ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
КАФЕДРА КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ

КВАЛІФІКАЦІЙНА РОБОТА

на тему: «Дослідження методів оптимізації протоколу OSPF для підтримки динамічної маршрутизації»

на здобуття освітнього ступеня бакалавра
зі спеціальності 123 Комп'ютерна інженерія
(код, найменування спеціальності)
освітньо-професійної програми Комп'ютерна інженерія
(назва)

Кваліфікаційна робота містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело

_____ *Михайло ТОКАР*
(підпис) *Ім'я ПРІЗВИЩЕ здобувача*

Виконав: здобувач вищої освіти гр. КІД-42
Михайло ТОКАР
Ім'я, ПРІЗВИЩЕ
Керівник: Ігор БУЧЕНКО
науковий ступінь, вчене звання *Ім'я, ПРІЗВИЩЕ*
Рецензент: _____
науковий ступінь, вчене звання *Ім'я, ПРІЗВИЩЕ*

Київ 2024

ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ

Навчально-науковий інститут Інформаційних технологій

Кафедра Комп'ютерної інженерії
Ступінь вищої освіти бакалавр
Спеціальність 123 Комп'ютерна інженерія
Освітньо-професійна програма Комп'ютерна інженерія

ЗАТВЕРДЖУЮ
Завідувач кафедрою Наталія
ЛАЩЕВСЬКА Ім'я, ПРІЗВИЩЕ
«__» _____ 2024 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

Токар Михайло Віталійович
(прізвище, ім'я, по батькові здобувача)

1. Тема кваліфікаційної роботи: Дослідження методів оптимізації протоколу OSPF для підтримки динамічної маршрутизації

керівник кваліфікаційної роботи Ігор БУЧЕНКО,
(Ім'я, ПРІЗВИЩЕ, науковий ступінь, вчене звання)
затверджені наказом Державного університету інформаційно-комунікаційних технологій від « 27 » лютого 2024 р. № 36

2. Строк подання кваліфікаційної роботи «3» червня 2024 р.

3. Вихідні дані до кваліфікаційної роботи:
1) теза за дослідженням на тему: «Сучасні методи оптимізації протоколу OSPF»;
2) теза за дослідженням на тему: «Вдосконалення протоколу OSPF для підтримки безпечної маршрутизації».

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)
1. Питання огляду протоколу OSPF та теоретичних основ динамічної маршрутизації.
2. Питання щодо аналізу недоліків протоколу OSPF.
3. Питання аналізу та порівняння методів оптимізації протоколу OSPF.

5. Перелік ілюстративного матеріалу: *презентація*

6. Дата видачі завдання «27» лютого 2024 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Збір даних	29.02-23.03.2024	Виконано
2	Обробка матеріалу	24.03-06.04.2024	Виконано
3	Розробка теоретичної частини	07.04-10.04.2024	Виконано
4	Дослідження проблемних питань	11.04-03.05.2024	Виконано
5	Висновки за результатами аналізу	04.05-08.05.2024	Виконано
6	Розробка презентації	09.05-10.05.2024	Виконано
7	Передзахист	13.05-17.05.2024	Виконано
8	Здача в деканат	24.05-03.06.2024	Виконано

Здобувач вищої освіти _____ Михайло ТОКАР
(підпис) *(ім'я, ПРІЗВИЩЕ)*

Керівник
кваліфікаційної роботи _____ Ігор БУЧЕНКО
(підпис) *(Ім'я, ПРІЗВИЩЕ)*

РЕФЕРАТ

Текстова частина кваліфікаційної роботи на здобуття освітнього ступеня бакалавра: 73 стор., 51 рис., 32 джерела.

Мета роботи – дослідження методів оптимізації протоколу OSPF для підтримки динамічної маршрутизації на поточному етапі розвитку технологій.

Об'єкт дослідження – методи оптимізації протоколу OSPF для підтримки динамічної маршрутизації в комп'ютерних інформаційних мережах.

Предмет дослідження – методи оптимізації протоколу OSPF для підтримки динамічної маршрутизації.

Короткий зміст роботи: В сучасних комп'ютерних мережах на цей момент використовується чимала кількість методів оптимізації протоколу OSPF для підтримки динамічної маршрутизації, що залучені в багатьох різних за призначенням мережах. Використання методів оптимізації протоколу OSPF для підтримки динамічної маршрутизації наразі й в подальшому розвитку мереж дозволяють покращити пропускну спроможність мереж організацій і підприємств, що використовують дані методи. Використання методів оптимізації протоколу OSPF для підтримки динамічної маршрутизації значно збільшує ефективність використання обчислювальних ресурсів обладнання, що також дозволяє значно зменшити затримки в мережі. Ці методи оптимізації протоколу OSPF мають велику кількість переваг, проте, також, і ряд недоліків, що вимагають від користувачів достатньої кваліфікованості для користування ними.

На основі проведеного дослідження було виявлено та проаналізовано чималу кількість рішень для оптимізації протоколу OSPF, що дозволяє зробити точні висновки про використання методів оптимізації протоколу OSPF для підтримки динамічної маршрутизації.

КЛЮЧОВІ СЛОВА: МЕТОД ОПТИМІЗАЦІЇ, МЕРЕЖА, ПРОТОКОЛ, ТРАФІК, ТЕХНОЛОГІЯ, КОМБІНАЦІЯ, ЗБІЖНІСТЬ, АНАЛІЗ.

Апробація результатів та публікації, які були опубліковані в ході проведення дослідження:

1. Токар М. В. Сучасні методи оптимізації протоколу OSPF. *Наука сьогодення: від досліджень до стратегічних рішень* : VI Міжнар. студент. наук. конф., м. Чернігів, 10 трав. 2024 р. С. 191–193.

2. Токар М. В. Вдосконалення протоколу OSPF для підтримки безпечної маршрутизації. *Застосування програмного забезпечення в інформаційно-комунікаційних технологіях* : Всеукр. науково-техн. конф., м. Київ, 24 квіт. 2024 р. С. 239–241.

ЗМІСТ

ВСТУП

У XXI столітті комп'ютерні мережі відіграють важливу роль у багатьох сферах життя, уявити сучасний світ без всесвітньої мережі Інтернет вже майже неможливо. Комп'ютерні мережі використовуються для забезпечення зв'язку між людьми та спілкування в цілому, передачі даних, зберігання й обробки інформації, розваг, навчання, медицини, банківських операцій, торгівлі, транспорту, виробництва та багатьох інших сферах. Вони є основою сучасного інформаційного суспільства і відіграють ключову роль у розвитку сучасних технологій та всесвітньої економіки. Зі стрімким розвитком мережевих технологій, підвищенням об’єму даних, що передаються мережами зростає важливість ефективного управління мережевим трафіком і оптимізації його маршрутизації.

Протокол OSPF (Open Shortest Path First) є одним протоколів маршрутизації в мережах TCP/IP. За цією назвою ховається протокол внутрішньої маршрутизації, який передає інформацію найкращим шляхом. Для знаходження найкращого шляху, протокол відстежує стан каналів, а динамічну маршрутизацію він забезпечує шляхом використання алгоритму Дейкстри (Dijkstra’s algorithm). Але його ефективність може страждати в умовах чимдалі більшого обсягу трафіку та складної топології мережі. Для ефективного функціонування комп'ютерних мереж необхідні протоколи маршрутизації, які дозволяють пакетам даних знаходити оптимальний шлях до пункту призначення. Аналізуючи дослідження та публікації, в яких описано розв’язання проблеми підвищення ефективності маршрутизації в мережах і використання для цього методів оптимізації протоколу OSPF [1, 2, 3, 4], було визначено конкретні найбільш ефективні методи оптимізації протоколу OSPF.

Актуальність даної роботи обумовлена необхідністю забезпечення стабільної та ефективної роботи мережі при мінливих умовах навантаження та конфігурації мережі. Оптимізація протоколу OSPF дозволить: підвищити продуктивність мережі, зменшити час на відновлення маршрутів у разі збоїв, забезпечити більшу гнучкість управління мережевим трафіком.

Метою для даної роботи ґрунтується на дослідженні та аналізу чинних методів оптимізації протоколу OSPF для підтримки динамічної маршрутизації, вивчення теоретичних основ протоколу OSPF, аналіз сучасних методів оптимізації OSPF, а також розробка рекомендацій щодо їх застосування на практиці. В ході дослідження вирішувалися наступні завдання: пошук та аналіз інформації з відкритих джерел, щодо методів оптимізації протоколу OSPF для підтримки динамічної маршрутизації; пошук та аналіз відкритих досліджень, публікацій та робіт на тему оптимізації протоколу OSPF для підвищення ефективності мережі; форматування та структуризація проаналізованої інформації; побудова висновків на основі отриманої інформації.

Об'єктом дослідження є протокол OSPF (Open Shortest Path First) та методи його оптимізації. Протокол OSPF – це динамічний протокол маршрутизації, який використовує алгоритм найкоротшого шляху SPF для просування пакетів мережею шляхом визначення оптимального маршруту до пункту призначення.

Предметом дослідження є конкретна частина оптимізації протоколу OSPF, а саме: методи оптимізації протоколу OSPF для підтримки динамічної маршрутизації.

Використаний метод дослідження – аналіз та виведення висновків на базі здобутої інформації. Таким методом дослідження було визначено не до кінця розкрите проблемне питання предмета дослідження. Під час проведення дослідження кваліфікаційної роботи, дістало подальший розвиток питання методів оптимізації протоколу OSPF для підтримки динамічної маршрутизації.

Апробація результатів та публікації, які були опубліковані в ході проведення дослідження:

РОЗДІЛ 1. ОГЛЯД ПРОТОКОЛУ OSPF. ТЕОРЕТИЧНІ ОСНОВИ ДИНАМІЧНОЇ МАРШРУТИЗАЦІЇ

1.1 Маршрутизація та її види

Для повного розуміння подальшого матеріалу спочатку треба розібратися з поняттям маршрутизації, маршрутизатора та інших видів мережевого обладнання. Маршрутизація – це процес знаходження маршруту пакету в будь-якій мережі [5]. Комп'ютерна мережа складається з безлічі комп'ютерів або вузлів, і маршрутів або зв'язків, що з'єднують ці вузли. Зв'язок між двома вузлами у взаємопов'язаній мережі може здійснюватися за різними маршрутами. Маршрутизація – це процес вибору найкращого маршруту з використанням деяких заздалегідь встановлених правил. Маршрутизація відбувається на мережевому рівні L3 – це третій рівень мережевої моделі OSI, який відповідає за маршрутизацію даних між різними мережами та підмережами. На цьому рівні використовуються протоколи, які визначають правила передавання та оброблення пакетів даних, а також адресну інформацію та маршрутні таблиці. На цьому рівні працюють такі пристрої, як маршрутизатори, шлюзи та комутатори [6].

Протоколи мережевого рівня можна розділити на два типи: протоколи зі встановленням з'єднання і протоколи без встановлення з'єднання. Протоколи зі встановленням з'єднання створюють тунель між двома точками обміну даних і гарантують надійну передачу інформації. Протоколи без встановлення з'єднання відправляють дані у відкритому вигляді мережею і не потребують підтвердження отримання або виправлення помилок. Маршрутизатори – це пристрої, що використовуються в комп'ютерних мережах для передачі даних між різними сегментами мережі. Вони приймають пакети даних, аналізують інформацію в заголовках цих пакетів і визначають оптимальний шлях для їхнього передавання до кінцевого пункту призначення.

Маршрутизація складається з двох основних компонентів: визначення оптимального шляху маршрутизації та передача груп інформації (пакетів) через об'єднану мережу. Визначення маршруту базується на низці показників (наприклад, довжині маршруту) або на їх комбінації. Програмна реалізація алгоритму маршрутизації розраховує індикатори маршруту для визначення найкращого маршруту до місця призначення. Алгоритми маршрутизації створюють і підтримують таблиці маршрутизації, щоб спростити процес визначення маршруту. Інформація про маршрутизацію залежить від залученого алгоритму маршрутизації.

Алгоритм маршрутизації заповнює таблицю маршрутизації деяким набором інформації. Асоціація призначення/наступного переходу повідомляє маршрутизатору, що найкраще досягти певного пункту призначення, надіславши пакет на певний маршрутизатор, який представляє наступний перехід до кінцевого пункту призначення. Коли надходить вхідний пакет, маршрутизатор перевіряє адресу призначення та намагається пов’язати цю адресу з наступним пересиланням.

Метрики надають інформацію про пріоритетність будь-якого каналу чи інтерфейсу. Маршрутизатори порівнюють показники, щоб визначити найкращий маршрут. Показники відрізнятимуться залежно від схеми алгоритму маршрутизації, що використовується.

Маршрутизатори спілкуються між собою, надсилаючи різні повідомлення та підтримуючи таблиці маршрутизації. Одним із типів повідомлень є повідомлення про оновлення маршруту. Оновлення маршрутизації зазвичай включають всю або частину таблиці маршрутизації. Аналізуючи інформацію про оновлення маршрутизації всіх маршрутизаторів, кожен маршрутизатор може створити повну схему топології мережі. Іншим прикладом повідомлень, якими обмінюються маршрутизатори, є повідомлення про статус з’єднання. Повідомлення про статус з’єднання інформують інші маршрутизатори про стан з’єднання відправника. Інформацію про підключення також можна використовувати для створення повної схеми топології мережі. Після визначення топології мережі маршрутизатор може визначити найкращий маршрут до пункту призначення пакетів.

Після вивчення адреси протоколу призначення пакета маршрутизатор визначає, чи знає він, як переслати пакет наступному маршрутизатору. У другому випадку (коли маршрутизатор не знає, як переслати пакет), пакет зазвичай ігнорується. У першому випадку маршрутизатор надсилає пакет наступному маршрутизатору, замінює фізичну адресу призначення фізичною адресою наступного маршрутизатора, а потім пересилає пакет. Наступне пересилання може бути або не бути для хосту кінцевого пункту призначення. Якщо ні, наступне пересилання, як правило, до іншого маршрутизатора, який виконує той самий процес прийняття рішення про передачу. Коли пакет проходить через конвергентну мережу, його фізична адреса змінюється, але адреса протоколу залишається незмінною.

Склад таблиці маршрутизації може відрізнятися, проте основні складові є такими (табл. 1.1):

– мережева адреса (network address) – адреса пункту призначення (адреса мережі або хосту, інформація про маршрут до яких записана в інших стовпцях);

– маска підмережі (network mask) – маска підмережі для адреси в першому стовпчику;

– адреса шлюзу (gateway address) – адреса маршрутизатора, якому необхідно посилати пакет, щоб доставити його хосту з адресою з першого стовпчика;

– інтерфейс (interface) – адреса мережевого адаптера, через який необхідно передавати пакети маршрутизатору, адреса якого зазначена в стовпці «адреса шлюзу»;

– метрика (metric) – число, що дає змогу порівняти відносну ефективність різних шляхів до однієї мети (фактично показує, скільки маршрутизаторів треба пройти, щоб дістатися до мети).

Таблиця 1.1 – Приклад таблиці маршрутизації

Адреса мережі	Маска підмережі	Адреса шлюзу	Інтерфейс	Метрика
127.0.0.0	255.0.0.0	127.0.0.1	127.0.0.1	1
0.0.0.0	0.0.0.0	198.21.17.7	198.21.17.5	1
56.0.0.0	255.0.0.0	213.34.12.4	213.34.12.3	15
116.0.0.0	255.0.0.0	213.34.12.4	213.34.12.3	13

Мережеві пристрої, які не мають можливості пересилати пакети між підмережами, називаються кінцевими системами (ES), а мережеві пристрої, які мають можливість пересилати пакети, називаються проміжними системами (IS). Проміжні системи далі поділяються на системи, які можуть обмінюватися даними в межах «домену маршрутизації» («внутрішньодоменна» IS), і системи, які можуть спілкуватися як у межах домену маршрутизації, так і з іншими доменами маршрутизації («міждоменна IS»). «Домен маршрутизації» зазвичай вважається частиною об'єднаної мережі під спільним адміністративним контролем і керується певним набором адміністративних інструкцій. Домени маршрутизації також називають «автономними системами» (AS). Для деяких протоколів домен маршрутизації можна додатково розділити на «області маршрутизації», але протоколи внутрішньодоменної маршрутизації також можна використовувати для обміну всередині та між областями.

Автономна система — це сукупність мереж під єдиним адміністративним контролем, яка забезпечує загальну політику маршрутизації для всіх маршрутизаторів, що містяться в автономній системі. Як правило, автономними системами керує один Інтернет-провайдер, який самостійно вибирає, які протоколи маршрутизації повинні використовуватися в конкретній автономній системі та як інформація маршрутизації повинна перерозподілятися між ними. Великі постачальники послуг і компанії можуть представити свої складові мережі як сукупність автономних систем. Реєстрація автономних систем, а також реєстрація IP-адрес і DNS-імен здійснюється централізовано. Номер автономної системи складається з 16 цифр і не має відношення до префікса IP-адреси мережі, що в ній міститься. Відповідно до цієї концепції, Інтернет має форму набору взаємопов'язаних автономних систем, кожна з яких складається з взаємопов'язаних

мереж. Це зроблено з метою забезпечення багаторівневого підходу до маршрутизації. До впровадження автономних систем передбачався дворівневий підхід, тобто маршрути спочатку визначалися як послідовність мереж, а потім вели безпосередньо до певного вузла в кінцевій мережі. З появою автономних систем з'явився третій вищий рівень маршрутизації – тепер маршрутизація виконується спочатку від серії автономних систем, потім до серії мереж перед тим, як вести до останнього вузла.

Автономні системи підключаються через зовнішні шлюзи та маршрутизатори (рис. 1.1). Між зовнішніми шлюзами дозволяється використовувати лише один протокол маршрутизації, і не будь-який, а той, який наразі визнається IETF як стандарт для зовнішніх шлюзів. Цей протокол маршрутизації називається Exterior Gateway Protocol (EGP) і наразі є BGP версії 4 (BGPv4). Усі інші протоколи (RIP, OSPF, IS-IS, EIGRP) є протоколами внутрішнього шлюзу (IGP, Interior Gateway Protocol). Протокол зовнішнього шлюзу відповідає за вибір маршрутів між автономними системами. Адреса точки входу сусідньої автономної системи вказується як адреса наступного роутера. Протокол внутрішнього шлюзу відповідає за маршрутизацію в автономній системі. Для транзитних автономних систем ці протоколи визначають точну послідовність маршрутизаторів від точки входу до точки виходу автономної системи.

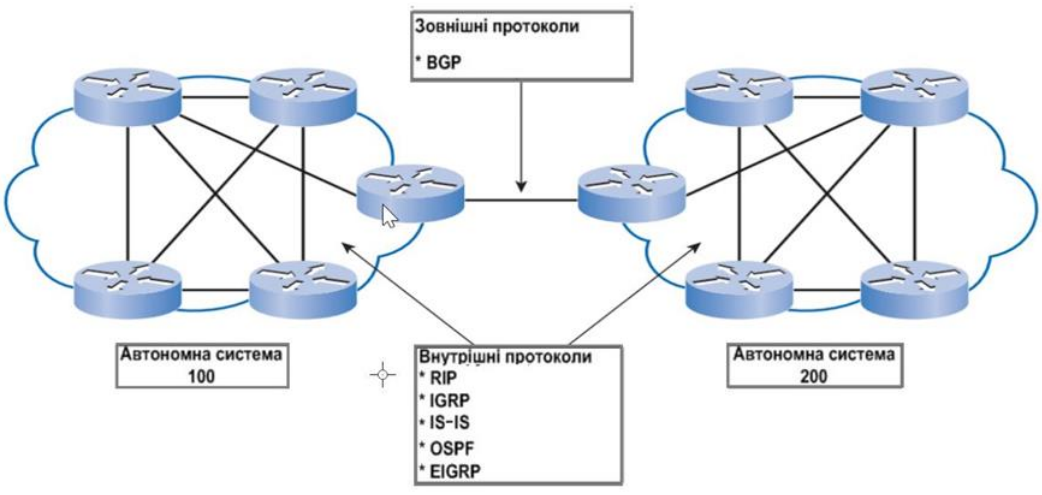


Рисунок 1.1 – Узагальнений вигляд автономних систем Інтернету

Автономні системи утворюють основу Інтернету. Концепція автономних систем приховує проблеми маршрутизації пакетів на нижчих рівнях (рівні мережі) від адміністраторів магістралі Інтернету. Для адміністратора не має значення, який протокол маршрутизації використовується всередині автономної системи, для нього існує тільки один протокол маршрутизації – BGPv4.

Класифікація способів маршрутизації пакетів у складових мережах [4]:

1. маршрутизація, що не вимагає наявності таблиць маршрутизації:

– широкомовна маршрутизація – кожен маршрутизатор передає пакет усім своїм безпосереднім сусідам, крім того, від якого отримав;

– маршрутизація за подіями (Event depended routing) – пакет до певної мережі призначення надсилається за маршрутом, що вже приводив раніше до успіху для цієї адреси призначення;

– маршрутизація за джерелом (source routing) – відправник додає до пакета інформацію про те, які проміжні маршрутизатори повинні бути задіяні в надсиланні пакета в мережу призначення;

2. маршрутизація на підставі таблиць маршрутизації:

– статична – таблиці маршрутизації складаються і вводяться в пам'ять кожного маршрутизатора вручну адміністратором мережі, мають нескінченний термін життя;

– динамічна (або адаптивна) – таблиці маршрутизації за допомогою спеціальних протоколів маршрутизації автоматично складаються на підставі інформації про конфігурацію мережі; будь-яка зміна конфігурації мережі автоматично відображається в таблицях маршрутизації; мають обмежений час життя.

Переваги статичної маршрутизації:

– відсутність значних навантажень на процесор та пам’ять маршрутизатора, що означає, що для маршрутизації можна використовувати дешевший маршрутизатор;

– вона має кращий параметр безпеки, оскільки лише адміністратор може дозволяти маршрутизацію до конкретних сегментів мережі;

– відсутність використання смуги пропускання між маршрутизаторами.

Недоліки статичної маршрутизації:

– у великій мережі адміністратору незручно вручну додавати кожен маршрут до таблиці маршрутизації на кожному маршрутизаторі;

– адміністратор повинен добре знати топологію мережі. Якщо приходить новий адміністратор, він повинен вручну додавати кожен маршрут.

Динамічні протоколи маршрутизації, що застосовуються в сучасних внутрішніх мережах, поділяють на дві групи (рис 1.2):

– дистанційно-векторний алгоритми (Distance Vector Algorithms, DVA);

– алгоритми за станом зв'язку (Link State Algorithms, LSA).

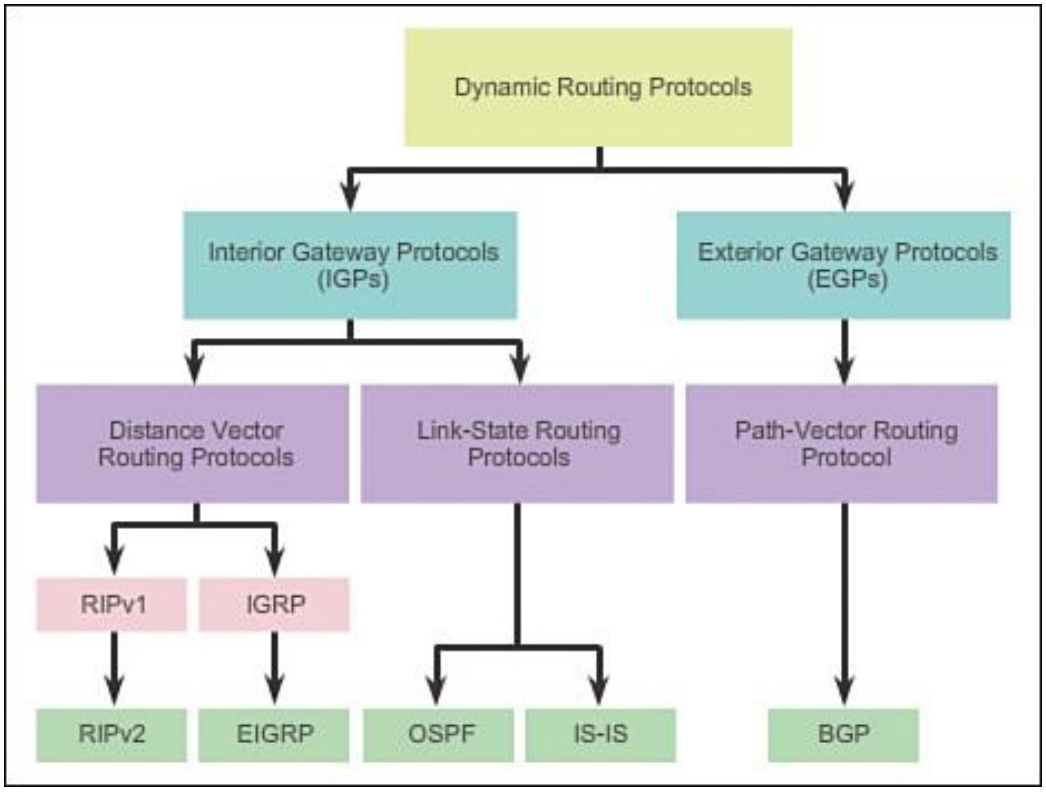


Рисунок 1.2 – Класифікація протоколів маршрутизації

В дистанційно-векторних алгоритмах (DVA) кожен маршрутизатор періодично транслює в мережу вектор, в якому зазначені відстані до всіх відомих мереж від цього маршрутизатора. Отримавши вектор відстані до відомої мережі від сусіда, маршрутизатор додає компоненти вектора до своєї власної відстані до цього

сусіда та доповнює свій вектор інформацією про інші мережі, які йому відомі. Потім він знову надсилає нове векторне значення через мережу. Згодом кожен маршрутизатор дізнається інформацію про всі інші маршрутизатори та відстань до них. Дистанційно-векторні алгоритми (DVA) добре працює лише в невеликих мережах. У великих мережах вони засмічують лінії зв'язку щільним періодичним трафіком, а крім того, не завжди правильно обробляють зміни конфігурації мережі. Найбільш поширеними протоколами на основі DVA є RIP (Routing Information Protocol), IGRP і EIGRP.

