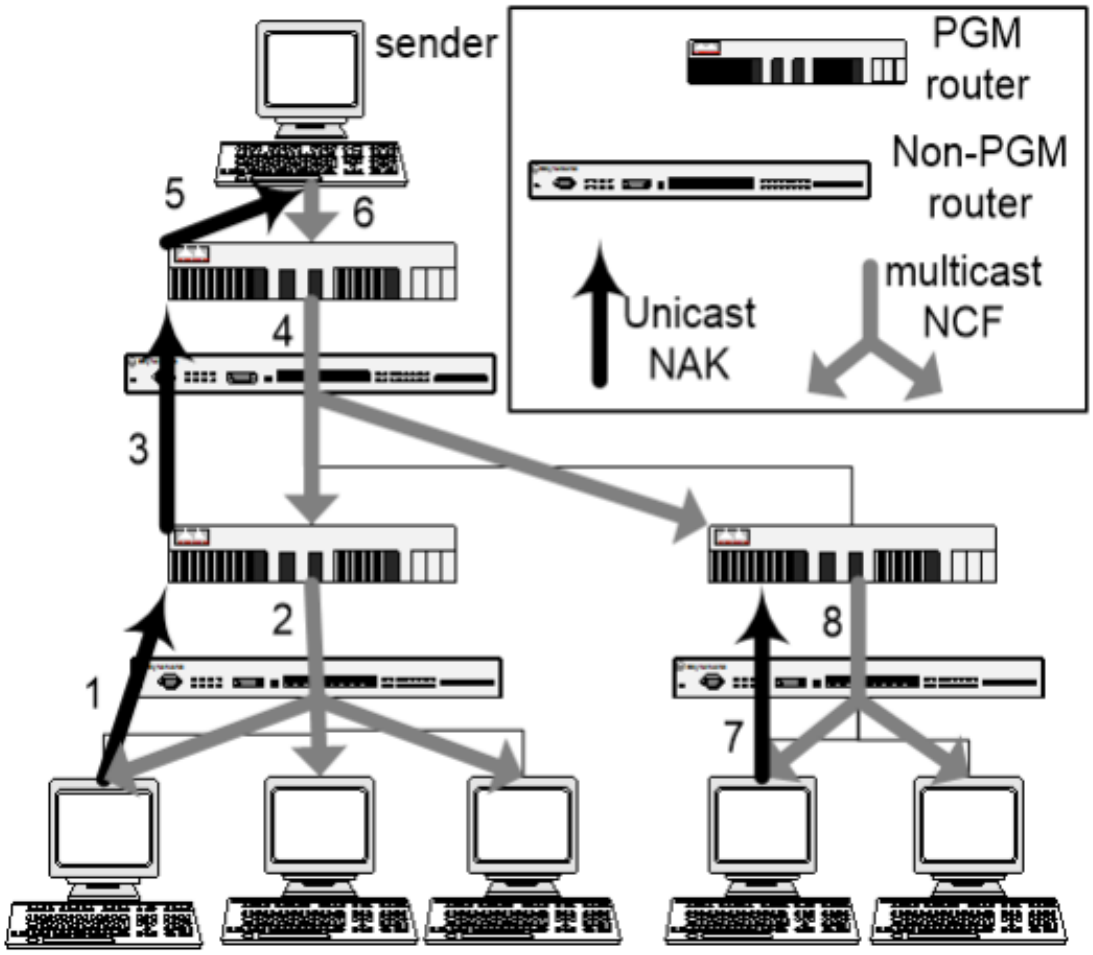

Рисунок 1.2. – Сценарій NAK/NCF

Ієрархія. PGM зберігає та підтримує в актуальному стані дерево шляху від кожного відправника multicast -повідомлень до слухачів. Кожен вузол дерева знає, звідки йому прийшов пакет і якщо лист дерева (слухач multicast -повідомлень) виявив, що пакет не отримано, підтвердження втраченого пакета може піднятися вгору від листа до кореня дерева (відправник multicast -повідомлень). На кожному вузлі дерева виставляється стан, що в підмережі втрачено пакет. При повторному надсиланні втрачений пакет може ігноруватися на вузлах, де цей стан не виставлено і дійде лише до тих підмереж, де сталася втрата пакета. Підтримка дерева шляху досягається завдяки повідомленням шляху (path messages) які надсилаються в multicast -групи через рівні проміжки часу кожним відправником [11].

1.2.2. Протокол M/TCP

Розроблений на основі ініційованої відправником багатоадресної передачі (SIM) та явної багатоадресної розсилки (Xcast) [9, 10].

M/TCP може застосовуватися до тих програм, у яких відправник ініціює передачу даних. Наприклад FTP, SMTP. Для додатків, що працюють із даними протоколами, потрібна лише реалізація на стороні відправника. Одержувачі приймають пакети як одноадресні, тому немає потреби змінювати як стеки протоколів на стороні одержувача, так і програми.

M/TCP надає можливість надійної багатоадресної передачі даних та підтримує існуючі програми, що працюють на основі звичайного TCP. Схема передачі у M/TCP представлена на рис. 1.3.

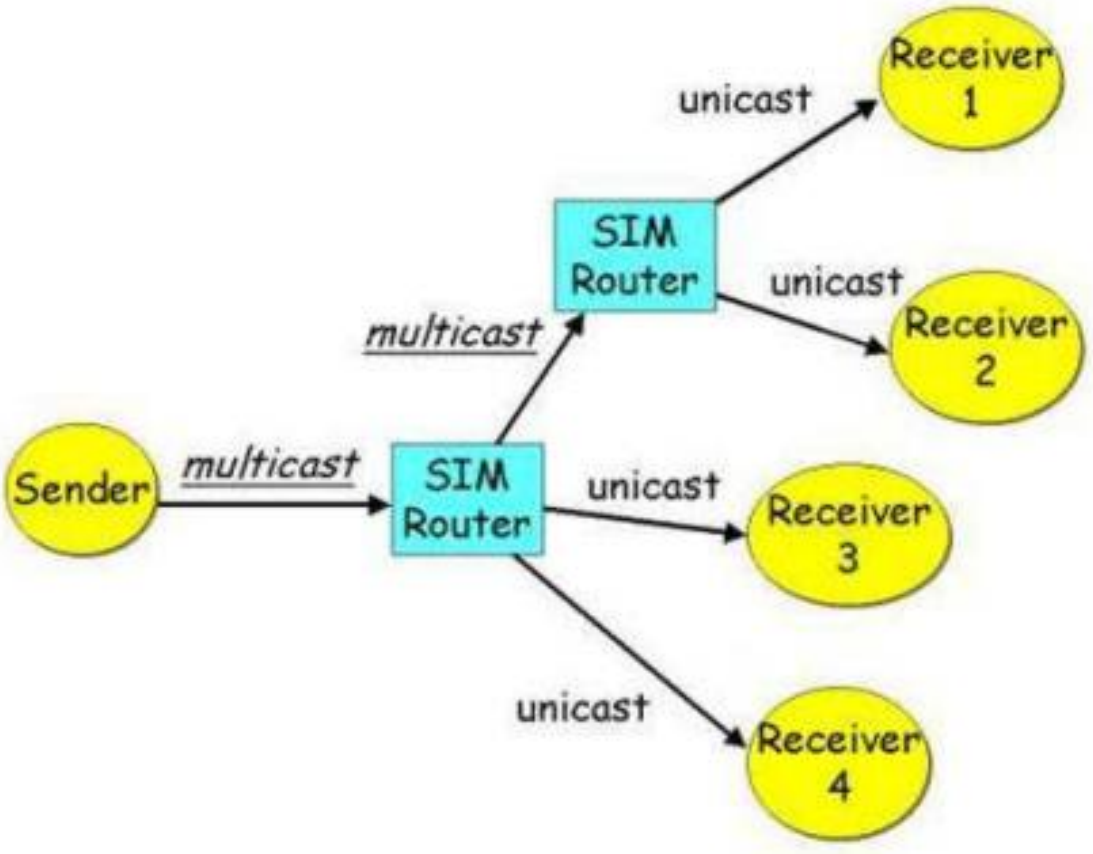

Рисунок 1.3 – Передача даних M/TCP

Основними недоліками M/TCP є необхідність підтримки з боку мережі, можливість роботи лише з невеликою кількістю приймачів та можливість роботи лише з певними типами додатків [10].

M/TCP призначений для підтримки додатків, що мають такі характеристики [10]:

– висока надійність. Оскільки M/TCP є різновидом TCP він забезпечує високу надійність передачі;

– асиметрична передача даних. Фактично, M/TCP призначений для багатоадресних обмінів одним-багатьом. M/TCP застосовується, в основному, для асиметричної програми, де трафік від відправника набагато більший, ніж трафік від одержувачів, передача ініціюється з боку відправника;

– не в режимі реального часу. Механізми управління швидкістю передачі, що використовуються в TCP, не підходять для передачі мультимедійних даних у реальному часі.

При передачі даних M/TCP враховує можливості кожного приймача. Коли піднабір одержувачів групи втрачає пакет, втрачений пакет передається знову у іншому потоці TCP. Таким чином швидкість передачі не залежить від найповільнішого приймача [14].

При втраті пакета виконується наступний алгоритм [14]:

– якщо два або більше одержувачів втратили той самий пакет, повторна передача буде виконуватися багатоадресною передачею;

– якщо один одержувач втратив пакет, передача буде виконана багатоадресно;

– після завершення повторної передачі нові пакети передаватимуть окремо створеній групі одержувачів, у яких спостерігалася втрата пакетів.

Ця група буде знову поєднана з основною, коли досягне однакової швидкості з вихідним потоком передачі.

1.2.3. Протокол UDP

Він є простим та достатньо швидкісним вирішенням задачі передачі пакетів у комп'ютерній мережі. UDP не дає жодних гарантійних зобов’язань їх доставляння та не здійснює контроль порядку прийняття даних [15]. Власне саме це і робить його дуже корисним для передачі таких даних, котрі не потребують точного доставляння, в т.ч. відео-даних (рис. 1.4).

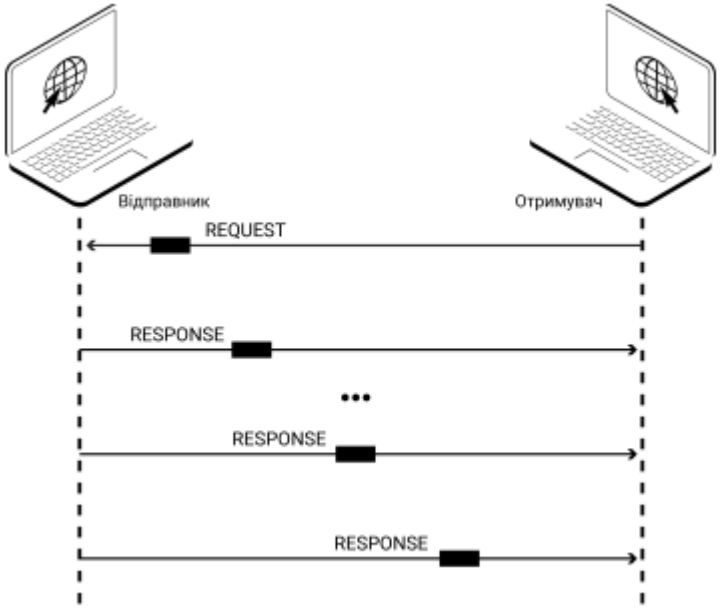

Рисунок 1.4 – З'єднання пристроїв для передачі даних при використанні UDP протоколу

Основне призначення UDP – передача пакетних даних в тих мережах, в яких застосовується їх комутація. Протокол вважає, що стандартний IP-протокол має нижчий ранг виконання. UDP також має переваги над TCP, в т.ч. застосування Broadcast та Multicast (рис. 1.5).

Broadcast здійснює пакетне надсилання даних усім пристроям в комп’ютерній мережі. Це корисно тоді, коли треба переслати ті ж самі дані множині пристроїв, наприклад, при розсиланні повідомлень щодо стану мережі

чи при мультимедія-транслюванні. Multicast UDP передає пакети даних лише визначеній групі пристроїв, котрі відповідають визначеним параметрам. Це дає ефективніше застосовувати пропускну здатність мережі, так як пакети надсилаються лише тим пристроям, котрі дійсно потребують їх [16].

