Форма № 24

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Факультет інформаційної безпеки та електронних комунікацій
(повне найменування факультету)

Кафедра радіотехніки та телекомунікацій
(повне найменування кафедри)

Пояснювальна записка

до дипломного проєкту (роботи)
баклавра
(ступінь вищої освіти)

на тему **ПОРІВНЯННЯ ПРОТОКОЛІВ ДИНАМІЧНОЇ МАРШРУТИЗАЦІЇ RIP ТА OSPF В КОМП'ЮТЕРНІЙ МЕРЕЖІ**
(назва теми)

Виконав(ла): студент(ка) 4 курсу, групи БК-911
Спеціальності
172 «Телекомунікації та радіотехніка»
(код і найменування спеціальності)
Освітня програма (спеціалізація)
«Інформаційні мережі зв'язку»
ОТРОЩЕНКО І.С.
(ПРІЗВИЩЕ та ініціали)
Керівник КОСТЕНКО В.О.
(ПРІЗВИЩЕ та ініціали)
Рецензент
(ПРІЗВИЩЕ та ініціали)

2025 рік

Форма № 25

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Факультет *Інформаційної безпеки та електронних комунікацій*
Кафедра *Радіотехніки та телекомунікацій*
Ступінь вищої освіти *бакалавр*
Спеціальність *172 «Телекомунікації та радіотехніка»*
(код і найменування)
Освітня програма (спеціалізація) *«Інформаційні мережі зв'язку»*
(назва освітньої програми (спеціалізації))

ЗАТВЕРДЖУЮ
В.о. завідувача кафедри *РТТ*
к.ф.-м.н., доц. Сергій САМОЙЛИК
«_____» *червня* 2025 року

З А В Д А Н Н Я
НА ДИПЛОМНИЙ ПРОЄКТ (РОБОТУ) СТУДЕНТА(КИ)

ОТРОЩЕНКА Ігоря Сергійовича
(ПРІЗВИЩЕ, ім'я, по батькові)

1. Тема проєкту (роботи) *Порівняння протоколів динамічної маршрутизації RIP та OSPF в комп'ютерній мережі*
керівник проєкту (роботи) *к.т.н., доцент, КОСТЕНКО В. О.*
(науковий ступінь, вчене звання, ПРІЗВИЩЕ, ім'я, по батькові)
затверджені наказом закладу вищої освіти від «*17*» *квітня* 2025 року № 189
2. Строк подання студентом проєкту (роботи) *19 червня 2025 року*
3. Вихідні дані до проєкту (роботи) *Дві локальні мережі, п'ять маршрутизаторів з підтримкою протоколів динамічної маршрутизації, сценарії розриву зв'язку.*

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) *Теоретичні основи динамічної маршрутизації. Протоколи RIP та OSPF: огляд і порівняння. Конфігурація маршрутизаторів в комп'ютерній мережі з динамічною маршрутизацією. Моделювання сценарія розриву зв'язку в OPNET Modeler.*

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, кількість слайдів, плакатів)
Презентація роботи в Microsoft PowerPoint з поясненням алгоритму проведення дослідження з моделюванням та порівнянням протоколів динамічної маршрутизації RIP та OSPF, опис послідовності налаштувань і параметрів, використаних у середовищі OPNET Modeler, скріншоти мережевих сценаріїв, що демонструють роботу кожного з протоколів (12 слайдів).

6. Консультанти розділів проєкту (роботи)

Розділ	ПРІЗВИЩЕ, ініціали та посада консультанта	Підпис, дата	
		завдання видав	прийняв виконане завдання
1-3	*КОСТЕНКО В. О., доцент кафедри РТТ*		
нормо-	*МОРОЗ Г. В., ст. викладач кафедри РТТ*		
контроль			

7. Дата видачі завдання «*17*» *квітня* 2025 року.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проєкту (роботи)	Строк виконання етапів проєкту (роботи)	Примітка
1	*Принцип роботи динамічної маршрутизації та її протоколів*	*17.03-17.04*	
2	*Методи боротьби з повільною конвергенцією*	*18.04-30.04*	
3	*Конфігурація маршрутизаторів для правильної роботи протоколів RIP та OSPF*	*01.05-14.05*	
4	*Моделювання роботи протоколів в мережі з розірванням зв'язку у програмному середовищі Opnet Modeler*	*15.05-30.05*	
5	*Оформлення дипломної роботи*	*01.06-14.06*	

Студент(ка) _____ *Ігор ОТРОЩЕНКО*
(підпис) (Ім'я ПРІЗВИЩЕ)

Керівник проєкту (роботи) _____ *Валер'ян КОСТЕНКО*
(підпис) (Ім'я ПРІЗВИЩЕ)

РЕФЕРАТ

Пояснювальна записка до бакалаврської роботи: 61 с., 6 табл., 20 рис., 12 джерел.

КОНВЕРГЕНЦІЯ, ДИНАМІЧНА МАРШРУТИЗАЦІЯ, ПЕРЕДАЧА ІНФОРМАЦІЇ, КОНФІГУРАЦІЯ МАРШРУТИЗАТОРА, ФОРМАТ ПОВІДОМЛЕННЯ, КОМП'ЮТЕРНІ МЕРЕЖІ, МОДЕЛЮВАННЯ.

Мета роботи – дослідити особливості роботи протоколів маршрутизації RIP та OSPF, провести їх порівняльний аналіз і визначити їхній вплив на час та ефективність конвергенції в мережі.

Об’єкт дослідження – протоколи динамічної маршрутизації у комп’ютерних мережах.

Предмет дослідження – механізми роботи, параметри конвергенції та ефективність протоколів RIP і OSPF у різних мережевих умовах.

Методи дослідження: теоретичний аналіз стандартів протоколів RIP та OSPF, симуляції у середовищі OPNET Modeler, порівняльний аналіз параметрів часу конвергенції, аналіз логів і статистики мережевого трафіку.

ЗМІСТ

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

RIP – (Routing Information Protocol) протокол маршрутизації з використанням алгоритму відстані до призначення

OSPF – (Open Shortest Path First) протокол маршрутизації, що використовує алгоритм пошуку найкоротшого шляху

IP – (Internet Protocol) протокол міжмережевої взаємодії

UDP – (User Datagram Protocol) протокол датаграм користувача

IGP – (Interior Gateway Protocol) протокол внутрішнього шлюзу

LSA – (Link-State Advertisement) повідомлення про стан зв'язку

LSU – (Link State Update) оновлення стану зв'язку

LSR – (Link State Request) запрос стану зв'язку

FSM – (Finite-State Machine) скінченний автомат

LSDB – (Link State Database) база даних станів каналів зв’язку

Mb/s – мегабіт в секунду

ВСТУП

Протокол маршрутизації – це мова, якою маршрутизатор розмовляє з іншими маршрутизаторами для обміну інформацією про доступність і стан мереж. Протоколи динамічної маршрутизації не лише виконують функції визначення шляху та оновлення маршрутної таблиці, але й визначають наступний найкращий шлях, якщо попередній найкращий шлях до пункту призначення стає непридатним. Здатність компенсувати зміни топології є найважливішою перевагою динамічної маршрутизації над статичною. Очевидно, що для того, щоб комунікація відбувалася, комунікатори повинні розмовляти однією мовою. Протоколи динамічної використовуються для полегшення обміну інформацією про маршрутизацію між маршрутизаторами. Існує вісім основних протоколів IP-маршрутизації, але якщо один маршрутизатор використовує RIP, а інший - OSPF, вони не можуть обмінюватися інформацією про маршрутизацію, оскільки не розмовляють однією мовою.

1 ДИНАМІЧНА МАРШРУТИЗАЦІЯ ТА ЇЇ ПРОТОКОЛИ

1.1 Статична маршрутизація проти динамічної

Недоліки статичної маршрутизації очевидні: вона не може пристосуватися до швидкого росту або швидких змін. У великих мережах, що швидко змінюються, таких як Інтернет, люди просто не можуть реагувати на зміни достатньо швидко, щоб впоратися з проблемами; необхідно використовувати автоматизовані методи. Вони також можуть допомогти підвищити надійність і швидкість реагування на збої в невеликих мережах, які мають альтернативні маршрути.

В мережевих архітектурах, які мають кілька фізичних шляхів, адміністратори зазвичай вибирають один з них як основний. Якщо маршрутизатори на основному шляху виходять з ладу, маршрути необхідно змінити, щоб перенаправити трафік по альтернативному шляху. Зміна маршрутів вручну забирає багато часу і може призвести до помилок. Таким чином, навіть у невеликих мережах слід використовувати автоматизовану систему для швидкої та надійної зміни маршрутів.

Для автоматизації завдання збереження точної інформації про доступність мережі внутрішні маршрутизатори зазвичай взаємодіють один з одним, обмінюючись або даними про доступність мережі, або інформацією про мережеву маршрутизацію, на основі якої можна визначити доступність. Після того, як інформація про доступність для всієї автономної системи зібрана, один з маршрутизаторів системи може повідомити її іншим автономним системам за допомогою протоколу зовнішнього шлюзу (Exterior Gateway Protocol).

Всі протоколи динамічної маршрутизації побудовані на основі алгоритму. Як правило, алгоритм – це покрокова процедура для вирішення

якоїсь задачі. Алгоритм маршрутизації повинен, як мінімум, визначати наступне:

а) процедуру передачі інформації про доступність мереж іншим маршрутизаторам;

б) процедуру отримання інформації про доступність від інших маршрутизаторів;

в) процедуру визначення оптимальних маршрутів на основі інформації про доступність, яку він має, і запису цієї інформації в таблицю;

г) процедуру реагування, компенсації та інформування про зміни топології в мережі.

Декілька тем, спільних для будь-якого протоколу маршрутизації – це визначення найкоротшого шляху, метрика, конвергенція і балансування навантаження.

Визначення найкоротшого шляху в динамічній маршрутизації – це процес, за допомогою якого маршрутизатори вибирають найкращий шлях для пересилання пакетів на основі мережевих умов у реальному часі. На відміну від статичної маршрутизації, де шляхи конфігуруються вручну, протоколи динамічної маршрутизації дозволяють маршрутизаторам автоматично оновлюватись та адаптуватись до змін у топології мережі.

Маршрутизатори використовують протоколи маршрутизації для обміну інформацією про доступні мережі та визначення потенційних шляхів до мереж призначення. Ці протоколи включають:

а) дистанційно-векторний протокол (наприклад RIP);

б) протокол з врахуванням стану каналу (OSPF);

в) протокол векторного шляху (BGP).

Протокол RIP є простою реалізацією векторної маршрутизації для локальних мереж. Він розділяє пристрої на активні і неактивні. Активні пристрої повідомляють іншим про свої маршрути; неактивні пристрої слухають RIP-повідомлення і використовують їх для оновлення своєї таблиці маршрутизації, але не повідомляють про них. Тільки маршрутизатор може

запускати RIP в активному режимі; хост повинен використовувати неактивний режим.

Маршрутизатор, на якому працює RIP в активному режимі, кожні 30 секунд надсилає повідомлення про оновлення маршрутизації. Оновлення містить інформацію, взяту з поточної бази даних маршрутизатора. Кожне оновлення містить набір пар, де кожна пара містить IP-адресу мережі та відстань до цієї мережі. RIP використовує метрику кількості переходів для вимірювання відстаней. У метриці RIP маршрутизатор визначається як такий, що знаходиться на відстані одного переходу від безпосередньо підключеної мережі, двох переходів від мережі, до якої можна дістатися через інший маршрутизатор, і так далі. Таким чином, кількість переходів або кількість хопів на шляху від заданого джерела до заданого пункту призначення означає кількість маршрутизаторів, з якими взаємодіє дейтаграма на цьому шляху.

Наприклад, від маршрутизатора R1 (Рисунок 1.1 – *Схема мережі*) до маршрутизатора R2 потрібно пройти 1 шлях, якщо пакети надсилаються до маршрутизатора R3, тоді кількість переходів дорівнює 2. Максимально дозволена кількість переходів в RIP – 15. Якщо маршрут має 16 переходів і більше, тоді він вважається недоступним. Очевидно, що використання кількості переходів для обчислення найкоротших шляхів не завжди дає оптимальні результати. Наприклад, шлях з кількістю переходів 3, який перетинає три мережі, може бути значно швидшим, ніж шлях з кількістю переходів 2, який перетинає два супутникових з'єднання. Щоб компенсувати відмінності в технологіях, багато реалізацій RIP дозволяють адміністраторам штучно налаштовувати високі значення кількості переходів при з'єднані з повільними мережами.

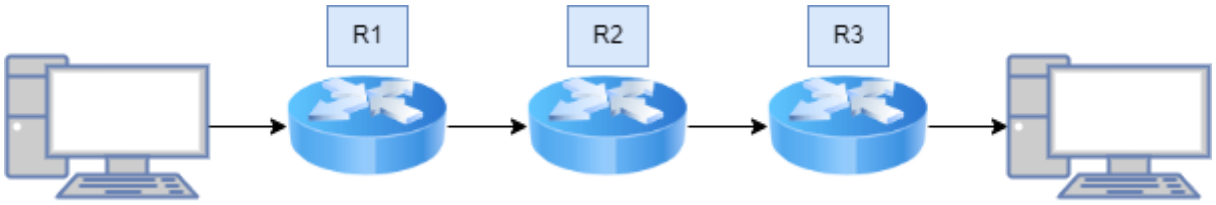

Рисунок 1.1 – Схема мережі

RIP визначає, що всі користувачі повинні витримувати тайм-аут маршрутів, які вони отримують через RIP. Коли маршрутизатор встановлює маршрут у своїй таблиці, він запускає таймер для цього маршруту. Таймер потрібно перезапускати щоразу, коли маршрутизатор отримує нове RIP-повідомлення з новим маршрутом. Маршрут стає недійсним, якщо минає 180 секунд, а маршрут не з'являється знову.

RIP повинен обробляти три типи помилок, спричинених базовим алгоритмом. По-перше, оскільки алгоритм явно не виявляє зациклення маршрутизації, RIP повинен або припускати, що учасникам можна довіряти, або вживати заходи для запобігання таким зацикленням. По-друге, для запобігання нестабільності RIP повинен використовувати низьке значення максимально можливої відстані (RIP використовує 16). Таким чином, для мереж, в яких допустима кількість переходів наближається до 16, адміністратори повинні розділити мережу на секції або використовувати альтернативний протокол. По-третє, дистанційно-векторний алгоритм, що використовується в RIP, може створити проблему повільної конвергенції, або проблему максимальної метрики маршруту, в якій виникають помилки через те, що повідомлення про оновлення маршрутизації повільно поширюються мережею.

Проблема повільної конвергенції полягає в інформаційному потоці. Якщо маршрутизатор повідомляє про короткий маршрут до якоїсь мережі, всі маршрутизатори, які його отримують, швидко реагують і встановлюють цей маршрут. Якщо маршрутизатор припиняє повідомляти маршрут, протокол

повинен покладатися на механізму тайм-ауту, перш ніж вважати маршрут недосяжним. Як тільки тайм-аут закінчується, маршрутизатор знаходить альтернативний маршрут і починає поширювати цю інформацію. На жаль, маршрутизатор не може знати, чи залежав альтернативний маршрут від маршруту, який щойно зник. Таким чином, інформація не завжди швидко передається.

