004.7
П 542

Міністерство освіти і науки України
Запорізька державна інженерна академія

Н. П. Полякова

МЕРЕЖНА МАРШРУТИЗАЦІЯ ТА КОМУТАЦІЯ

Навчально-методичний посібник

для студентів ЗДІА
спеціальності 121 «Інженерія програмного забезпечення»
денної та заочної форм навчання

Міністерство освіти і науки України
Запорізька державна інженерна академія

Затверджено до друку
рішенням науково-методичної ради ЗДІА
протокол № 8 від14.06.2018

МЕРЕЖНА МАРШРУТИЗАЦІЯ ТА КОМУТАЦІЯ

Навчально-методичний посібник

для студентів ЗДІА
спеціальності 121 «Інженерія програмного забезпечення»
денної та заочної форм навчання

Рекомендовано до видання
на засіданні кафедри ПЗАС,
протокол № 16 від 3.04.2018 р.

Запоріжжя
ЗДІА
2018

ББК 004.7
П 542

Н. П. Полякова, доцент

Відповідальний за випуск: *зав. кафедри ПЗАС,*
професор В.Г.Вербицький

Рецензенти :
Л. І. Цвіркун*, к.т.н., професор, заступник завідувача кафедрою автоматизації та комп'ютерних систем НТУ «Дніпровська політехніка»;*
С. В. Солодухін*, к.е.н., доцент, декан факультету економіки та менеджменту, доцент кафедри економіки та інформаційних технологій Запорізької державної інженерної академії.*

Полякова Н. П.
П 542 Мережна маршрутизація та комутація: навчально-методичний посібник для студентів ЗДІА спеціальності 121 «Інженерія програмного забезпечення» денної та заочної форм навчання / Полякова Н. П.; Запорізька державна інженерна академія. – Запоріжжя: ЗДІА, 2018. – 160с.

Навчально-методичний посібник призначено для студентів спеціальності 121 „Інженерія програмного забезпечення", що навчаються за планом підготовки освітньо-кваліфікаційного рівня бакалавр. Методичне видання містить теоретичний матеріал з курсу "Мережна маршрутизація та комутація", завдання до лабораторних робіт, матеріал для самостійного поглибленого вивчення матеріалу, завдання з контрольної роботи для студентів заочної форми навчання, глосарій, питання до іспиту.

ЗМІСТ

ВСТУП

Навчально-методичний посібник відображає матеріал для вивчення трьох змістовних модулів дисципліни „Мережна маршрутизація”. Вивчення дисципліни «Мережна маршрутизація» базується на знаннях та навичках, які були отримані студентами раніше, при вивченні нормативного курсу «Комп’ютерні мережі (локальні, корпоративні, глобальні)».

Посібник базується на матеріалах електронного курсу програми мережної академії Cisco CCNA Exploration 4.0 «Routing Protocols and Concepts» і призначений для полегшення сприйняття студентами цього англомовного курсу.

В першому модулі розглядаються Тема 1 «Вступ до маршрутизації та пересилання пакетів», Тема 2 «Статична маршрутизація», Тема 3 «Вступ до протоколів динамічної маршрутизації», Тема 4 «Дистанційно-векторні протоколи маршрутизації». Студент вивчить багато важливих термінів, випробує різні концепції під час виконанні лабораторних робіт у Packet Tracer. Знання з цього модуля контролюються за допомогою комп’ютерного тестування.

До модуля 2 включено Тему 5 «RIP v1 – протокол маршрутної інформації», Тему 6 «VLSM і CIDR» та Тему 7 «RIP v2». Виконання лабораторних робіт передбачено як у Packet Tracer, так і на реальному обладнанні Cisco. Знання з цього модуля контролюються за допомогою комп’ютерного тестування.

Модуль 3 включає Тему 8 «Протоколи маршрутизації з урахуванням стану каналу» та Тему 9 «OSPF». Виконання лабораторних робіт передбачено як у Packet Tracer, так і на реальному обладнанні Cisco. Модульний контроль передбачено у вигляді домашньої контрольної роботи.

У посібнику надається як теоретичний матеріал, так і матеріал для виконання лабораторних робіт, контрольної роботи, питання для підготовки до модульного контролю, а також для самостійного поглибленого вивчення матеріалу. Краще орієнтуватися у безлічі нових термінів допоможе глосарій.

У разі ретельного вивчення матеріалу та успішного виконання усіх видів робіт, які передбачені програмою мережної академії Cisco, студент отримає не тільки оцінку з дисципліни, але і сертифікат про успішне завершення навчання за курсом мережної академії Cisco CCNA Exploration 4.0 «Routing Protocols and Concepts».

Тема 1. Вступ до маршрутизації та пересилання пакетів

Ви навчитеся:

- Ідентифікувати маршрутизатор, як комп'ютер з операційною системою, яка призначена для маршрутизації.
- Описувати структуру таблиці маршрутизації.
- Описувати, як маршрутизатор визначає шлях і комутує пакети.

В центрі мережі знаходиться маршрутизатор що з'єднує одну мережу з іншою. Ефективність міжмережних зв'язків залежить, великою мірою від здібності маршрутизаторів до передачі пакетів, використовуючи найбільш ефективний шлях.

Окрім комутації пакетів, маршрутизатор надає інші послуги:

- Гарантія 24x7 придатності. Використання запасних шляхів у разі збою первинного маршруту.
- Інтеграція послуг даних, відео, і голосу по дротяних і бездротових мережах. Маршрутизатори використовують Якість обслуговування (QOS) для установки пріоритетів пакетів IP.
- Пом'якшують напади черв'яків, вірусів і т.п. на мережу, дозволяючи або забороняючи пересилку пакетів.

Основна відповідальність маршрутизатора - пересилка пакетів від однієї мережі до наступної.

1.1 Внутрішня частина маршрутизатора

1.1.1 Маршрутизатори - це комп'ютери

Маршрутизатори мають такі ж самі технічні засоби і програмні компоненти, які є в будь-якому комп'ютері:

- CPU,
- RAM,
- ROM,
- Операційна система.

Маршрутизатор з'єднує безліч мереж. Це означає, що він має безліч інтерфейсів, кожен з яких належить до іншої мережі IP. Коли маршрутизатор отримує IP пакет на одному інтерфейсі, він визначає який інтерфейс використовувати, аби переслати пакет адресатові.

Основне завдання маршрутизатора - направляти пакети, що призначаються для локальних і видалених мереж:

- ***Визначення кращого шляху для пакету***
- ***Відправлення пакетів у напрямі їх адресата***

Маршрутизатор використовує свою таблицю маршрутизації, аби визначити кращий шлях пересилки. Коли маршрутизатор отримує пакет, він розглядає

його IP адресу призначення і шукає відповідності з мережною адресою в таблиці маршрутизації маршрутизатора. Таблиця маршрутизації також включає інтерфейс, який використовується, аби переслати пакет. Як тільки відповідність знайдена, маршрутизатор інкапсулює пакет IP у фрейм канального рівня відповідно до типу мережі exit інтерфейсу, і відправляє його далі.

Для того, щоб дізнатися про віддалені мережі і сформувати для них таблиці маршрутизації, маршрутизатор використовує як статичні маршрути, так і протоколи динамічної маршрутизації.

1.1.2 Процесор маршрутизатора та пам'ять

Різні моделі маршрутизаторів мають загальні апаратні компоненти, розташовані в різних місцях усередині маршрутизатора (рис. 1.1).

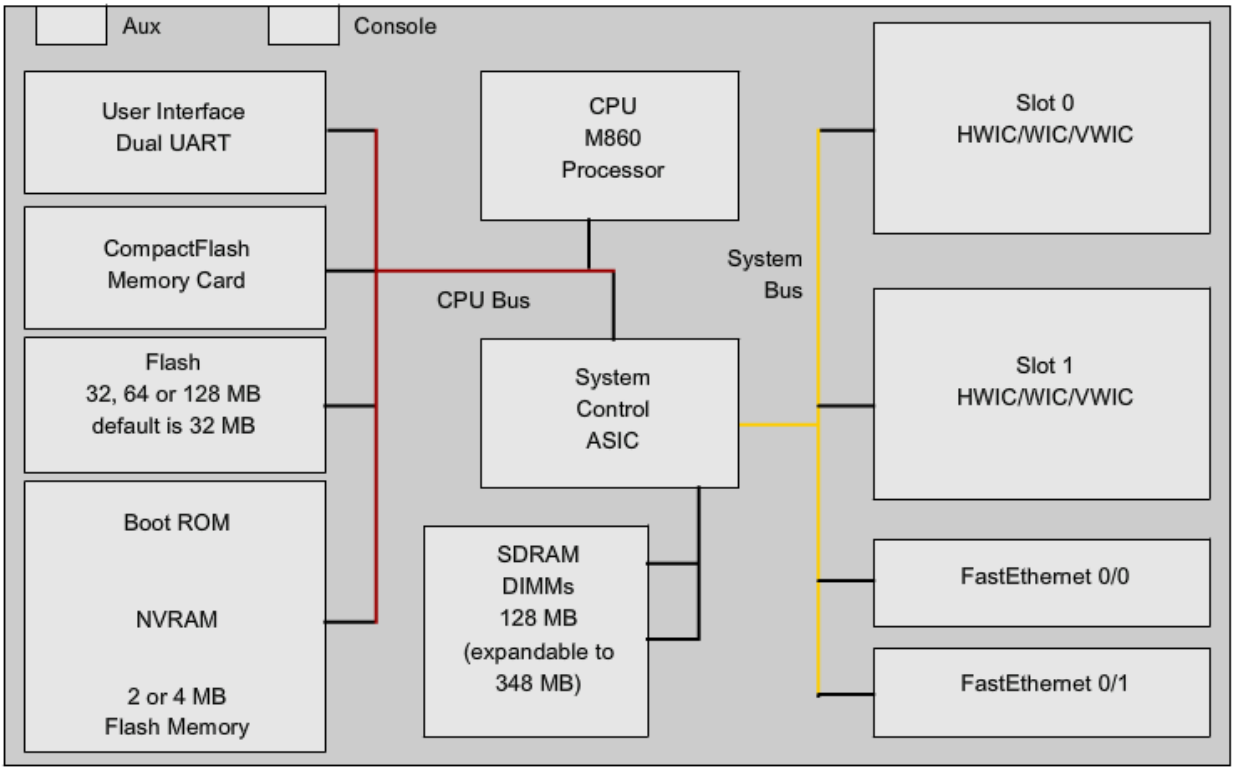


Рис. 1.1 Апаратні компоненти маршрутизатора

Компоненти маршрутизатора і їх Функції

Подібно до ПК, маршрутизатор також включає:

- Центральний процесор - виконує команди операційної системи, ініціалізацію, функції маршрутизації і комутації.
- Пам'ять з довільним доступом (оперативна) – зберігає інструкції і дані, необхідні для їх виконання процесором.
- Пам'ять лише для читання – постійне сховище, де зберігаються: інструкції по початковому завантаженню, основне діагностичне ПЗ і коротка версія операційної системи. Це вбудоване ПЗ і воно не потребує модифікації та апгрейту. Не втрачає вміст при перезапуску.

Флеш пам'ять

Незалежна комп'ютерна пам'ять, яка може бути електрично очищена і перезаписана. Використовується як постійна пам'ять для операційної системи. Під час завантаження маршрутизатора операційна система копіюється в оперативну пам'ять.

Флеш пам'ять не втрачає вміст при перезавантаженні маршрути затора.

NVRAM

NVRAM не втрачає інформацію, коли живлення відключене. Використовується операційною системою Cisco як постійна пам'ять для файлу конфігурації запуску ***startup-config***. NVRAM зберігає його вміст навіть, коли маршрутизатор перезавантажений або відключений.

1.1.3. Міжмережна операційна система

Подібно до будь-якої операційної системи на будь-якому комп'ютері, Cisco IOS керує апаратними і програмними ресурсами маршрутизатора, у тому числі розміщенням пам'яті, процесами, захистом, і файловою системою. Це багатозадачна операційна система в якій інтегровані маршрутизація, комутація, міжмережна взаємодія і функції телекомунікації.

Існує декілька образів ОС. Образ – це файл, який містить повну ОС для даного маршрутизатора, залежно від його моделі. Зазвичай чим більше можливостей, тим більше розмір файлу. Тим більше потребується оперативної пам'яті і флеш-пам’яті. Наприклад, деякі можливості – це здатність виконувати IPv6 або здатність виконувати NAT (трансляцію мережних адрес).

Як і інші, операційна система власний інтерфейс користувача. Хоча деякі маршрутизатори забезпечують графічний інтерфейс (GUI) користувача, інтерфейс командного рядка - набагато загальніший метод конфігурації маршрутизаторів Cisco.

Під час завантаження ***startup-config*** файл з NVRAM копіюється в оперативну пам’ять і запам'ятовується як ***running-config*** файл. ОС виконує конфігураційні команди. Будь-які зміни, введені мережним адміністратором, запам'ятовуються в running-config і негайно виконуються ОС.

1.1.4 Процес завантаження маршрутизатора

Процес Boot-up - чотири основні фази в наступному порядку:

1. Виконання тестування при завантаженні
2. Завантаження програми початкового завантаження
3. Визначення розташування і завантаження програмного забезпечення Cisco ОС (ОС може бути розташована у флеш, але може і на TFTP сервері, якщо повний образ не знайдений, то завантажується коротка версія з ROM)

4. Виявлення і завантаження файлу конфігурації запуску або вхід в конфігураційний режим.

1.2 Формування таблиці маршрутизації

1.2.1 Вступ до таблиці маршрутизації

Первинна функція маршрутизатора - переслати пакет у напрямі його мережі призначення, використовуючи IP адресу призначення пакету. Для цього він шукає інформацію в таблиці маршрутизації.

Таблиця маршрутизації – це файл даних в оперативній пам’яті, який використовується, аби запам'ятати інформацію про маршрути як безпосередньо підключених, иак і віддалених. Таблиця маршрутизації містить вказівки щодо пересилки пакету. Вони говорять маршрутизатору, що адресат може бути оптимально досягнутий, якщо послати пакет до певного маршрутизатора, який представляє "Наступний перехід" по дорозі до кінцевого адресата. Також може бути вказано який вихідний інтерфейс веде до адресата.

Безпосередньо підключена мережа – це мережа, яка безпосередньо прикріплена до одного з інтерфейсів маршрутизатора. Коли інтерфейс маршрутизатора сформований з IP адресою і маскою підмережі, інтерфейс стає хостом на цій прикріпленій мережі. Мережна адреса і маска підмережі інтерфейсу, разом з іншою інформацією записуються в таблицю маршрутизації як безпосередньо підключена мережа.

Віддалена мережа – це мережа, яка безпосередньо не з’єднана з маршрутизатором. Іншими словами, віддалена мережа - мережа, яка може бути досягнута тільки за рахунок відправки пакету до іншого маршрутизатора. Віддалені мережі можуть бути додані до таблиці маршрутизації, або використовуючи протокол динамічної маршрутизації, або конфігурацією статичних маршрутів.

1.2.2. Безпосередньо підключені мережі

Після того, як інтерфейс маршрутизатора сконфігурований і активований командою no shutdown, інтерфейс повинен отримати сигнал від іншого пристрою (маршрутизатор, комутатор, хаб, і т.п.) перед тим, як стан інтерфейсу переїде в «up». Після цього в таблицю маршрутизації додається безпосередньо підключена мережа.

Ця інформація має бути в таблиці маршрутизації до того, як туди потраплять статичні і динамічні маршрути.

1.2.3 Статична маршрутизація

Віддалені мережі додаються до таблиці маршрутизації або ручною конфігурацією статичних маршрутів, або з використанням протоколу динамічної маршрутизації.

Маршрут додається до таблиці маршрутизації тільки, якщо вихідний ін-

терфейс для нього є працездатним. Статичний маршрут позначається літерою S.

Коли використовувати статичні маршрути

- Якщо мережа складається з невеликого числа маршрутизаторів.
- Якщо мережа з'єднана з Інтернет лише через одного провайдера.
- Якщо велика мережа сформована по топології hub-and-spoke.

Зазвичай, таблиці маршрутизації містять комбінацію статичних маршрутів і динамічних маршрутів. Але, таблиця маршрутизації повинна спочатку містити безпосередньо підключені мережі, перед тим, як будь-який статичний або динамічний маршрут зможе використовуватися.

1.2.4 Динамічна маршрутизація

Віддалені мережі можуть також бути додані до таблиці маршрутизації, використовуючи протокол динамічної маршрутизації, наприклад RIP.

Протоколи динамічної маршрутизації використовуються маршрутизаторами, аби вивчити інформацію про досяжність і стан віддалених мереж. Ці протоколи виконують декілька дій, у тому числі:

- Виявлення мереж
- Оновлення і підтримка таблиць маршрутизації

Автоматичне виявлення мереж

Це здатність протоколу маршрутизації поширювати інформацію про мережі, які він знає, серед інших маршрутизаторів, які використовують такий самий протокол маршрутизації.

Підтримка таблиць маршрутизації

Після початкового мережевого відкриття, протоколи динамічної маршрутизації модифікують і підтримують мережі в своїх таблицях маршрутизації. Динамічні протоколи не лише визначають найкращі шляхи до різних мереж, вони також визначать новий кращий шляху, якщо діючий шлях стає непридатною (або, якщо є зміни в топології). По цих причинах, динамічні протоколи мають перевагу над статичними маршрутами. Будь-які зміни топології компенсуються без участі адміністратора.

1.3 Висновки

1.3.1 Резюме

Маршрутизатори та комп'ютери містять багато подібних компонентів, таких як CPU, RAM, ROM, і операційна система.

Основна задача маршрутизатора – з'єднувати багато мереж, та пересилати пакети з однієї мережі до іншої. Тому зазвичай маршрутизатор має багато інтерфейсів. Кожен інтерфейс належить до іншої IP мережі.

Маршрутизатор має таблицю маршрутизації, де перелічені всі мережі, про які він знає. Таблиця містить безпосередньо підключені мережі, а також мережі віддалені. Віддалені мережі можуть бути досягнуті лише пересилкою пакету іншому маршрутизатору.

Віддалені мережі додаються до таблиці маршрутизації у два способи: мережний адміністратор вручну формує статичні маршрути, або впроваджує протокол динамічної маршрутизації.

Протоколи динамічної маршрутизації автоматично враховують зміни топології, без втручання адміністратора. Частіше за все таблиця маршрутизації містить як статичні, так і динамічні маршрути.

Маршрутизатор приймає маршрутне рішення на Рівні 3, на Мережному рівні.

1.3.2 Питання для самоперевірки

1. Опишіть компоненти маршрутизатора та їх призначення.
2. Опишіть процес завантаження маршрутизатора.
3. Які основні задачі виконує маршрутизатор?
4. Опишіть необхідні кроки базової конфігурації маршрутизатора.
5. Опишіть важливість таблиці маршрутизації. Для чого вона потрібна?
6. Якими трьома способами маршрутизатор може дізнатися про мережі?
7. Які поля є найбільш необхідними у IP-заголовку?
8. Опишіть процес інкапсуляції/деінкапсуляції при проходженні пакету від джерела до адресату.

1.3.3 Матеріали для самостійного поглибленого вивчення теми

Дослідіть процес запиту web сторінки з web серверу. Які процеси та протоколи приймають участь у цьому процесі?

Зокрема, для чого потрібен ARP, як використовується DNS, TCP, Ethernet frame?

Тема 2. Статична маршрутизація

Ви навчитеся:

- Визначати основну роль маршрутизатора у мережі.
- Знаходити безпосередньо підключені мережі в таблиці маршрутизації.
- Використовувати CDP.
- Описувати статичні маршрути та маршрути сумарні.
- Шукати несправності в мережах зі статичною маршрутизацією.

Статичні маршрути є найбільш загальними і не вимагають такого процесорного часу і додаткового навантаження, як протоколи динамічної маршрутизації.

2.1 Маршрутизатори та мережі

2.1.1 Роль маршрутизатора

Маршрутизатор – це спеціалізований комп'ютер, який грає ключову роль в будь-якій мережі передачі даних (рис. 2.1). Маршрутизатори перш за все відповідають за взаємне з'єднання мереж за рахунок:

- Визначення кращого шляху відсилання пакету.
- Відправлення пакетів у напрямі їх адресата.

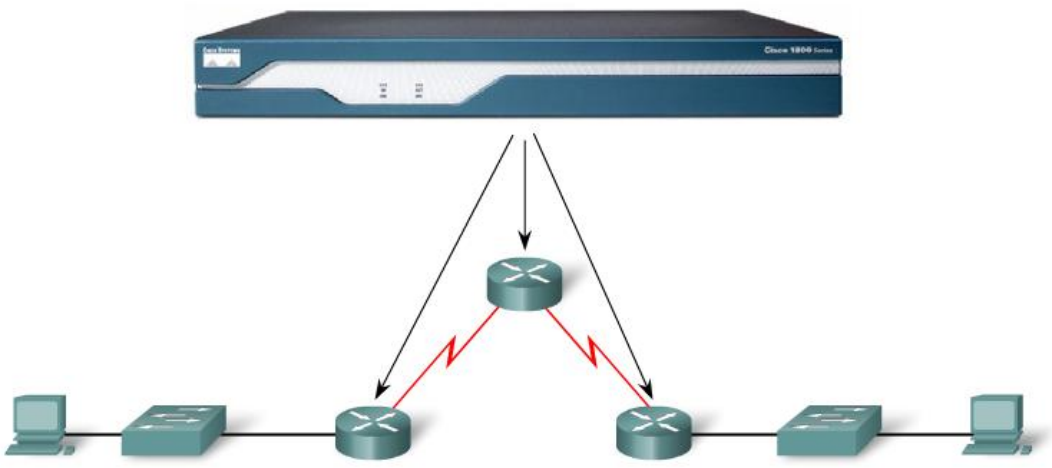


Рис. 2.1 Місце маршрутизатора в мережі

Маршрутизатори виконують пересилку пакету, вивчаючи віддалені мережі та підтримуючи маршрутну інформацію. Вирішення маршрутизатора про напрям пакету ґрунтується на інформації Рівня 3, IP адресі призначення.

Таблиця маршрутизації маршрутизатора використовується, аби знайти кращу відповідність між IP адресою призначення пакету і мережною адресою в таблиці маршрутизації. Таблиця маршрутизації остаточно визначить exit інтерфейс для пересилки пакету і маршрутизатор інкапсулює пакет у фрейм, відповідний вихідному інтерфейсу.

2.1.2 Типова топологія

На рис. 2.2 та 2.3 зображено топологія для дослідження статичної маршрутизації. Кожний маршрутизатор у цьому прикладі - Cisco 1841, який має наступні інтерфейси:

- Два FastEthernet інтерфейси: FastEthernet 0/0 та FastEthernet 0/1
- Два серійні інтерфейси: Serial 0/0/0 та Serial0/0/1

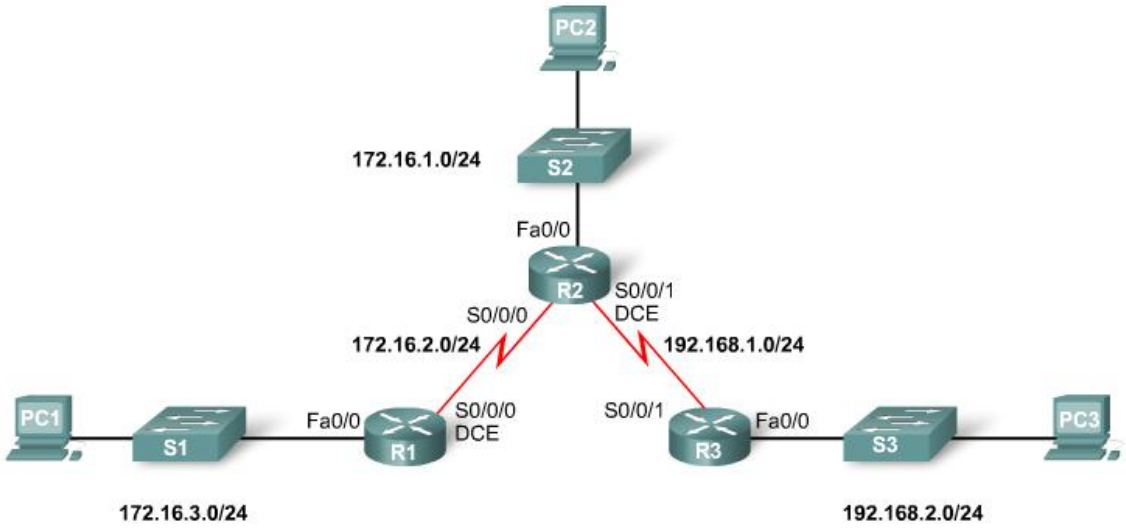


Рис. 2.2 Топологія для дослідження теми

Device	Interface	IP Address	Subnet Mask	Default Gateway
R1	Fa0/0	172.16.3.1	255.255.255.0	N/A
	S0/0/0	172.16.2.1	255.255.255.0	N/A
R2	Fa0/0	172.16.1.1	255.255.255.0	N/A
	S0/0/0	172.16.2.2	255.255.255.0	N/A
	S0/0/1	192.168.1.2	255.255.255.0	N/A
R3	Fa0/0	192.168.2.1	255.255.255.0	N/A
	S0/0/1	192.168.1.1	255.255.255.0	N/A
PC1	NIC	172.16.3.10	255.255.255.0	172.16.3.1
PC2	NIC	172.16.1.10	255.255.255.0	172.16.1.1
PC3	NIC	192.168.2.10	255.255.255.0	192.168.2.1

Рис. 2.3 Адресна схема

2.2 Дослідження безпосередньо приєднаних мереж

2.2.1 Концепції таблиці маршрутизації

Таблиця маршрутизації – це структура даних, яка використовується для збереження маршрутної інформації, отриманої з різних джерел. Основна мета таблиці маршрутизації - забезпечити маршрутизатор шляхами до різних мереж призначення.

Таблиця маршрутизації складається з переліку "відомих" мережених адрес - адрес, які безпосередньо підключені, сформовані статично, і вивчені динамічно. R1 і R2 поки мають лише маршрути для безпосередньо підключених мереж.

Спостерігати у реальному часі за тим, як маршрути додаються до таблиць маршрутизації можна за допомогою команди ***debug ip routing*** (рис. 2.4).

```
R2#debug ip routing
IP routing debugging is on

R2(config)#int fa0/0
R2(config-if)#ip address 172.16.1.1 255.255.255.0
R2(config-if)#no shutdown

%LINK-3-UPDOWN: Interface FastEthernet0/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up

RT: add 172.16.1.0/24 via 0.0.0.0, connected metric [0/0]
RT: interface FastEthernet0/0 added to routing table
```

Рис. 2.4 Додавання маршрутів до таблиці маршрутизації у реальному часі.

Для видалення з таблиці маршрутизації безпосередньо підключеної мережі необхідно використовувати дві команди ***shutdown*** і ***no ip address***. Порядок їх виконання не має значення.

Примітка: Користуватися командами ***debug all*** та ***debug ip routing*** потрібно економно. Вони корисні під час пошуку несправностей, але інтенсивно витрачають ресурси процесора і пам'ять. Рекомендується відключати відладку, як тільки вона не потрібна.

2.2.2 Cisco Discovery Protocol (CDP)

Виявлення мереж з CDP

Cisco Discovery Protocol (CDP) - потужний засіб моніторингу мережі і пошуку несправностей. CDP - інструмент для збору інформації, використовується мережними адміністраторами, аби отримати інформацію про безпосередньо підключені пристрої Cisco. CDP - proprietary інструмент, який дозволяє Вам дістати доступ до короткого звіту протоколів і адресної інформації про Cisco пристрої, які безпосередньо підключені. За замовчанням, кожен пристрій Cisco посилає періодично повідомлення, які відомі як CDP анонси (рис. 2.5), безпосередньо приєднаним пристроям Cisco. Ці анонси містять інформацію, таку як наприклад типи пристроїв, які підключені, інтерфейси маршрутизатора до яких вони приєднані, інтерфейси, які використовуються для підключення і номера моделей пристроїв.

Більшість мережних пристроїв, за визначенням, не працюють в ізоляції. Пристрій Cisco часто має інші пристрої Cisco у якості сусідів по мережі. Інформація, зібрана від інших пристроїв, може допомогти вам в ухваленні рішень при

проектуванні мережі, під час пошуку несправностей, і, при внесенні змін в устаткуванні. CDP може використовуватися як інструмент виявлення мереж, допомагаючи вам побудувати логічну топологію мережі, коли така документація відсутня або недостатньо детальна.

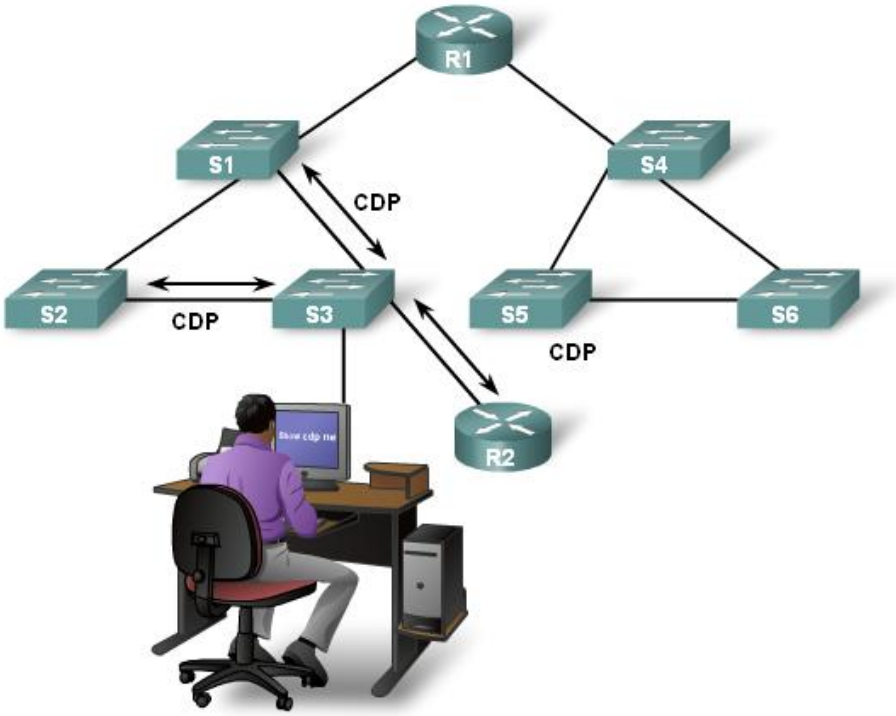


Рис. 2.5 Виявлення CDP-сусідів

Сусіди Рівня 3

На цей момент в конфігурації нашої топології, ми лише безпосередньо підключили сусідів. На рівні 3, протоколи маршрутизації розглядають як сусідів пристрої, які мають один адресний простір.

Наприклад, R1 і R2 - сусіди. Обидва - члени 172.16.1.0/24 мережі. R2 і R3 - також сусіди, тому що вони обидва ділять мережу 192.168.1.0/24. Але R1 і R3 не - сусіди, тому що вони не ділять адресний простір.

Сусіди Рівня 2

CDP працює лише на рівні 2. Тому, CDP сусідами є Cisco пристрої, які безпосередньо підключені фізично і поділяють один канал зв'язку.

Передбачимо, що всі маршрутизатори і комутатори на рис. 2.5 - Cisco пристрої, на яких виконується CDP. Чи можете ви визначити CDP сусідів для кожного пристрою?

Зверніть увагу на різницю між сусідами Рівня 2 і сусідами Рівня 3. Комутатори - не сусіди маршрутизаторам на Рівні 3, тому що комутатори працюють лише на Рівні 2. Проте, комутатори - сусіди Рівня 2 для тих маршрутизаторів, які до них безпосередньо підключені.

Давайте поглянемо, як CDP може бути корисний мережному адміністратору.

Функціонування CDP

Розглянемо детально виведену інформацію командами ***show cdp neighbors*** і ***show cdp neighbors detail*** на рис. 2.6. R3 зібрав детальну інформацію про R2 і комутатор, підключений до Fast Ethernet інтерфейсу R3.

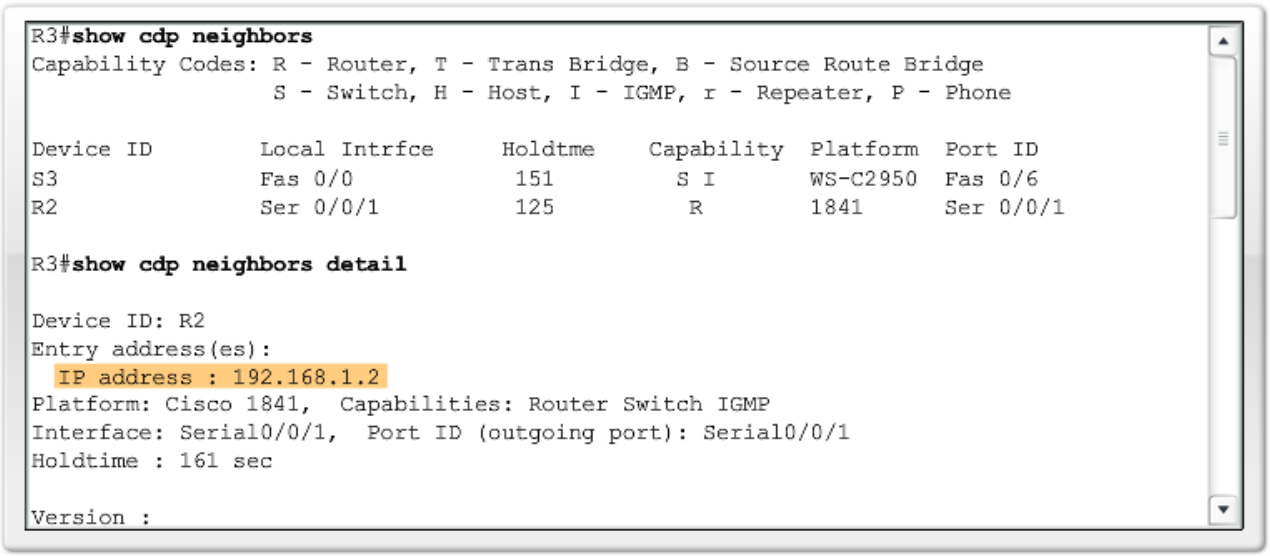


Рис. 2.6 Збір інформації про CDP-сусідів.

CDP працює на Канальному рівні, тому два або більше мережних пристроїв Cisco, наприклад маршрутизатори, які підтримують різні протоколи (наприклад, IP і Novell IPX) Мережного рівня, можуть дізнатися один про одного.

Коли пристрій Cisco завантажується, CDP стартує за замовчанням. CDP автоматично виявляє сусідні пристрої Cisco, виконуючі CDP, не дивлячись на те, який стек протоколів на них працює. CDP обмінюється інформацією про апаратне і програмне забезпечення пристрою зі своїми безпосередньо підключеними сусідами CDP.

CDP забезпечує наступну інформацію про кожен пристрій - CDP сусід:

Device identifiers - Наприклад, сформоване ім'я комутатора

Address list – по одній адресі Мережного рівня для кожного підтримуваного протоколу

Port identifier - ім'я локального або видаленого порту

Capabilities list (Можливості) - Наприклад, цей пристрій - маршрутизатор або комутатор

Platform - базова апаратна платформа пристрою; наприклад, маршрутизатор серії Cisco 7200

holdtime - час утримання інформації - скільки вона зберігатиметься на сусідньому пристрої.

2.2.3 Використання CDP для дослідження мережі

Команди CDP show

Інформація, зібрана CDP протоколом, може бути переглянута командою ***show cdp neighbors***. Для кожного CDP сусіда, наступна інформація відображується:

- Neighbor device ID
- Local interface
- Holdtime value, in seconds
- Neighbor device capability code
- Neighbor hardware platform
- Neighbor remote port ID

Команда show cdp neighbors detail також показує IP адресу сусіднього пристрою. CDP покаже IP адресу сусіда незалежно від того, чи можете Ви пропінгувати сусіда. Ця команда дуже корисна, коли два маршрутизатори Cisco не можуть спілкуватися через свій канал зв'язку, який відкрили. Ккоманда допоможе визначити, можливо один із сусідів CDP має помилку в конфігурації IP.

В ситуації дослідження мережі, знання IP адреси CDP сусіда - часто вся інформація, потрібна для того, щоб зайти по Telnet на цей пристрій. Зі встановленою сесією Telnet, інформація може бути зібрана про сусідів наступного пристрою Cisco. Таким чином, ви можете сформувати логічну топологію.

Відключення CDP

CDP представляє ризик для безпеки. Оскільки деякі версії IOS посилають CDP анонси за замовчанням, важливо знати, як відключити CDP.