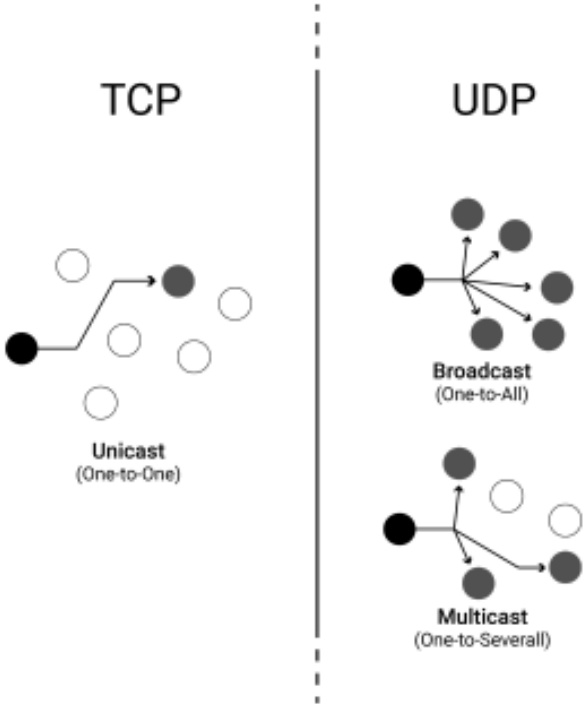

Рисунок 1.5 – Методи Broadcast та Multicast UDP протоколу та Unicast TCP протоколу

1.3. Порівняльний аналіз методів передачі даних

За результатами огляду існуючих RM протоколів можна зробити висновок, що програми, котрі працюють з протоколом PGM, не отримують інформацію про те, чи отримали всі приймачі дані, PGM за рахунок механізму NACK і своєї ієрархії є відмінно масштабованим, забезпечує повністю надійну доставку даних, в ньому відсутня можливість роботи з односторонніми каналами та переривчастими потоками даних, також рекомендована часткова підтримка з боку мережі. PGM найкраще підходить для передачі файлів.

M/TCP ж задовольняє потреба додатків отримання інформації у тому, що це

приймачі отримали дані з допомогою використання механізму ACK, але здатний працювати з дуже невеликою кількістю приймачів, і цей протокол здатний працювати з переривчастими потоками даних. M/TCP потрібна повна підтримка на мережевих вузлах. M/TCP найкраще підходить для текстових чатів або програм, що передають Push –сповіщення [17].

У табл. 1.1 наведено порівняння вищезгаданих протоколів та протоколу UDP.

Позначення стовпців 1 – 6 мають такий зміст:

1- Інформація про те, що всі приймачі отримали дані

2- Масштабованість

3- Цілком надійна доставка даних

4- Обмежена за часом доставка даних

5- Підтримка односпрямованих каналів

6- Вимога підтримки з боку мережі

Таблиця 1.1

Порівняння різних транспортних протоколів

Протокол	Вимоги додатків				Обмеження з боку мережі		Типові програми
	1	2	3	4	5	6	
UDP	-	+	-	+	+	-	Аудіо-відео трансляції, IPTV
PGM	-	+	+	-	-	+/-	Передача файлів
M/TCP	+	-	+	-	-	+	Текстові чати, передача Push-повідомлень
*	+	+	+	-	+	-	Передача керуючих команд у реальному часі

* - Неіснуючий, на даний момент, протокол (НБТП)

1.4. Висновки до розділу

Розглянуто особливості предметної області. Виходячи з проведеного аналізу наявних транспортних протоколів, можна зробити висновок, що існуючі багатоадресні реалізації вирішують вузьке коло завдань.

В даний момент відсутній протоколу, котрий здатний працювати з переривчастим потоком даних, що не вимагає підтримки з боку мережі та добре масштабується. Прикладом використання такого протоколу є додаток, що передає управляючі команди величезній кількості приймачів.

РОЗДІЛ 2
ТЕОРЕТИЧНА ЧАСТИНА

2.1. Архітектура мережі

Нехай відправники та одержувачі будуть підключені до магістральної мережі через локальні комутатори доступу, прямо чи опосередковано через вузли доступу (рис. 2.1).

Рисунок 2.1 – Модель мережі

Багатоадресне дерево, корінням якого є відправник, а вершинами є всі одержувачі, визначається на мережному рівні та називається глобальним [18].

Також є локальні багатоадресні дерева, які є частинами глобального. Глобальне багатоадресне дерево показано суцільними лініями на рис. 2.2. Одержувачі в локальній області, обслуговуються L_i, позначаються як $R_{i,j}$. Локальні комутатори доступу є одержувачами [19].

Єдина послуга, котру протокол очікує від основної мережі, це встановлення дерева багатоадресної передачі від відправника до одержувачів. Однак резервування ресурсів не є необхідним для належного функціонування НБТП. Його функція полягає в тому, щоб послідовно доставляти пакети від відправника до одержувачів уздовж дерева багатоадресної передачі, незалежно від того, як дерево створено та як розподілено ресурси.

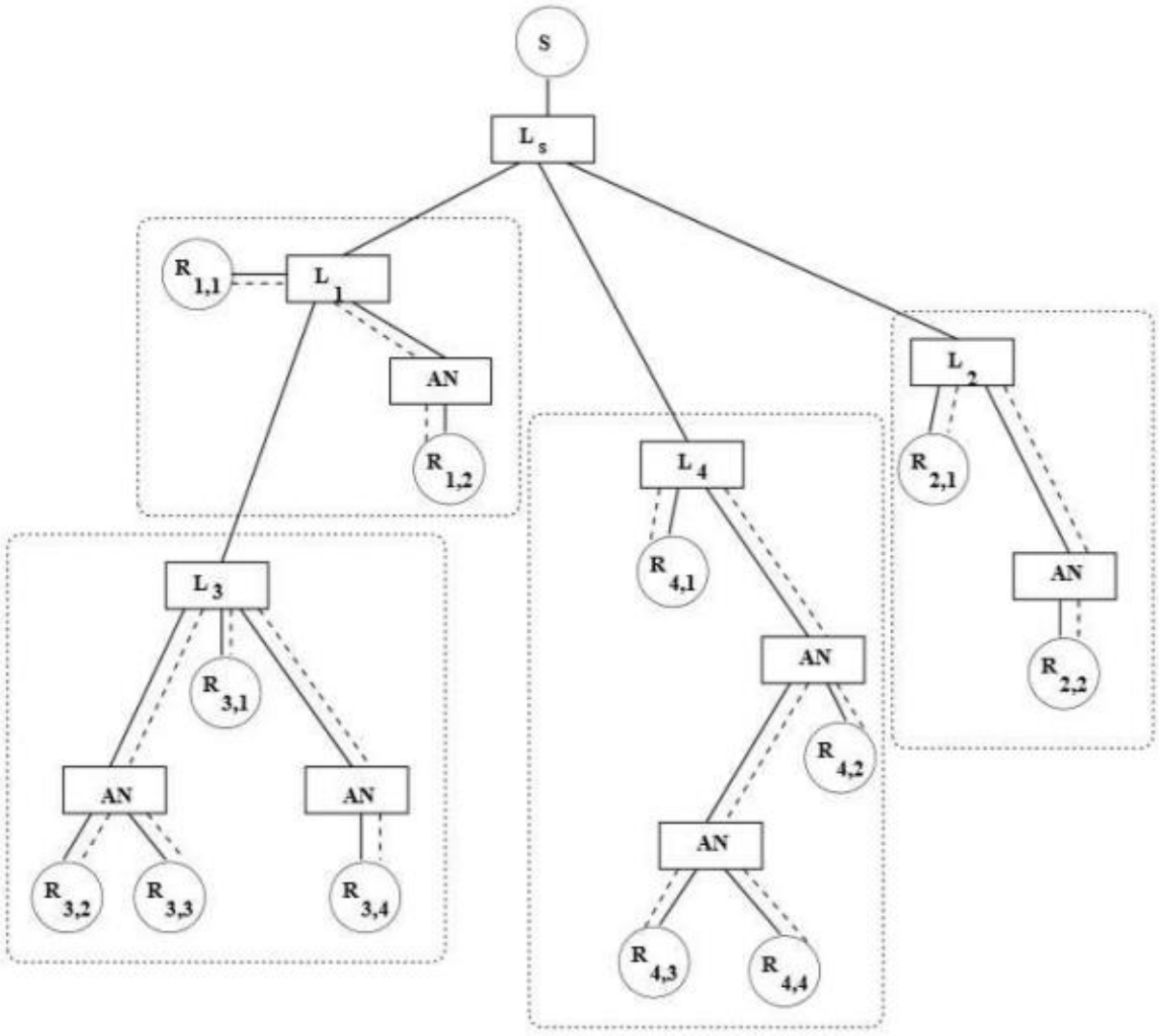

Рисунок 2.2 – Глобальне багатоадресне дерево

Варто навести деякі припущення, зроблені при проектуванні НБТП.

1. Приймачі можуть бути згруповані в локальні області на основі їх близькості в мережі. Наприклад, якщо припускається ієрархічна схема адресації, тоді групування приймачів може бути виконано на основі коду міста в Інтернет-протоколі. У мережі можливе групування приймачів за допомогою поля TTL IP-пакетів.

2. Дерево багатоадресної адресації, що ґрунтується на відправнику й охоплює всіх одержувачів, налаштовано на мережевому рівні (рівень ATM у контексті мереж банкоматів). Це називається глобальне багатоадресне дерево, щоб відрізняти його від локального багатоадресного дерева, яке є його частиною.

Дерево глобальної багатоадресної передачі показано суцільними лініями на рис. 2.2. Приймачі в локальній області, які обслуговується, позначаються $R_{i,j}$ Потрібно зауважити, що L_i позначає комутатор локального доступу для і-ї області, а не приймач.

3. Протокол може бути застосованим для надійної групової передачі «точка-багато точок». Для схеми «багато точок – до багатьох» надійна багатоадресна розсилка можлива, якщо налаштовані дерева багатоадресної розсилки кожного відправника.

2.2. Опис розроблюваного протоколу

НБТП забезпечує послідовну «оптову» доставку без втрат даних від одного відправника групі одержувачів.

Відправник забезпечує надійну доставку шляхом вибіркової повторної передачі втрачених пакетів у відповідь на запит повторної передачі від одержувачів. Якщо кожен приймач надсилає свій статус (ACK/NACK) до відправника, це призводить до дроселювання відправника, що є добре відомою проблемою імплозії ACK [20].

Крім того, якщо деякі приймачі розташовані далеко від відправника, і

відправник повторно передає втрачені пакети цим віддаленим приймачам, наскрізна затримка значно збільшується, а пропускна здатність значно зменшується.

НБТП був розроблений, щоб полегшити проблему імплозій ACK за допомогою ієрархічного підходу на основі дерева. Головною особливістю даного НБТП є угруповання приймачів у локальні області та використання DR у кожній локальній області. Незважаючи на те, що відправник багатоадресно передає кожен пакет усім одержувачам, що знаходяться в глобальному багатоадресному дереві, тільки виділені приймачі відправляють повідомлення відправнику із зазначенням, які пакети вони отримали та які пакети вони не отримали. Одержувачі локальної області передають їх у відповідний DR. Варто зазначити, що DR не консолідує повідомлення про стан приймачів у своїй локальній області, але використовує ці повідомлення для виконання локальних передач повторним одержувачам, значно зменшуючи наскрізну затримку. Таким чином, відправник бачить лише DR, а DR бачить лише одержувачів у своїй локальній області. Обробка повідомлень про стан доставки даних відбувається виключно між відправником та виділеними приймачами, що дозволяє уникнути проблеми імплозій ACK.

На рис. 2.2 приймачі $R_{i,1}$ вибрано як DR для групи $R_{i,j}$, у локальних областях, який обслуговує L_i. Дерево локальної багатоадресної розсилки, що має коріння в $R_{i,1}$, визначається як частина глобального багатоадресного дерева, що охоплює $R_{i,j}$ у локальному регіоні, що обслуговується L_i. Локальні багатоадресні дерева позначені пунктирними лініями.

2.3. Складові алгоритму роботи

2.3.1 Основні кроки алгоритму

1. Відправник здійснює багатоадресну передачу пакетів даних усім одержувачам у глобальному багатоадресному дереві. Ця багатоадресна передача

називається глобальною багатоадресною розсилкою.

2. Кожен виділений приймач надсилає повідомлення з інформацією про доставку у формі пакетів стану з певною періодичністю. Кожен пакет містить інформацію про те, які пакети даних були успішно прийняті виділеними приймачами. На підставі цих повідомлення відправник визначає, які пакети повинні бути передані повторно.

Якщо кількість виділених приймачів, що вимагають повторну передачу пакета, перевищує певний поріг, повторна передача виконується багатоадресно, в іншому випадку приймач повторно передає пакети тільки виділеним приймачам, що запитують. Далі кожен виділений приймач багатоадресно розсилає дані у межах своєї локальної області.

3. Кожен приймач, котрий не є виділеним, надсилає свій статус своєму RD через рівні проміжки часу. Виділений приймач здійснює повторну локальну багатоадресну передачу пакета, якщо кількість приймачів у його області, які вимагають повторну передачу, перевищує граничне значення; в іншому випадку пакет передається одноадресно для приймачів, які запросили його повторну передачу.

4. Відправник багатоадресно надсилає нові пакети, якщо у вікні є вільне місце.

Відправник ділить дані, які підлягають передачі, пакети даних фіксованого розміру, крім останнього. Пакет даних ідентифікується типом DATA, а тип DATA EOF ідентифікує останній пакет даних. Відправник призначає кожному пакету даних порядковий номер, починаючи з 0.