Алгоритми стану зв’язку (LSA) надають кожному маршрутизатору достатньо інформації для побудови точної діаграми мережі. Всі маршрутизатори працюють за одним графом, що робить процес маршрутизації стійкішим до змін конфігурації. Кожен маршрутизатор використовує граф мережі, щоб знайти найкращий маршрут для кожної мережі на основі певних критеріїв. Кожен маршрутизатор періодично обмінюється HELLO пакетами зі своїми найближчими сусідами, щоб дізнатися про стан ліній зв'язку та підключених нього портів. Якщо статус будь-якої лінії зв'язку змінюється, тільки в цьому випадку повідомлення про зміну зв'язку буде надіслано всім іншим маршрутизаторам. Алгоритм стану зв’язку (LSA) найчастіше використовується у великих мережах зі складною топологією, що включає мережеві петлі. Службовий трафік, створюваний алгоритмом LSA, більш інтенсивний, ніж алгоритмом DVA.

1.2 Поняття та принципи динамічної маршрутизації

При використанні статичної маршрутизації, потрібно прописувати руками таблицю маршрутизації на кожному роутері [7]. Використання протоколів маршрутизації дає нам змогу уникнути цього нудного одноманітного процесу і помилок, пов'язаних із людським фактором. Як зрозуміло з назви, протоколи динамічної маршрутизації покликані будувати таблиці маршрутизації самі,

автоматично, виходячи з поточної конфігурації мережі. Динамічна маршрутизація використовується для спілкування маршрутизаторів один з одним. Маршрутизатори передають один одному інформацію про те, які мережі наразі під'єднані до кожного з них. Маршрутизатори спілкуються, використовуючи протоколи маршрутизації. Користувацький процес, за допомогою якого маршрутизатори можуть спілкуватися із сусідніми маршрутизаторами, називається демоном маршрутизації (routing daemon). Демони маршрутизації обмінюються між собою інформацією, яка дає їм змогу заповнити таблицю маршрутизації найбільш оптимальними маршрутами.

Модель TCP/IP підтримує двох демонів для підтримки динамічної маршрутизації: routed і gated. Демон gated також підтримує протокол інформації про маршрутизацію (RIP), протокол зовнішнього шлюзу (EGP), протокол межового шлюзу (BGP), протокол найкоротшого шляху (OSPF), IS-IS, ICMP і ICMPv6. Крім того, демон gated підтримує Simple Network Management Protocol (SNMP). Демон routed підтримує лише протокол інформації про маршрутизацію.

Автоматично створювані таблиці маршрутизації повинні забезпечувати раціональність проходження пакетів через мережу. Критеріями вибору раціонального маршруту можуть бути:

– найкоротша відстань (під відстанню розуміється кількість проміжних маршрутизаторів – хопів, через які проходить пакет);

– найменший час проходження;

– комплексний критерій, що враховує відстань, пропускну здатність каналів між маршрутизаторами, надійність каналів, затримки тощо.

Відповідно до динамічних алгоритмів маршрутизації, які автоматично формують таблиці маршрутизації, висуваються такі вимоги:

– алгоритми динамічної маршрутизації повинні забезпечувати раціональність маршруту;

– алгоритми динамічної маршрутизації повинні бути достатньо простими, щоб не вимагати занадто великого обсягу обчислень або породжувати інтенсивний службовий трафік;

– алгоритми динамічної маршрутизації повинні мати властивість збіжності, що означає завжди підтримувати узгоджену структуру таблиці маршрутизації на всіх маршрутизаторах у мережі протягом прийнятного часу;

– до того ж протоколи динамічної маршрутизації передбачують наявність відмовостійкості. Вони повинні нормально функціонувати навіть у випадку непередбачуваних подій, таких як відмова обладнання, високе навантаження або помилки реалізації пакетів. Стабільність є критично важливою, оскільки маршрутизатори є вузловими точками мережевого з'єднання, і їхні помилки можуть призвести до проблем для всієї глобальної мережі. Маючи мережу зі статичною маршрутизацією, буде вкрай складно організувати резервні канали – нікому постійно відстежувати доступність того чи іншого сегмента мережі.

Наприклад, якщо в такій мережі (рис 1.3) розірвати зв’язок між R2 і R3, то пакети з R1, як і раніше, будуть іти на R2, де будуть знищені, бо їх нікуди відправити.

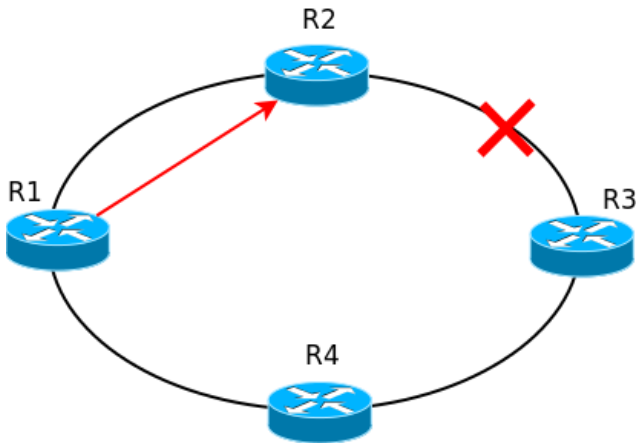


Рисунок 1.3 – Проста мережа з перерваним маршрутом

Протоколи динамічної маршрутизації протягом декількох секунд (а то й мілісекунд) дізнаються про проблеми в мережі та перебудовують свої таблиці маршрутизації, і у вищеописаному випадку пакети відправлятимуться вже за актуальним маршрутом (рис 1.4).

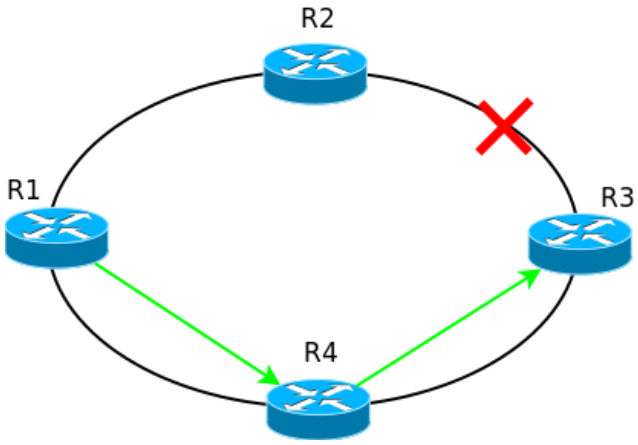


Рисунок 1.4 – Мережа з актуальним маршрутом

Не менш важливим є балансування трафіку. Протоколи динамічної маршрутизації в штатному режимі, майже без налаштувань, підтримують цю функцію і тому не потрібно додавати надлишкові маршрути вручну, вираховуючи їх самостійно.

Також впровадження динамічної маршрутизації сильно полегшує масштабування мережі. Коли додається новий елемент у мережу або підмережу на наявному маршрутизаторі потрібно виконати лише кілька дій, щоб усе запрацювало, і ймовірність помилки мінімальна, водночас інформація про зміни миттєво розходиться по всім пристроям. Абсолютно но те ж саме можна сказати й про глобальні зміни топології.

Переваги динамічної маршрутизації:

– проста в налаштуванні порівняно зі статичною маршрутизацією;

– ефективніший вибір найкращого маршруту до віддаленої мережі призначення.

Недоліки динамічної маршрутизації:

– споживає більше пропускної здатності каналів для спілкування з іншими сусідами;

– параметр безпеки нижчий, ніж при статичній маршрутизації.

1.3 Протоколи динамічної маршрутизації

Як вже було сказано раніше, протоколи динамічної маршрутизації поділяються на дистанційно-векторні алгоритми (Distance Vector Algorithms, DVA) та алгоритми стану зв'язків (Link State Algorithms, LSA) [8]. Перелік протоколів динамічної маршрутизації:

– RIPv1 (застарів) – класовий дистанційно-векторний протокол для внутрішньої маршрутизації;

– IGRP (застарів) – класовий протокол для внутрішньої маршрутизації, розроблений компанією Cisco (не використовується після виходу IOS 12. 2 і пізніших версій);

– RIPv2 – безкласовий дистанційно-векторний протокол для внутрішньої маршрутизації;

– EIGRP – безкласовий дистанційно-векторний протокол для внутрішньої маршрутизації зі встановленням з'єднання розроблений компанією Cisco, який використовується для розподіленої маршрутизації в локальних мережах;

– OSPF – безкласовий протокол внутрішньої маршрутизації за станом каналу зі встановленням з'єднання, який використовується для розподіленої маршрутизації в локальних мережах;

– IS-IS – безкласовий протокол внутрішньої маршрутизації, за станом каналу;

– BGP – безкласовий протокол за вектором маршруту для зовнішньої маршрутизації без встановлення з'єднання, який використовується для розподіленої маршрутизації між різними автономними системами AS.

Найбільша різниця між протоколами класової маршрутизації та безкласовими протоколами маршрутизації полягає в тому, що протоколи класової маршрутизації не передають інформацію про маску підмережі під час оновлення маршруту. Протоколи безкласової маршрутизації включають інформацію про маску підмережі в оновлення маршрутизації.

Спочатку було розроблено два протоколи внутрішньої маршрутизації IPv4 – RIPv1 і IGRP. Вони були створені коли мережева адреса призначалася на основі класу (наприклад, класу A, B чи C). У той час протоколи маршрутизації не вимагали включати маски підмережі в пакети оновлення маршрутизації, оскільки маску підмережі можна було визначити за першим октетом мережевої адреси. Лише протоколи RIPv1 і IGRP є протоколами класової маршрутизації. Усі інші протоколи маршрутизації IPv4 та IPv6 є безкласовими протоколами. В IPv6 ніколи не застосовувалась класова адресація. Оскільки RIPv1 та IGRP не включають інформацію про маску підмережі у свої оновлення, то ці протоколи не можуть надавати маски підмережі змінної довжини (VLSM) і тому не можуть використовуватися для безкласової міждоменної маршрутизації (CIDR). Також класові протоколи маршрутизації створюють деякі проблеми в «розірваних» мережах. «Розірвана» мережа є та, в якій підмережі в межах базової мережі одного класу розділені мережевими адресами іншого класу.

Більшість протоколів маршрутизації мають метричні структури та алгоритми, які не сумісні з іншими протоколами. У мережі з декількома протоколами маршрутизації обмін маршрутною інформацією та можливість вибору найкращого шляху серед декількох протоколів є критично важливими. Адміністративна відстань визначає надійність протоколу маршрутизації. Кожен протокол маршрутизації має пріоритет у порядку від найбільш до найменш надійного (правдоподібного) за допомогою значення адміністративної відстані. Адміністративна відстань – це перший критерій, який використовує маршрутизатор, щоб визначити, який протокол маршрутизації використовувати, якщо два протоколи надають інформацію про маршрут до одного і того ж пункту призначення. Адміністративна відстань є мірою надійності джерела маршрутної інформації. Адміністративна відстань має лише локальне значення і не вказується в оновленнях маршрутизації.

Чим менше значення адміністративної відстані, тим надійніший протокол (табл. 1.2). Наприклад, якщо маршрутизатор отримує маршрут до певної мережі як від Open Shortest Path First (OSPF) (адміністративна відстань за замовчуванням –

110), так і від Internal Gateway Routing Protocol (IGRP) (адміністративна відстань за замовчуванням – 100), маршрутизатор обирає IGRP, оскільки IGRP є надійнішим. Це означає, що маршрутизатор додає версію маршруту IGRP до таблиці маршрутизації. Якщо ви втрачаєте джерело інформації, отриманої з IGRP (наприклад, через відключення електроенергії), програмне забезпечення використовує інформацію, отриману з OSPF, до тих пір, поки інформація, отримана з IGRP, не з'явиться знову.

Таблиця 1.2 – Значення адміністративної відстані за замовчуванням у Cisco

Протокол	Адміністративна відстань
Connected interface	0
Static route	1
Enhanced Interior Gateway Routing Protocol (EIGRP) summary route	5
External Border Gateway Protocol (BGP)	20
Internal EIGRP	90
IGRP	100
OSPF	110
Intermediate System-to-Intermediate System (IS-IS)	115
Routing Information Protocol (RIP)	120
Exterior Gateway Protocol (EGP)	140
On Demand Routing (ODR)	160
External EIGRP	170
Internal BGP	200
Unknown	255

1.4 Основні поняття та принципи протоколу OSPF

OSPF – це динамічний протокол маршрутизації, який використовує алгоритм найкоротшого шляху для визначення оптимального маршруту до пункту призначення [9, 10, 11]. OSPF широко використовується як в IPv4, так і в IPv6 мережах. OSPF був розроблений в якості альтернативи протоколу маршрутизації на базі векторів відстані RIP. На початку розвитку мережевих технологій та Інтернету протокол RIP був основним. Однак, коли RIP використовував кількість переходів як єдиний показник метрики для визначення найкращого маршруту, швидко виникло багато труднощів. При використанні цього підходу масштабованість великих мереж, що містять кілька шляхів на різних швидкостях, сильно обмежена. OSPF має багато значних переваг перед RIP, забезпечуючи швидшу конвергенцію та масштабованість для великих мереж.

Коли використовується протокол стану каналу, маршрутизатор не обмінюється інформацією про відстані зі своїми сусідами. Натомість кожен маршрутизатор активно тестує статус своїх каналів до кожного сусіднього маршрутизатора та надсилає цю інформацію іншим своїм сусідам, які можуть спрямувати потік даних в автономну систему. Кожен маршрутизатор приймає інформацію про стан каналу і вже на її підставі будує повну таблицю маршрутизації. З практичної точки зору основна відмінність полягає в тому, що протокол стану каналу працює значно швидше, ніж протокол вектора відстаней. Потрібно зазначити, що у випадку протоколу стану каналу значно швидше здійснюється збіжність мережі. Під поняттям збіжності (converge) ми маємо на увазі стабілізацію мережі після будь-яких змін, як, наприклад, несправності маршрутизатора або виходу з ладу каналу.

Нижче перелічені основні переваги протоколу OSPF:

- необмежений розмір мережі;
- автономну систему можна розділити на зони маршрутизації;
- швидке встановлення маршрутів;

– під час маршрутизації враховується тип IP-сервісу (type-of-service – ToS), тобто різні сервіси можуть мати різні маршрути;

– кожен інтерфейс може мати метрику на основі: пропускної здатності, надійності, завантаженості черги пакетів, часу повернення, розміру максимального блоку даних, який можна передати через канал. Для кожного виду IP-сервісу може бути призначена окрема вартість;

– коли маршрути мають однакову вартість, OSPF рівномірно розподіляє трафік між цими маршрутами, що називається балансуванням навантаження (Load balancing);

– підтримка масок підмережі;

– підтримка ненумерованих каналів «точка-точка» між маршрутизаторами без IP-адрес. Цей спосіб дозволяє заощаджувати IP-адреси;

– використання автентифікації;

– використання групової (multicast) адресації замість широкомовної.

OSPF – це протокол внутрішнього шлюзу (IGP) для маршрутизації пакетів Інтернет-протоколу (IP) в межах одного домену маршрутизації, наприклад, автономної системи [12]. Він збирає інформацію про стан з'єднання з доступних маршрутизаторів і будує топологічну карту мережі. Топологія представляється у вигляді таблиці маршрутизації на інтернет-рівень, який маршрутизує пакети виключно на основі їхньої IP-адреси призначення.

OSPF виявляє зміни в топології, такі як обриви зв'язку, і за лічені секунди знаходить нову структуру маршрутизації без петель (loop-free). Він обчислює дерево найкоротших шляхів для кожного маршруту за допомогою методу, заснованого на алгоритмі Дейкстри. Політика маршрутизації OSPF для побудови таблиці маршрутів керується метриками зв'язку, пов'язаними з кожним інтерфейсом маршрутизації. Факторами витрат можуть бути відстань до маршрутизатора (round-trip time), пропускна здатність каналу або доступність і надійність каналу, виражені у вигляді простих безрозмірних чисел. Це забезпечує динамічний процес балансування навантаження трафіку між маршрутами з однаковою вартістю.

OSPF розділяє мережу на зони маршрутизації, щоб спростити керування та оптимізувати використання трафіку та ресурсів. Зони ідентифікуються 32-розрядними числами, які або просто виражені в десятковому вигляді, або зазвичай у тій же десятковій системі нумерації на основі октетів, що й адреси IPv4. За домовленістю область 0 (нуль) або 0.0.0.0 представляє основну або магістральну область мережі OSPF. Адміністратори зазвичай вибирають IP-адресу основного маршрутизатора в області як ідентифікатор області, хоча за бажанням можна вибрати інші ідентифікатори області. Кожна додаткова область повинна мати з'єднання з магістральною областю OSPF. Такі з'єднання підтримуються з'єднувальними маршрутизаторами, які називаються прикордонними маршрутизаторами області (ABR). ABR підтримує окрему базу даних про стан зв'язку для кожної області, яку він обслуговує, і підтримує підсумовані маршрути для всіх областей мережі.

OSPF працює з IPv4 та IPv6, але не використовує транспортні протоколи, такі як UDP або TCP. Він інкапсулює свої дані безпосередньо в IP-пакети з протоколом 89. Це відрізняється від інших протоколів маршрутизації, таких як Routing Information Protocol (RIP) і Border Gateway Protocol (BGP). OSPF реалізує власні функції виявлення та виправлення транспортних помилок. OSPF також використовує групову адресацію для розповсюдження маршрутної інформації в широкомовному домені. Він резервує групові адреси 224.0.0.5 (IPv4) і ff02::5 (IPv6) для всіх SPF-маршрутизаторів (AllSPFRouters) і 224.0.0.6 (IPv4) і ff02::6 (IPv6) для всіх призначених маршрутизаторів (AllDRouters). Для не широкомовних мереж спеціальні положення конфігурації полегшують виявлення сусідів. Групові IP-пакети OSPF ніколи не проходять через IP-маршрутизатори, вони ніколи не проходять більше одного переходу. Тому протокол можна вважати протоколом канального рівня, але часто його також відносять до прикладного рівня в моделі TCP/IP. Він має функцію віртуального з'єднання, яка може бути використана для створення тунелю суміжності через декілька хопів. OSPF поверх IPv4 може безпечно працювати між маршрутизаторами, опціонально використовуючи різні методи автентифікації, щоб дозволити тільки довіреним маршрутизаторам брати

участь у маршрутизації. OSPFv2 підтримує автентифікацію на основі MD5 (Message Digest 5) і SHA (Secure Hash Algorithm). OSPFv3 (IPv6) покладається на стандартну безпеку протоколу IPv6 (IPsec) і не має внутрішніх методів автентифікації. Якщо автентифікацію дозволено, маршрутизатори OSPF можуть приймати від однорангових вузлів тільки такі повідомлення про оновлення маршрутизації, які зашифровані за попередньо узгодженим паролем.

OSPFv3 вносить зміни в реалізацію протоколу IPv4. За винятком віртуальних каналів, всі сусідні АТС використовують виключно локальну адресацію каналів IPv6. Протокол IPv6 працює для кожного каналу, а не на основі підмережі. Вся інформація про IP-префікси була вилучена з оголошень про стан каналу і з Hello пакетів, що робить OSPFv3 по суті незалежним від протоколу. Попри розширення IP-адресації до 128 біт в IPv6, ідентифікація зон і маршрутизаторів все ще базується на 32-бітних числах.

Терміни, необхідні для розуміння протоколу OSPF:

– інтерфейс (interface/link) – з’єднання між маршрутизатором та однією з мереж, до якої він підключений;

– оголошення про стан каналу (link-state advertisement, LSA) – оголошення описує всі канали маршрутизатора, усі інтерфейси та статус каналів;

– стан каналу (link state) – статус зв’язку між двома маршрутизаторами, що здійснюється за допомогою пакетів LSA;

– метрика (metric) – умовний показник «вартості» передачі даних по каналу;

– автономна система (autonomous system) складається з групи маршрутизаторів, які обмінюються інформацією про маршрутизацію через загальний протокол маршрутизації;

– зона (area) – сукупність мереж і маршрутизаторів з однаковим ідентифікатором зони;

– суміжність (adjacency) – зв'язок, встановлений між певними суміжними маршрутизаторами з метою обміну інформацією про маршрутизацію;

– hello protocol – використовується для забезпечення сусідських відносин;

– сусіди (neighbours) – це маршрутизатори з інтерфейсами в загальній мережі;

– база даних сусідів (neighbours database) – перелік всіх сусідів;

– база даних стану каналів (link state database, LSDB) – перелік усіх записів про стан каналів;

– ідентифікатор маршрутизатора (router ID) – унікальне 32-бітове число, яке гарантує унікальність маршрутизатора в межах однієї автономної системи.

Протокол OSPF визначає наступні категорії маршрутизаторів:

– внутрішній маршрутизатор (IR) має всі свої інтерфейси, що належать до однієї області. У таких маршрутизаторів тільки одна база даних стану каналів;

– прикордонний маршрутизатор (Area border router, ABR) – це маршрутизатор, який з'єднує одну або кілька зон з основною магістральною мережею. Він вважається членом усіх зон, до яких він підключений. ABR зберігає в пам'яті кілька екземплярів бази даних стану з'єднання, по одному для кожної області, до якої підключений цей маршрутизатор;

– магістральний маршрутизатор (BR, BBR) має інтерфейс до магістральної мережі. Магістральні маршрутизатори також можуть бути прикордонними маршрутизаторами;

– прикордонний маршрутизатор автономної системи (ASBR) – це маршрутизатор, який підключений за допомогою більш ніж одного протоколу маршрутизації і який обмінюється інформацією про маршрутизацію з маршрутизаторами автономних систем. Прикордонний маршрутизатор автономної системи може знаходитися в будь-якому місці автономної системи і бути внутрішнім, прикордонним чи магістральним маршрутизатором. ASBR використовується для розподілу маршрутів, отриманих від інших, зовнішніх AS, у своїй власній автономній системі. ASBR створює зовнішні LSA для зовнішніх адрес і розсилає їх у всі області через ABR. Маршрутизатори в інших зонах використовують ABR як наступні переходи для доступу до зовнішніх адрес. Потім ABR пересилають пакети до ASBR, який оголошує зовнішні адреси.

Мережа поділяється на області OSPF, які є логічними групами хостів і мереж (рис. 1.5). Область включає з'єднувальний маршрутизатор, що має інтерфейс для кожної підключеної лінії мережі. Кожен маршрутизатор підтримує окрему базу даних про стан з'єднання для своєї області, інформація з якої може бути узагальнена для решти мережі з'єднувальним маршрутизатором. Таким чином, топологія області невідома за її межами. Це зменшує трафік маршрутизації між частинами автономної системи. OSPF може обробляти тисячі маршрутизаторів, більше турбуючись про досягнення ємності таблиці інформаційної бази переадресації (FIB), коли мережа містить багато маршрутів і пристроїв нижчого класу. Сучасні маршрутизатори нижчого класу мають гігабайт або більше оперативної пам'яті, що дозволяє їм обробляти багато маршрутизаторів в області 0.