Якщо ви хочете відключити CDP глобально, для всього пристрою, використовуйте цю команду:

*Router(config) #**no cdp run***

Якщо хочете зупинити CDP анонси на відповідному інтерфейсі:

*Router(config-if) #**no cdp enable***

2.3 Статична маршрутизація з "Next Hop" адресами

2.3.1 Призначення і синтаксис команди ip route

Статичні маршрути зазвичай використовуються при маршрутизації до мережі-заглушки. Мережа-заглушка - це мережа, доступна по єдиному маршруту. Для прикладу, поглянете на рис. 2.7. Ми бачимо, що будь-яка мережа, прикріплена до R1, має єдиний шлях досягти інших адресатів. Тому, мережа 172.16.3.0 – є заглушкою, і R1 - стабовим маршрутизатором. Піднімати у цьому випадку протокол динамічної маршрутизації - марна витрата ресурсів.

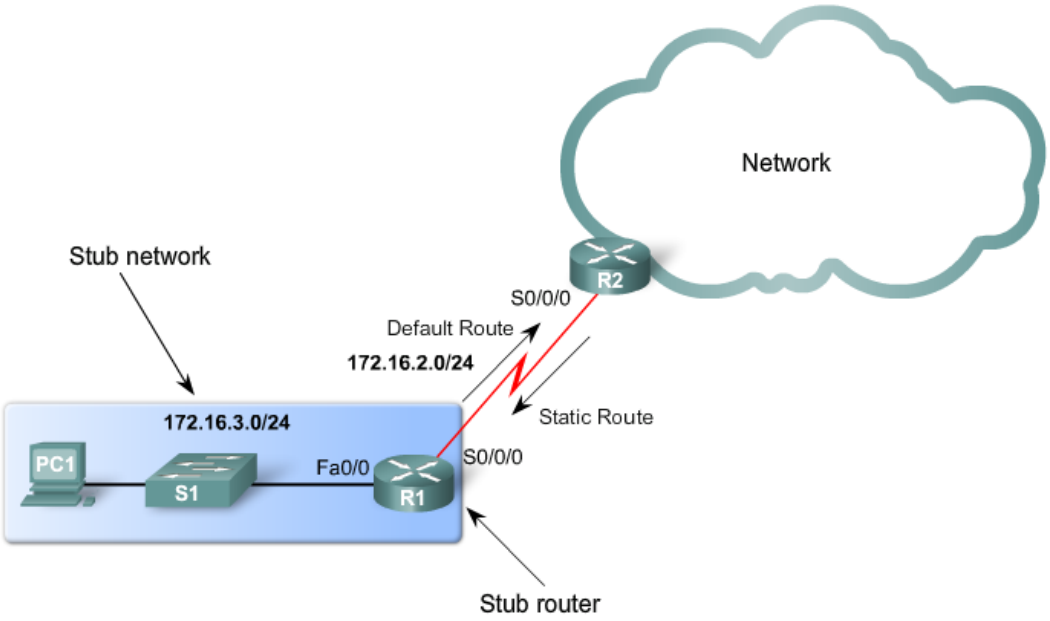


Рис. 2.7 Мережа – заглушка

Команда для конфігурування статичного маршруту - ***ip route***. ЇЇ повний синтаксис:

Router(config)#ip route prefix mask {ip-address | interface-type interface-number [ip-address]} [distance] [name] [permanent] [tag tag]

Існує біль простий синтаксис цієї команди:

Router(config)#ip route network-address subnet-mask {ip-address | exit-interface }

Використовуються наступні параметри:

network-address – адреса мережі призначення для віддаленої мережі, яка додається до таблиці маршрутизації.

subnet-mask – маска підмережі для віддаленої мережі, яка додається до таблиці маршрутизації.

Один чи обидва наступні параметри мають використовуватися:

ip-address - IP адреса next-hop маршрутизатора
exit-interface – вихідний інтерфейс, який буде використовуватися для пересилки пакетів до мережі призначення.

2.3.2 Конфігурація статичних маршрутів

На рис. 2.8 показано конфігурування першого статичного маршрута для нашої топології. Потім show ip route на R1 показує таблицю маршрутизації.

s -означает що маршрут статичний

172.16.1.0 - мережна адреса для маршруту

/24 – маска підмережі для цього маршруту

[1/0] - Адміністративна відстань і метрика для статичного маршруту

172.16.2.2 - адреса IP next-hop маршрутизатора

```
R1#debug ip routing
(**output omitted**)

R1#conf t
R1(config)#ip route 172.16.1.0 255.255.255.0 172.16.2.2

00:20:15: RT: add 172.16.1.0/24 via 172.16.2.2, static metric [1/0]

R1#show ip route
Codes: C - connected, S - static, I - IGRP, 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

     172.16.0.0/24 is subnetted, 3 subnets
S       172.16.1.0 [1/0] via 172.16.2.2
C       172.16.2.0 is directly connected, Serial0/0/0
C       172.16.3.0 is directly connected, FastEthernet0/0
R1#
```

2.8 Конфігурування статичного маршруту

2.3.3 Принципи таблиці маршрутизації і статичні маршрути

Принцип 1: "Кожен маршрутизатор приймає своє рішення сам, ґрунтуючись на інформації, яку має в своїй власній таблиці маршрутизації."

Принцип 2: "Той факт, що один маршрутизатор має певну інформацію в його таблиці маршрутизації не означає, що інші маршрутизатори мають таку ж інформацію."

Принцип 3: "Пересилка інформації по шляху від однієї мережі до іншої не забезпечує пересилку інформації по зворотному шляху."

2.3.4 Зміна статичних маршрутів

Коли заздалегідь сформований статичний маршрут потрібно змінити:

- ***Мережа призначення більше не існує, і тому статичний маршрут має бути видаленим.***
- ***Є зміни в топології, і або проміжну адресу, або exit інтерфейс доведеться замінити.***

Статичний маршрут має бути видалений і новий сформований. Аби видалити статичний маршрут, додайте **no** перед командою IP маршруту, який потрібно видалити.

no ip route 192.168.2.0 255.255.255.0 172.16.2.2

Тепер конфігуруємо новий маршрут.

2.4 Висновки

2.4.1 Резюме

Статичні маршрути можуть використовуватися, аби досягти віддалених мереж. Віддалені мережі - мережі, які можуть бути досягнуті, відправленням пакета іншому маршрутизатору. Статичні маршрути легко сформувати. Проте, у великих мережах ця ручна операція може стати громіздкою. Статичні маршрути все ще використовуються - навіть, коли протокол динамічної маршрутизації реалізовано.

Статичні маршрути можуть бути сформовані з next-hop адресою IP, яка є зазвичай IP адресою next-hop маршрутизатора. На двоточкових мережах зазвичай ефективніше сформувати статичний маршрут з exit інтерфейсом. На мережах множинного доступу, як наприклад Ethernet, для статичного маршруту необхідно вказати як next-hop адресу IP, так і exit інтерфейс.

Статичні маршрути мають задану за замовчанням адміністративну відстань "1". Ця адміністративна відстань застосовується як до статичних маршрутів, сформованих з next-hop адресою так і з exit-інтерфейсом.

Статичний маршрут лише буде введений до таблиці маршрутизації, якщо next-hop адреса IP може бути розрішена до exit інтерфейсу.

У багатьох випадках, декілька статичних маршрутів можуть бути сформовані як єдиний сумарний маршрут. Це дає менше входів в таблиці маршрутизації і призводить до швидшого процесу пошуку в таблиці маршрутизації. Останній сумарний маршрут - заданий за замовчанням маршрут, сформований з 0.0.0.0 мережною адресою і 0.0.0.0 маскою підмережі. Якщо немає певнішої відповідності в таблиці маршрутизації, таблиця маршрутизації використовуватиме заданий за замовчанням маршрут, аби переслати пакет до іншого маршрутизатора.

2.4.2 Питання для самоперевірки

1. За допомогою яких команд можна розглянути інформацію щодо конфігурації інтерфейсу?
2. Що таке CDP? З яких причин адміністратор може його відключити?
3. Запишіть синтаксис команди ip route.
4. Що таке рекурсивний перегляд таблиці маршрутизації і коли він відбувається?
5. Чому статичний маршрут потрібно видалити з конфігурациі перед його модифікацією?
6. Опишіть переваги використання сумарних маршрутів, зокрема маршрута по замовчанню?
7. Запишіть команди, які використовують під час пошуку несправностей в мережі.

Тема 3. Вступ до протоколів динамічної маршрутизації

Ви навчитеся:

- Описувати роль протоколів динамічної маршрутизації і їх місце в контексті проектування сучасних мереж.
- Ідентифікувати різні способи класифікації протоколів маршрутизації.
- Описувати як метрики використовуються протоколами маршрутизації й ідентифікувати тип метрики, що використовується протоколом.
- Визначати адміністративну відстань маршруту й описувати її важливість у процесі маршрутизації.
- Ідентифікувати різні елементи таблиці маршрутизації.
- Для реальних умов розробляти й застосовувати схеми розподілу на підмережі.

Ця глава є вступом до протоколів динамічної маршрутизації, у тому числі розглядається їхня класифікація, яку метрику вони використовують для визначення найкращого шляху, і переваги використання протоколів динамічної маршрутизації.

Протоколи динамічної маршрутизації зазвичай використовуються у великих мережах для спрощення адміністрування й зменшення роботи, порівняно з використанням тільки статичних маршрутів. Зазвичай, у мережі використовується комбінація протоколу динамічної маршрутизації й статичних маршрутів. У більшості мереж використовується тільки один протокол динамічної маршрутизації, однак зустрічаються випадки, коли в різних частинах мережі можуть використовуватися різні протоколи маршрутизації.

Для професіонала в галузі мереж важливо розуміти концепції й роботу різних протоколів динамічної маршрутизації. Професіонал повинен бути здатний прийняти інформоване рішення щодо того, коли використовувати протокол динамічної маршрутизації і який саме протокол буде найкращим вибором для конкретного середовища.

3.1 Вступ і переваги

3.1.1 Перспективи та історія

Еволюція протоколів динамічної маршрутизації

Протоколи динамічної маршрутизації використовуються в мережах з початку 1980-х. Перша версія RIP була випущена в 1982, але деякі з основних алгоритмів цього протоколу використовувалися в мережі ARPANET ще з 1969.

Оскільки мережі еволюціонували, відповідно з'явилися більш складні, нові протоколи динамічної маршрутизації. На рис 3.1 показана класифікація протоколів динамічної маршрутизації. Протоколи, які будуть розглядатися в даному курсі, виділені на цьому рисунку.

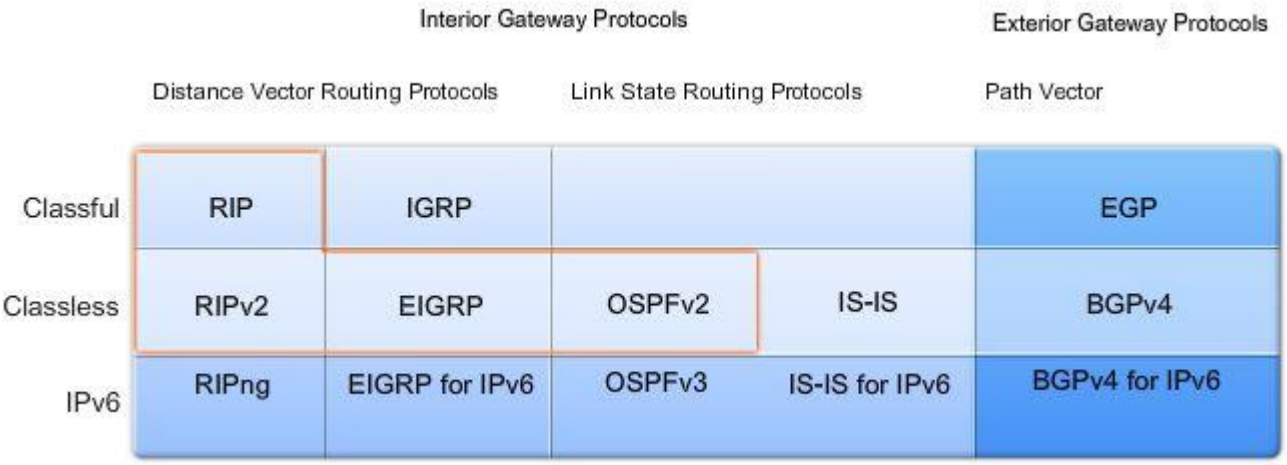


Рис. 3.1 Класифікація протоколів динамічної маршрутизації.

Одним з перших протоколів динамічної маршрутизації був Routing Information Protocol (RIP). RIP еволюціонував у більш нову версію RIPv2. Однак, більш нова версія RIP також не масштабується для використання у великих мережах. Щоб відповідати потребам великих мереж, розроблено два вдосконалені протоколи динамічної маршрутизації: Open Shortest Path First (OSPF) та Intermediate System-to-Intermediate System (IS-IS). Cisco розробила Interior Gateway Routing Protocol (IGRP) - протокол маршрутизації внутрішнього шлюзу та Enhanced IGRP (EIGRP) - удосконалений протокол маршрутизації внутрішнього шлюзу, який також добре масштабується у великих мережних реалізаціях.

Крім того, необхідно з'єднати різні мережі й забезпечити маршрутизацію між ними. Border Gateway Routing (BGP) - протокол граничного шлюзу використовується сьогодні між ISP і більшістю їх клієнтів для обміну маршрутною інформацією.

З появою безлічі користувацьких пристроїв, що використовують IP, адресний простір IPv4 практично вичерпалося. Тому з'явився IPv6. Щоб підтримувати комунікацію, засновану на IPv6, були розроблені більш нові версії протоколів IP маршрутизації (див. рис. 3.1).

Протокол IGRP є попередником EIGRP і зараз вважається застарілим. Тому далі більш докладно будуть розглядатися протоколи RIP, EIGRP і OSPF.

Роль протоколу динамічної маршрутизації

Протоколи динамічної маршрутизації використовуються, щоб сприяти обміну маршрутною інформацією між маршрутизаторами. Протоколи динамічної маршрутизації дозволяють маршрутизаторам динамічно ділитися інформацією про вивчені мережі й автоматично додавати цю інформацію до своїх власних таблиць маршрутизації.

Протоколи динамічної маршрутизації визначають найкращий шлях до кожної мережі, яка потім додається до таблиці маршрутизації. ***Одне з основних переваг використання протоколу динамічної маршрутизації - обмін марш-***

рутною інформацією щоразу, коли є зміни в топології. Цей обмін дозволяє маршрутизаторам автоматично довідуватися про нові мережі, а також знаходити альтернативні шляхи, коли відмовляє один з каналів зв'язку до поточної мережі.

У порівнянні зі статичною маршрутизацією, протоколи динамічної маршрутизації вимагають менше витрат часу адміністратора. Однак, витрати при використанні протоколів динамічної маршрутизації - це виділення частини ресурсів маршрутизатора для роботи протоколу, включаючи процесорний час і пропускну здатність мережі. Незважаючи на переваги динамічної маршрутизації, статична маршрутизація також використовується. Іноді більше підходить статична маршрутизація, в інших випадках кращим вибором є динамічна маршрутизація. Найчастіше зустрічається комбінація обох типів маршрутизації в будь-якій мережі помірного рівня складності.

3.1.2 Виявлення мереж та підтримка таблиці маршрутизації

Мета протоколів динамічної маршрутизації

Протокол динамічної маршрутизації - це набір процесів, алгоритмів, і повідомлень, які використовуються для обміну маршрутною інформацією, і заповнення таблиці маршрутизації найкращими шляхами, які обрав протокол динамічної маршрутизації.

Цілі протоколу динамічної маршрутизації:

- ***Виявлення віддалених мереж.***
- ***Підтримка актуальної маршрутної інформації.***
- ***Вибір найкращого шляху до мереж призначення.***
- ***Здатність знайти новий найкращий шлях, якщо поточний шлях більш не доступний.***

Які компоненти входять до протоколу динамічної маршрутизації ?

- ***Структури даних***. - Деякі протоколи динамічної маршрутизації використовують таблиці й/або бази даних для своїх операцій. Ця інформація зберігається в RAM.
- ***Алгоритм*** - кінцевий перелік кроків, які використовуються при виконанні завдання. Протоколи динамічної маршрутизації використовують алгоритми для поширення маршрутної інформації та для визначення найкращого шляху.
- ***Повідомлення*** протоколу динамічної маршрутизації. - Протоколи динамічної маршрутизації використовують різні види повідомлень для виявлення сусідніх маршрутизаторів, обміну маршрутною інформацією, і розв'язку інших завдань по вивченню й підтримці точної інформації про мережу.

Робота протоколу динамічної маршрутизації

Всі протоколи динамічної маршрутизації мають одну мету - довідатися про віддалені мережі та швидко адаптуватися до змін у топології. Метод, який використовується протоколом динамічної маршрутизації для досягнення цієї мети, залежить від алгоритму, що їм використовується та від робочих характеристик цього протоколу. Робота протоколу динамічної маршрутизації сильно залежить від типу протоколу маршрутизації й самого протоколу. Загалом, робота протоколу динамічної маршрутизації може бути описана в такий спосіб:

- Маршрутизатор посилає й одержує повідомлення з маршрутною інформацією на своїх інтерфейсах.
- Маршрутизатор обмінюється маршрутними повідомленнями й маршрутною інформацією з іншими маршрутизаторами, які використовують такий самий протокол маршрутизації.
- Маршрутизатори обмінюються маршрутною інформацією, щоб довідатися про віддалені мережі.
- Коли маршрутизатор виявляє зміну топології, протокол маршрутизації може анонсувати цю зміну іншим маршрутизаторам.

Примітка: Розуміння роботи протоколу динамічної маршрутизації, і його використання у реальних мережах вимагає твердого знання IP адресації та розподілу мереж на підмережі.

3.1.3 Переваги

Використання статичної маршрутизації

Динамічна маршрутизація, безумовно має деякі переваги над статичною. Однак статична маршрутизація все ще використовується сьогодні в мережах. Фактично, мережі зазвичай використовують комбінацію як статичної, так і динамічної маршрутизації.

Статична маршрутизація має кілька основних застосувань, у тому числі:

- Забезпечує простоту підтримки таблиці маршрутизації в маленьких мережах, для яких не очікується значний ріст.
- Маршрутизація в/з мереж-заглушок (див. Тему 2).
- Використання єдиного маршруту за замовчуванням, як шляху до будь-якої мережі, яка не має більш певної відповідності з іншим маршрутом у таблиці маршрутизації.

Переваги й недоліки статичної маршрутизації

При безпосередньому порівнянні динамічної й статичної маршрутизації видно, що переваги одного методу є недоліками іншого.

Переваги статичної маршрутизації:

- Мінімальна потреба у процесорному часі.
- Простіше для розуміння адміністратора.
- Легко конфігурувати.

Недоліки статичної маршрутизації:

- Конфігурація й обслуговування забирають багато часу.
- Конфігурація піддається помилкам, особливо у великих мережах.
- Для заміни маршруту потрібне втручання адміністратора.
- Погано масштабується з ростом мереж; обслуговування стає громіздким.
- Вимагає повних знань про всій мережі для належної реалізації.

Переваги й недоліки динамічної маршрутизації

Переваги динамічної маршрутизації:

- В адміністратора менше роботи з підтримки конфігурації при додаванні або видаленні мереж.
- Протоколи автоматично реагують на зміни топології.
- Конфігурація менш піддана помилкам.
- Більш масштабовані, ріст мережі зазвичай не є проблемою.

Недоліки динамічної маршрутизації:

- Використовуються ресурси маршрутизатора (процесорний час, пам'ять і пропускна здатність каналів зв'язку).
- Для конфігурації, перевірки, і пошуку несправностей потрібно більше знань адміністратора.

3.2 Класифікація протоколів динамічної маршрутизації

3.2.1 Короткий огляд

Класифікація протоколів динамічної маршрутизації

Протоколи динамічної маршрутизації можуть бути класифіковані по різних групах відповідно до їхніх характеристик (рис. 3.2). Протоколи, які використовуються найбільш часто:

- RIP - дистанційно-векторний протокол внутрішньої маршрутизації.
- IGRP - дистанційно-векторний протокол внутрішньої маршрутизації розроблений Cisco (застарілий для 12.2 IOS і новіших).
- OSPF - протокол внутрішньої маршрутизації з урахуванням стану каналу.
- IS-IS - протокол внутрішньої маршрутизації з урахуванням стану каналу.
- EIGRP - вдосконалений дистанційно-векторний протокол внутрішньої маршрутизації, розроблений Cisco.
- BGP - векторний протокол зовнішньої маршрутизації.

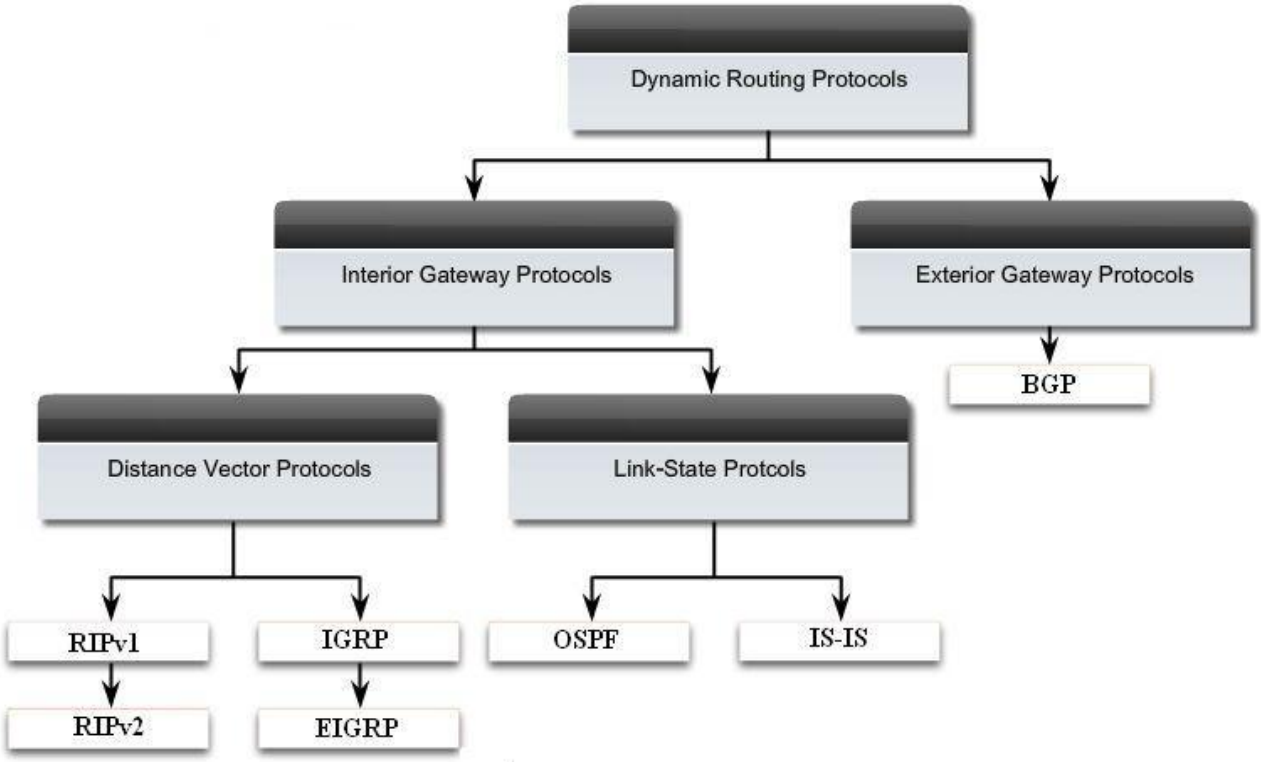


Рис. 3.2 Класифікація протоколів динамічної маршрутизації

3.2.2 IGP і EGP

Автономна система (autonomous system - AS) - інакше відома як домен маршрутизації - група маршрутизаторів під загальним адмініструванням. Типові приклади - внутрішня мережа компанії й мережа провайдера Інтернет. Оскільки Інтернет базується на концепції автономної системи, потрібно два види протоколів маршрутизації: внутрішні й зовнішні (рис. 3.3).

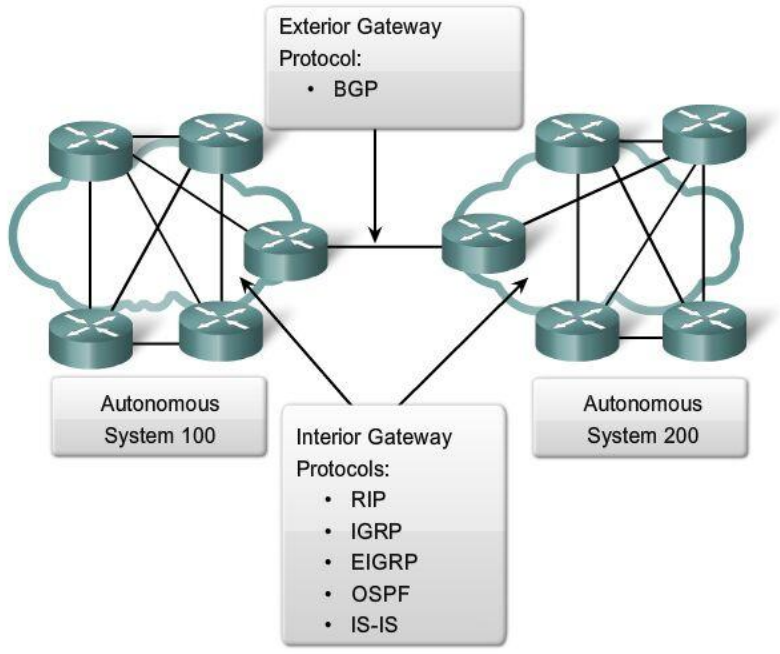


Рис 3.3 IGP і EGP протоколи маршрутизації

Цими типами протоколів є:

Interior Gateway Protocols (IGP) - протоколи внутрішніх шлюзів - використовуються для маршрутизації усередині автономної системи.

Exterior Gateway Protocols (EGP) - протоколи зовнішніх шлюзів - використовуються для маршрутизації між автономними системами.

Характеристики IGP і EGP протоколів маршрутизації

IGP використовуються для маршрутизації в межах домену маршрутизації, мереж під адміністративним контролем однієї організації. Автономна система зазвичай складається з безлічі індивідуальних мереж, що належать компаніям, школам, та іншим установам. IGP використовується, щоб здійснювати маршрутизацію у межах автономної системи, а також усередині окремих мереж. IGP для IP включають RIP, IGRP, EIGRP, OSPF, і IS-IS.

Протоколи маршрутизації, а точніше алгоритми, які в них працюють, використовують метрику для визначення найкращого шляху до мережі. Метрика, що використовується протоколом RIP - це число переходів (hop count), тобто число маршрутизаторів, які пакет повинен пройти для досягнення іншої мережі. OSPF використовує пропускну здатність, щоб визначити найкоротший шлях.

EGP, з іншого боку, розроблені для використання між різними автономними системами, які перебувають під керуванням різних адміністраторів. BGP є на сьогодні єдиним життєздатним EGP, він є протоколом маршрутизації в Інтернет. BGP - протокол вектора шляху (path vector protocol), який може використовувати багато різних атрибутів для виміру маршрутів. На рівні провайдера ISP, часто є більш важливі проблеми, ніж вибір найшвидшого шляху. BGP зазвичай використовується між ISP і іноді між компанією та провайдером.

3.2.3 Дистанційно-векторні протоколи й протоколи з урахуванням стану каналу

Interior Gateway Protocols (IGPs) можуть бути класифіковані по двом типам:

- Дистанційно-векторні протоколи маршрутизації.
- Протоколи маршрутизації з урахуванням стану каналу.

Робота дистанційно-векторного протоколу маршрутизації

«Дистанційно-векторний» означає, що маршрути анонсуються як вектори відстані й напрямку. Відстань визначена в термінах метрики, наприклад число переходів, а напрямок - це просто next-hop маршрутизатор або exit-інтерфейс. Дистанційно-векторні протоколи зазвичай використовують ***алгоритм Беллмана Форда*** для визначення найкращого шляху.

Деякі ***дистанційно-векторні протоколи періодично відправляють повністю таблиці маршрутизації всім підключеним сусідам***. У великих мережах ці маршрутні оновлення можуть стати величезними, що викликає істотний трафік на каналах.

Хоча алгоритм Беллмана-Форда таки накопичує досить знань, щоб підтримувати базу даних доступних мереж, алгоритм не дозволяє маршрутизатору знати точну топологію мережі. ***Маршрутизатор знає тільки маршрутну інформацію, отриману від його сусідів.***

Єдина інформація, яку маршрутизатор знає про віддалену мережу, - відстань або метрика, щоб досягти цієї мережі і який шлях або інтерфейс використовувати, щоб дістатися до неї. Дистанційно-векторні протоколи не мають фактичної карти мережної топології.

Дистанційно-векторні протоколи працюють найкраще в ситуаціях, де:

- ***Мережа проста й плоска, та не вимагає спеціального ієрархічного проектування.***
- ***Адміністратори не мають досить знань для конфігурування й пошуку несправностей у протоколах з урахуванням стану каналу.***
- ***Певні типи мереж, як наприклад hub-and-spoke мережі.***
- ***Повільна конвергенція в мережі не є перешкодою.***

Функції й робота дистанційно-векторного протоколу маршрутизації будуть розглянуті далі. Ви також довідаєтеся про роботу й конфігурації дистанційно-векторних протоколів маршрутизації RIP і EIGRP.

Робота протоколу з урахуванням стану каналу

На контрасті з роботою дистанційно-векторного протоколу маршрутизації, маршрутизатор, на якому працює протокол маршрутизації з урахуванням стану каналу, може створити "Повне уявлення" або топологію мережі, збираючи інформацію від усіх інших маршрутизаторів. Усі маршрутизатори стану каналу використовують ідентичну "карту" мережі. Маршрутизатор стану каналу використовує інформацію про стан каналу, щоб створити карту топології й вибрати кращий шлях до всіх мереж призначення в топології.

Протоколи з урахуванням стану каналу не використовують періодичні оновлення. Після того, як мережа конвергує, оновлення посилають тільки з появою змін у топології.

Протоколи з урахуванням стану каналу працюють найкраще в ситуаціях, де:

- Мережний проект ієрархічний, зазвичай у великих мережах.
- Адміністратори мають гарні знання про впровадження протоколу маршрутизації з урахуванням стану каналу.
- Швидка конвергенція для мережі є критичною вимогою.

Функції й операції протоколу маршрутизації з урахуванням стану каналу будуть пояснені в останніх темах. Ви також довідаєтеся про роботу й конфігурацію протоколу маршрутизації з урахуванням стану каналу OSPF.

3.2.4 Класова й безкласова

Класові протоколи маршрутизації

Класові протоколи маршрутизації не розсилають маску підмережі у оновленнях маршрутної інформації. Перші протоколи маршрутизації, наприклад RIP були класовими. У цей час призначення адрес ґрунтувалося на класах A, B, або C. Протоколу маршрутизації не потрібно було розсилати маску, тому що вона завжди могла бути визначена на основі першого октету адреси.

Класові протоколи маршрутизації не можуть використовуватися, коли мережа розділена на підмережі з використанням більш ніж однієї маски підмережі (рис. 3.4). Класові протоколи маршрутизації не підтримують маски змінної довжини (VLSM).

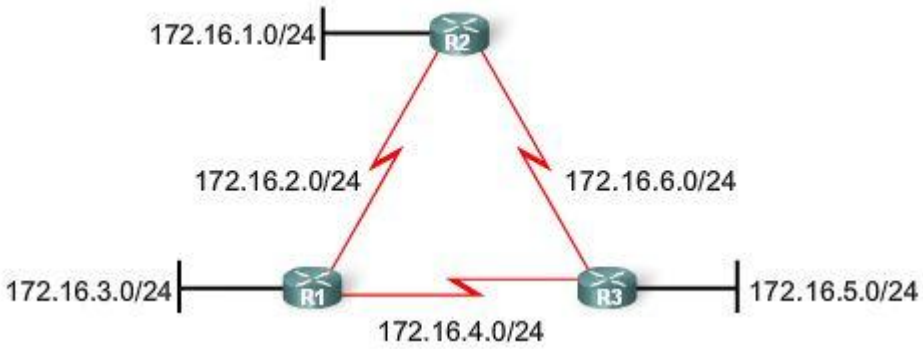


Рис. 3.4 Класовий протокол маршрутизації: однакова маска використовується для всієї топології

Є й інші обмеження на класові протоколи маршрутизації, включаючи їх нездатність до підтримки несуміжних мереж.

До класових протоколів маршрутизації відносяться RIPv1 та IGRP.

Безкласові протоколи маршрутизації

Безкласові протоколи маршрутизації включають маску підмережі разом з адресою мережі ***в маршрутні оновлення***. Сучасні мережі засновані не тільки на класах, маску підмережі не можна визначити за значенням першого октету. Сьогодні в більшості мереж потрібні безкласові протоколи маршрутизації через їхню ***підтримку VLSM***, несуміжних мереж (discontiguous networks) і інших особливостей.

Безкласові протоколи маршрутизації - RIPv2, EIGRP, OSPF, IS-IS, BGP.

На рис. 3.5 показана безкласова версія мережі, в одній і тій же топології використовуються маски /30 і /27. До того ж, ця топологія використовує несуміжні мережі.

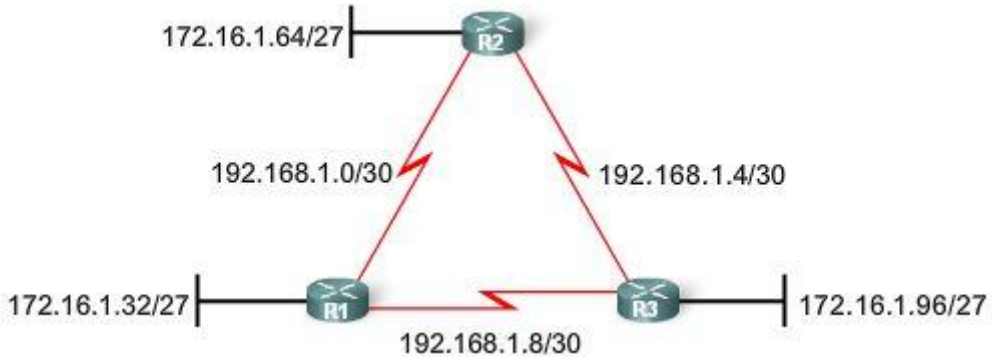


Рис. 3.5 Класовий протокол маршрутизації: у топології можуть використовуватися різні маски

3.2.5 Конвергенція

Що таке конвергенція?

Конвергенція - тоді, коли таблиці маршрутизації всіх маршрутизаторів перебувають у несуперечливому стані. Мережа конвергувала, коли всі маршрутизатори мають повну й точну інформацію про мережу. Час конвергенції - це час, який потрібен маршрутизаторам для поширення інформації, обчислення найкращих шляхів, і модифікації їхніх таблиць маршрутизації. ***Мережа не в повній мірі працездатна, поки процес конвергенції не завершено, тому, для більшості мереж потрібен короткий час конвергенції.***

Конвергенція - процес одночасно партнерський і незалежний. Маршрутизатори діляться інформацією один з одним, але повинні незалежно обчислити вплив змін топології на їхні власні маршрути. Т.ч. до нової топології вони адаптуються незалежно.

Властивості конвергенції включають швидкість поширення маршрутної інформації й обчислення оптимальних шляхів. Чим швидше конвергенція, тим краще протокол маршрутизації. Загалом, RIP і IGRP конвергують повільно, тоді як EIGRP і OSPF мають більш швидку конвергенцію.

3.3 Метрика

3.3.1 Мета метрики

Бувають випадки, коли протокол маршрутизації довідається більш ніж один маршрут до того самого адресата. Щоб обрати найкращий шлях, протокол маршрутизації повинен бути здатний оцінити доступні шляхи й зробити вибір. Для цієї мети використовується метрика. ***Метрика*** – це значення, що використовується протоколами маршрутизації для призначення вартості досягнення віддалених мереж. Метрика використовується, щоб визначити, який шлях найбільш кращий, коли є безліч шляхів до однієї віддаленої мережі, ***найбільш кращий шлях – це шлях з найменшою метрикою***.

Кожний протокол маршрутизації використовує свою власну метрику.

Наприклад, RIP використовує кількість переходів, EIGRP використовує комбінацію пропускної здатності й затримки. Cisco реалізація OSPF використовує пропускну здатність. Кількість переходів - метрика, яку найбільше просто собі уявити. Це число маршрутизаторів, через які пакет повинен пройти, щоб досягти мережі призначення. На рис. 3.6 для R3 мережа 172.16.3.0 перебуває у двох переходах, або на відстані двох маршрутизаторів.