Інший метод, який використовується для вирішення проблеми повільної конвергенції – це Hold down. Цей метод змушує маршрутизатора ігнорувати інформацію про мережу протягом певного періоду часу після отримання повідомлення про те, що мережа недоступна. Зазвичай, період призупинення встановлюється на 60 секунд. Ідея полягає в тому, щоб почекати достатньо довго, щоб переконатися, що всі машини отримали повідомлення, а не помилково прийняли інформацію, яка є неактуальною. Слід зазначити, що всі пристрої, які беруть участь в обміні RIP, повинні використовувати ідентичні поняття затримки, інакше можуть виникнути петлі маршрутизації. Недоліком методу затримки є те, що якщо виникають петлі маршрутизації, то вони зберігаються протягом усього періоду затримки. Ще важливіше те, що метод призупинення зберігає всі неправильні маршрути протягом періоду призупинення, навіть якщо існують альтернативи.

Останній метод вирішення проблеми повільної конвергенції називається Poison Reverse. Як тільки з'єднання зникає, маршрутизатор, який встановлює з'єднання, зберігає запис протягом декількох періодів оновлення. Щоб зробити Poison Reverse найбільш ефективним, його потрібно поєднувати з автоматичними оновленнями. Вони змушують маршрутизатор негайно надсилати інформацію, замість того, щоб чекати на наступну періодичну передачу. Негайно надсилаючи оновлення, маршрутизатор мінімізує час, протягом якого він вразливий.

Використання широкомовленної передачі, можливість утворення петель маршрутизації та використання утримання для запобігання повільній конвергенції може зробити RIP вкрай неефективним у глобальній мережі.

Широкомовлення завжди вимагає значної пропускної здатності. Навіть якщо не виникає проблем, періодична трансляція з усіх машин означає, що трафік зростає зі збільшенням кількості маршрутизаторів. Потенціал петель маршрутизації також може бути небезпечним, коли пропускна здатність лінії обмежена. Як тільки лінії перенасичуються пакетами, що зациклюються, маршрутизаторам може бути важко або неможливо обмінюватися повідомленнями про маршрутизацію, необхідними для розірвання петель. Крім того, у глобальній мережі періоди утримання настільки довгі, що таймери, які використовуються протоколами вищих рівнів, можуть закінчитися і призвести до розриву з'єднання. Незважаючи на ці відомі проблеми, багато груп продовжують використовувати RIP як IGP в глобальних мережах.

1.2 Формат повідомлення RIP1

RIP-повідомлення можна умовно поділити на два типи: повідомлення з інформацією про маршрутизацію та повідомлення, що використовуються для запиту інформації. Обидва типи повідомлень використовують однаковий формат, який складається з фіксованого заголовка, за яким слідує необов'язковий список пар мереж і відстаней. На рисунку 1.2 показано формат повідомлення, що використовується у версії 1 протоколу, яка відома як RIP1. На Рисунок 1.2 – *Формат повідомлення RIP1*поле Command визначає операцію відповідно до таблиці 1.1.

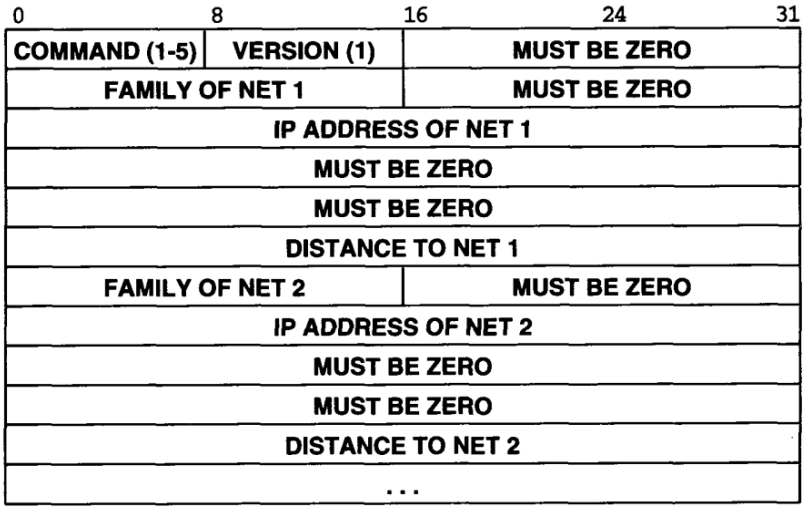

Рисунок 1.2 – Формат повідомлення RIP1

Маршрутизатор або хост може запитати у іншого маршрутизатора інформацію про маршрутизацію, надіславши команду запиту.

Таблиця 1.1 – Значення номеру команди у відповідному полі

Команда	Значення
1	Запит на отримання часткової або повної інформації про маршрут
2	Відповідь містить пари мережа-відстань з таблиці маршрутизації відправника
3	Увімкнути режим трасування
4	Вимкнути режим трасування
5	Зарезервовано для внутрішнього використання Sun Microsystems
9	Запит на оновлення
10	Оновлення відповіді
11	Підтвердження оновлення

Маршрутизатори відповідають на запити за допомогою команди відповіді. Однак у більшості випадків маршрутизатори періодично надсилають небажані повідомлення відповіді. Поле Version містить номер версії протоколу (у цьому випадку 1) і використовується приймачем для перевірки правильності інтерпретації повідомлення.

1.3 Адреси RIP1

Універсальність RIP також проявляється у способі передачі мережевих адрес. Формат адреси не обмежується використанням TCP/IP; його можна використовувати з багатьма наборами мережевих протоколів. Як показано на рисунку 1.2, кожна мережева адреса, яку повідомляє RIP, може мати довжину до 14 октетів. Звичайно, для IP-адреси потрібно лише 4; RIP визначає, що решта октетів повинні бути нульовими. Поле Family of Network ідентифікує сімейство протоколів, під яким слід інтерпретувати мережеву адресу. RIP використовує значення, присвоєні сімействам адрес в операційній системі 4BSD UNIX (IP-адресам присвоюється значення 2).

Окрім звичайних IP-адрес, RIP використовує правило, згідно з якого адреса 0.0.0.0 позначає маршрут за умовчанням. RIP додає метрику відстані до кожного маршруту, який він оголошує, включаючи маршрути за умовчанням. Таким чином, можна налаштувати два маршрутизатори на використання маршруту за умовчанням (наприклад, маршруту до Інтернету) з різними метриками, зробивши один з них основним, а інший – резервним.

Останнє поле кожного запису в RIP-повідомленні, Distance to Network 1 містить цілочисельне значення відстані до вказаної мережі.

1.4 RIP1 Інтерпретація та агрегація маршрутів

Оскільки протокол RIP спочатку був розроблений для використання з класовими адресами, версія 1 не містила жодних положень про маску

підмережі. Коли адресація підмереж була додана до IP, версія RIP1 була розширена, щоб дозволити маршрутизаторам обмінюватися адресами підмереж. Однак, оскільки повідомлення про оновлення RIP1 не містять явної інформації про маску, було додано важливе обмеження: маршрутизатор може включати адреси хостів або підмереж в оновлення маршрутизації, якщо всі одержувачі можуть однозначно інтерпретувати ці адреси. Зокрема, маршрути підмережі можуть бути включені до оновлень, що надсилаються через мережу, яка є частиною приставкою підмережі, і тільки якщо маска підмережі, що використовується, збігається з маскою підмережі, що використовується для адреси. По суті, це обмеження означає, що RIP1 не можна використовувати для поширення адрес підмереж змінної довжини або безкласових адрес.

1.5 Розширення RIP2

Обмеження на інтерпретацію адрес означає, що версія RIP1 не може використовуватися для передачі адрес підмереж змінної довжини або безкласових адрес, що використовуються в CIDR. Коли була визначена версія RIP2, протокол було розширено, щоб включити явну маску підмережі разом з кожною адресою. Крім того, оновлення RIP2 включають явну інформацію про наступний вузол, що запобігає виникненню петель маршрутизації та повільній конвергенції. В результаті, RIP2 пропонує значно розширену функціональність, а також покращену стійкість до помилок.

1.6 Формат повідомлення RIP2

Формат повідомлень, що використовується в RIP2, є розширенням формату RIP1, з додатковою інформацією, що займає невикористані октети поля адреси. Зокрема, кожна адреса містить явний наступний крок, а також явну маску підмережі, як показано на рисунку 1.3.

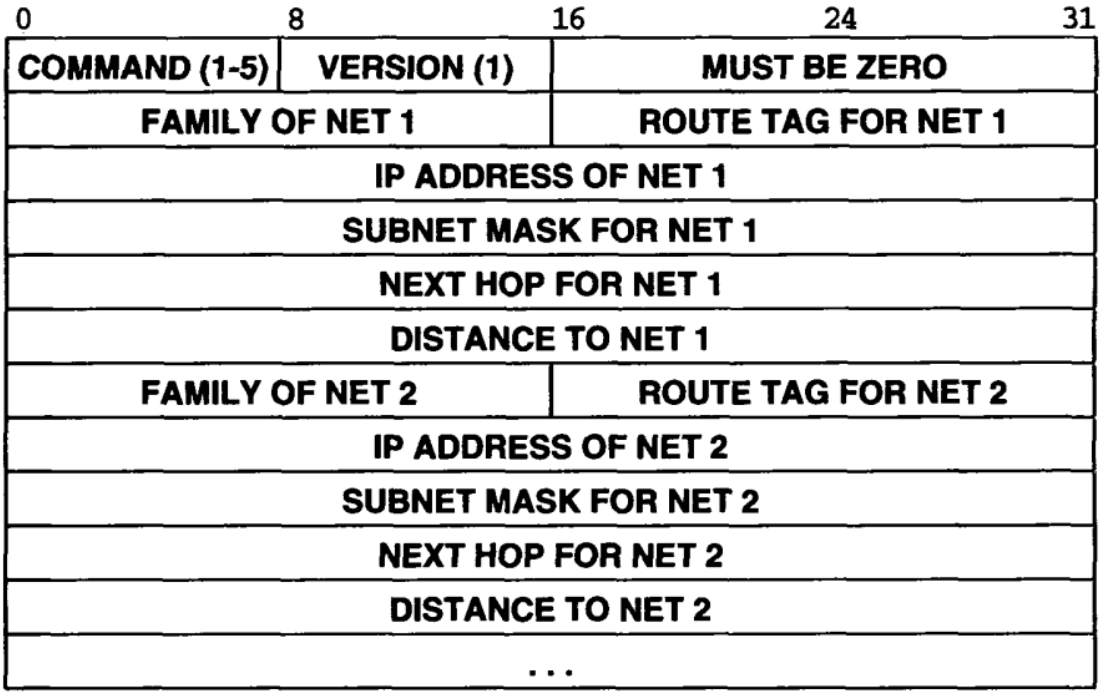

Рисунок 1.3 – Формат повідомлень RIP2

RIP2 також додає 16-бітне поле Route Tag до кожного запису. Маршрутизатор повинен надсилати такий самий тег, який він отримує під час передачі маршруту. Таким чином, тег забезпечує спосіб передачі додаткової інформації, такої як початковий маршрут. Зокрема, якщо RIP2 дізнається маршрут від іншої автономної системи, він може використовувати тег Route Tag для передачі номера автономної системи.

Оскільки номер версії в RIP2 займає той самий октет, що і в RIP1, обидві версії протоколів можуть використовуватися на даному маршрутизаторі одночасно без перешкод. Перед обробкою вхідного повідомлення програмне забезпечення RIP перевіряє номер версії.

1.7 Передача RIP-повідомлень

Повідомлення RIP не містять явного поля довжини або явної кількості записів. Замість цього RIP припускає, що основний механізм доставки

повідомить одержувачу довжину вхідного повідомлення. Зокрема, при використанні з TCP/IP, повідомлення RIP покладаються на UDP, щоб повідомити одержувачу довжину повідомлення. RIP працює на порту UDP 520. Хоча RIP-запит може надходити з інших UDP-портів, UDP-порт призначення для запитів завжди є 520, як і порт-джерело, з якого надходять широкомовні повідомлення RIP.

1.8 Відкритий протокол SPF (OSPF)

Алгоритм маршрутизації за станом з'єднання, який використовує SPF для обчислення найкоротших шляхів, масштабується краще, ніж алгоритм вектора відстані. Щоб прискорити впровадження технології стану зв'язку, робоча група Internet Engineering Task Force розробила протокол внутрішнього шлюзу, який використовує алгоритм стану зв'язку. Новий протокол, який отримав назву Open SPF (OSPF), вирішує кілька важливих завдань:

а) як випливає з назви, специфікація доступна в опублікованій літературі. Зробити його відкритим стандартом, який будь-хто може реалізувати без сплати ліцензійних платежів, спонукало багатьох постачальників підтримати OSPF. Як наслідок, він став популярною заміною пропрієтарним протоколам;

б) OSPF включає в себе маршрутизацію типів послуг. Адміністратори можуть встановити кілька маршрутів до певного пункту призначення, по одному для кожного пріоритету або типу послуги. Під час маршрутизації дейтаграми маршрутизатор з OSPF використовує як адресу призначення, так і поле типу сервісу в IP-заголовку для вибору маршруту. OSPF є одним з перших протоколів TCP/IP, який пропонує маршрутизацію за типом сервісу;

в) OSPF забезпечує балансування навантаження. Якщо адміністратор вказує кілька маршрутів до певного пункту призначення з однаковою вартістю, OSPF розподіляє трафік між усіма маршрутами порівну. Знову ж

таки, OSPF є одним з перших відкритих IGP, який запропонував балансування навантаження; такі протоколи, як RIP, обчислюють єдиний маршрут до кожного пункту призначення;

г) щоб забезпечити зростання і полегшити керування мережами, OSPF дозволяє розбивати мережі і маршрутизатори на підмножини, які називаються зонами. Кожна зона є самодостатньою; знання топології зони залишається прихованим від інших зон. Таким чином, кілька груп в межах одного вузла можуть співпрацювати у використанні OSPF для маршрутизації, навіть якщо кожна група зберігає можливість змінювати топологію своєї внутрішньої мережі незалежно;

ґ) протокол OSPF визначає, що всі обміни між маршрутизаторами можуть бути автентифіковані. OSPF допускає різноманітні схеми автентифікації, і навіть дозволяє одній зоні вибирати схему, відмінну від іншої зони. Ідея автентифікації полягає в тому, щоб гарантувати, що тільки надійні маршрутизатори поширюють інформацію про маршрутизацію. Щоб зрозуміти, чому це може бути проблемою, розглянемо, що може статися при використанні RIP1, який не має автентифікації. Якщо зловмисник використовує персональний комп'ютер для розповсюдження RIP-повідомлень з інформацією про низькозатратні маршрути, інші маршрутизатори та хости, на яких запущено RIP, змінять свої маршрути і почнуть надсилати дейтаграми на цей персональний комп'ютер;

д) OSPF включає підтримку маршрутів для хостів, підмереж і безкласових маршрутів, а також класові маршрути для конкретних мереж;

е) для забезпечення максимальної гнучкості OSPF дозволяє адміністраторам описувати віртуальну топологію мережі, яка абстрагується від деталей фізичних з'єднань. Наприклад, адміністратор може налаштувати віртуальне з'єднання між двома маршрутизаторами в графі маршрутизації, навіть якщо фізичне з'єднання між двома маршрутизаторами вимагає зв'язку через транзитну мережу;

є) OSPF дозволяє маршрутизаторам обмінюватися інформацією про маршрутизацію, отриманою від інших (зовнішніх) вузлів. По суті, один або декілька маршрутизаторів, які мають зв'язок з іншими вузлами, отримують інформацію про ці вузли і включають її при відправці повідомлень про оновлення. Формат повідомлень розрізняє інформацію, отриману із зовнішніх джерел, та інформацію, отриману від маршрутизаторів, підключених до вузла, тому немає ніякої двозначності щодо джерела або надійності маршрутів.