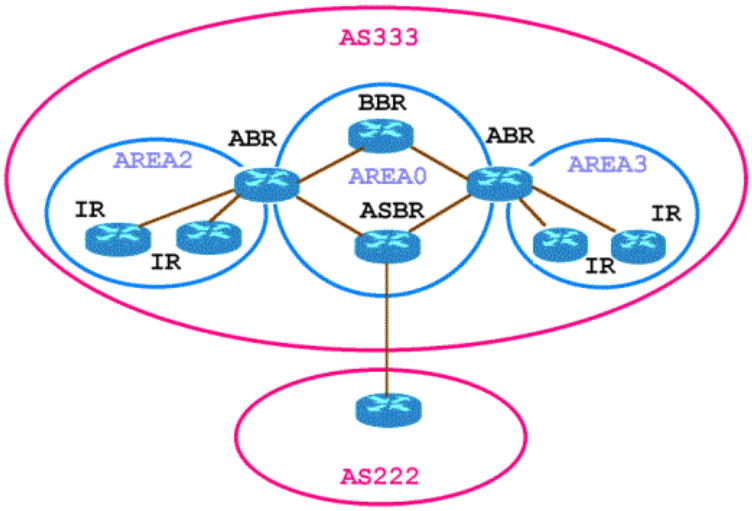


Рисунок 1.5 – Побудова мережі за допомогою OSPF

Унікальність зони ідентифікуються за допомогою 32-бітових чисел. Ідентифікатори зон зазвичай записуються в крапково-десятковій системі числення, знайомій з адресації IPv4. Однак вони не є IP-адресами і можуть без конфлікту дублювати будь-яку IPv4 адресу. Ідентифікатори областей для реалізацій IPv6 (OSPFv3) також використовують 32-розрядні ідентифікатори, записані в тій же нотації. Коли крапкове форматування опущено, більшість реалізацій розширюють область 1 до ідентифікатора області 0.0.0.1, але деякі з них, як відомо, розширюють її до 1.0.0.0.

У протоколі OSPF визначені кілька типів зон: магістральна (Backbone), стандартна за замовчуванням (Non-backbone/regular), тупикова (Stub), повністю тупикова (Totally stubby), не повністю тупикова (Not-so-stubby), не зовсім повністю тупикова (Not-So-Totally Stubby).

Магістральна зона (також відома як область 0 або область 0.0.0.0) утворює ядро мережі OSPF. Всі інші області підключаються до неї безпосередньо або через інші маршрутизатори. OSPF вимагає цього, щоб запобігти виникненню петель маршрутизації. Маршрутизація між областями відбувається через маршрутизатори, підключені до магістральної області та до їхніх власних асоційованих областей. Це логічна і фізична структура для «домену OSPF» і приєднана до всіх ненульових областей в домені OSPF.

Стандартна зона – це просто не магістральна (ненульова) зона без певних особливостей, яка генерує та отримує зведені та зовнішні LSA. Магістральна зона – це окремий тип такої зони.

Тупикова зона – це область, яка не отримує маршрутні оголошення ззовні від AS, а маршрутизація з цієї області повністю базується на маршруті за замовчуванням. ABR видаляє LSA типів 4 і 5 з внутрішніх маршрутизаторів, надсилає їм маршрут за замовчуванням 0.0.0.0 і перетворюється на шлюз за замовчуванням. Це зменшує розмір таблиці маршрутизації LSDB для внутрішніх маршрутизаторів. Виробники мережевих систем впровадили модифікації базової концепції тупикової зони, такі як Totally stubby area (TSA) і Not-so-stubby area (NSSA), які є розширенням в маршрутизаторах Cisco Systems.

Повністю тупикова зона не приймає інформацію про зовнішні маршрути для автономної системи і маршрути з інших зон. Єдиний спосіб для руху за межами зони – це маршрут за замовчуванням, який є єдиним LSA типу 3, оголошеним у цій зоні. Коли є лише один маршрут, що виходить за межі області, процесору маршрутизації потрібно приймати менше рішень про маршрутизацію, що знижує використання системних ресурсів. Інколи кажуть, що TSA може мати лише один ABR. Повністю тупикові області зазвичай використовуються у філіях або віддалених офісах для оптимізації маршрутизації та економії пропускної здатності.

Не повністю тупикова зона (NSSA) – це тип тупикової зони, яка може імпортувати зовнішні маршрути автономної системи та надсилати їх до інших зон, але не може отримувати зовнішні маршрути AS зовнішніх зон з інших зон. Вона дозволяє вводити зовнішні маршрути як LSA типу 7, які перетворюються в LSA типу 5 на ABR (Area Border Router) перед тим, як бути оголошеними в інших зонах. NSSA корисні при інтеграції OSPF із зовнішнім доменом маршрутизації, наприклад, при підключенні до Інтернету або партнерської мережі.

Не зовсім повністю тупикова зона блокує більшість підсумованих маршрутів і дозволяє маршрут за замовчуванням. Проте вона дозволяє обмежену кількість зовнішніх маршрутів у вигляді LSA типу 7, пропонуючи компроміс між повністю ізольованою мережею і мережею, яка потребує деяких зовнішніх лінків. Цей тип області корисний, коли мережа потребує зовнішніх маршрутів, але при цьому прагне підтримувати оптимальну ефективність маршрутизації.

Протоколом OSPF передаються такі типи повідомлень: Hello пакет, опис бази даних (DBD), оголошення про стан з'єднання (LSA), що в сою чергу включають запит стану каналу (LSR), оновлення стану каналу (LSU), підтвердження стану каналу (LSAck).

Повідомлення Hello в OSPF використовуються як форма привітання, щоб дозволити маршрутизатору виявити інші сусідні маршрутизатори в своїх локальних каналах і мережах. Повідомлення встановлюють зв'язки між сусідніми пристроями (так звані суміжності) і передають ключові параметри про те, як OSPF буде використовуватися в автономній системі або області. Під час нормальної роботи маршрутизатори надсилають привітальні повідомлення своїм сусідам через певні проміжки часу (інтервал привітання); якщо маршрутизатор перестає отримувати привітальні повідомлення від сусіда, то через певний проміжок часу (інтервал неактивності) маршрутизатор вважає, що сусід вийшов з ладу. Вони передають вміст бази даних стану з'єднань (LSDB) для даної області від одного маршрутизатора до іншого. Для передачі великої бази даних LSDB може знадобитися кілька повідомлень, для чого пристрій-відправник призначається

пристроєм-лідером і надсилає повідомлення послідовно, а послідовник (одержувач інформації з LSDB) відповідає на них підтвердженням.

Оголошення про стан з'єднання (LSA) є основним засобом зв'язку протоколу маршрутизації OSPF для Інтернет-протоколу (IP). Воно повідомляє локальну топологію маршрутизації маршрутизатора всім іншим локальним маршрутизаторам в тій же області OSPF. OSPF розроблений для масштабування, тому деякі LSA розсилаються не на всі інтерфейси, а лише на ті, що належать до відповідної області. Таким чином, детальна інформація може зберігатися локально, в той час, як підсумкова інформація розсилається на решту мережі. Оригінальний OSPFv2, що підтримує тільки IPv4, і новіший OSPFv3, сумісний з IPv6, мають в цілому схожі типи LSA.

Повідомлення запиту стану каналу (LSR) використовуються одним маршрутизатором для запиту оновленої інформації про частину LSDB від іншого маршрутизатора. У повідомленні вказується канал (и), для якого (их) пристрій, що запитує, хоче отримати актуальну інформацію.

Повідомлення про оновлення стану з'єднання (LSU) містять оновлену інформацію про стан певних з'єднань у LSDB. Вони надсилаються у відповідь на повідомлення запиту стану каналу, а також регулярно розсилаються маршрутизаторами в широкомовному або груповому режимі. Їхній вміст використовується для оновлення інформації в LSDB маршрутизаторів, які їх отримують.

Повідомлення про підтвердження стану зв'язку (LSAck) забезпечують надійність процесу обміну даними про стан зв'язку, явно підтверджуючи отримання повідомлення про оновлення стану зв'язку [13].

Перелік типів повідомлень про стан каналу (LSA):

– type 1 LSA – Router LSA – оголошення про стан каналу маршрутизатора. Ці LSA розповсюджуються всіма маршрутизаторами. LSA містить опис усіх каналів маршрутизатора та вартість (cost) кожного каналу. Поширюється тільки в межах однієї області;

– type 2 LSA – Network LSA – оголошення про стан каналів мережі. Поширюється DR у мережах із множинним доступом. LSA містить опис усіх маршрутизаторів (включаючи DR), підключених до мережі. Поширюється тільки в межах однієї області;

– type 3 LSA – Network Summary LSA – підсумкове оголошення про стан мережевого каналу, що поширюється прикордонними маршрутизаторами. Це оголошення описує лише маршрути до мереж за межами області, а не маршрути в межах автономної системи. Прикордонні маршрутизатори надсилають окремі оголошення для кожної відомої їм мережі. Коли маршрутизатор отримує Network Summary LSA від прикордонного маршрутизатора, він не виконує алгоритм обчислення найкоротшого шляху, а тільки додає до вартості маршруту зазначеного в LSA вартість маршруту до прикордонного маршрутизатора. Слідом маршрут до мережі через прикордонний маршрутизатор вставляється в таблицю маршрутизації;

– type 4 LSA – ASBR Summary LSA – підсумкове оголошення про статус каналу прикордонного маршрутизатора автономної системи. Оголошення розповсюджується прикордонними маршрутизаторами. Summary LSA ASBR відрізняються від Summary LSA тим, що замість розповсюдження інформації про мережу, поширюють інформацію про прикордонні маршрутизатори автономної системи;

– type 5 LSA – AS External LSA – оголошення про стан зовнішнього каналу автономної системи. Оголошення розповсюджується по всій автономній системі прикордонними маршрутизаторами. У цьому оголошенні описується зовнішній маршрут щодо автономної системи OSPF або маршрут за замовчуванням, зовнішній щодо автономної системи OSPF;

– type 6 LSA – Multicast OSPF LSA – спеціалізовані LSA, які використовують групові OSPF додатки (не задіяно в Cisco);

– type 7 LSA – AS External LSA for NSSA – оголошення щодо стану зовнішніх каналів автономної системи в зоні NSSA, що може передаватися тільки в зоні NSSA. На кордоні зони прикордонний маршрутизатор перетворює type 7 LSA на type 5 LSA;

– type 8 LSA – Link LSA – анонсує link-local адресу і префікси маршрутизатора всім іншим маршрутизаторам, що розділяють канал (link). Відправляється тільки якщо на каналі присутній більш ніж один маршрутизатор. Поширюються тільки в межах каналу (link);

– type 9 LSA – Intra-Area-Prefix LSA ставить у відповідність список префіксів IPv6 і маршрутизатор, вказуючи на Router LSA, список префіксів IPv6 і транзитну мережу, вказуючи на Network LSA. Розповсюджується тільки в межах однієї зони.

РОЗДІЛ 2. АНАЛІЗ НЕДОЛІКІВ ПРОТОКОЛУ OSPF

2.1 Принцип роботи протоколу OSPF

Для повного розуміння потреб оптимізації протоколу OSPF потрібно мати знання про принцип роботи даного протоколу [14], потрібно знати як встановлюються сусідські зв’язки між маршрутизаторами та знати про сам процес обміну повідомленнями між маршрутизаторами. Через те, що OSPFv2 працює поверх IP, а конкретно, він заточений тільки під IPv4 (OSPFv3 не залежить від протоколів 3-го рівня і тому може працювати з IPv6), то пояснення його роботи буде приведено на прикладі найпростішої мережі з двома маршрутизаторами (рис. 2.1).

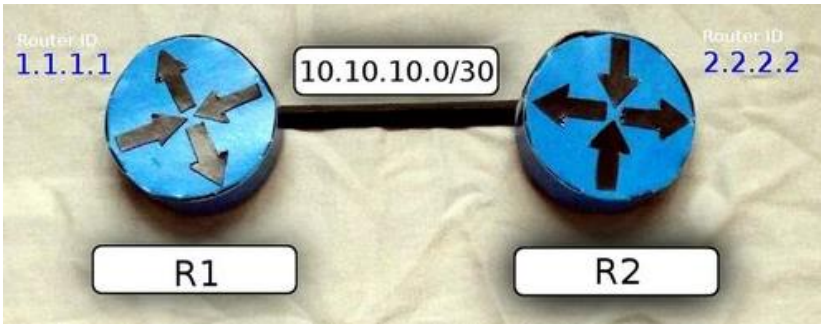


Рисунок 2.1 – Схема спрощеної мережі

Треба зазначити, що для того, щоб між маршрутизаторами зав'язалося сусідство (стосунки суміжності), мають виконуватися такі умови:

– в OSPF на підключених один до одного маршрутизаторах має бути налаштований однаковий інтервал Hello. За замовчуванням 10 секунд у широкомовних мережах, таких як Ethernet. По суті, це повідомлення про кореляцію конкретного з'єднання між двома маршрутизаторами. Тобто кожні 10 секунд кожен маршрутизатор надсилає своєму сусідові пакет Hello, кажучи «Привіт, я існую»;

– Dead Interval також повинні бути однаковими. Зазвичай це чотири інтервали Hello по 40 секунд. Якщо протягом цього періоду від сусіда не отримано

Hello, він вважається недоступним, і починається процес перебудови локальної бази даних LSDB і надсилання оновлень усім сусідам;

– інтерфейси, підключені один до одного, повинні бути в одній підмережі;

– OSPF може зменшити навантаження на ЦП маршрутизаторів шляхом поділу автономних систем AS на зони. Номери зон також повинні збігатися;

– кожен маршрутизатор, який бере участь у процесі OSPF, має свій унікальний ідентифікатор – Router ID. Якщо не зазначити його вручну, маршрутизатор автоматично вибере цю адресу на основі інформації про підключений інтерфейс (вибираючи найвищу адресу з інтерфейсу, який був активним під час запуску процесу OSPF). Ось чому зазвичай створюється інтерфейс Loopback, якому призначається адреса з маскою /32 і призначається ідентифікатор маршрутизатора. Це дуже зручний інструмент для обслуговування та усунення несправностей;

– повинен збігатися розмір повідомлення MTU.

Процес налаштування зв’язку для обміну повідомленнями між маршрутизаторами складається з таких пунктів:

1. стан OSPF – DOWN (рис. 2.2). В цей момент у мережі нічого не відбувається – усі маршрутизатори мовчать;

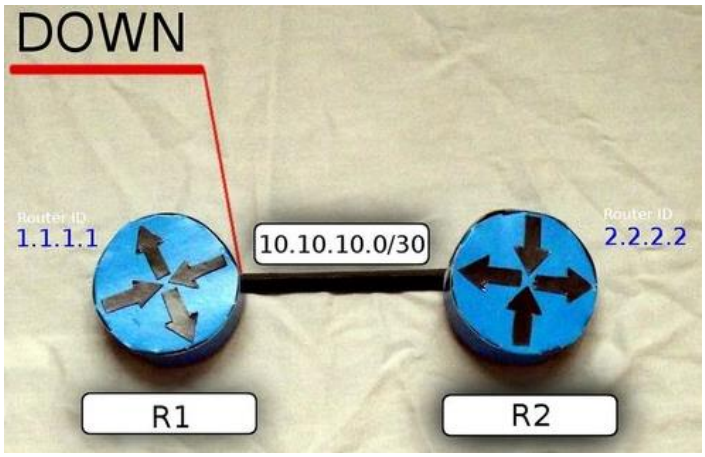


Рисунок 2.2 – Стан DOWN

2. маршрутизатор надсилає Hello повідомлення (рис. 2.3) на групову адресу 224.0.0.5 з усіх інтерфейсів, на яких працює OSPF. TTL (час життя) таких

повідомлень дорівнює одному стрибку, тому їх отримають лише маршрутизатори в одному сегменті мережі. R1 переходить у стан INIT. У пакети вкладається така інформація: Router ID; Hello Interval; Dead Interval; Сусіди; Маска підмережі; Ідентифікатор області; Router Priority; Адреси DR і BDR маршрутизаторів; Пароль автентифікації.

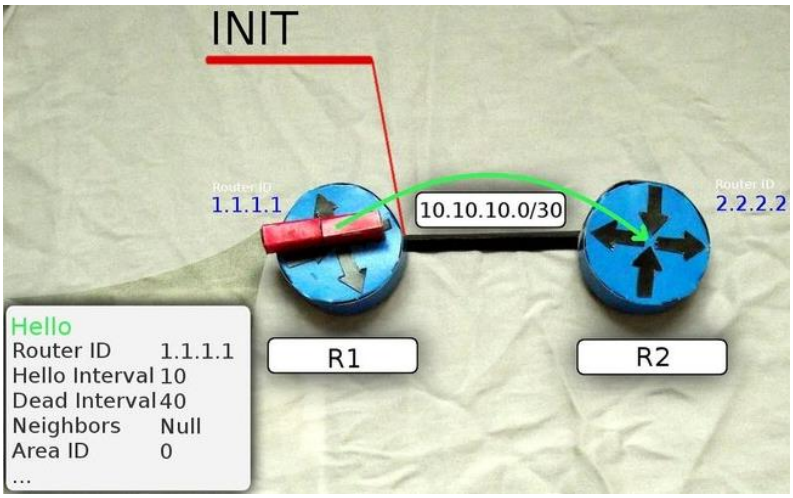


Рисунок 2.3 – Стан INIT на R1

Повідомлення Hello від маршрутизатора R1 несе в собі його Router ID і не містить Neighbors, тому що у нього їх поки що немає.

Після отримання групового повідомлення маршрутизатор R2 додає R1 у свою таблицю сусідів, якщо збіглися всі необхідні параметри (рис. 2.4).

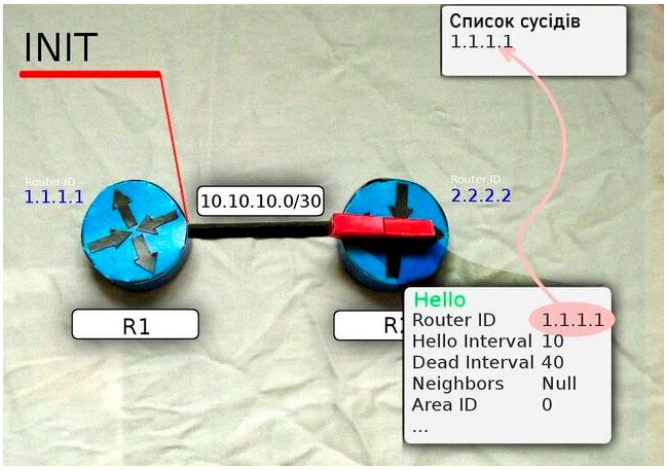


Рисунок 2.4 – Оновлення сусідів на R2

І тоді відправляє на R1 вже одноадресне нове повідомлення Hello, де міститься Router ID цього маршрутизатора, а в списку Neigbors перераховані всі його сусіди. Серед інших сусідів у цьому списку є Router ID R1, тобто R2 вже вважає його сусідом (рис. 2.5);

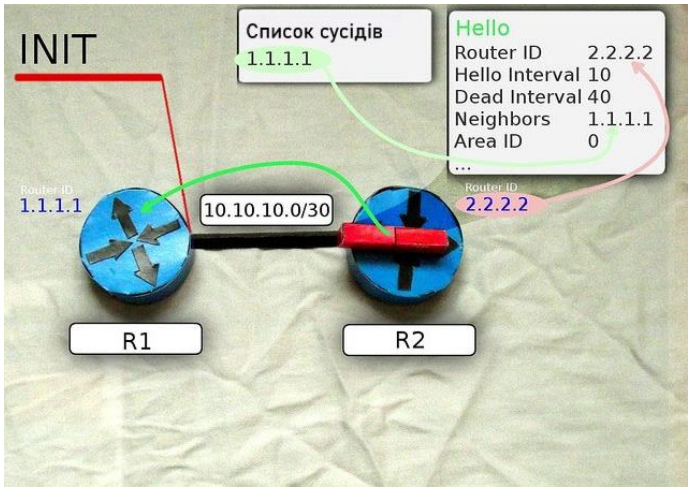


Рисунок 2.5 – Список сусідів на R2

3. коли R1 отримує це повідомлення Hello від R2, він гортає список сусідів і знаходить у ньому свій власний Router ID, він додає R2 до свого списку сусідів (рис. 2.6). Тепер R1 і R2 є сусідами один до одного – це означає, що між ними встановлюється зв'язок суміжності, а маршрутизаторі R1 встановлюється стан TWO WAY. Далі відбувається вибір DR і BDR;

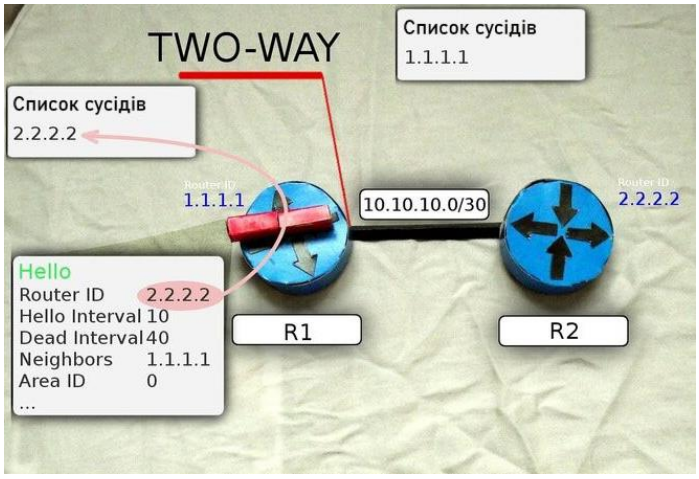


Рисунок 2.6 – Список сусідів на R1

4. потім усі сусіди переходять до стану EXSTART. Тоді всі сусіди між собою вирішують, хто головний. Ним стає маршрутизатор із найбільшим Router ID – R2;

5. коли вибрано головний вузол, сусіди переходять у стан EXCHANGE (рис. 2.7) і обмінюються DBD-повідомленнями (або DD) – Data Base Description, які містять в собі опис LSDB (Link State Data Base). Так вони повідомляють, що знають про такі підмережі.

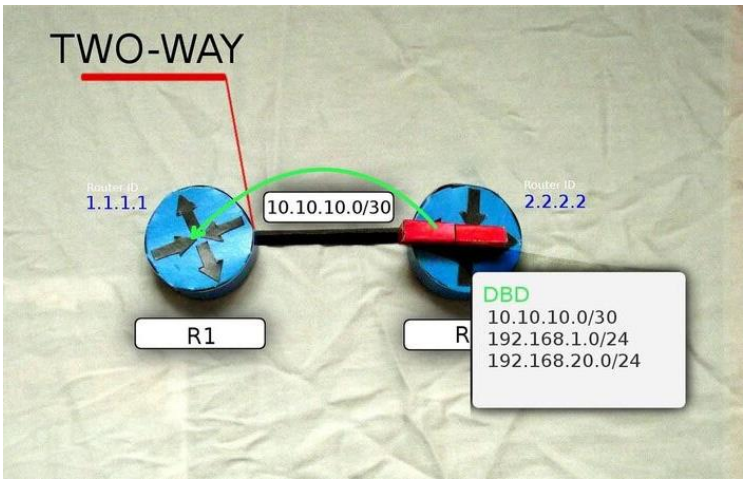


Рисунок 2.7 – Стан EXCHANGE

LSDB – це база даних про стан лінків. У початковому стані маршрутизатор знає лише ті інтерфейси, на яких запущено процес OSPF. Під час роботи кожен маршрутизатор збирає всю інформацію про мережу і складає топологію. Вона і стає LSDB, яка повинна бути однаковою для всіх членів зони. Першим, хто надсилає свій DBD, є маршрутизатор, обраний як основний маршрутизатор на цьому інтерфейсі – 2.2.2.2. Слідом за ним 1.1.1.1 робить те саме;

6. після отримання повідомлення маршрутизатори R1 і R2 відправляють один одному підтвердження DBD (LSAck), порівнюють нову інформацію з інформацією в LSDB і, якщо є різниця, відправляють один одному Link State Request (LSR) і переходять в новий стан LOADING (рис. 2.8). У LSR вони повідомляють: «Я нічого не знаю про цю мережу. Розкажіть мені більше»;

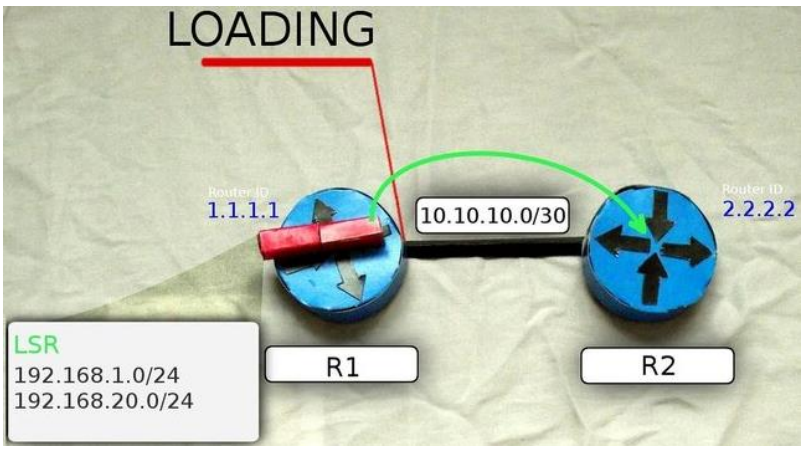


Рисунок 2.8 – Стан LOADING

7. R2 отримує LSR від R1 і надсилає Link State Update (LSU), що містить Link State Advertisement (LSA) з детальною інформацією про необхідні підмережі (рис. 2.9).