Одержувач періодично надсилає пакети ACK відправнику або DR. Пакет ACK містить нижній кінець вікна прийому (L) та бітовий вектор фіксованої довжини для отримання розміру вікна, що вказує, які пакети отримані та які пакети втрачені.

У табл. 2.1 перелічені типи пакетів, які у даному протоколі.

Таблиця 2.1

Типи пакетів

Тип	Пояснення
ACK	Пакет підтвердження доставки
ACK TXNOW	Пакет вимоги негайної передачі
DATA	Пакет даних
DATA EOF	Останній пакет даних
RESET	Пакет для розриву з'єднання
RTT MEASURE	Пакет для вимірювання часу прийому-передачі
RTT ACK	ACK для пакету RTT MEASURE
SND ACK TOME	Пакет для вибору AP

2.3.2. Встановлення з'єднання

З'єднання ідентифікується парою кінцевих точок: джерела та призначення. Кінцева точка джерела складається з мережевої адреси відправника та номера порту; кінцева точка призначення складається з адреси багатоадресної групи та номера порту [21]. Кожне з'єднання має набір пов'язаних параметрів з'єднання (табл. 2.2).

НБТП передбачає наявність диспетчера сеансів, який відповідає за надання відправнику та одержувачу відповідних параметрів з'єднання. Протокол використовує стандартні значення для будь-якого параметра з'єднання, якщо він не вказаний явно.

Як тільки диспетчер сеансів надав відправнику та одержувачам інформацію про сеанс, одержувачі ініціалізують блок управління з'єднанням та залишаються у непідключеному стані; тим часом відправник розпочинає передачу даних. Отримавши пакет даних від відправника, одержувач переходить із непід’єднаного стану до під’єднаного стану. У такому стані приймачі періодично генерують АСК,

підтримуючи з'єднання активним.

Таблиця 2.2

Параметри з’єднання

Параметр	Пояснення
Wr	розмір вікна прийому у пакетах
Ws	розмір вікна відправлення у пакетах
Tdally	затримка після надсилання останнього пакета
Tretx	затримка після надсилання останнього пакета
Trtt	інтервал часу для вимірювання часу прийому-передачі
Tsap	інтервал часу для відправки SND ACK TOME
Tsend	інтервал часу для надсилання пакетів даних
Tack	інтервал часу для надсилання пакетів стану
Packet	розмір пакета даних у октетах
Cache	розмір пакета даних у октетах
MCASTthresh	поріг багатоадресної повторної передачі

Під час роботи протоколу відправник не має інформації про отримувачів. Одержувачі можуть вступати в multicast- групу або залишати її без інформування відправника [22]. Отже, мета цього НБТП – забезпечити надійну доставку поточним учасникам сеансу багатоадресної передачі. Оскільки відправник не має інформації про одержувачів, завершення сеансу базується на таймері. Після того як відправник передає останній пакет даних, запускається таймер, який закінчується через секунду. (Виділений приймач також запускає таймер після коректного отримання всіх пакетів даних.) Після закінчення таймера, відправник видаляє всю інформацію про стан з'єднання. Повідомлення АСК від отримувача скидає таймер до початкового значення.

Кожен одержувач видаляє свій блок управління з'єднанням та припиняє

надсилання ACK після коректного отримання всіх пакетів даних. DR веде себе як звичайний одержувач, за винятком того, що він видаляє свій блок управління з'єднанням тільки після закінчення часу таймера.

Оскільки період часу між передачею послідовних ACK від одержувачів набагато менше, ніж Tdally, відправник припускає, що всі приймачі коректно прийняли весь обсяг інформації, або сталася позаштатна ситуація. Можливі приклади позаштатних ситуацій: вихід із ладу мережевих пристроїв, вихід приймачів із групи [23]. Під час виникнення цих ситуацій відбувається розрив з'єднання шляхом надсилання пакету RESET. Наприклад, коли протокол визначає, що відправна програма завершила роботу до того, як передача даних була успішно завершена, він використовує пакет RESET для того, щоб інформувати всі приймачі про закриття з'єднання.

2.3.3. Об'єкти протоколу

НБТП має три основні об'єкти: Sender (Відправник), (DR) Призначений одержувач, Receiver (Одержувач) (див. рис. 2.3)

Об'єкт Sender має метод T_CONTROLLER, який вирішує, чи відправник повинен передавати нові пакети (використовуючи метод Tx), проводити повторну передачу втрачених пакетів (використовуючи метод RTx) або розсилати з інформацією про те, що даний відправник є обробником повідомлень ACK (використовуючи метод AP_A). Метод STATUS_PROCESSOR обробляє повідомлення ACK від одержувачів і оновлює відповідні структури даних

Також об'єкт Sender має таймери T_Sent, T_Retx і T_Sap , що регулюють роботу методів Tx, RTx і AP_A відповідно. Таймер T_Dally використовується для розриву з'єднання.

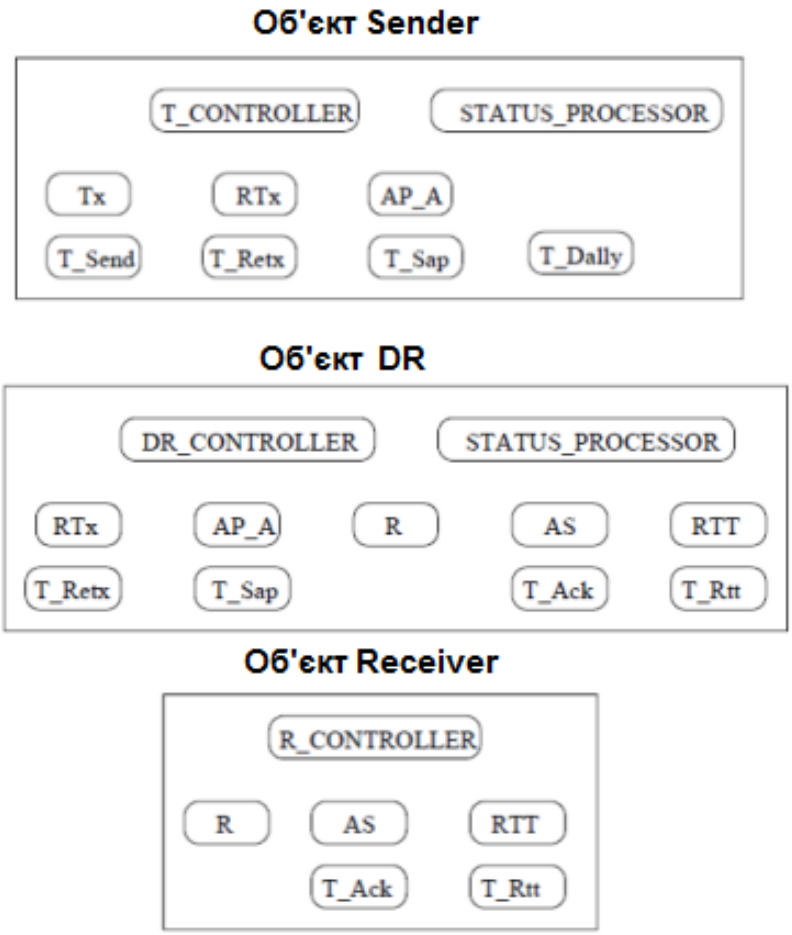

Рисунок 2.3 – Об’єкти протоколу

Метод R_CONTROLLER об'єкта Receiver відповідає за доставку даних приймаючому додатку (використовуючи метод R), за відправлення повідомлення ACK (використовуючи метод AS) або за відправку пакетів вимірювання часу прийому-передачі (використовуючи метод RTT) для динамічного обчислення затримки кругової між приймачем і його обробником повідомлень ACK.

Слід зазначити, що об'єкт Receiver має два таймери: T_Ack та T_Rtt, даний таймери контролюють роботу методів AS та RTT відповідно. Метод R не управляється таймером, а активується в той момент, коли приймаючий додаток запитує пакети.

Фактично об'єкт DR є комбінацією об'єктів Receiver та Sender.

2.3.4. Передача даних

Відправник відправляє багатоадресно пакети даних з інтервалом певним параметром T_{send}. Кількість пакетів, переданих за кожен інтервал, зазвичай залежить від доступного простору у вікні надсилання. Відправник протягом T_{send} може відправити число пакетів, котре не перевищує число W_s . Таким чином, максимальну швидкість передачі MaxTR можна розрахувати за формулою (2.1):

$$MaxTR = \frac{W_s * Packet\_Size}{T_{send}} \quad (2.1)$$

2.3.5. Підтвердження доставки

Приймачі за допомогою методу AS надсилають пакети ACK з певною періодичністю, вказуючи статус вікна прийому. Приймачі використовують бітовий масив розміром W_r біт (розмір вікна прийому) для запису інформації про коректно прийняті пакети, що зберігаються в буфері (рис. 2.4)

Кожен біт відповідає одному слоту пакета буфері приймача. Значення «1» означає, що пакет прийнято коректно. Наприклад, на рис. 2.4 показано вікно прийому з восьми пакетів; пакети 16, 17 та 19 отримані правильно і зберігаються в буфері. Коли одержувач відправляє ACK відправнику або виділеного приймача, він включає значення лівого краю вікна прийому L і бітовий вектор.

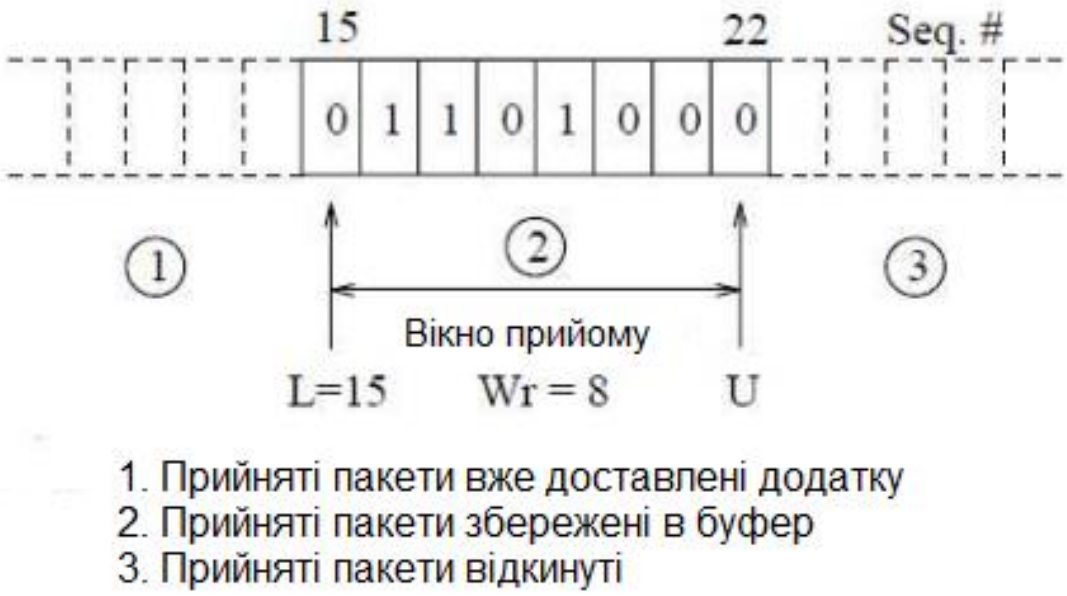

Рисунок 2.4 – Приклад вікна прийому

Слід зазначити, що приймач передає пакети додатку у послідовності. Наприклад, якщо приймач отримує пакет 15 від відправника і не отримує пакет 18, він може доставити пакети 15, 16 та 17 додатку та збільшити значення L до 18.

2.3.6. Вимірювання кругової затримки та розрахунок $T_{аск}$

Приймачі періодично надсилають повідомлення АСК. Якщо дані повідомлення надсилаються надто часто, обробник повідомлень АСК може припинити повторну передачу одного й того самого пакета, не знаючи, чи той прийнятий одержувачами правильно. Щоб запобігти надлишковим повторним передачам, кожен приймач динамічно вимірює час прийому-передачі між собою та своїм обробником повідомлень АСК, використовуючи пакет RTT_MEASURE. На основі отриманих даних, кожен приймач обчислює T_{ack} інтервал між передачею послідовних повідомлень АСК.

Приймач відправляє перший пакет RTT_MEASURE одразу після встановлення з'єднання. Наступні пакети RTT_MEASURE відправляються з фіксованим інтервалом T_{rtt}. Щоб виміряти час прийому передачі, приймач

поміщає локальну тимчасову мітку в пакет RTT_MEASURE і відправляє пакет своєму обробнику повідомлень ACK. Коли обробник отримує пакет RTT_MEASURE, він негайно змінює тип пакета на RTT_ACK і відправляє пакет назад до приймача. Після отримання пакету RTT_ACK приймач розраховує час прийому-передачі як різницю між часом, коли пакет RTT_ACK був прийнятий і тимчасовою міткою, що зберігається в ньому.