1.9 Формат повідомлення OSPF

Кожне OSPF-повідомлення починається з фіксованого 24-октавного заголовка, як показано на рисунку 1.4.

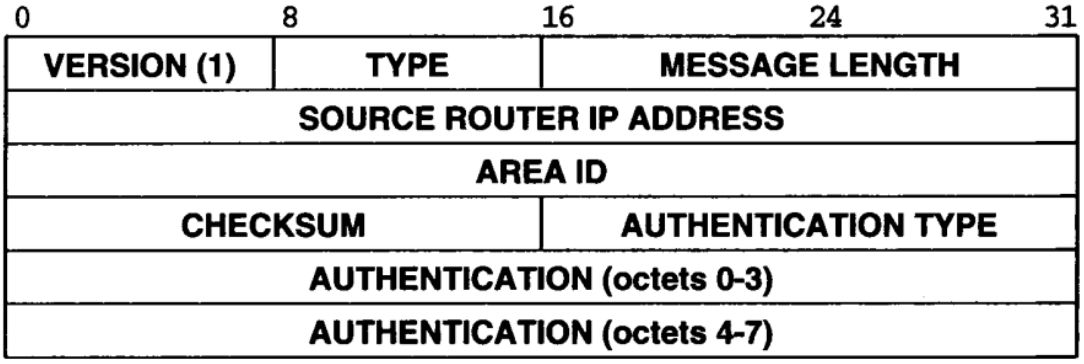

Рисунок 1.4 – Формат повідомлення OSPF

Поле VERSION вказує версію протоколу. Поле TYPE ідентифікує тип повідомлення як один з типів.

В таблиці 1.2 наведено значення типів повідомлень.

Таблиця 1.2 – Значення типів повідомлень

Тип	Значення
1	Hello-повідомлення (використовується для перевірки доступності)
2	Database Description (топологія)
3	Запит статусу каналу зв'язку
4	Оновлення статусу каналу зв'язку
5	Підтвердження стану з'єднання

Поле з назвою SOURCE ROUTER IP ADDRESS містить адресу відправника, а поле з назвою AREA ID – 32-розрядний ідентифікаційний номер зони.

Оскільки кожне повідомлення може містити автентифікацію, поле AUTHENTICATION TYPE вказує, яка схема автентифікації використовується (наразі 0 означає відсутність автентифікації, а 1 – використання простого пароля).

1.10 Формат Hello-повідомлення OSPF

OSPF періодично надсилає Hello-повідомлення на кожне з'єднання, щоб встановити і перевірити доступність сусідів. Формат показано на рисунку 1.5.

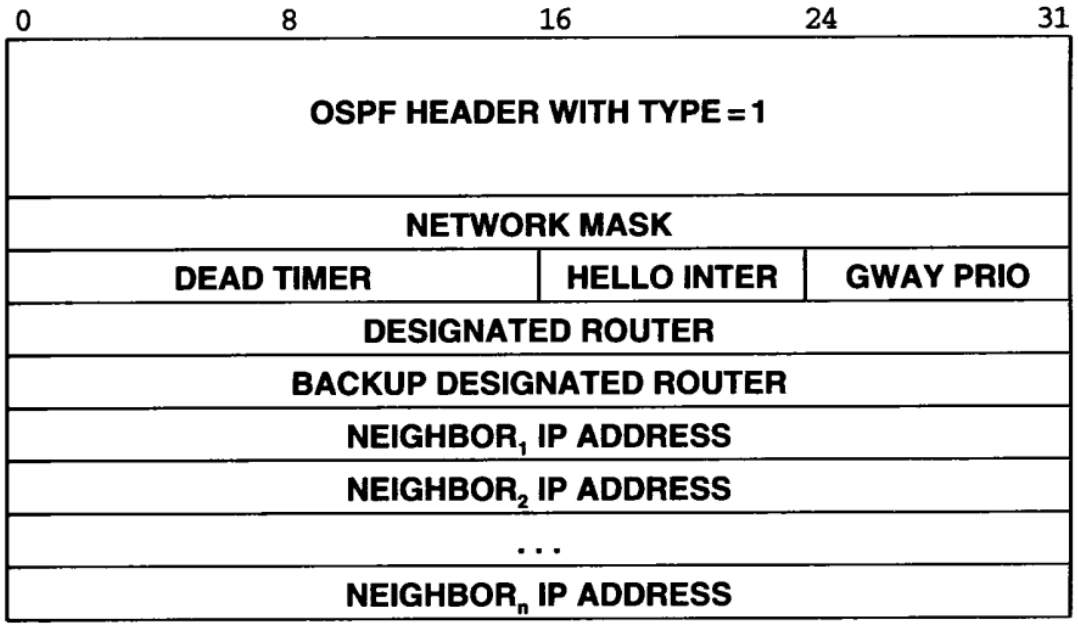

Рисунок 1.5 – Формат Hello-повідомлення

Поле NETWORK MASK містить маску мережі, через яку було надіслано повідомлення. Поле DEAD TIMER задає час у секундах, після якого номінальний сусід, що відповідає, вважається неактивним. Поле HELLO INTER – номінальний інтервал у секундах між повідомленнями привітання. Поле GWAY PRIO є цілочисельним пріоритетом цього маршрутизатора і використовується при виборі резервного маршрутизатора. Поля DESIGNATED ROUTER і BACKUP DESIGNATED ROUTER містять IP-адреси, які дають відправнику інформацію про призначений маршрутизатор і резервний маршрутизатор для мережі, через яку надсилається повідомлення. Поля, позначені як NEIGHBOR IP ADDRESS, містять IP-адреси всіх сусідів з від яких відправник нещодавно отримав Hello-повідомлення.

1.11 Формат повідомлення OSPF Database Description

Маршрутизатори обмінюються повідомленнями OSPF Database Description для ініціалізації своєї бази даних топології мережі. Під час обміну один маршрутизатор виступає в ролі головного, а інший – в ролі залежного.

Залежний маршрутизатор підтверджує кожне повідомлення про опис бази даних відповіддю. Формат повідомлення показано на рисунку 1.6.

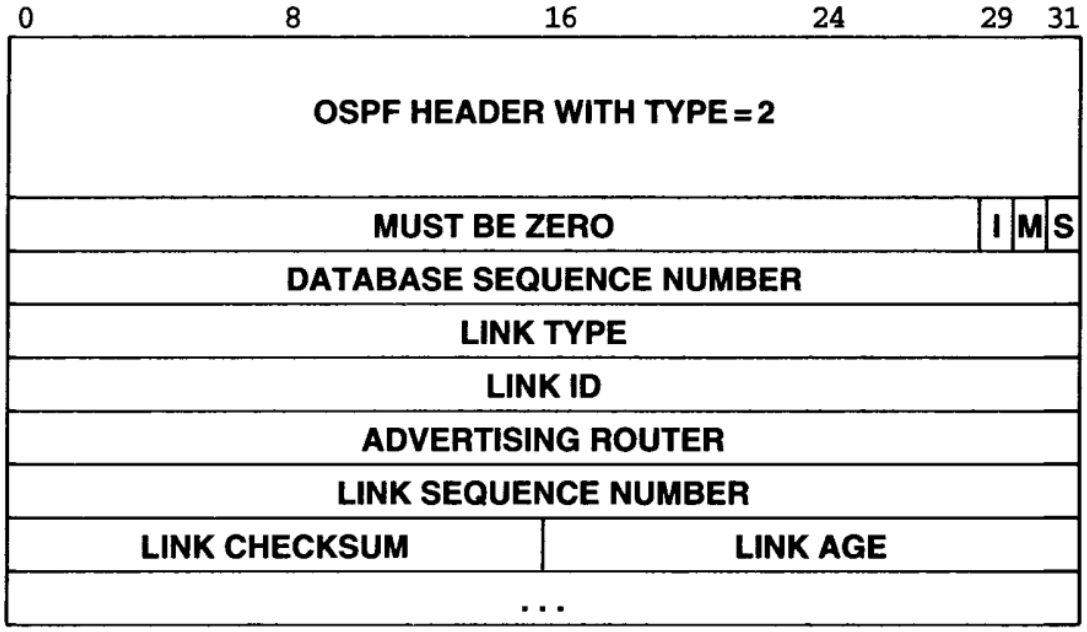

Рисунок 1.6 – Формат повідомлення OSPF Database Description

Оскільки база даних топології може бути великою, її можна розділити на декілька повідомлень за допомогою бітів I та M. Біт I встановлюється в початковому повідомленні; біт M встановлюється в I, якщо слідують додаткові повідомлення. Біт S вказує, чи було повідомлення надіслано головним(I) або залежним (0). Поле DATABASE SEQUENCE NUMBER нумерує повідомлення послідовно, щоб одержувач міг визначити, чи не пропущено якесь із них. Початкове повідомлення містить випадкове ціле число R; наступні повідомлення містять послідовні цілі числа, що починаються з R.

Поля від LINK TYPE до LINK AGE описують одне з'єднання в топології мережі; вони повторюються для кожного з'єднання. Поле LINK TYPE описує з'єднання відповідно до таблиці 1.3.

Таблиця 1.3 – Значення типів повідомлень

Тип з'єднання	Значення
1	З'єднання з маршрутизатором
2	Мережеве підключення
3	Підсумкове з'єднання (IP-мережа)
4	Підсумкове з'єднання (посилання на транзитний маршрутизатор)
5	Стороннє з'єднання (посилання на інший вузел)

Поле LINK ID містить ідентифікатор для з'єднання (це може бути IP-адреса маршрутизатора або мережі, залежно від типу з'єднання).

Поле ADVERTISING ROUTER вказує адресу маршрутизатора, який встановлює це з'єднання, а поле LINK SEQUENCE NUMBER містить ціле число, згенероване цим маршрутизатором, щоб гарантувати, що повідомлення не буде пропущено або отримано в неправильному порядку. Поле LINK CHECKSUM забезпечує додаткову гарантію того, що інформація про посилання не була пошкоджена. Нарешті, поле LINK AGE також допомагає впорядкувати повідомлення – воно містить час у секундах з моменту встановлення з'єднання.

1.12 Формат повідомлення OSPF Link Status Request

Після обміну повідомленнями про опис бази даних з сусідом маршрутизатор може виявити, що частина його бази даних застаріла. Щоб попросити сусіда надати оновлену інформацію, маршрутизатор надсилає повідомлення Link Status Request. У повідомленні перераховуються конкретні посилання, як показано на рисунку 1.7. Сусід відповідає актуальною інформацією про ці посилання, яку він має. Показані три поля повторюються для кожного каналу, про стан якого запитується інформація. Якщо список запитів довгий, може знадобитися більше одного повідомлення про запит.

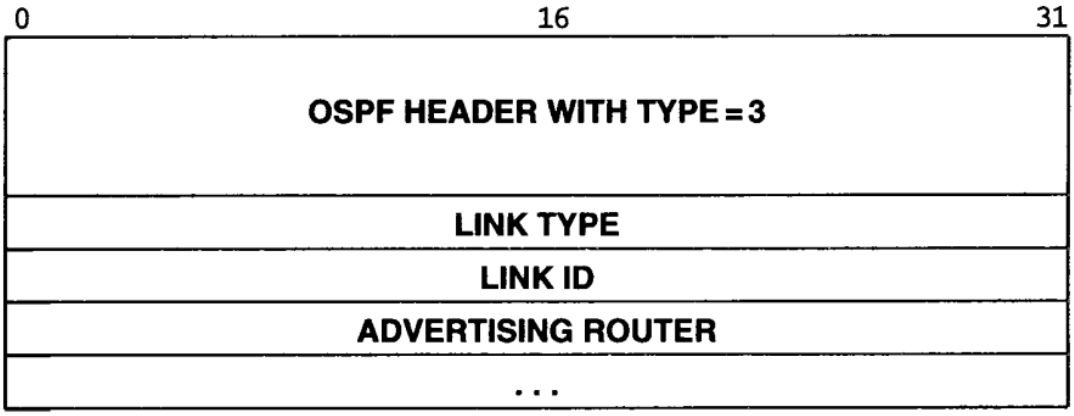

Рисунок 1.7 – Формат повідомлення OSPF Link Status Update

Маршрутизатори транслюють стан посилань за допомогою повідомлень про оновлення стану посилань. Кожне повідомлення складається зі списку оголошень, як показано на рисунку 1.8.

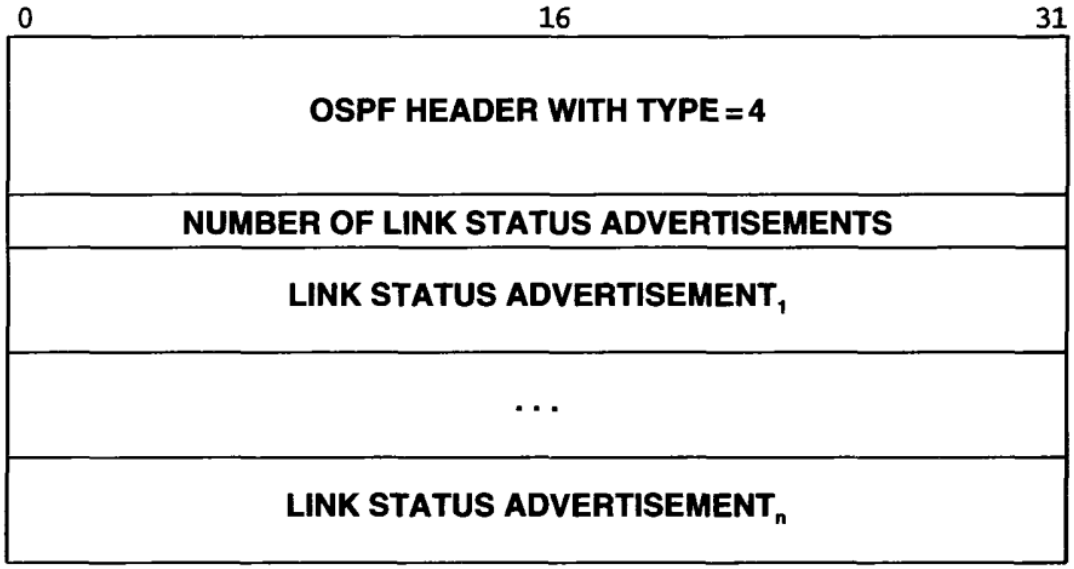

Рисунок 1.8 – Список оголошень

Кожне повідомлення про статус посилання має формат заголовка, як показано на рисунку 1.9. Значення, що використовуються в кожному полі, такі ж самі, як і в повідомленні з описом бази даних.