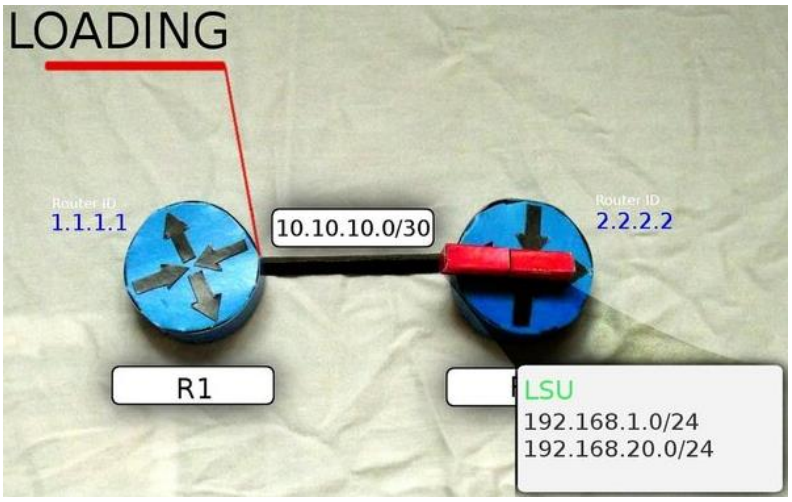


Рисунок 2.9 – R2 посилає LSU на R1

Отримавши останню порцію даних про всі підмережі, R1 будує свою LSDB, він переходить у кінцевий стан FULL STATE (рис. 2.10).

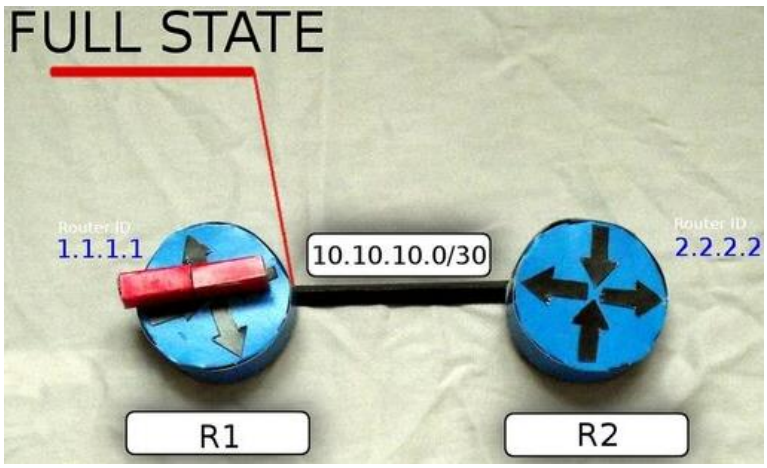


Рисунок 2.10 – Стан Full State

Коли всі маршрутизатори в зоні досягають стану FULL STATE, всі маршрутизатори повинні мати абсолютно однакову LSDB, тому що вони вивчають одну і ту ж мережу. На практиці це означає, що маршрутизатори знають всю мережу, що, як і де з'єднано;

8. на цьому етапі кожен маршрутизатор знає все про мережу, проте ці знання ще не застосовані для маршрутизації. Потім OSPF використовує алгоритм Дейкстри (також відомий як SPF – Shortest Path First) для обчислення найкоротшого маршруту до кожного маршрутизатора в регіоні. Це відбувається тому, що він знає всю топологію. Чим нижча метрика, тим кращий маршрут.

Наприклад, у такій мережі (рис. 2.11) з R1 в R3 можна дістатися безпосередньо або через R2. Очевидно, перший варіант коштуватиме менше. Але це за умови, що скрізь однаковий тип інтерфейсів. А якщо, між R1 і R3 наявне модемне з'єднання в 56 кбіт/с або вкрай нестабільний GPRS лінк, тоді у них буде дуже висока вартість і OSPF віддасть перевагу довшому, але швидшому шляху. Знайдений шлях потім додається в таблицю маршрутизації. Тепер кожні 10 секунд кожен маршрутизатор буде відправляти Hello-пакети, а кожні 30 хвилин розсилаються LSA – тоді дані вже вважаються застарілими, їх треба оновити, навіть якщо змін не було.

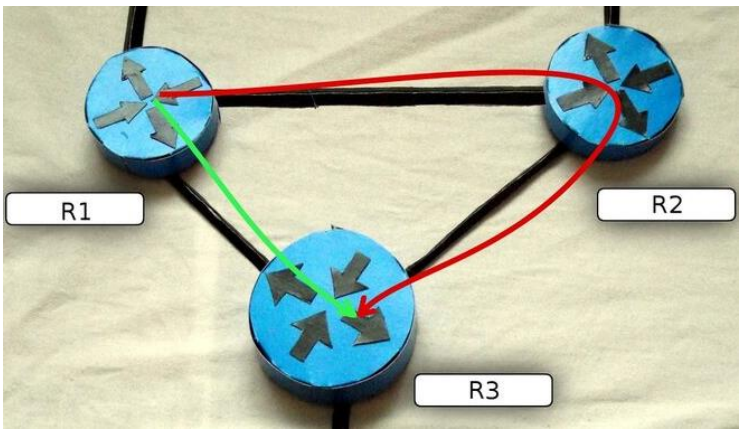


Рисунок 2.11 – Мережа

Тут виникає один нюанс: чекати 40 секунд (Dead Interval) і тільки потім починати перебудовувати таблицю – це занадто довго й може значно вплинути на передачу трафіку в мережах, залежних від швидкості та завадостійкості. Отже, щойно падає будь-який з лінків (або декілька), маршрутизатор змінює свою LSDB і генерує LSU, привласнюючи їй номер більший, ніж він був раніше (у кожної LSDB є номер, який береться з останнього отриманого LSA).

Це LSU повідомлення розсилається на групову адресу 224.0.0.5. Маршрутизатори, які отримали його, перевіряють номер LSA, що містяться в LSU.

Якщо номер більший, ніж номер поточної LSA маршрутизатора – LSDB змінюється. (Версія LSDB стара, інформація нова).

Якщо номер такий самий, нічого не відбувається. Цей маршрутизатор уже отримав цей LSA якимось іншим шляхом.

Якщо номер отриманого LSA менший за локальну LSDB, це означає, що в маршрутизатора вже актуальніша інформація, і він надсилає новий LSA (на основі своєї LSDB) відправнику колишнього.

Після виконаних (або невиконаних) дій сусідові, від якого прийшов LSU, пересилаються LSAck ніби, «інформація отримана – все гаразд», а іншим сусідам надсилається початковий LSU без змін. На цьому маршрутизаторі знову запускається алгоритм SPF і, за необхідності, оновлюється таблиця маршрутизації. Загалом, усе це відбувається з метою підтримання актуальності інформації на всіх пристроях – LSDB має бути однаковою у всіх.

Тут треба обмовитися, що маршрутизатор помічає зміни тільки при прямому підключенні до свого сусіда. Якщо між ними буде, наприклад, комутатор, то пристрій не виявить падіння фізичного інтерфейсу і нічого не робитиме. Для таких ситуацій є два рішення: Налаштувати таймери. Для OSPF їх можна зменшити до рівня мілісекунд; Використовувати протокол BFD (Bidirectional Forwarding Detection). Він дає змогу відстежувати стан лінків на мілісекундному рівні. У конфігурації BFD зв'язується з іншими протоколами і дає змогу дуже швидко повідомити кому треба, що є проблеми на мережі.

Як можна помітити, на всі повідомлення є підтвердження: або це LSAck, або відповідь Hello на Hello. Це реалізовано через відмову від TCP – таким способом можна переконатися в успішному доставленні повідомлень.

2.2 Аналіз проблем та потреб оптимізації протоколу OSPF

Завдяки сукупності своїх характеристик протокол OSPF став одним з основних протоколів внутрішнього шлюзу (Interior Gateway Protocol, IGP) в сучасних комп’ютерних інформаційних мережах [15]. Протокол OSPF поширює інформацію про доступні маршрути між маршрутизаторами однієї автономної системи AS. Оскільки, протокол OSPF згідно з RFC 2328 не є запатентованим тобто є open source протоколом, таким же, як і протокол RIP, це стало одною з причин його широкомасштабного розповсюдження. Також OSPF на відміну від RIP, має суттєво більшу швидкість збіжності (перерахунку таблиці маршрутизації), він не має обмеження на довжину шляху в 15 хопів, враховує пропускну здатність каналів мережі при виборі маршруту. Все це робить OSPF вкрай потужним та масштабованим протоколом внутрішньодоменної маршрутизації.

Протокол OSPF застосовується для динамічної маршрутизації пакетів у низці мереж різного призначення. Серед них такі як:

– корпоративні мережі та великі корпоративні кампуси, де OSPF використовується забезпечення високої швидкості, масштабованості та надійності маршрутизації між окремими будівлями та між багатьма пов'язаними підмережами;

– провайдерські мережі, що використовують OSPF для маршрутизації пакетів між своїми клієнтами та мережею Інтернет;

– також OSPF широко використовується в центрах обробки даних та інших установах для роботи з великими даними Big Data, де потрібна висока пропускна здатність, висока збіжність та масштабованість мережі;

– у широкомасштабних промислових підприємствах OSPF застосовується для маршрутизації між великою кількістю вузлів і підмереж;

– великі університетські або дослідницькі мережі використовують OSPF для забезпечення маршрутизації та зв'язку між різними дослідницькими вузлами, які можуть територіально знаходитися навіть у різних країнах.

Враховуючи ширину застосування протоколу в сучасних мережах, його поважний вік у більш ніж 30 років, очевидно, що в ході використання протоколу OSPF почали виявлятися його суттєві недоліки. Вони могли сильно впливати на швидкість обміну інформації в мережах та їх відмовостійкість. Саме тому багатьма організаціями та вченими проводились дослідження з метою виявлення цих проблем, а також методів їх усунення.

Результатом досліджень стали чітко виділені проблеми протоколу OSPF.

Значною є проблема масштабованості при збільшенні розміру мережі. Хоча OSPF може масштабуватися для підтримки роботи великих мереж, він може стати занадто громіздким і складним в управлінні в мережах з великою кількістю маршрутизаторів та областей. Оскільки при збільшенні кількості маршрутизаторів в межах автономної системи, також збільшується кількість маршрутної інформації, яка передається між маршрутизаторами. Кожна зміна конфігурації мережі призводить до повторного запуску алгоритму SPF на всіх маршрутизаторах мережі. Виконується повний перерахунок таблиць маршрутизації на всіх маршрутизаторах. Це призводить до збільшення кількості передачі трафіку маршрутної інформації,

що має значні наслідки для пропускної здатності інформаційних каналів та впливає на продуктивність мережі.

Підвищене навантаження на процесор і пам'ять, бо в процесі роботи протокол OSPF може створювати значне навантаження на процесор і вбудовану пам'ять маршрутизаторів, що особливо помітно в великих мережах. Йому потрібно багато інформації для розрахунку найкращого маршруту для кожного пункту призначення. Для зберігання цієї інформації протокол OSPF споживає більше пам'яті, ніж інші протоколи маршрутизації. Це може призвести до зниження продуктивності та навіть до збоїв або відмови інформаційних каналів. Своєю чергою, відмова інформаційних каналів або окремих маршрутизаторів призводить до повторного запуску алгоритму SPF та перерахунку таблиць маршрутизації.

Проблема підвищення затримки стає актуальною для великих мереж, оскільки протоколу OSPF потрібно постійно обчислювати найкоротший шлях між усіма парами маршрутизаторів у мережі. Таким чином виникає проблема для задач, які потребують низької затримки, таких як VoIP, пристрої IoT (Internet of Things), онлайн-конференції, фінансові операції, торгівля на біржі та онлайн-ігри.

Відсутність вбудованих механізмів шифрування для пакетів маршрутизації протоколу OSPF, що викликає проблеми з кібербезпекою, а саме неавторизованим доступом до маршрутних таблиць. Тому компаніям та організаціям може знадобитися впровадити додаткові заходи безпеки, наприклад, IPsec, щоб захистити маршрутизацію інформації протоколом OSPF.

Відсутність автентифікації маршрутизаторів, які передають інформацію мережею. Автентифікація може знадобитися для перевірки того, що маршрутизатор, який надсилає інформацію про маршрут, є дійсним маршрутизатором мережі OSPF.

Період часу, що необхідний протоколу OSPF для досягнення збіжності після будь-яких змін в топології мережі, може бути досить великим. Збільшення цього періоду спричиняє тимчасові проблеми із доступністю каналів зв’язку, збільшення часу простою мережі та зниження продуктивності мережі.

Протокол OSPF покладається на широкомовну передачу (broadcast) оновлень стану з'єднань, яка вже стає неефективною у мережах з великою кількістю маршрутизаторів або в певних мережевих топологіях. Розсилка маршрутної інформації на всі порти, включаючи ті, яким ця інформація не призначена, спричиняє зріст завантаженості мережі для передачі непотрібного трафіку.

При організації маршрутизації протоколом OSPF виникають складнощі у налаштуванні OSPF на маршрутизаторах та управлінні процесами маршрутизації, що особливо помітно у великих мережах. Складність зростає, коли потрібно налаштувати такі функції, як автентифікація, підсумовування маршрутів та віртуальні канали. Тому коректне налаштування OSPF можуть провести тільки висококваліфіковані спеціалісти після спеціалізованих експертиз з дослідження топології конкретної мережі.

2.3 Опис існуючих методів оптимізації протоколу OSPF

Розв’язання вищезазначених проблем було вкрай необхідно для коректного та стабільного функціонування мереж автономних систем. В умовах постійного зростання кількості пристроїв, підключених до мережі, та маршрутизаторів, а також збільшення обсягів інформації, що передається мережею, через недоліки протоколу OSPF з’являються проблеми зі стабільною роботою мережевих сервісів. Щоб вирішити або нівелювати ці недоліки були створені ряд запобіжних заходів та модифікацій протоколу OSPF.

Для вирішення проблеми масштабованості в мережі можна використовувати багатозонну архітектуру OSPF (Multi-Area OSPF, M-OSPF), ієрархічний дизайн мережі OSPF (Hierarchical Network Design), налаштування виключень SPF, фільтрацію маршрутів (Route filtering), підсумовування маршрутів (Route Summarization), сегментну маршрутизацію (Segment Routing) та віртуальні канали (Virtual Links).

Ієрархічна (трирівнева) модель міжмережевої взаємодії [16] створена для проєктування надійної, масштабованої та економічно ефективної мережі. Умовно мережі можна класифікувати на основі кількості обслуговуваних пристроїв: малі мережі (до 200 пристроїв), середні мережі (від 200 до 1 000 пристроїв), великі мережі (більше 1000 пристроїв). Конструкції мереж сильно різняться залежно від розміру та потреб організацій. Потреби мережевої інфраструктури для невеликої організації з малою кількістю пристроїв будуть менш складними, ніж інфраструктура великої організації зі значною кількістю пристроїв і з'єднань. Високорівнева топологічна схема для мережі великого підприємства складається з головного кампусу, що з'єднує малі, середні та великі ділянки (рис. 2.12).

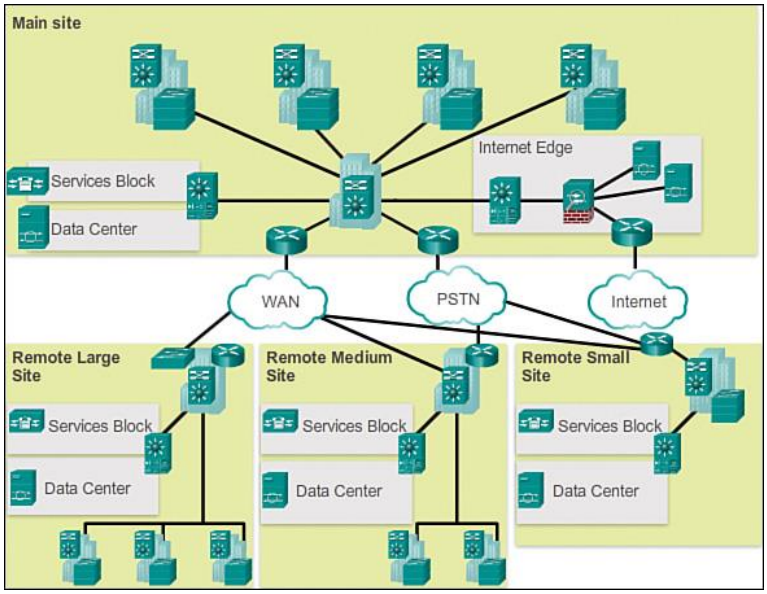


Рисунок 2.12 – Топологія мережі великого підприємства

Ієрархічна структура поділяє мережу на окремі рівні, кожен з яких виконує певну функцію, що визначає його роль у загальній мережі (рис. 2.13). Це дозволяє архітекторам оптимізувати і вибирати відповідне мережеве обладнання, програмне забезпечення та інструменти для виконання конкретних функцій цього рівня мережі. Ієрархічна модель застосовується для проєктування як локальних, так і глобальних мереж. Перевага поділу мережі на менші частини полягає в тому, що локальний трафік залишається локальним. Тільки трафік, призначений для інших мереж, рухається вгору по ієрархії.

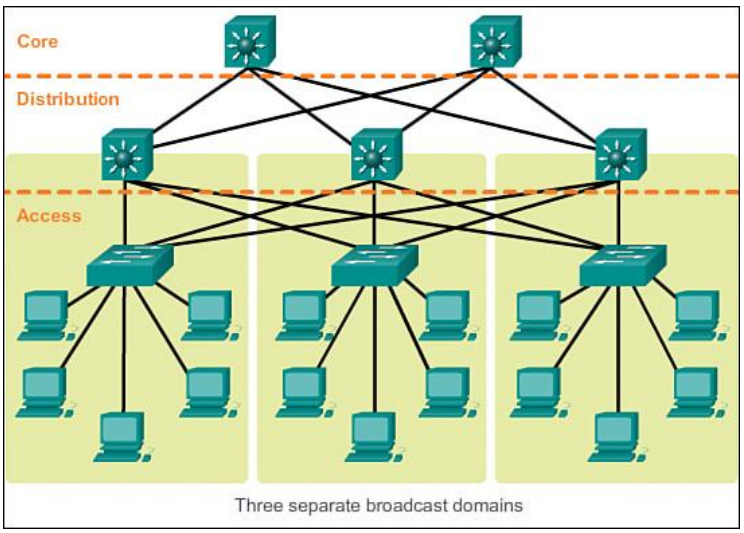


Рисунок 2.13 – Три окремі домени мережі великого підприємства

Локальна мережа кампусу підприємства включає наступні три рівні (рис. 2.14):

– рівень доступу забезпечує доступ робочих груп/користувачів до мережі;

– рівень розподілу забезпечує підключення на основі політик і контролює межу між рівнем доступу й основним рівнем;

– рівень ядра забезпечує швидкий транзит трафіку між розподільчими комутаторами в межах корпоративного кампусу.

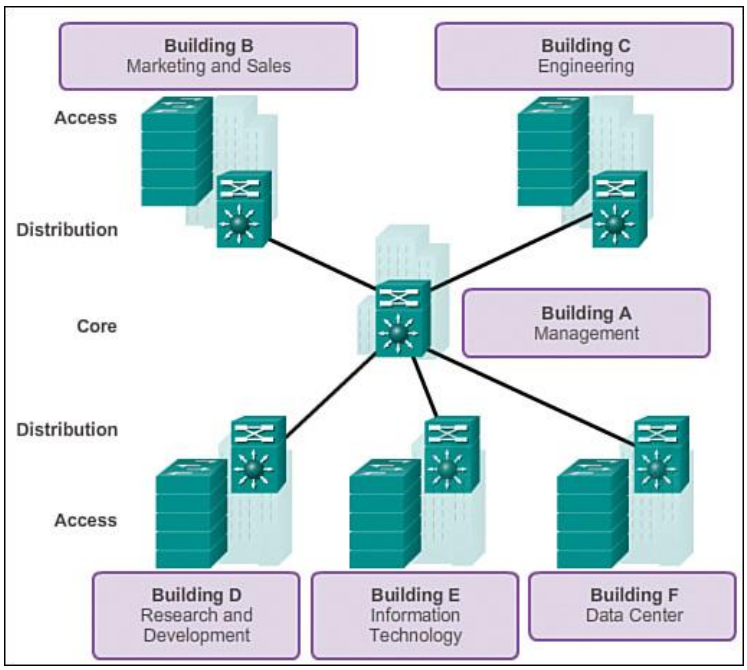


Рисунок 2.14 – Мережі з багатьох будівель

Всередині локальної мережі виділений рівень доступу надає кінцевим пристроям доступ до мережі. У середовищі WAN він може надавати віддаленим працівникам або віддаленим об'єктам доступ до корпоративної мережі через WAN-з'єднання.

Рівень доступу виконує ряд деяких функцій, зокрема:

– комутація другого рівня;

– висока доступність каналів зв’язку;

– безпека портів;

– класифікація та маркування QoS і межі довіри;

– перевірка протоколу ARP;

– віртуальні списки контролю доступу (VACL);

– Power over Ethernet (PoE) і допоміжні VLAN для VoIP.

Рівень розподілу агрегує дані, отримані від комутаторів рівня доступу, перед тим, як вони передаються на рівень ядра для маршрутизації до кінцевого пункту призначення. Пристрій рівня розподілу є центральною точкою в розподільчих шафах. Для сегментації робочих груп та ізоляції мережевих проблем у кампусі використовується маршрутизатор, що надає висхідні послуги для маршрутизаторів рівня доступу.

Рівень розподілу надає такі функції:

– агрегацію каналів LAN або WAN;

– безпеку на основі політик у вигляді списків контролю доступу (ACL) і фільтрації;

– послуги маршрутизації між локальними й віртуальними мережами, а також між доменами маршрутизації (наприклад, EIGRP і OSPF);

– резервування і балансування навантаження;

– межа для агрегації та підсумовування маршрутів, налаштована на інтерфейсах у напрямку до базового рівня;

– широкомовний контроль домену, оскільки маршрутизатори не пересилають широкомовні повідомлення.

Рівень ядра складається з високошвидкісних мережевих пристроїв, призначені для якнайшвидшої комутації пакетів і з'єднання різних компонентів кампусу, таких як розподільчі модулі, сервісні модулі, центри обробки даних і межа WAN. Ядро повинно мати надлишкові апаратні можливості. Ядро агрегує трафік з усіх пристроїв рівня розподілу, тому воно повинно бути здатним швидко пересилати великі обсяги даних.

На рівні ядра необхідно враховувати такі фактори:

– високошвидкісна маршрутизація;

– надійність та відмовостійкість;

– масштабування шляхом використання швидшого, а не потужнішого обладнання;

– уникнення маніпуляцій з пакетами, що вимагають великих витрат потужностей процесора, викликаних потребами безпеки, перевіркою, класифікацією якості обслуговування (QoS) або іншими процесами.

Фільтрація маршрутів OSPF [17] – це метод визначення маршрутів, які можуть бути оголошені або отримані від сусідніх маршрутизаторів, проте неможливо фільтрувати маршрути, які оголошені між маршрутизаторами в одній зоні. Це означає, що нам потрібно налаштувати фільтрацію на ABR або ASBR. ABR може фільтрувати LSA типів 3 і 4, а ASBR може фільтрувати LSA типів 5 і 7. Фільтрація використовується для оптимізації пам'яті, інженерії трафіку, та підвищення безпеки. Кожен маршрутизатор у зоні має спільний доступ до повної копії бази даних OSPF. Фільтрацію в OSPF доречніше називати фільтрацією LSA, бо фільтруються саме оголошення LSA, які відповідають певним мережевим адресатам. Причиною для того, що фільтрація відбувається тільки на ABR або ASBR, є те, що як протокол стану каналу, кожен маршрутизатор в межах однієї області за визначенням повинен мати ідентичну базу даних стану каналу (LSDB) для обчислення найкоротшого шляху (SPF).

По суті, *підсумовування маршрутів* [18] – це процес об'єднання підмереж у більшу підмережу, яка буде оголошена вищим маршрутизаторам для полегшення управління/виправлення несправностей і економії ресурсів. Однак при роботі з

OSPF ви можете об'єднувати підмережі тільки на рівні ABR або ASBR. Підсумовування в межах області неможливе. OSPF використовує LSA типу 3 для міжобласних маршрутизаторів і LSA типу 5 для зовнішніх префіксів, які перерозподіляються в OSPF. В OSPF існує два типи підсумовування: міжобласне підсумовування та зовнішнє підсумовування. Міжобласне підсумовування маршрутів може бути виконано тільки на прикордонному (Area Border Router, ABR) і підсумовувати маршрути з певної області в магістральну область. Підсумовування зовнішніх маршрутів може бути виконано тільки на ASBR і підсумовує маршрути до зовнішнього пункту призначення.