2.3.7. Обробка ACK та повторні передачі

Відправник або виділений приймач обробляє повідомлення ACK від одержувачів у локальній області. На основі отриманих повідомлень ACK від приймачів, обробник може ідентифікувати втрачені пакети, які слід передати повторно. Один або кілька одержувачів можуть втратити той самий пакет. Обробник повідомлень ACK визначає, чи повинен втрачений пакет бути повторно переданий з використанням одноадресної або багатоадресної розсилки. Для цієї мети існують два параметри: T_{retx} та $MCAST_{tresh}$, а також черга повторної передачі. Якщо повідомлення ACK містить запит на повторну передачу, номер послідовності пакета, що запитується, додається в чергу повторної передачі. Черга повторної передачі містить: номер послідовності, пакети, які повинні бути передані повторно, лічильник C містить число, що дорівнює кількості приймачів, які повинні повторно отримати цей пакет, таблицю адрес приймачів, які запитують цей пакет і покажчик на наступний елемент черги. Після закінчення інтервалу T_{retx} обробник повідомлень ACK (за допомогою методу RTx) поміщає елемент у чергу повторної передачі. Якщо значення лічильника перевищує значення $MCAST_{tresh}$, відправник або виділений приймач розсилає втрачені пакети багатоадресно, в іншому випадку, приймач, використовуючи таблицю адрес, відсилає пакети одноадресно.

Відправник використовує три змінні swin_lb, send_next та avail_win для управління вікном відправки. Змінна swin_lb зберігає нижню межу вікна

відправки, send_next вказує на наступний номер послідовності, який буде використовуватися під час відправлення пакетів даних, avail_win - це доступний розмір вікна для відправки даних. Відправник збільшує значення send_next і зменшує значення avail_win після відправлення даних. Коли повідомлення АСК, що підтверджують отримання пакетів з номером послідовності рівним swin_lb, приймаються, swin_lb збільшується так само, як і avail_win. Ілюстрацію роботи механізму вікна відправки можна бачити на рис. 2.5.

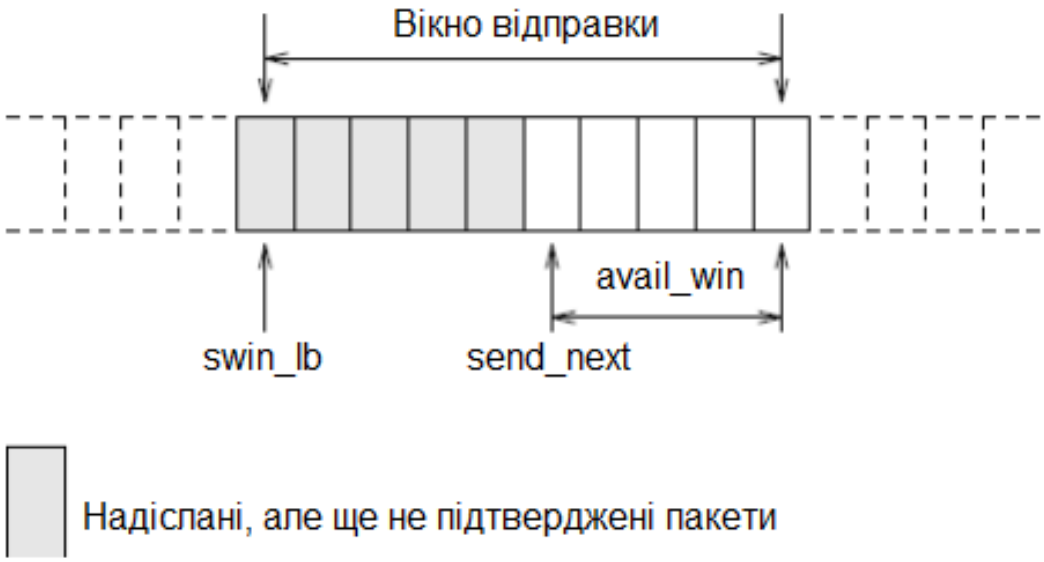

Рисунок 2.5 – Приклад вікна відправлення

Щоб визначити, скільки нових пакетів має бути передано в наступному інтервалі відправки, відправник обчислює найменше значення L (L_{min}) серед значень L, прийнятих у повідомленнях АСК у проміжок часу T_{send}. Якщо L_{min} більше, ніж swin_lb, значення avail_win збільшується на число рівне L_{min}- swin_lb і swin_lb стає рівним L_{min} . Значення swin_lb більше не зменшується. Якщо приймач відправляє повідомлення АСК зі значенням L, меншим ніж swin_lb, дані повідомлення ігноруються.

Пізніше під’єднання приймачів. НБТП дозволяє одержувачам приєднуватись у будь-який момент поточного сеансу. Приймачу, який

приєднується після початку передачі даних, потрібно передати пакети, які він не отримав. Крім того, деякі приймачі можуть «відставати» через різні причини, такі як перевантаження мережі. Існує дві функції, які, працюючи в сукупності, дозволяють приймачам, що «відстали», отримати відсутні дані:

— запит негайної передачі;

— кешування даних у відправнику та у виділених приймачах.

Перша працює так. Коли приймач приєднується пізно, він починає отримувати багатоадресні пакети від відправника, і, переглядаючи номер послідовності даних пакетів, він може відразу виявити, що пропустив більш ранні пакети. У цей момент приймач відправляє пакет ACK_TXNOW, щоб запросити свого виділеного приймача або відправника негайну передачу попередніх пакетів.

Пакет ACK_TXNOW відрізняється від пакета АСК лише полем "тип пакета". Коли обробник повідомлень АСК приймає пакет ACK_TXNOW від приймача, він перевіряє бітовий масив V і негайно передає пропущені пакети або пакет приймачеві, використовуючи одноадресну передачу.

Кешування даних. НБТП дозволяє одержувачам приєднуватися до поточного сеансу в будь-який час і при цьому надійно отримувати всі дані. Однак ця гнучкість не безкоштовна. Для забезпечення працездатності вищеописаної функції відправник та виділені приймачі повинні буферизувати файл протягом всієї сесії багатоадресної передачі даних. Це дозволяє приймачам вимагати повторну передачу будь-яких раніше надісланих даних у своїх обробників повідомлень АСК.

У НБТП використовується дворівневий механізм кешування. Найновіші пакети даних кешуються в пам’яті, а решта зберігаються на диску.

2.3.8. Вибір призначених приймачів та формування локальних областей

Протокол передбачає, що є деяка інформація про приблизне місцезнаходження одержувачів, і на основі цієї інформації певні сервери або

приймачі з найменшою ір-адресою в підмережі вибираються як DR.

Далі кожен DR і відправник періодично відправляють пакет SEND_ACK_TOME приймачам, у якому для поля TTL задано певне значення. Приймач вибирає свого DR за пакетом SEND_ACK_TOME з найменшим значенням TTL. Таким чином, локальні області формуються навколо кожного DR.

2.3.9. Управління потоком

Простий віконний механізм керування потоком не є достатнім для НБТП в середовищі Інтернет. Основна причина полягає в тому, що в моделі багатоадресної розсилки в Інтернеті одержувачі можуть приєднатися або залишити сеанс багатоадресної передачі, не повідомляючи відправника. Таким чином, відправник не знає, хто одержувачі в будь-який момент протягом тривалості сеансу багатоадресної передачі.

Тому, якщо при розробці протокол транспортного рівня для забезпечення гарантованої доставки пакетів даних усім поточним учасникам багатоадресного сеансу, не знаючи явно учасників, потрібна інша техніка для керування потоком. Варто зауважити, що якби НБТП використовував простий віконний механізм керування потоком, то відправник мав би знати, чи всі DR на рівні 1 отримали пакети до того, як вікно розгорнеться. Однак відправник може не знати, скільки DR рівня 1 є, оскільки базове дерево багатоадресної розсилки може змінитися, і або нові DR можуть додаватися до дерева багатоадресної розсилки динамічно, або старі DR можуть залишитися! А це вже погано!

Для вирішення цією ситуації, відправник працює в циклі [24]. Відправник передає вікно, повне нових пакетів, у першому циклі, а на початку наступного циклу він оновлює вікно надсилання та передає стільки нових пакетів, скільки є вільного місця для його вікна надсилання. Оновлення вікна виконується наступним чином. Замість того, щоб переконатися, що кожен DR рівня 1 отримав пакети, відправник переконається, що всі DR, які надіслали повідомлення про

стан протягом певного проміжку часу, успішно отримали відповідні пакети, перш ніж перейти до нижнього кінця свого вікна надсилання. Зауважте, що просування вікна надсилання не означає, що відправник відкидає пакети поза вікном. Пакети все ще зберігаються в кеші для відповіді на запити повторної передачі. Крім того, відправник ніколи не передає більше, ніж повне вікно пакетів протягом фіксованого інтервалу, таким чином обмежуючи максимальну швидкість передачі до визначеної формулою (2.1). Таким чином, цю схему керування потоком можна назвати віконним керуванням потоком на основі швидкості.

2.3.10. Уникнення заторів

НБТП надає механізми, щоб уникнути переповнення вже перевантаженої мережі новими пакетами, не погіршуючи ситуацію. Схема, яка використовується в протоколі для виявлення перевантаження, описана нижче.

НБТП використовує запити на повторну передачу від приймачів як індикацію можливого перевантаження мережі. Відправник використовує вікно перевантаження cong_win, щоб зменшити швидкість передачі даних у разі перевантаження. Під час T_{send} відправник обчислює кількість ACK, N, із запитом на повторну передачу. Якщо перевищує порогове значення CONGtresh, cong_win встановлюється на одиницю. Оскільки відправник завжди обчислює придатне для використання вікно надсилання як Min(avail_win, cong_win), значення одиниці зменшує швидкість передачі даних щонайбільше до одного пакета даних на T_{send}, якщо avail_win не дорівнює нулю. Якщо N не перевищує CONGthres під час T_{send}, відправник збільшує cong_win на одиницю, доки cong_win не досягне Ws. Процедура встановлення cong_win до одиниці та лінійного збільшення cong_win називається повільним запуском і використовується в реалізації стандартного TCP. Відправник починає повільно чекати на отримання ACK від далеких одержувачів.

2.4. Багаторівнева ієрархія

НБТП був описаний раніше як дворівнева система, в якій відправник здійснює групову розсилку всім одержувачам і DR; і DR повторно передає втрачені пакети одержувачам у відповідних локальних областях. Однак обмеження дворівневої ієрархії очевидні з точки зору масштабованості, і багаторівнева ієрархія є бажаною.

Необхідно згадати, що кожен DR періодично надсилає пакети SEND ACK TOME уздовж дерева багатоадресної розсилки, і кожен одержувач вибирає DR, чий пакет SEND ACK TOME має найбільше значення TTL. Крім того, зауважте, що кожен DR також є приймачем. Таким чином, якщо кожен DR ігнорує власні пакети SEND ACK TOME, він вибере DR, що знаходиться найменше вгорі від себе, як його DR, і надсилатиме свої повідомлення про статус цьому DR під час сеансу багатоадресної передачі. Рис. 2.6 ілюструє цю ідею.

По суті, якщо на шляху від відправника до групи одержувачів є n DR, і ці DR мають різну кількість переходів від відповідних одержувачів, то в n-рівневій ієрархії буде n локальних областей, так що DR n-го рівня надішле свій статус до DR рівня n-1, DR рівня n-1 відправить свій статус до DR рівня n-2 і так далі, доки DR рівня 1 не надішле свій статус для відправника (DR на рівні 0).

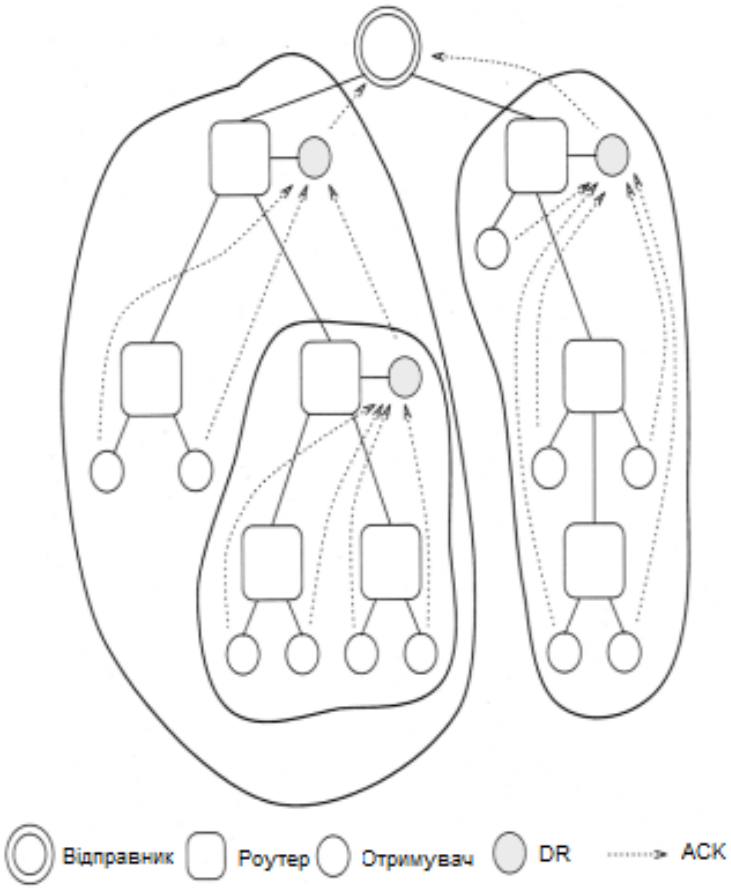

Рисунок 2.6 – Багаторівнева ієрархія DR

Тобто DR на і-му рівні діє як приймач і-1-го рівня для всіх і (i=n,...., 1), де нульовий рівень відноситься до глобального дерева багатоадресної розсилки, що має корінь у відправника.