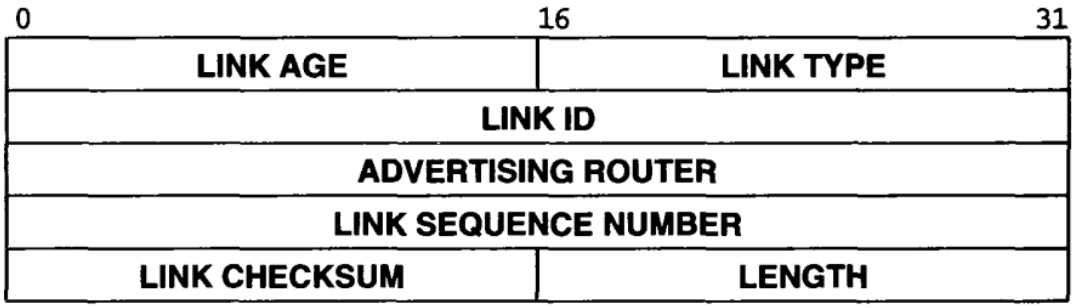

Рисунок 1.9 – Формат заголовка

Після заголовка стану з'єднання йде один з чотирьох можливих форматів для опису з'єднань від маршрутизатора до певної області, з'єднань від маршрутизатора до певної мережі, з'єднань від маршрутизатора до фізичних мереж, які складають єдину підмережу IP, або з'єднань від маршрутизатора до мереж на інших вузлах. У всіх випадках поле LINK TYPE в заголовку стану з'єднання вказує, який з форматів було використано. Таким чином, маршрутизатор, який отримує повідомлення про оновлення статусу посилання, точно знає, які з описаних пунктів призначення знаходяться всередині вузла, а які – ззовні.

1.13 Маршрутизація з неповною інформацією

Хости можуть маршрутизувати лише з частковою ідентифікацією, оскільки вони покладаються на маршрутизатори. Тепер має бути зрозуміло, що не всі маршрутизатори мають повну інформацію. Більшість автономних систем мають один маршрутизатор, який з'єднує автономну систему з іншими автономними системами. Наприклад, якщо сайт підключений до глобальної

мережі Інтернет, принаймні один маршрутизатор повинен мати з'єднання, яке веде від сайту до інтернет-провайдера. Маршрутизатори в автономній системі знають про пункти призначення в межах цієї автономної системи, але вони використовують маршрут за умовчанням для надсилання всього іншого трафіку до провайдера.

Як здійснювати маршрутизацію з неповною інформацією, стає очевидним, якщо ми розглянемо таблиці маршрутизації. Маршрутизатори в центрі Інтернету мають повний набір маршрутів до всіх можливих пунктів призначення, які вони отримують від системи маршрутизації; такі маршрутизатори не використовують маршрутизацію за замовчуванням. Насправді, якщо адреса мережі призначення не з'являється в базі даних маршрутизації, існує лише дві можливості: або адреса не є дійсною адресою призначення, або адреса дійсна, але наразі недоступна (наприклад, через несправність маршрутизаторів або мереж, що ведуть до цієї адреси). Маршрутизатори, крім тих, що належать провайдерам в центрі Інтернету, зазвичай не мають повного набору маршрутів; вони покладаються на маршрут за замовчуванням для обробки мережевих адрес, які вони не визначають.

Використання маршрутів за замовчуванням для більшості маршрутизаторів має два наслідки. По-перше, це означає, що помилки локальної маршрутизації можуть залишитися невиявленими. Наприклад, якщо машина в автономній системі неправильно направить пакет на зовнішню автономну систему замість локального маршрутизатора, зовнішня система направить його назад (можливо, на іншу точку входу). Таким чином, може здатися, що зв'язок збережено, навіть якщо маршрутизація є неправильною. Для невеликих автономних систем з високошвидкісною локальною мережею ця проблема може здатися несерйозною, але в глобальній мережі неправильна маршрутизація може мати катастрофічні наслідки. По-друге, позитивним моментом є те, що використання маршрутів за замовчуванням, коли це можливо, означає, що повідомлення про

оновлення маршрутизації, якими обмінюються більшість маршрутизаторів, будуть набагато меншими, ніж вони були б, якби потрібно було включати повну інформацію.

2 КОНФІГУРАЦІЯ МАРШРУТИЗАТОРІВ В МЕРЕЖАХ З ДИНАМІЧНОЮ МАРШРУТИЗАЦІЄЮ

2.1 Конфігурація RIP

Завдяки простоті RIP, конфігурація є простим завданням. Існує одна команда для увімкнення процесу RIP і по одній команді для кожної мережі, на якій RIP має бути запущено. Окрім цього, RIP має небагато параметрів конфігурації (рис. 2.1).

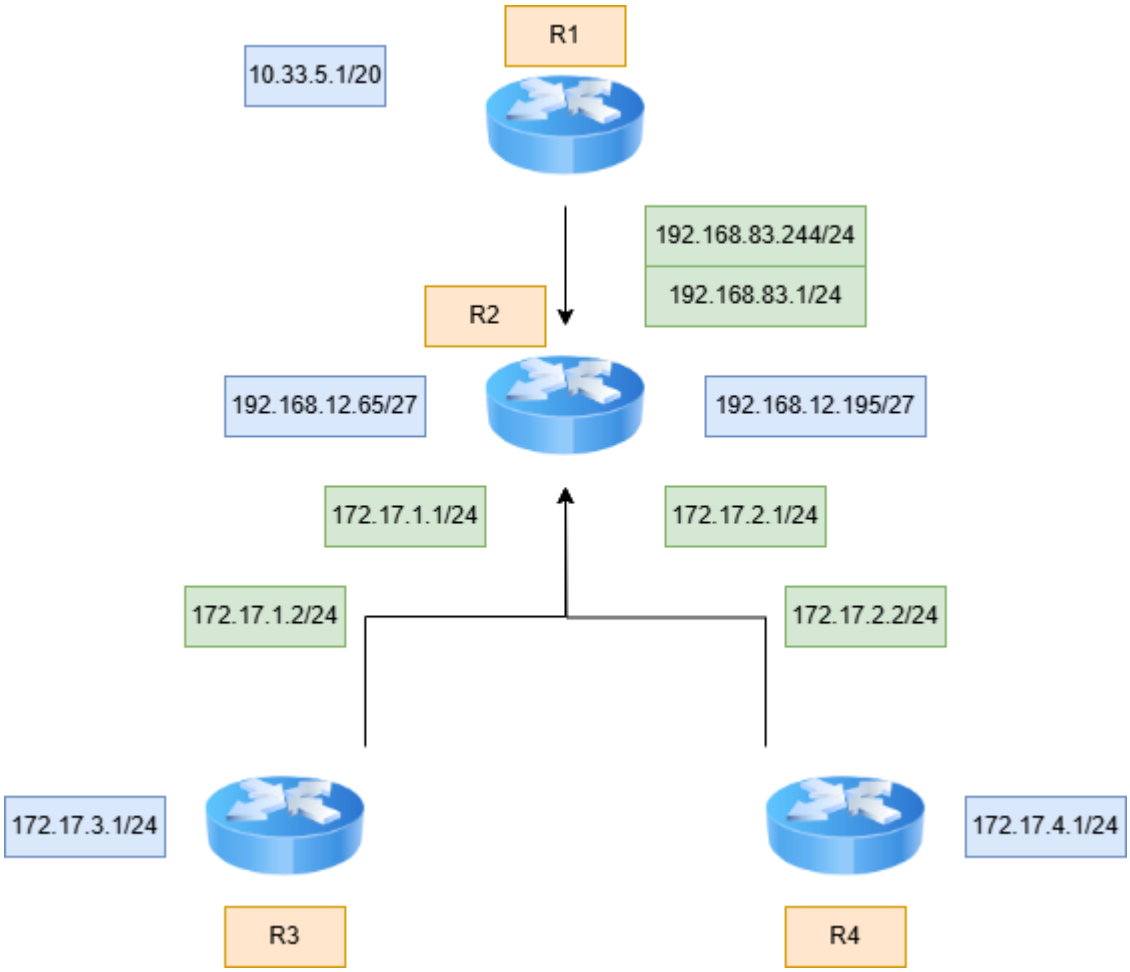

Рисунок 2.1 – Мережа з чотирма маршрутизаторами

Маршрутизатор R1 підключено до двох підмереж мережі 172.17.0.0.

Для налаштування RIP необхідно виконати лише два кроки:

а) увімкнути RIP за допомогою команди router rip;

б) вказати кожну основну мережу, на якій потрібно запустити RIP, за допомогою команди network:

```
Router(config)#router rip
Router(config-router)#network 172.17.0.0
```

Аналогічно, R4 має дві підмережі тієї самої мережі і буде налаштовуватися за допомогою тих самих команд.

Виконання будь-якої команди маршрутизатора переводить маршрутизатор у режим конфігурування маршрутизатора. RIP не передає маску підмережі. Це означає, що за допомогою команди network не можна вказати жодної підмережі – лише основні мережеві адреси класів A, B або C. RIP може працювати на будь-якому інтерфейсі, сконфігурованому з будь-якою адресою, що належить до мережі, вказаної за допомогою команди *network*. Маршрутизатор R1 підключено до двох мереж – 10.0.0.0 і 192.168.83.0. Тому необхідно вказати обидві мережі:

```
Router(config)#router rip
Router(config-router)#network 10.0.0.0
Router(config-router)#network 192.168.83.0
```

Маршрутизатор має одне підключення до мережі 192.168.83.0, підключення до двох підмереж 192.168.12.0 та підключення до двох підмереж 172.17.0.0. Його конфігурація виглядає наступним чином:

```
Router(config)#router rip
Router(config-router)#network 172.17.0.0
```

Router(config-router)#network 192.168.12.0
Router(config-router)#network 192.168.83.0

2.2 Пошук та усунення несправностей RIP

Виправлення несправностей RIP відносно просте. Більшість труднощів з класовими протоколами, такими як RIP, пов'язані або з неправильно налаштованими масками підмереж, або з розірваними підмережами. Якщо таблиця маршрутизації містить неточні або відсутні маршрути, потрібно перевірити всі підмережі на нерозривність і всі маски підмереж на відповідність. Також може бути проблеми, коли маршрутизатор, що може обробляти великі обсяги трафіку або високу частоту оновлень маршрутів надсилає кілька повідомлень RIP на маршрутизатор з обмеженою здатністю швидко обробляти вхідні оновлення. У такому випадку другий маршрутизатор може не встигати обробляти оновлення так швидко, як вони надходять, і інформація про маршрутизацію може бути втрачена. Параметр Output-delay можна використовувати у команді RIP для встановлення міжпакетної затримки від 8 до 50 мілісекунд (за замовчуванням встановлено значення 0 мілісекунд). Простота RIP гарантують, що він буде використовуватися ще багато років. Однак сама простота RIP обмежує його застосування невеликими мережами.

В таблиці 2.1 наведено команди для конфігурації RIP1.

2.3 Конфігурація RIPv2

Оскільки RIPv2 є лише вдосконаленням RIPv1, а не окремим протоколом, команди для керування маршрутизатором з RIPv2 використовуються точно таким же чином.

Таблиця 2.1 – Команди для конфігурації RIP1

Команда	Опис
debug ip rip	Показує RIP-трафік з маршрутизатора
ip address	Конфігурує інтерфейс із вказаною ip-адресою як додаткову адресу
neighbor	Встановлює зв'язок, вказаний за ip-адресою, як найближчий до інтерфейсу
network	Вказує вказану мережу як таку, на якій буде запущено RIP
offset-list	Дозволяє додати або відняти значення до/від метрики маршруту, що проходить через інтерфейс або фільтрується за ACL (access list)
output-delay	Дозволяє уповільнити надсилання маршрутних оновлень по RIP, щоб уникнути перевантаження мережі або буферів на пристроях із обмеженими ресурсами.
passive-interface	Забороняє маршрутизатору надсилати оновлення маршрутизації через вказаний інтерфейс.
router rip	Включає RIP
timers basic	Дозволяє вручну задати значення чотирьох ключових таймерів, що контролюють обмін маршрутною інформацією.

За замовчуванням процес RIP, налаштований на маршрутизаторі Cisco, надсилає лише повідомлення RIPv1, але приймає як RIPv1, так і RIPv2. Цей параметр за замовчуванням змінюється за допомогою команди *version*, як у наступному прикладі:

```
Router(config)#router rip
Router(config-router)#version 2
Router(config-router)#network 172.25.0.0
Router(config-router)#network 192.168.50.0
```

У цьому режимі маршрутизатор надсилає та отримує лише повідомлення RIPv2. Аналогічно, маршрутизатор можна налаштувати на надсилання та отримання лише повідомлень RIPv1:

```
Router(config)#router rip
Router(config-router)#version 1
Router(config-router)#network 172.25.0.0
Router(config-router)#network 192.168.50.0
```

Параметри за замовчуванням можна відновити, ввівши команду no version у режимі конфігурації маршрутизатора.

2.4 Автентифікація RIPv2

Реалізація Cisco автентифікації повідомлень RIPv2 включає вибір простого пароля або автентифікації MD5, а також можливість визначення декількох ключів або паролів. Маршрутизатор можна налаштувати на використання різних ключів у різний час.

Нижче наведено кроки для налаштування автентифікації RIPv2:

а) визначити ланцюг ключів з ім'ям;

б) визначити ключ на ланцюги ключів;

в) увімкнути автентифікацію на інтерфейсі та вказати ланцюг ключів, який буде використовуватися;

г) вказати, чи буде інтерфейс використовувати автентифікацію відкритим текстом або MD5;

ґ) налаштувати керування ключами.

У наступному прикладі на R1 налаштовано ланцюжок ключів з назвою Keys. Ключ 1, єдиний ключ на ланцюжку, має пароль passwd; інтерфейс використовує цей ключ з автентифікацією MD5 для перевірки оновлень від R2:

```
Router(config)#key chain Keys
Router(config-keychain)#key 1
Router(config-keychain-key)#key-string passwd
Router(config-keychain-key)#interface ethernet 0
Router(config-if)#ip rip authentication key-chain Keys
Router(config-if)#ip rip authentication mode md5
```

Ланцюг ключів повинен бути налаштований, навіть якщо на ньому лише один ключ. Хоча всі маршрутизатори, які будуть обмінюватися автентифікованими оновленнями, повинні мати однаковий пароль, ім'я ланцюга ключів має значення лише на локальному маршрутизаторі. Наприклад, R2 може мати ланцюг з ім'ям Keys2, але для зв'язку з R1 необхідно, щоб пароль був passwd.