Сегментна маршрутизація [19] (Segment Routing, SR) дозволяє гнучко визначати наскрізні шляхи в топології IGP, кодуючи шляхи як послідовності топологічних субшляхів, які називаються сегментами. Ці сегменти оголошуються протоколами маршрутизації стану каналу (IS-IS і OSPF). Префіксні сегменти представляють найкоротший шлях до префікса (або вузла) з урахуванням ECMP, відповідно до стану топології IGP. Сегменти суміжності являють собою стрибок через певну суміжність між двома вузлами в IGP. Зазвичай, префіксний сегмент – це шлях з декількома переходами, в той час, як сегмент суміжності, в більшості випадків, є шляхом з одним переходом. Площина керування SR може бути застосована як до площини даних IPv6, так і до площини даних MPLS, і вона не вимагає ніякої додаткової сигналізації (крім розширень IGP). Площина даних IPv6 виходить за рамки цієї специфікації; вона не застосовується до OSPFv2, який підтримує тільки сімейство адрес IPv4. При використанні в мережах MPLS шляхи SR не потребують сигналізації LDP або RSVP-TE. Однак, SR може взаємодіяти в присутності LSP, створених за допомогою RSVP або LDP. В сегментній маршрутизації існують кілька ідентифікаторів сегментів (SID): префіксний SID, SID суміжності, SID суміжності з локальною мережею, зв’язаний SID.

Для розв’язання проблеми підвищеної затримки каналів зв’язку та проблеми доступності можна використовувати методи пришвидшеної конвергенції, такі як налаштування Dead та Hello таймерів, Fast Hello, виявлення двонаправленої переадресації (BFD), Loop-Free Alternates (LFAs) та інкрементний SPF (iSPF).

Пакети Fast hello [20] – це hello-пакети, які надсилаються з інтервалом не менш як 1 секунди. Маршрутизатори надсилають hello-пакети своїм сусідам для підтримки зв'язку з ними. Пакети привітання надсилаються з інтервалом, що налаштовується у секундах. За замовчуванням це 10 секунд для каналу Ethernet і 30 секунд для не широкомовного каналу. Підвищення частоти надсилання hello-пакетів дає змогу збільшити швидкість збіжності мережі та швидкість виявлення сусідів, проте збільшує навантаження на інформаційні канали та апаратну частину маршрутизаторів. Пакети привітання містять список усіх сусідів, для яких було отримано пакет привітання протягом мертвого інтервалу. Мертвий інтервал (dead) також можна конфігурувати у секундах, і за замовчуванням він у чотири рази перевищує значення інтервалу привітання. Значення всіх інтервалів привітання та мертвих інтервалів має бути однаковим у мережі. В ситуації коли маршрутизатор не отримує пакет привітання від сусіда протягом мертвого інтервалу, він оголошує сусіда неактивним. Fast hello пакети конфігуруються за допомогою команди ip ospf dead-interval. Мертвий інтервал встановлюється на 1 секунду, а значення hello-множника встановлюється на кількість пакетів привітання, які ви хочете відправити протягом цієї 1 секунди, таким чином забезпечуючи мілісекундні або "швидкі" пакети привітання. Коли на інтерфейсі налаштовано надсилання hello-пакетів з інтервалом менше ніж 1 секунда, то в hello-пакеті, який надсилають із цього інтерфейсу, hello-інтервал дорівнюватиме 0. Hello-інтервал, отриманий у hello-пакетах, які приходять на цей інтерфейс, ігнорується. Dead-інтервал має бути однаковим.

Bidirectional Forwarding Detection (BFD) [21] – це короткочасний метод виявлення збоїв на шляху переадресації між двома сусідніми маршрутизаторами з низькими накладними витратами, включаючи інтерфейси, канали передачі даних і площини переадресації. BFD вмикається на рівні інтерфейсу та протоколу маршрутизації.

Протокол BFD практично не впливає на продуктивність CPU маршрутизатора, тому інтервал між BFD пакетами може бути скорочено до

мінімальних 50 мс, що забезпечує швидку збіжність. При цьому інтервалі в 50 мс максимальна кількість стабільних сеансів становить 4 штуки.

Маршрутизатори Cisco підтримують асинхронний режим BFD, в якому два маршрутизатори обмінюються керівними пакетами для активації та підтримки сусідніх сеансів BFD. Щоб створити сеанс BFD, його необхідно налаштувати на обох системах. Після включення BFD на рівні інтерфейсу і маршрутизатора для відповідних протоколів маршрутизації створюється сеанс BFD, узгоджуються таймери BFD, і BFD-сусіди починають надсилати один одному керівні BFD-пакети з узгодженим інтервалом.

Loop-Free Alternate (LFA) Fast Reroute (FRR) [22] дозволяє OSPF швидко перемикатися (протягом 50 мс) на резервний шлях, коли основний шлях виходить з ладу. Без LFA FRR, OSPF повинен перезапустити SPF, щоб знайти новий шлях, коли основний шлях виходить з ладу. Коли в мережі відбувається відмова каналу або вузла, неминуче виникає період перебоїв в доставці трафіку, поки протокол маршрутизації не адаптується до нової топології. У сучасному світі додатки дуже чутливі до будь-якої втрати трафіку, і тому переривання трафіку, викликане конвергенцією протоколів стану з'єднання, таких як OSPF і Intermediate System – Intermediate System (ISIS), може негативно вплинути на роботу сервісів. В стоковому стані протокол OSPF не розраховує резервний маршрут. Метою використання LFA є розрахунок резервного маршруту для маршрутизації трафіку у випадку відмови безпосередньо підключеного каналу або вузла на первинному шляху. LFA розраховує резервний наступний хоп для кожного основного наступного хопу і відповідно програмує таблицю Cisco Express Forwarding (CEF).

Для OSPF існують два методи обчислення LFA:

– для кожного каналу: всі префікси, які доступні через певний канал, мають однакову адресу наступного переходу. IGP може обчислити резервну адресу наступного переходу для всіх префіксів, які використовують один і той самий канал. Якщо канал виходить з ладу, тоді всім префіксам автоматично призначається однакова резервна адреса наступного переходу. Перевагою LFA на канал є те, що він вимагає менше процесорних циклів і пам'яті, ніж LFA на префікс. Недоліком,

однак, є те, що як тільки первинний канал виходить з ладу, ви на резервний канал раптово перекладається велике навантаження;

– для кожного префікса: IGP обчислює LFA для кожного префікса. Це вимагає більше процесорних циклів і пам'яті, але забезпечує покращене балансування навантаження. При виходу з ладу основного шляху, префікси можуть використовувати різні резервні шляхи, розподіляючи трафік всією мережею.

Запуск повного обчислення SPT при зміні топології коли в області відбуваються зміни в оголошенні про стан з'єднання (LSA) типу 1 або 2 – це добре, бо тоді створюється оновлений SPT з найкоротшими шляхами до всіх пунктів призначення. Недоліком, однак, є те, що також обчислюються шляхи, які не змінилися з моменту останнього повного SPF. Протоколом OSPF підтримується метод перерахунку тільки тієї частини SPT, яка змінилася, який називається *інкрементним SPF (iSPF)* [23]. Оскільки тепер не потрібно постійно перераховувати SPF повністю, навантаження на процесор маршрутизатора зменшується, а час збіжності покращується. З іншого боку, маршрутизатор зберігає попередню стару копію SPT, що займає додатковий об'єм пам'яті. Однак якщо зміна оголошень LSA типу 1 або 2 відбувається на самому обчислювальному маршрутизаторі, то виконується повне обчислення SPT. Існує три сценарії, в яких інкрементний SPF має позитивний вплив:

– додавання (або видалення) листового вузла до гілки;
– обрив зв'язку у non-SPT;
– обрив зв'язку у гілці SPT.

Для зменшення завантаженості мережевих каналів та ресурсів заліза маршрутизаторів можна використовувати пасивні інтерфейси, Loopback інтерфейси та тупикові області (stub areas), протокол RSVP (Resource Reservation Protocol) для резервування пропускної здатності певних потоків трафіку.

Інтерфейс зворотного зв'язку Loopback [24] – це віртуальний інтерфейс, який залишається увімкненим (активним) після команди no shutdown, доки не вимкнути його за допомогою команди shutdown. Інтерфейс зворотного зв'язку допомагає подолати збої на шляху. Він доступний з будь-якого фізичного інтерфейсу і не

залежить від їх стану, тому, якщо хоч один з фізичних інтерфейсів активний, доступ до маршрутизатора залишається через loopback інтерфейс. Це робить loopback інтерфейси ідеальними для призначення IP-адрес, коли потрібна єдина адреса, яка не залежить від стану будь-якого фізичного інтерфейсу мережевого пристрою. Використовуються для тестування та адміністрування мережі, для ідентифікації маршрутизатора у мережі OSPF, незалежно від стану фізичних інтерфейсів, для визначення доступності каналу зв'язку та в якості резервного інформаційного каналу.

Для підвищення кібербезпеки маршрутизації протоколом OSPF можна використовувати механізми шифрування та автентифікації за допомогою протоколу IPsec, з використанням алгоритмів MD5 та SHA.

Автентифікація використовується для перевірки того, що маршрутизатор, який надсилає інформацію про маршрут, є дійсним маршрутизатором мережі OSPF, а також використовується для запобігання несанкціонованого доступу до пакетів та їх фальсифікації. Це допомагає запобігти неавторизованим або незахищеним маршрутизаторам від надсилання оновлень маршрутизації, які можуть спричинити збої в роботі мережі. Алгоритм MD5 шифрує будь-які дані у форматі 128-бітового хешу (контрольної суми), який досить складно підробити. Алгоритм використовується для перевірки довжини даних, коли відбувається їх передача в зашифрованому вигляді. В сучасних умовах MD5 хоч і є швидшим за SHA, але є менш захищеним до злому, тому він визнаний застарілим. Також при великих об'ємах даних можуть виникати криптографічні колізії, тобто створення двох різних вхідних повідомлень, які дають однаковий хеш. Алгоритм SHA-3 є сучаснішим та в змозі створювати цифрові відбитки різної довжини (224, 256, 384 або 512 біт) із вхідних даних будь-якого розміру.

Найпоширенішим протоколом для шифрування інформації є *комплекс протоколів IPsec* [25]. Він дозволяє здійснювати підтвердження автентичності (автентифікацію), перевірку цілісності та шифрування IP-пакетів. IPsec також включає протоколи для захищеного обміну ключами в інтернеті. Комплекс протоколів IPsec захищає від таких типів атак:

– атаки перехоплення та зміни даних, що спрямовані на модифікацію даних в OSPF пакетах (Data Modification);

– атака отруєння маршрутних таблиць (Routing Table Poisoning), що використовується зловмисниками для модифікації таблиць маршрутизації шляхом впровадження фальшивих маршрутів;

– атака IP Spoofing полягає у підробці зловмисником IP-адреси для того, щоб виглядати як легітимний вузол мережі, що використовується для передачі фальшивих пакетів, введення фальшивих маршрутів або порушення топології мережі;

– атака IP Sniffing полягає у перехопленні пакетів в мережі для того, щоб дізнатися про структуру мережі, дізнатися маршрутну інформацію або щоб отримати конфіденційну інформацію, таку як облікові дані та особисті повідомлення;

– атака перехоплення сеансу (Session Hijacking), коли зловмисник намагається перехопити діючий сеанс між двома вузлами мережі;

– атака повторення пакетів (Replay), коли зловмисник перехоплює та повторює раніше відправлені пакети в мережі з метою виконання певних дій більше одного разу або з метою збільшення часу виконання алгоритму SPF. Для запобігання цьому використовується механізм Replay Protection;

– атаки з використанням розриву з'єднання (Denial of Service), що спрямовані на перенавантаження мережевих ресурсів, що запобігаються шляхом заборони неправомірного доступу до мережевих послуг;

– атака (Man-in-the-middle) коли зловмисник перехоплює та контролює комунікацію між двома сторонами без їхнього відома з метою перегляду, перехоплення та модифікації OSPF пакетів;

– атака Hello пакетами, коли зловмисники посилають в мережу велику кількість Hello пакетів для перенавантаження мережі або зміни її топології для додавання скомпрометованих маршрутів.

Архітектура IPsec складається з трьох рівнів (рис. 2.15): верхній рівень для захисту віртуального каналу протоколами AH та ESP, середній рівень з

криптографічними алгоритмами та нижній рівень з базою даних про всі протоколи й алгоритми IPsec.

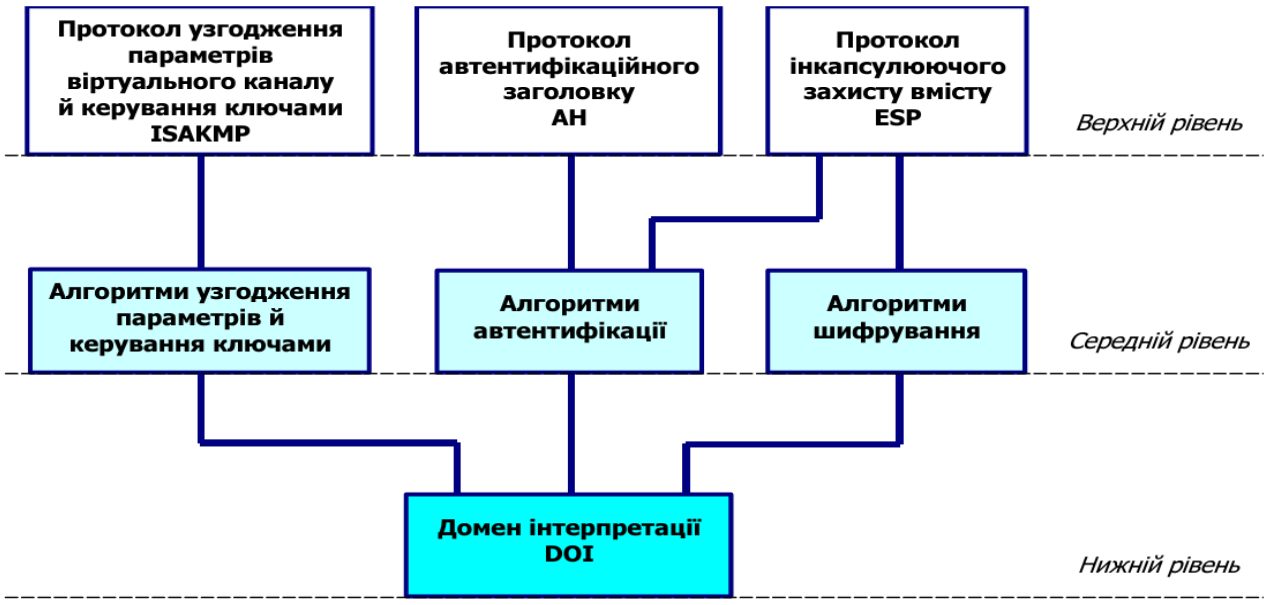


Рисунок 2.15 – Рівні структури IPSec

На верхньому рівні протокол Authentifitication Header (AH) забезпечує автентифікацію, цілісність IP-пакетів та захист даних від відтворення зловмисниками в мережі. Проте AH не має механізмів для захисту конфіденційності даних – зловмисники не можуть їх змінювати, але можуть проглядати. А от вже протокол Encapsulating Security Payload (ESP) забезпечує конфіденційність даних шляхом шифрування (рис. 2.16). Крім того, він дозволяє ідентифікувати відправника даних, а також забезпечити цілісність даних та захист від відтворення інформації.

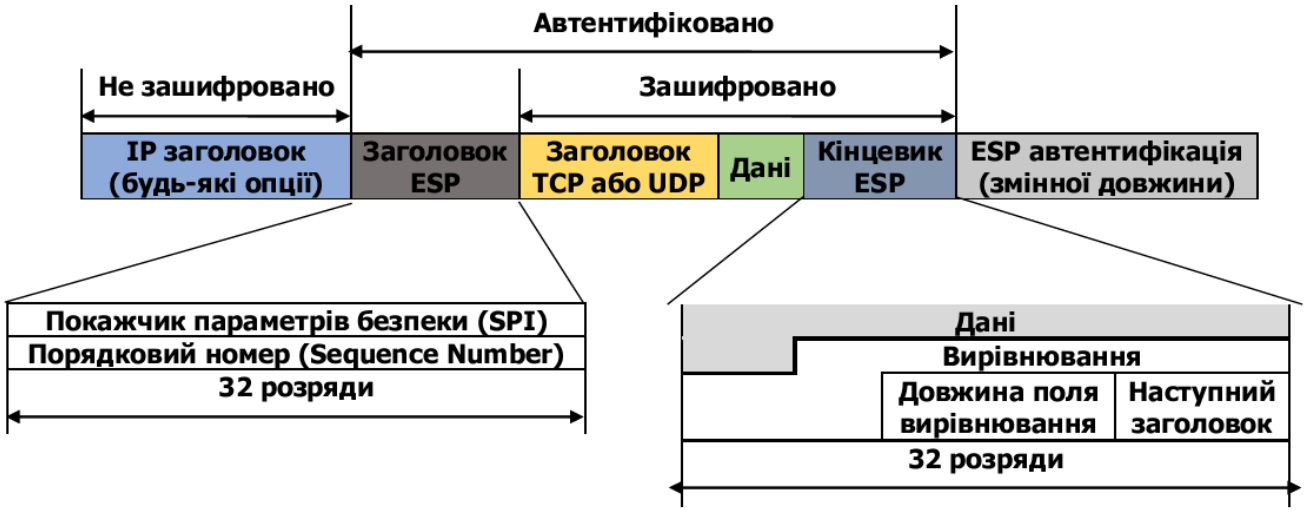


Рисунок 2.16 – Протокол інкапсулюючого захисту (ESP)

Коли одночасно застосовуються засоби шифрування та ідентифікації даних, то система спочатку ідентифікує пакет, а якщо ідентифікація виконана успішно, то розшифровує пакет. Такий спосіб обробки пакетів знижує навантаження на систему та зменшує ризик злому захисту за допомогою атаки типу "відмова в обслуговуванні". Протокол ESP застосовується двома способами: у режимі відкритої передачі та у режимі тунелю. У режимі відкритої передачі заголовок ESP вказується після заголовка IP дейтаграми. Якщо дейтаграма вже має заголовок IPSec, то заголовок ESP розміщується перед цим заголовком. Кінцевик ESP та ідентифікаційні дані, якщо вони є, вказуються після поля даних.

РОЗДІЛ 3. АНАЛІЗ ТА ПОРІВНЯННЯ МЕТОДІВ ОПТИМІЗАЦІЇ ПРОТОКОЛУ OSPF

3.1 Налаштування мережі за допомогою OSPF у Cisco Packet Tracer

Для дослідження процесу роботи протоколу OSPF [26, 27, 28, 29, 30] на практиці було обрано програму для симуляції роботи реальних комп'ютерних мереж Packet Tracer від компанії Cisco.

Після запуску програми Packet Tracer була створена проста мережа (рис. 3.1) для тестування протоколу OSPF, що складалася з маршрутизаторів R1, R2, R3 та кінцевих пристроїв PC0 та PC1, що є персональними комп'ютерами. З'єднання маршрутизаторів між собою було виконано за допомогою кабелів Serial DCE. З'єднання комп'ютерів та маршрутизаторів було виконано прямими мідними кабелями.

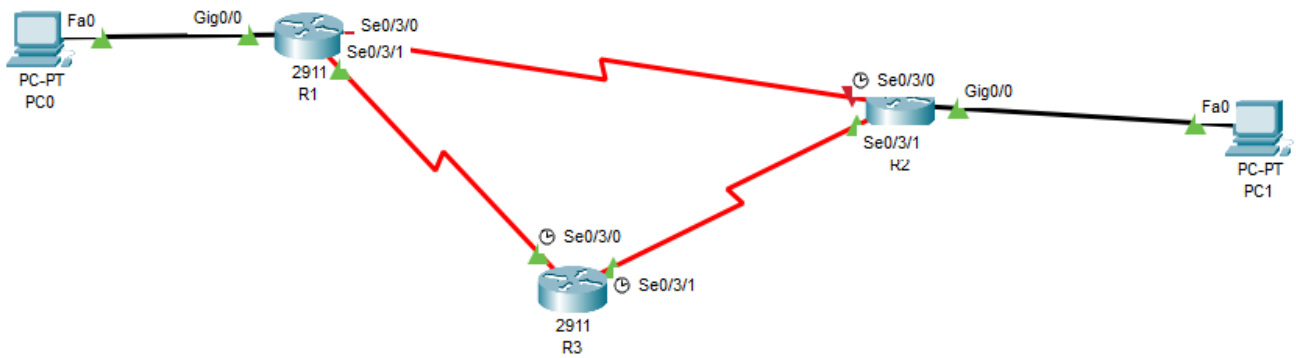


Рисунок 3.1 – Мережа для тестування протоколу OSPF

Слідом на комп'ютерах PC0 та PC1 було налаштовано IP-адресацію згідно з таблицею 3.1.

Таблиця 3.1 – Схема IP-адресації на комп’ютерах PC0 і PC1

Пристрій	IP-адреса	Маска підмережі	Шлюз
PC0	192.168.1.11	255.255.255.0	192.168.1.1
PC1	192.168.2.11	255.255.255.0	192.168.2.1

Наступним кроком була налаштована IP-адресація на маршрутизаторах R1, R2 та R3 згідно з таблицею 3.2 та рис. 3.2, рис. 3.3, рис. 3.4.

Таблиця 3.2 – Схема IP-адресації на маршрутизаторах R1, R2 та R3

Пристрій	Інтерфейс	IP-адреса	Маска підмережі	Clock rate
R1	Gig0/0	192.168.1.1	255.255.255.0	-
	Se0/3/0	20.0.0.1	255.0.0.0	64000
	Se0/3/1	10.0.0.2	255.0.0.0	64000
R2	Gig0/0	192.168.2.1	255.255.255.0	-
	Se0/3/0	20.0.0.2	255.0.0.0	Auto
	Se0/3/1	30.0.0.2	255.0.0.0	64000
R3	Se0/3/0	10.0.0.1	255.0.0.0	Auto
	Se0/3/1	30.0.0.1	255.0.0.0	Auto

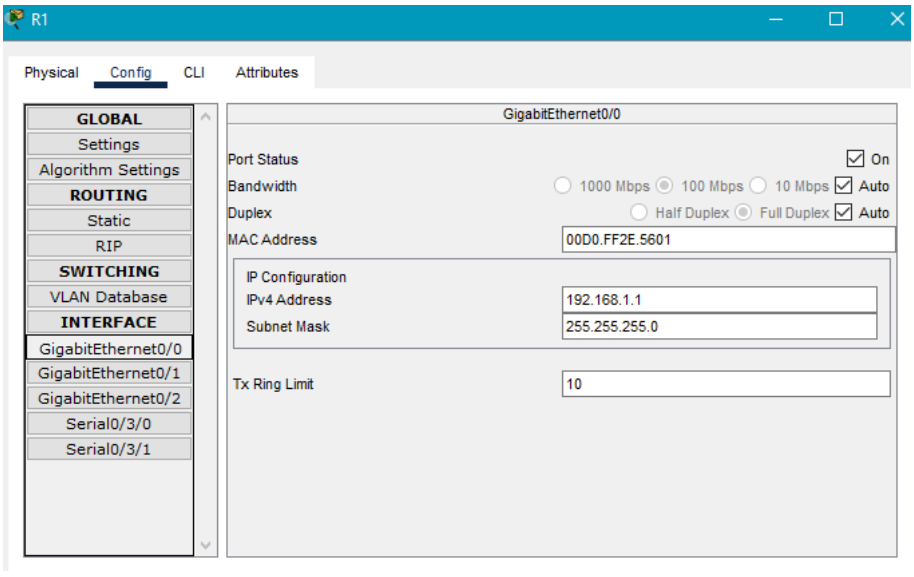


Рисунок 3.2 – Конфігурація порту Gig0/0 на R1

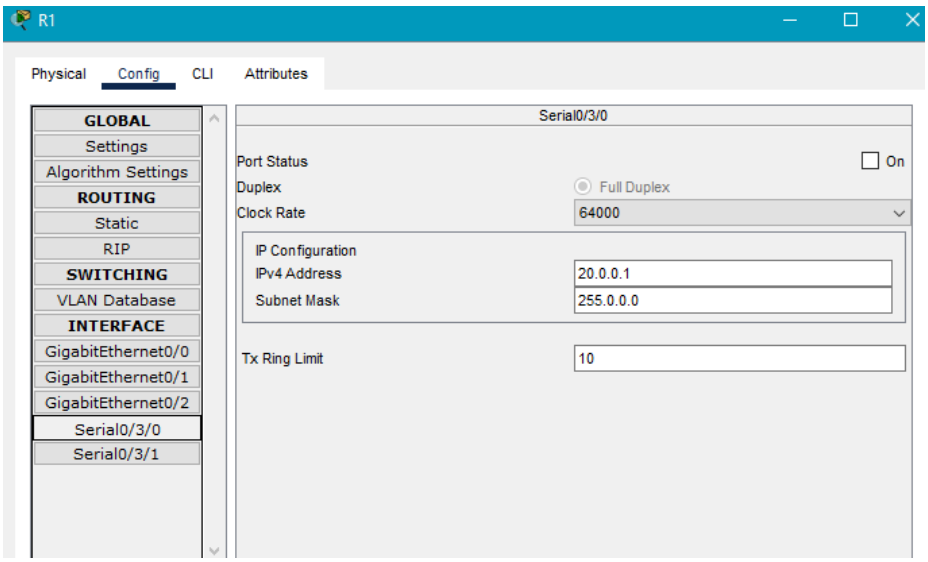


Рисунок 3.3 – Конфігурація порту Se0/3/0 на R1

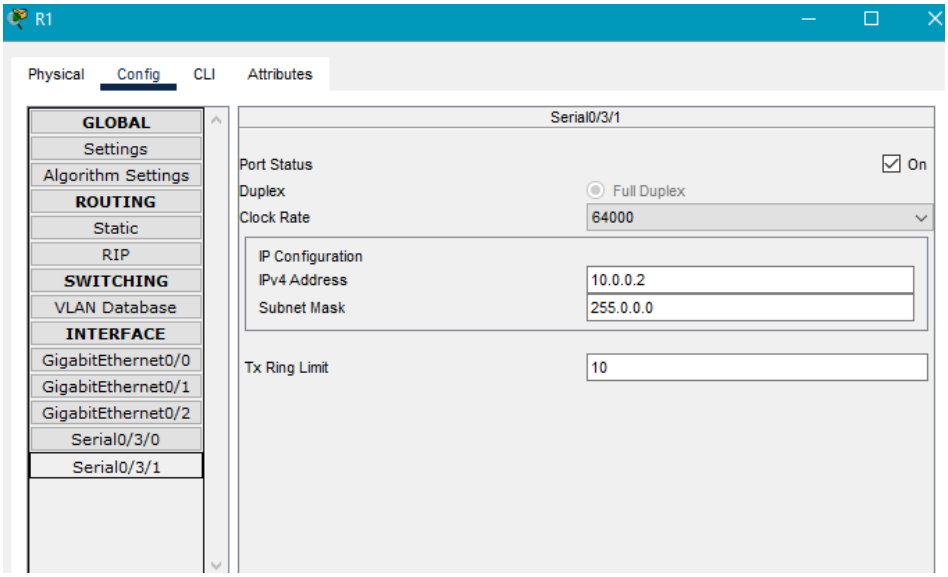


Рисунок 3.4 – Конфігурація порту Se0/3/1 на R1

Для встановлення динамічної маршрутизації протоколом OSPF були виконані відповідні налаштування конфігурації на маршрутизаторах R1, R2 та R3. Налаштування виконувались в CLI (рис. 3.5).