2.5. Висновки до розділу

У цьому розділі розроблено алгоритм роботи НБТП, відповідно до вимог, визначених у розділі 1.

Наведено модель мережі та глобальне багатоадресне дерево. Було описано основні механізми роботи розроблюваного протоколу: алгоритм функціонування, встановлення з’єднання, складові елементи, підтвердження доставляння даних,

опрацювання ACK та рецидивні передачі, формування локальних областей, управління потоком даних та мпосіб уникнення заторів.

Також докладно описано реалізацію багаторівневої ієрархвії в НБТП.

РОЗДІЛ 3
ПРАКТИЧНА ЧАСТИНА

3.1. Опис віртуального стенду

Віртуальний стенд є набіом із чотирьох вузлів, з'єднаних так, як показано на рис. 3.1. Кожен вузол є віртуальною машиною з операційною системою Ubuntu 12.04.5 LTS. Вузли S, R3, R4 мають один мережевий інтерфейс (рис. 3.1).

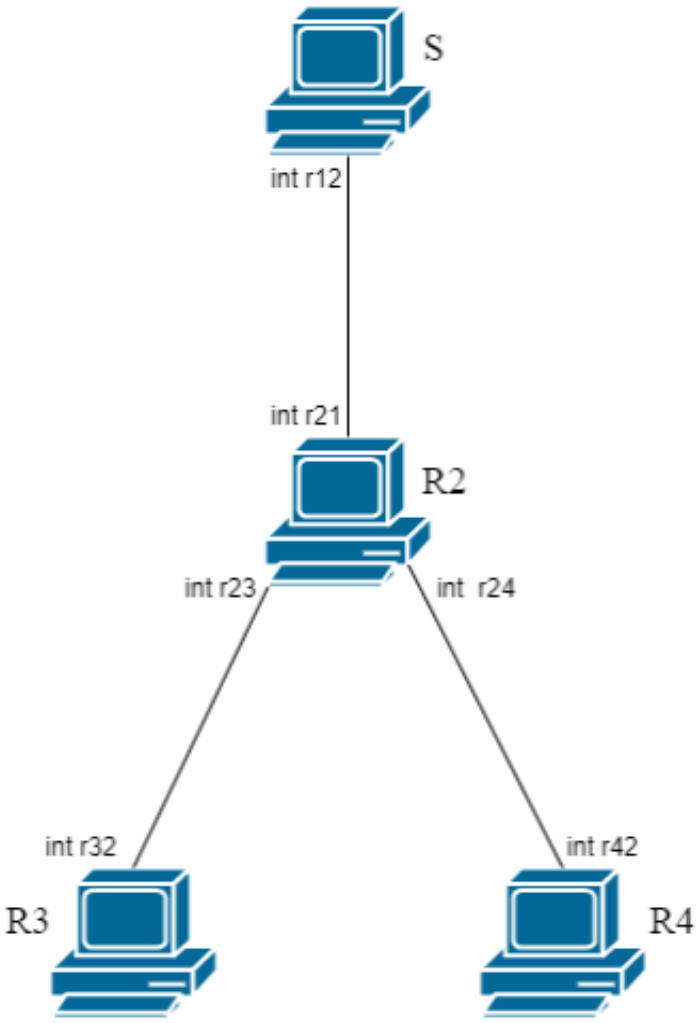

Рисунок 3.1 – Модель віртуального стенду

На рис. 3.2 відображено модель у середовищі моделювання.

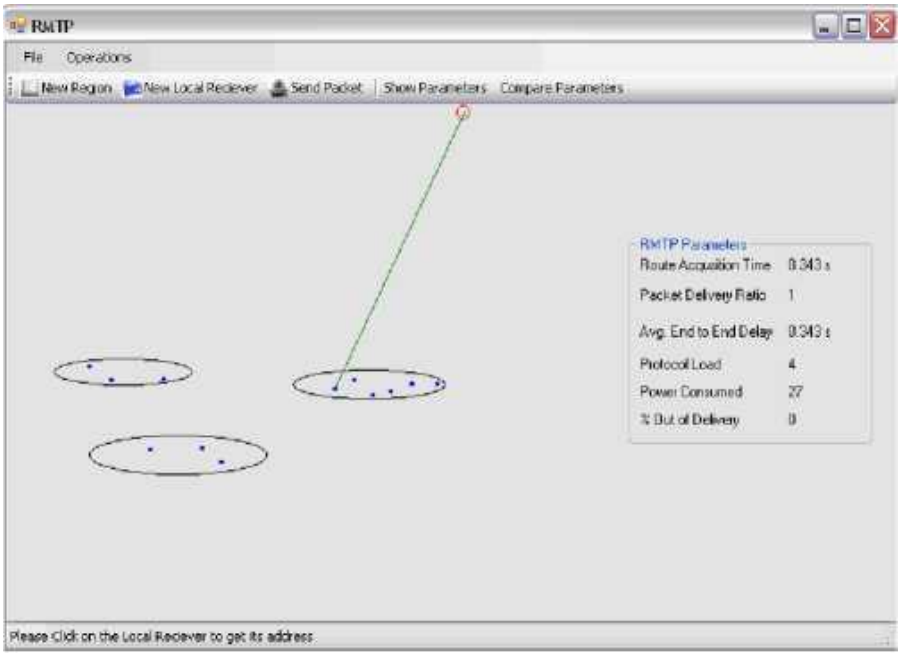

Рисунок 3.2 – Моделювання стенду

На рис. 3.3 наведено скрін-шот вікна з інформацією щодо поточного стану інтерфейсу r 32 на вузлі R3.

```
r32       Link encap:Ethernet  HWaddr 16:15:76:c5:84:68
          inet6 addr: fe80::1415:76ff:fec5:8468/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:22 errors:0 dropped:0 overruns:0 frame:0
          TX packets:20 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:1876 (1.8 KB)  TX bytes:1200 (1.2 KB)
```

Рисунок 3.3 – Інформація про стан інтерфейсу r32 на вузлі R3

Як мережа використовується комутований Ethernet. Вузол R2 має три мережеві інтерфейси об'єднаних в мережевий міст за допомогою утиліти iproute2. Це потрібно, оскільки R2 у цій моделі є DR і йому необхідно широкомовно передавати пакети вузлам R3 та R4.

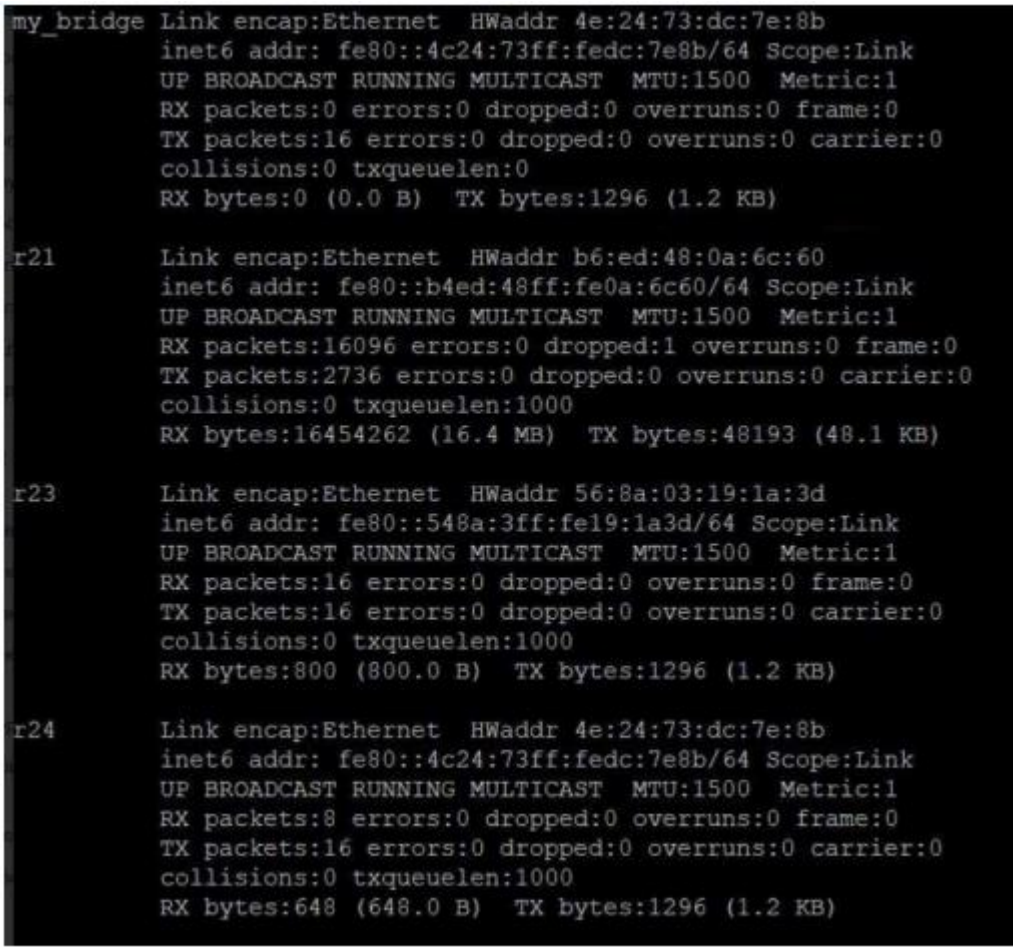

Рисунок 3.4 – Інформація про стан інтерфейсів на вузлі R2

Оскільки реалізація протоколу виконана мовою Python, на всіх вузлах встановлено Python версії 2.3.7. Віртуальні машини працюють у середовищі VMWare ESXi. Для зв'язку використовуються віртуальні канали GigabitEthemet.

3.2. Опис експерименту

Реалізацію протоколу було виконано мовою Python. Для перевірки працездатності реалізації проведено описаний нижче експеримент. Для використання в реалізації був обраний такий інтерфейс програмування додатків як сирий сокет (англ. raw socket), ці сокети дозволяють реалізувати в просторі користувача нові протоколи стека IPv4 і не виставляють будь-яких обмежень у формуванні пакета на відправку.

На всіх вузлах віртуального стенду ініціалізується запуск програми із файлу main.py (рис. 3.5). Залежно від аргументу запуску, вузол працює в одному з трьох режимів:

1. Відправник (аргумент "-s");
2. Приймач (аргумент "-r");
3. DR (аргумент "-d").

```
if __name__ == '__main__':
    interfaces = os.popen("ip a | egrep -o 'r[0-9]+'").read().split('\n')
    print(interfaces)
    interfaces.pop()

    sockets = [None] * len(interfaces)
    for i in range(0, len(interfaces)):
        sockets[i] = socket.socket(socket.AF_PACKET, socket.SOCK_RAW, socket.htons(3))
        sockets[i].bind((interfaces[i], 0))
    print('Switching on ' + ' '.join(interfaces))

    if len(sys.argv) < 2:
        print("Need some arguments")
        sys.exit(1)
    if sys.argv[1] == "-s":
        filename = sys.argv[2]
        sender(sockets[0], filename)
```

Рисунок 3.5 – Фрагмент вмісту файлу main.py

Як тестування роботи алгоритму проводиться передача файлу від відправника S групі приймачів R2-R4. В рамках даного експерименту DR є вузол R2. На цьому вузлі і виконується той основний механізм, котрий гарантує надійну доставку, а саме – здійснюються агрегація підтверджень доставлення пакетів даних від інших приймачів та повторна їх передача у разі втрати сегментів даних (рис. 3.6).

```
def receiver_d(socket_in):
    f = open("received.py", 'wb')
    sender_addr = ""
    src_addr = socket_in.getsockname()[4]
    print "My address :", bytes_addr_to_str(src_addr)
    local_receivers = [src_addr]
    num_receive_frame = 0
    num_err_frame = 0
    max_err_frame = 1
    while True:
        print "Wait frame..."
        raw_data, addr = socket_in.recvfrom(MAX_SIZE)
        dst_addr, src_addr, proto, frame_type, num_packet, is_last, data = parse_frame(raw_data)
        print('\nReceived Frame:')
        print('Destination:\t{} \nSource:\t\t{} \nProto: {} \nType: {} \nNum Frame: {} \nIs Last
        Packet: {}'.format(
            bytes_addr_to_str(dst_addr), bytes_addr_to_str(src_addr), bytes_addr_to_str(proto),
            frame_type, num_packet, is_last))
```

Рисунок 3.6 – Фрагмент коду тестування алгоритму

Окрім того, вузол R2, так само, як і вузли R3 і R4 приймає пакети даних від відправника R1, відсилаючи йому повідомлення про підтвердження доставки. У свою чергу вузли R3 і R4 працюють у режимі звичайного приймача, надсилаючи повідомлення про успішну доставку свого виділеного приймача R2 (рис. 3.7).

```
def receiver(socket_in, rd_address):
    f = open("received.py", 'wb')
    is_last = False
    rd_address = str_addr_to_bytes(rd_address)
    src_addr = socket_in.getsockname()[4]
    proto = b'\x89\x99'
    print "My address :", bytes_addr_to_str(src_addr)
    print "RD address :", bytes_addr_to_str(rd_address)
    rd_frame = struct.pack(HEADER_FORMAT, rd_address, src_addr, proto, TYPE_MSG_RD, 0,
    False)
    socket_in.sendall(rd_frame)
    print('\nSending Frame to RD:')
    print('Destination:\t{} \nSource:\t\t{} \nProto: {} \nType: {} \nNum Frame: {} \nIs Last
Packet:
    {}'.format(
        bytes_addr_to_str(rd_address), bytes_addr_to_str(src_addr), bytes_addr_to_str(proto),
        TYPE_MSG_RD, 0, is_last))
```

Рисунок 3.7 – Демонстрація роботи вузлів

В рамках поточної реалізації DR вибирається вручну, тому вузли R3, R4 заздалегідь володіючи цією інформацією, відправляють вузлу R2 повідомлення про те, що вони знаходяться в його локальній області, R2 у свою чергу реєструє їх і починає обробляти повідомлення про підтвердження доставки.