2.5 Пошук та усунення несправностей RIPv2

Дві проблеми конфігурації, характерні для RIPv2 – це невідповідність версій і неправильно налаштована автентифікація. Обидві проблеми легко виявити командою ip rip events. Хоча RIPv2 має деякі суттєві покращення порівняно з RIPv1, він все ще обмежений максимальною швидкістю до 15 хопів, а отже, не підходить для невеликих мереж.

В таблиці 2.2 наведено список команд для конфігурації RIP2.

Таблиця 2.2 – Список команд для конфігурації RIP2

Команда	Опис
accept-lifetime	Використовується в контексті автентифікації маршрутизаторів, зокрема для ключів аутентифікації
auto-summary	Використовується в для ввімкнення або вимкнення автоматичної маршрутизованої агрегації (сумаризації) класових мереж
debug ip rip	В режимі реального часу показує, які RIP-пакети відправляються і приймаються, і що саме в них міститься
Ip classless	Використовується для вказівки маршрутизатору, як поводитися з неповними маршрутами
ip rip authentication key-chain	Використовується для ввімкнення аутентифікації на інтерфейсі, шляхом вказання ланцюжка ключів, який містить паролі (ключі), що використовуються для перевірки справжності RIP-пакетів
Ip rip authentication mode	Використовується на інтерфейсі маршрутизатора, щоб увімкнути режим аутентифікації та вказати тип аутентифікації
Ip rip receive version	Використовується для вказівки, яку версію RIP маршрутизатор повинен приймати на певному інтерфейсі
Ip rip send version	Використовується для вказівки, яку версію протоколу RIP маршрутизатор повинен відправляти на конкретному інтерфейсі
Ip split-horizon	Використовується для ввімкнення або вимкнення механізму «розділеного горизонту» (split horizon) на інтерфейсі маршрутизатора

Кінець таблиці 2.2

Ip subnet-zero	Дозволяє маршрутизатору використовувати нульову підмережу (subnet zero) – тобто, першу підмережу з усіма нулями в частині хоста
Key	Використовується для налаштування ключового ланцюжка
key chain	Використовується для створення ланцюжка ключів
key-string	Використовується для встановлення фактичного секретного ключа (пароля) всередині ключового ланцюжка
Network	Використовується для визначення, які мережі повинні брати участь у маршрутизації і які інтерфейси маршрутизатора будуть задіяні
Passive-interface	Забороняє надсилання маршрутних оновлень через вказаний інтерфейс
Router rip	Використовується для входу в конфігураційний режим маршрутизатора, де налаштовується протокол маршрутизації RIP
Send-lifetime	Використовується всередині ланцюга ключів для визначення часового періоду, протягом якого ключ буде активно використовуватися для відправки аутентифікаційних повідомлень
Show ip route	Використовується для відображення таблиці маршрутизації Ipv4 на маршрутизаторі
Version	Використовується для вибору версії протоколу RIP, яку маршрутизатор буде використовувати

2.6 Конфігурація OSPF

Багато опцій і змінних конфігурації, доступних для OSPF, часто роблять його очевидним вибором у великих IP-мережах. Однак іноді висловлюється думка, що конфігурація OSPF «занадто складна», щоб бути гарним вибором для невеликих мереж. Так, конфігурація цього протокола трохи складніша чим конфігурація RIP, але він також підходить до невеликих мереж.

Три кроки, необхідні для запуску базового процесу OSPF:

а) визначити область, до якої буде приєднано кожен інтерфейс маршрутизатора;

б) увімкнути OSPF за допомогою команди router ospf id;

в) вказати інтерфейси, на яких буде запущено OSPF, і їх області за допомогою команди network area.

На відміну від ідентифікатора процесу, пов'язаного з IGRP і EIGRP, ідентифікатор процесу OSPF не є автономним системним номером. Ідентифікатор процесу може бути будь-яким натуральним числом і не має значення за межами маршрутизатора, на якому він налаштований. Cisco IOS дозволяє запускати декілька процесів OSPF на одному маршрутизаторі. Ідентифікатор процесу просто відрізняє один процес від іншого в межах пристрою.

Команда network, яка використовується з раніше розглянутим протоколам, дозволяє вказати лише головну мережеву адресу. Якщо деякі інтерфейси у мережі не повинні виконувати протокол маршрутизації, з цими протоколами слід використовувати команду passive-interface. Команда network area є набагато гнучкішою, що відображає повністю безкласову природу OSPF. Будь-який діапазон адрес можна вказати за допомогою пари (адреса, зворотна маска). Область можна вказати у десятковому вигляді.

Конфігурація для кожного маршрутизатора в мережі (рис. 2.1).

Для маршрутизатора R1:

```
Router(config)#router ospf 10
Router(config-router)#network 10.33.5.0 0.0.15.255 area 0
Router(config-router)#network 192.168.83.0 0.0.0.255 area 0
Router(config-router)#network 192.168.12.192 0.0.0.31 area 0
```

Для маршрутизатора R2:

```
Router(config)#router ospf 20
Router(config-router)#network 192.168.12.64 0.0.0.31 area 0
Router(config-router)#network 192.168.12.192 0.0.0.31 area 0
Router(config-router)#network 172.17.1.0 0.0.0.255 area 0
Router(config-router)#network 172.17.2.0 0.0.0.255 area 0
```

Для маршрутизатора R3:

```
Router(config)#router ospf 30
Router(config-router)#network 172.17.1.0 0.0.0.255 area 0
Router(config-router)#network 172.17.3.0 0.0.0.255 area 0
```

Для маршрутизатора R4:

```
Router(config)#router ospf 40
Router(config-router)#network 172.17.2.0 0.0.0.255 area 0
Router(config-router)#network 172.17.4.0 0.0.0.255 area 0
```

Перше, на що слід звернути увагу, це те, що ідентифікатори процесів різні для кожного маршрутизатора. Зазвичай ці номери однакові в усьому інтернеті для уніфікації конфігурації. Тут ідентифікатори процесів налаштовані по-різному лише для того, щоб продемонструвати, що вони не мають жодного значення за межами маршрутизатора. Ці чотири процеси з різними номерами можуть взаємодіяти. Наступне, на що слід звернути увагу, - це формат команди network area. Після мережевої частини йде IP-адреса та інверсна маска. Коли процес OSPF вперше стає активним, він «перевірить» IP-адреси всіх активних інтерфейсів на відповідність пари (адреса, інверсна

маска) першої команди. Всі інтерфейси, які збігаються, будуть призначені до області, вказаної у частині команди area. Після цього процес перевірить адреси всіх інтерфейсів, які не збігаються з першою командою, на відповідність другій команді. Процес перевірки IP-адрес на відповідність командам триває доти, доки не буде знайдено відповідність для всіх інтерфейсів або доки не буде використано всі команди. Важливо зазначити, що цей процес відбувається послідовно, починаючи з першої команди.

2.7 Пошук та усунення несправностей OSPF

Вирішення проблем з OSPF іноді може бути складним, особливо в великій мережі. Однак проблема маршрутизації в OSPF не відрізняється від проблем з будь-яким іншим протоколом маршрутизації; причина завжди буде однією з наступних:

а) відсутня інформація про маршрут;

б) некоректна (неточна) інформація про маршрут.

Перевірка таблиці маршрутів залишається основним джерелом інформації для усунення несправностей. Використання команди show ip ospf database для перегляду різних LSA також дасть важливу інформацію. Наприклад, якщо посилання нестабільне, LSA, який його оголошує, буде часто змінюватися. Цей стан відображається у номері послідовності, який помітно вищий за номери інших LSA. Ще одним знаком нестабільності є LSA, час, що минув від останнього оновлення залишається низьким.

Потрібно мати на увазі, що база даних LS кожного маршрутизатора в межах області однакова. Тому, якщо є підозра, що сама база даних пошкоджена на деяких маршрутизаторах, можна перевірити базу даних LS для всієї області, дослідивши базу даних LS одного маршрутизатора. Ще одна хороша практика – зберігати копію бази даних стану з'єднань для кожної області.

Якщо є підозра, що база даних стану з'єднання пошкоджена або що дві бази даних не синхронізовані, можна скористатися командою show ip ospf data base database-summary, щоб побачити кількість LSA в базі даних кожного маршрутизатора. Для даної області кількість кожного типу LSA має бути однаковою на всіх маршрутизаторах. Далі команда show ip ospf database покаже контрольні суми для кожного LSA в базі даних маршрутизатора. У межах певної області контрольна сума кожного LSA має бути однаковою в базі даних кожного маршрутизатора.

В таблиці 2.3 наведено список команд для конфігурації OSPF.

Таблиця 2.3 – Список команд для конфігурації OSPF

Команда	Опис
Area Authentication	Використовується для увімкнення аутентифікації на рівні області
area default-cost	Використовується для задання дефолтної вартості (cost) маршруту за замовчуванням, який генерується маршрутизатором як Summary LSA для певної області
area nssa	Використовується для налаштування області як NSSA (Not-So-Stubby Area) — особливого типу області з обмеженим маршрутизуючим трафіком
area range	Використовується для агрегації (сумування) мережевих адрес, які належать до певної області, і оголошення їх у вигляді одного сумарного маршруту за межами цієї області
area area-id	Використовується для визначення і налаштування області маршрутизатора у процесі OSPF
area virtual-link	Використовується для налаштування віртуального лінку (virtual link) між двома маршрутизаторами OSPF через проміжну область

Продовження таблиці 2.3

debug ip ospf adj	Використовується на маршрутизаторах для виведення детальної інформації про встановлення і стан OSPF-суміжностей
ip ospf authentication-key	Використовується для налаштування ключа аутентифікації на конкретному інтерфейсі в OSPF, щоб забезпечити просту (text-based) аутентифікацію OSPF пакетів
ip ospf cost	Використовується для ручного встановлення вартості (cost) інтерфейсу в OSPF
ip ospf dead-interval	Використовується для налаштування часу очікування (dead interval) OSPF інтерфейсу
ip ospf demand-circuit	Використовується для увімкнення режиму demand circuit (режиму вимоги) на інтерфейсі OSPF
ip ospf hello-interval	Використовується для налаштування інтервалу надсилання Hello-пакетів на інтерфейсі OSPF
ip ospf message-digest-key	Використовується для налаштування MD5-аутентифікації на інтерфейсі OSPF
ip ospf network	Використовується для налаштування типу мережі OSPF на конкретному інтерфейсі
ip ospf priority	Використовується для налаштування пріоритету інтерфейсу в процесі вибору DR (Designated Router) та BDR (Backup Designated Router) у OSPF
ip ospf retransmit-interval	Налаштовує інтервал часу (у секундах), через який OSPF повторно надсилає LSA (Link-State Advertisement), якщо не отримано підтвердження від сусіда
ip ospf transmit-delay	Налаштовує час (у секундах), який OSPF враховує як затримку передачі LSA по інтерфейсу

Продовження таблиці 2.3

maximum-paths	Визначає максимальну кількість рівноцінних маршрутів (equal-cost multipath, ECMP), які маршрутизатор може використовувати одночасно для маршрутизації трафіку
neighbor	Використовується лише для мереж типу non-broadcast (NBMA) або point-to-multipoint non-broadcast, де OSPF не може автоматично виявити сусідів
network area	Використовується для визначення, які інтерфейси маршрутизатора братимуть участь в OSPF та призначення цих інтерфейсів до певної зони
ospf auto-cost reference-bandwidth	Використовується для налаштування еталонної пропускної здатності (reference bandwidth) в OSPF, щоб правильно обчислювати метрику (cost) на основі швидкості інтерфейсів
ospf log-adjacency-changes	Використовується для логування змін стану OSPF-суміжності (adjacency) в системному журналі маршрутизатора
router ospf	Використовується для запуску OSPF-процесу маршрутизації
show ip ospf	Використовується на маршрутизаторах для перегляду загальної інформації про OSPF-процеси, які запущені на пристрої
show ip ospf border-routers	Використовується на маршрутизаторах для перегляду списку прикордонних маршрутизаторів OSPF (ABR та ASBR), які відомі в топології OSPF
show ip ospf database	Відображає LSA-базу даних OSPF

Кінець таблиці 2.3

show ip ospf database router	Використовується для перегляду Router LSAs (тип 1) у базі даних OSPF
show ip ospf database network	Використовується для перегляду Network LSAs (тип 2) у базі даних OSPF
show ip ospf database summary	Використовується для перегляду Summary LSAs (тип 3 і тип 4) у базі даних OSPF
show ip ospf database asbr-summary	Використовується на маршрутизаторах для перегляду таблиці бази даних OSPF, зокрема – ASBR Summary LSA (тип 3 або тип 4, залежно від контексту)
show ip ospf database nssa-external	Використовується на маршрутизаторах для перегляду NSSA External LSAs (тип 7) у базі даних OSPF
show ip ospf database external	Використовується на маршрутизаторах для перегляду типу 5 LSA (AS External LSA) у базі даних OSPF
show ip ospf	Використовується для перегляду загального статусу процесу OSPF
show ip ospf interface	Використовується на маршрутизаторах для перегляду детальної інформації про кожен інтерфейс
show ip ospf neighbor	Відображає інформацію про сусідів OSPF, з якими маршрутизатор встановив (або намагається встановити) OSPF-суміжність (neighbor adjacency)
show ip ospf virtual-links	Використовується для перегляду стану та параметрів віртуальних лінків OSPF (Virtual Links)
timers lsa-group-pacing	Використовується для контролю частоти групових оновлень LSAs (Link State Advertisements)

3 МОДЕЛЮВАННЯ ТА СИМУЛЯЦІЯ МЕРЕЖІ З КОНВЕРГЕНЦІЄЮ

3.1 Програмне забезпечення для моделювання

На сьогоднішній день практично неможливо спроектувати цілісну мережеву систему лише на основі теоретичних розрахунків. Однак, якщо ми проводимо дослідження, проектування та розробку в реальному мережевому середовищі, ми не тільки понесемо великі витрати, але й матимемо труднощі зі збором та аналізом даних. У практичній роботі переважає використання програмного забезпечення для моделювання мереж для моделювання та оцінки продуктивності мережі. Таке програмне забезпечення, як NS3, OMNET++ і OPNET, може регулювати параметри мережі в змодельованому середовищі для досягнення максимального використання. Порівняно з NS3 та OMNET++, OPNET є більш надійним і має потужні вбудовані модулі, які включають різноманітні прикладні протоколи, а також моделі реального комунікаційного обладнання. Щоб запустити симуляцію, користувачам потрібно лише вибрати відповідні моделі і зв'язати їх відповідним чином у редакторі з графічним інтерфейсом користувача. У цій роботі для моделювання мережі з конвергенцією використовується найпопулярніше програмне забезпечення для моделювання мереж OPNET Modeler (рис. 3.1). Його використовують для різних цілей в різних областях: моделювання корпоративних мереж можна використовувати всі типи пристроїв для побудови конкретної мережі. Якщо час відгуку операцій, таких як онлайн-транзакції, база даних та інші, повільніший, ніж у звичайних умовах, Modeler може знайти вузькі місця в роботі сервісів, мережі та серверів шляхом захоплення та аналізу критичного потоку даних; для мереж провайдерів Modeler фокусується на моделюванні послуг та трафіку, щоб провайдер міг ефективно виявляти помилки в конфігурації; крім того, Modeler надає

відкрите середовище, яке дозволяє користувачам створювати нові протоколи та пристрої, визначати та моделювати всі їхні деталі для дослідницьких цілей.