Налаштування маршрутизатора R1:

1. *Router>enable*;
2. *Router#configure terminal*;
3. *Router(config)#hostname R1*;
4. *R1(config)#router ospf 1*;

5. *R1(config-router)#network 192.168.1.0 0.0.0.255 area*;
6. *R1(config-router)#network 10.0.0.0 0.255.255.255 area 0*;
7. *R1(config-router)#network 20.0.0.0 0.255.255.255*;
8. *R1(config-router)#end*;
9. *R1# copy running-config startup-config*.

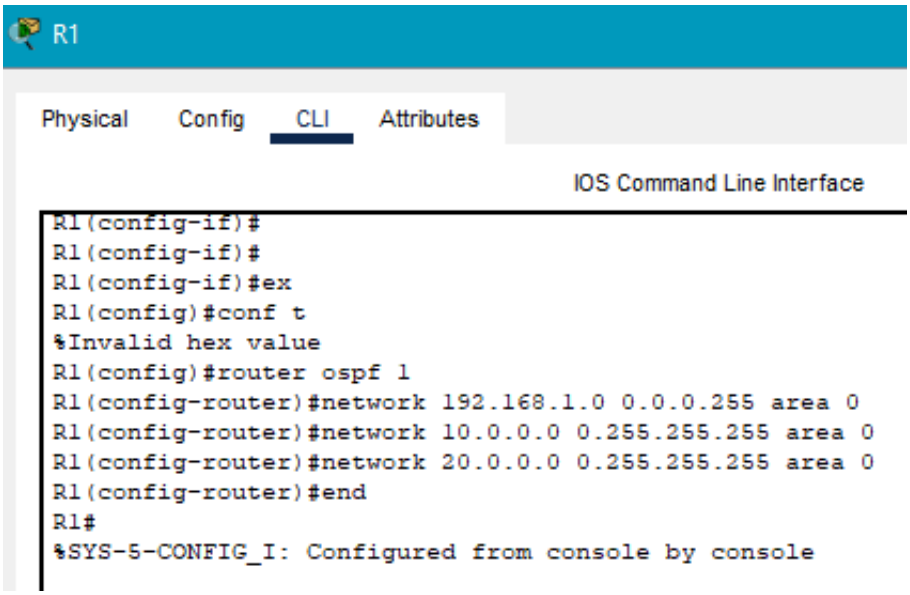


Рисунок 3.5 – Конфігурація OSPF на R1

Налаштування маршрутизатора R2 (рис. 3.6):

1. *Router>enable*;
2. *Router#configure terminal*;
3. *Router(config)#hostname R2*;
4. *R2(config)#router ospf 1*;
5. *R2(config-router)#network 192.168.2.0 0.0.0.255 area*;
6. *R2(config-router)#network 20.0.0.0 0.255.255.255 area 0*;
7. *R2(config-router)#network 30.0.0.0 0.255.255.255*;
8. *R2(config-router)#end*;
9. *R2# copy running-config startup-config*.

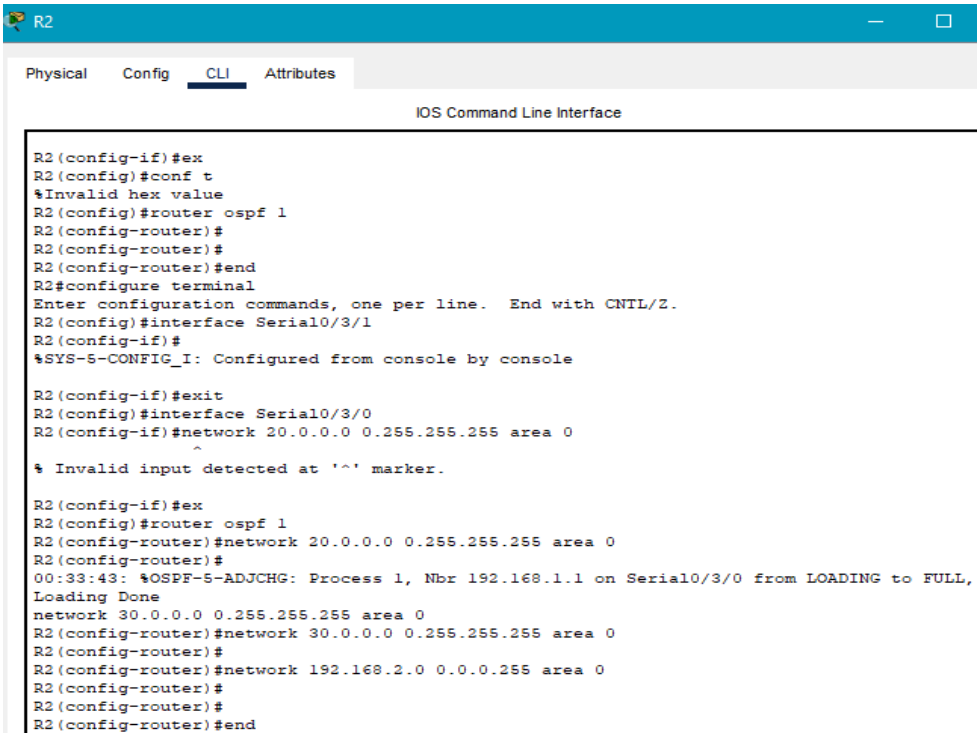


Рисунок 3.6 – Конфігурація OSPF на R2

Налаштування маршрутизатора R3 (рис. 3.7):

1. *Router>enable*;
2. *Router#configure terminal*;
3. *Router(config)#hostname R3*;
4. *R3(config)#router ospf 1*;
5. *R3(config-router)#network 10.0.0.0 0.255.255.255 area 0*;
6. *R3(config-router)#network 30.0.0.0 0.255.255.255 area 0*;
7. *R3(config-router)#end*;
8. *R3# copy running-config startup-config*.

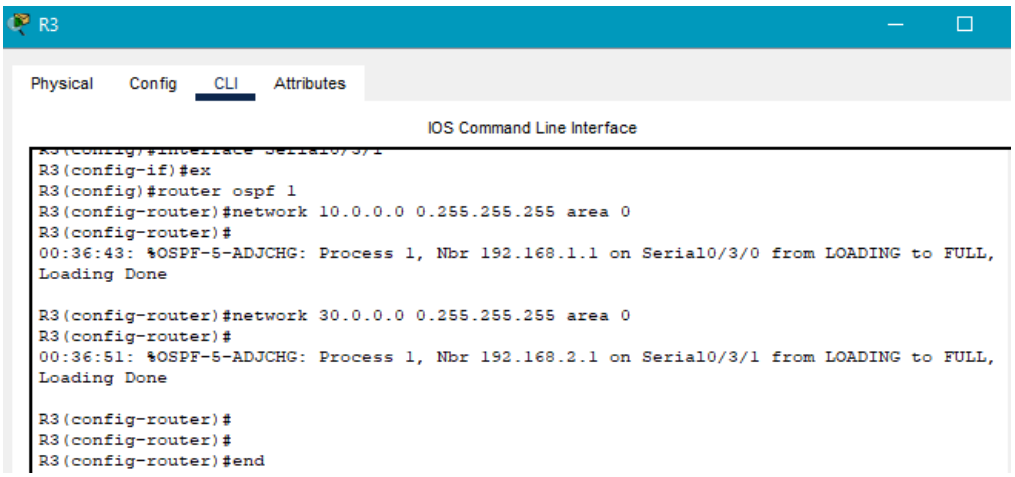


Рисунок 3.7 – Конфігурація OSPF на R3

Після всіх закінчення всіх налаштувань виконується перевірка працездатності створеної мережі шляхом надсилання пакета Simple PDU з PC0 на PC1. Тут можна побачити (рис. 3.8), що пакет проходить найшвидшим шляхом через маршрутизатори R1 та R2.

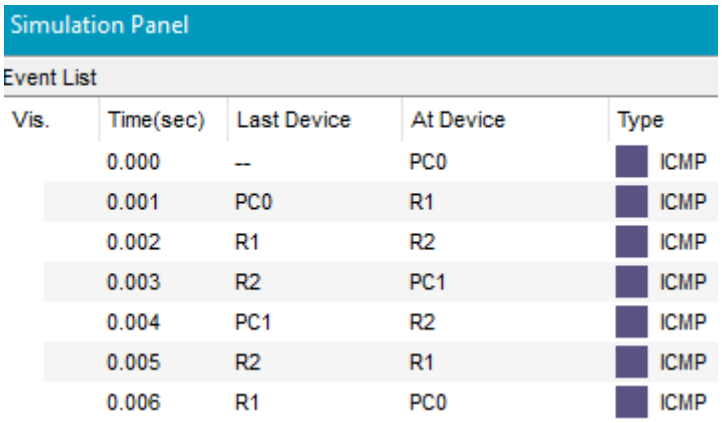


Рисунок 3.8 – Пакет пройшов найшвидшим шляхом

Проте це не є дійсним доказом функціонування протоколу OSPF. Щоб довести коректну роботу цього протоколу треба відключити інтерфейс Se0/3/0 на R1 (рис. 3.9), тим самим розірвавши прямий зв’язок між маршрутизаторами R1 та R2. Завдяки протоколу OSPF передача повідомлень між PC0 та PC1 можлива через маршрут R1> R3> R2.

Simulation Panel

Event List

Vis.	Time(sec)	Last Device	At Device	Type
	0.000	--	PC0	ICMP
	0.001	PC0	R1	ICMP
	0.002	R1	R3	ICMP
	0.003	R3	R2	ICMP
	0.004	R2	PC1	ICMP
	0.005	PC1	R2	ICMP
	0.006	R2	R3	ICMP
	0.007	R3	R1	ICMP
	0.008	R1	PC0	ICMP

Рисунок 3.9 – Пакет пройшов довшим шляхом через R3

Щоб переконатися про створення сусідських зв’язків між маршрутизаторами мережі потрібно переглянути вміст таблиць сусідських зв’язків на кожному з маршрутизаторів (рис. 3.10, рис. 3.11, рис. 3.12).

R1

```
R1>en
R1#show ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
192.168.2.1       0   FULL/  -        00:00:31    20.0.0.2        Serial0/3/0
30.0.0.1          0   FULL/  -        00:00:36    10.0.0.1        Serial0/3/1
R1#
```

Рисунок 3.10 – Таблиця сусідських зв’язків на R1

R2

```
R2#
%SYS-5-CONFIG_I: Configured from console by console

R2#show ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
30.0.0.1          0   FULL/  -        00:00:37    30.0.0.1        Serial0/3/1
192.168.1.1       0   FULL/  -        00:00:31    20.0.0.1        Serial0/3/0
R2#
```

Copy

Рисунок 3.11 – Таблиця сусідських зв’язків на R2

R3

```
R3#
%SYS-5-CONFIG_I: Configured from console by console

R3#show ip ospf neighbor


Neighbor ID     Pri   State           Dead Time   Address         Interface
192.168.1.1       0   FULL/  -        00:00:37    10.0.0.2        Serial0/3/0
192.168.2.1       0   FULL/  -        00:00:32    30.0.0.2        Serial0/3/1
R3#
```

Copy

Рисунок 3.12 – Таблиця сусідських зв’язків на R3

Тепер потрібно переглянути вміст таблиць маршрутизації (рис. 3.13, рис. 3.14, рис. 3.15).

R1

```
R1#
R1#show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

     10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C       10.0.0.0/8 is directly connected, Serial0/3/1
L       10.0.0.2/32 is directly connected, Serial0/3/1
     20.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C       20.0.0.0/8 is directly connected, Serial0/3/0
L       20.0.0.1/32 is directly connected, Serial0/3/0
O    30.0.0.0/8 [110/128] via 10.0.0.1, 00:06:02, Serial0/3/1
                [110/128] via 20.0.0.2, 00:06:02, Serial0/3/0
     192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C       192.168.1.0/24 is directly connected, GigabitEthernet0/0
L       192.168.1.1/32 is directly connected, GigabitEthernet0/0
O    192.168.2.0/24 [110/65] via 20.0.0.2, 00:06:02, Serial0/3/0
```

Рисунок 3.13 – Таблиця маршрутизації на R1

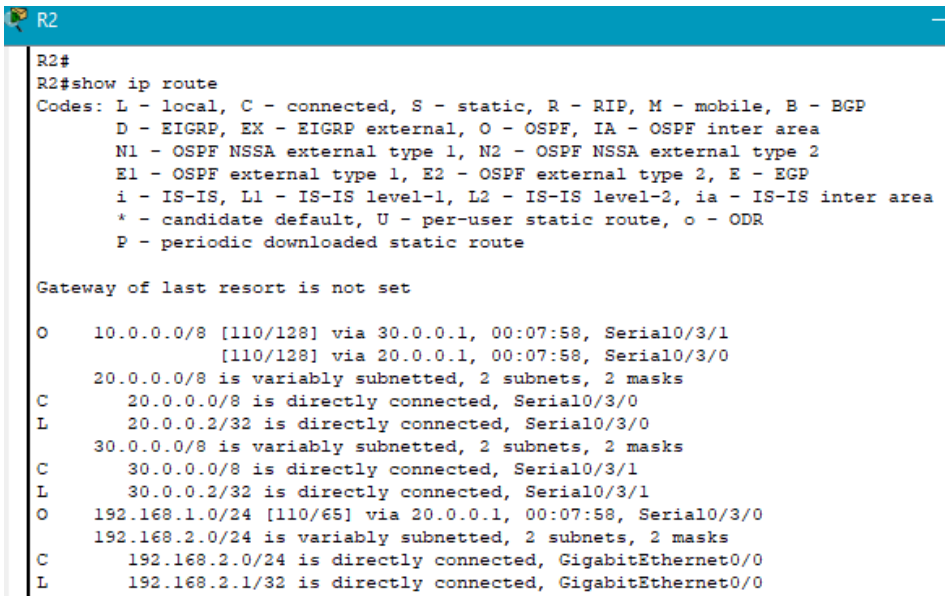


Рисунок 3.14 – Таблиця маршрутизації на R2

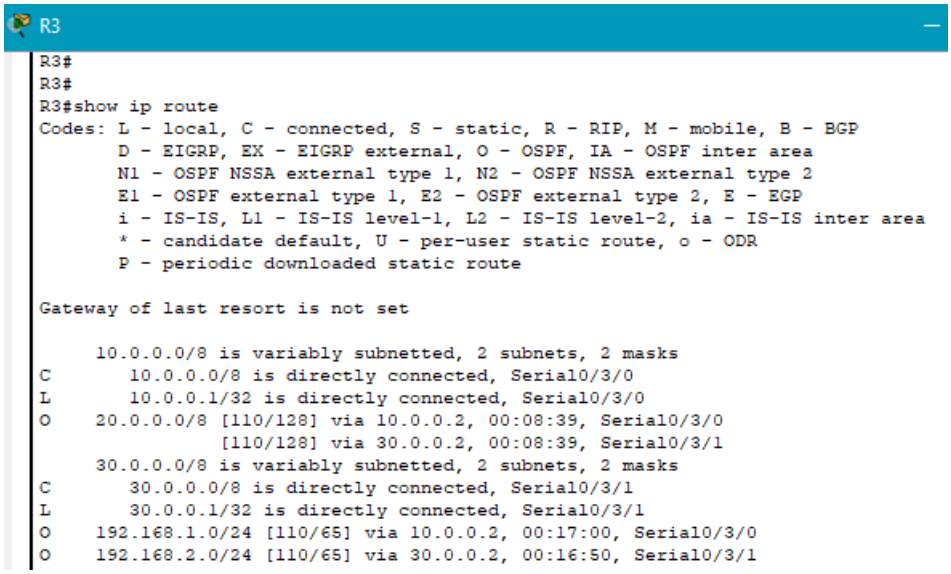


Рисунок 3.15 – Таблиця маршрутизації на R3

Після проведення вищезазначених дій можна дійти висновку, що протокол OSPF працює правильно та динамічна маршрутизації між клієнтами мережі можлива.

3.2 Використання методів оптимізації OSPF у Cisco IOS

Cisco IOS – це багатозадачна операційна система для маршрутизаторів і мережевих комутаторів Cisco, що виконує функції мережевої організації, маршрутизації, комутації та передачі даних.

Фільтрація маршрутів дозволяє маніпулювати потоками трафіку на прикордонному маршрутизаторі, фільтруючи оголошені маршрути до інших маршрутизаторів. Вона можлива з використанням різних політик маршрутів, списків доступу ACL і списків IP-префіксів.

Фільтрація за підсумовуванням. Найпростішим методом фільтрації маршрутів оголошення є використання опції not-advertise при налаштуванні підсумовування префіксів. Використання ключового слова not-advertise запобігає створенню будь-яких LSA типу 3 для будь-яких мереж в межах цього діапазону підсумовування. Це дозволяє маршрутам поширюватися лише в межах своєї області та не виходити в інші області через прикордонний маршрутизатор (ABR). Повна команда підсумовування виглядає наступним чином: area X range Y.Y.Y.Y Z.Z.Z.Z not-advertise.

Фільтрація за областю (Area Filtering). Фільтрування за допомогою підсумовування є простим, але не достатньо функціональним рішенням. Фільтрація за областю дає змогу краще налаштовувати та контролювати поширення маршрутів між областями. Фільтрація за областями OSPF налаштовується за допомогою команди area X filter-list prefix Y з останнім ключовим словом in для вхідної фільтрації або out для вихідної фільтрації. X – це номер області, а Y – ім'я префікса. Ім'я префікса налаштовується за допомогою команди ip prefix-list Y seq X після deny або permit, а вже потім префікс (наприклад, 192.168.1.0/24).

Загальні кроки для налаштування фільтрації за зонами для OSPFv3:

1. *Enable*;
2. *configure terminal*;
3. *router ospfv3 process-id*;

4. *area area-id filter-list prefix prefix-list-name {in | out}*;

5. *end*;

6. *ipv6 prefix-list list-name [seq seq-number] {deny ipv6-prefix/prefix-length | permit ipv6-prefix/prefix-length | description text} [ge ge-value] [le le-value]*.

Локальна фільтрація застосовується у випадках, коли маршрути потрібно фільтрувати локально на маршрутизаторі, перш ніж вони будуть додані до таблиці маршрутизації. Команда distribution list запобігає додаванню маршрутів. Список розподілу налаштовується командою distribute-list в режимі конфігурації процесу OSPF, а потім:

– номер ACL;

– ім'я ACL;

– префікс з ключовим словом prefix, а потім ім'я prefix-list;

– карта маршруту з ключовим словом route-map і ім'ям route-map.

Команда завершується фінальним ключовим словом in для вхідного або out для вихідного трафіку. Фільтрація за картою маршруту route-map має такі параметри:

– фільтрація на основі тегу маршруту, де користувачі можуть призначати мітки зовнішнім маршрутам, коли вони перерозподіляються в OSPF. Потім користувач може заборонити або дозволити ці маршрути в домені OSPF, ідентифікувавши цей тег у карті маршрутів і командами distribute-list in або distribute-list out;

– фільтрація на основі типу маршруту. У OSPF зовнішні маршрути можуть бути типу 1 або типу 2. Користувачі можуть створювати карти маршрутів, які відповідають типу 1 або 2, а потім використовувати команду distribute-list in для фільтрації певних префіксів. Крім того, карти маршрутів можуть ідентифікувати внутрішні маршрути (міжзонні та внутрішньозонні), а потім ці маршрути можуть бути відфільтровані;

– фільтрація на основі джерела маршруту застосовується тоді, коли виконується збіг за джерелом маршруту, джерело маршруту представляє Router ID ініціатора LSA, в якому оголошено префікс;

– фільтрація на основі інтерфейсу застосовується тоді, коли виконується збіг на вихідному інтерфейсі для маршруту, який OSPF намагається додати в таблицю маршрутизації;

– фільтрація на основі наступного переходу застосовується тоді, коли виконується збіг на наступному переході для маршруту, який OSPF намагається додати в таблицю маршрутизації.

Загальні кроки для налаштування фільтрації вхідних пакетів за допомогою команди route-map:

1. *Device(config)#router*;
2. *Device(config-router)#address-family ipv4 unicast*;
3. *Device(config-router-af)#distribute-list route-map rmap-name in*.

Параметри фільтрації маршруту за route-map: match interface, match ip address, match ip next-hop, match ip route-source, match metric, match route-type, match tag.

Налаштування фільтрації маршрутів по префіксу дозволяє виключити мережу 10.0.5.0/24 з обміну інформацією в мережі OSPF з метою зменшення кількості маршрутів, запобігання несанкціонованого доступу ресурсів в цій мережі або для зменшення непотрібного трафіку (рис. 3.16, рис. 3.17).

```
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#ip prefix-list FILTER deny 10.0.5.0/24
R1(config)#ip prefix-list FILTER permit 0.0.0.0/0 le 32
R1(config)#route-map FILTER-ROUTES permit 10
R1(config-route-map)#match ip address prefix-list FILTER
R1(config-route-map)#ex
R1(config)#router ospf 1
R1(config-router)#distribute-list route-map FILTER-ROUTES in
R1(config-router)#
R1(config-router)#end
R1#
```

Рисунок 3.16 – Налаштування фільтрації маршрутів по префіксу на R1

```
R1#show ip prefix-list FILTER
ip prefix-list FILTER: 2 entries
   seq 5 deny 10.0.5.0/24
   seq 10 permit 0.0.0.0/0 le 32
R1#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

     10.0.0.0/24 is subnetted, 6 subnets
C       10.0.10.0 is directly connected, FastEthernet0/0
O       10.0.2.0 [110/20] via 10.0.1.2, 00:00:34, FastEthernet0/1
C       10.0.3.0 is directly connected, FastEthernet1/0
C       10.0.1.0 is directly connected, FastEthernet0/1
O       10.0.4.0 [110/21] via 10.0.3.2, 00:00:34, FastEthernet1/0
                 [110/21] via 10.0.1.2, 00:00:34, FastEthernet0/1
O       10.0.20.0 [110/30] via 10.0.1.2, 00:00:34, FastEthernet0/1
R1#
```

Рисунок 3.17 – Перевірка фільтрації маршрутів по префіксу на R1

Підсумовування маршрутів допомагає зменшити трафік OSPF і зменшити кількість запусків алгоритму SPF для обчислення маршрутів. OSPF може підсумовувати маршрути тільки на ABR і ASBR. Загальна команда для підсумовування маршрутів в OSPF така: *(config-router) area AREA_ID range IP_ADDRESS MASK*. AREA_ID – це ідентифікатор зони, в якій підсумовуються маршрути, IP_ADDRESS MASK – це сумарний маршрут для оголошення в зоні.