Вузол S працює в режимі відправника, на даному вузлі відбувається розбиття файлу, що відправляється на сегменти даних і подальша широкомовна відправка даних сегментів одержувачам, також вузол отримує повідомлення про підтвердження доставки і в разі втрати пакетів даних робить повторну передачу (рис. 3.8).

```
def sender(socket_out, filename):
    f = open(filename, "rb")
    data_size = MAX_SIZE - HEADER_SIZE
    src_addr = socket_out.getsockname()[4]
    print "My address :", bytes_addr_to_str(src_addr)
    dst_addr = b'\xff\xff\xff\xff\xff\xff' # FF:FF:FF:FF:FF:FF
    proto = b'\x89\x99'
    print("\nStart file sending...")
    data_parts = []
    data = f.read(data_size)
    while data != "":
        data_parts.append(data)
        data = f.read(data_size)
    if len(data_parts) == 0:
        print("len(data_parts) == 0")
        print("Stop file sending...")
        return
    num_packet = 0
    data = data_parts[num_packet]
```

Рисунок 3.8 – Демонстрація роботи вузла S

3.3. Результати експерименту

Перед початком передачі файлу вузли R3 і R4 відправили повідомлення вузлу R2, зареєструвавшись у локальній зоні (рис. 3.9).

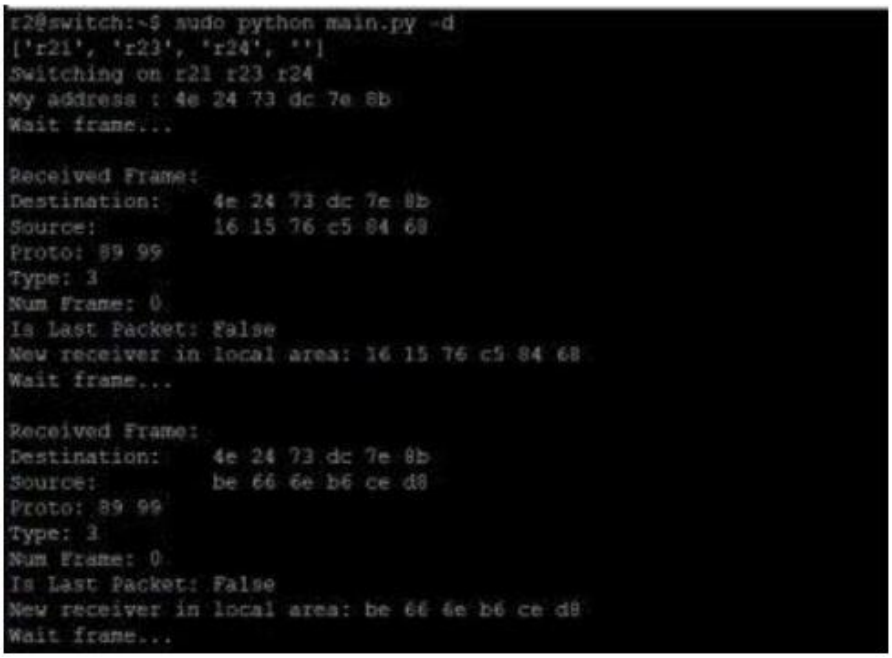

Рисунок 3.9 – Отримання реєстраційних пакетів від вузлів R3 та R4

Після успішної реєстрації вузол S ініціював надсилання пакета даних відправникам. Вузол R2 здійснював пересилання отриманого пакета приймачам R3 і R4, агрегуючи отримані від них повідомлення про доставку і у разі успішної передачі відправляв повідомлення про успішну передачу відправника, і запитував надсилання наступного сегмента (рис. 3.10 – 3.13).

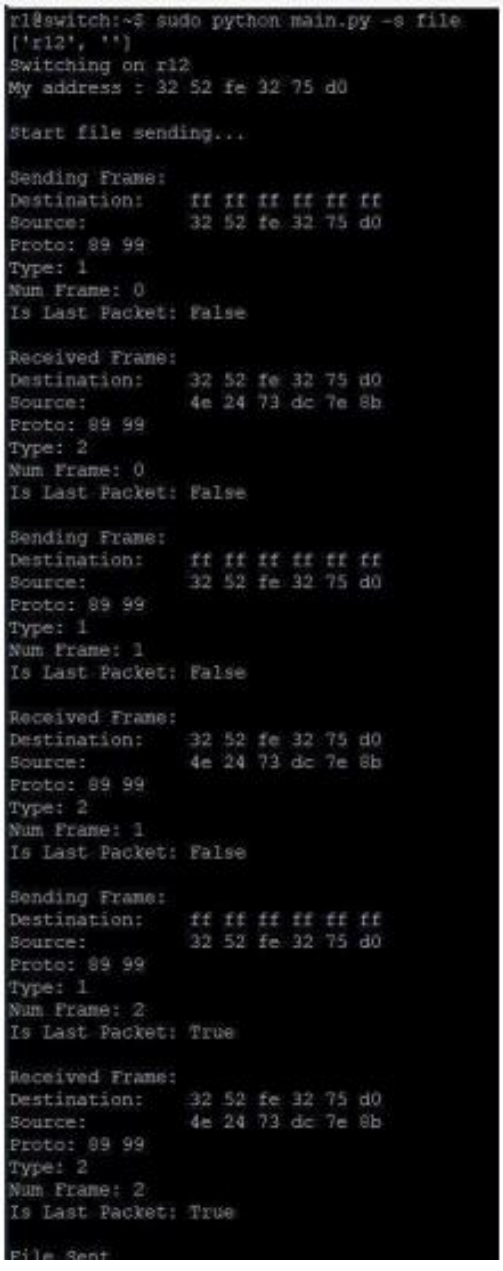

Рисунок 3.10 – Вивід роботи протоколу з вузла R1

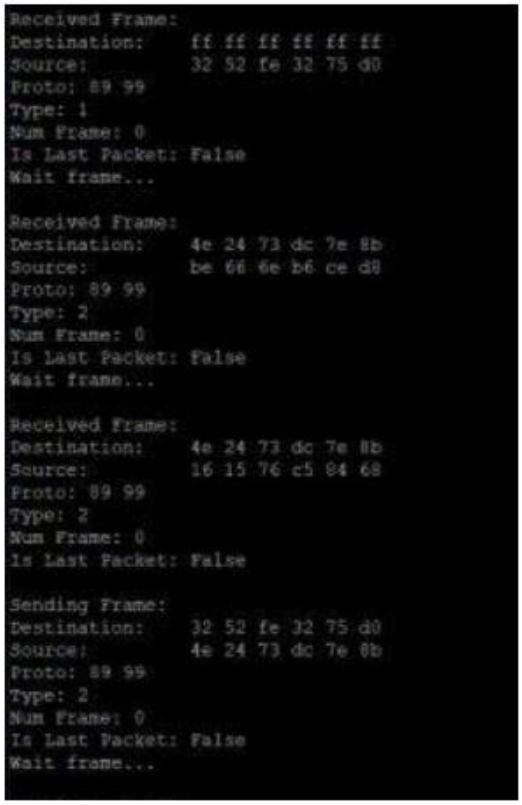

Рисунок 3.11 – Фрагмент виводу роботи протоколу з вузла R2

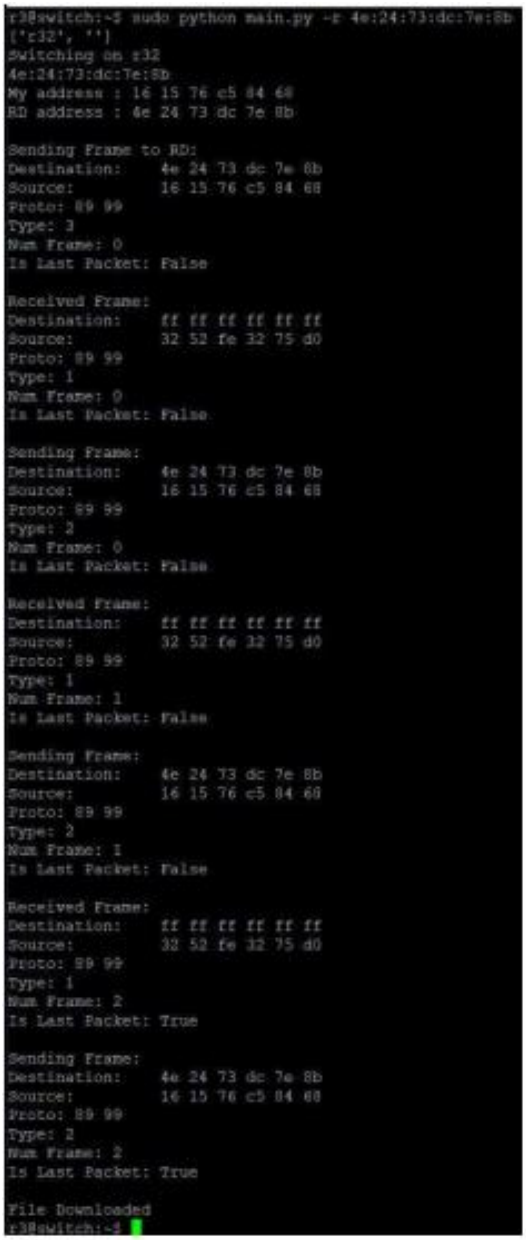

Рисунок 3.12 – Виведення роботи протоколу з вузла R3

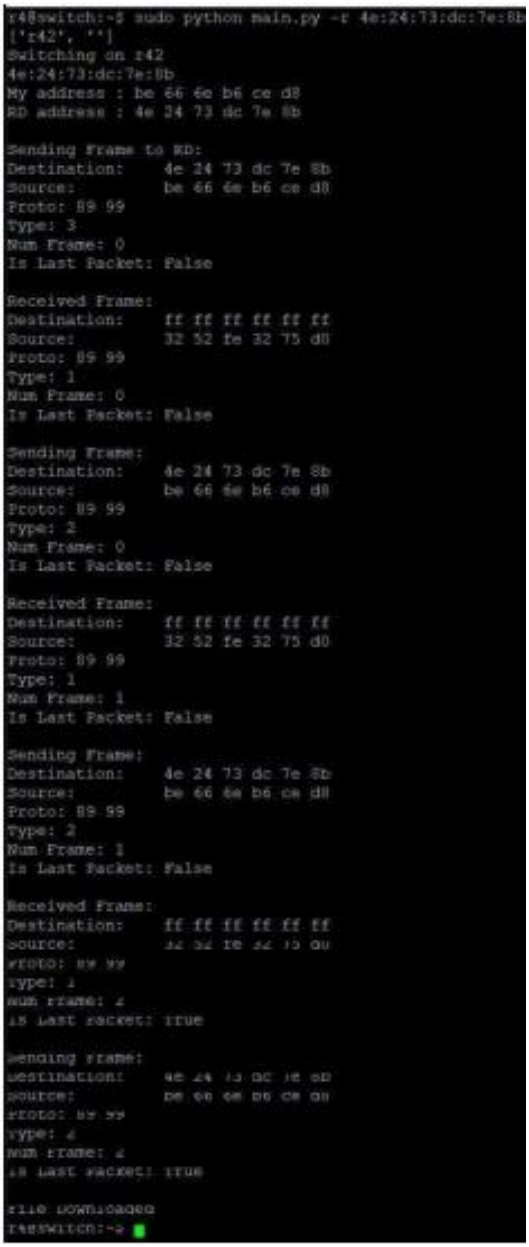

Рисунок 3.13 – Виведення роботи протоколу з вузла R4

В результаті експерименту від відправника S було успішно передано файл, розміром 3 кілобайти вузлам R2, R3 і R4.