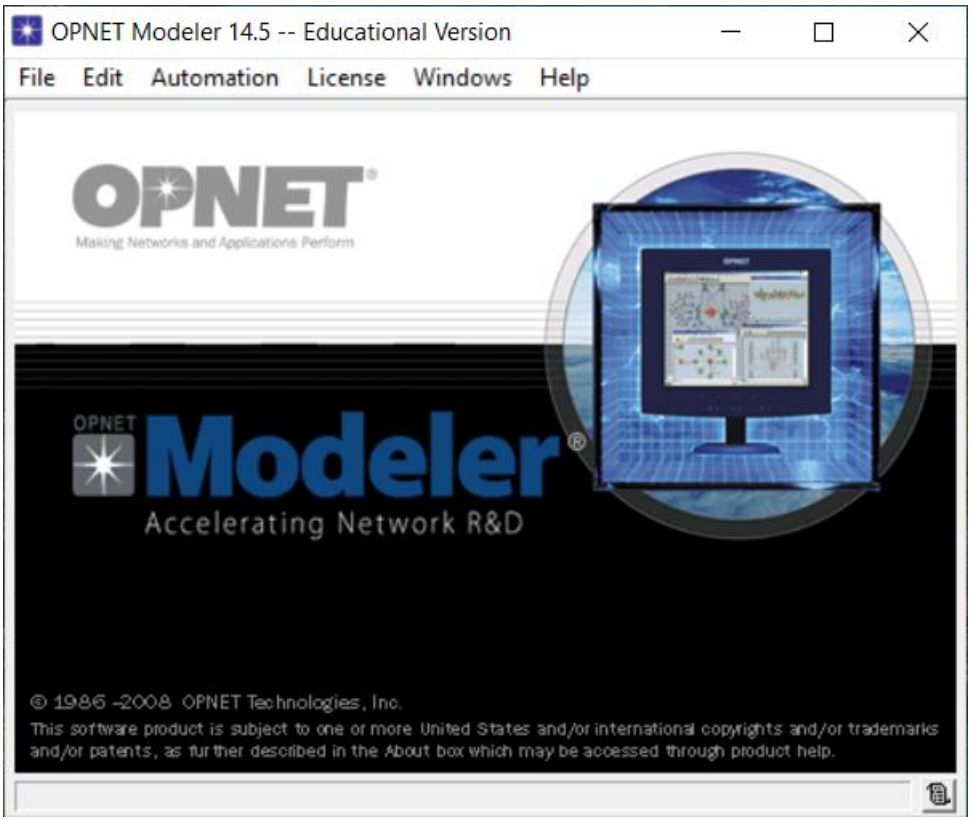

Рисунок 3.1 – Графічний інтерфейс OPNET Modeler 14.5

OPNET Modeler підтримує багаторівневе моделювання та надає широкий спектр інтерфейсів для розробки. Він має наступні можливості:

а) ієрархічне моделювання мережі. З точки зору протоколу, моделювання вузлів відповідає моделі OSI: прикладний (Application Layer), представницький (Presentation Layer), сеансовий (Session Layer), транспортний (Transport Layer), мережний (Network Layer), канальний (Data Link Layer), фізичний (Physical Layer);

б) прості механізми моделювання. Процес моделювання можна розділити на три рівні: нижній рівень – це модель процесу, що використовує кінцевий автомат для опису протоколів; другий рівень – це модель вузла,

побудована за відповідною моделлю протоколу, що відображає характеристики пристрою; верхній рівень – це мережева модель. Вона складається з вузлів і зв'язків між вузлами. Топологія мережі може бути налаштована безпосередньо на цьому рівні моделі. Трирівневі моделі повністю відповідають реальному протоколу, пристрою і мережі, таким чином, відображають кожну особливість мережі;

в) Finite State Machine (FSM) Modeler використовує потужний підхід скінченних автоматів для підтримки детальної специфікації протоколів, ресурсів, додатків, алгоритмів і правил обробки черг. Користувачі можуть включити аналітичну модель, запрограмувавши її за допомогою мови C, і, нарешті, FSM є представником потоку коду для конкретного рівня процесу (буфер, процесор і т.д.);

г) зручний та широкий спектр протоколів OPNET Modeler надає до 400 бібліотечних функцій та лаконічну модель протоколу. Велика кількість широко використовуваних протоколів вбудована в ядро системи, і їх легко використовувати без будь-якого програмування;

ґ) підтримка різноманітних прикладних моделей В OPNET попередньо визначені майже всі найпоширеніші прикладні моделі, включаючи однорідний розподіл, пуассонівський розподіл, біноміальний розподіл та інші 22 розподіли. Користувачі також можуть зробити вибірку реальних операцій і перетворити її на функцію щільності ймовірності в редакторі, що надається OPNET для введення даних для моделювання;

д) повна відкритість. Весь код і протоколи повністю відкриті для користувача, і кожен вузол має чітке позначення, що дозволяє легко налаштовувати і модифікувати моделі, включаючи функції для легкого створення моделей з нуля.

3.2 Модель мережі

Модель складається з наступних вузлів: локальні мережі (LN1 та LN2), маршрутизатори (R1-R5) та спеціальний вузол відновлення після збоїв (Failure Recovery) (рис. 3.2).

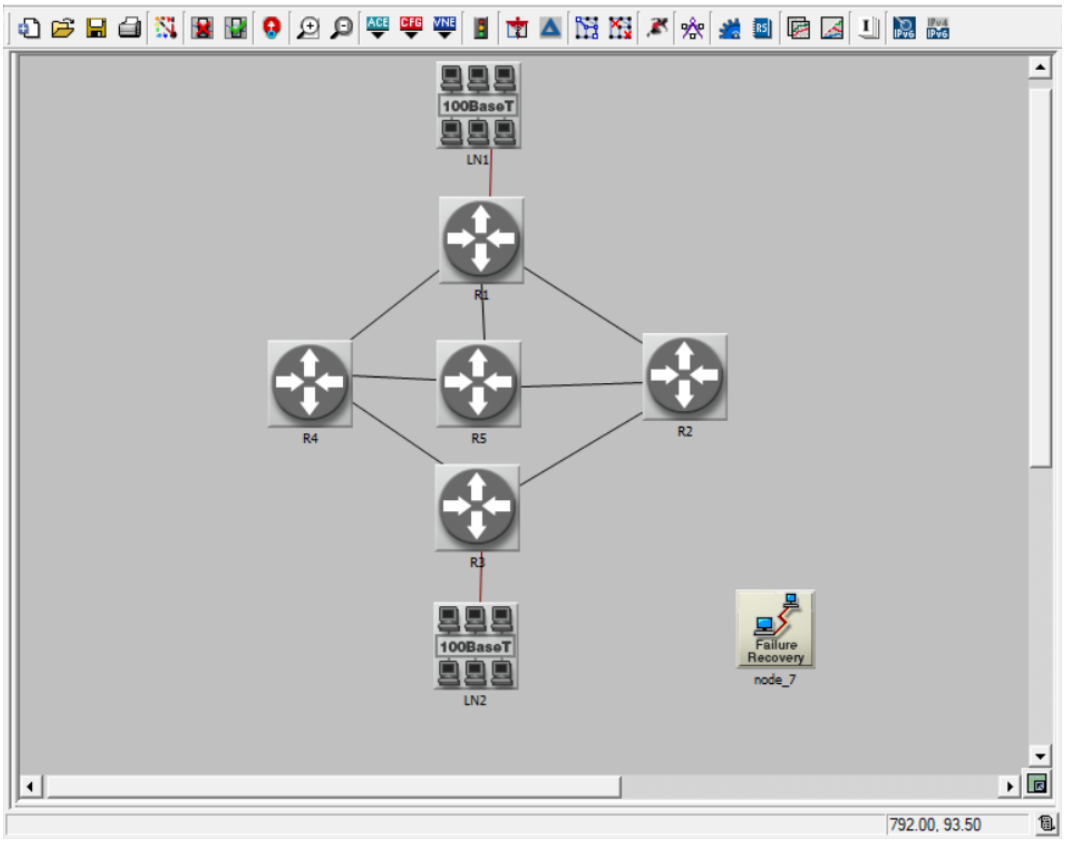

Рисунок 3.2 – Модель мережі

Маршрутизатори з'єднанні технологією PPP/DS3. Ця мережева технологія, яка використовує протокол «точка-точка» (PPP) на фізичному рівні DS3 (Digital Signal 3). DS3 – це тип мережевого кабелю, який використовується для передачі цифрових сигналів на високій швидкості, зазвичай 44,736 Мбіт/с. PPP забезпечує спосіб встановлення з'єднання «точка-точка», в той час як DS3 забезпечує базовий фізичний рівень для цього з'єднання. У мережевій інфраструктурі з'єднання PPP/DS3 використовуються для з'єднання різних офісів. Вузли 100Base-T – це

представлення локальної мережі Fast Ethernet у комутованій топології. Кількість робочих станцій – 10, а швидкість комутації – 500,000pkts/sec. Швидкість комутації визначає час обслуговування кожного пакета під час проходження через комутаційний модуль цього вузла. Всі пакети, тобто вхідні, вихідні та транзитні, підлягають цій затримці. Крім того, вхідні та вихідні пакети зазнають затримки при передачі, яка базується на швидкості передачі даних, змодельованій у вузлі, а також затримок на вищих рівнях, таких як затримки на рівні IP та прикладних програм, якщо це можливо. Вузли LN1 та LN2 з'єднанні з маршрутизаторами R1 та R3 відповідно кабелем 100BaseT. Це дуплексне з'єднання Ethernet, що працює на швидкості 100 Мбіт/с. Вузол відновлення після збоїв сконфігурований на розрив зв'язку та відновлення між маршрутизаторами R1 з R2 та R1 з R4 (рис. 3.3).

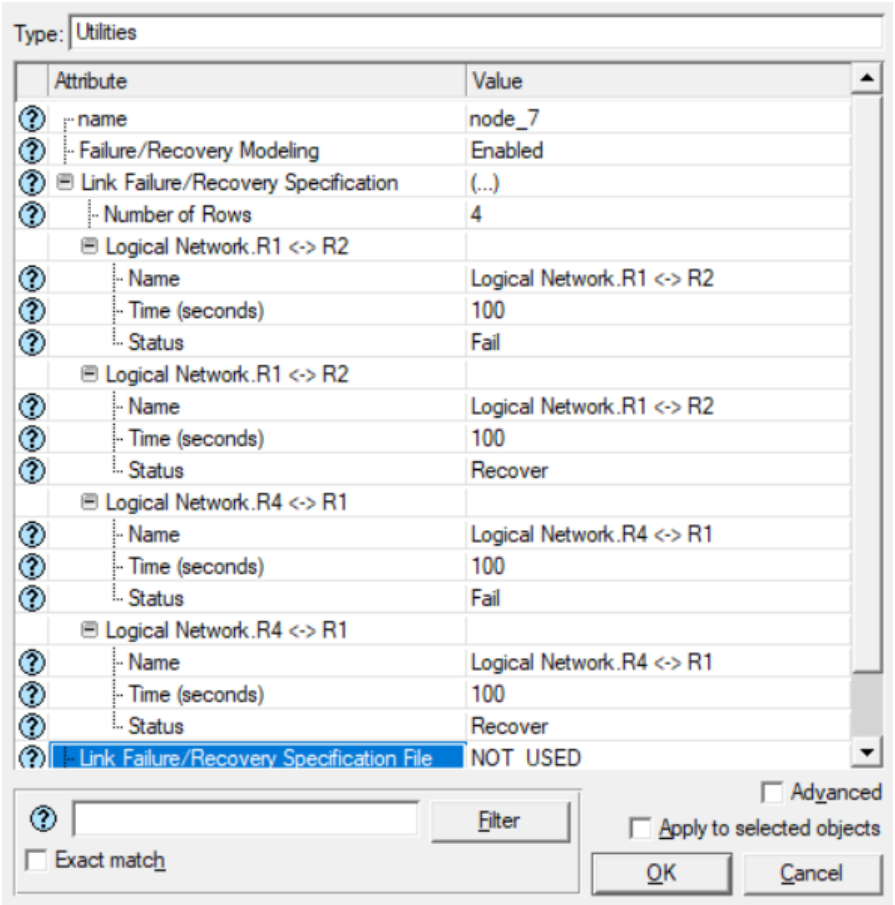

Рисунок 3.3 – Конфігурація вузла Failure Recovery

У моделі реалізовано два незалежні сценарії:

а) перший сценарій використовує протокол RIP, який базується на алгоритмі відстань-вектор;

б) другий – OSPF, що належить до класу протоколів стану каналу.

Обидва сценарії дозволяють провести порівняння часу конвергенції, тобто періоду, необхідного мережі для відновлення стабільного стану після внесення змін (наприклад, відмови одного з маршрутизаторів або інтерфейсів).

Для оцінки ефективності роботи протоколів динамічної маршрутизації в умовах зміни топології мережі, в OPNET Modeler були обрані відповідні глобальні та локальні статистичні параметри, що дозволяють дослідити як сам процес конвергенції, так і загальну активність вузлів. Ці параметри можна включити у вікні Choose Individual Statistics.

Глобальні статистики (Global Statistics): у категорії IP → Network Convergence Time було активовано параметр Network Convergence, який відображає загальний час, необхідний мережі для досягнення стабільного стану після змін у топології (наприклад, відмова каналу або вузла). Цей показник є ключовим при порівнянні протоколів RIP та OSPF, оскільки дозволяє кількісно оцінити, наскільки швидко відновлюється маршрутизація.

Статистика на рівні вузлів (Node Statistics): у категорії IP для окремих маршрутизаторів були вибрані наступні параметри:

а) Traffic Sent – кількість IP-пакетів, надісланих даним маршрутизатором;

б) Traffic Received – кількість IP-пакетів, отриманих маршрутизатором.

Ці показники дозволяють відстежити обмін трафіком між вузлами до, під час та після конвергенції, а також виявити можливі втрати трафіку внаслідок затримки оновлення маршрутів.

Статистика таблиць маршрутизації: також було активовано параметр Route Table → Total Number of Updates, що відображає загальну кількість

оновлень маршрутної таблиці, які були виконані маршрутизатором у процесі симуляції. Цей показник дає уявлення про частоту обміну інформацією.

У рамках дослідження в середовищі OPNET Modeler була змодельована ідентична топологія для двох протоколів – RIP та OSPF, з однаковими параметрами мережі та часовим інтервалом симуляції – 10 хвилин (600 секунд) для кожного (рис. 3.4).