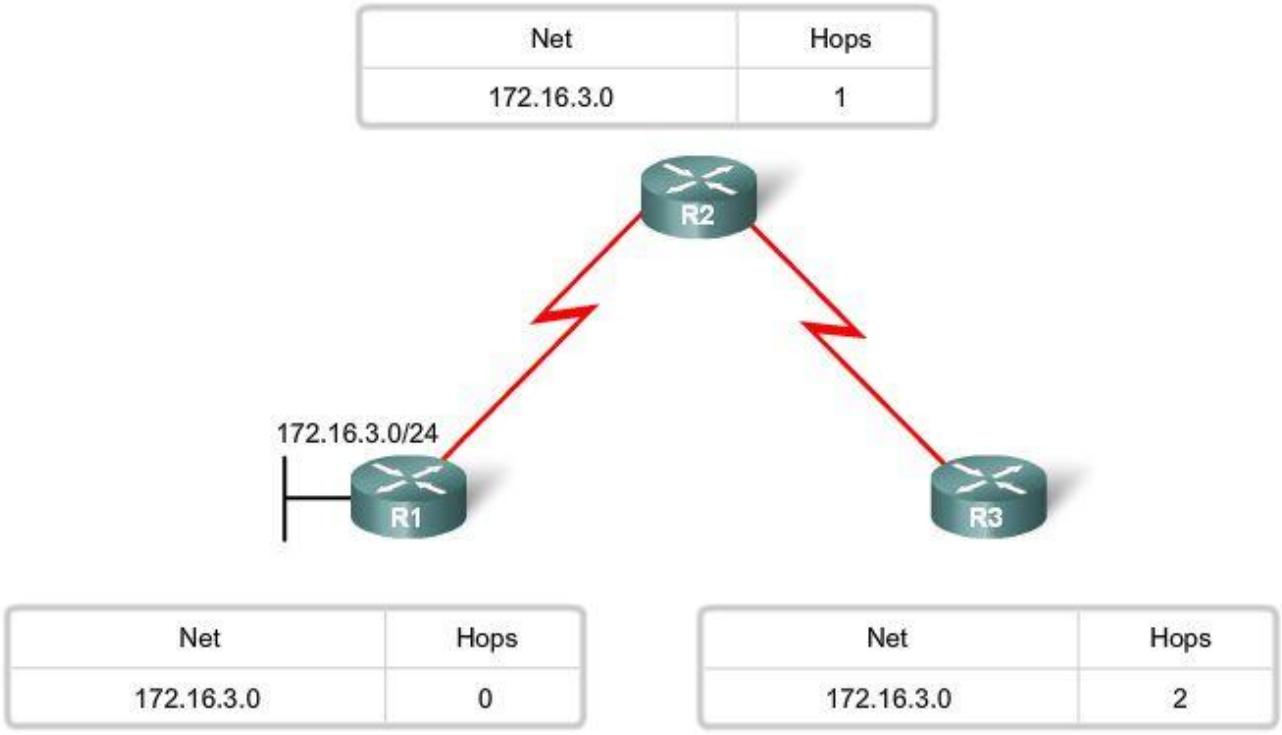


Рис. 3.6 Метрики

Яка метрика використовується для конкретного протоколу маршрутизації, і як вона обчислюється, розглядається під час детального опису цього протоколу маршрутизації.

3.3.2 Метрики й протоколи маршрутизації

Метричні параметри

Різні протоколи маршрутизації використовують різні метрики. Метрика, що використовується одним протоколом маршрутизації, не може бути порівняна з метрикою, що використовується іншим протоколом маршрутизації. Два різні протоколи маршрутизації, можливо, оберуть різні шляхи до того самого адресата завдяки використанню іншої метрики.

RIP обрав би шлях з найменшою кількістю переходів, тоді як OSPF обрав би шлях з найвищою пропускною здатністю.

Метрики, що використовуються протоколами IP маршрутизації, є такими:

Hop count - простий показник, який перелічує число маршрутизаторів, які пакет має пройти.

Bandwidth - пропускна здатність - перевага надається шляху з найвищою пропускною здатністю.

Load - Завантаження - Розглядається трафік для конкретного каналу зв'язку.

Delay - затримка, ураховує час, який необхідно пакету, щоб подолати шлях.

Reliability - надійність - Оцінює ймовірність відмови каналу зв'язку, обчислюється залежно від кількості помилок на інтерфейсі або попередніх відмов каналу.

Cost - вартість - значення, що встановлюється IOS або мережним адміністратором, щоб вказати перевагу для маршруту. Вартість може бути метрикою, комбінацією метрик або політикою.

Поле метрики в таблиці маршрутизації

Метрикою для кожного протоколу маршрутизації є:

RIP: Hop count - У якості кращого шляху обирається маршрут із самим маленьким числом переходів.

IGRP і EIGRP: Bandwidth, Delay, Reliability, and Load - У якості кращого обирається маршрут із самою маленькою величиною комплексної метрики, яка розраховується по цій безлічі параметрів. За замовчуванням, при розрахунках використовуються тільки пропускна здатність і затримка.

IS-IS і OSPF: Cost - У якості кращого шляху обирається маршрут з найнижчою вартістю. Cisco реалізація OSPF використовує пропускну здатність.

Протоколи маршрутизації визначають найкращий шлях, ґрунтуючись на найменшій метриці.

Нехай у мережі використовується протокол маршрутизації RIP. Метрика, зіставлена з певним маршрутом, може бути переглянута за допомогою команди ***show ip route***. Значення метрики – це друга величина у квадратних дужках в записі маршруту.

*R 192.168.8.0/24 [120/**2**] via 192.168.4.1, 00:00:26, Serial0/0/1*

У цьому випадку в маршрутизатора є маршрут до мережі 192.168.8.0/24, яка перебуває на відстані двох переходів.

3.3.3 Балансування навантаження

Що відбувається, коли два або більше маршрутів до одного адресата мають ідентичні метричні значення? Як буде маршрутизатор вирішувати, який з них використовувати для відправлення пакета? У цьому випадку, маршрутизатор не обирає тільки один маршрут. Він робить балансування завантаження ("load balances") між цими шляхами рівної вартості.

Щоб подивитися, чи виконується балансування завантаження, перевірте таблицю маршрутизації. Балансування завантаження включене, якщо

два або більше маршрутів пов'язані з одним адресатом (рис. 3.7).

```
R2#show ip route
 (**output omitted**)

R    192.168.6.0/24 [120/1] via 192.168.2.1, 00:00:24, Serial0/0/0
                    [120/1] via 192.168.4.1, 00:00:26, Serial0/0/1
```

Рис. 3.7 Балансування завантаження по шляхах рівної вартості

Примітка: Балансування завантаження може виконуватися по пакетах, або по адресату.

Всі протоколи маршрутизації, розглянуті в цьому курсі, здатні автоматично проводити балансування завантаження трафіка аж до чотирьох маршрутів рівної вартості за замовчуванням. EIGRP також здатний до балансування завантаження через шляхи нерівної вартості.

3.4 Адміністративні відстані

3.4.1 Мета адміністративної відстані

Безліч джерел маршруту

Маршрутизатор може довідатися про маршрут до однієї і тієї ж мережі з більш ніж одного джерела. Наприклад, до мережі сформований статичний маршрут і маршрутизатор одержав інформацію про неї від динамічного протоколу маршрутизації, наприклад RIP. Маршрутизатор повинен вибрати, який маршрут встановити.

Примітка: Не плутати зі шляхами однакової вартості. Безліч маршрутів до однієї і тієї ж мережі можуть бути встановлені тільки, якщо вони отримані від того ж самого джерела. Наприклад, два рівні за вартістю маршрути можуть бути встановлені статично або отримані обоє по протоколу RIP.

Хоча й зрідка, але в одній і тій же мережі може бути розгорнуто кілька протоколів динамічної маршрутизації. У деяких випадках, може бути необхідно маршрутизувати ту саму мережну адресу, використовуючи кілька протоколів маршрутизації, наприклад RIP і OSPF. Оскільки різні протоколи маршрутизації використовують різні метрики, RIP використовує число переходів, а OSPF використовує пропускну здатність, то не можливо порівняти метрики, щоб визначити найкращий шлях.

Тоді, яким же чином маршрутизатор визначає, який маршрут йому встановити в таблиці маршрутизації, якщо він довідався про одну й ту ж мережу з декількох джерел?

Мета адміністративної відстані

Administrative distance (AD) - адміністративна відстань визначає перевагу джерела маршруту. Кожне джерело маршрутної інформації - включаючи певні

протоколи маршрутизації, статичні маршрути, і навіть безпосередньо підключені мережі - розташовано за пріоритетами у порядку від найбільш до найменш кращого, з використанням величини адміністративної відстані. Маршрутизатори Cisco використовують можливості AD, щоб вибрати кращий шлях, якщо довідалися про одну й ту ж мережу призначення із двох або більш різних джерел маршрутної інформації.

Адміністративна відстань – це ціле значення від 0 до 255. Більш низьке значення - джерело маршруту, якому надається більше переваги. Адміністративна відстань 0 - найбільша перевагая. Тільки безпосередньо підключена мережа має адміністративну відстань 0, яка не може бути змінена.

Адміністративна відстань для статичних маршрутів і протоколів динамічної маршрутизації може бути змінена.

Адміністративна відстань 255 означає, що маршрутизатор не довіряє джерелу цього маршруту, і він не буде встановлений у таблиці маршрутизації.

Примітка: ***Термін trustworthiness - вартість надійності - звичайно використовується при визначенні адміністративної відстані. Менша адміністративна відстань відповідає більш надійному маршруту.***

Величина AD – це перше значення у квадратних дужках маршрутного запису з таблиці маршрутизації. У прикладі на рис. 3.8 R2 має маршрут до мережі 192.168.6.0/24 з адміністративною відстанню 90.

*D 192.168.6.0/24 [**90**/2172416] via 192.168.2.1, 00:00:24, Serial0/0/0*

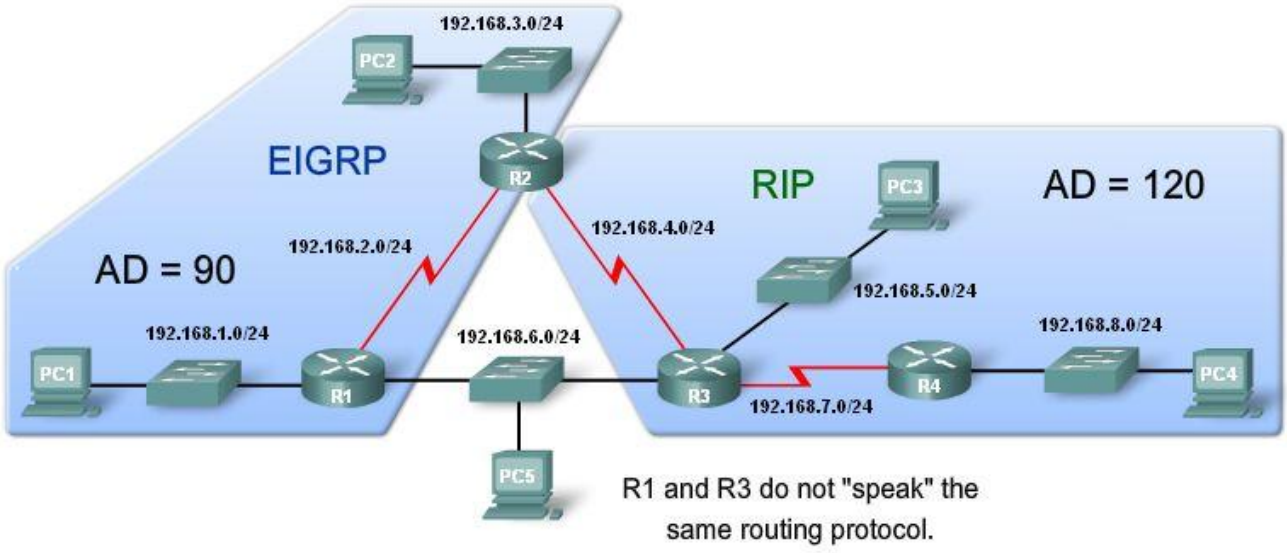


Рис. 3.8 Порівняння адміністративних відстаней

На рис. 3.8, на маршрутизаторі R2 працюють одночасно два протоколи маршрутизації RIP і EIGRP. R2 вивчив маршрут 192.168.6.0/24 від R1 через EIGRP оновлення, і від R3 через RIP оновлення. ***RIP має адміністративну від-***

стань 120, але EIGRP має більш низьку адміністративна відстань 90. Тому R2 додає маршрут, вивчений із використанням EIGRP до таблиці маршрутизації й пересилає всі пакети до мережі 192.168.6.0/24 на маршрутизатор R1.

Що відбудеться якщо канал до R1 стане недоступний? Тоді R2 не буде мати маршруту до 192.168.6.0. Насправді, R2 усе ще має маршрутну інформацію RIP для 192.168.6.0, збережену в базі даних RIP. Це можна перевірити командою ***show ip rip database*** (рис. 3.9). Ця команда показує всі RIP маршрути, вивчені R2, незалежно від того, чи встановлений цей RIP маршрут в таблиці маршрутизації.

```
R2#show ip rip database
192.168.3.0/24    directly connected, FastEthernet0/0
192.168.4.0/24    directly connected, Serial0/0/1
192.168.5.0/24
    [1] via 192.168.4.1, Serial0/0/1
192.168.6.0/24
    [1] via 192.168.4.1, Serial0/0/1
192.168.7.0/24
    [1] via 192.168.4.1, Serial0/0/1
192.168.8.0/24
    [2] via 192.168.4.1, Serial0/0/1
```

Рис. 3.9 База даних RIP

3.4.2 Протоколи динамічної маршрутизації

Перевірити значення AD можна за допомогою команди ***show ip route***. Крім того, величина AD може бути перевірена командою ***show ip protocols***. Ця команда відображає всю інформацію про протоколи маршрутизації, які працюють на маршрутизаторі.

Різні протоколи маршрутизації мають різну адміністративну відстань (рис 3.10).

Route Source	Administrative Distance
Connected	0
Static	1
EIGRP summary route	5
External BGP	20
Internal EIGRP	90
IGRP	100
OSPF	110
IS-IS	115
RIP	120
External EIGRP	170
Internal BGP	200

Рис. 3.10 Адміністративні відстані, прийняті за замовчуванням.

3.4.3 Статичні маршрути

Як відомо з Теми 2, статичні маршрути встановлюються адміністратором, який прагне вручну сформувати кращий шлях до адресата. Із цієї причини, статичні маршрути мають задане за замовчуванням значення AD=1. Це означає, що після безпосередньо підключених мереж, які мають задане за замовчуванням значення AD=0, статичні маршрути – джерело, якому надається перевага.

Бувають ситуації, коли адміністратор формує статичний маршрут до того ж адресата, який вже вивчений із протоколу динамічної маршрутизації, але з використанням іншого шляху. Статичний маршрут потрібно сформувати з більшим AD, ніж у протокола маршрутизації. Якщо на шляху, який використовується протоколом динамічної маршрутизації відмовить канал зв'язку, маршрут цього протоколу буде вилучений з таблиці маршрутизації. Статичний маршрут у цьому випадку стане єдиним джерелом і автоматично буде доданий у таблицю маршрутизації. Такий підхід зветься плаваючим статичним маршрутом.

Статичний маршрут, що використовує next-hop IP адресу або exit-інтерфейс, має задану за замовчуванням величину AD=1 (рис 3.11). Однак, AD величина не відображається командою, коли ви формуєте статичний маршрут із зазначенням exit-інтерфейсу. Коли статичний маршрут сформований з exit-інтерфейсом, команда show ip route показує мережу, яка безпосередньо підключена через цей інтерфейс (рис. 3.12).

```
R2#show ip route


Gateway of last resort is not set

     172.16.0.0/24 is subnetted, 3 subnets
C       172.16.1.0 is directly connected, FastEthernet0/0
C       172.16.2.0 is directly connected, Serial0/0/0
S       172.16.3.0 is directly connected, Serial0/0/0
C    192.168.1.0/24 is directly connected, Serial0/0/1
S    192.168.2.0/24 [1/0] via 192.168.1.1
```

Рис.3.11 Статичні маршрути в таблиці маршрутизації

```
R2#show ip route 172.16.3.0
Routing entry for 172.16.3.0/24
Known via "static", distance 1, metric 0 (connected)
  Routing Descriptor Blocks:
  * directly connected, via Serial0/0/0
      Route metric is 0, traffic share count is 1
```

Рис. 3.12 Адміністративна відстань для статичного маршруту

Статичний маршрут до мережі 172.16.3.0 відображається як безпосередньо підключена мережа, і адміністративна відстань не зазначена (рис. 3.11). Загальною помилкою є вважати, що величина AD для цього маршруту повинна бути рівною 0, оскільки його статус "directly connected." Однак, це неправильне

припущення. За замовчуванням, AD=1 для будь-якого статичного маршруту, включаючи ті, які сформовані з exit-інтерфейсом. Пам'ятайте, що AD=0 мають тільки безпосередньо підключені мережі. Це можна перевірити, розширивши команду ***show ip route*** опцією ***[route]***. Вказавши ***[route]*** можна побачити докладну інформацію про маршрут, включно з величиною AD (рис. 3.12).

3.4.4. Безпосередньо підключені мережі

Безпосередньо підключені мережі з'являються в таблиці маршрутизації, як тільки сформовано IP адресу на інтерфейсі й інтерфейс переходить у стан «up». Величина AD для безпосередньо підключених мереж становить 0, що означає те джерело, до якого найбільше довіри. Найкращий маршрут для маршрутизатора, якщо він має один зі своїх інтерфейсів, безпосередньо з'єднаний із цією мережею. З цієї причини, адміністративну відстань безпосередньо підключеної мережі не можна поміняти й ніяке інше джерело маршруту не може мати адміністративну відстань 0.

Команда ***show ip route*** на рис. 3.13 відображає безпосередньо підключені мережі без інформації про AD. Це схоже на інформацію про статичні маршрути, сформовані з exit-інтерфейсом. Відмінність тільки в літері ***C*** на початку запису, яка є ознакою безпосередньо підключеної мережі.

```
R2#show ip route
Codes: C - connected, S - static, I - IGRP, 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

     172.16.0.0/24 is subnetted, 3 subnets
C       172.16.1.0 is directly connected, FastEthernet0/0
C       172.16.2.0 is directly connected, Serial0/0/0
S       172.16.3.0 is directly connected, Serial0/0/0
C    192.168.1.0/24 is directly connected, Serial0/0/1
S    192.168.2.0/24 [1/0] via 192.168.1.1
```

Рис. 3.13 Безпосередньо підключені мережі в таблиці маршрутизації

Команда ***show ip route 172.16.1.0*** показує адміністративну відстань для безпосередньо підключеної мережі рівну 0 (рис. 3.14).

```
R2#show ip route 172.16.1.0
Routing entry for 172.16.1.0/24
Known via "connected", distance 0, metric 0 (connected, via interface)
  Routing Descriptor Blocks:
  * directly connected, via FastEthernet0/0
      Route metric is 0, traffic share count is 1
```

Рис. 3.14 Адміністративна відстань для безпосередньо підключеної мережі

3.5 Висновки

3.5.1 Резюме

Протоколи динамічної маршрутизації використовуються маршрутизаторами, щоб автоматично дізнаватися про віддалені мережі від інших маршрутизаторів.

Ви довідалися, що протоколи маршрутизації можуть бути класифіковані або як класові й безкласові, або як дистанційно-векторні й стану каналу або вектора шляху. Протокол маршрутизації може бути протоколом внутрішнього шлюзу або зовнішнього шлюзу.

Протоколи маршрутизації не тільки виявляють віддалені мережі, але й мають процедуру для підтримки актуальної мережної інформації. Коли є зміни в топології, завдання протоколу маршрутизації інформувати інші маршрутизатори про ці зміни.

Коли є зміни в мережній топології, деякі протоколи маршрутизації можуть поширювати цю інформацію через домен маршрутизації швидше, ніж інші протоколи маршрутизації. Процес приведення всіх таблиць маршрутизації до стану несуперечності називається конвергенцією. Конвергенція - це коли всі маршрутизатори в одному домені маршрутизації мають повну й точну інформацію про мережу.

Метрики використовуються протоколами маршрутизації, щоб визначити кращий шлях або найкоротший шлях досягнення мережі призначення. Різні протоколи маршрутизації можуть використовувати різні метрики. Зазвичай, менша метрика означає кращий шлях. П'ять переходів, щоб дістатися мережі краще, ніж 10 переходів.

Маршрутизатори іноді дізнаються про безліч маршрутів до однієї і тієї ж мережі як від статичних маршрутів, так і від протоколів динамічної маршрутизації. Коли маршрутизатор дізнається про мережу призначення більш ніж від одного джерела, маршрутизатори Cisco використовують адміністративну відстань, щоб визначити, яке джерело використовувати. Кожен протокол динамічної маршрутизації має унікальну адміністративну відстань, не виключаючи статичні маршрути й безпосередньо підключені мережі. Більш низьке значення адміністративної відстані означає більшу довіру до джерела маршрутизації.

Безпосередньо підключена мережа – це маршрут, якому завжди надається найбільша перевага, за нею ідуть статичні маршрути, а потім різні протоколи динамічної маршрутизації.

3.5.2 Питання для самоперевірки

1. Поясніть, чому статична маршрутизація може бути більш кращої, ніж динамічна ?
2. Які чотири способи класифікації протоколів динамічної маршрутизації

Ви знаєте?

3. Які метрики найчастіше використовуються протоколами динамічної IP маршрутизації?
4. Що таке адміністративна відстань і в чому її важливість?
5. Можна сказати, що кожний маршрутизатор повинен мати принаймні один статичний маршрут. Поясніть, чому це твердження може бути істинним?

3.5.3 Матеріали для самостійного поглибленого вивчення теми

Border Gateway Protocol (BGP) – протокол маршрутизації між автономними системами – це протокол маршрутизації Інтернет. Хоч BGP тільки стисло оглядається у цьому курсі, Ви можете зацікавитися переглядом маршрутних таблиць деяких ключових маршрутизаторів Інтернет.

Сервера маршрутизації використовуються для перегляду BGP маршрутів в Інтернет. Різноманітні web сайти надають доступ до серверів маршрутизації (route server), наприклад ***www.traceroute.org.*** Після обрання серверу маршрутизації, Ви можете розпочати telnet сесію на цьому сервері маршрутизації. Цей сервер віддзеркалює ключовий маршрутизатор Інтернет, який частіше за все є Cisco маршрутизаторомr.

Потім Ви можете використати команду ***show ip route*** для перегляду акту маршрутизатора Інтернет. Використайте цю команду з відкритою IP адресою, наприклад show ip route 207.62.187.0.

Ви не зможете зрозуміти всю інформацію, яка надається, але дасть Вам змогу відчути розмір таблиці маршрутизації на ключовому сервері Інтернету.

Тема 4. Дистанційно-векторні протоколи маршрутизації

Ви навчитеся:

- Ідентифікувати характеристики дистанційно-векторних протоколів маршрутизації.
- Описувати процес дослідження нових мереж на прикладі дистанційно-векторного протоколу маршрутизації RIP.
- Описувати процеси, які використовують дистанційно-векторні протоколи маршрутизації для підтримки таблиць маршрутизації.
- Описувати умови, що ведуть до утворювання петель маршрутизації, і пояснювати їхній вплив на продуктивність маршрутизатора.
- Ідентифікувати типи дистанційно-векторних протоколів маршрутизації, які використовуються сьогодні.

Ця тема описує характеристики, роботу й функціональність дистанційно-векторних протоколів маршрутизації. Існують переваги й недоліки використання протоколу маршрутизації будь-якого типу. Тому, описуються умови, що впливають на роботу дистанційно-векторних протоколів, пастки в роботі дистанційно-векторних протоколів, а також засоби подолання таких пасток. Розуміння роботи дистанційно-векторного протоколу маршрутизації є критичним для розгортання, перевірки й пошуку несправностей таких протоколів.

4.1 Вступ до дистанційно-векторних протоколів маршрутизації

4.1.1 Дистанційно-векторні протоколи маршрутизації

Протоколи динамічної маршрутизації допомагають мережному адміністратору заощаджувати багато часу в порівнянні із процесом конфігурування й підтримки статичних маршрутів. Тому протоколи динамічної маршрутизації є правильним вибором для великих мереж.

До дистанційно-векторних протоколів відносяться RIP, IGRP і EIGRP.

RIP

Routing Information Protocol (RIP) – протокол маршрутної інформації - був описаний в RFC 1058. Він має наступні ключові характеристики:

- У якості метрики для вибору шляху використовує число переходів (Hop count).
- Якщо число переходів для мережі більше, ніж 15, RIP не може підтримувати маршрут до цієї мережі.
- Оновлення маршрутизації за замовчуванням виконуються кожні 30 секунд широкомовно або у вигляді групової розсилки.

IGRP

Interior Gateway Routing Protocol (IGRP) - протокол маршрутизації внутрішнього шлюзу - приватний протокол, розроблений Cisco. IGRP має наступні ключові характеристики:

- Пропускна здатність, затримка, завантаження й надійність використовуються, щоб сформувати комплексну метрику.
- Оновлення маршрутної інформації по замовченню розсилаються широкомовно кожні 90 секунд.
- IGRP є попередником EIGRP і на сьогодні застарів.

EIGRP

Enhanced IGRP (EIGRP) – вдосконалений протокол маршрутизації внутрішнього шлюзу - приватний дистанційно-векторний протокол маршрутизації Cisco. EIGRP має наступні ключові характеристики:

- Він може виконувати балансування завантаження по шляхах нерівної вартості.
- Він використовує Diffusing Update Algorithm (DUAL), щоб обчислити найкоротший шлях.
- Немає ніяких періодичних оновлень, як в RIP і IGRP. Оновлення маршрутної інформації посилають тільки, коли є зміни в топології.

4.1.2 Дистанційно-векторна технологія

Значення Distance Vector

«Дистанційно-векторні» означає, що маршрути анонсуються як вектори відстані й напрямку. Відстань визначена в термінах метрики, як наприклад число переходів, а напрямок - просто next-hop маршрутизатор або exit-інтерфейс.

Маршрутизатор, що використовує дистанційно-векторний протокол маршрутизації, не має відомостей про повний шлях до мережі призначення. Замість цього маршрутизатор знає тільки:

- Напрямок, або інтерфейс, на який пакети повинні бути переслані.
- Відстань, або як далеко розташована мережа призначення.

Наприклад, на рис. 4.1, R1 знає, що відстань до мережі 172.16.3.0/24 становить 1 перехід, і що напрямок – це вихідний інтерфейс S0/0/0.

Робота дистанційно-векторних протоколів маршрутизації

Деякі дистанційно-векторні протоколи маршрутизації змушують маршрутизатор періодично передавати всю таблицю маршрутизації кожному зі своїх сусідів. Цей метод неефективний, тому що відновлення витрачають не тільки пропускну здатність, але й ресурси процесора маршрутизатора для обробки оновлень.

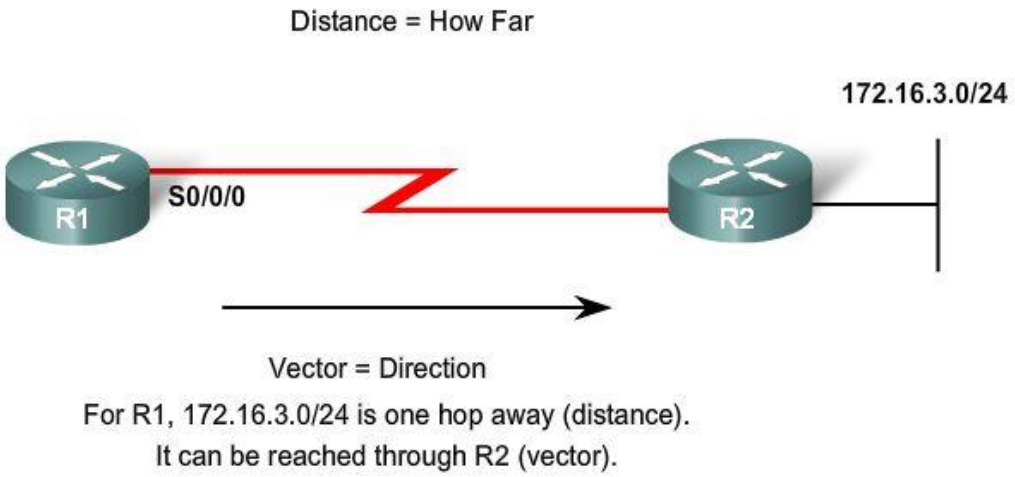


Рис. 4.1 Пояснення терміну «дистанційно-векторний»

Дистанційно-векторні протоколи маршрутизації мають наступні загальні характеристики:

Періодичні оновлення посилаються через регулярні інтервали часу (30 секунд для RIP і 90 секунд для IGRP). Навіть якщо топологія не змінюється кілька днів, періодичні оновлення продовжують відсилатися всім сусідам.

Сусіди - це маршрутизатори, які ділять з маршрутизатором один канал зв'язку й сконфігуровані на використання такого ж протоколу маршрутизації. Маршрутизатор має відомості тільки про мережні адреси своїх власних інтерфейсів і адреси віддалених мереж, яких він може дістатися через своїх сусідів. Маршрутизатори, що використовують дистанційно-векторні протоколи маршрутизації, не проінформовані про топологію мережі в цілому.

Широкомовні оновлення відсилаються на адресу 255.255.255.255. Сусідні маршрутизатори, які працюють з тим самим протоколом маршрутизації, будуть обробляти ці оновлення. Усі інші пристрої також оброблять оновлення аж до Рівня 3 перед тим, як скинуть його. Деякі дистанційно-векторні протоколи маршрутизації використовують групові адреси розсилки замість широкомовних.

Оновлення всієї таблиці маршрутизації відсилаються, з деякими виключеннями, які обговорюються пізніше, періодично до всіх сусідів. Сусіди, що одержують ці оновлення, повинні обробити повне оновлення, щоб знайти потрібну інформацію й відкинути іншу. Деякі дистанційно-векторні протоколи маршрутизації, подібно EIGRP, не посилають періодичні оновлення таблиці маршрутизації.

4.1.3 Алгоритми протоколів маршрутизації

Мета алгоритму

В основі дистанційно-векторного протоколу маршрутизації перебуває алгоритм. Алгоритм використовується, щоб обчислити кращі шляхи, а потім ві-

дправити цю інформацію сусідам.

Алгоритм – це процедура для виконання певного завдання. Різні протоколи маршрутизації використовують різні алгоритми, щоб встановити маршрути в таблиці маршрутизації, відправити оновлення сусідам, і прийняти рішення при визначенні шляху.

Алгоритм, який використовується для протоколів маршрутизації, визначає наступні процеси:

- Механізм для відсилання й одержання маршрутної інформації.
- Механізм для обчислення кращих шляхів і установки маршрутів у таблицю маршрутизації.
- Механізм для виявлення й реакції на зміни топології.

4.1.4 Характеристики протоколів маршрутизації

Протоколи маршрутизації можна порівнювати, ґрунтуючись на наступних характеристиках:

Time to Convergence - Час конвергенції - Час конвергенції визначає, як швидко маршрутизатори в мережній топології, поділяться інформацією, і досягнуть стану несуперечливого знання. Більш швидка конвергенція - більш кращий протокол. Коли непослідовні таблиці маршрутизації не оновлені до точних через повільну конвергенцію в мережі, яка зазнала змін, можуть з'являтися петлі маршрутизації.

Scalability - Універсальність або масштабованість - визначає, наскільки великою може стати мережа, ґрунтуючись на розгорнутому протоколі маршрутизації. Чим більше мережа, тим більше масштабований протокол маршрутизації потрібен.

Безкласовий Classless(використання VLSM) або класовий Classful - Безкласові протоколи включають маску підмережі у оновлення. Ця особливість підтримує використання масок змінної довжини (VLSM) і кращу суммаризацію маршрутів. Класові протоколи не включають маску підмережі й не можуть підтримувати VLSM.

Resource Usage - Використання ресурсів - включає вимоги протоколу маршрутизації такі, як наприклад простір пам'яті, використання процесора і пропускної здатності. Більш високі вимоги до ресурсів роблять необхідним використання могутніших технічних засобів, щоб підтримувати роботу протоколу маршрутизації, на додаток до пересилання пакетів.

Implementation and Maintenance - Реалізація й обслуговування - Реалізація й обслуговування описує рівень знань, які вимагаються від мережного адміністратора, щоб розгорнути й підтримувати мережу, засновану на розгорнутому протоколі маршрутизації.

Переваги й недоліки дистанційно-векторних протоколів маршрутизації показано в таблиці 4.1.

Таблиця 4.1 Переваги й недоліки дистанційно-векторних протоколів маршрутизації

Переваги:	Недоліки:
Проста реалізація й обслуговування. Рівень знань, який потрібен для розгортання й пізніше для обслуговування мережі з дистанційно векторним протоколом, є невисоким	***Повільна конвергенція.*** Використання періодичних оновлень є причиною повільної конвергенції. Навіть якщо використовуються самі передові техніки, наприклад миттєві оновлення, загальна конвергенція однаково повільніше, ніж у протоколів з урахуванням стану каналу.
Низькі вимоги до ресурсів. Дистанційно-векторні протоколи зазвичай не мають потреби у великих обсягах пам'яті для зберігання інформації. Вони не вимагають потужного процесора. Залежно від розміру мережі й адресної схеми, вони не вимагають великої пропускної здатності каналу для розсилання маршрутних оновлень. Однак, більша пропускна здатність може знадобитися, якщо розгорнути дистанційно-векторний протокол у великій мережі.	***Обмежена масштабованість.*** Низька конвергенція може обмежити розмір мережі, оскільки більші мережі будуть вимагати більше часу для поширення маршрутної інформації.
	Петлі маршрутизації. Петлі маршрутизації можуть виникати, якщо через повільну конвергенцію в зміненій мережі з'являються непослідовні, не оновлені таблиці маршрутизації.

4.2 Виявлення мереж

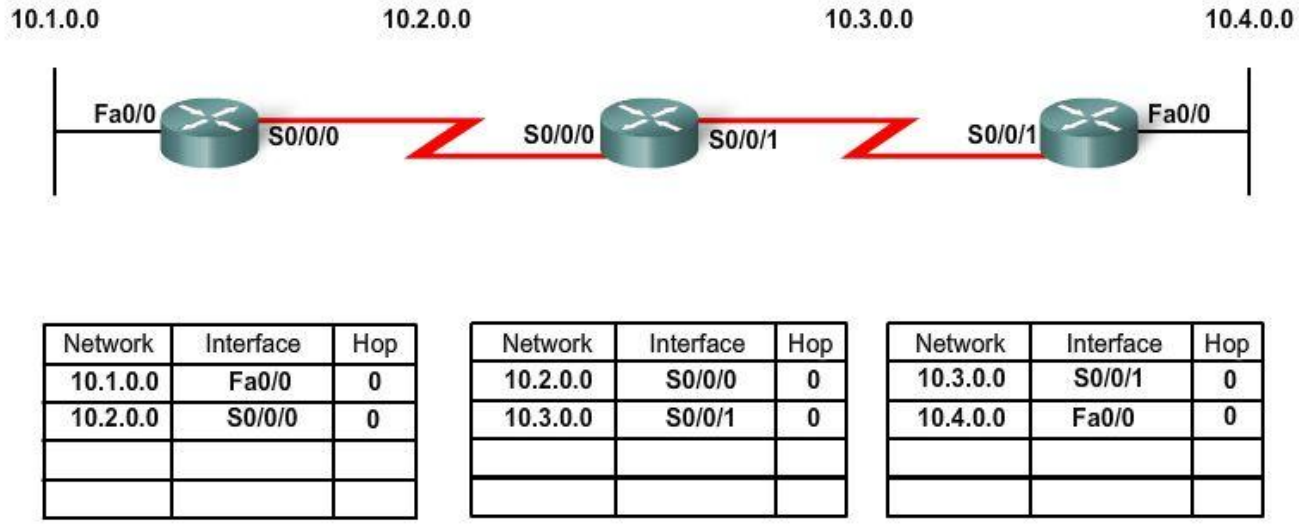
4.2.1 Холодний старт

Коли маршрутизатор стартує, він не знає нічого про мережну топологію. Він навіть не знає, чи є пристрої на інших кінцях його каналів. Єдина інформація, яка відома маршрутизатору - та, що є в його власному конфігураційному файлі, який збережено в NVRAM. Як тільки маршрутизатор успішно завантажився, він застосовує збережену конфігурацію. Якщо IP адресація сформована правильно, то маршрутизатор спочатку виявить свої власні безпосередньо підключені мережі.

Початкове виявлення мереж

У прикладі на рис. 4.2, після холодного запуску й перед обміном маршрутною інформацією, маршрутизатори спочатку виявляють свої власні безпосередньо підключені мережі й маски підмереж. Ця інформація додається до їхніх таблиць маршрутизації:

Із цією початковою інформацією маршрутизатори починають обмін маршрутною інформацією.