Приклад запуску даної команди на маршрутизаторі (рис. 3.18, рис. 3.19):

1. *Router(config)#router ospf 1*;
2. *Router(config-router)#area 1 range 10.10.10.0 255.255.255.0*.

```
R1#
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#router ospf 1
R1(config-router)#area 0 range 40.0.0.0 255.0.0.0
R1(config-router)#ex
R1(config)#ex
R1#
%SYS-5-CONFIG_I: Configured from console by console

R1#write
Building configuration...
[OK]
R1#
```

Рисунок 3.18 – Налаштування підсумовування маршрутів на R1

```
R1#show ip route ospf
O    30.0.0.0 [110/128] via 10.0.0.1, 01:55:52, Serial0/3/1
              [110/128] via 20.0.0.2, 01:55:52, Serial0/3/0
     40.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
O       40.0.0.0 is a summary, 00:00:00, Null0
O    192.168.2.0 [110/65] via 20.0.0.2, 01:55:52, Serial0/3/0

R1#
```

Рисунок 3.19 – Перевірка підсумовування маршрутів на R1

Сегментна маршрутизація (Segment Routing, SR) дозволяє гнучко визначати наскрізні шляхи в топології IGP, кодуючи шляхи як послідовності топологічних субшляхів, які називаються сегментами. Загальні кроки для налаштування сегментації маршрутів OSPF:

1. *configure*;
2. *router ospf process-name*;
3. *segment-routing mpls*;
4. *area 0*;
5. *mpls traffic-eng area*;
6. *mpls traffic-eng router-id interface*;
7. *segment-routing mpls*;
8. *exit*;
9. *mpls traffic-eng*;
10. *commit*.

Другий крок вмикає маршрутизацію OSPF для вказаного процесу маршрутизації та переводить маршрутизатор у режим налаштування.

Третій крок дозволяє маршрутизацію сегментів з використанням даних масиву MPLS у процесі маршрутизації, а також для всіх областей та інтерфейсів у процесі маршрутизації. Він також вмикає маршрутизацію сегментів для пересилання на всіх інтерфейсах у процесі маршрутизації та встановлює SID, отримані OSPF, у таблицю пересилання.

Четвертий крок вмикає режим конфігурації зони.

П’ятий крок вмикає функціонал інженерії трафіку IGP.

Шостий крок, наприклад, може встановити інтерфейс зворотного loopback зв'язку для інженерії трафіку.

Сьомий крок вмикає маршрутизацію сегментів з використанням масиву даних MPLS в області та на всіх інтерфейсах в області. А ще він вмикає переадресацію маршрутизації сегментів на всіх інтерфейсах в області та встановлює SID, отримані OSPF, в таблиці переадресації.

Дев’ятий крок вмикає функціонал інженерії трафіку на вузлі. Вузол оголошує атрибути каналу інженерії трафіку в IGP, який заповнює базу даних інженерії трафіку (TED). Головний вузол інженерії трафіку вимагає, щоб TED обчислив і підтвердив шлях політики SR-TE.

Конфігурація пакетів для швидкої збіжності *Fast hello* виконується такими загальними кроками:

1. *Router> enable*;
2. *Router# configure terminal*;
3. *interface type number*, наприклад, *Router(config)# interface gigabitethernet 0/0/0*;
4. *ip ospf dead-interval minimal hello-multiplier multiplier*, наприклад, *Router(config-if)# ip ospf dead-interval minimal hello-multiplier 5*;
5. *end*;
6. *show ip ospf interface [interface-type interface-number]*, наприклад *Router# show ip ospf interface gigabitethernet 0/0/1*.

В четвертому кроці задається інтервал, протягом якого має бути отриманий хоч один пакет привітання, інакше сусід вважається неактивним.

В шостому кроці відображається інформація про інтерфейс, який бере участь в процесі OSPF.

Після налаштування на порті Ge0/0 довжини інтервалів dead-interval на 20 секунд та hello-interval на 2 секунди, значно збільшується швидкість збіжності мережі, оскільки це значно швидше за 40 та 10 секунд відповідно (рис. 3.20, рис. 3.21).

```
R1(config-if)#ip ospf hello-interval 2
R1(config-if)#ip ospf dead-interval 20
R1(config-if)#show ip ospf interface gigabitEthernet 0/0
```

Рисунок 3.20 – Налаштування довжини інтервалів на порті Ge0/0

```
R1#show ip ospf interface gigabitEthernet 0/0

GigabitEthernet0/0 is up, line protocol is up
  Internet address is 192.168.1.1/24, Area 0
  Process ID 1, Router ID 192.168.1.1, Network Type BROADCAST, Cost: 1
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 192.168.1.1, Interface address 192.168.1.1
  No backup designated router on this network
  Timer intervals configured, Hello 2, Dead 20, Wait 20, Retransmit 5
    Hello due in 00:00:00
  Index 1/1, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 1
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 0, Adjacent neighbor count is 0
  Suppress hello for 0 neighbor(s)
R1#
```

Рисунок 3.21 – Перевірка довжини інтервалів на порті Ge0/0

Для конфігурації *Bidirectional Forwarding Detection (BFD)* на одному інтерфейсі маршрутизатора виконуються наступні кроки:

1. *Router> enable*;
2. *Router# configure terminal*;
3. *interface vlan1*;
4. *ip ospf bfd*;
5. *bfd interval 50 min_rx 50 multiplier 3*;
6. *end*.

У третьому кроці треба вказати інтерфейс для налаштування.

У четвертому кроці вмикається BFD на інтерфейсі.

У п’ятому кроці вказуються параметри сеансу BFD, такі як мінімальний інтервал між пакетами та множник кількості сеансів BFD.

Для конфігурації BFD на всіх інтерфейсах маршрутизатора виконуються наступні кроки:

1. *Router> enable*;
2. *Router# configure terminal*;
3. *router ospf 100*;

4. *bfd all-interfaces*;
5. *exit*.

Третім кроком створюється процес OSPF на маршрутизаторі.

Четвертим кроком вмикається BFD на всіх інтерфейсах.

Перед конфігурацією BFD час виявлення недоступності сусіда становить стандартні 40 секунд (рис 3.22).

```
R1# show ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
10.0.1.2        1     FULL/BDR        00:00:39    10.0.1.2        FastEthernet0/1
```

Рис 3.22 – Час виявлення недоступності сусіда без BFD

Після конфігурації BFD на маршрутизаторі час виявлення недоступності сусіда буде становити близько 150 мілісекунд (рис. 3.23).

```
R1# show bfd neighbors detail

IPv4 Sessions
NeighAddr               LD/RD           RH/RS           State     Int
10.0.1.2                3/4             Up              Up        Fa0/1

Session state is UP and not using echo function.
Session Host: Software
OurAddr: 10.0.1.1
Handle: 1
Local Diag: 0, Demand mode: 0, Poll bit: 0
MinTxInt: 50000, MinRxInt: 50000, Multiplier: 3
Received MinRxInt: 50000, Received Multiplier: 3
Holddown (hits): 0(0), Hello (hits): 0(0)
Rx Count: 10
Tx Count: 10
Elapsed time watermarks: 0 0 (last: 0)
Registered protocols: OSPF CEF
Uptime: 00:00:10
Last packet: Version: 1               - Diagnostic: 0
             State bit: Up            - Demand bit: 0
             Poll bit: 0              - Final bit: 0
             C bit: 0                 - A bit: 0
             Multiplier: 3            - Length: 24
             My Discr.: 3             - Your Discr.: 4
             Min tx interval: 50000   - Min rx interval: 50000
             Min Echo interval: 0
```

Рис 3.23 – Час виявлення недоступності сусіда з BFD

В ситуаціях, коли основний маршрут недоступний *Loop-Free Alternate (LFA) Fast Reroute (FRR)* дозволяє швидко перемикати проходження трафіку на резервний маршрут за близько 50 мілісекунд. Процес конфігурації складається з таких кроків:

1. *Router(config)#router ospf 1*;
2. *Router(config-router)#fast-reroute X enable Y prefix-priority Z.*

Де замість X ставиться per-prefix або keep-all-paths. Per-prefix означає, що маршрутизатор підтримує LFA FRR тільки для окремих префіксів. Keep-all-paths означає, що протокол OSPF буде відстежувати усі маршрути, включаючи основний та резервний.

Замість Y ставиться enable area, що показує яка конкретна зона буде захищена LFA FRR.

Замість Z ставиться prefix-priority high або low, що показує які пріоритетні префікси буде захищати LFA FRR. При виборі високого пріоритету OSPF обробляє префікси loopback і /32 з вищим пріоритетом, обчислюючи LFA для них трохи раніше, ніж для інших префіксів. Коли виконується вибір низького пріоритету, OSPF обчислює LFA для всіх і будь-яких префіксів.

Приклад команди виглядає так:

– *R1(config-router)#fast-reroute per-prefix enable area 0 prefix-priority low.*

Перед запуском команди LFA FRR маршрутизатор R1 не має резервних маршрутів (рис 3.24).

```
R1# show ip route

Gateway of last resort is not set

     10.0.0.0/24 is subnetted, 5 subnets
C       10.0.10.0 is directly connected, FastEthernet0/0
O       10.0.1.0 [110/20] via 10.0.1.2, 00:12:48, FastEthernet0/1
C       10.0.3.0 is directly connected, FastEthernet1/0
C       10.0.1.0 is directly connected, FastEthernet0/1
O       10.0.2.0 [110/30] via 10.0.1.2, 00:12:48, FastEthernet0/1
O       10.0.4.0 [110/40] via 10.0.3.2, 00:12:48, FastEthernet1/0
O       10.0.5.0 [110/50] via 10.0.3.2, 00:12:48, FastEthernet1/0
```

Рисунок 3.24 – Таблиця маршрутизації до запуску LFA FRR

Після запуску команди LFA FRR на маршрутизаторі R1 з’являється резервний маршрут, а отже підвищується відмовостійкість маршрутизації в мережі (рис 3.25, рис. 3.26).

```
R1# show ip route

Gateway of last resort is not set

     10.0.0.0/24 is subnetted, 5 subnets
C       10.0.10.0 is directly connected, FastEthernet0/0
O       10.0.1.0 [110/20] via 10.0.1.2, 00:12:48, FastEthernet0/1
C       10.0.3.0 is directly connected, FastEthernet1/0
C       10.0.1.0 is directly connected, FastEthernet0/1
O       10.0.2.0 [110/30] via 10.0.1.2, 00:12:48, FastEthernet0/1
O       10.0.4.0 [110/40] via 10.0.3.2, 00:12:48, FastEthernet1/0
O       10.0.5.0 [110/50] via 10.0.3.2, 00:12:48, FastEthernet1/0
  (FRR active, backup via 10.0.3.2)
```

Рисунок 3.25 – Таблиця маршрутизації після запуску LFA FRR

```
R1# show ip ospf fast-reroute

OSPF Fast Reroute per-prefix:
  Area 0:
    Prefix 10.0.2.0/24
      Priority: low
      Primary path: via 10.0.1.2, metric 30
      Backup path: via 10.0.3.2, metric 50
```

Рисунок 3.26 – Перевірка резервного маршруту

Виконання *інкрементного SPF* є ефективнішим, ніж повне виконання алгоритму SPF, що дозволяє OSPF швидше досягати збіжності з новою топологією маршрутизації у відповідь на будь-яку зміну топології мережі.

Процес увімкнення інкрементного SPF на маршрутизаторі описується такими пунктами:

1. *Router> enable*;
2. *Router# configure terminal*;
3. *Router(config)# router ospf 1*;

4. *Router(config-router)# ispf*;
5. *Router(config-router)# end*.

Без використання інкрементного SPF повний перерахунок алгоритму SPF займає 4 мс (рис. 3.27).

```
SPF 5 executed 00:00:35 ago, SPF type Full
  SPF calculation time (in msec):
  SPT    Intra  D-Intr Summ   D-Summ Ext7   D-Ext7 Total
  2      4      0      0      0      0      0      4
  RIB manipulation time (in msec):
  RIB Update    RIB Delete
  0             0
  LSIDs processed R:5 N:5 Stub:2 SN:0 SA:0 X7:0
  Change record
  LSIDs changed 9
  Changed LSAs. Recorded is LS ID and LS type:
  10.0.5.1(R) 10.0.20.1(R) 10.0.5.2(R) 10.0.10.1(R)
  10.0.10.1(N) 10.0.5.2(N) 10.0.4.1(N) 10.0.5.1(R)
  10.0.20.1(R)
```

Рисунок 3.27 – Повний перерахунок алгоритму SPF

Після увімкнення інкрементного SPF маршрутизатор виконав додатковий перерахунок алгоритму SPF, що зайняв менше часу, а саме – 2 мс (рис. 3.28).

```
SPF 16 executed 00:00:16 ago, SPF type Incremental
  SPF calculation time (in msec):
  SPT    Intra  D-Intr Summ   D-Summ Ext7   D-Ext7 Total
  1      2      0      0      0      0      0      2
  RIB manipulation time (in msec):
  RIB Update    RIB Delete
  0             0
  LSIDs processed R:2 N:2 Stub:1 SN:0 SA:0 X7:0
  Change record
  LSIDs changed 3
  Changed LSAs. Recorded is LS ID and LS type:
  10.0.2.1(R) 10.0.2.1(N) 10.0.20.1(R)
```

Рисунок 3.28 – Додатковий перерахунок алгоритму SPF

Для конфігурування *інтерфейсу Loopback* потрібно виконати наступні кроки(рис. 3.29, рис. 3.30):

1. *Router> enable*;
2. *Router# configure terminal*;
3. *interface loopback number*, де вказується номер інтерфейсу, який підлягає створенню та конфігурації, наприклад, *Router(config)# interface loopback 0*;

4. *ip address ip-address mask [secondary]*, де вказується IP-адреса для увімкненого Loopback інтерфейсу, наприклад, *Router(config-if)# ip address 10.20.1.2 255.255.255.0*;

5. *Router(config-if)# end*;

6. *show interfaces loopback number*, де вказується інформація про конкретний Loopback інтерфейс, наприклад, *Router# show interfaces loopback 0*;

7. *exit*.

```
R1(config)#interface loopback 0

R1(config-if)#
%LINK-5-CHANGED: Interface Loopback0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, changed state to up

R1(config-if)#ip add 10.0.0.5 255.255.255.255
% 10.0.0.5 overlaps with Serial0/3/1
R1(config-if)#ip add 40.0.0.1 255.255.255.255
R1(config-if)#ex
```

Рисунок 3.29 – Конфігурація Loopback інтерфейсу на маршрутизаторі R1

```
R1#
R1#sh ip interface brief | exclude unassigned
Interface              IP-Address      OK? Method Status                Protocol
GigabitEthernet0/0     192.168.1.1     YES manual up                    up
Serial0/3/0            20.0.0.1        YES manual up                    up
Serial0/3/1            10.0.0.2        YES manual up                    up
Loopback0              40.0.0.1        YES manual up                    up
R1#
```

Рисунок 3.30 – Перевірка Loopback інтерфейсу на маршрутизаторі R1

Задача забезпечення автентифікації пакетів OSPFv3 покладена на *стек протоколів IPsec*. Спеціалізовані програми для захисту трафіку застосовують API IPsec для прослуховування та конфігурації безпечних каналів передачі. OSPFv3 вимагає наявність заголовка автентифікації IPv6 AH або заголовку IPv6 ESP для забезпечення цілісності, автентифікації та конфіденційності обміну маршрутами. IPsec можна налаштовувати на кожному інтерфейсі або в межах усієї зони. Щоб налаштувати IPsec, проводиться налаштування політики безпеки, яка є комбінацією індексу політики безпеки (SPI) і ключа (ключ використовується для створення та перевірки хеш-сум).

Налаштування автентифікації пакетів OSPFv3 на одному інтерфейсі виконується наступними кроками:

1. *Device> enable*;
2. *Device# configure terminal*;
3. *interface type number*, де вказується тип та номер інтерфейсу;
4. далі прописується одна з команд:
   - *ospfv3 authentication {ipsec spi} {md5 | sha1}{ key-encryption-type key} | null*, наприклад, *Device(config-if)# ospfv3 authentication md5 0 27576134094768132473302031209727*;
   - *ipv6 ospf authentication {null | ipsec spi spi authentication-algorithm [key-encryption-type] [key]}*, наприклад, *Device(config-if)# ipv6 ospf authentication ipsec spi 500 md5 1234567890abcdef1234567890abcdef*.

Налаштування автентифікації пакетів OSPFv3 у всій зоні виконується наступними кроками:

1. *Device> enable*;
2. *Device# configure terminal*;
3. *ipv6 router ospf process-id*, наприклад, *Device(config)# ipv6 router ospf 1*;
4. *area area-id authentication ipsec spi spi authentication-algorithm [key-encryption-type] key*, наприклад, *Device(config-rtr)# area 1 authentication ipsec spi 678 md5 1234567890ABCDEF1234567890ABCDEF*.

3.3 Комбінація методів для оптимізації протоколу OSPF

Враховуючи проведені дослідження існуючих методів оптимізації протоколу OSPF та зазначену в минулих розділах інформацію буде доречно розробити методику оптимізації маршрутизації протоколом OSPF у великих мережах. Ця методика полягає у порівнянні і виборі найбільш ефективних методів оптимізації з

однаковими задачами та їх комбінація для одночасного використання в мережах з OSPF.

Для задач підвищення масштабованості доречно одразу використовувати ієрархічний дизайн мережі OSPF при проєктуванні та конфігурації конкретної інформаційної мережі для зменшення навантаження на апаратне забезпечення маршрутизаторів і скорочення часу формування LSDB, ніж потім мати проблеми з масштабуванням та з безперебійним передаванням трафіку. Також ефективними будуть підсумовування маршрутів в зонах автономних систем для зменшення об'ємів маршрутних таблиць, використання віртуальних каналів, фільтрація маршрутів для обмеження обсягу маршрутної інформації, яка поширюється в мережі.

Щоб зменшити затримки в доступі до інформаційних каналів та інтерфейсів в мережі краще використовувати Bidirectional Forwarding Detection (BFD) замість пакетів Fast hello, оскільки Fast hello генерує занадто багато hello пакетів, що підвищує навантаження на маршрутизатори та канали зв'язку. В той час як Fast hello залежить від інтервалів Hello пакетів, BFD значно швидше виявляє відмови на рівні інтерфейсів. Зменшенню затримок також сприяє застосування інкрементного SPF для пришвидшення збіжності після змін топології та використання Loop-Free Alternate (LFA) Fast Reroute (FRR) для швидкого перемикання проходження трафіку на резервний маршрут.

Всі ці методи також зменшують використання апаратних ресурсів маршрутизаторів та пропускної спроможності каналів.

Щоб підвищити параметр безпеки для трафіку, що передається мережею OSPF доречно реалізувати автентифікацію маршрутизаторів в мережі та шифрування трафіку за допомогою протоколу IPsec.

ВИСНОВКИ

В ході виконання даної кваліфікаційної роботи були опрацьовані питання дослідження та аналізу існуючих методів оптимізації протоколу OSPF для підтримки динамічної маршрутизації, були проаналізовані різні дослідження та статті на тему оптимізації протоколу OSPF. На основі цього аналізу були надані висновки ефективності методів та доцільності їх використання у сучасних інформаційних мережах. Були виявлені та опрацьовані проблеми та недоліки динамічної маршрутизації за допомогою протоколу OSPF. Була наведена методика комбінації існуючих методів оптимізації протоколу OSPF для підтримки динамічної маршрутизації.

Основною метою кваліфікаційної роботи було розв’язання проблемного питання оптимізації протоколу OSPF для підтримки динамічної маршрутизації. Після аналізу великої кількості загальнодоступних опублікованих досліджень, статей та публікацій від різних авторів було досягнуто остаточного висновку щодо питання пошуку методів оптимізації протоколу OSPF для підтримки динамічної маршрутизації у великих мережах. Розв’язанням цього питання виявилась комбінація певних методів з різними аспектами щодо оптимізації протоколу OSPF. В кінці було наведено список методів оптимізації протоколу OSPF для підтримки динамічної маршрутизації у великих мережах, які є найбільш слушними та ефективними.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Makarenko S. THE IMPROVED OSPF PROTOCOL FOR HIGH NETWORK STABILITY. Proceedings of Telecommunication Universities. 2018. Vol. 4, no. 2. P. 82–90. URL: https://doi.org/10.31854/1813-324x-2018-2-82-90 (дата звернення: 06.05.2024).

2. IMPROVED ALGORITHM FOR THE PACKET ROUTING IN TELECOMMUNICATION NETWORKS / R. I. Liskevych et al. Ukrainian Journal of Information Technology. 2021. Vol. 3, no. 1. P. 114–119. URL: https://doi.org/10.23939/ujit2021.03.114 (дата звернення: 06.05.2024).

3. Harris R. Strategies for Optimizing OSPF Convergence – This Bridge is the Root. This Bridge is the Root. URL: https://thisbridgeistheroot.com/blog/strategies-for-optimizing-ospf-convergence (дата звернення: 06.05.2024).

4. Improving Convergence Speed and Scalability in OSPF: A Survey / M. Goyal et al. IEEE Communications Surveys & Tutorials. 2012. Vol. 14, no. 2. P. 443–463. URL: https://doi.org/10.1109/surv.2011.011411.00065 (дата звернення: 06.05.2024).

5. Contributors to Wikimedia projects. Routing – Wikipedia. Wikipedia, the free encyclopedia. URL: https://en.wikipedia.org/wiki/Routing (дата звернення: 06.05.2024).

6. Протоколи маршрутизації. StudFiles. URL: https://studfile.net/preview/7669189/page:44/ (дата звернення: 06.05.2024).

7. Difference between Static and Dynamic Routing – GeeksforGeeks. GeeksforGeeks. URL: https://www.geeksforgeeks.org/difference-between-static-and-dynamic-routing/?ref=lbp (дата звернення: 06.05.2024).

8. Types of Routing Protocols (3.1.4) > Cisco Networking Academy's Introduction to Routing Dynamically | Cisco Press. Cisco Press: Source for Cisco Technology, CCNA, CCNP, CCIE Self-Study | Cisco Press. URL: https://www.ciscopress.com/articles/article.asp?p=2180210&seqNum=7 (дата звернення: 06.05.2024).

9. Олещенко Л. ОРГАНІЗАЦІЯ КОМП'ЮТЕРНИХ МЕРЕЖ. DSpace :: ELAKPI :: Репозиторій КПІ ім. Ігоря Сікорського. URL: https://ela.kpi.ua/server/api/core/bitstreams/245d8d52-1715-4f2e-9e6a-e4bd17db7d4d/content (дата звернення: 06.05.2024).

10. Contributors to Wikimedia projects. Open Shortest Path First – Wikipedia. Wikipedia, the free encyclopedia. URL: https://en.wikipedia.org/wiki/Open_Shortest_Path_First (дата звернення: 06.05.2024).

11. OSPF: “34 Things to remember”. IT Tips for Systems and Network Administrators. URL: https://skminhaj.wordpress.com/2016/02/15/ospf-34-things-to-remember/ (дата звернення: 06.05.2024).

12. Open Shortest Path First Protocol | CQR. CQR. URL: https://cqr.company/ua/wiki/protocols/open-shortest-path-first-protocol/ (дата звернення: 06.05.2024).

13. Протокол маршрутизації OSPF. Державний університет "Житомирська політехніка" - Освітній портал. URL: https://learn.ztu.edu.ua/mod/resource/view.php?id=146734 (дата звернення: 06.05.2024).

14. Understand Open Shortest Path First (OSPF) – Design Guide. Cisco. URL: https://www.cisco.com/c/en/us/support/docs/ip/open-shortest-path-first-ospf/7039-1.html (дата звернення: 06.05.2024).

15. Томас Т. М. Структура та реалізація мереж на основі протоколу OSPF : навч. посіб. / пер. з англ. К. А. Птиціна. 2-ге вид. Київ : Вільямс, 2004. 816 с.

16. Hierarchical Network Design Overview (1.1) > Cisco Networking Academy Connecting Networks Companion Guide: Hierarchical Network Design | Cisco Press. Cisco Press: Source for Cisco Technology, CCNA, CCNP, CCIE Self-Study | Cisco Press. URL: https://www.ciscopress.com/articles/article.asp?p=2202410&seqNum=4 (дата звернення: 06.05.2024).

17. IP Routing: OSPF Configuration Guide – OSPFv3 Route Filtering Using Distribute-List [Cisco Cloud Services Router 1000V Series]. Cisco. URL: https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/iproute_ospf/configuration/xe-16/iro-xe-16-book/OSPFv3-Route-Filter-Using-Dist-List.html (дата звернення: 06.05.2024).