Нижче наведена табл. 3.1, що відображає кількість пакетів, надісланих та прийнятих кожним вузлом для передачі файлу в порівнянні з протоколом TCP.

Таблиця 3.1

Кількість пакетів, надісланих та прийнятих кожним вузлом

	TCP		Реалізований протокол	
Пакети / Вузол	Надіслано	Отримано	Надіслано	Отримано
S	15	12	3	3
R2	4	5	3	11
R3	4	5	4	3
R4	4	5	4	3

Виходячи з цих даних, наведених в табл. 3.1, можна зробити висновок, що при використанні TCP кількість трафіку зростає лінійно в залежності від кількості приймачів.

При використанні НБТП навантаження на мережу розподіляється рівномірно, кількість трафіку в мережі скорочується і залежить від кількості локальних зон і кількості приймачів в кожній локальній зоні.

3.4. Висновок до розділу

У цьому розділі було описано віртуальний стенд, у якому відбувається перевірка роботи демонстраційної реалізації описаного у другому розділі алгоритму роботи протоколу.

В результаті аналізу даних, отриманих під час експерименту, підтверджено функціонування основних механізмів протоколу та проведено порівняння роботи поточної реалізації з роботою протоколу TCP.

Проведено дослідження продуктивності фактичної реалізації в Інтернеті. Проект також включає використання періодичних повідомлень про стан і використання механізму вибіркової повторної передачі для підвищення пропускної здатності.

Наведено показник продуктивності реалізації НБТП для передачі даних на різних вузлах та розрахунку параметрів.

РОЗДІЛ 4
ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ

4.1. Охорона праці

Метою кваліфікаційної роботи магістра є розробка НБТП. Оскільки, проведення робіт з розробки та використання алгоритму передбачає застосування комп'ютерної техніки, зокрема ПК та периферійних пристроїв, то обов'язковим є дотримання вимог з охорони праці і техніки безпеки.

Для ефективної і безпечної роботи колективу працівників з розробки ПЗ комп'ютерних систем, в тому числі і фахівців з розробки транспортного протоку, необхідно організувати безпечні умови праці. При цьому керівник організації несе безпосередню відповідальність за порушення нормативно-правових актів з охорони праці [25]. Окрім цього, на робочих місцях працівників необхідно забезпечити дотримання вимог, затверджених Наказом Мінсоцполітики від 14.02.2018 за № 207 «Про затвердження Вимог щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями». Згідно Вимог приміщення, де розміщені робочі місця операторів, крім приміщень, у яких розміщені робочі місця операторів великих ЕОМ загального призначення (сервер), мають бути оснащені системою автоматичної пожежної сигналізації відповідно до цих вимог;

– переліку однотипних за призначенням об'єктів, які підлягають обладнанню автоматичними установками пожежогасіння та пожежної сигналізації, затвердженого наказом Міністерства України з питань надзвичайних ситуацій та у справах захисту населення від наслідків Чорнобильської катастрофи від 22.08.2005 N 161, зареєстрованого в Міністерстві юстиції України 05.09.2005 за N 990/11270 (НАПБ Б.06.004-2005);

– Державних будівельних норм "Інженерне обладнання будинків і споруд. Пожежна автоматика будинків і споруд", затверджених наказом Держбуду

України від 28.10.98 N 247 (далі - ДБН В.2.5-56:2014, з димовими пожежними сповіщувачами та переносними вуглекислотними вогнегасниками.

В інших приміщеннях допускається встановлювати теплові пожежні сповіщувачі. Приміщення, де розміщені робочі місця операторів, мають бути оснащені вогнегасниками, кількість яких визначається згідно з вимогами ДСТУ 4297:2004 «Пожежна техніка. Технічне обслуговування вогнегасників». Загальні технічні вимоги і з урахуванням граничнодопустимих концентрацій вогнегасної рідини відповідно до вимог НАПБ А.01.001-2014. Приміщення, в яких розміщуються робочі місця операторів сервера загального призначення, обладнуються системою автоматичної пожежної сигналізації та засобами пожежогасіння відповідно до вимог ДБН В.2.5-56:2014, ДБН В.2.5-56:2010, НАПБ А.01.001-2014 і вимог нормативно-технічної та експлуатаційної документації виробника. Проходи до засобів пожежогасіння мають бути вільними.

Лінія електромережі для живлення комп’ютера та периферійних пристроїв повинні бути виконаними як окрема групова трипровідна мережа шляхом прокладання фазового, нульового робочого та нульового захисного провідників. Нульовий захисний провідник використовується для заземлення (занулення) електроприймачів. Не допускається використовувати нульовий робочий провідник як нульовий захисний провідник. Нульовий захисний провідник прокладається від стійки групового розподільного щита, розподільного пункту до розеток електроживлення. Не допускається підключати на щиті до одного контактного затискача нульовий робочий та нульовий захисний провідники.

Площа перерізу нульового робочого та нульового захисного провідника в груповій трипровідній мережі має бути не менше площі перерізу фазового провідника. Усі провідники мають відповідати номінальним параметрам мережі та навантаження, умовам навколишнього середовища, умовам розподілу провідників, температурному режиму та типам апаратури захисту, вимогам НПАОП 40.1-1.01-97.

У приміщенні, де одночасно експлуатуються понад п’ять комп’ютерів, на помітному, доступному місці встановлюється аварійний резервний вимикач, який може повністю вимкнути електричне живлення приміщення, крім освітлення. Комп’ютери повинні підключатися до електромережі тільки за допомогою справних штепсельних з'єднань і електророзеток заводського виготовлення.

У штепсельних з'єднаннях та електророзетках, крім контактів фазового та нульового робочого провідників, мають бути спеціальні контакти для підключення нульового захисного провідника. Їхня конструкція має бути такою, щоб приєднання нульового захисного провідника відбувалося раніше, ніж приєднання фазового та нульового робочого провідників. Порядок роз’єднання при відключенні має бути зворотним. Не допускається підключати комп’ютери до звичайної двопровідної електромережі, в тому числі – з використанням перехідних пристроїв. Електромережі штепсельних з'єднань та електророзеток для живлення комп’ютерної техніки повинні бути виконаними за магістральною схемою, по 3-6 з'єднань або електророзеток в одному колі. Штепсельні з'єднання та електророзетки для напруги 12 В та 42 В за своєю конструкцією мають відрізнятися від штепсельних з'єднань для напруги 127 В та 220 В. Штепсельні з'єднання та електророзетки, розраховані на напругу 12 В та 42 В, мають візуально (за кольором) відрізнятися від кольору штепсельних з'єднань, розрахованих на напругу 127 В та 220 В.

При підвищенні ефективності контролю доступу в приміщення, де для забезпечення безпеки мешканців, співробітників і збереження майна використовуються ДС, важливим, з точки зору охорони праці, є забезпечення достатньої величини природного та штучного освітлення, які визначені у НПАОП 0.00-7.15-18. Організація робочого місця фахівця із дослідження методів та програмно-апаратних засобів оптимізаційних процесів на основі ГА повинна забезпечувати відповідність усіх елементів робочого місця та їх розташування ергономічним вимогам ДСТУ 8604:2015 «Дизайн і ергономіка. Робоче місце для виконання робіт у положенні сидячи. Загальні ергономічні вимоги». Відстань від

екрана до ока фахівців, які працюють за комп’ютером визначається згідно з вимогами ДСанПіН 3.3.2.007-98.

Розміщення принтера або іншого пристрою введення-виведення інформації на робочому місці має забезпечувати добру видимість екрана комп’ютера, зручність ручного керування пристроєм введення-виведення інформації в зоні досяжності моторного поля згідно з вимогами ДСанПіН 3.3.2.007-98.

Таким чином, у результаті аналізу вимог щодо охорони праці користувачів комп’ютерів, визначено особливості організації робочих місць, вимог з електробезпеки, природного та штучного освітлення для ефективної і безпечної роботи фахівців з дослідження та розробка НБТП.

4.2. Планування та порядок проведення евакуації населення з районів наслідків впливу НС техногенного та природного характеру

В умовах неповного забезпечення захисними спорудами в містах та інших населених пунктах, що мають об'єкти підвищеної небезпеки, основним засобом захисту населення є евакуація і розміщення його у зонах, які є безпечними для проживання людей [26].

Евакуації підлягає населення, яке проживає в населених пунктах, що знаходяться у зонах можливого катастрофічного затоплення, можливого небезпечного радіоактивного забруднення, хімічного ураження, в районах виникнення стихійного лиха, аварій і катастроф (якщо виникає безпосередня загроза життю та здоров'ю людей). Залежно від обставин, які склалися на час надзвичайної ситуації, може бути проведено загальну або часткову евакуацію населення тимчасового або безповоротного характеру. Загальна евакуація проводиться за рішенням Кабінету Міністрів України для всіх категорій населення і планується на випадок:

– можливого небезпечного радіоактивного забруднення територій навколо атомних електростанцій (якщо виникає безпосередня загроза життю та здоров'ю людей, які проживають в зоні ураження);

– виникнення загрози катастрофічного затоплення місцевості з чотиригодинним добіганням проривної хвилі.

Часткова евакуація проводиться за рішенням Кабінету Міністрів України у разі загрози або виникнення надзвичайної ситуації техногенного та природного характеру.

Під час проведення часткової евакуації завчасно вивозиться не зайняте у сферах виробництва та обслуговування населення: діти, учні навчальних закладів, вихованці дитячих будинків, разом з викладачами та вихователями, студенти, пенсіонери та інваліди, які утримуються у будинках для осіб похилого віку, разом з обслуговуючим персоналом і членами їх сімей [26].

У сфері захисту населення і територій від надзвичайних ситуацій техногенного та природного характеру евакуація населення планується на випадок:

– аварії на атомній електростанції з можливим забрудненням території; усіх видів аварій з викидом сильнодіючих отруйних речовин; загрози катастрофічного забруднення місцевості :

– лісових і торф'яних пожеж, землетрусів, зсувів, інших геофізичних і гідрометеорологічних явищ з тяжкими наслідкам, що загрожують населеним пунктам.

Загальна евакуація проводиться шляхом вивезення основної частини населення з міст і небезпечних районів усіма видами наявних транспортних засобів на відповідній адміністративній території та виведення найбільш витривалої його частини пішки. Часткова евакуація проводиться з використанням транспортних засобів, що експлуатуються за діючим графіком. На органи виконавчої влади, органи місцевого самоврядування та керівників об'єктів, які проводять евакуацію населення, покладається:

– планування і проведення евакуації працівників та членів їх сімей;

– подання до відповідних транспортних органів розрахунків потреби у транспортних засобах для вивезення працівників і членів їх сімей до безпечних районів;

– контроль за плануванням, підготовкою і проведенням евакуаційних заходів підвідомчими об'єктами;

– визначення та підготовка безпечного району для розміщення евакуйованих працівників і членів їх сімей.

Інші заходи та порядок проведення евакуації викладено у постанові Кабінету Міністрів від 26 жовтня 2001р. № 1432 про затвердження Положення про порядок проведення евакуації населення у разі загрози або виникнення надзвичайних ситуацій техногенного та природного характеру.

У плані евакуації, складовою частиною якого є карта (схема), зазначаються:

– висновки з оцінки обстановки у разі виникнення надзвичайної ситуації;

– порядок оповіщення населення про початок евакуації;

– кількість населення, яке підлягає евакуації, за віковими категоріями;

– терміни проведення евакуації;

– склад евакуаційних органів і терміни приведення їх у готовність;

– кількість населення, яке вивозиться різними видами транспортних засобів окремо і виводиться пішки;

– розподілення об'єктів за збірними евакуаційними пунктами, пунктами посадки, районами (пунктами) розміщення та евакуаційними напрямками; маршрути евакуації;

– райони (пункти) розміщення евакуйованого населення; пункти посадки на транспортні засоби, пункти висадки у безпечному районі, порядок доставки населення з пунктів висадки до районів (пунктів) розміщення;

– заходи щодо організації приймання, розміщення, захисту та життєзабезпечення евакуйованого населення у безпечному районі;

– порядок організації управління і зв'язку.

План приймання і розміщення евакуйованого населення включає також розділ з транспортного забезпечення евакуації, в якому зазначається [26]:

– кількість транспортних засобів кожного виду і термін їх подачі до пунктів посадки;

– кількість населення, яке підлягає евакуації;

– терміни відправлення евакуйованого населення у безпечні райони;

– терміни прибуття евакуйованого населення до пунктів посадки;

– маршрути руху транспортних засобів;

– кількість рейсів.