Network	Interface	Hop
10.1.0.0	Fa0/0	0
10.2.0.0	S0/0/0	0

Network	Interface	Hop
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/1	0

Network	Interface	Hop
10.3.0.0	S0/0/1	0
10.4.0.0	Fa0/0	0

Рис. 4.2 Виявлення мереж – холодний старт

4.2.2 Початковий обмін маршрутною інформацією

Якщо протокол маршрутизації сконфігуровано, маршрутизатори починають обмінюватися маршрутними оновленнями. ***Спочатку, ці оновлення включають тільки інформацію про їхні безпосередньо підключені мережі***. При одержанні оновлення, маршрутизатор перевіряє його на нову інформацію. Будь-які маршрути, які не перебувають у цей час у його таблиці маршрутизації, додаються до неї.

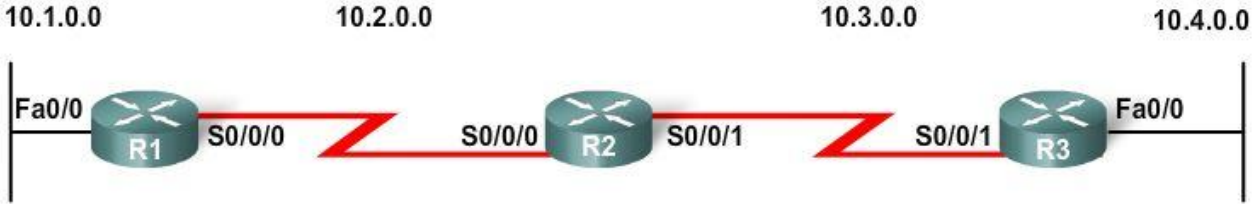
Початковий обмін

Усі три маршрутизатори відсилають свої таблиці маршрутизації своїм сусідам. Розглянемо приклад, як маршрутизатор R2 обробляє оновлення:

- Посилає оновлення про мережу 10.3.0.0 через інтерфейс Serial 0/0/0.
- Посилає оновлення про мережу 10.2.0.0 через інтерфейс Serial 0/0/1.
- Одержує оновлення від R1 про мережу 10.1.0.0 з метрикою 1.
- Зберігає мережу 10.1.0.0 у таблиці маршрутизації з метрикою 1.
- Одержує оновлення від R3 про мережу 10.4.0.0 з метрикою 1.
- Зберігає мережу 10.4.0.0 у таблиці маршрутизації з метрикою 1

Після цього першого кола обміну оновленнями, кожний маршрутизатор знає про мережі, які безпосередньо підключені до його сусідів (рис. 4.3). Повне

знання й конвергенція мережі не будуть мати місця, доки не відбудеться наступний обмін маршрутною інформацією.



Network	Interface	Hop
10.1.0.0	Fa0/0	0
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/0	1

Network	Interface	Hop
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/1	0
10.1.0.0	S0/0/0	1
10.4.0.0	S0/0/1	1

Network	Interface	Hop
10.3.0.0	S0/0/0	0
10.4.0.0	Fa0/0	0
10.2.0.0	S0/0/1	1

Рис. 4.3 Стан таблиць маршрутизації після першого кола обміну оновленнями

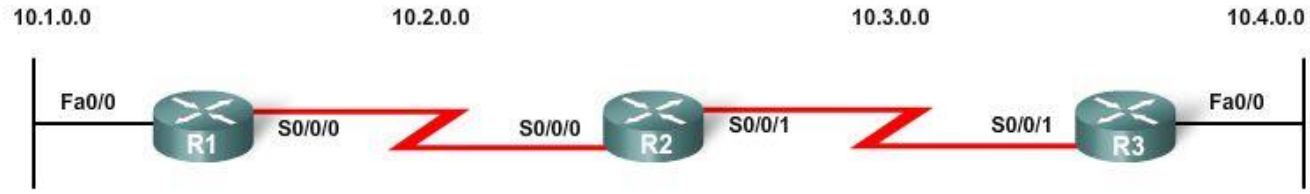
4.2.3 Обмін маршрутною інформацією

У цей момент маршрутизатори знають про всі власні безпосередньо підключені мережі й про мережі, які підключені до їх безпосередніх сусідів. Продовжуючи рух у напрямку конвергенції, маршрутизатори проводять наступне коло періодичних оновлень (рис. 4.4). Кожен маршрутизатор знову перевіряє отримані оновлення на нову інформацію.

Розглянемо, як працює з оновленнями маршрутизатор R1:

- Посилає оновлення про мережу 10.1.0.0 через інтерфейс Serial 0/0/0.
- Посилає оновлення про мережі 10.2.0.0 і 10.3.0.0 через інтерфейс Ethernet0/0.
- Одержує оновлення від R2 про мережу 10.4.0.0 з метрикою 2.
- Зберігає мережу 10.4.0.0 у таблиці маршрутизації з метрикою 2.
- Це ж оновлення від R2 містить інформацію про мережу 10.3.0.0 з метрикою 1. Змін не відбувається, оскільки маршрутна інформація та ж сама.

Примітка: Дистанційно-векторні протоколи зазвичай, здійснюю техніку, відому як розщеплення обрію. Розщеплення обрію запобігає надсиланню інформації на той же інтерфейс, з якого вона була отримана. Наприклад, R2 не зможе надіслати оновлення по Serial 0/0/0, що містить мережу 10.1.0.0, тому що R2 довідався про цю мережу через Serial 0/0/0.



Network	Interface	Hop
10.1.0.0	Fa0/0	0
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/0	1
10.4.0.0	S0/0/0	2

Network	Interface	Hop
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/1	0
10.1.0.0	S0/0/0	1
10.4.0.0	S0/0/1	1

Network	Interface	Hop
10.3.0.0	S0/0/1	0
10.4.0.0	Fa0/0	0
10.2.0.0	S0/0/1	1
10.1.0.0	S0/0/1	2

Рис. 4.4 Результат наступного кола розсилки періодичних оновлень

4.2.4 Конвергенція

Кількість часу, що необхідний мережі для конвергенції прямо пропорційний розміру мережі.

Швидкість досягнення стану конвергенції залежить від того:

- Як швидко маршрутизатори поширюють зміни в топології в маршрутних оновленнях до своїх сусідів.
- Яка швидкість обчислення кращих маршрутів, з використання нової маршрутної інформації.

Мережа не цілком працездатна, поки не закінчилася конвергенція, тому мережні адміністратори віддають перевагу протоколам маршрутизації з більш коротким часом конвергенції.

4.3 Підтримка таблиці маршрутизації

4.3.1 Періодичні оновлення: RIP v1 і IGRP

Підтримка таблиці маршрутизації

Багато дистанційно-векторних протоколів використовують періодичні оновлення, щоб обмінятися маршрутною інформацією зі своїми сусідами й, щоб підтримувати актуальну маршрутну інформацію в таблиці маршрутизації. RIP і IGRP - приклади таких протоколів.

Маршрутизатори періодично відправляють таблицю маршрутизації сусідам. Термін «періодичні оновлення» означає той факт, що маршрутизатор відправляє всю таблицю маршрутизації своїм сусідам через певний інтервал. ***Для RIP, ці оновлення надсилають кожні 30 секунд широкомовно (255.255.255.255), незалежно від того, були або не було змін в топології.*** Цей 30-секундний інтервал - таймер оновлення маршруту, який також допомагає у відстеженні віку маршрутної інформації в таблиці маршрутизації.

Вік маршрутної інформації в таблиці маршрутизації освіжається щораз

при одержанні оновлення. Це дозволяє підтримувати таблицю маршрутизації при змінах топології. Зміни можуть відбуватися з кількох причин, у тому числі:

- Відмова каналу.
- Введення нового каналу.
- Відмова маршрутизатора.
- Зміна параметрів каналу.

Таймери RIP

Окрім таймера оновлення, IOS реалізує три додаткові таймери для RIP:

- Invalid
- Flush
- Holddown

Invalid Timer – Таймер дійсності маршруту. Якщо оновлення не було отримано, щоб обновити існуючий маршрут, за замовчуванням після 180 секунд маршрут відзначається, як неправильний, шляхом встановлення метрики в 16. Маршрут зберігається в таблиці маршрутизації, поки не мине ***flush таймер***.

Flush Timer – Таймер скидання маршруту. За замовчуванням, flush таймер установлений на 240 секунд, що на 60 секунд більше, чим ***invalid таймер***. Коли flush таймер минає, маршрут видаляється з таблиці маршрутизації.

Holddown Timer – Таймер утримання інформації. Цей таймер стабілізує маршрутну інформацію й допомагає запобігти петлям маршрутизації протягом періодів, коли топологія конвергує з новою інформацією. Як тільки маршрут відзначений, як недосяжний, він повинен залишатися в стані утримання досить довго для того, щоб усі маршрутизатори в топології довідалися про недосяжність цієї мережі. За замовчуванням, holddown таймер установлений в 180 секунд.

Значення таймерів можна перевірити двома командами: ***show ip route*** і ***show ip protocols.*** Зверніть увагу на рис. 4.5, що команда ***show ip route*** для кожного маршруту, вивченого через RIP, показує час, який минув з моменту останнього оновлення, виражене в секундах.

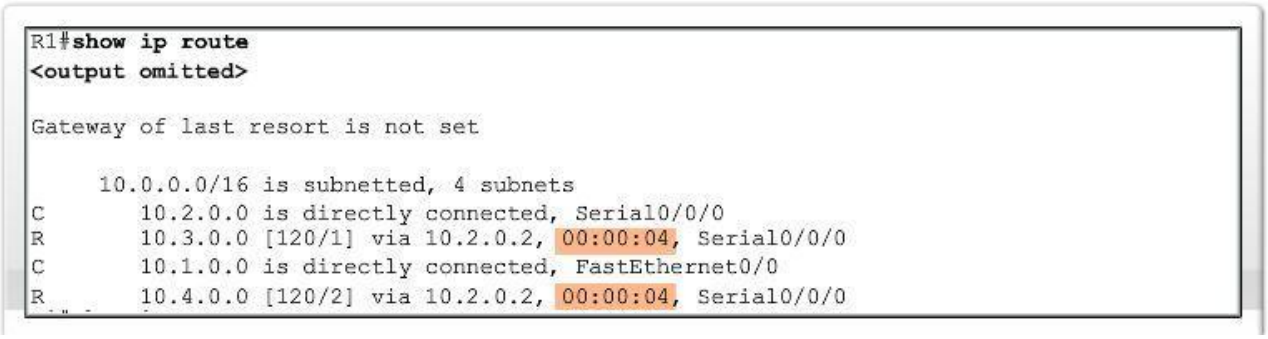


Рис. 4.5 Перевірка значень таймерів командою ***show ip route***

Така ж інформація може бути отримана при використанні команди ***show ip protocols*** (рис. 4.6) під заголовком ***Last Update.*** Також виводяться значення за замовчуванням таймерів ***invalid, holddown***, і ***flush.***

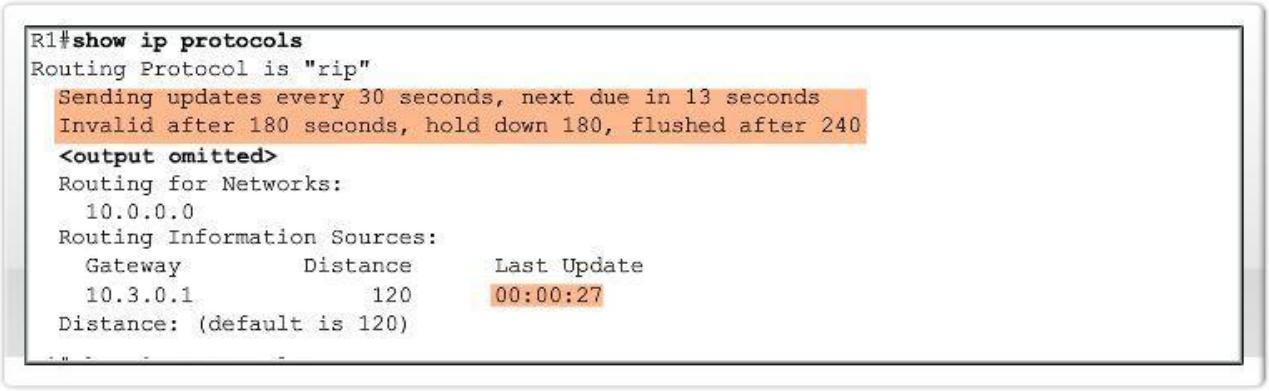


Рис. 4.6 Перевірка значень таймерів командою ***show ip protocols***

4.3.2 Обмежені оновлення (bounded updates): EIGRP

На відміну від інших дистанційно-векторних протоколів маршрутизації, EIGRP не посилає періодичні оновлення. Замість цього, EIGRP посилає обмежені оновлення про маршрут, коли міняється шлях або метрика для цього маршруту. Коли новий маршрут стає доступним або, коли маршрут потрібно вилучити, ***EIGRP посилає оновлення тільки про цю мережу замість повної таблиці***. Ця інформація відправляється тільки тим маршрутизаторам, яким вона потрібна.

EIGRP використовує оновлення, які є:

- Не періодичними, тому що вони не розсилаються через регулярні інтервали часу.
- Часткові оновлення посилають тільки, ***коли є зміни в топології, які впливають на маршрутну інформацію***.
- Обмежені (bounded), означає, що ***поширення часткових оновлень автоматично обмежується так, що тільки ті маршрутизатори, яким потрібна інформація, одержать оновлення.***

4.3.3 Миттєві (triggered) оновлення

Щоб підвищити швидкість конвергенції, при зміні топології RIP використовує миттєві оновлення. Миттєве оновлення - це оновлення таблиці маршрутизації, яке посилається негайно у відповідь на зміну маршрутизації. Миттєві оновлення не очікують, поки минуть таймери оновлення. Маршрутизатор, що виявив зміну топології, негайно відправляє повідомлення оновлення суміжним маршрутизаторам. Маршрутизатори, що одержують, у свою чергу, генерують миттєві оновлення, які повідомляють їхніх сусідів про зміни.

Миттєві оновлення посилають, коли відбувається одна з подій:

- ***Інтерфейс міняє свій стан («up» або «down»).***
- ***Маршрут перейшов у стан "недосяжний".***
- ***Маршрут встановлений у таблиці маршрутизації.***

Використання тільки миттєвих оновлень було б досить, якби можна було гарантувати, що хвиля оновлення досягне кожного маршрутизатора негайно.

Однак, є дві проблеми з миттєвими оновленнями:

- Пакети, що містять повідомлення оновлення, можуть бути скинуті або зіпсовані одним з каналів у мережі.
- Миттєві оновлення не відбуваються миттєво. Можливо, що маршрутизатор, який ще не одержав миттєве оновлення, розішле регулярне оновлення у невдалий час, змушуючи поганий маршрут бути знову встановленим у сусіда, який уже одержав миттєве оновлення.

4.3.4 Випадкова флуктуація (random jitter)

Проблеми із синхронізованими оновленнями

Коли безліч маршрутизаторів передають маршрутні оновлення в той самий час на LAN сегменті множинного доступу, пакети оновлення можуть попадати в колізії й це може бути причиною затримки або споживання занадто великої пропускної здатності.

Примітка: Колізії - це проблема, що виникає тільки з хабами, але не з комутаторами.

Посилка оновлень у той самий час відомий, як синхронізація оновлень. Синхронізація може стати проблемою з дистанційно-векторними протоколами маршрутизації, оскільки вони використовують періодичні оновлення. Тому що більшість таймерів маршрутизаторів синхронізуються, у мережі з'являється більше колізій оновлень і більше затримок. Спочатку, оновлення маршрутизаторів не будуть синхронізовані. Але через якийсь час, таймери всієї мережі стануть глобально синхронізованими.

Розв'язок

Щоб запобігти синхронізації оновлень між маршрутизаторами, ***Cisco IOS використовує випадкову змінну, названу RIP_JITTER, яка віднімає різну кількість часу від інтервалу оновлення для кожного маршрутизатора в мережі. Це випадкова флуктуація, або змінна кількість часу, у діапазоні від 0% до 15% зазначеного інтервалу оновлення. Таким чином, інтервал оновлення змінюється випадково в діапазоні від 25 до 30 секунд для заданого за замовченням 30-секундного інтервалу.***

4.4 Петлі маршрутизації

4.4.1 Визначення

Що таке петля маршрутизації?

Петля маршрутизації - це умова, при якій пакет безупинно передається в межах серії маршрутизаторів без досягнення коли-небудь їм мережі призначення. Петля маршрутизації може виникати, коли два або більше маршрутизаторів мають маршрутну інформацію, яка неправильно вказує, що шлях

до недосяжного адресата існує.

Петля може бути результатом:

- ***Неправильно сформованих статичних маршрутів***
- Неправильно сконфігурованого перерозподілу маршруту.(redistribution - це процес передачі маршрутної інформації від одного протоколу маршрутизації іншому).
- ***Непослідовні таблиці маршрутизації, не оновлені через повільну конвергенцію в мережі, що зазнала змін.***
- Маршрути неправильно сформовані або встановлені скинуті маршрути.

Дистанційно-векторні протоколи маршрутизації прості у своїй роботі. Їхня простота призводить до недоліків протоколу, наприклад до петель маршрутизації. Проблема петель маршрутизації практично не виникає в протоколах маршрутизації з урахуванням стану каналу, але при певних обставинах може з'являтися й там.

Примітка: Протокол IP має свій власний механізм, щоб запобігти нескінченному блуканню пакета по мережі. IP має поле Час життя (Time-to-live -TTL) і його значення зменшується на 1 на кожному маршрутизаторі. Якщо TTL рівно 0, маршрутизатор скидає пакет.

До чого призводять петлі маршрутизації?

Петля маршрутизації може мати руйнівний ефект для мережі, призводити до деградації її продуктивності або навіть до відмови мережі.

Петля маршрутизації може створити наступні умови:

- Пропускна здатність каналу буде використовуватися для циклічного трафікау взад-уперед між маршрутизаторами в петлі.
- Процесори маршрутизаторів будуть перевантажені в період зациклення.
- Процесори маршрутизаторів будуть обтяжені пересиланням некорисного пакета, що буде негативно впливати на конвергенцію мережі.
- Маршрутні оновлення можуть бути загублені або не бути обробленими у встановлений термін. Ці умови ввели б додаткові петлі маршрутизації, зробивши ситуацію ще гірше.
- Пакети можуть бути загублені в "чорних дірах."

Є цілий ряд механізмів, доступних, щоб виключити петлі маршрутизації, насамперед для дистанційно-векторних протоколів маршрутизації. Ці механізми включають:

- Визначення максимальної метрики для перешкоди прагненню до нескінченності.
- Holddown таймери.
- Розщеплення обрію.

- Отруєний маршрут або poison reverse.
- Миттєві (Triggered) оновлення

4.4.2 Проблема: лічімо до нескінченності

Прагнення до нескінченності - умова, яка існує, коли неправильні маршрутні оновлення збільшують метричне значення до "нескінченності" для мережі, яка більше не доступна.

4.4.3 Встановлення максимуму

Щоб в остаточному підсумку зупинити збільшення метрики, "нескінченність" визначається встановленням максимального метричного значення. Наприклад, ***RIP визначає нескінченність як 16переходів - показник "недосяжності мережі".*** Як тільки показник досягає 16, маршрутизатори відзначають маршрут, як недосяжний.

4.4.4 Запобігання петель маршрутизації за допомогою таймерів утримання інформації (Holddown)

Дистанційно-векторні протоколи використовують миттєві оновлення, щоб прискорити процес конвергенції. Крім миттєвих оновлень, маршрутизатори, що використовують дистанційно-векторні протоколи також посилають періодичні оновлення. Давайте уявимо собі що дана мережа нестабільна. Інтерфейси скинуті за короткий час у стан «up», потім «down», потім знову «up». Маршрут коливається. Використовуючи миттєві оновлення, маршрутизатори можуть реагувати занадто швидко й випадково створити петлю маршрутизації. Петля маршрутизації також може бути створена періодичними оновленнями, які посилаються маршрутизаторами в період нестабільності. Holddown таймери запобігають створенню петель маршрутизації через ці умови. Вони також допомагають запобігати умові прагнення до нескінченності.

Holddown таймери використовуються, щоб запобігти регулярним оновленням для недоречного оновлення маршруту, який, можливо пропав. Holddown таймери інструктують маршрутизатори, щоб утримувати будь-які зміни, які впливають на маршрути протягом зазначеного періоду часу. Якщо маршрут ідентифікований як зниклий або можливо зниклий, будь-яка інша інформація для цього маршруту, що містить той же статус, або гірше, має ігноруватися впродовж визначеної кількості часу (holddown period). Це означає, що маршрутизатори залишать маршрут, відзначений, як недосяжний у цьому стані протягом періоду часу, який є досить довгим, щоб були поширені таблиці маршрутизації із самою актуальною інформацією.

Holddown Таймери працюють у такий спосіб:

1. Маршрутизатор одержує оновлення від сусіда, що мережа, яка раніше була доступна, зараз більше недосяжна.

2. Маршрутизатор відзначає мережу, як таку, що можливо пропала й запускає holddown таймер.
3. Якщо оновлення з кращою метрикою для цієї мережі отримане від будь-якого сусіднього маршрутизатора протягом holddown періоду, мережа відновлюється й holddown таймер (таймер утримання інформації) видаляється.
4. Якщо оновлення від будь-якого іншого сусіда отримане протягом holddown періоду з такою ж або ще гіршою метрикою для цієї мережі, це оновлення ігнорується. ***Це дає більше часу для того, щоб інформація про зміни топології поширилася по всій мережі.***
5. Маршрутизатори все ще направляють пакети до мереж призначення, які позначені , що як можливо зниклі. Це дозволяє маршрутизатору подолати будь-які проблеми, пов'язані з нестійким забезпеченням зв'язку. Якщо мережа призначення дійсно недоступна й пакети переслані - чорна діра створюється й існує, поки таймер утримання не минає.

4.4.5 Правило розщепленого обрію (split horizon rule)

Інший метод, що перешкоджає появі петель маршрутизації, викликаних повільною конвергенцією дистанційно-векторних протоколів маршрутизації, - розщеплення обрію. Правило розщепленого обрію свідчить, що маршрутизатор не повинен анонсувати мережу через інтерфейс, по якому це оновлення прибуло.

Примітка: Розщеплення обрію може бути відключене адміністратором. За певних умов це виправдано, аби досягти правильної маршрутизації.

4.4.6 Розщеплений горизонт з отруєнням маршруту

Отруєння маршруту (Route Poisoning)

Це ще один метод, який використовується дистанційно-векторними протоколами маршрутизації для запобігання петлям маршрутизації. Отруєння маршруту використовується, аби помітити маршрут, як недосяжний в маршрутних оновленнях, які посилаються іншим маршрутизаторам. Недосяжність інтерпретується як метрика, встановлена в максимальне значення. Для RIP, отруєний маршрут має метрику 16.

Отруєння маршруту підвищує швидкість процесу конвергенції, оскільки інформація про мережу поширюється швидше.

Розщеплений обрій з отруєним реверсом (Split Horizon with Poison Reverse)

Отруєний реверс може комбінуватися з технікою розщепленого обрію. Правило розщепленого обрію з отруєним реверсом: коли пересилаються оновлення по конкретному інтерфейсу, позначити будь-які мережі, які були вивчені від цього інтерфейсу, як недосяжні.

Концепція розщепленого обрію з отруєним реверсом: явно повідомити маршрутизатор, що маршрутом потрібно нехтувати, краще, ніж, не повідомляти про маршрут взагалі. Отруєний реверс, по суті замінює правило розщепленого обрію.

Примітка: Правило розщепленого обрію використовується за замовченням. Проте розщеплений обрій з отруєним реверсом, може не бути присутнім за замовченням у всіх реалізаціях IOS.

4.4.7 IP і TTL

Час життя (Time to Live – TTL) - це 8-бітове поле в IP заголовку, яке обмежує число переходів, які пакет може пройти по мережі перед тим, як він буде скинутий. Завдання поля TTL - уникати ситуації, коли пакет, який не може бути доставлений адресатові, поширюється по мережі нескінченно. 8-розрядне поле TTL встановлюється пристроєм, що відправляє пакет. TTL зменшується кожним маршрутизатором на маршруті до його адресата. Якщо поле TTL досягає нуля до того, як пакет прибуде до свого адресата, пакет скидається і маршрутизатор посилає ICMP повідомлення про помилку назад, до джерела пакету.

4.5 Дистанційно-векторні протоколи маршрутизації сьогодні

4.5.1 RIP і EIGRP

Для дистанційно-векторних протоколів у нас - лише дві альтернативи: RIP або EIGRP. Рішення, про те, який з протоколів використовувати в даній ситуації визначається цілим рядом чинників:

- Розмір мережі.
- Сумісність моделей маршрутизаторів.
- Вимоги до знань адміністратора.

RIP

За ці роки, RIP еволюціонував від класового протоколу маршрутизації (RIPv1) до безкласового протоколу (RIPv2). RIPv2 - стандартизований протокол маршрутизації, який працює в змішаному середовищі маршрутизаторів від різних виробників. Маршрутизатори від різних виробників, можуть спільно використовувати RIP. Це - один з протоколів маршрутизації, що найлегше конфігуруються, це робить його хорошим вибором для маленьких мереж. Проте, RIPv2 все ще має обмеження. Як RIPv1, так і RIPv2 мають метрику маршруту, яка ґрунтується лише на числі переходів і має обмеження в 15 переходів.

Особливості RIP:

- Підтримка розщепленого обрію і розщепленого обрію з отруєним реверсом для запобігання циклам.
- Здібність до балансування навантаження, аж до шести шляхів рівної вартості (за замовченням використовується чотири шляхи).

RIPv2 представив наступні удосконалення RIPv1:

- Включає маску підмережі в маршрутні оновлення, що робить його безкласовим протоколом маршрутизації.
- Має аутентифікаційний механізм, аби забезпечити безпеку оновлень таблиці маршрутизації.
- Підтримує маски змінної довжини (VLSM).
- Використовує групові адреси замість широкомовних.
- Підтримує ручну сумаризацію маршруту.

EIGRP

Вдосконалений IGRP (EIGRP) розвивався з IGRP, іншого дистанційно-векторного протоколу. EIGRP - безкласовий, дистанційно-векторний протокол маршрутизації з можливостями, що притаманні протоколам з урахуванням стану каналу. Проте, на відміну від RIP або OSPF, EIGRP є приватним протоколом, Cisco, що розвивається, і працює лише на маршрутизаторах Cisco.

Особливості EIGRP :

- Миттєві (Triggered) оновлення (EIGRP не використовує періодичні оновлення).
- Використання топологічної таблиці для підтримки всіх маршрутів, отриманих від сусідів, а не лише кращих шляхів.
- Встановлення суміжності з сусідніми маршрутизаторами, з використанням EIGRP hello протоколу.
- Підтримка VLSM і ручної сумаризації маршруту. Це дозволяє EIGRP створювати ієрархічно структуровані великі мережі.

Переваги EIGRP:

- Хоча маршрути поширюються у формі дистанційно-векторній, метрика заснована на мінімальній пропускній спроможності і кумулятивній затримці замість числа переходів.
- Швидка конвергенція завдяки використанню Diffusing Update Algorithm (DUAL) алгоритму при розрахунку маршруту. DUAL дозволяє додавати резервні маршрути в топологічну таблицю EIGRP, які використовуються в разі збою первинного маршруту. Оскільки це - локальна процедура, перемикання на резервний маршрут не залучає до цього процесу інші маршрутизатори.
- Обмежені оновлення (Bounded updates) означають, що EIGRP використовує менше пропускної спроможності, особливо у великих мережах з безліччю маршрутів.
- EIGRP підтримує безліч протоколів мережного рівня через модулі, залежні від протоколу (Protocol Dependent Modules – PDM), які включають підтримку IP, IPX, і AppleTalk.

4.6 Висновки

4.6.1 Резюме

Один із способів класифікації протоколів маршрутизації - за типом алгоритму, який вони використовують, аби визначити кращий шлях до мережі призначення. Протоколи маршрутизації можуть бути класифіковані, як дистанційно-векторні, протоколи з урахуванням стану каналу і вектора шляху. «Дистанційно-векторні» означає, що маршрути анонсуються як вектори відстані і напряму. Відстань визначена в термінах метрики, як наприклад число переходів, а напрям – це просто next-hop маршрутизатор або exit-інтерфейс.

До дистанційно-векторних протоколів відносяться:

RIPv1
RIPv2
IGRP
EIGRP

Маршрутизатори, які використовують дистанційно-векторні протоколи маршрутизації, визначають кращий шлях до віддалених мереж, базуючись на інформації, яку вони отримують від своїх сусідів. Дистанційно-векторні протоколи маршрутизації не мають карти топології, як протоколи з урахуванням стану каналу.

Виявлення мереж - важливий процес для будь-якого протоколу маршрутизації. Деякі дистанційно-векторні протоколи маршрутизації, як наприклад RIP, проходять через покроковий процес вивчення і поширення інформації своїм сусідам. Як тільки маршрути вивчені від одного сусіда, ця інформація передається іншим сусідам зі збільшенням метрики маршрутизації.

Протоколи маршрутизації також повинні підтримувати таблиці маршрутизації актуальними і точними. RIP обмінюється інформацією з таблиці маршрутизації зі своїми сусідами кожні 30 секунд. EIGRP, інший дистанційно-векторний протокол, не посилає періодичні оновлення і посилає лише «обмежені» оновлення, якщо є зміни в топології і лише до тих маршрутизаторів, яким ця інформація потрібна.

RIP також використовує таймери, аби визначити, коли сусідній маршрутизатор більше не є досяжним, або, коли деякі з маршрутизаторів, можливо, не мають поточної маршрутної інформації. Це буває зазвичай, якщо мережа ще не сконвергована з урахуванням недавніх змін в топології. Дистанційно-векторні протоколи маршрутизації також використовують миттєві (triggered) оновлення, аби прискорити конвергенцію.

Недолік дистанційно-векторних протоколів маршрутизації, - схильність до формування петель маршрутизації. Петлі маршрутизації можуть з'являтися, коли мережа знаходиться в неконвергованому стані. Дистанційно-векторні протоколи маршрутизації використовують holddown таймери, аби перешкодити маршрутизатору використовувати інший маршрут до недавно зниклої мережі,

поки всі маршрутизатори ще не мали досить часу дізнатися про цю зміну в топології.

Розщеплений обрій і розщеплений обрій з отруєним реверсом також використовуються маршрутизаторами, аби перешкоджати появі петель маршрутизації. Правило розщепленого обрію свідчить, що маршрутизатор не повинен ніколи анонсувати маршрут через той інтерфейс, з якого він дізнався про цей маршрут. Розщеплений обрій з отруєним реверсом означає, що краще явно заявляти, що даний маршрутизатор не має маршруту до мережі, отруюючи маршрут метрикою, що означає недосяжність маршруту.

Дистанційно-векторні протоколи маршрутизації дуже популярні у багатьох адміністраторів, оскільки вони зазвичай прості для розуміння і реалізації.

4.6.2 Питання для самоперевірки

1. Коротко поясніть основні принципи роботи RIP і IGRP.
2. Пояснить поняття конвергенції і чому конвергенція така важлива?
3. Які 4 основних таймера використовує RIP? Яке призначення і тривалість кожного з них?
4. Які 5 технік використовують дистанційно-векторні протоколи маршрутизації для запобігання появі петель маршрутизації?

4.6.3 Матеріали для самостійного поглибленого вивчення теми

Розуміння дистанційно-векторного алгоритму не є складним. Існує безліч книг і online ресурсів, які демонструють, як алгоритми, такі як Беллмана-Форда, використовуються в комп'ютерних мережах. Ось деякі з цих джерел:

- Interconnections, Bridges, Routers, Switches, and Internetworking Protocols, by Radia Perlman
- Cisco IP Routing, by Alex Zinin
- Routing the Internet, by Christian Huitema

Тема 5. RIP v1 Routing Information Protocol – протокол маршрутної інформації

Ви навчитеся:

- Описувати функції, характеристики і роботу протоколу RIPv1.
- Конфігурувати пристрої для використання з RIPv1.
- Перевіряти правильність роботи RIPv1.
- Описувати як RIPv1 виконує автоматичну суммаризацію.
- Конфігурувати, перевіряти і шукати несправності поширення маршруту за умовчанням в мережі з RIPv1.
- Використовувати техніку, що рекомендується, для вирішення проблем, пов'язаних з RIPv1.

За ці роки протоколи маршрутизації еволюціонували, аби відповідати потребам складних мереж, що збільшуються. Перший протокол, який використовувався, був Routing Information Protocol (RIP). RIP до цих пір популярний через свою простоту.

Розуміння RIP є важливим для вивчення мереж по двох причинах. По-перше, RIP все ще використовується сьогодні. Ви можете зіткнутися з мережною реалізацією, яка досить велика, аби потребувати протоколу маршрутизації, але досить проста, аби ефективно використовувати RIP. По-друге, знайомство з багатьма фундаментальними поняттями RIP допоможе вам порівняти RIP з іншими протоколами. Розуміння того, як працює RIP, зробить вивчення інших протоколів маршрутизації простішим.

5.1 RIPv1: дистанційно-векторний, класовий протокол маршрутизації

5.1.1 Історія і перспективи

Історія RIP

RIP - найстаріший дистанційно-векторний протокол маршрутизації. Хоча в RIP відчувається недолік витонченості більш сучасних протоколів, його простота і широке використання - застава довговічності. Форма RIP для IPv6, названа RIPng (наступне покоління), зараз також доступна.

RIP еволюціонував від старішого протоколу, Xerox, що розвивався, який називався Gateway Information Protocol (GWINFO). З розвитком Xerox Network System (XNS), GWINFO розвивається в RIP. Пізніше він отримав популярність, тому що був реалізований в програмному забезпеченні Berkeley Software Distribution (BSD) як демон, названий демоном маршрутизації ("route-dee"). Різні інші виробники зробили свої власні реалізації RIP, що злегка розрізняються. Розуміючи потребу в стандартизації протоколу, Чарльз Хендрікс написав RFC 1058 в 1988 році, в якому він задокументував існуючий протокол і конкретизував деякі удосконалення. З того часу, RIP був покращений - RIPv2 в 1994 р. і RIPng в 1997г.

5.1.2 Характеристики RIPv1 і формат повідомлення

Характеристики RIP

- ***RIP - дистанційно-векторний протокол маршрутизації.***
- ***RIP використовує лише число переходів у якості метрики для вибору шляху.***
- ***Маршрути, що анонсуються, з числом переходів, більшим, ніж 15 недосяжні.***
- ***Повідомлення розсилаються широкомовно кожні 30 секунд.***

Інкапсуляція повідомлення RIPv1 показана на рис. 5.1. Дані RIP повідомлення інкапсульовані в сегмент UDP. Як порт відправника, так і порт одержувача встановлені в 520. IP заголовок і заголовок data link рівня додають широкомовні адреси призначення перед тим, як повідомлення розсилається нас всі інтерфейси, сконфігуровані з RIP.

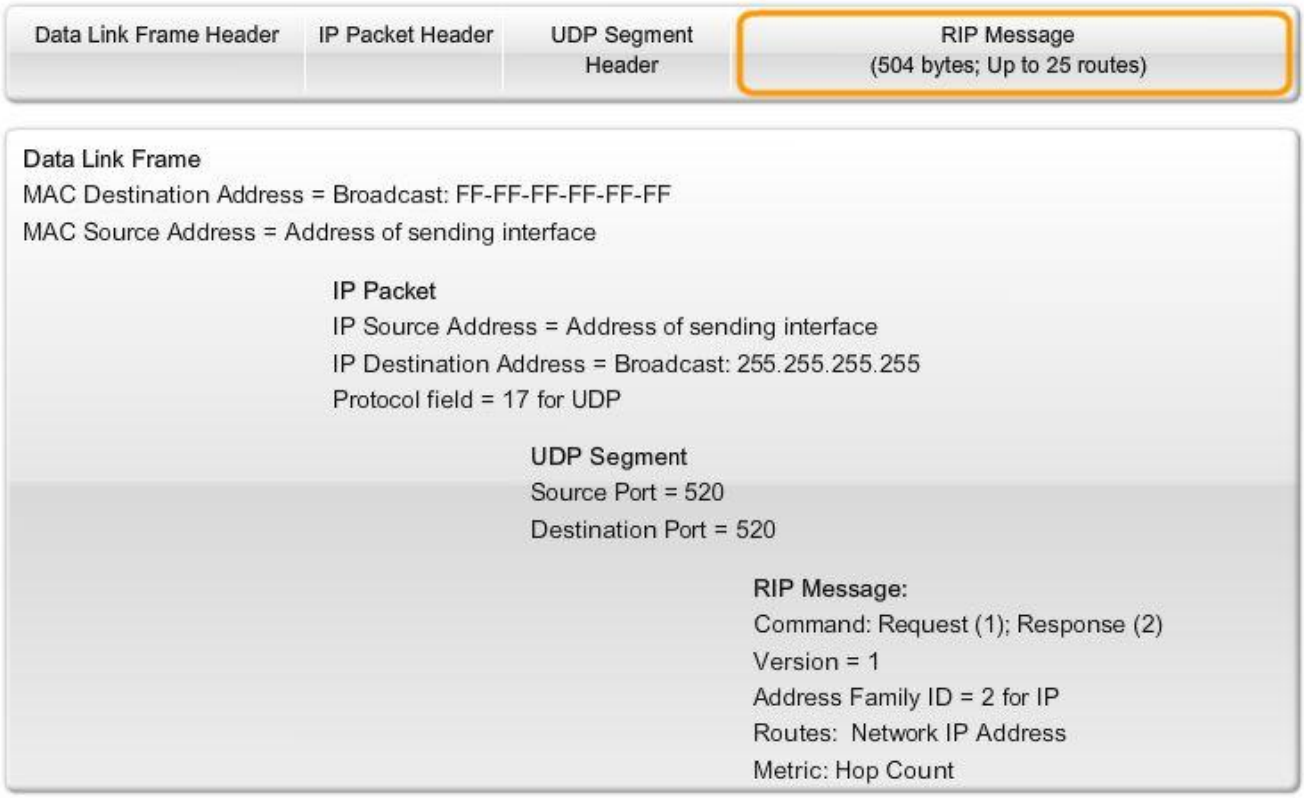


Рис. 5.1 Інкапсуляція повідомлення RIPv1

Формат повідомлення RIP: RIP заголовок

Розглянемо поля чотири-байтового заголовка RIP (рис. 5.2). Поле Command конкретизує тип повідомлення. Поле Version встановлене в 1 для версії RIPv1. Третє поле має бути нульовим, аби забезпечити майбутній розвиток протоколу.