18. Configuring OSPF Route Summarization | Free CCNA Workbook. Free CCNA Workbook |. URL: https://www.freeccnaworkbook.com/workbooks/ccna/configuring-ospf-route-summarization (дата звернення: 06.05.2024).

19. Configure Segment Routing for OSPF Protocol. Cisco: Software, Network, and Cybersecurity Solutions – Cisco. URL: http://www.cisco.com/c/en/us/td/docs/routers/ncs6000/software/segment-routing/configuration/guide/b-segment-routing-cg-ncs6k/b-segment-routing-cg-ncs6k_chapter_0101.pdf (дата звернення: 06.05.2024).

20. IP Routing: OSPF Configuration Guide – OSPF Support for Fast Hello Packets [Cisco Cloud Services Router 1000V Series]. *Cisco*. URL: https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/iproute_ospf/configuration/xe-16/iro-xe-16-book/iro-fast-hello.html (дата звернення: 06.05.2024).

21. Configuring Bidirectional Forwarding Detection. Cisco: Software, Network, and Cybersecurity Solutions – Cisco. URL: https://www.cisco.com/c/en/us/td/docs/wireless/asr_901/Configuration/Guide/b_asr901-scg/b_asr901-scg_chapter_010110.pdf (дата звернення: 06.05.2024).

22. IP Routing: OSPF Configuration Guide - OSPFv2 Loop-Free Alternate Fast Reroute [Cisco Cloud Services Router 1000V Series]. Cisco. URL: https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/iproute_ospf/configuration/xe-16/iro-xe-16-book/iro-lfa-frr.html (дата звернення: 06.05.2024).

23. IP Routing: OSPF Configuration Guide – OSPF Incremental SPF [Cisco Cloud Services Router 1000V Series]. Cisco. URL: https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/iproute_ospf/configuration/xe-16/iro-xe-16-book/iro-incre-spf.html (дата звернення: 06.05.2024).

24. Interface and Hardware Component Configuration Guide, Cisco IOS XE Gibraltar 16.12.x - Configuring Virtual Interfaces [Cisco ASR 1000 Series Aggregation Services Routers]. *Cisco*. URL: https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/interface/configuration/xe-16-12/ir-xe-16-12-book/ir-cfg-vir-if-xe.html (дата звернення: 06.05.2024).

25. IP Routing: OSPF Configuration Guide - IPv6 Routing: OSPFv3 Authentication Support with IPsec [Cisco Cloud Services Router 1000V Series]. Cisco. URL: https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/iproute_ospf/configuration/xe-16/iro-xe-16-book/ip6-route-ospfv3-auth-ipsec.html (дата звернення: 06.05.2024).

26. Open shortest path first (OSPF) router roles and configuration – GeeksforGeeks. GeeksforGeeks. URL: https://www.geeksforgeeks.org/open-shortest-path-first-ospf-router-roles-configuration/ (дата звернення: 06.05.2024).

27. Optimizing OSPF Behavior > OSPF Implementation | Cisco Press. Cisco Press: Source for Cisco Technology, CCNA, CCNP, CCIE Self-Study | Cisco Press. URL: https://www.ciscopress.com/articles/article.asp?p=2294214&seqNum=3 (дата звернення: 06.05.2024).

28. ComputerNetworkingNotes. How to configure OSPF Routing Protocol. *ComputerNetworkingNotes*.
URL: https://www.computernetworkingnotes.com/ccna-study-guide/ospf-configuration-step-by-step-guide.html (дата звернення: 06.05.2024).

29. Налаштування роботи протоколу маршрутизації OSPF. *Інформаційний портал Технічного фахового коледжу*. URL: https://e-tk.lntu.edu.ua/pluginfile.php/19445/mod_resource/content/0/Практична%20робота%2011.%20Налаштування%20роботи%20протоколу%20маршрутизації%20OSPF.pdf (дата звернення: 06.05.2024).

30. IP Routing: OSPF Configuration Guide – Configuring OSPF [Cisco Cloud Services Router 1000V Series]. Cisco. URL: https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/iproute_ospf/configuration/xe-16/iro-xe-16-book/iro-cfg.html (дата звернення: 06.05.2024).

31. Токар М. В. Сучасні методи оптимізації протоколу OSPF. *Наука сьогодення: від досліджень до стратегічних рішень* : VI Міжнар. студент. наук. конф., м. Чернігів, 10 трав. 2024 р. С. 191–193.

32. Токар М. В. Вдосконалення протоколу OSPF для підтримки безпечної маршрутизації. *Застосування програмного забезпечення в інформаційно-комунікаційних технологіях* : Всеукр. науково-техн. конф., м. Київ, 24 квіт. 2024 р. С. 239–241.

Додаток А

ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ

(Презентація)

ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
Навчально-науковий інститут Інформаційних технологій
Кафедра Комп'ютерної інженерії

КВАЛІФІКАЦІЙНА РОБОТА
НА ЗДОБУТТЯ ОСВІТНЬОГО СТУПЕНЯ БАКАЛАВРА

на тему: «Дослідження методів оптимізації протоколу OSPF для підтримки динамічної маршрутизації»

Виконав студент групи КІД-42
Токар Михайло Віталійович

Керівник кваліфікаційної роботи:
Бученко Ігор Анатолійович,
старший викладач кафедри КІ

Київ – 2024

Мета роботи – дослідження методів оптимізації протоколу OSPF для підтримки динамічної маршрутизації на поточному етапі розвитку технологій.

Об'єкт дослідження – процес оптимізації протоколу OSPF для підтримки динамічної маршрутизації в комп'ютерних інформаційних мережах.

Предмет дослідження – методи оптимізації протоколу OSPF для підтримки динамічної маршрутизації.

2

Актуальність обраної теми

В сучасних комп'ютерних мережах на цей момент використовується чимала кількість методів оптимізації протоколу OSPF для підтримки динамічної маршрутизації, що залучені в багатьох різних за призначенням мережах. Використання методів оптимізації протоколу OSPF для підтримки динамічної маршрутизації наразі й в подальшому розвитку мереж дозволяють покращити пропускну спроможність мереж організацій і підприємств, що використовують дані методи. Використання методів оптимізації протоколу OSPF для підтримки динамічної маршрутизації значно збільшує ефективність використання обчислювальних ресурсів обладнання, що також дозволяє значно зменшити затримки в мережі.

Продовження актуальності обраної теми

Ці методи оптимізації протоколу OSPF мають велику кількість переваг, проте, також, і ряд недоліків, що вимагають від користувачів достатньої кваліфікованості для користування ними.

На основі проведеного дослідження було виявлено та проаналізовано чималу кількість рішень для оптимізації протоколу OSPF, що дозволяє зробити точні висновки про використання методів оптимізації протоколу OSPF для підтримки динамічної маршрутизації.

4

Використання протоколу OSPF

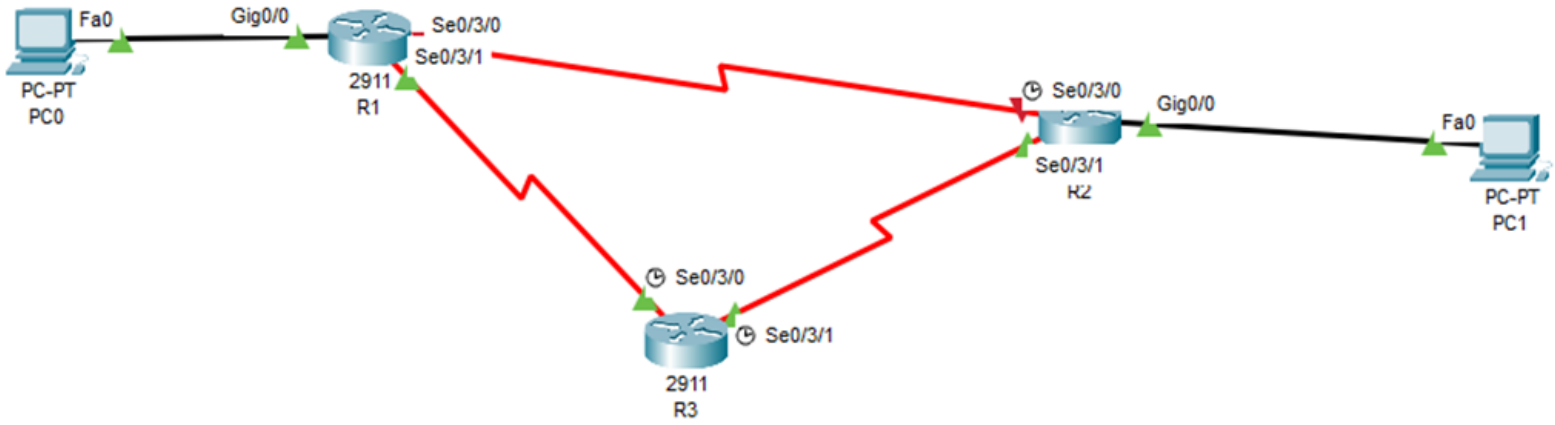
Динамічна маршрутизація забезпечує автоматичне визначення шляху передачі даних в комп'ютерних мережах з використанням протоколів, які дозволяють маршрутизаторам самим обмінюватися між собою інформацією про стан мережі без втручання системних адміністраторів.

OSPF – це динамічний протокол маршрутизації трафіку в межах одної автономної системи. Він обчислює дерево найкоротших шляхів SPT за допомогою алгоритму Дейкстри для визначення оптимального маршруту до пункту призначення. OSPF виявляє зміни в топології, такі як обриви зв'язку, і за лічені секунди знаходить нову топологію для маршрутизації трафіку.

5

Принцип роботи протоколу OSPF

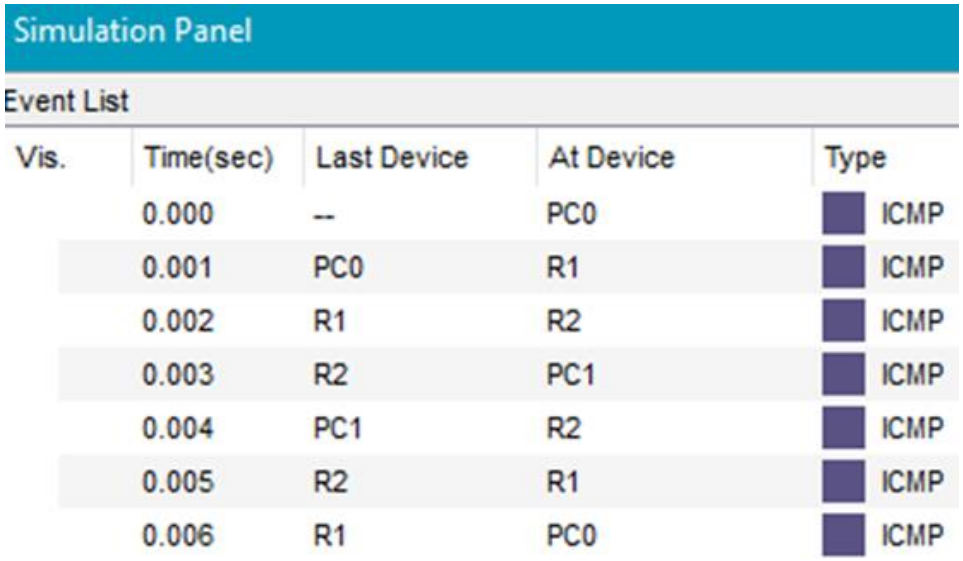
В процесі роботи протокол OSPF автоматично визначає доступний найшвидший маршрут для передачі пакетів до пункту призначення.



6

Принцип роботи протоколу OSPF

У випадку, коли доступні маршрути з однаковими каналами та інтерфейсами, протокол OSPF обирає маршрут з найменшою кількістю переходів.



Simulation Panel

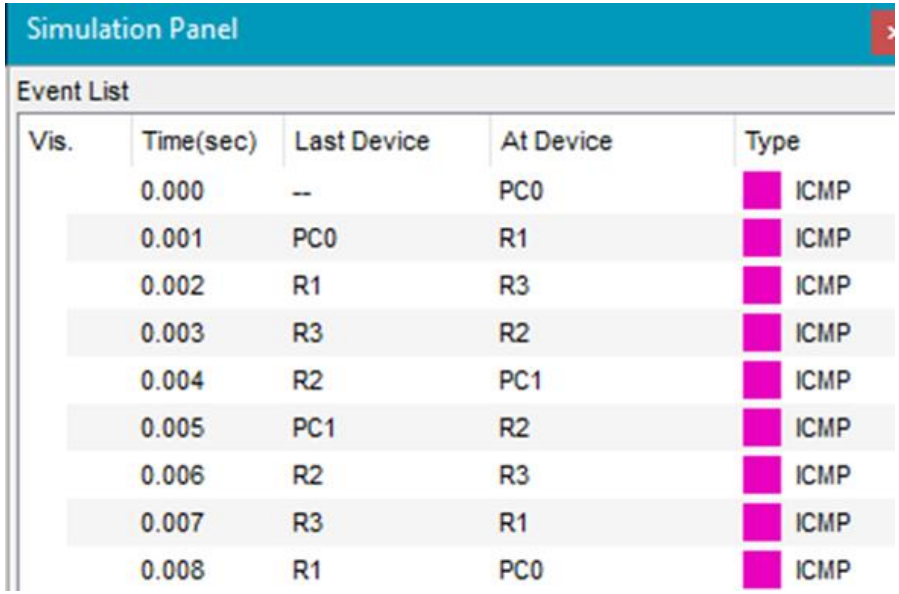
Event List

Vis.	Time(sec)	Last Device	At Device	Type
	0.000	--	PC0	ICMP
	0.001	PC0	R1	ICMP
	0.002	R1	R2	ICMP
	0.003	R2	PC1	ICMP
	0.004	PC1	R2	ICMP
	0.005	R2	R1	ICMP
	0.006	R1	PC0	ICMP

7

Принцип роботи протоколу OSPF

Якщо найшвидший маршрут стає недоступним за деяких причин, то протокол OSPF автоматично перемикає передачу пакетів на інший доступний маршрут, хоч з більшою кількістю переходів, меншою швидкістю та більшими затримками.



8

Потреби оптимізації протоколу OSPF:

- полегшення масштабованості при збільшенні розміру мережі;
- зменшення кількості трафіку маршрутної інформації;
- зменшення завантаженості каналів передачі інформації;
- зменшення завантаженості апаратного забезпечення маршрутизаторів;
- запобігання постійних запусків алгоритму SPF для повного перерахунку таблиць маршрутизації;
- потреба в зменшенні затримки передачі пакетів у великих мережах;
- зменшення періоду часу для досягнення збіжності після будь-яких змін в топології мережі;
- організація механізму для шифрування маршрутної інформації і автентифікації маршрутизаторів учасників мережі.

9

Методи оптимізації протоколу OSPF

Для вирішення проблеми масштабованості в мережі потрібно використовувати ієрархічний дизайн мережі OSPF (Hierarchical Network Design), фільтрацію маршрутів (Route filtering), підсумовування маршрутів (Route Summarization) та сегментну маршрутизацію (Segment Routing).

На даних рисунках показано процес налаштування підсумовування маршрутів. Завдяки підсумовуванню видно, що кількість записів IP-адрес в маршрутній таблиці роутера R1 зменшилася.

```
R1#
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#router ospf 1
R1(config-router)#area 0 range 40.0.0.0 255.0.0.0
R1(config-router)#ex
R1(config)#ex
R1#
%SYS-5-CONFIG_I: Configured from console by console

R1#write
Building configuration...
[OK]
R1#
```

```
R1#show ip route ospf
O    30.0.0.0 [110/128] via 10.0.0.1, 01:55:52, Serial0/3/1
              [110/128] via 20.0.0.2, 01:55:52, Serial0/3/0
     40.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
O       40.0.0.0 is a summary, 00:00:00, Null0
O    192.168.2.0 [110/65] via 20.0.0.2, 01:55:52, Serial0/3/0

R1#
```

10

Методи оптимізації протоколу OSPF

Зображений на даних рисунках процес налаштування фільтрації маршрутів по префіксу на роутері R1 дозволяє виключити мережу 10.0.5.0/24 з обміну інформацією в мережі OSPF.

```
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#ip prefix-list FILTER deny 10.0.5.0/24
R1(config)#ip prefix-list FILTER permit 0.0.0.0/0 le 32
R1(config)#route-map FILTER-ROUTES permit 10
R1(config-route-map)#match ip address prefix-list FILTER
R1(config-route-map)#ex
R1(config)#router ospf 1
R1(config-router)#distribute-list route-map FILTER-ROUTES in
R1(config-router)#
R1(config-router)#end
R1#
```

```
R1#show ip prefix-list FILTER
ip prefix-list FILTER: 2 entries
   seq 5 deny 10.0.5.0/24
   seq 10 permit 0.0.0.0/0 le 32
R1#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

     10.0.0.0/24 is subnetted, 6 subnets
C       10.0.10.0 is directly connected, FastEthernet0/0
O       10.0.2.0 [110/20] via 10.0.1.2, 00:00:34, FastEthernet0/1
C       10.0.3.0 is directly connected, FastEthernet1/0
C       10.0.1.0 is directly connected, FastEthernet0/1
O       10.0.4.0 [110/21] via 10.0.3.2, 00:00:34, FastEthernet1/0
                 [110/21] via 10.0.1.2, 00:00:34, FastEthernet0/1
O       10.0.20.0 [110/30] via 10.0.1.2, 00:00:34, FastEthernet0/1
R1#
```

Методи оптимізації протоколу OSPF

Для розв'язання проблеми підвищеної затримки каналів зв'язку та проблеми доступності потрібно використовувати методи пришвидшеної конвергенції, такі як налаштування Dead та Hello таймерів, пакети Fast Hello, виявлення двонаправленої переадресації (BFD), Loop-Free Alternates (LFAs), Loopback інтерфейси та інкрементний SPF (iSPF).

Зменшення Dead та Hello таймерів дозволяє значно знизити час збіжності мережі після змін у топології.

```
R1(config-if)#ip ospf hello-interval 2
R1(config-if)#ip ospf dead-interval 20
R1(config-if)#show ip ospf interface gigabitEthernet 0/0
```

```
R1#show ip ospf interface gigabitEthernet 0/0

GigabitEthernet0/0 is up, line protocol is up
  Internet address is 192.168.1.1/24, Area 0
  Process ID 1, Router ID 192.168.1.1, Network Type BROADCAST, Cost: 1
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 192.168.1.1, Interface address 192.168.1.1
  No backup designated router on this network
  Timer intervals configured, Hello 2, Dead 20, Wait 20, Retransmit 5
    Hello due in 00:00:00
  Index 1/1, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 1
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 0, Adjacent neighbor count is 0
  Suppress hello for 0 neighbor(s)
R1#
```

Методи оптимізації протоколу OSPF

Застосування BFD дозволяє знизити час виявлення недоступності сусіда до близько 150 мілісекунд в порівнянні зі стандартними 40 секунд.

```
R1# show ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
10.0.1.2        1     FULL/BDR        00:00:39    10.0.1.2        FastEthernet0/1
```

```
R1# show bfd neighbors detail

IPv4 Sessions
NeighAddr                LD/RD            RH/RS           State     Int
10.0.1.2                 3/4              Up              Up        Fa0/1

Session state is UP and not using echo function.
Session Host: Software
OurAddr: 10.0.1.1
Handle: 1
Local Diag: 0, Demand mode: 0, Poll bit: 0
MinTxInt: 50000, MinRxInt: 50000, Multiplier: 3
Received MinRxInt: 50000, Received Multiplier: 3
Holddown (hits): 0(0), Hello (hits): 0(0)
Rx Count: 10
Tx Count: 10
Elapsed time watermarks: 0 0 (last: 0)
Registered protocols: OSPF CEF
Uptime: 00:00:10
Last packet: Version: 1                - Diagnostic: 0
             State bit: Up             - Demand bit: 0
             Poll bit: 0               - Final bit: 0
             C bit: 0                  - A bit: 0
             Multiplier: 3             - Length: 24
             My Discr.: 3              - Your Discr.: 4
             Min tx interval: 50000    - Min rx interval: 50000
             Min Echo interval: 0
```

Методи оптимізації протоколу OSPF

Завдяки застосуванню LFA FRR маршрутизатори можуть швидко за близько 50 мілісекунд перекинути передачу пакетів на інший або резервний маршрут.

```
R1# show ip route

Gateway of last resort is not set

     10.0.0.0/24 is subnetted, 5 subnets
C       10.0.10.0 is directly connected, FastEthernet0/0
O       10.0.1.0 [110/20] via 10.0.1.2, 00:12:48, FastEthernet0/1
C       10.0.3.0 is directly connected, FastEthernet1/0
C       10.0.1.0 is directly connected, FastEthernet0/1
O       10.0.2.0 [110/30] via 10.0.1.2, 00:12:48, FastEthernet0/1
O       10.0.4.0 [110/40] via 10.0.3.2, 00:12:48, FastEthernet1/0
O       10.0.5.0 [110/50] via 10.0.3.2, 00:12:48, FastEthernet1/0
  (FRR active, backup via 10.0.3.2)
```

```
R1# show ip ospf fast-reroute

OSPF Fast Reroute per-prefix:
  Area 0:
    Prefix 10.0.2.0/24
      Priority: low
      Primary path: via 10.0.1.2, metric 30
      Backup path: via 10.0.3.2, metric 50
```

Методи оптимізації протоколу OSPF

Час виконання перерахунку повного алгоритму SPF можна знизити завдяки використанню інкрементного SPF. На даних зображеннях видно, що виконання повного SPF займає 4 мілісекунди, коли виконання часткового SPF займає лише 2 мілісекунди.

```
SPF 5 executed 00:00:35 ago, SPF type Full
  SPF calculation time (in msec):
  SPT    Intra  D-Intr Summ   D-Summ Ext7   D-Ext7 Total
  2      4      0      0      0      0      0      4
  RIB manipulation time (in msec):
  RIB Update    RIB Delete
  0             0
  LSIDs processed R:5 N:5 Stub:2 SN:0 SA:0 X7:0
  Change record
  LSIDs changed 9
  Changed LSAs. Recorded is LS ID and LS type:
  10.0.5.1(R) 10.0.20.1(R) 10.0.5.2(R) 10.0.10.1(R)
  10.0.10.1(N) 10.0.5.2(N) 10.0.4.1(N) 10.0.5.1(R)
  10.0.20.1(R)
```

```
SPF 16 executed 00:00:16 ago, SPF type Incremental
  SPF calculation time (in msec):
  SPT    Intra  D-Intr Summ   D-Summ Ext7   D-Ext7 Total
  1      2      0      0      0      0      0      2
  RIB manipulation time (in msec):
  RIB Update    RIB Delete
  0             0
  LSIDs processed R:2 N:2 Stub:1 SN:0 SA:0 X7:0
  Change record
  LSIDs changed 3
  Changed LSAs. Recorded is LS ID and LS type:
  10.0.2.1(R) 10.0.2.1(N) 10.0.20.1(R)
```

Методи оптимізації протоколу OSPF

Для підвищення кібербезпеки маршрутизації трафіку протоколом OSPF потрібно використовувати механізми шифрування та автентифікації за допомогою протоколу IPsec.

16

ВИСНОВКИ

В ході виконання даної кваліфікаційної роботи були опрацьовані питання дослідження та аналізу існуючих методів оптимізації протоколу OSPF для підтримки динамічної маршрутизації, були проаналізовані різні дослідження та статті на тему оптимізації протоколу OSPF. На основі цього аналізу були надані висновки ефективності методів та доцільності їх використання у сучасних інформаційних мережах. Були виявлені та опрацьовані проблеми та недоліки динамічної маршрутизації за допомогою протоколу OSPF. Була наведена методика комбінації існуючих методів оптимізації протоколу OSPF для підтримки динамічної маршрутизації.

ПРОДОВЖЕННЯ ВИСНОВКІВ

Основною метою кваліфікаційної роботи було розв’язання проблемного питання оптимізації протоколу OSPF для підтримки динамічної маршрутизації. Після аналізу великої кількості загальнодоступних опублікованих досліджень, статей та публікацій від різних авторів було досягнуто остаточного висновку щодо питання пошуку методів оптимізації протоколу OSPF для підтримки динамічної маршрутизації у великих мережах. Розв’язанням цього питання виявилась комбінація певних методів з різними аспектами щодо оптимізації протоколу OSPF. В кінці було наведено список методів оптимізації протоколу OSPF для підтримки динамічної маршрутизації у великих мережах, які є найбільш слушними та ефективними.

Результати впровадження

1. Токар М. В. Сучасні методи оптимізації протоколу OSPF. *Наука сьогодення: від досліджень до стратегічних рішень* : VI Міжнар. студент. наук. конф., м. Чернігів, 10 трав. 2024 р. С. 191–193.

2. Токар М. В. Вдосконалення протоколу OSPF для підтримки безпечної маршрутизації. *Застосування програмного забезпечення в інформаційно-комунікаційних технологіях* : Всеукр. науково-техн. конф., м. Київ, 24 квіт. 2024 р. С. 239–241.

19