На всіх громадян, які підлягають евакуації, завчасно складаються списки за об'єктами і житлово-експлуатаційними організаціями у трьох примірниках, один з яких залишається на об'єкті або в житлово-експлуатаційній організації, другий (у разі одержання рішення про проведення евакуації) після уточнення списків надсилається на збірний евакуаційний пункт, третій - до евакуаційної комісії району (пункту) розміщення.

З отриманням рішення (сигналу) про проведення евакуації евакуаційні комісії уточнюють завдання керівникам об'єктів щодо проведення евакуаційних заходів, контролюють стан оповіщення населення, його збору, формування колон (через начальників маршрутів), забезпечують переміщення їх до пунктів евакуації, а також разом з транспортними службами - готовність транспортних засобів до перевезень, уточнюють порядок їх використання, підтримують постійний зв'язок з начальниками маршрутів та з органами виконавчої влади безпечних районів, інформують їх про хід евакуації.

У райони розміщення евакуаційних органів та населення, яке підлягає евакуації, направляються представники евакуаційних комісій для вирішення питань приймання, розміщення і життєзабезпечення евакуйованого населення.

Керівник органу виконавчої влади і евакуаційна комісія безпечного району, організовують підготовку пунктів висадки, розгортають приймальний евакуаційний пункт, уточнюють кількість прибулих і порядок подачі транспортних засобів для їх вивезення з пунктів висадки, а також з проміжних пунктів евакуації до пунктів розміщення, контролюють роботу керівників об'єктів безпечних районів з прийому і розміщення евакуйованого населення.

У разі оголошення евакуації громадяни самостійно на міських транспортних засобах, які у цей період працюють цілодобово, прибувають на збірні евакуаційні пункти. Працівники цих пунктів розподіляють громадян, які підлягають евакуації, за транспортними засобами, інструктують їх і забезпечують посадку на транспортні засоби.

Евакуйовані громадяни повинні мати при собі паспорт, військовий квиток, документ про освіту, трудову книжку або пенсійне посвідчення, свідоцтво про народження, гроші і цінності, продукти харчування і воду на 3 доби, постільну білизну, необхідний одяг і взуття загальною вагою не більш як 50 кілограмів на кожного члена сім'ї. Дітям дошкільного віку вкладається у кишеню або пришивається до одягу записка, де зазначається прізвище, ім'я та по батькові, домашня адреса, а також ім'я та по батькові матері і батька.

4.3. Висновки до розділу

В цьому розділі проаналізовано важливі питання охорони праці та безпеки в надзвичайних ситуаціях, висвітлено питання планування та порядку проведення евакуації населення з районів наслідків впливу НС техногенного та природного характеру.

ВИСНОВКИ

Упродовж виконання роботи було досліджено основні підходи до управління багатоадресною передачею інформації, обмеження галузі застосування надійних багатоадресних протоколів, що накладаються додатками та мережею.

Також було проведено огляд та порівняльний аналіз вже існуючих рішень у галузі RM протоколів, таких як протоколи PGM та M/TCP. На основі даного аналізу було виявлено коло завдань, які не можуть вирішити вищезгадані протоколи та визначені вимоги до нового RM протоколу.

Було розроблено алгоритм нового протоколу, що відповідає вищевказаним критеріям. Мовою Python була створена тестова реалізація даного алгоритму та розгорнута на лабораторному стенді. Як перевірку роботи здійснювалася відправка файлу від відправника групі одержувачів.

У майбутньому планується доопрацювання реалізації, додавання до неї механізмів, які забезпечують найбільш стабільну роботу алгоритму протоколу.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Технологія “клієнт/сервер”. URL: https://ami.lnu.edu.ua/wp-content/uploads/2020/04/Client_Server_1.pdf (дата звертання 01.12.2023).

2. Mehmet Ulema, Bin Wu. Next generation service overlay networks *// IEEE Communications Magazine 50(1)*: pp. 52-53 · January 2012

3. Кренцін М. Д., Куперштейн Л. М. Аналіз тенденцій розвитку піринговых мереж. *Вісник Хмельницького національного університету*. *Технічні науки.* 2021. Т. 4, № 299. С. 25–29

4. Paul, S. (1998). Reliable Multicast Transport Protocol (RMTP). In: Multicasting on the Internet and its Applications. Springer, Boston, MA.

5. Salekul Islam, Nasif Muslim, J. William Atwood, "A Survey on Multicasting in Software-Defined Networking", *IEEE Communications Surveys & Tutorials*, vol.20, no.1, pp.355-387, 2018.

6. Xinchang Zhang, Meihong Yang, Lu Wang, Meng Sun, "An OpenFlow-Enabled Elastic Loss Recovery Solution for Reliable Multicast", *IEEE Systems Journal*, vol.12, no.2, pp.1945-1956, 2018.

7. M. Handley, S. Floyd. The Reliable Multicast Design Space for Bulk Data Transfer // Digital Fountain, Inc. August 2000

8. . Whetten, L. Vicisano. Reliable Multicast Transport Building Blocks for One-to-Many Bulk-Data Transfer // Digital Fountain, Inc. January 2001

9. Сабат Р.М. Основні механізми підтвердження доставки даних в мережі. *Інформаційні моделі, системи та технології: Праці XI наук.-техн. конф. (Тернопіль, 13-14 грудня 2023 р.)*, Тернопіль, 2023. С. 176.

10. Giacomo Morabito, Sergio Palazzo. Modeling and Analysis of TCPLike Multicast Congestion Control in Hybrid Terrestrial/Satellite IP Networks // *IEEE Journal on Selected Areas in Communications 22(2):*pp. 401-412 · February 2004

11. Jim Gemmell, Todd Montgomery. The PGM Reliable Multicast Protocol // *IEEE Journal on Selected Areas in Communications 17(1):*pp. 16-22 · January *2003*

12. T. Shome, S. Gupta, "Performance enhancement of pragmatic general multicast (PGM) protocol using a local loss recovery strategy", 2013 *IEEE/CIC International Conference on Communications in China (ICCC),* pp.27-32, 2013.

13. Vasaka Visoottiviseth, Takuya Mogami. M/TCP: The Multicastextension to Transmission Control Protocol // The Graduate School of Information Science, Nara Institute of Science and Technology, Japan 2015.

14. Wolfgang Braun, Manuel Albert, Toerless Eckert, Michael Menth, "Performance comparison of resilience mechanisms for stateless multicast using BIER", 2017 *IFIP/IEEE Symposium on Integrated Network and Service Management (IM),* pp.230-238, 2017.

15. Яка різниця між TCP та UDP? URL: https://www.thefastcode.com/uk-uah/article/what-s-the-difference-between-tcp-and-udp (дата звертання 10.12.2023).

16. KunPeng Mao, Songtao Guo, Yuanyuan Yang, Guiyan Liu, "MRDC: Multicast Data Restoration in Fat-Tree Data Center Networks", 2016 *IEEE Trustcom/BigDataSE/ISPA*, pp.1500-1507, 2016.

17. Tingwei Zhu, Fang Wang, Yu Hua, Dan Feng, Yong Wan, Qingyu Shi, Yanwen Xie, "MCTCP: Congestion-aware and robust multicast TCP in Software-Defined networks", 2016 *IEEE/ACM 24th International Symposium on Quality of Service (IWQoS),* pp.1-10, 2016.

18. Dong Han, Yunyan Xiong, Meisheng Li, "Research on Reliable Multicast in EBSN", 2016 *International Conference on Network and Information Systems for Computers (ICNISC)*, pp.23-25, 2016.

19. Khang-Siang Wong, Tat-Chee Wan, Way-Chuang Ang, "A survey on current status of Disruption Tolerant Network support for Multicast", 2016 *3rd International Conference on Computer and Information Sciences (ICCOINS)*, pp.276-281, 2016.

20. P. M. Jawandhiya, M. Ali, S. F. Husain, M. Parate, and J. Deshpande, "Reliable Multicast Transport Protocol: RMTP," *International Journal of Advanced Computer Science and Applications, v*ol. 1, pp. 74-80, 2010.

21. Jie Li, Malathi Veeraraghavan, Steve Emmerson, Robert. D. Russell, "File Multicast Transport Protocol (FMTP)", 2015 *15th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing*, pp.1037-1046, 2015.

22. Hicham Khalifé, Vania Conan, Jérémie Leguay, Thrasyvoulos Spyropoulos, "Point to multipoint transport in multichannel wireless environments", 2013 *IEEE Wireless Communications and Networking Conference (WCNC)*, pp.1404-1409, 2013.

23. Yuanlong Tan, Malathi Veeraraghavan, Hwajung Lee, Steven Emmerson, Jack W. Davidson, "High-performance reliable network-multicast over a trial deployment", *Cluster Computing*, vol.25, no.4, pp.2931, 2022.

24. Martin Küttler, Maksym Planeta, Jan Bierbaum, Carsten Weinhold, Hermann Härtig, Amnon Barak, Torsten Hoefler, "Corrected trees for reliable group communication", *Proceedings of the 24th Symposium on Principles and Practice of Parallel Programming*, pp.287, 2019.

25. Зеркалов Д.В. Охорона праці в галузі: Загальні вимоги. Навчальний посібник. К.: Основа. 2011. 551 с.

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

ДОДАТОК А

Тези конференції

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ТЕРНОПІЛЬСЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ ІМЕНІ ІВАНА ПУЛЮЯ

М А Т Е Р І А Л И

XI НАУКОВО-ТЕХНІЧНОЇ КОНФЕРЕНЦІЇ

«ІНФОРМАЦІЙНІ МОДЕЛІ, СИСТЕМИ ТА ТЕХНОЛОГІЇ»

13-14 грудня 2023 року

ТЕРНОПІЛЬ
2023

УДК 621.39
Р.М. Сабат, І.О.Баран, к.т.н., доц.
Тернопільський національний технічний університет імені Івана Пулюя, Україна

ОСНОВНІ МЕХАНІЗМИ ПІДТВЕРДЖЕННЯ ДОСТАВКИ ДАНИХ В МЕРЕЖІ

R.M. Sabat, O. Baran, Ph.D., Assoc. Prof.

MAIN MECHANISMS FOR CONFIRMATION OF DATA DELIVERY ON THE NETWORK

Дві основні проблеми, що виникають при розробці протоколу з технологією надійної багатоадресної передачі даних в мережі: контролю навантаження; забезпечення гарної пропускної спроможності.

Втрата пакетів відіграє головну роль щодо цих проблем і є основною перешкодою, яку необхідно подолати для досягнення хорошої пропускної спроможності та забезпечення роботи мережі без перевантажень. Таким чином, реєстрація втрати пакета та реагування на неї має вирішальне значення для вирішення цих проблем. Механізми контролю за доставкою можуть використовувати один або кілька методів.

ACK-based mechanism (механізм на основі підтвердження пакету даних) є найпростішим. Кожен одержувач відправляє відправнику пакет із підтвердженням після кожного прийнятого пакета даних. Якщо підтвердження не надходить, відправник здійснює повторне відправлення пакетів. Такі механізми обмежені дуже невеликою кількістю одержувачів через нездатність відправника обробляти велику кількість підтверджень.

Tree-based ACK Mechanisms (механізм на основі дерева підтверджень) організований за структурою зв'язкового ациклічного графа , в якому одержувачі є кінцевими вершинами, а коренем є відправник. Одержувачі генерують ACK -пакет для батьківського вузла, який, у свою чергу, агрегує ці ACK -пакети та відправляє своєму батьківському вузлу, таким чином усі підтвердження доходять до відправника, цей механізм відноситься лише до передачі підтверджень про доставку, в той час як дані передаються як зазвичай від відправника до одержувача. Даний механізм є добре масштабованим і має хорошу стійкість до відмов, але вимагає певного ступеня підтримки з боку мережі.

NAK-based mechanisms (механізм на основі негативних підтверджень), замість надсилання підтверджень для кожного отриманого пакета даних одержувачі надсилають так звані NAK для кожного пакета даних, який вони не отримали. Це має низку переваг перед механізмами на основі ACK:

– відправнику не потрібно знати точну кількість одержувачів. Це усуває етап побудови топології, необхідний алгоритмів з урахуванням ACK;

– підвищення відмовостійкості;

– потрібно лише одне NAK від будь-якого з одержувачів, щоб донести до відправника інформацію про те який пакет втрачено і в якого числа одержувачів.

Недоліками є те, що відправнику важче визначити той момент, коли можна звільнити буфер передачі, також необхідні додаткові механізми, щоб визначити чи дійсно всі пакети даних дійшли до одержувача.

FEC (Forward Error Correction) – механізм на основі прямої корекції помилок є добре відомим методом захисту даних від ушкоджень. Найпростіша форма FEC на рівні пакета полягає в тому, щоб застосувати до групи пакетів бітову операцію порозрядного поділу (XOR), внаслідок цієї операції формується новий пакет, який відправляється разом із рештою.