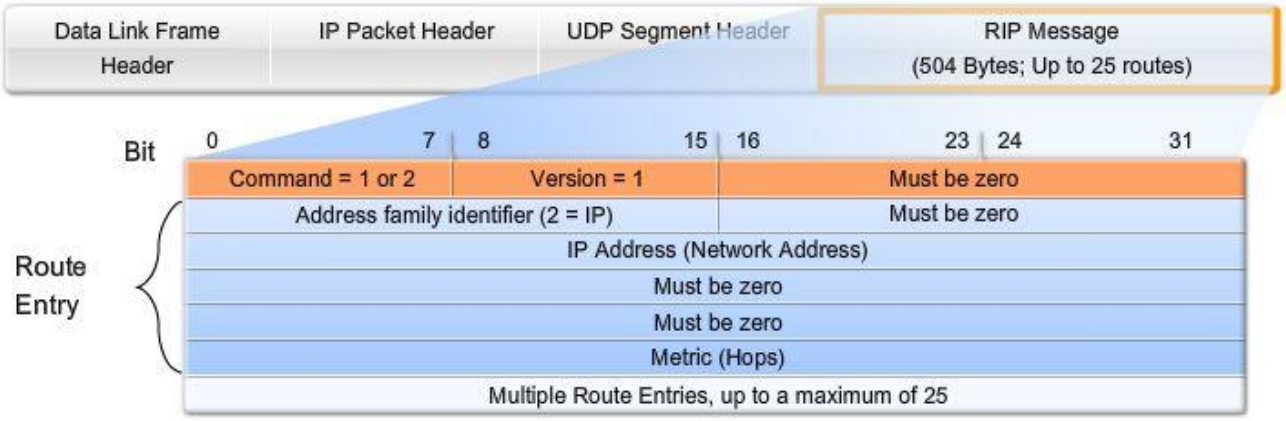


Рис. 5.2 Формат RIP повідомлення

Формат повідомлення RIP: маршрутні записи

Маршрутний запис включає три поля: Address family identifier ідентифікатор сімейства адрес (встановлено в 2 для IP, якби маршрутизатор запитав всю таблицю маршрутизації, тоді встановлюється в 0), адреса IP, і Metric - метрика. Такий маршрутний запис представляє один маршрут призначення з асоційованою метрикою. Одне оновлення RIP може містити аж до 25 маршрутних записів. Максимальний розмір датаграми складає 504 байти, без врахування IP і UDP заголовків.

Чому так багато полів встановлені в нуль?

RIP розроблявся до IP і використовувався для інших мережних протоколів (подібних XNS). BSD також зробив свій вплив. Спочатку, додатковий простір був доданий з наміром підтримки великих адресних просторів в майбутньому. RIPv2 сьогодні використовує більшість з цих порожніх полів.

5.1.3 RIP v1: Робота

Процес запита/відгука RIP

RIP використовує повідомлення двох типів, вказані в полі Command: повідомлення запиту і повідомлення відгуку.

Кожен сконфігурований для RIP інтерфейс посилає повідомлення запиту при старті, запрошуючи, аби всі RIP-сусіди надіслали йому свої повні таблиці маршрутизації. Повідомлення відповіді посилається назад всіма RIP-сусідами. Коли маршрутизатор, отримує відповіді, він оцінює кожен маршрутний запис. Якщо запис маршруту новий, одержуючий маршрутизатор встановлює маршрут в таблиці маршрутизації. Якщо маршрут вже знаходиться в таблиці, існуючий маршрут замінюється, якщо новий має менше число переходів. Маршрутизатор, який стартував, посилає миттєве (triggered) оновлення на всі дозволені для RIP інтерфейси. Це повідомлення містить його власну таблицю маршрутизації, так що RIP-сусіди будуть проінформовані про будь-які нові маршрути.

Класи IP адрес і класова маршрутизація

IP адреси, призначені для хостів, спочатку ділилися на 3 класи: A, B, C. Кожному класу була призначена задана за замовчанням маска підмережі, як показано на рис. 5.3.

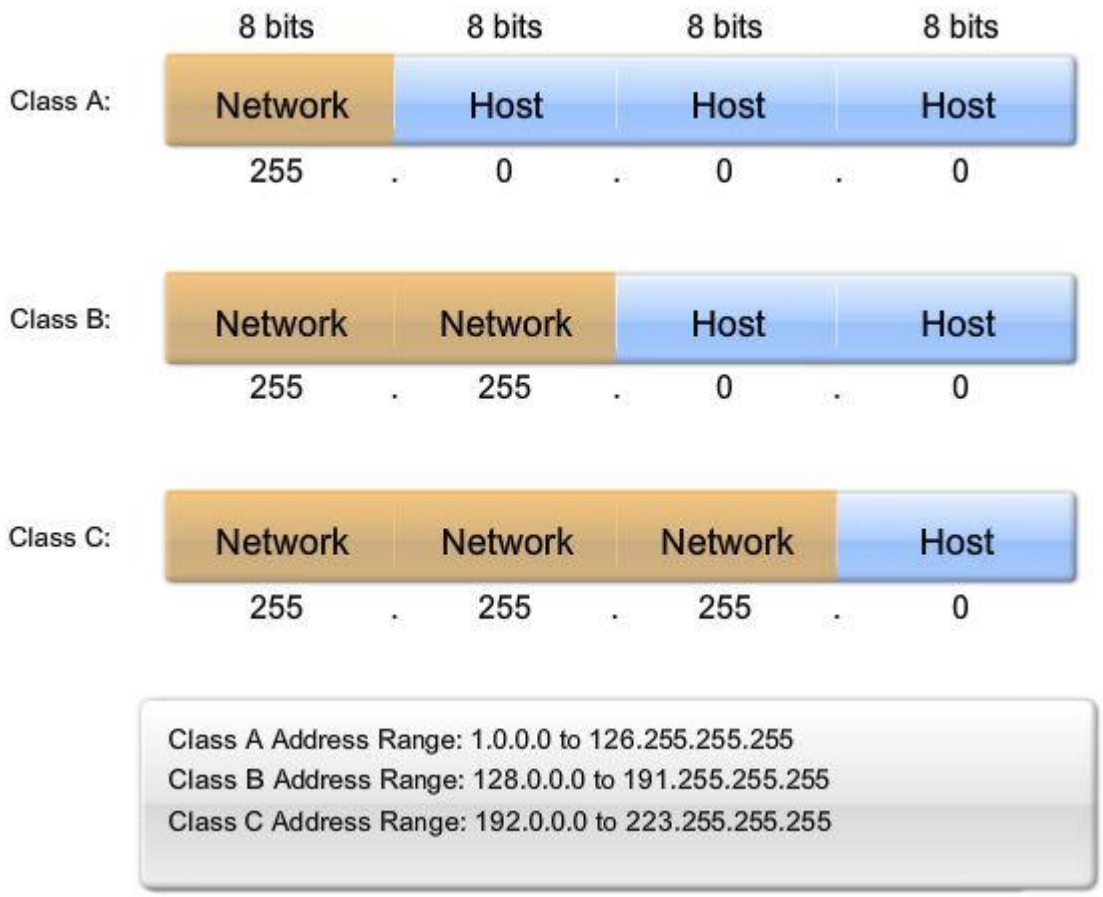


Рис. 5.3 Маски підмережі за замовчанням для різних класів IP-адрес

RIP - класовий протокол маршрутизації. RIPv1 не надсилає інформацію про маску підмережі в оновленні. Тому, маршрутизатор або використовує маску підмережі, сформовану на локальному інтерфейсі, або застосовує задану за замовчанням маску підмережі, ґрунтуючись на класі адреси. Завдяки цьому обмеженню, мережі RIPv1 не можуть бути несуміжними і вони не можуть підтримувати VLSM.

5.1.4 Адміністративна відстань

Адміністративна відстань (AD) - це міра надійності (або перевага) джерела маршруту. RIP має задану за замовчанням адміністративну відстань 120. При порівнянні з іншими протоколами маршрутизації внутрішнього шлюзу, RIP - протокол маршрутизації, якому найменше віддається перевага. IS-IS, OSPF, IGRP, і EIGRP всі мають нижчі значення AD за замовчанням.

Пам'ятаєте, що ви можете перевірити адміністративну відстань, використовуючи команду ***show ip route*** (рис.5.4) або ***show ip protocols*** (рис. 5.5).

```
R3#show ip route
Codes: C - connected, S - static, I - IGRP, 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

R    192.168.1.0/24 [120/1] via 192.168.6.2, 00:00:05, Serial0/0/0
R    192.168.2.0/24 [120/1] via 192.168.6.2, 00:00:05, Serial0/0/0
                    [120/1] via 192.168.4.2, 00:00:05, Serial0/0/1
R    192.168.3.0/24 [120/1] via 192.168.4.2, 00:00:05, Serial0/0/1
C    192.168.4.0/24 is directly connected, Serial0/0/1
C    192.168.5.0/24 is directly connected, FastEthernet0/0
C    192.168.6.0/24 is directly connected, Serial0/0/0
```

Рис. 5.4 Перевірка адміністративної відстані командою show ip route

```
R3#show ip protocols
Routing Protocol is "rip"
  <output omitted>
  Redistributing: rip
  Default version control: send version 1, receive any version
    Interface             Send  Recv  Triggered RIP  Key-chain
    FastEthernet0/0       1     1 2
    Serial0/0/0           1     1 2
    Serial0/0/1           1     1 2
  Automatic network summarization is in effect
  Routing for Networks:
    192.168.4.0
    192.168.5.0
    192.168.6.0
  Routing Information Sources:
    Gateway         Distance      Last Update
    192.168.6.2          120      00:00:10
    192.168.4.2          120      00:00:18
  Distance: (default is 120)
```

Рис. 5.5 Перевірка адміністративної відстані командою show ip protocols

5.2 Основи конфігурації RIP v1

5.2.1 Основи конфігурації RIP v1

На рис. 5.6 топологія, яка складається з трьох маршрутизаторів. Використовується 5 IP-адрес класу C (рис. 5.7).

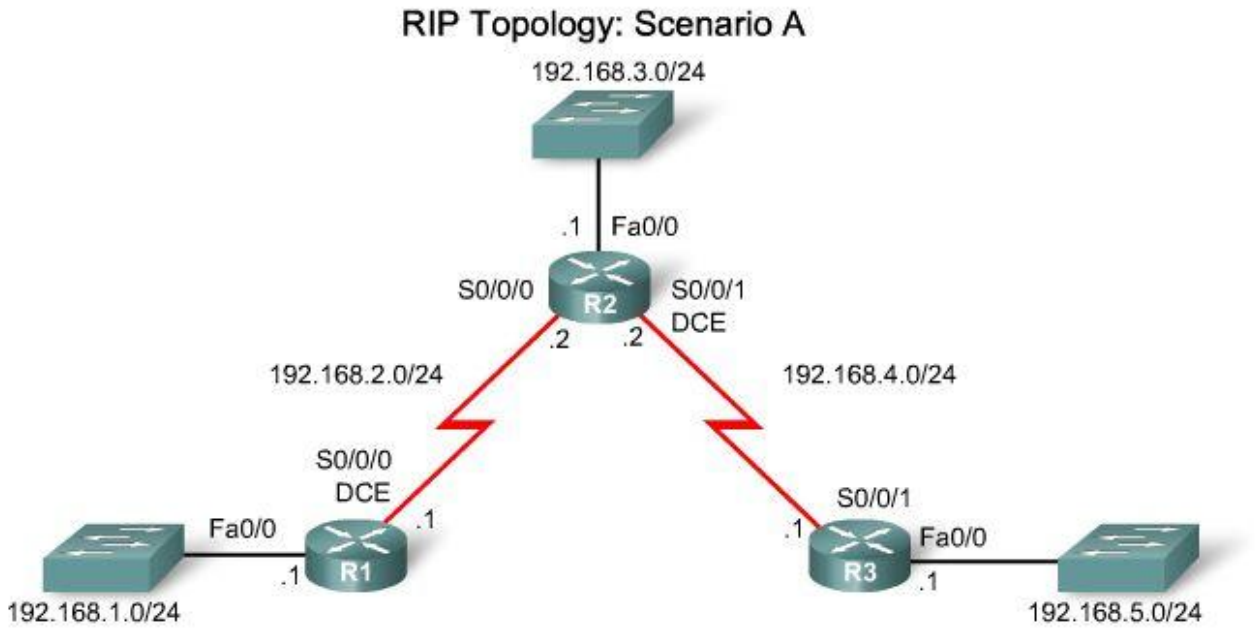


Рис. 5.6 RIP топологія: сценарій A

Device	Interface	IP Address	Subnet Mask
R1	Fa0/0	192.168.1.1	255.255.255.0
	S0/0/0	192.168.2.1	255.255.255.0
R2	Fa0/0	192.168.3.1	255.255.255.0
	S0/0/0	192.168.2.2	255.255.255.0
	S0/0/1	192.168.4.2	255.255.255.0
R3	Fa0/0	192.168.5.1	255.255.255.0
	S0/0/1	192.168.4.1	255.255.255.0

Рис. 5.7 Адресна таблиця: сценарій A

5.2.2 Запуск RIP : Команда route rip

Аби запустити протокол динамічної маршрутизації, увійдіть до режиму глобальної конфігурації.

Аби увійти до режиму конфігурації RIP, введіть router rip. Зверніть увагу, що рядок підказки при цьому зміниться:

R1(config-router)# ***router rip***
R1(config-router)#

Ця команда безпосередньо не запускає процес RIP. Замість цього, вона забезпечує доступ до конфігурації параметрів протоколу маршрутизації. Жодні оновлення при цьому не посилаються.

Якщо вам потрібно видалити RIP з пристрою, використовуйте команду ***no***

router rip. Ця команда зупиняє процес RIP і видаляє всі існуючі конфігурації RIP.

5.2.3 Визначення мереж

Шляхом входу в режим конфігурації RIP маршрутизації, ми вказуємо маршрутизатору, що потрібно виконувати RIP. Але маршрутизатору все ще потрібно знати, які локальні інтерфейси він має використовувати для комунікації з іншими маршрутизаторами, і які локально підключені мережі він повинен анонсувати цим маршрутизаторам. Аби дозволити RIP маршрутизацію для мережі, використовуйте команду ***network*** в режимі конфігурації маршрутизації і введіть класову мережну адресу для кожної безпосередньо підключеної мережі.

*Router(config-router)#**network** directly-connected-classful-network-address*

Команда network:

- Дозволяє RIP на всіх інтерфейсах, які належать вказаній мережі. Інтерфейси, що належать до цієї мережі, як посилатимуть, так і отримуватимуть оновлення RIP.
- Анонсує вказану мережу в RIP оновленнях маршрутизації, які відправляє іншим маршрутизаторам кожні 30 секунд.

Примітка: Якщо ви вводите адресу підмережі, IOS автоматично перетворює її на класову мережну адресу. Наприклад, якщо ви вводите команду network 192.168.1.32, маршрутизатор перетворить її на network 192.168.1.0.

Що трапляється, якщо ви вводите адресу підмережі або IP адресу інтерфейсу замість класової мережевої адреси при конфігурації RIP?

*R3(config) #**router rip***
*R3(config-router) #**network 192.168.4.0***
*R3(config-router) #**network 192.168.5.1***

IOS не надає повідомлення про помилку. Замість цього, IOS виправляє введення і вводить класову мережну адресу.

R3#show running-config
!
router rip
network 192.168.4.0
network 192.168.5.0
!

На рис. 5.8 командою network с конфігуровані на всіх трьох маршрутизаторах з топології 5.6 всі безпосередньо підключені мережі.

```
R1(config)#router rip
R1(config-router)#network 192.168.1.0
R1(config-router)#network 192.168.2.0
```

```
R2(config)#router rip
R2(config-router)#network 192.168.2.0
R2(config-router)#network 192.168.3.0
R2(config-router)#network 192.168.4.0
```

```
R3(config)#router rip
R3(config-router)#network 192.168.4.0
R3(config-router)#network 192.168.5.0
```

Рис. 5.8 Конфігурація RIP

5.3 Перевірка і пошук несправностей

5.3.1 Перевірка RIP: show ip route

Потужні команди для пошуку несправностей

Аби перевірити і знайти несправності маршрутизації, спочатку використовують ***show ip route*** і ***show ip protocols***. Якщо ви не можете ізолювати проблему, використовуючи ці дві команди, то використовуйте ***debug ip rip***, аби поглянути, що відбувається.

Пам'ятаєте, перед тим, як ви конфігуруєте будь-яку маршрутизацію - статичну або динамічну – потрібне впевнитися, що всі необхідні інтерфейси знаходяться в стані "up" і "up" за допомогою команди ***show ip interface brief.***

Команда ***show ip route*** перевіряє, що маршрути, отримані від RIP-сусідів, встановлені в таблиці маршрутизації. Літера R вказує маршрути RIP. Якщо маршрутизація правильно сформована на всіх маршрутизаторах, і конвергенція пройшла, команда show ip route покаже, що кожен маршрутизатор має повну таблицю маршрутизації, з маршрутом до кожної мережі в топології.

Як видно на рис. 5.6, в топології 5 мереж. На рис. 5.9, на прикладі маршрутизатора R1 видно, що кожен маршрутизатор має в таблиці маршрутизації список п'яти мереж.

```
R1#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
(**output omitted**)

Gateway of last resort is not set

R    192.168.4.0/24 [120/1] via 192.168.2.2, 00:00:02, Serial0/0/0
R    192.168.5.0/24 [120/2] via 192.168.2.2, 00:00:02, Serial0/0/0
C    192.168.1.0/24 is directly connected, FastEthernet0/0
C    192.168.2.0/24 is directly connected, Serial0/0/0
R    192.168.3.0/24 [120/1] via 192.168.2.2, 00:00:02, Serial0/0/0
```

Рис. 5.9 Таблиця маршрутизації для маршрутизатора R1

Інтерпретація результатів команди show ip route

Розглянемо один маршрут, вивчений маршрутизатором R1 через RIP.

R 192.168.5.0/24 [120/2] via 192.168.2.2, 00:00:23, Serial0/0/0

Код R показує, що RIP фактично працює на цьому маршрутизаторі. Даний маршрут вивчений через RIP. Якщо RIP не повністю сконфігурований, ви не побачите жодних RIP маршрутів.

Далі вказані віддалена мережна адреса і маска підмережі (192.168.5.0/24).

AD величина (120 для RIP) і метрика (2 переходи) вказані в дужках [120/2].

Вказана next-hop IP адреса маршрутизатора, що анонсував маршрут (R2 на 192.168.2.2) і скільки секунд пройшло з моменту останнього оновлення (00:00:23, в даному випадку).

Нарешті, exit інтерфейс, який використовуватиме цей маршрутизатор для трафіку, що призначається віддаленій мережі, - Serial 0/0/0.

5.3.2 Перевірка RIP: show ip protocols

Інтерпретація результатів команди show ip protocols

Якщо мережа відсутня в таблиці маршрутизації, перевірте конфігурацію маршрутизації, використовуючи ***show ip protocols***. Команда show ip protocols відображує протокол маршрутизації, який в даний час сконфігурований на маршрутизаторі. Можна також перевірити більшість параметрів RIP:

- Чи сформована RIP маршрутизація.
- Чи правильні інтерфейси посилають і отримують оновлення RIP.
- Маршрутизатор анонсує правильні мережі?
- RIP-сусіди посилають оновлення?

Ця команда також корисна при перевірці працездатності інших протоколів маршрутизації, наприклад EIGRP і OSPF.

На рис. 5.10 цифрами вказані наступні елементи:

1. Маршрутизація RIP конфігурована і працює на маршрутизаторі R2. Як мінімум один активний інтерфейс, і пов'язана з ним команда network необхідна, аби маршрутизація RIP стартувала.
2. Це - таймери, які показують, коли наступний круг оновлень буде посланий з цього маршрутизатора - 23 секунди від теперішнього моменту, в нашому прикладі.
3. Ця інформація має відношення до фільтрації оновлень і перерозподілу маршрутів, якщо вони сконфігуровані на даному маршрутизаторі.
4. Цей блок виводу містить інформацію, про те, яка версія RIP на даний час сконфігурована, і які інтерфейси беруть участь в оновленнях RIP.

5. Ця частина виводу показує, що маршрутизатор R2 на даний час виконує суммаризацію і за замовчанням використовуватиме аж до чотирьох маршрутів рівної вартості для балансування завантаження.
6. Перераховано класові мережі, які сформовані командою network. Ці мережі R2 включатиме в свої RIP-оновлення.
7. Тут перераховані сусіди RIP, як джерела маршрутної інформації.
   Gateway - це next-hop IP адреса сусіда, який посилає оновлення R2.
   Distance - це AD, яке використовує R2 для оновлень, посланих цим сусідом.
   Last Update – час в секундах з тих пір, як останнє оновлення було отримане від цього сусіда.

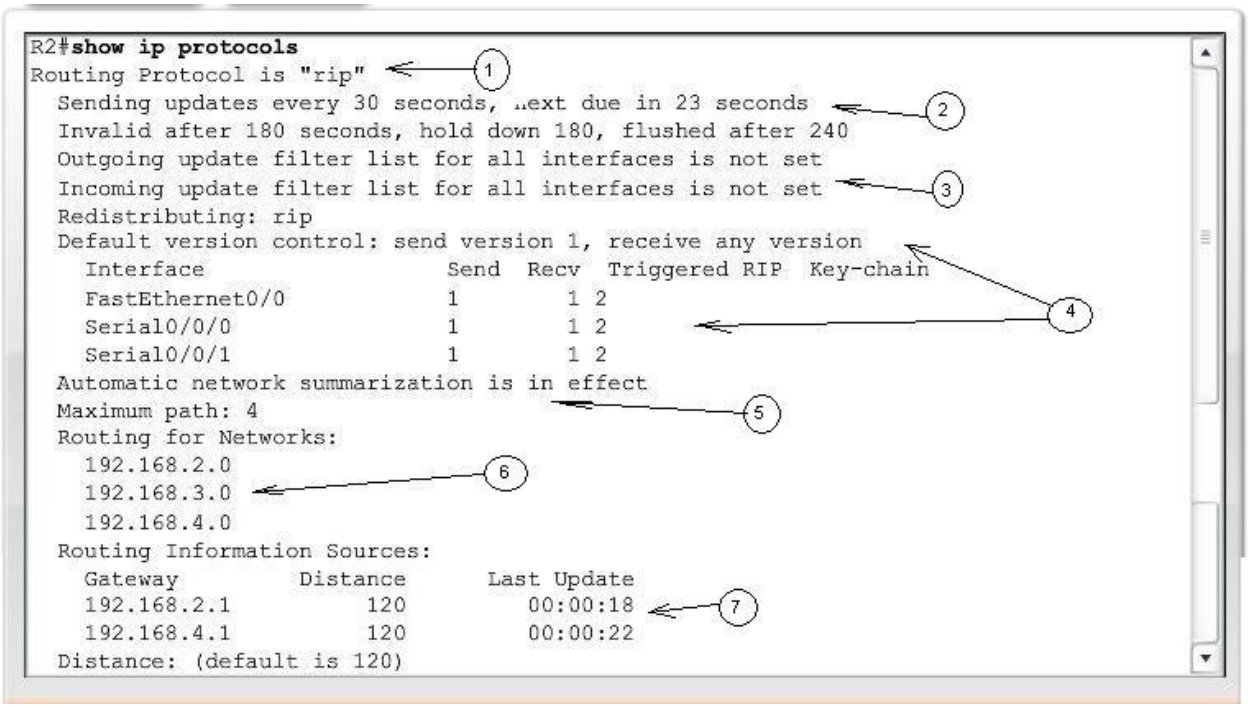


Рис. 5.10 Інтерпретація результатів команди show ip protocols

5.3.3 Перевірка RIP: debug ip rip

Інтерпретація результатів команди debug ip rip

Більшість помилок конфігурації RIP - це використання некоректної команди ***network***, або конфігурація несуміжних підмереж в класовому середовищі. Як показано на рис. 5.11, знайти проблеми, пов'язані з оновленнями RIP допоможе ефективна команда ***debug ip rip***. Ця команда відображує RIP- оновлення: як вони посилаються і надходять. Оскільки оновлення періодичні, вам доведеться почекати наступний круг оновлень аби побачити всі оновлення. На рис. 5.11 оновлення позначені цифрами:

1. Ми бачимо оновлення, що приходить від R1 на інтерфейс Serial 0/0/0. Зверніть увагу, що R1 відправляє лише один маршрут для мережі 192.168.1.0. Жодні інші маршрути не посилаються, інакше було б пору-

шено правило розщепленого обрію. R1 не дозволено анонсувати назад до R2 ті мережі, які R2 раніше відправив до R1.

2. Наступне оновлення, яке отримане, прийшло від R3. Знову, із-за правила розщепленого обрію, R3 посилає лише один маршрут - до 192.168.5.0 мережі.
3. R2 посилає свої власні оновлення. Спершу, R2 формує оновлення, аби послати його через інтерфейс FastEthernet0/0. Оновлення включає повну таблицю маршрутизації за винятком мережі 192.168.3.0, яка приєднана до FastEthernet0/0.
4. Потім, R2 формує оновлення, аби відправити його до R3. Три маршрути включаються. R2 не анонсує мережі 192.168.4.0 і 192.168.5.0 із-за правила розщепленого обрію.
5. Нарешті, R2 формує оновлення, аби відправити його R1. Три маршрути включаються. R2 не рекламує мережі 192.168.1.0 і 192.168.2.0 із-за правила розщепленого обрію.

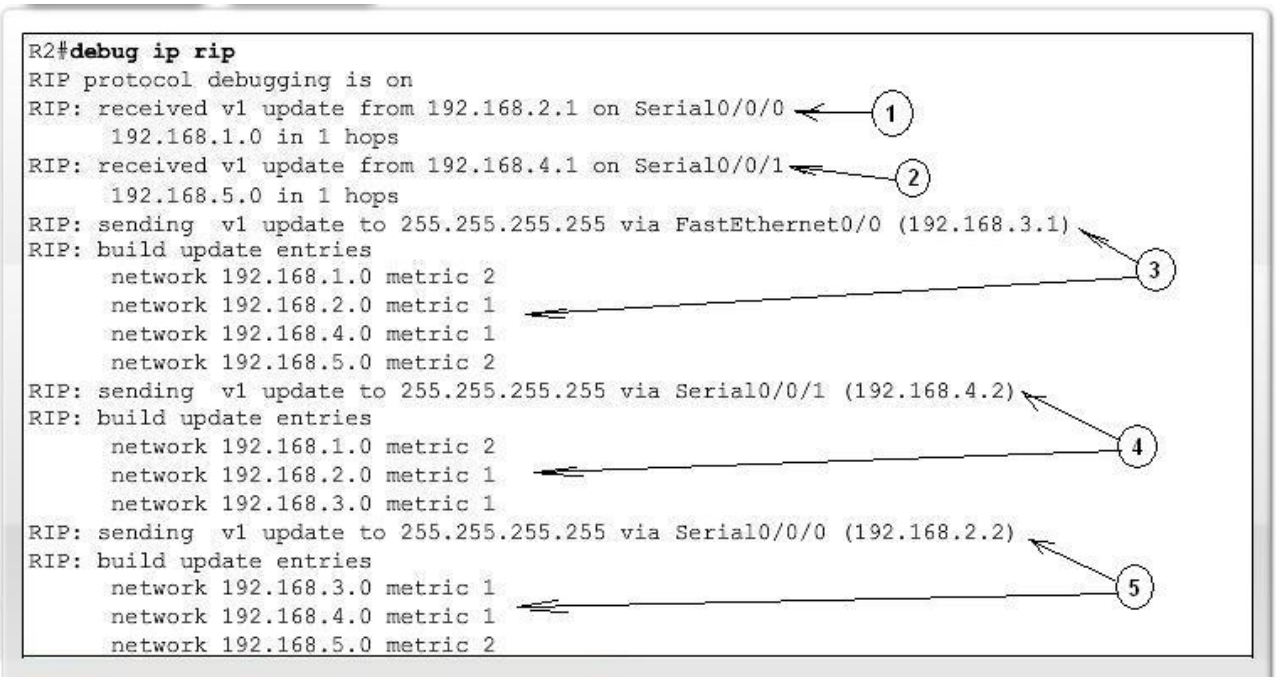


Рис. 5.11 Інтерпретація результатів команди debug ip rip

Примітка: Якщо почекати ще 30 секунд, буде повтор оновлень, тому що RIP посилає періодичні оновлення кожні 30 секунд.

Аби зупинити контроль оновлень RIP на R2, введіть команду ***no debug ip rip*** або ***undebug all***.

Розглядаючи цю інформацію, ми можемо перевірити, що маршрутизація RIP на R2 повністю працездатна. Але можна також побачити, як можна оптимізувати RIP маршрутизацію на R2. Чи є необхідність R2 посилати оновлення на FastEthernet0/0? Ми поглянемо в наступному розділі, як можна запобігти непотрібним оновленням.

5.3.4 Пасивні інтерфейси

Непотрібні оновлення RIP впливають на мережу

Як видно в попередньому прикладі, R2 посилає оновлення на FastEthernet0/0, хоча в цій локальній мережі не існує RIP пристроїв. R2 не знає про це і, в результаті, посилає оновлення кожні 30 секунд. Посилка непотрібних оновлень в LAN впливає на мережу трьома способами:

1. Пропускна спроможність займається, за рахунок пересилки непотрібних оновлень. Оскільки оновлення RIP широкомовні, комутатори перешлють оновлення на всі порти.
2. Всі пристрої LAN повинні обробити оновлення аж до Транспортного рівня, де одержуючий пристрій скине оновлення.
3. Анонсування оновлень в широкомовній мережі – це ризик безпеки. Оновлення RIP можуть бути перехоплені програмами – аналізаторами протоколів. Маршрутні оновлення можуть бути змінені і послані назад на маршрутизатор, що зіпсує таблицю маршрутизації фальшивою метрикою, і трафік прямуватиме неправильно.

Зупинка непотрібних оновлень RIP

Можна зупинити оновлення, видаливши 192.168.3.0 мережу з конфігурації, використовуючи команду ***no network 192.168.3.0***, проте тоді R2 не зможе анонсувати цю LAN як маршрут в оновленнях, відправлених до R1 і R3. Правильне рішення - використовувати команду ***passive-interface***, яка запобігає передачі маршрутних оновлень через вказаний інтерфейс маршрутизатора, але дозволяє цю мережу анонсувати іншим маршрутизаторам. Введіть команду ***passive-interface*** в режимі конфігурації маршрутизації.

*Router(config-router) #**passive-interface** interface-type interface-number*

Порівняйте рис. 5.12 з рис. 5.10. На рис. 5.12 зверніть увагу, що інтерфейс більше не відображається під Interface, але він є в новому розділі Passive Interface(s). Також зверніть увагу, що мережа 192.168.3.0 все ще відображається під Routing for Networks. Це означає, що ця мережа все ще включається як маршрутний запис в оновлення RIP, які вирушають до R1 і R3.

Всі протоколи маршрутизації підтримують команду passive-interface .

5.4 Автоматична суммаризація

5.4.1 Змінена топологія: сценарій B

Для обговорення автоматичної сумаризації, була модифікована RIP-топологія, як показано на рис. 5.13 і 5.14. Проведені наступні зміни:

Використовуються три класові мережі: 172.30.0.0/16
192.168.4.0/24
192.168.5.0/24

```
R2(config)#router rip
R2(config-router)#passive-interface FastEthernet 0/0
R2(config-router)#end
R2#show ip protocols
Routing Protocol is "rip"
    Sending updates every 30 seconds, next due in 14 seconds
    Invalid after 180 seconds, hold down 180, flushed after 240
    Outgoing update filter list for all interfaces is
    Incoming update filter list for all interfaces is
    Redistributing: rip
    Default version control: send version 1, receive any version
        Interface              Send  Recv  Triggered RIP  Key-chain
        Serial0/0/0            1     1 2
        Serial0/0/1            1     1 2
    Automatic network summarization is in effect
    Routing for Networks:
        192.168.2.0
        192.168.3.0
        192.168.4.0
    Passive Interface(s):
        FastEthernet0/0
    Routing Information Sources:
    Gateway         Distance      Last Update
       192.168.2.1          120      00:00:27
       192.168.4.1          120      00:00:23
Distance: (default is 120)
```

Рис. 5.12 Відміна непотрібних оновлень командою passive-interface

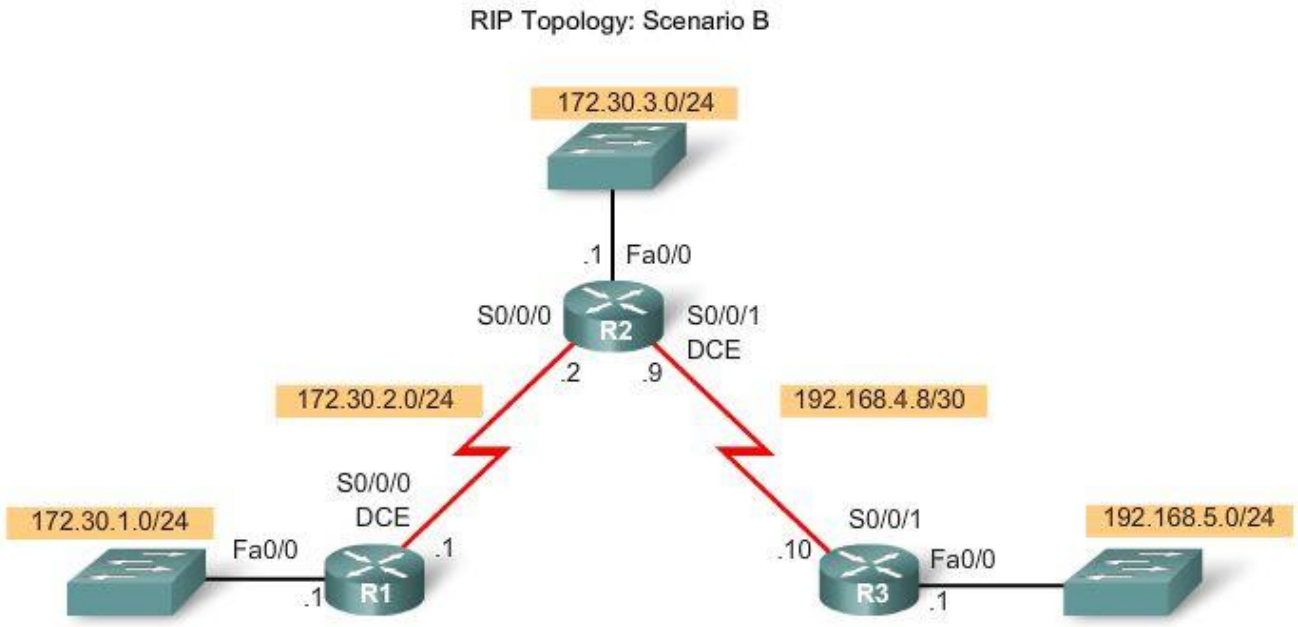


Рис. 5.13 RIP топологія: сценарій B

Мережа 172.30.0.0/16 поділена на три підмережі: 172.30.1.0/24
172.30.2.0/24
172.30.3.0/24

Мережа 192.168.4.0/24 поділена на підмережі з однією підмережею 192.168.4.8/30.