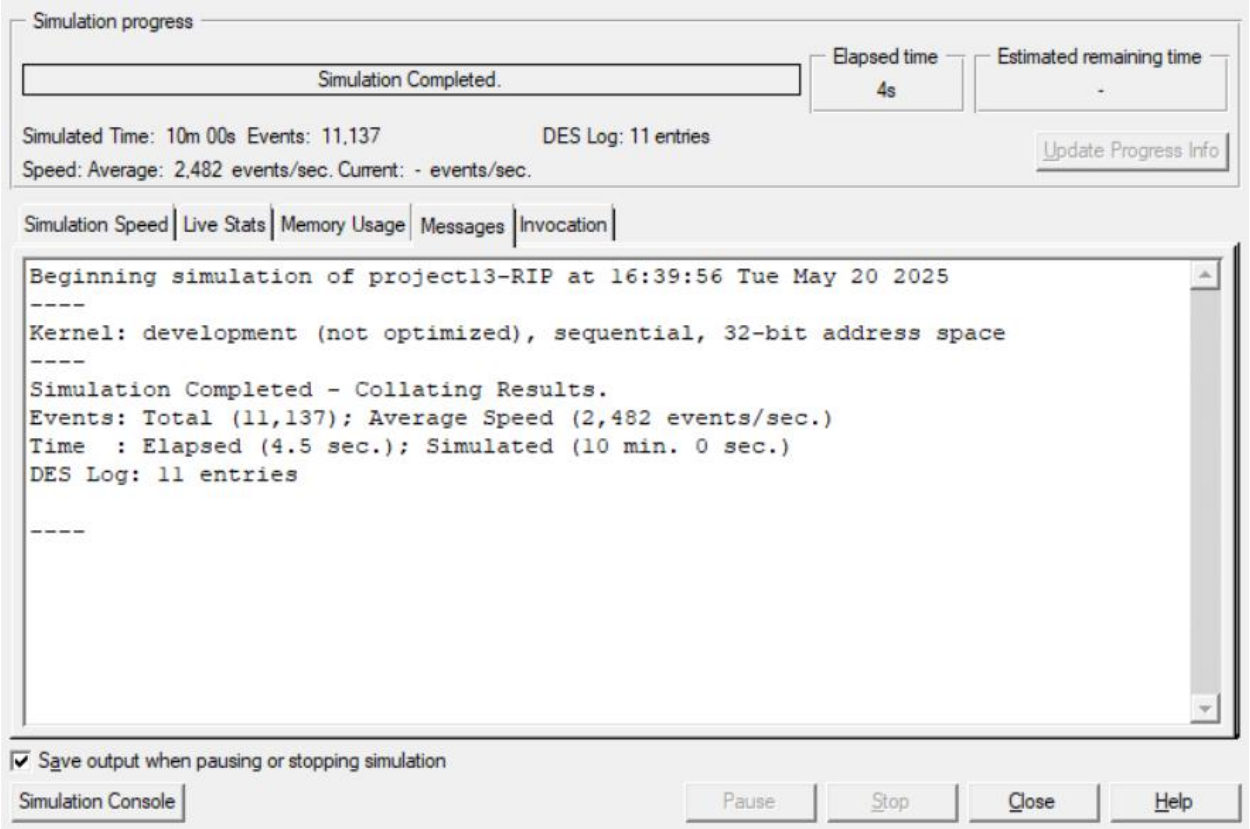

Рисунок 3.4 – Процес симуляції

На основі графіка (нижче), який демонструє середній час конвергенції для протоколів RIP (червона лінія) та OSPF (синя лінія), можна зробити такі спостереження.

На початку симуляції час конвергенції для мережі з протоколом RIP досягає близько 10.8 секунд. Це пов’язано з тим, що RIP оновлює свої таблиці періодично кожні 30 секунд (рис. 3.5) та не має повної інформації про

топологію мережі. Далі значення поступово знижується, але залишається в межах 7.5-5.5 секунд навіть через декілька хвилин після запуску моделі. Це свідчить про повільне оновлення маршрутної інформації та обмеження алгоритму Bellman-Ford, на якому базується RIP. У разі зміни топології (наприклад, при відмові маршрутизатора або лінку) RIP має затримку у виявленні змін та розповсюдженні нових маршрутів, що може негативно впливати на якість обслуговування в реальній мережі (рис. 3.6).

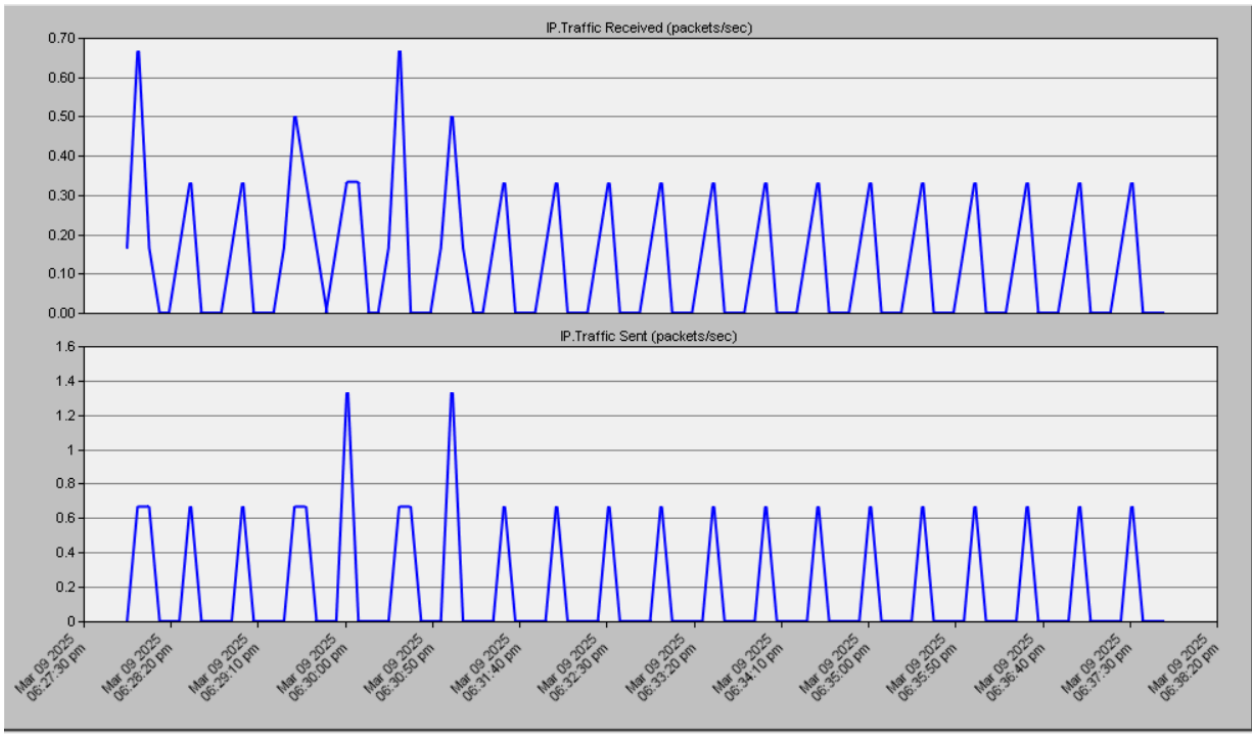

Рисунок 3.5 – Трафік на маршрутизаторі R1 для протокола RIP

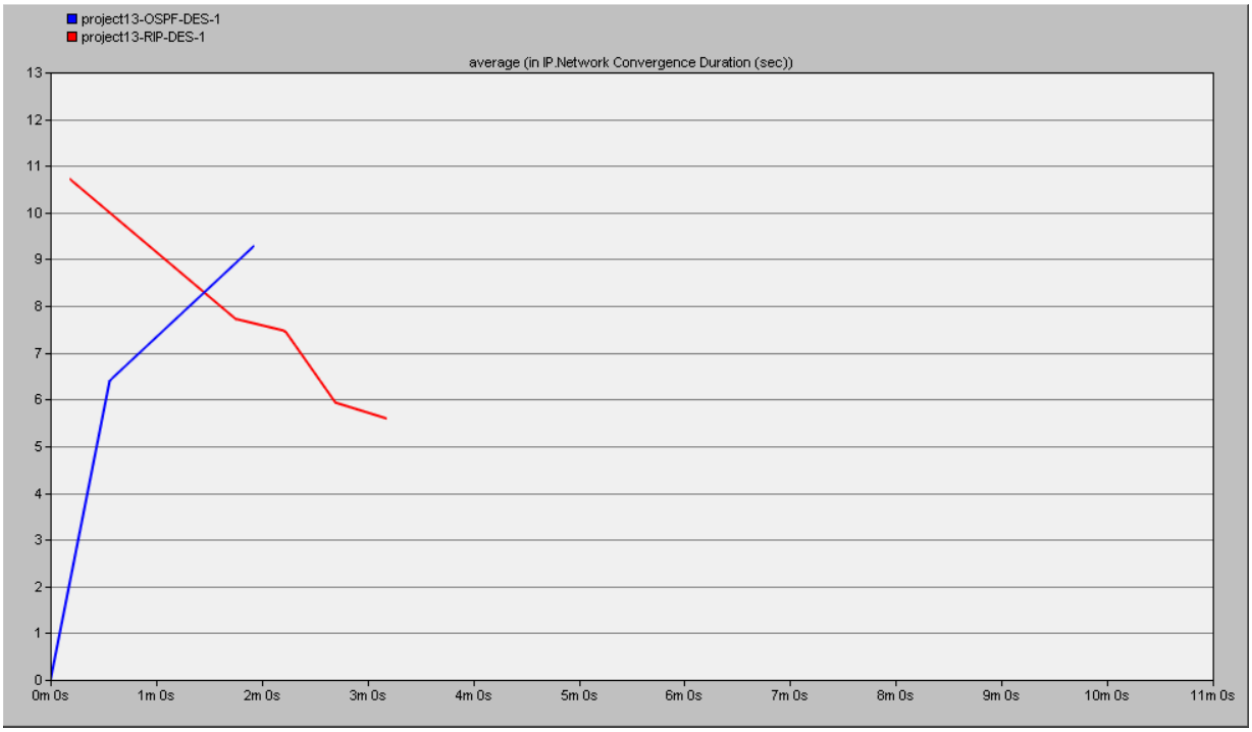

Рисунок 3.6 – Результати часу конвергенції

Час конвергенції в OSPF на початку нижчий – приблизно 6.3 секунд, і поступово зменшується до 4.5-5 секунд на подальших етапах моделювання. OSPF використовує стан каналу (link-state) та алгоритм Дейкстри (Shortest Path First), завдяки чому маршрутизатори мають повну карту мережі та можуть швидко перебудувати маршрути після події. Щойно виявляється зміна (через Hello-протокол або LSA-повідомлення), оновлення передається миттєво по всій області, і нові маршрути обчислюються локально, що значно зменшує час відгуку. OSPF, як протокол стану каналу, генерує велику кількість трафіку на етапі ініціалізації (рис. 3.7). Це видно на графіку: на початку симуляції спостерігаються піки до 3.8 пакетів/с. Після завершення початкової побудови топології та розрахунку SPF-дерева, трафік різко зменшується і стабілізується приблизно на рівні 0.4 пакетів/с. Це пов’язано з тим, що OSPF передає оновлення тільки при виявленні змін у топології, або періодично через LSA refresh. Таким чином, початкове навантаження —

високе, але в подальшому OSPF поводиться дуже ефективно, мінімізуючи службовий трафік.

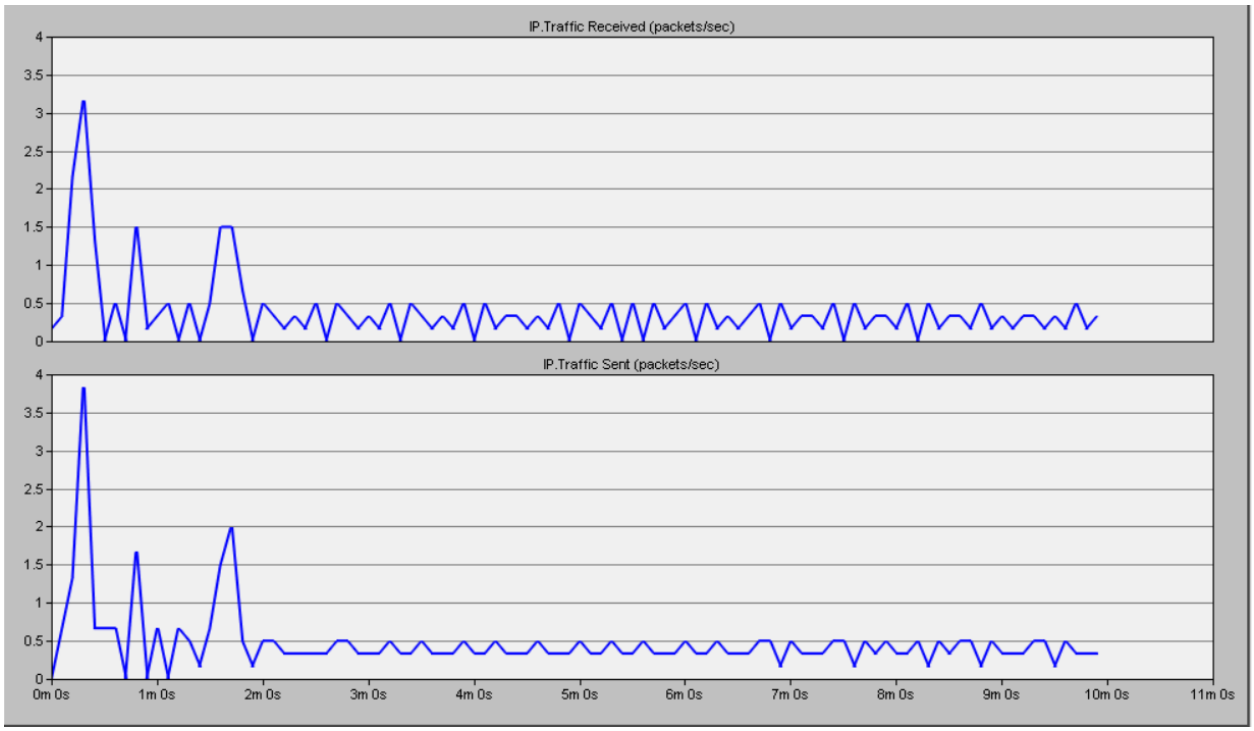

Рисунок 3.7 – Трафік на маршрутизаторі R1 для протокола OSPF

Аналогічна ситуація спостерігається також на інших маршрутизаторах (рис. 3.8, 3.9).