Device	Interface	IP Address	Subnet Mask
R1	Fa0/0	172.30.1.1	255.255.255.0
	S0/0/0	172.30.2.1	255.255.255.0
R2	Fa0/0	172.30.3.1	255.255.255.0
	S0/0/0	172.30.2.2	255.255.255.0
	S0/0/1	192.168.4.9	255.255.255.252
R3	Fa0/0	192.168.5.1	255.255.255.0
	S0/0/1	192.168.4.10	255.255.255.252

Рис. 5.14 Адресна таблиця: сценарій B

Розглянемо детально конфігурацію кожного маршрутизатора. Зверніть увагу, що команди no shutdown і clock rate виконувати не потрібно, оскільки вони вже були сконфігуровані в сценарії A. Проте, оскільки додаються нові мережі, процес RIP маршрутизації буде повністю видалений командою no router rip, перш ніж відновити його знову.

На рис. 5.15, зверніть увагу, що для R1 обидві підмережі було сконфігуровано командою network. Ця конфігурація технічно некоректна, оскільки RIPv1 посилає в своїх оновленнях класову мережеву адресу, і не посилає підмережі. Тому IOS міняє конфігурацію на класову, як показано на рис. 5.15 після виконання команди show run.

```
R1(config)#interface fa0/0
R1(config-if)#ip address 172.30.1.1 255.255.255.0
R1(config-if)#interface S0/0/0
R1(config-if)#ip address 172.30.2.1 255.255.255.0
R1(config-if)#no router rip
R1(config)#router rip
R1(config-router)#network 172.30.1.0
R1(config-router)#network 172.30.2.0
R1(config-router)#passive-interface FastEthernet 0/0
R1(config-router)#end
R1#show run
 (**output omitted**)

!
router rip
 passive-interface FastEthernet0/0
 network 172.30.0.0
!
```

Рис. 5.15 Конфігурація маршрутизатора R1

На рис. 5.16 конфігурація R2. Зверніть увагу, що підмережа 192.168.4.8 сконфігурована командою network. Знову, ця конфігурація технічно некоректна і IOS міняє її на 192.168.4.0 в поточній конфігурації.

```
R2(config)#interface S0/0/0
R2(config-if)#ip address 172.30.2.2 255.255.255.0
R2(config-if)#interface fa0/0
R2(config-if)#ip address 172.30.3.1 255.255.255.0
R2(config-if)#interface S0/0/1
R2(config-if)#ip address 192.168.4.9 255.255.255.252
R2(config-if)#no router rip
R2(config)#router rip
R2(config-router)#network 172.30.0.0
R2(config-router)#network 192.168.4.8
R2(config-router)#passive-interface FastEthernet 0/0
R2(config-router)#end
R2#show run
 (**output omitted**)
!
router rip
 passive-interface FastEthernet0/0
 network 172.30.0.0
 network 192.168.4.0
!
 (**output omitted**)
R2#
```

Рис. 5.16 Конфігурація маршрутизатора R2

Маршрутизація для R3 сконфігурована коректно (рис. 5.17). Поточна конфігурація збігається з тією, яка вводилася.

```
R3(config)#interface fa0/0
R3(config-if)#ip address 192.168.5.1 255.255.255.0
R3(config-if)#interface S0/0/1
R3(config-if)#ip address 192.168.4.10 255.255.255.252
R3(config-if)#no router rip
R3(config)#router rip
R3(config-router)#network 192.168.4.0
R3(config-router)#network 192.168.5.0
R3(config-router)#passive-interface FastEthernet 0/0
R3(config-router)#end
R3#show run
 (**output omitted**)
!
router rip
 passive-interface FastEthernet0/0
 network 192.168.4.0
 network 192.168.5.0
!
```

Рис. 5.17 Конфігурація маршрутизатора R3

Примітка: На оцінних і сертифікаційних іспитах, використання адреси підмережі замість класової мережної адреси в команді network вважається помилкою.

5.4.2 Граничні маршрутизатори і автоматична сумаризація

Як ви знаєте, RIP - класовий протокол маршрутизації, який автоматично підсумовує класові мережі через основні мережні кордони. На рис. 5.18 видно, що R2 має інтерфейси в більш ніж одній основній класовій мережі. Це робить

R2 граничним (boundary) маршрутизатором в RIP. Інтерфейси Serial 0/0/0 і FastEthernet 0/0 на R2, обоє знаходяться усередині 172.30.0.0 кордону. Serial 0/0/1 інтерфейс усередині 192.168.4.0 кордону.

Оскільки граничні маршрутизатори підсумовують RIP підмережі до головної класової мережі, оновлення для 172.30.1.0, 172.30.2.0 і 172.30.3.0 мереж автоматично підсумовуватимуться в 172.30.0.0 при відсиланні по інтерфейсу Serial 0/0/1 маршрутизатора R2.

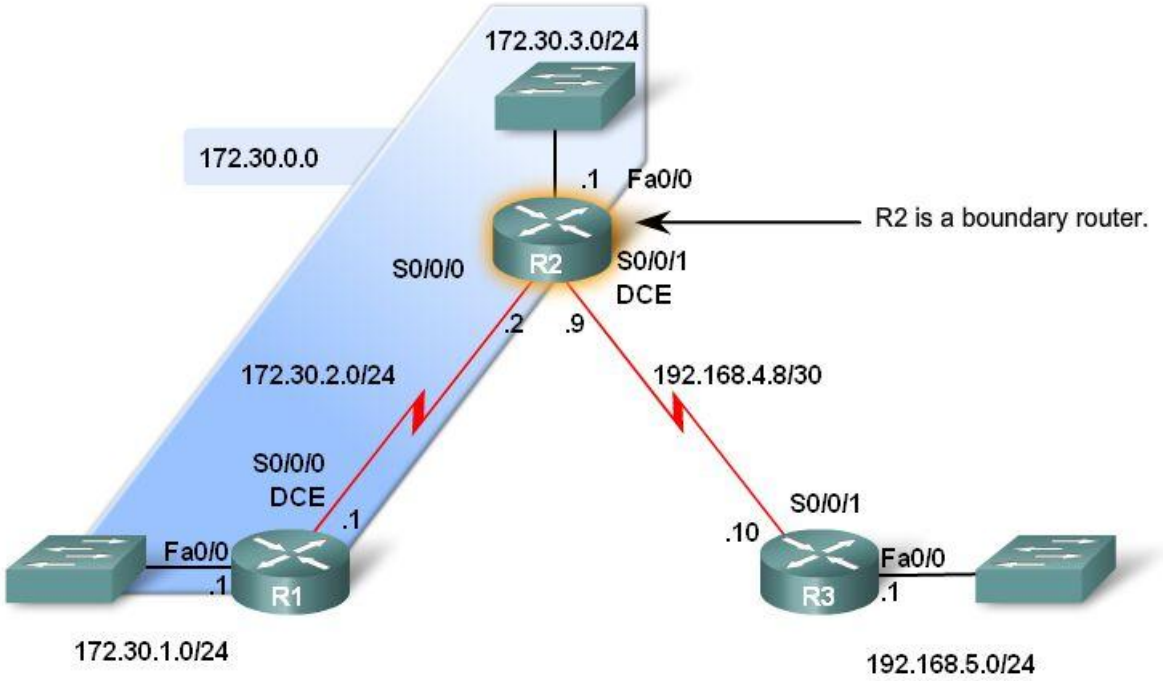


Рис. 5.18 Граничний маршрутизатор RIP

5.4.3 Обробка оновлень RIP

Правила обробки оновлень RIPv1

Наступне два правила управляють оновленнями RIPv1:

- Якщо маршрутне оновлення і інтерфейс, по якому воно отримане, належать одній і тій же класовій мережі, маска підмережі інтерфейсу застосовується до мережі, отриманої в маршрутному оновленні.
- Якщо маршрутне оновлення і інтерфейс, по якому воно отримане, належать різним класовим мережам, до мережі, отриманої в маршрутному оновленні застосовується класова маска.

Приклад обробки оновлення RIPv1

На рис. 5.19 R2 отримує оновлення від R1 і вносить мережу до таблиці маршрутизації. Як R2 визначає, що ця підмережа має маску підмережі /24 (255.255.255.0)? R2 знає це, тому що:

- R2 отримав цю інформацію на інтерфейс, який належить тій самій класовій мережі (172.30.0.0), що і оновлення, яке прийшло, 172.30.1.0.

- IP адреса, на якій R2 отримав повідомлення "172.30.1.0 in 1 hops", це інтерфейс Serial 0/0/0 з IP адресою 172.30.2.2 і маскою підмережі 255.255.255.0 (/24).
- R2 використовує свою власну маску підмережі на цьому інтерфейсі і застосовує її до всіх інших 172.30.0.0 підмереж, отриманих на цьому інтерфейсі, – в даному випадку, 172.30.1.0.
- Підмережа 172.30.1.0 /24 заноситься в таблицю маршрутизації.

```
R2#debug ip rip
RIP protocol debugging is on
RIP: received v1 update from 172.30.2.1 on Serial0/0/0
      172.30.1.0 in 1 hops
 (**output omitted**)

R2#undebug all
All possible debugging has been turned off
R2#show ip route
 <output omitted>

Gateway of last resort is not set

     172.30.0.0/24 is subnetted, 3 subnets
R       172.30.1.0 [120/1] via 172.30.2.1, 00:00:18, Serial0/0/0
C       172.30.2.0 is directly connected, Serial0/0/0
C       172.30.3.0 is directly connected, FastEthernet0/0
     192.168.4.0/30 is subnetted, 1 subnets
C       192.168.4.8 is directly connected, Serial0/0/1
R    192.168.5.0/24 [120/1] via 192.168.4.10, 00:00:16, Serial0/0/1
R2#
```

Рис. 5.19 Приклад обробки оновлення RIPv1

Маршрутизатори, на яких працює RIPv1, обмежені використанням однакової маски підмережі для всіх підмереж однієї і тієї ж класової мережі.

Безкласові протоколи маршрутизації, подібні RIPv2 дозволяють для однієї і тієї ж класової мережі використовувати різні маски для різних підмереж, відомі як маски змінної довжини (VLSM).

5.4.4 Надсилання оновлень RIP

Використання режиму відладки для спостереження за автоматичною сумаризацією

При відсиланні оновлення, граничний маршрутизатор R2 включатиме мережну адресу і метрику. Якщо маршрутний запис оновлення посилається в іншу основну мережу, то мережна адреса в маршрутному записі підсумовується до класової мережної адреси. Так R2 поступає для мереж 192.168.4.0 і 192.168.5.0. Він відправляє ці класові мережі R1 (рис. 5.20).

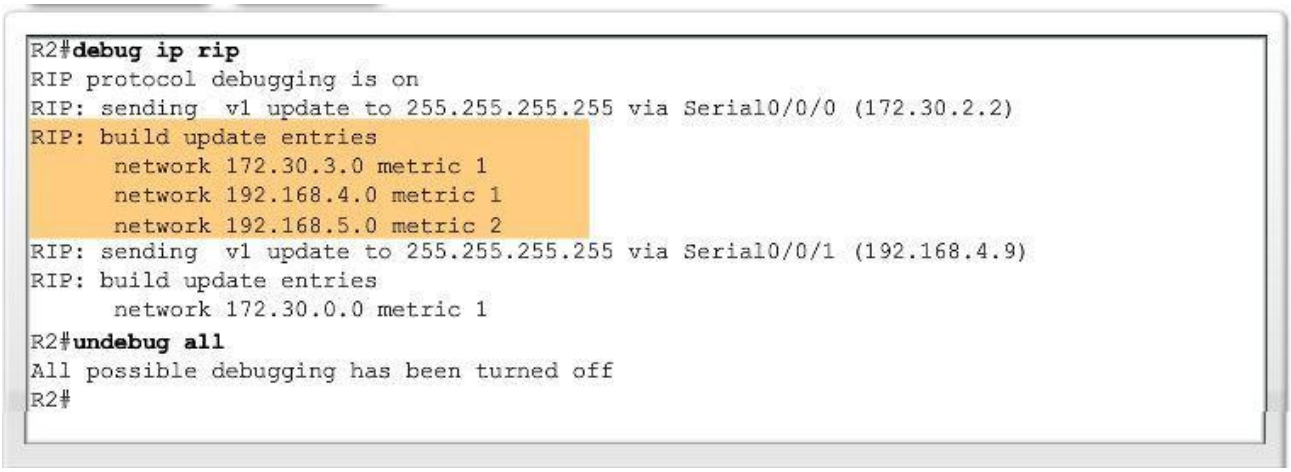


Рис. 5.20 Маршрути, які R2 надсилає до R1.

R2 також має маршрути до 172.30.1.0/24, 172.30.2.0/24 і 172.30.3.0/24 підмереж. У оновленнях R2 до R3 на Serial0/0/1, R2 посилає лише підсумовану класову мережну адресу 172.30.0.0 (рис. 5.21).

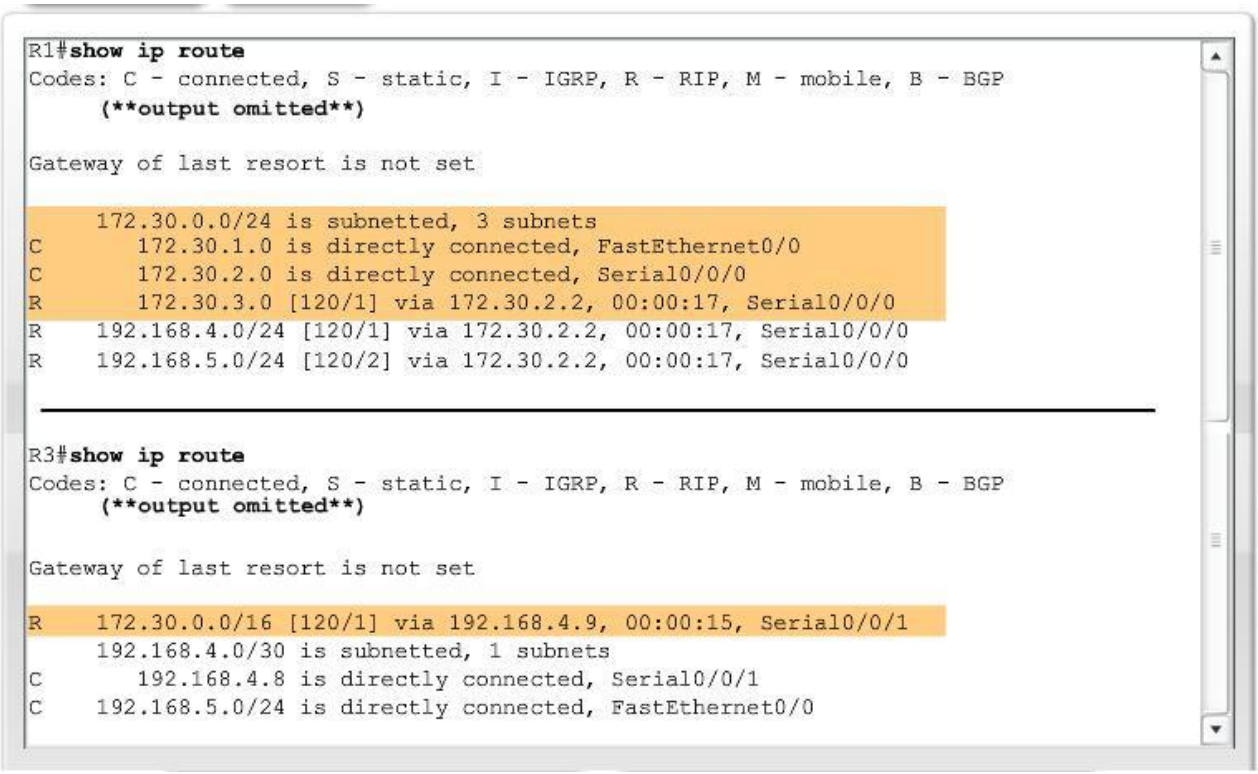


Рис. 5.21 Порівняння маршрутів до мережі 172.30.0.0 на R2 і R3

Якщо маршрутний запис в оновленні посилається в межах класової мережі, використовується маска підмережі вихідного інтерфейсу, аби визначити мережну адресу, яка анонсується. R2 відправляє підмережу 172.30.3.0 до R1, використовуючи маску підмережі на Serial0/0/0 (рис. 5.20).

R1 отримує оновлення 172.30.3.0 на інтерфейс Serial0/0/0, який має адресу інтерфейсу 172.30.2.1/24. Оскільки маршрутне оновлення і інтерфейс належать одній і тій самій класовій мережі, R1 застосовує маску інтерфейсу /24 до 172.30.3.0 маршруту.

Зверніть увагу на рис. 5.21, що R1 має три маршрути для 172.30.0.0 основної мережі, яка була розбита на підмережі /24 або 255.255.255.0. R3 має лише один маршрут до 172.30.0.0 мережі, і мережа не поділена на підмережі. R3 має класову мережу в своїй таблиці маршрутизації. Проте, було б помилкою передбачати, що R3 не має повного забезпечення зв'язку. R3 надішле будь-які пакети, що призначаються 172.30.1.0/24, 172.30.2.0/24, і 172.30.3.0/24 мережам до R2, тому що всі три мережі належать 172.30.0.0/16 і доступні через R2.

5.4.5 Переваги і недоліки автоматичної сумаризації

Переваги автоматичної сумаризації

Як видно на рис. 5.21, RIP автоматично сумаризує оновлення між класовими мережами. Оскільки оновлення 172.30.0.0 послано на інтерфейс Serial 0/0/1 в іншу класову мережу (192.168.4.0), RIP посилає лише одне оновлення для всієї класової мережі замість окремих для кожної з підмереж. Цей процес подібний до сумаризації декількох статичних маршрутів в єдиний статичний маршрут. Чому автоматична сумаризація має переваги?

- Оновлення меншого розміру використовують менше пропускної здатності між R2 і R3.
- R3 має єдиний маршрут до 172.30.0.0/16 мережі, не дивлячись на те, скільки в ній підмереж і як вона розбита на підмережі. Використання єдиного маршруту призводить до швидшого перегляду таблиці маршрутизації на R3.

Чи є недоліки в автоматичної сумаризації? Так, коли в топології присутні несуміжні мережі.

Недолік автоматичної сумаризації

Як видно на рис. 5.22 і 5.23, адресна схема змінена. Ця топологія буде використана для демонстрації основного недоліку класових протоколів маршрутизації, подібних RIPv1, - відсутність підтримки несуміжних мереж.

Класові протоколи маршрутизації не включають маску підмережі в маршрутні оновлення. Мережі автоматично підсумовуються через класові мережні кордони, і одержуючий оновлення маршрутизатор не в змозі визначити маску маршруту. Це пов'язано з тим, що одержуючий інтерфейс, можливо, має іншу маску, ніж маршрути, що анонсуються.

Зверніть увагу, що R1 і R3 обоє мають підмережі від 172.30.0.0/16 мережі, тоді як R2 таких підмереж не має. По суті, R1 і R3 - граничні маршрутизатори для 172.30.0.0/16, тому що вони розділені іншою основною мережею, 209.165.200.0/24. Це розділення створює несуміжну мережу, оскільки дві групи 172.30.0.0/24 підмережі розділено як мінімум одній іншою класовою мережею. 172.30.0.0/16 - несуміжна мережа.

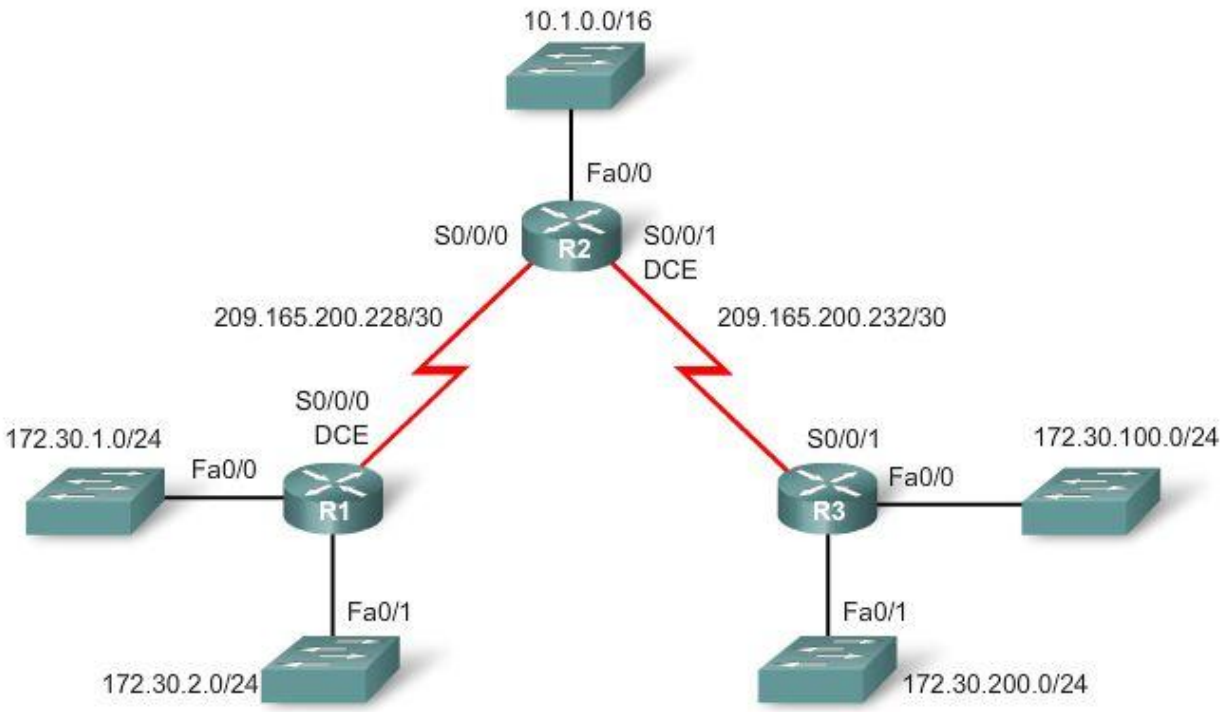


Рис. 5.22 Топологія: недоліки автоматичної сумаризації

Device	Interface	IP Address	Subnet Mask
R1	Fa0/0	172.30.1.1	255.255.255.0
	Fa0/1	172.30.2.1	255.255.255.0
	S0/0/0	209.165.200.229	255.255.255.252
R2	Fa0/0	10.1.0.1	255.255.0.0
	S0/0/0	209.165.200.230	255.255.255.252
	S0/0/1	209.165.200.233	255.255.255.252
R3	Fa0/0	172.30.100.1	255.255.255.0
	Fa0/0	172.30.200.1	255.255.255.0
	S0/0/1	209.165.200.234	255.255.255.252

Рис. 5.23 Адресна схема з несуміжними мережами

Несуміжні топології не конвергуються з RIPv1

Конфігурація RIPv1 на рис. 5.24 правильна, але неможливо визначити всі мережі в цій несуміжній топології. Аби зрозуміти, чому, пам'ятайте, що маршрутизатор анонсуватиме лише головні мережні адреси по інтерфейсах, які не належать маршруту, що анонсується. В результаті, R1 не анонсуватиме 172.30.1.0 або 172.30.2.0 до R2 через 209.165.200.0 мережу. R3 не анонсуватиме 172.30.100.0 або 172.30.200.0 до R2 через 209.165.200.0 мережу. Обоє і R1 і R3, анонсуватимуть 172.30.0.0 класову мережну адресу.

Що в результаті? Без включення маски підмережі в маршрутне оновлення, RIPv1 не може анонсувати повністю певну маршрутну інформацію, яка дозволить маршрутизаторам правильно маршрутизувати 172.30.0.0/24 підмережі.

```
R1(config)#router rip
R1(config-router)#network 172.30.0.0
R1(config-router)#network 209.165.200.0
```

```
R2(config)#router rip
R2(config-router)#network 10.0.0.0
R2(config-router)#network 209.165.200.0
```

```
R3(config)#router rip
R3(config-router)#network 172.30.0.0
R3(config-router)#network 209.165.200.0
```

Рис. 5.24 Конфігурація маршрутизаторів

Як видно на рис. 5.25, R1 не має жодних маршрутів до локальних мереж, підключених до R3. Аналогічно, R3 не матиме жодних маршрутів до локальних мереж, підключених до R1.

```
R1#show ip route
Codes: C - connected, S - static, I - IGRP, 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

R    10.0.0.0/8 [120/1] via 209.165.200.230, 00:00:26, Serial0/0/0
     172.30.0.0/24 is subnetted, 3 subnets
C       172.30.1.0 is directly connected, FastEthernet0/0
C       172.30.2.0 is directly connected, FastEthernet0/1
     209.165.200.0/30 is subnetted, 2 subnets
C       209.165.200.228 is directly connected, Serial0/0/0
R       209.165.200.232 [120/1] via 209.165.200.230, 00:00:26, Serial0/0/0
```

Рис. 5.25 Таблиця маршрутизації R1

R2 на рис. 5.26 має два шляхи рівної вартості до мережі 172.30.0.0. R2 проводитиме балансування навантаження трафіку, призначеного для будь-якої підмережі 172.30.0.0. Це означає, що R1 отримає половину трафіку і R3 отримає іншу половину трафіку, незалежно від того, чи знаходиться адресат в їх локальних мережах.

```
R2#show ip route
Codes: C - connected, S - static, I - IGRP, 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

     10.0.0.0/16 is subnetted, 1 subnets
C       10.1.0.0 is directly connected, FastEthernet0/0
R    172.30.0.0/16 [120/1] via 209.165.200.234, 00:00:14, Serial0/0/1
                   [120/1] via 209.165.200.229, 00:00:19, Serial0/0/0
     209.165.200.0/30 is subnetted, 2 subnets
C       209.165.200.228 is directly connected, Serial0/0/0
C       209.165.200.232 is directly connected, Serial0/0/1
```

Рис. 5.26 Таблиця маршрутизації R2

5.5 Висновки

5.5.1 Резюме

RIP (версія 1) - класовий, дистанційно-векторний протокол маршрутизації. RIPv1 був одним з перших протоколів маршрутизації, розробленим для маршрутизації IP пакетів. RIP використовує число переходів як метрику, показник 16 переходів має на увазі, що маршрут недосяжний. В результаті, RIP може використовуватися лише в мережах, де між будь-якими двома мережами не більше ніж п'ятнадцять маршрутизаторів.

Повідомлення RIP інкапсульовані в UDP сегмент, з вихідним портом і портом призначення 520. Маршрутизатори RIP відправляють свої повні таблиці маршрутизації сусідам кожні 30 секунд, за винятком тих маршрутів, які потрапляють під правило розщепленого обрію.

RIP запускається командою ***router rip*** в режимі глобальної конфігурації. Команда ***network*** використовується, аби вказати, які інтерфейси на маршрутизаторі будуть доступні для RIP, для чого вказується класова мережна адреса для кожної безпосередньо підключеної мережі. Команда network дозволяє інтерфейсу посилати і отримувати оновлення RIP, а також анонсувати цю мережу в оновленнях RIP до інших маршрутизаторів.

Команда ***debug ip rip*** може використовуватися, аби переглядати оновлення RIP, які посилає і отримує маршрутизатор. Аби запобігти надсиланню оновлень RIP через інтерфейс, такий як LAN, де немає інших маршрутизаторів, використовується команда ***passive-interface***.

RIP маршрути відображуються в таблиці маршрутизації з літерою R і мають адміністративну відстань 120. Задані за замовчанням маршрути поширюються в RIP, шляхом конфігурації статичного заданого за замовчанням маршруту і використанням команди RIP ***default-information originate*** .

RIPv1 автоматично підсумовує підмережі за їх класовою адресою при пересилці оновлення через інтерфейс, який належить іншій класовій мережі, ніж адреса підмережі маршруту. Оскільки RIPv1 – класовий протокол маршрутизації, маска підмережі не входить до складу оновлення. Коли маршрутизатор отримує оновлення RIPv1, RIP повинен визначити маску підмережі для маршруту. Якщо маршрут належить тій самій класовій мережі, що і оновлення, RIPv1 застосовує маску підмережі одержуючого інтерфейсу. Якщо маршрут належить іншій класовій мережі, ніж одержуючий інтерфейс, RIPv1 застосовує задану за замовчанням класову маску.

Команда ***show ip protocols*** може використовуватися для відображення інформації про будь-який протокол маршрутизації, працюючий на маршрутизаторі. Відносно RIP, ця команда відображує інформацію про таймери, стан автоматичної сумаризації, які мережі допущені на цьому маршрутизаторі для RIP, і іншу інформацію.

Оскільки RIPv1 - класовий протокол маршрутизації, він не підтримує несуміжні мережі або VLSM.

5.6.2 Питання для самопревірки

1. Які основні характеристики RIPv1?
2. Маршрутизатор HQ має з'єднання з трьома маршрутизаторами у філіях (BR1, BR2, BR3) і з'єднання з Інтернетом через ISP (рис. 5.27). Між HQ і маршрутизаторами філій сконфігурований протокол RIPv1. Запишіть команди, необхідні для конфігурації маршрутизації RIPv1 на R1.

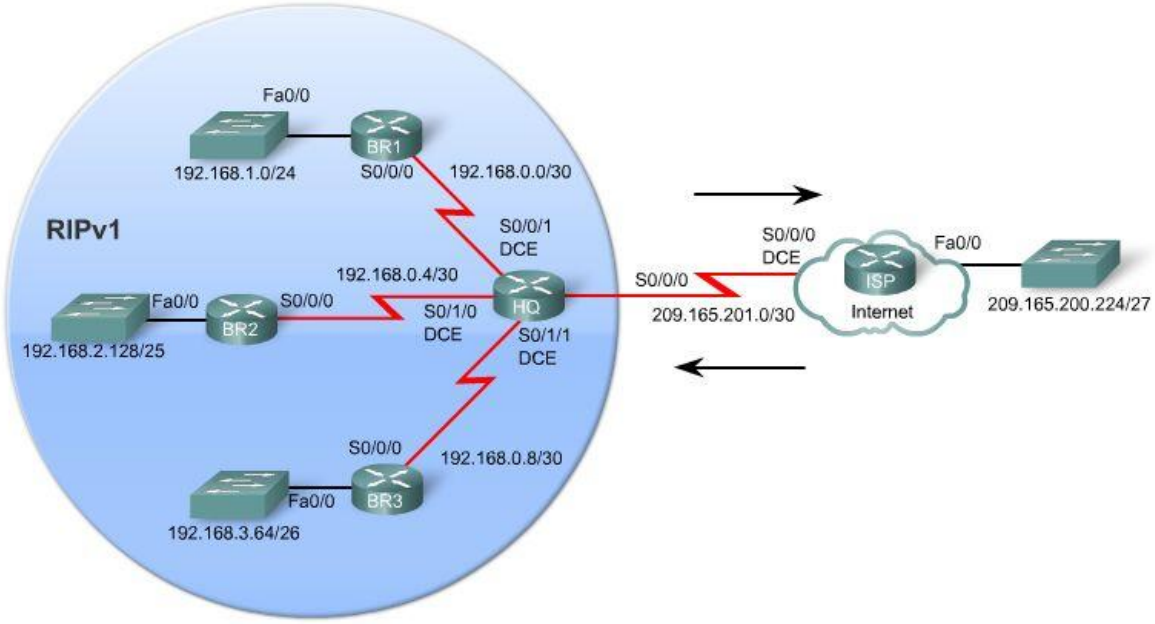


Рис. 5.27 Задана топологія

3. Запишіть три команди, які використовуються для перевірки і пошуку несправностей RIPv1?

4. Яка мета команди passive-interface? Наведіть приклад використання цієї команди на R1, включаючи командний рядок маршрутизатора для топології на рис. 5.27.
5. Що таке граничний (boundary) маршрутизатор в RIPv1?
6. Чому Ви не станете конфігурувати протокол динамічної маршрутизації для обміну маршрутною інформацією зі своїм ISP?
7. Запишіть повну конфігурацію маршрутизації для HQ, включаючи RIPv1, маршрут за замовчанням і поширення маршруту за замовчанням на маршрутизаторах філій (рис. 5.27).
8. Яка команда визначення статичного маршруту підсумує всі мережі, доступні через HQ (рис. 5.27)?

5.6.3 Матеріали для самостійного поглибленого вивчення теми

RFCs (Request for Comments) – це серія документів, які запропоновані IETF (Internet Engineering Task Force) як стандарти Інтернет або нові концепції. RFC 1058 – оригінальний RFC для RIP, написаний Чарльзом Хедріком.

RFC можуть бути доступні на ряді web-сайтів, включаючи www.ietf.org. Прочитайте всі частини RFC 1058. Багато інформації з цього документа Вам вже знайома, але зустрінеться і додаткова.

Тема 6. RIP v2

Ви навчитеся:

- Описувати обмеження RIPv1.
- Застосовувати основні команди конфігурації RIPv2 і оцінювати безкласові маршрутні оновлення RIPv2.
- Аналізувати інформацію, що виводиться маршрутизатором, аби побачити, що RIPv2 підтримує VLSM і CIDR.
- Ідентифікувати команди перевірки RIPv2.
- Конфігурувати, перевіряти і шукати несправності RIPv2.

RIP версії 2 (RIPv2) визначений в RFC 1723. Це - перший безкласовий протокол маршрутизації, який нами вивчається. Хоча RIPv2 - відповідний протокол маршрутизації для деяких середовищ, він поступається в популярності іншим протоколам маршрутизації, наприклад EIGRP, OSPF, і IS-IS, які пропонують більше можливостей і більш масштабовані.

В той же час, менш популярні, ніж інші протоколи маршрутизації, обидві версії RIP все ще актуальні в деяких ситуаціях. Хоча в RIP відчувається недолік можливостей багатьох новіших протоколів, його явна простота і поширення в багатьох операційних системах робить його ідеальним кандидатом для маленьких, гомогенних мереж, де необхідна підтримка безлічі виробників - особливо в середовищах Unix.

Оскільки необхідно розуміти RIPv2 - навіть якщо ви не використовуєте його - ця тема зосередиться на відмінностях між класовим протоколом маршрутизації (RIPv1) і безкласовим протоколом (RIPv2) більш, ніж на деталях RIPv2. Основне обмеження RIPv1 - він класовий протокол маршрутизації. Як відомо, класовий протокол маршрутизації не включає маску підмережі разом з мережною адресою в маршрутні оновлення, що може викликати проблеми з несуміжними підмережами або мережами, які використовують VLSM. Оскільки RIPv2 - безкласовий протокол маршрутизації, маски підмережі входять до складу маршрутних оновлень, роблячи RIPv2 більш сумісним з сучасними середовищами.

RIPv2 - фактично розширення можливостей RIPv1, а не повністю новий протокол. Деякі з цих розширених можливостей включають:

- ***Next-hop адреси входять в маршрутні оновлення.***
- ***Використання групових адрес при надсиланні оновлень.***
- ***Доступні аутентифікаційні можливості.***

Подібно RIPv1, RIPv2 - дистанційно-векторний протокол маршрутизації. Обидві версії RIP використовують наступні особливості і обмеження:

- ***Використання таймера утримання інформації (holddown) і інших таймерів для запобігання петлям маршрутизації.***
- ***Використання розщепленого обрію (split horizon) або розщепле-***

ного обрію з отруєним реверсом (split horizon with poison reverse) для запобігання петлям маршрутизації.

- ***Використання миттєвих (triggered) оновлень при змінах в топології для швидшої конвергенції.***
- ***Максимальна кількість переходів обмежена 15 переходами, з числом переходів 16 асоціюються недосяжні мережі.***

6.1 Обмеження RIP v1

6.1.1 Топологія для розгляду

На рис. 6.1 показана топологія, яка використовується в даній темі. R1 і R3 мають підмережі, які є частиною класової мережі 172.30.0.0/16 (класу B), – див. рис. 7.2. R1 і R3 з'єднані з R2, використовуючи підмережі класової мережі 209.165.200.0/24 (класу C). Ця топологія несуміжна і не конвергована, тому що 172.30.0.0/16 розділені мережею 209.165.200.0/24.