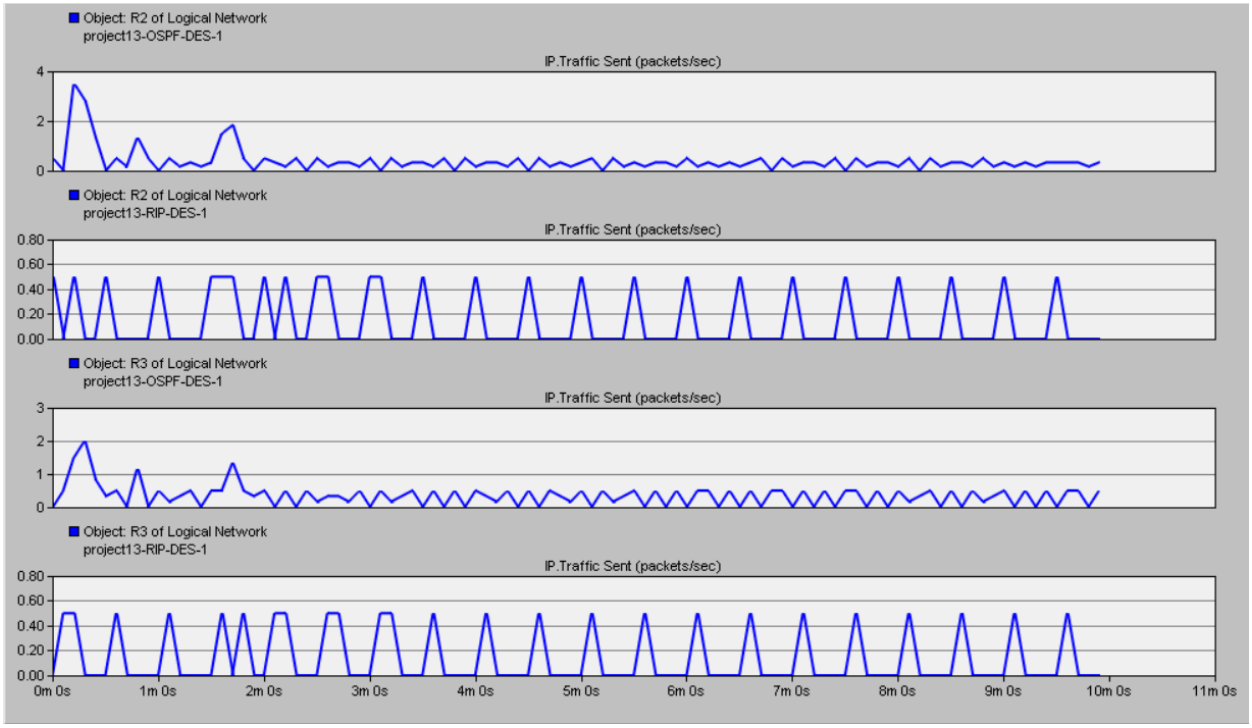

Рисунок 3.8 – Трафік на маршрутизаторах R2 та R3

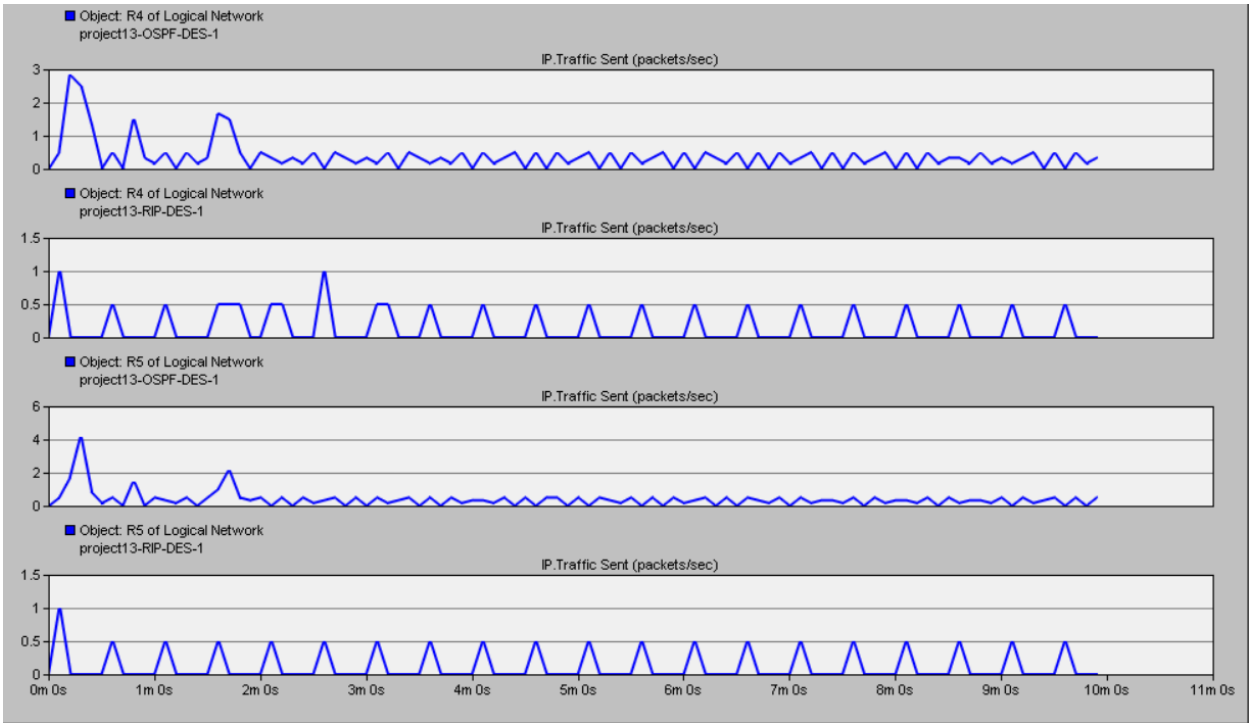

Рисунок 3.9 – Трафік на маршрутизаторах R4 та R5

3.3 Таблиці маршрутизації

Таблиця маршрутизації – це ключовий компонент у роботі маршрутизатора, який використовується для визначення найкращого шляху доставки пакетів у мережі. Вона містить набір записів (маршрутів), які вказують, як дістатися до різних мереж або вузлів.

Кожен запис у таблиці маршрутизації зазвичай містить такі поля:

а) мережеву адресу призначення (Destination Network);

б) маску підмережі;

в) адресу наступного стрибка (Next Hop) або вихідний інтерфейс;

г) вартість маршруту (Metric) – показник ефективності маршруту (наприклад, кількість хопів, затримка, пропускна здатність тощо);

ґ) тип маршруту (динамічний або статичний);

д) протокол, який встановив маршрут (наприклад, RIP, OSPF, EIGRP тощо).

Маршрутизатор використовує таблицю маршрутизації щоразу, коли надходить IP-пакет. Він перевіряє адресу призначення пакета, знаходить відповідний маршрут у таблиці і передає пакет далі – або до наступного маршрутизатора, або безпосередньо до кінцевого пристрою. Таким чином, таблиці маршрутизації є основою ефективного, надійного і масштабованого передавання даних у комп’ютерних мережах.

У ході експериментального моделювання було також здійснено порівняльний аналіз протоколів маршрутизації OSPF та RIP шляхом моніторингу середньої кількості оновлень у таблиці маршрутів (Route Table: Total Number of Updates).

Ці таблиці відіграють ключову роль у процесі маршрутизації, оскільки зберігають найактуальнішу інформацію про топологію мережі. Оновлення в таблиці відбуваються щоразу, коли маршрутизатор дізнається про новий маршрут або коли наявний маршрут змінюється чи зникає (рис. 3.10).

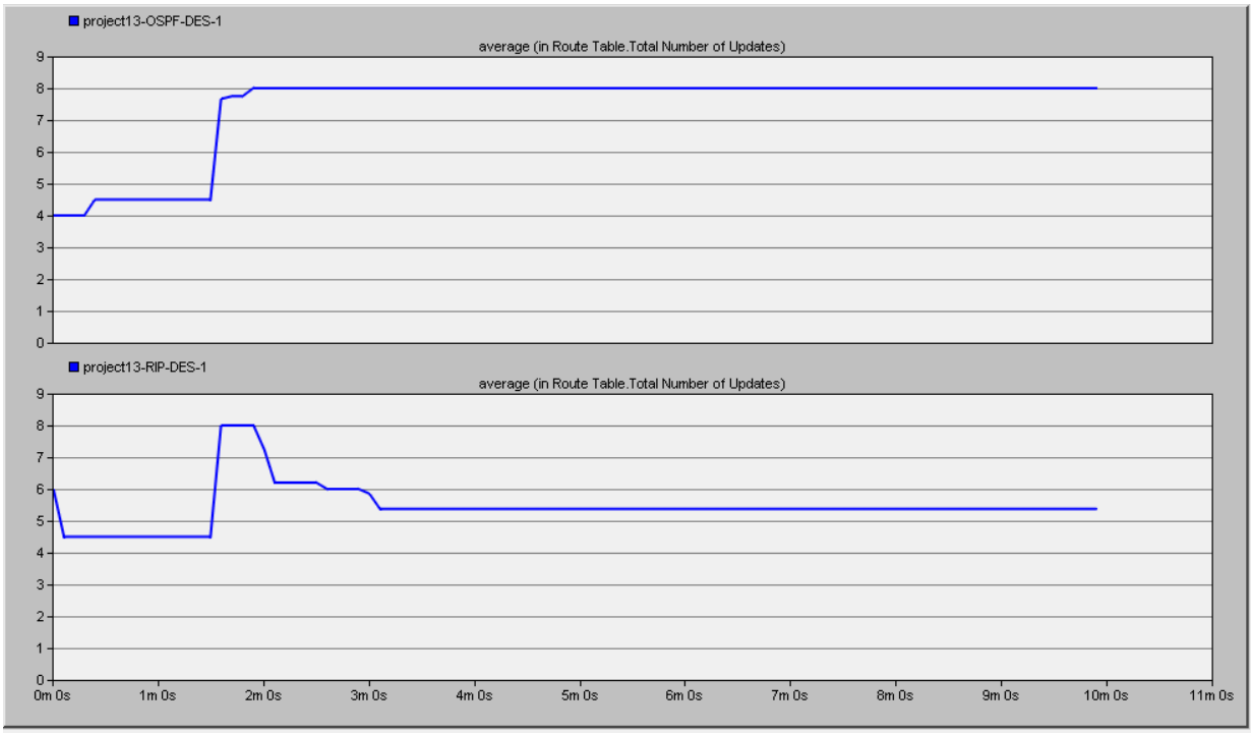

Рисунок 3.10 – Графік кількості оновлень таблиць маршрутизації

У випадку з OSPF ми спостерігаємо швидкий ріст кількості записів в таблиці оновлень, що досягла стабільного значення вже приблизно через 90 секунд. Після цього оновлення припинились, що свідчить про завершення фази конвергенції. Така поведінка є типовою для OSPF, оскільки він використовує стани каналів (link-state) та обмінюється лише змінами в топології, а не повними таблицями.

Натомість RIP демонструє більш хаотичну поведінку таблиці оновлень. Протягом перших кількох хвилин спостерігалися значні коливання кількості оновлень, а стабільне значення було досягнуто лише після приблизно 3 хвилин. Це зумовлено природою RIP, який базується на табличному обміні (distance-vector) та надсилає повні оновлення маршрутів кожні 30 секунд. Як наслідок, це створює додаткове навантаження на мережу та уповільнює конвергенцію.

ВИСНОВКИ

У процесі виконання дипломної роботи було проведено дослідження особливостей функціонування протоколів динамічної маршрутизації RIP та OSPF, а також здійснено порівняльний аналіз їхнього впливу на процес конвергенції в комп'ютерній мережі.

У результаті теоретичного аналізу було виявлено, що:

а) протокол RIP базується на алгоритмі відстані до призначення та має обмежену швидкість конвергенції через періодичну передачу маршрутної інформації та обмеження у кількості переходів (макс. 15 хопів);

б) протокол OSPF є більш сучасним і використовує алгоритм стану каналу (link-state), що забезпечує значно швидшу та надійнішу конвергенцію, особливо у великих або складних топологіях.

Практична частина роботи, реалізована в середовищі OPNET Modeler, продемонструвала, що:

а) у випадку зміни топології (наприклад, відмови маршрутизатора), час конвергенції OSPF був у кілька разів меншим, ніж у RIP;

б) OSPF забезпечує більш детальне бачення структури мережі, завдяки чому має кращу масштабованість і стабільність.

На основі проведеного дослідження можна зробити такі узагальнення:

а) протокол RIP є простим у налаштуванні та придатним для невеликих мереж, однак має значні обмеження щодо швидкості та ефективності конвергенції;

б) протокол OSPF є оптимальним вибором для середніх і великих мережевих середовищ, де критично важлива швидка реакція на зміни у мережевій інфраструктурі;

в) конвергенція є ключовим показником ефективності протоколу маршрутизації, від якого залежить надійність та якість обслуговування в мережі.

Таким чином, результати роботи можуть бути використані як основа для вибору протоколу маршрутизації при проєктуванні або оптимізації комп’ютерних мереж різного масштабу.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Cisco CCENT/CCNA ICND1 100-101. – Нью-Йорк: академічне видання 2015. – 903 с.

2. Cisco CCNA ICND2 200-101 Маршрутизація та комутація. – Нью-Йорк: академічне видання, 3, – 2015. – 737 с.

4. Tanenbaum A.S. Computer Networks / A.S. Tanenbaum, D. Wetherall – Pearson Education. – 2021. – P. 944.

5. Forouzan B**.** Data Communications and Networking / B. Forouzan – New York: McGraw-Hil. – 2013. – P. 1264.

6. RFC 1058 – Routing Information Protocol. [Електронний ресурс]. – Режим https://datatracker.ietf.org/doc/html/rfc1058 (дата звернення 20.03.2025) – Назва з екрану.

7. RFC 2328 – OSPF Version 2. [Електронний ресурс]. – Режим доступу: https://datatracker.ietf.org/doc/html/rfc2328 (дата звернення 22.03.2025) – Назва з екрану.

8. Cisco Systems. Configuring Routing Protocols. [Електронний ресурс]. – Режим доступу: https://www.cisco.com/c/en/us/td/docs/wireless/asr_901/Configuration/Guide/b_asr901-scg/b_asr901-scg_chapter_010010.pdf (дата звернення 10.04.2025) – Назва з екрану.

9. Cisco Systems. IP Routing: Protocol-Independent Configuration Guide, Cisco IOS Release 15S. [Електронний ресурс]. – Режим доступу: https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/iproute_pi/configuration/15-s/iri-15-s-book.pdf (дата звернення 28.05.2025) – Назва з екрану.

10. Doyle J. Routing TCP/IP, Volume 1 / J. Doyle, J. Carroll – Cisco Press, 2005. – P. 944.

11. Parkhurst W.R. Cisco OSPF Command and Configuration Handbook / W.R. Parkhurst. – Cisco Press, 2002. – P. 528.

12. Cisco Systems. Basic Router Configuration. [Електронний ресурс]. – Режим доступу: https://www.cisco.com/c/en/us/td/docs/routers/access/800M/software/800MSCG/routconf.html (дата звернення 02.06.2025) – Назва з екрану.