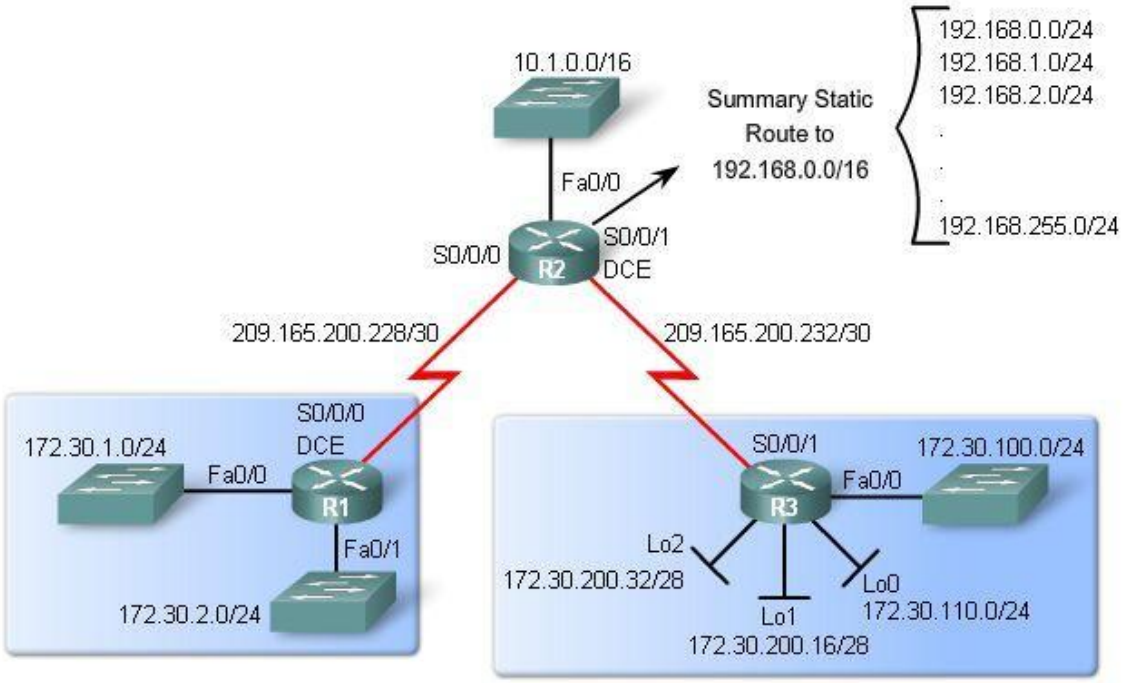


Рис. 6.1 Топологія мережі

Сумарний маршрут

Топологія на рис. 6.1 показує, що R2 має статичний сумарний маршрут до мережі 192.168.0.0/16.

Ми можемо включити інформацію про статичний маршрут в оновлення протоколу маршрутизації. Це називається перерозподілом маршруту (redistribution). Зрозуміло, що цей сумарний маршрут викличе проблеми з RIPv1, тому що 192.168.0.0/16 не є класовою адресою і включає всі /24 варіанти 192.168.0.0/16, як показано в топології.

Маршрутизатори R1 і R3 містять VLSM мережі і ділять адресний простір

172.30.0.0/16 класової мережі.

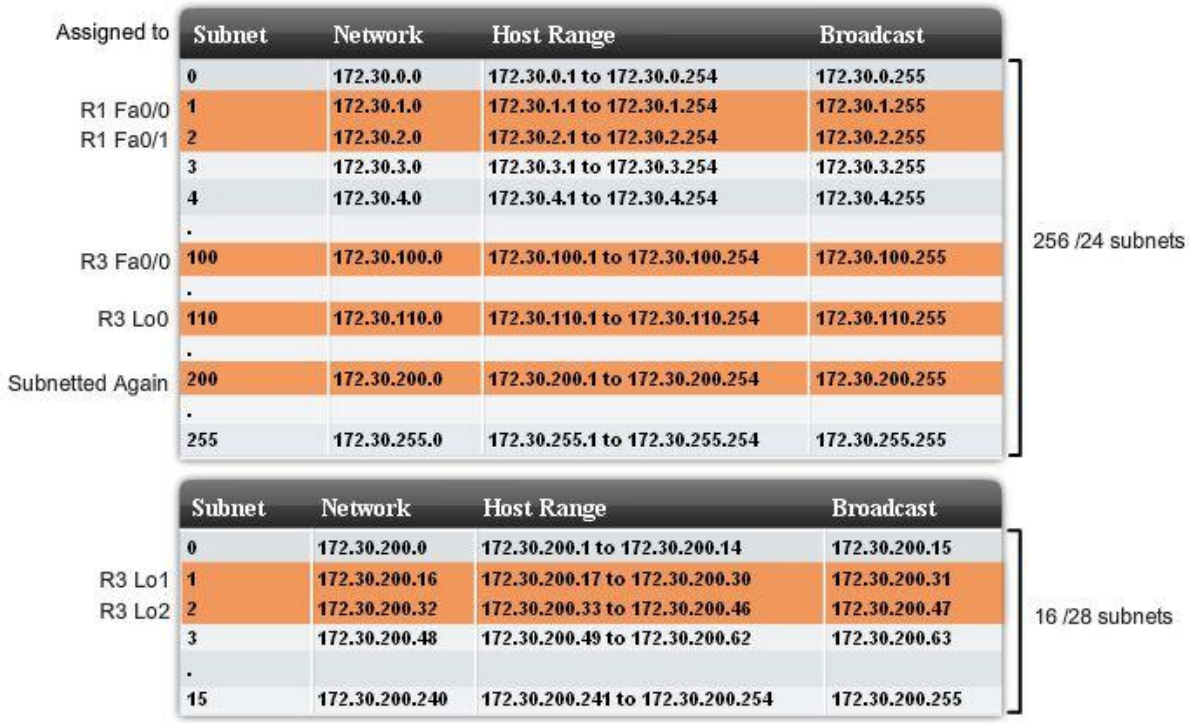
Device	Interface	IP Address	Subnet Mask
R1	Fa0/0	172.30.1.1	255.255.255.0
	Fa0/1	172.30.2.1	255.255.255.0
	S0/0/0	209.165.200.230	255.255.255.252
R2	Fa0/0	10.1.0.1	255.255.0.0
	S0/0/0	209.165.200.229	255.255.255.252
	S0/0/1	209.165.200.233	255.255.255.252
R3	Fa0/0	172.30.100.1	255.255.255.0
	Lo0	172.30.110.1	255.255.255.0
	Lo1	172.30.200.17	255.255.255.240
	Lo2	172.30.200.33	255.255.255.240
	S0/0/1	209.165.200.234	255.255.255.252

Рис. 6.2 Адресна схема

VLSM

Розглянемо VLSM адресну схему на рис. 6.3. Як показано у верхній таблиці, обоє R1 і R3 мають мережу 172.30.0.0/16, поділену на /24 підмережі. Чотири з цих /24 підмереж призначені: дві для R1 (172.30.1.0/24 і 172.30.2.0/24) і дві для R3 (172.30.100.0/24 і 172.30.110.0/24).

У нижній таблиці, береться підмережа 172.30.200.0/24 і ділиться ще на підмережі, використовуючи 4 перших біта для підмереж і ще залишилися 4 біта для хостів. В результаті маска 255.255.255.240 або /28. Підмережа 1 і Підмережа 2 призначені R3.



Assigned to	Subnet	Network	Host Range	Broadcast
	0	172.30.0.0	172.30.0.1 to 172.30.0.254	172.30.0.255
R1 Fa0/0	1	172.30.1.0	172.30.1.1 to 172.30.1.254	172.30.1.255
R1 Fa0/1	2	172.30.2.0	172.30.2.1 to 172.30.2.254	172.30.2.255
	3	172.30.3.0	172.30.3.1 to 172.30.3.254	172.30.3.255
	4	172.30.4.0	172.30.4.1 to 172.30.4.254	172.30.4.255
	.			
R3 Fa0/0	100	172.30.100.0	172.30.100.1 to 172.30.100.254	172.30.100.255
	.			
R3 Lo0	110	172.30.110.0	172.30.110.1 to 172.30.110.254	172.30.110.255
	.			
Subnetted Again	200	172.30.200.0	172.30.200.1 to 172.30.200.254	172.30.200.255
	.			
	255	172.30.255.0	172.30.255.1 to 172.30.255.254	172.30.255.255

256 /24 subnets

Subnet	Network	Host Range	Broadcast
0	172.30.200.0	172.30.200.1 to 172.30.200.14	172.30.200.15
1	172.30.200.16	172.30.200.17 to 172.30.200.30	172.30.200.31
2	172.30.200.32	172.30.200.33 to 172.30.200.46	172.30.200.47
3	172.30.200.48	172.30.200.49 to 172.30.200.62	172.30.200.63
.			
15	172.30.200.240	172.30.200.241 to 172.30.200.254	172.30.200.255

R3 Lo1
R3 Lo2

16 /28 subnets

Рис. 6.3 Поділ мереж на підмережі в адресній схемі

RFC 1918 приватні адреси

Ви маєте вже бути знайомі з приватною адресацією RFC 1918 (рис. 6.4). У всіх прикладах для внутрішніх мереж використовуються приватні IP адреси. Але, коли IP трафік маршрутизується через WAN лінії до ISP, або, коли внутрішнім користувачам потрібно звернутися до зовнішніх сайтів, повинна використовуватися відкрита (public) адреса IP.

Class	Prefix/Mask	Address Range
A	10.0.0.0/8	10.0.0.0 to 10.255.255.255
B	172.16.0.0/12	172.16.0.0 to 172.31.255.255
C	192.168.0.0/16	192.168.0.0 to 192.168.255.255

Рис. 6.4 RFC 1918 приватні адреси

IP адреси, які використовуються в прикладах від Cisco

Ви, можливо, звернули увагу, що WAN зв'язки між R1, R2, і R3 використовують відкриті (public) адреси IP. Хоча ці IP адреси - не приватні згідно RFC 1918, Cisco придбав деякий відкритий адресний простір, аби використовувати його в прикладах.

Адреси, показані на рис. 6.5, - дійсні відкриті адреси IP, які автоматично маршрутизуються в Інтернет. Cisco встановив ці адреси для освітніх цілей. Вони використовуються в прикладах, там де необхідно використовувати відкриті адреси.

Prefix/Mask	Address Range
209.165.200.224/27	209.165.200.224 to 209.165.200.255
209.165.201.0/27	209.165.201.0 to 209.165.201.31
209.165.202.128/27	209.165.202.128 to 209.165.202.159

Рис. 6.5 IP адреси, які використовуються в прикладах від Cisco

На рис. 7.1, R1, R2, і R3 підключені, використовуючи 209.165.200.224/27 з відкритого адресного простору Cisco. Оскільки WAN зв'язкам необхідно лише дві адреси, 209.165.200.224/27 ділиться на підмережі з маскою /30. У топології підмережа 1 призначена для WAN каналу між R1 і R2. Підмережа 2 призначена для WAN каналу між R2 і R3.

Петлеві (Loopback) інтерфейси

Зверніть увагу на рис. 6.2, що R3 використовує петлеві інтерфейси (Lo0, Lo1, і Lo2). loopback інтерфейс – це програмний інтерфейс, який використовується, аби емулювати фізичний інтерфейс. Подібно до інших інтерфейсів, йому може бути призначена IP адреса. Петлеві інтерфейси також використовуються іншими протоколами маршрутизації, наприклад OSPF, для різних цілей.

У лабораторному середовищі, петлеві інтерфейси корисні для створення додаткових мереж без необхідності додавання фізичних інтерфейсів на маршрутизатор. Петлевий інтерфейс можна пропінгувати і підмережа може анонсуватися в маршрутних оновленнях. Тому, петлеві інтерфейси ідеальні для моделювання множинних мереж, приєднаних до одного маршрутизатора. У нашому прикладі, R3 не потребує чотири LAN інтерфейси, аби продемонструвати множинні підмережі і VLSM. Замість цього, ми використовуємо петлеві інтерфейси.

6.1.2 Обмеження топології RIPv1

Статичні маршрути і Null інтерфейси

Аби сформувати статичний супермережний маршрут на R2, використовується наступна команда:

*R2(config) #**ip route 192.168.0.0 255.255.0.0 Null0***

Пам'ятаєте, що сумаризація маршруту дозволяє одному маршрутному запису високо рівня представляти багато маршрутів низького рівня, тим самим скорочуючи розмір таблиць маршрутизації? Статичний маршрут на R2 використовує маску /16, аби підсумовувати все 256 мереж, в діапазоні від 192.168.0.0/24 до 192.168.255.0/24.

Адресний простір, представлений статичним сумарним маршрутом 192.168.0.0/16, фактично не існує. Для того, щоб симулювати цей статичний маршрут, ми використовуємо нульовий інтерфейс (Null0) як exit-интерфейс. Вам не потрібно використовувати жодних команд, аби створити або конфігурувати нульовий інтерфейс. Він завжди в стані «up», але не пересилає і не отри-

мує трафік. Трафік, відправлений нульовому інтерфейсу, скидається. Для наших цілей, нульовий інтерфейс служитиме як exit-интерфейс нашого статичного маршруту. Відомо, що статичний маршрут повинен мати активний exit-интерфейс перш, ніж він буде встановлений в таблиці маршрутизації. Використання нульового інтерфейсу дозволить R2 анонсувати статичний маршрут через RIP, навіть якщо мережі, що належать 192.168.0.0/16, фактично не існують.

Перерозподіл (redistribution) маршруту

Друга команда, яку потрібно ввести, - команда перерозподілу статичного маршруту:

*R2(config-router) #**redistribute static***

Перерозподіл бере маршрути з одного джерела маршрутизації і посилає їх іншому джерелу маршрутизації. У нашому прикладі, ми хочемо, аби процес RIP на R2 перерозподілив наш статичний маршрут (192.168.0.0/16) шляхом імпортування маршруту в RIP, а потім відправив його до R1 і R3, використовуючи процес RIP.

Перевірка і тестування з'єднання

Аби перевірити, чи забезпечує топологія зв'язок, ми спочатку перевіряємо, що обидва серійні канали на R2 знаходяться в стані «up», використовуючи команду ***show ip interface brief***.

Але чи може R2 пропінгувати мережі на R1 і R3? Чи є проблеми забезпечення зв'язку з класовим протоколом маршрутизації і несуміжними підмережами 172.30.0.0? Перевіримо зв'язок між маршрутизаторами.

На рис. 6.6 ми бачимо лише 50% успіху, коли R2 пінгує підмережі 172.30.0.0 на R1 або R3.

```
R2#ping 172.30.1.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.30.1.1, timeout is 2 seconds:
!U!.!
Success rate is 60 percent (3/5), round-trip min/avg/max = 28/29/32 ms

R2#ping 172.30.100.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.30.100.1, timeout is 2 seconds:
!U!.!
Success rate is 60 percent (3/5), round-trip min/avg/max = 28/28/28 ms
R2#
```

Рис. 6.6 Пінг R2 до підмереж 172.30.0.0 успішний лише частково

На рис. 6.7 видно, що R1 може пінгувати 10.1.0.1, але інтерфейс 172.30.100.1 для нього є недосяжним.

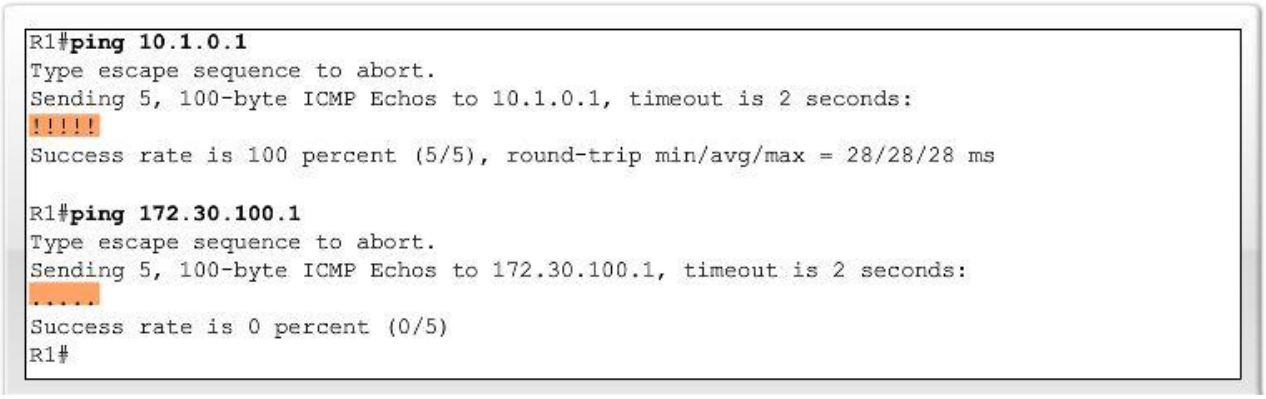


Рис. 6.7 Мережа 172.30.100.0 на R3 для R1 недосяжна

Видно, що є очевидна проблема, при спробі зв'язатися з 172.30.0.0 несуміжними підмережами. Спробуємо дослідити і вирішити цю проблему.

6.1.3 RIPv1: несуміжні мережі

Ви вже знаєте, що RIPv1 - класовий протокол маршрутизації. Як видно з формату повідомлення RIPv1 (рис. 5.2), він не включає маску підмережі в маршрутні оновлення. Тому, RIPv1 не може підтримувати несуміжні мережі, VLSM, або CIDR-супермережі. Як повинен змінитися формат цього повідомлення, аби включати маску підмережі?

Оскільки маска підмережі не включається в оновлення, RIPv1 і інші класові протоколи маршрутизації повинні підсумовувати мережі по класових мережних кордонах. Як можна бачити на рис. 6.8, при посилці оновлень до R2, RIPv1 і на R1, і на R3 підсумовуватиме їх 172.30.0.0 підмережі за класовою мережною адресою 172.30.0.0. З боку R2, обидва оновлення мають рівну вартість в 1 перехід, аби досягти мережі 172.30.0.0/16. Як видно на рис. 6.9 R2 встановлює обидва шляхи рівної вартості в таблицю маршрутизації. І з цієї причини має 50% успіху при з'ясуванні досяжності цих мереж на рис.6.6.

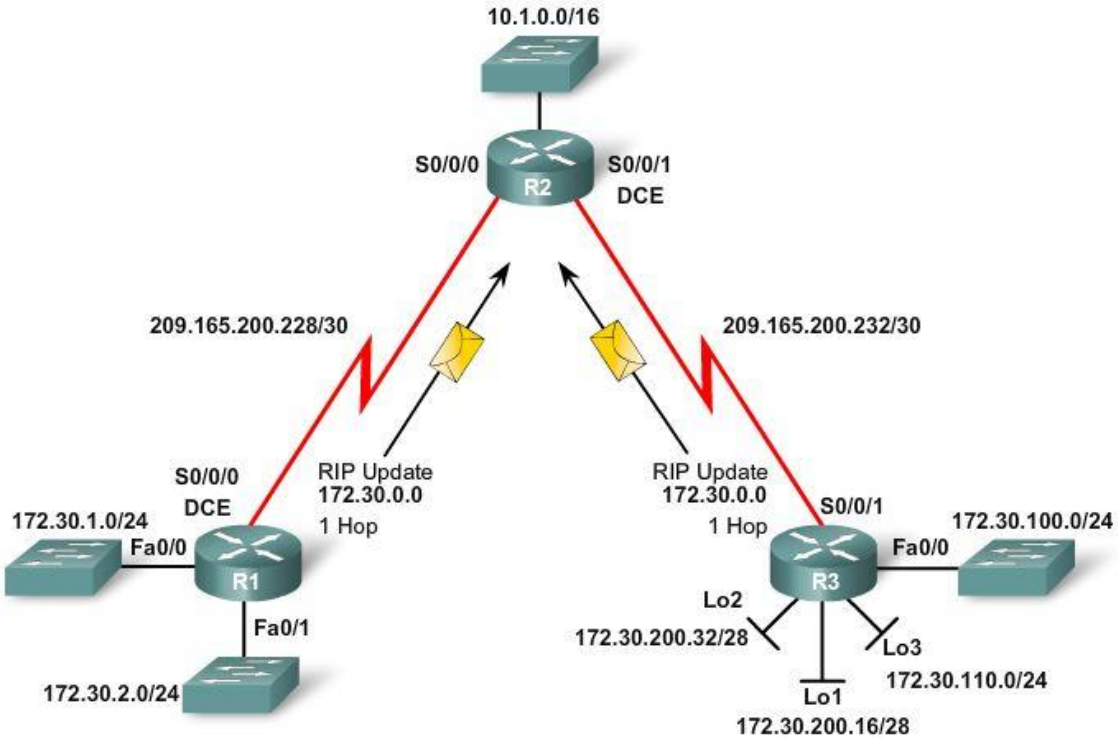


Рис. 6.8 Автоматична сумаризація

```
R2#show ip route
Codes: C - connected, S - static, I - IGRP, 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

R    172.30.0.0/16 [120/1] via 209.165.200.230, 00:00:09, Serial0/0/0
                   [120/1] via 209.165.200.234, 00:00:11, Serial0/0/1
     209.165.200.0/30 is subnetted, 2 subnets
C       209.165.200.232 is directly connected, Serial0/0/1
C       209.165.200.228 is directly connected, Serial0/0/0
     10.0.0.0/16 is subnetted, 1 subnets
C       10.1.0.0 is directly connected, FastEthernet0/0
S    192.168.0.0/16 is directly connected, Null0
```

Рис. 6.9 R2 має два шляхи рівної вартості до 172.30.0.0

У свою чергу R2 не стане включає 172.30.0.0 мережа в свої оновлення ні до R1 ні до R3 за правилом розщепленого горизонту.

6.1.4 RIPv1 : немає підтримки VLSM

Оскільки RIPv1 не надсилає маску підмережі в оновленнях, він не може підтримувати VLSM. Маршрутизатор R3 сформований з VLSM підмережами, які відносяться до мережі 172.30.0.0/16 класу B:

172.30.100.0/24 (FastEthernet 0/0)
172.30.110.0/24 (Loopback 0)
172.30.200.16/28 (Loopback 1)

172.30.200.32/28 (Loopback 2)

RIPv1 або підсумовує підмережі до класового кордону, або використовує маску підмережі вихідного інтерфейсу, аби визначити, які підмережі анонсувати.

Аби продемонструвати, як RIPv1 використовує маску підмережі вихідного інтерфейсу, до топології доданий R4, сполучений з R3 через інтерфейс FastEthernet0/0 по мережі 172.30.100.0/24 (рис. 6.10).

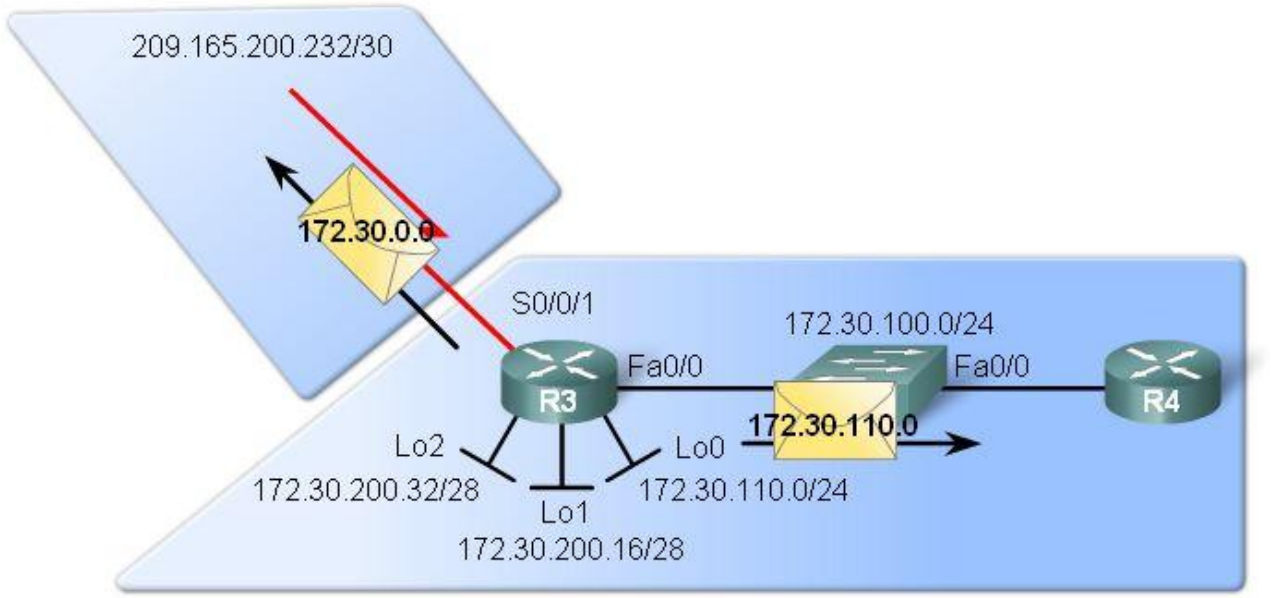


Рис. 6.10 Оновлення RIPv1 не підтримують VLSM

Зверніть увагу, що лише одна підмережа 172.30.110.0 зі всіх підмереж 172.30.0.0 включена в анонс до маршрутизатора R4. Інші підмережі не включаються в оновлення до R4 тому що мають маску, відмінну від маски, встановленої на FastEthernet0/0. Ось чому всі підмережі повинні використовувати однакову маску підмережі при використанні класового протоколу маршрутизації. Також примітно, що R3 пересилає класовий маршрут 172.30.0.0 через Serial 0/0/1.

6.1.5 RIP v1: немає підтримки CIDR

На рис. 6.11 ми сформували на R2 статичний маршрут до мережі 192.168.0.0/16 і інструктували RIP, аби він включав цей маршрут в свої оновлення.

```
R2(config)#router rip
R2(config-router)#redistribute static
R2(config-router)#network 10.0.0.0
R2(config-router)#network 209.165.200.0
R2(config-router)#exit
R2(config)#ip route 192.168.0.0 255.255.0.0 null0
```

Рис. 6.11 Конфігурація маршрутизації на R2

На рис. 6.12 ми можемо бачити, що статичний маршрут входить в таблицю маршрутизації R2.

```
R2#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
(**output omitted**)
R    172.30.0.0/16 [120/1] via 209.165.200.230, 00:00:09, Serial0/0/0
                   [120/1] via 209.165.200.234, 00:00:11, Serial0/0/1
     209.165.200.0/30 is subnetted, 2 subnets
C       209.165.200.232 is directly connected, Serial0/0/1
C       209.165.200.228 is directly connected, Serial0/0/0
     10.0.0.0/16 is subnetted, 1 subnets
C       10.1.0.0 is directly connected, FastEthernet0/0
S    192.168.0.0/16 is directly connected, Null0
```

Рис. 6.12 Таблиця маршрутизації R2

Використовуючи ***debug ip rip*** на R2, ми звертаємо увагу на те, що RIPv1 не включає маршрут 192.168.0.0/16 в свої RIP-оновлення до R1, або R3 (рис. 6.13). Чому цей маршрут не включається? Річ в тім, що маршрут 192.168.0.0/16 відноситься до класу C, а маска використовується менша, ніж за замовчанням для даного класу.

```
R2#debug ip rip
RIP protocol debugging is on
(**output omitted**)
RIP: received v1 update from 209.165.200.230 on Serial0/0/0
     172.30.0.0 in 1 hops
RIP: received v1 update from 209.165.200.234 on Serial0/0/1
     172.30.0.0 in 1 hops
R2#
RIP: sending v1 update to 255.255.255.255 via Serial0/0/0 (209.165.200.229)
RIP: build update entries
        network 10.0.0.0 metric 1
        subnet 209.165.200.232 metric 1
RIP: sending v1 update to 255.255.255.255 via Serial0/0/1 (209.165.200.233)
RIP: build update entries
        network 10.0.0.0 metric 1
        subnet 209.165.200.228 metric 1
R2#
```

Рис. 6.13 RIPv1 не анонсує статичний маршрут до R1 і R3.

Оскільки маска не узгоджується з маскою класу, RIPv1 не включатиме цей маршрут в свої оновлення до інших маршрутизаторів.

RIPv1 та інші класові протоколи маршрутизації не можуть підтримувати CIDR-маршрути, які є сумарними маршрутами з більш маленькою маскою підмережі, ніж класова маска маршруту. RIPv1 нехтує цими супермережами в таблиці маршрутизації і не включає їх в оновлення до інших маршрутизаторів. Це, тому що одержуючий маршрутизатор зміг би застосувати лише класову маску, а не коротшу маску /16.

Примітка: Якби статичний маршрут 192.168.0.0 був сформований з маскою /24 або більшою, цей маршрут увійшов би до оновлень RIP. Одержуючі маршрутизатори застосували б до цього оновлення класову маску /24.

6.2 Конфігурація RIPv2

6.2.1 Запуск і перевірка RIPv2

Порівняння форматів повідомлень RIPv1 і RIPv2

RIPv2 визначений в RFC 1723. Подібно до версії 1, RIPv2 інкапсулюється в UDP сегмент, що використовує порт 520, і може переносити аж до 25 маршрутів. Хоча RIPv2 має той же основний формат повідомлення, що і RIPv1, додано два істотні розширення.

Перше розширення у форматі повідомлення RIPv2 - поле маски підмережі, яке дозволяє включати 32 розрядну маску в маршрутний запис RIP (рис. 6.14). В результаті, одержуючий маршрутизатор більше не залежить від маски підмережі вхідного інтерфейсу або класової маски, визначаючи маску підмережі для маршруту.

Друге істотне розширення до формату повідомлення RIPv2 - додавання Next Hop адреси. Next Hop адреса використовується, аби ідентифікувати кращий наступний перехід - якщо він існує - ніж адреса пересилаючого маршрутизатора. Якщо поле встановлене в 0.0.0.0, адреса маршрутизатора, що відіслав, - краща адреса наступного переходу.

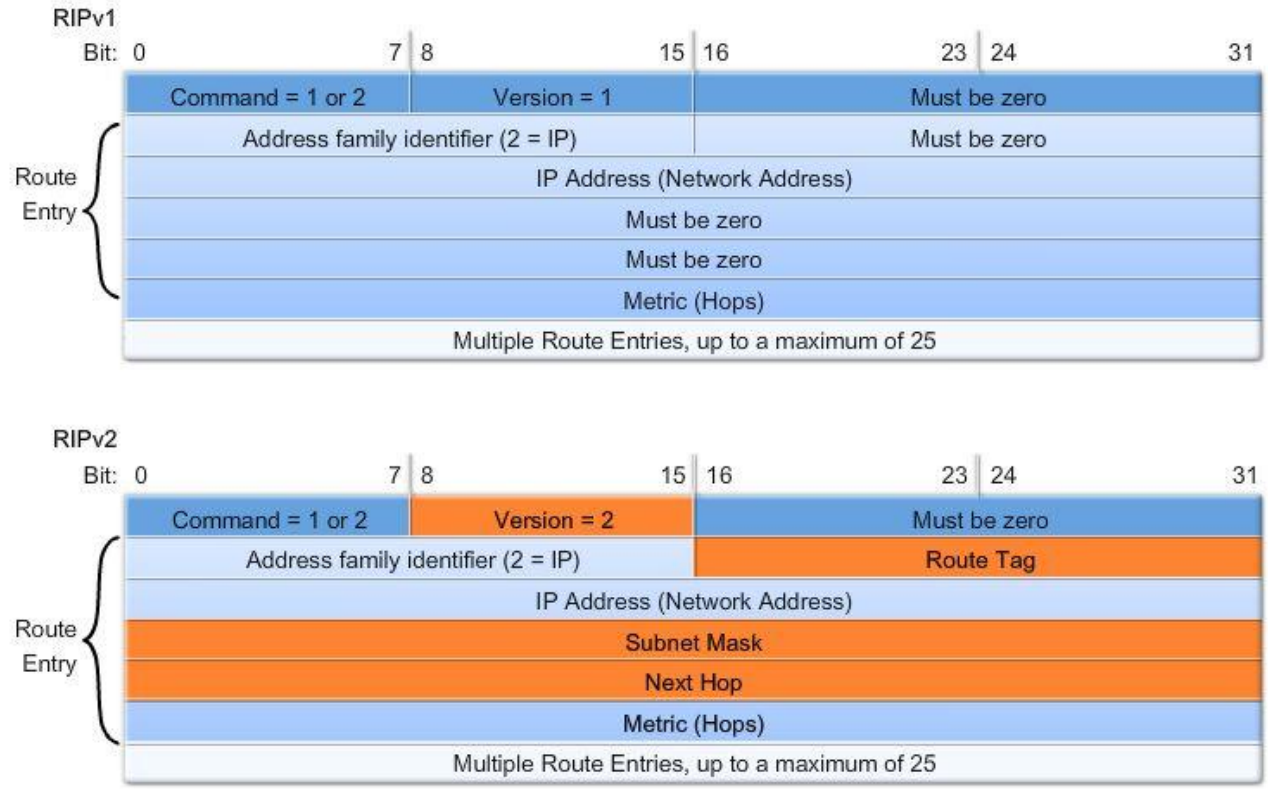


Рис. 6.14 Порівняння форматів повідомлень RIPv1 і RIPv2

Версія 2

За замовчанням, коли процес RIP сформований на маршрутизаторі Cisco, то виконується RIPv1. Проте навіть якщо маршрутизатор посилає лише повідо-

млення RIPv1, він може інтерпретувати як RIPv1, так і повідомлення RIPv2 (рис. 6.15). Маршрутизатор RIPv1 при цьому нехтуватиме полями RIPv2 в маршрутному записі.

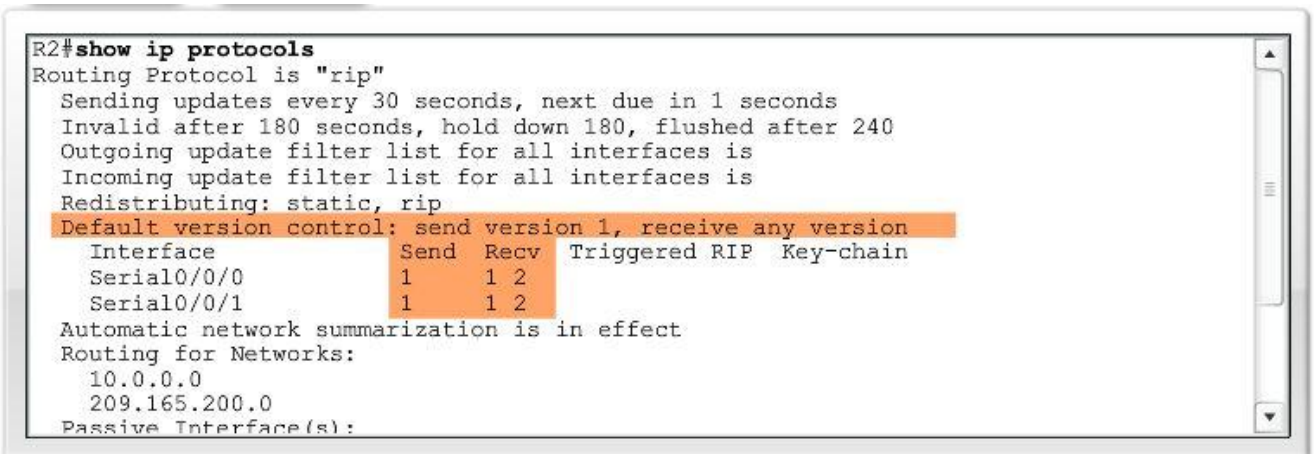


Рис. 6.15 Команда show ip protocols перевіряє, що R2 сформований для RIPv1, але отримує повідомлення RIP для обох версій.

Для того, щоб використовувати RIPv2, потрібно виконати команду:

*R2(config-router) #**version 2***

на всіх маршрутизаторах доменк маршрутизації. Процес RIP тепер включатиме маску підмережі у всі оновлення, роблячи RIPv2 безкласовим протоколом маршрутизації.

Коли маршрутизатор сформований для версії 2, лише повідомлення RIPv2 відсилаються і приймаються.

Задана за замовчанням поведінка RIPv1 може бути відновлена, якщо використовувати команду no version в режимі конфігурації маршрутизатора. Проте, для цих цілей також можна використовувати команду version 1. В цьому випадку лише повідомлення RIPv1 відсилатимуться і прийматимуться.

6.2.2 Автосумаризація і RIPv2

Дослідження таблиць маршрутизації

Оскільки RIPv2 - безкласовий протокол маршрутизації, ми чекали б побачити індивідуальні підмережі 172.30.0.0 в таблицях маршрутизації. Проте, коли ми розглядаємо таблицю маршрутизації для R2, ми все ще бачимо сумарний 172.30.0.0/16 маршрут з тими ж двома шляхами рівної вартості (рис. 6.9). Маршрутизатори R1 і R3 все ще не знають про підмережі 172.30.0.0 іншого маршрутизатора.

Єдина різниця, поки що між RIPv1 і RIPV2 та, що R1 і R3 кожен має маршрут до 192.168.0.0/16 супермережі. Цей маршрут був статичним маршрутом, сформованим на R2 і перерозподіленим RIP (рис. 6.16).

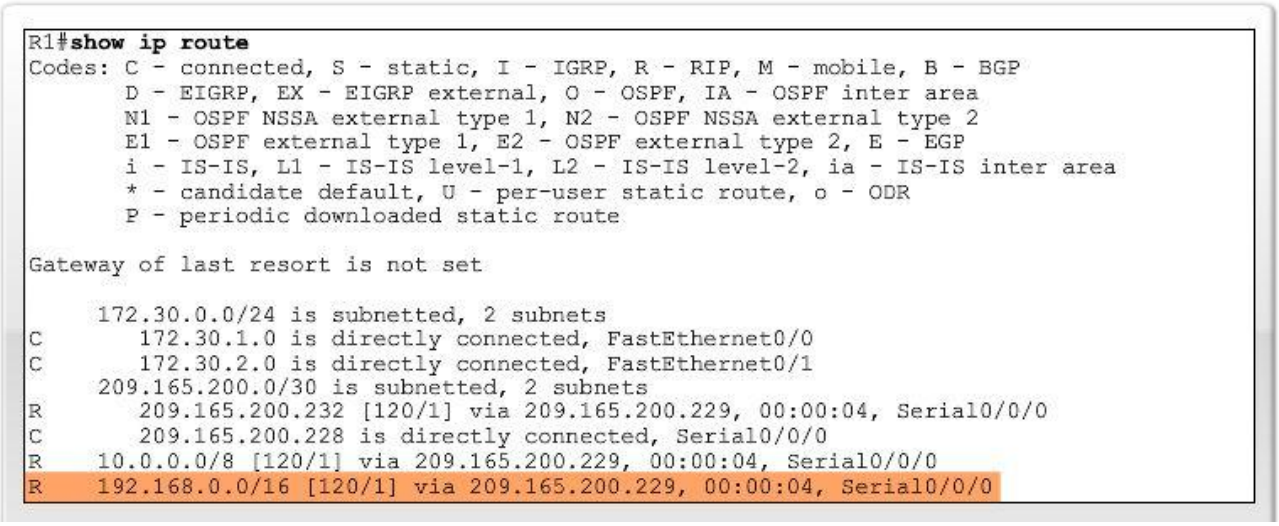


Рис. 6.16 Тепер на R1 присутній супермережний маршрут

Аби досліджувати, які маршрути посилає і отримує RIPv2 на R1, використовуємо команду ***debug ip rip***

RIP: sending v2 update to 224.0.0.9 via Serial0/0 (209.165.200.230)
172.30.0.0/16 via 0.0.0.0, metric 1, tag 0

Зверніть увагу, що RIPv2 посилає і мережну адресу і маску підмережі. Проте, надісланий маршрут – сумарна класова мережева адреса, 172.30.0.0/16, а не окремі 172.30.1.0/24 і 172.30.2.0/24 підмережі.

За замовчанням, RIPv2 автоматично підсумовує мережі на класових мережних кордонах, так само як RIPv1. Тому R1 і R3 все ще підсумовують свої 172.30.0.0 підмережі за класовою адресою 172.30.0.0 при надсиланні оновлень на інтерфейси 209.165.200.228 і 209.165.200.232, відповідно. Команда show ip protocols на рис. 6.17 перевіряє, що "автоматична сумаризація включена".

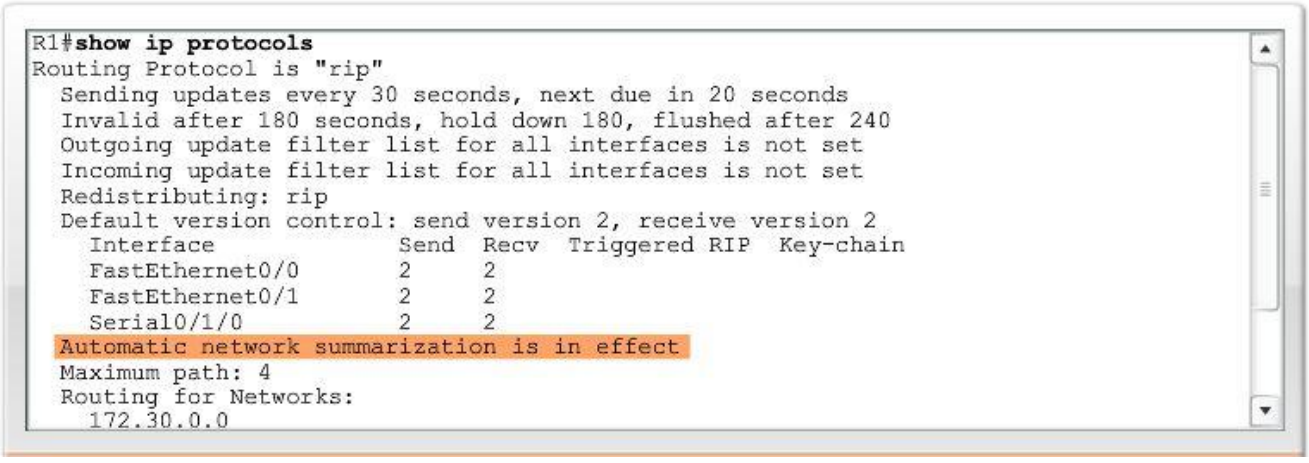


Рис. 6.17 Автоматична сумаризація замовчанням включена

6.2.3 Відключення автосумаризації в RIPv2

За допомогою команди ***no auto-summary*** в режимі конфігурації маршрутизації, можна змінити поведінку RIPv2. Ця команда не працює з RIPv1. Тому спочатку потрібно конфігурувати ***version 2***.

В цьому випадку RIPv2 включатиме всі підмережі і їх відповідні маски в свої маршрутні оновлення. Команда ***show ip protocols*** може використовуватися, аби перевірити, що автоматична сумаризація відключена - "automatic network summarization is not in effect."

Тепер, коли ми використовуємо безкласовий протокол маршрутизації RIPv2 і ми відключили автоматичну сумаризацію, таблиця маршрутизації для R2 містить індивідуальні підмережі для 172.30.0.0/16 (рис. 6.18). Немає більше маршруту з двома шляхами рівної вартості.

```
R2#show ip route
 (**output omitted**)

Gateway of last resort is not set

     172.30.0.0/16 is variably subnetted, 6 subnets, 2 masks
R       172.30.200.32/28 [120/1] via 209.165.200.234, 00:00:09, Serial0/0/1
R       172.30.200.16/28 [120/1] via 209.165.200.234, 00:00:09, Serial0/0/1
R       172.30.2.0/24 [120/1] via 209.165.200.230, 00:00:03, Serial0/0/0
R       172.30.1.0/24 [120/1] via 209.165.200.230, 00:00:03, Serial0/0/0
R       172.30.100.0/24 [120/1] via 209.165.200.234, 00:00:09, Serial0/0/1
R       172.30.110.0/24 [120/1] via 209.165.200.234, 00:00:09, Serial0/0/1
     209.165.200.0/30 is subnetted, 2 subnets
C       209.165.200.232 is directly connected, Serial0/0/1
C       209.165.200.228 is directly connected, Serial0/0/0
     10.0.0.0/16 is subnetted, 1 subnets
C       10.1.0.0 is directly connected, FastEthernet0/0
S    192.168.0.0/16 is directly connected, Null0
```

Рис. 6.18 R2 має всі підмережі в своїй таблиці маршрутизації

Таблиця маршрутизації для R1 (рис. 6.19) містить всі підмережі 172.30.0.0/16, у тому числі, отримані від R3.

```
R1#show ip route
 (**output omitted**)

Gateway of last resort is not set

     172.30.0.0/16 is variably subnetted, 6 subnets, 2 masks
R       172.30.200.32/28 [120/2] via 209.165.200.229, 00:00:01, Serial0/0/0
R       172.30.200.16/28 [120/2] via 209.165.200.229, 00:00:01, Serial0/0/0
C       172.30.1.0/24 is directly connected, FastEthernet0/0
C       172.30.2.0/24 is directly connected, FastEthernet0/1
R       172.30.100.0/24 [120/2] via 209.165.200.229, 00:00:01, Serial0/0/0
R       172.30.110.0/24 [120/2] via 209.165.200.229, 00:00:01, Serial0/0/0
     209.165.200.0/30 is subnetted, 2 subnets
R       209.165.200.232 [120/1] via 209.165.200.229, 00:00:02, Serial0/0/0
C       209.165.200.228 is directly connected, Serial0/0/0
     10.0.0.0/16 is subnetted, 1 subnets
R       10.1.0.0 [120/1] via 209.165.200.229, 00:00:02, Serial0/0/0
R    192.168.0.0/16 [120/1] via 209.165.200.229, 00:00:02, Serial0/0/0
```

Рис. 6.19 R1 має всі підмережі в таблиці маршрутизації

Аналогічно, таблиця маршрутизації для R3 містить всі підмережі

172.30.0.0/16, у тому числі отримані від R1. Ця мережа конвергована.

Оновлення RIPv2 надсилаються на групову адресу 224.0.0.9. RIPv1 посилає оновлення як широкомовні 255.255.255.255. Є декілька переваг використання групової адреси. Групова доставка повідомлень може займати меншу пропускну спроможність мережі. Крім того, оновлення групової розсилки вимагають менше обробки не RIP пристроями. Під RIPv2, будь-який пристрій, який не сформований для RIP, відкине фрейм на Канальному рівні. При широкомовній розсилці RIPv1, всі пристрої в широкомовній мережі, наприклад Ethernet, повинні обробити оновлення RIP повністю аж до Транспортного рівня, де пристрій виявляє, що пакет призначався для процесу, який не існує.

6.3 VLSM і CIDR

6.3.1 RIPv2 і VLSM

Оскільки безкласові протоколи маршрутизації, подібні RIPv2 можуть пересилати як мережну адресу, так і маску підмережі, їм не потрібно підсумовувати мережі по їх класових адресах. Тому, безкласові протоколи маршрутизації підтримують VLSM. Маршрутизаторам, які використовують RIPv2 більше не потрібно використовувати маску вхідного інтерфейсу, аби визначити маску підмережі маршруту, що анонсується. Мережа і маска явно входять в кожне маршрутне оновлення.

У мережах, які використовують VLSM адресну схему, безкласовий протокол маршрутизації важливий, аби поширювати всі мережі разом з їх правильними масками підмережі.

На рис. 6.20 видно, що з RIPv2, R3 може включати всі свої 172.30.0.0 підмереж, посилаючи оновлення до R4. Даний рисунок відрізняється цим від рис. 6.10, на якому було показано використання класового протоколу маршрутизації. Це, тому що RIPv2 може включати належну маску підмережі разом з мережевою адресою в оновлення.

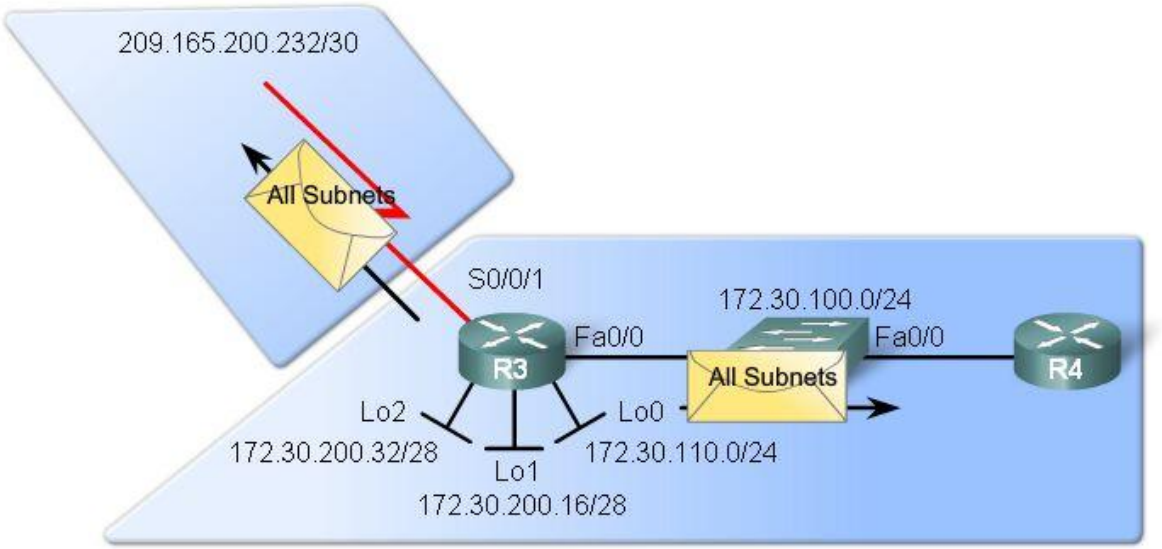


Рис. 6.20 RIPv2 підтримує VLSM

6.3.2 RIPv2 і CIDR

Одна з цілей CIDR, як заявлено в RFC 1519 - "забезпечити механізм для агрегації маршрутної інформації." Ця мета включає поняття супермережі. Супермережа - блок послідовних класових мереж, які адресуються як єдина мережа. На маршрутизаторі R2, ми сформували супермережу - статичний маршрут до єдиної мережі, яка використовується, аби представити множинні мережі або підмережі.

Супермережі мають маски, більш маленькі, ніж класова маска (тут /16, замість класової /24). Для супермережі, яка входить в маршрутне оновлення, протокол маршрутизації повинен мати можливість перенесення маски. Іншими словами, це має бути безкласовий протокол маршрутизації, подібний RIPv2.

Статичний маршрут на R2 включає маску, яка менше ніж класова маска:

*R2(config) #**ip route 192.168.0.0 255.255.0.0 Null0***

Оскільки використовується безкласовий протокол маршрутизації, ця CIDR-супермережа буде входити в оновлення, що надсилаються R2. Для цього навіть не доведеться відключати автоматичну сумаризацію.

6.4 Перевірка і пошук несправностей RIPv2

6.4.1 Команди для перевірки і пошуку несправностей RIPv2

Є декілька способів перевірки і пошуку несправностей RIPv2:

1. Переконайтеся, що всі інтерфейси в стані «up» і працюють.
2. Перевірте кабельну систему.
3. Переконаєтеся, що правильно встановлені адреса і маска підмережі IP на кожному інтерфейсі.
4. Видаліть будь-які непотрібні конфігураційні команди, які більше не потрібні, або були замінені іншими командами.

Команда ***show ip route*** використовується першою для перевірки мережної конвергенції. При розгляді таблиці маршрутизації, можна бачити, яких маршрутів в ній немає, і які зайві.

show ip interface brief: Часто мережа відсутня в таблиці маршрутизації, тому що інтерфейс знаходиться в стані «down» або неправильно конфігурований. Команда show ip interface brief швидко перевіряє стан всіх інтерфейсів.

Команда ***show ip protocols*** перевіряє декілька критичних елементів, у тому числі, що RIP запущений, версію RIP, стан автоматичної сумаризації, і мережі, які були вказані в network інструкціях. Джерела маршрутної інформації, зазначені внизу, – це сусіди RIP, від яких маршрутизатор в даний час отримує оновлення.

Команда ***debug ip rip*** використовується, аби розглянути вміст маршрутних оновлень, які надсилаються і отримуються маршрутизатором. Інколи мар-

шрут отримується маршрутизатором, але не додається до таблиці маршрутизації. Причина в тому, що для цієї мережі може бути також сформований статичний маршрут. За замовчанням, статичний маршрут має нижчу адміністративну відстань, ніж будь-який протокол динамічної маршрутизації, і до таблиці маршрутизації буде доданий саме він.

Команда ping. Пропінгуйте локальні інтерфейси. Потім інтерфейси маршрутизатора на безпосередньо підключених мережах. Продовжуйте пінгувати інтерфейси на кожному наступному маршрутизаторі. Як тільки пінг зазнає невдачі, досліджуйте маршрутизатори, між якими пінг потерпів невдачу, аби визначити її причину.

Команда ***show running-config*** використовується, аби перевірити всі команди, сформовані в даний час.

6.4.2 Загальні проблеми RIPv2

При пошуку проблем в RIPv2, є декілька областей для дослідження (рис. 6.21).

```
R1#show running-config
Building configuration...
 !
hostname R1
 !
interface FastEthernet0/0
 ip address 172.30.1.1 255.255.255.0
 !
interface FastEthernet0/1
 ip address 172.30.2.1 255.255.255.0
 !
interface Serial0/0/0
 ip address 209.165.200.230 255.255.255.252
 clock rate 64000
 !
router rip
 version 2
 network 172.30.0.0
 network 209.165.200.0
 no auto-summary
 !
***output omitted***
 !
end
```

Рис. 6.21 Перевірка основних несправностей RIPv2

Версія

Спочатку потрібно перевірити, що версія 2 сформована на всіх маршрутизаторах. Хоча RIPv1 і RIPv2 сумісні, RIPv1 не підтримує несуміжні підмережі, VLSM, або маршрути CIDR-супермережі. Завжди краще використовувати один і той же протокол маршрутизації на всіх маршрутизаторах.

Інструкції network

Можливо, була вказана некоректно network команда. Пам'ятаєте, що network інструкція робить дві речі:

- ***Дозволяє протоколу маршрутизації посилати і отримувати оновлення на будь-яких локальних інтерфейсах, які належать цій мережі.***
- ***Включає цю мережу в оновлення, що посилаються сусіднім маршрутизаторам.***

Помилка в цій команді призведе до того, що розсилатимуться неправильні оновлення або на інтерфейсі взагалі оновлення а ні розсилатимуться а ні не прийматимуться.

Автоматична сумаризація

Якщо є потреба в розсилці конкретних підмереж, а не лише сумарних маршрутів, переконайтеся, що автоматична сумаризація відключена.

6.4.3 Аутентифікація

Більшість протоколів маршрутизації посилають свої оновлення та іншу маршрутну інформацію, використовуючи IP пакети. IS-IS є виключенням і тут не обговорюється. Турбота про безпеку будь-якого протоколу маршрутизації - можливість прийняття помилкових маршрутних оновлень, що приходять від атакуючого хоста. Він намагається перервати роботу мережі або, спробує аналізувати пакети. Інше джерело недійсних оновлень - неправильно сконфігурований маршрутизатор.

Незалежно від причин, хорошою практикою є аутентифікувати маршрутну інформацію, якою обмінюються маршрутизатори. RIPv2, EIGRP, OSPF, IS-IS, і BGP можуть бути сконфігуровані так, щоб аутентифікувати маршрутну інформацію. Ця практика гарантує, що маршрутизатори прийматимуть маршрутну інформацію лише від інших маршрутизаторів, які були сконфігуровані з таким самим паролем або аутентифікаційною інформацією.

Примітка: Аутентифікація не кодує таблицю маршрутизації.

6.5 Висновки

6.5.1 Резюме

RIPv2 - безкласовий, дистанційно-векторний протокол маршрутизації, як визначено в RFC 1723. Оскільки RIPv2 - безкласовий, він включає маску підмережі разом з мережними адресами в маршрутні оновлення. Як і інші безкласові протоколи маршрутизації, RIPv2 підтримує супермережі CIDR, VLSM і несуміжні мережі.

Ми побачили, що класові протоколи маршрутизації, подібні RIPv1, не можуть підтримувати несуміжні мережі, тому що вони автоматично проводять сумаризацію на класових кордонах. Маршрутизатор, який отримує маршрутні оновлення від безлічі маршрутизаторів, що анонсують один і той же класовий маршрут, не може визначити, яка підмережа належить якому сумарному маршруту. Ця нездатність призводить до несподіваних результатів, у тому числі пакетів, що невірно маршрутизуються.

Задана за замовчанням версія RIP - версія 1. Команда ***version 2*** використовується, аби модифікувати поведінку RIP до RIPv2.

Подібно RIPv1, RIPv2 автоматично проводить сумаризацію на класових мережних кордонах. Проте, для RIPv2 автоматична сумаризація може бути відключена командою ***no auto-summary***. Автоматична сумаризація має бути відключена, аби підтримувати несуміжні мережі. RIPv2 також підтримує супермережі CIDR і VLSM, тому що конкретна маска підмережі включається в кожне маршрутне оновлення. Ви можете використовувати команду ***debug ip rip***, аби переконатися що маска підмережі пересилається в оновленнях RIPv2.

Команда ***show ip protocols*** відображуватиме, що RIP зараз пересилає і отримує версію 2 оновлення і чи включена автоматична сумаризація.

6.5.2 Питання для самоперевірки

1. Що таке несуміжні мережі і чому класовий протокол маршрутизації, такий як RIPv1, не може підтримувати несуміжні мережі?
2. На маршрутизаторі ROUTERX працює класовий протокол маршрутизації RIPv1. ROUTERX має VLSM підмережі в таблиці маршрутизації, які є частиною мережі 10.0.0.0/8. Якщо ROUTERX посилає оновлення через інтерфейс з адресою 10.10.10.1/24, то які підмережі з мережі 10.0.0.0/8 розсилатимуться через цей інтерфейс?
3. На маршрутизаторі ROUTERX працює класовий протокол маршрутизації RIPv1. ROUTERX має VLSM підмережі в таблиці маршрутизації, які є частиною мережі 10.0.0.0/8. Якщо ROUTERX посилає оновлення через інтерфейс з адресою 192.168.1.1/24, то які підмережі з мережі 10.0.0.0/8 розсилатимуться через цей інтерфейс?
4. Як відключити автоматичну сумаризацію для RIPv2?
5. У яких випадках слід відключати автоматичну сумаризацію для RIPv2?
6. Яка основна характеристика безкласового протоколу маршрутизації?
7. Яка за замовчанням поведінка RIPv2 відносно автоматичної сумаризації?
8. Чи включає RIPv2 маски підмереж в свої оновлення, якщо включена автоматична сумаризація?
9. На всіх маршрутизаторах працює RIPv2. На R1 скасована автоматична сумаризація, але на R3 автоматична сумаризація все ще включена. Які маршрути ми сподіваємося побачити в таблиці маршрутизації на R2 (рис. 6.22)?
10. Рис. 6.22. Якщо R2 має пакет, призначений вузлу 172.30.1.10, чи буде пакет перенаправлений в потрібному напрямку маршрутизатору R1?
11. Маршрутизатори R1 і R2 на рис. 6.23 обмінюються маршрутними оновленнями, використовуючи RIPv2. R2 має статичний маршрут за замовчанням до R3, і R3 має статичний маршрут за замовчанням до R2. Якщо на маршрутизаторі R1 автоматична сумаризація відключена, чи будуть для R2 досяжні підмережі і на R1 і на R3?

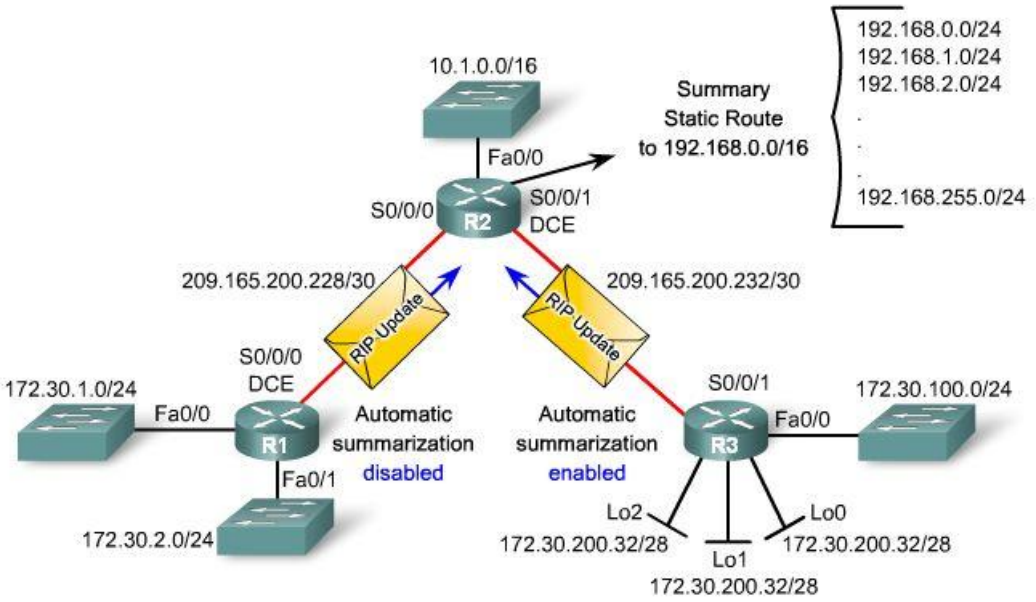


Рис. 6.22 Топологія до питання 9 та питання 10

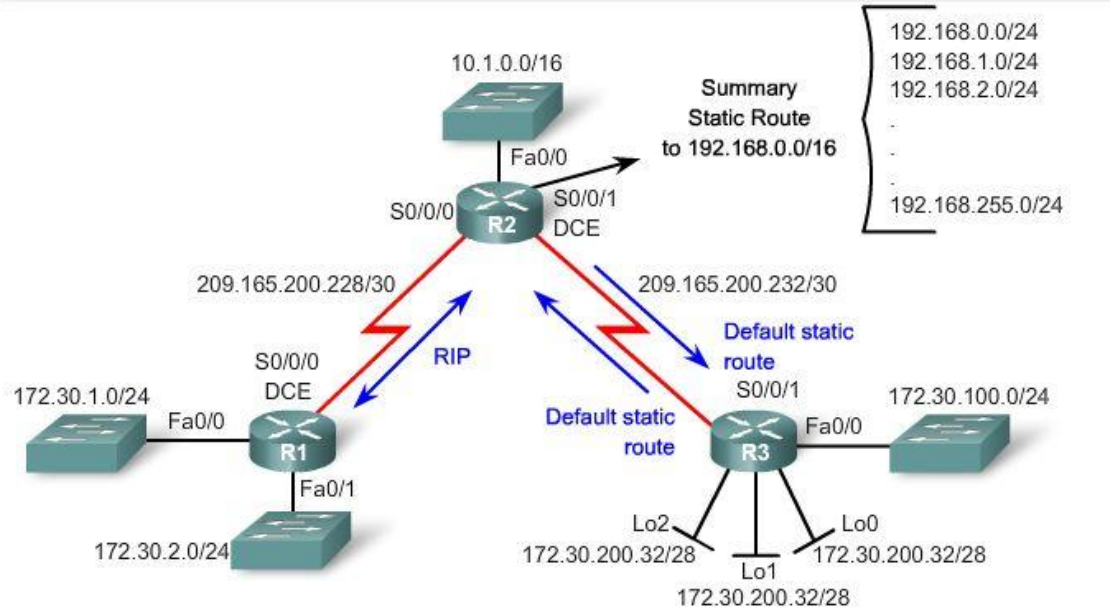


Рис. 6.23 Топологія до питанні 11

6.5.3 Матеріал для самостійного поглибленого вивчення теми

RFC 1723 - це RFC для RIPv2. Цей документ може бути доступний на web сайті www.ietf.org. Прочитайте всі частини RFC 1723, аби дізнатися більше про цей безкласовий протокол маршрутизації.

Використовуючи Packet Tracer, створіть дві несуміжні класові мережі. Кожна з цих мереж повинна мати декілька маршрутизаторів і підмереж, використовуйте VLSM. Між двома групами цих мереж додайте маршрутизатор, що зв'язує їх. Використовуйте різні класові мережі між цим маршрутизатором і кожною з двох несуміжних мереж. Використовуйте цей сценарій, аби досліджувати проблеми при використанні RIPv1 і розглянути, як RIPv2 може бути використаний для вирішення цих проблем маршрутизації.

Тема 7. Протоколи маршрутизації з урахуванням стану каналу (Link-State Routing Protocols)

Ви навчитеся:

- Описувати основні можливості й концепції протоколів маршрутизації з урахуванням стану каналу.
- Перераховувати переваги й вимоги протоколів маршрутизації з урахуванням стану каналу.

В Темі 3, "Вступ до протоколів динамічної маршрутизації", ми розглядали різницю між маршрутизацією дистанційно-векторною та за станом каналу. Дистанційно-векторні протоколи маршрутизації можна зрівняти з дорожніми покажчиками, які надають Вам інформацію тільки про відстань і напрямок. У той же час, протоколи маршрутизації за станом каналу можна порівняти із процесом вибору шляхи при наявності карти. З картою Ви можете бачити всі потенційні маршрути й визначити свій власний шлях.

Дистанційно-векторні протоколи маршрутизації повинні ухвалювати маршрутні рішення про перевагу шляху, ґрунтуючись на відстані до мережі призначення (метриці) і довірі до іншого маршрутизатора в тому, що він анонсує дійсну відстань до мережі призначення.

Протоколи маршрутизації з урахуванням стану каналу використовують інший підхід. Вони створюють топологічну карту мережі й кожний маршрутизатор використовує цю карту, щоб визначити найкоротший шлях досягнення адресата.

Маршрутизатори, на яких виконується протокол маршрутизації з урахуванням стану каналу, посилають інформацію про стан своїх каналів (links) іншим маршрутизаторам домену маршрутизації. Цей стан каналів відповідає безпосередньо підключеним мережам маршрутизатора й містить у собі інформацію про тип мережі та про всіх сусідні маршрутизатори, на яких встановлений такий же протокол маршрутизації з урахуванням стану каналу.

Кінцевою метою є, щоб кожний маршрутизатор одержав усю інформацію про стан каналів на всіх інших маршрутизаторах в області маршрутизації. Із цією інформацією про стан каналу, кожен маршрутизатор може створити свою власну топологічну карту мережі й незалежно обчислити найкоротший шлях до кожної мережі.

В цій темі розглядаються основні концепції протоколів маршрутизації з урахуванням стану каналу. В Темі 8 ми застосуємо ці концепції до протоколу OSPF.

7.1. Маршрутизація з урахуванням стану каналу

7.1.1 Протоколи маршрутизації з урахуванням стану каналу

Протоколи маршрутизації з урахуванням стану каналу також відомі як протоколи вибору найкоротшого шляху й побудовані навколо алгоритму Дейкстри вибору найкоротшого шляху (Shortest Path First - SPF).

Протоколи IP маршрутизації з урахуванням стану каналу показані на рис. 7.1:

- ***Open Shortest Path First (OSPF) – протокол вибору першого найкоротшого шляху.***
- ***Intermediate System-to-Intermediate System (IS-IS) – протокол обміну маршрутною інформацією між проміжними системами***

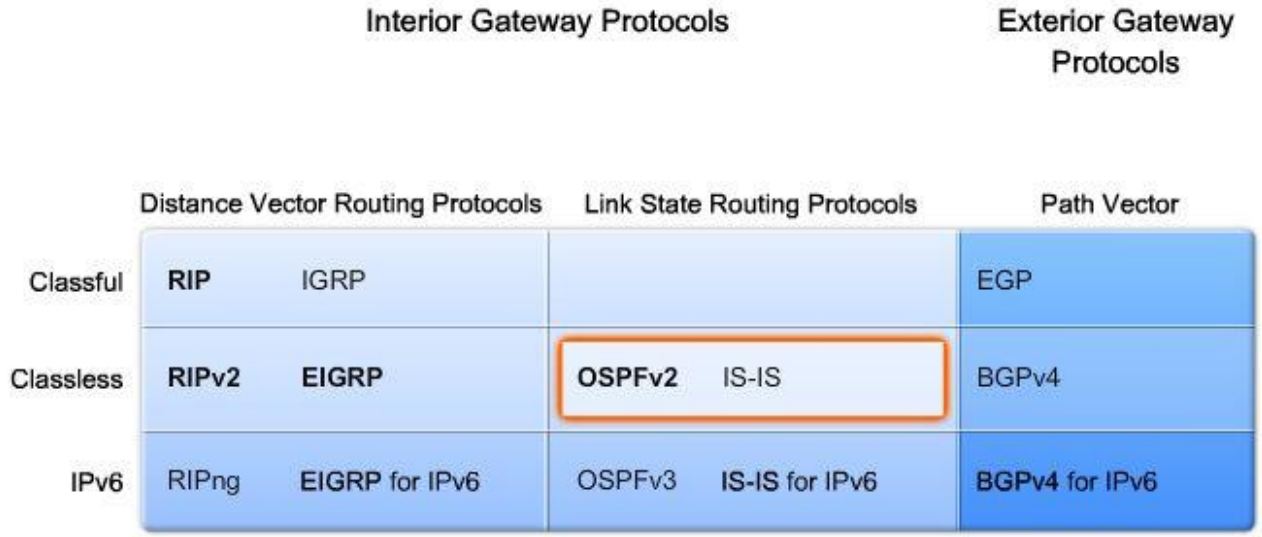


Рис. 7.1 Місце протоколів маршрутизації з урахуванням стану каналу в системі класифікації

Протоколи маршрутизації з урахуванням стану каналу мають репутацію більш складних, у порівнянні з дистанційно-векторними. Однак, основна функціональність і конфігурація протоколів маршрутизації стану каналу не є більш складними. Основні операції OSPF можуть бути сконфігуровані командою ***ospf process-id*** і операторами ***network***, що схоже на протоколи RIP та EIGRP.

7.1.2 Введення до SPF алгоритму

Алгоритм Дейкстри зазвичай називають алгоритмом вибору найкоротшого шляху SPF. Цей алгоритм накопичує вартості уздовж кожного шляху, від джерела до адресата.

На рис. 7.2 кожен шлях позначено значенням вартості. Вартість найкоротшого шляху для R2, щоб відправити пакет до LAN, приєднаної до R3, становить 27:

від R2 до R1 (20) + від R1 до R3 + від R3 до LAN(2) = 27

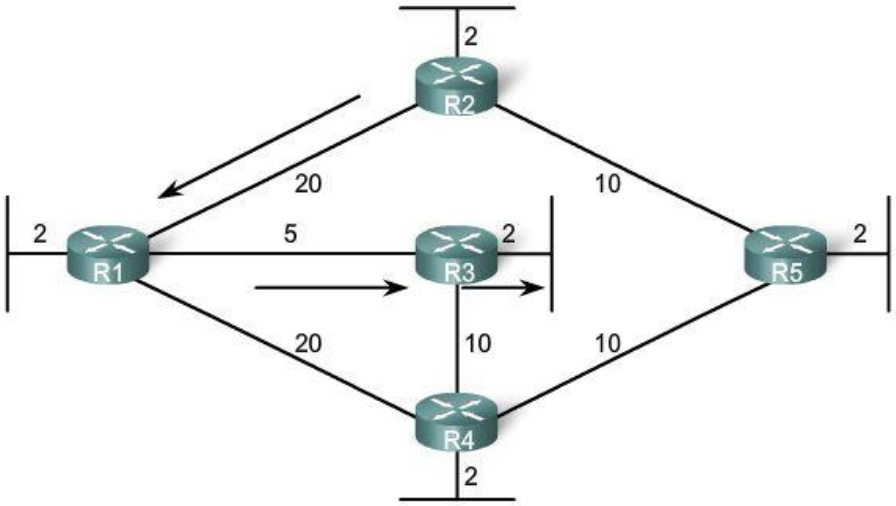
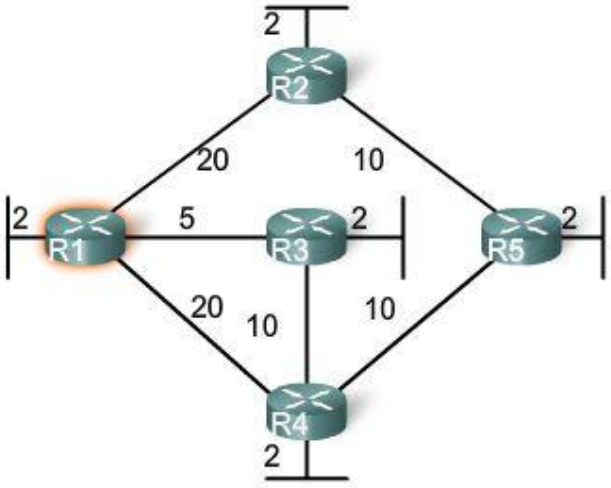


Рис. 7.2 Алгоритм SPF Дейкстри

Для інших маршрутизаторів вартість найкоротшого шляху до даної LAN буде іншою, оскільки кожний маршрутизатор виконує SPF алгоритм і визначає вартість зі своєї точки зору.

Для R1 найкоротший шлях до кожної LAN - поряд з вартістю - показаний у таблиці на рис 7.3. Зверніть увагу, що найкоротший шлях – це не обов'язково шлях з найменшою кількістю переходів (hops). Наприклад, розглянемо шлях до R4 LAN. Ви можете думати, що R1 буде вести відсилання безпосередньо до R4. Однак, вартість досягнення R4 LAN безпосередньо через R4 (22) вище, ніж вартість досягнення R4 LAN через R3 (17).



Destination	Shortest Path	Cost
R2 LAN	R1 to R2	22
R3 LAN	R1 to R3	7
R4 LAN	R1 to R3 to R4	17
R5 LAN	R1 to R3 to R4 to R5	27

Рис. 7.3 SPF дерево для R1

7.1.3 Процес маршрутизації з урахуванням стану каналу

Як же працює протокол маршрутизації з урахуванням стану каналу? Усі маршрутизатори в нашій топології мають завершити наступний основний процес маршрутизації з урахуванням стану каналу, щоб досягти стану конвергенції:

1. ***Кожний маршрутизатор вивчає свої власні канали, свої безпосередньо підключені мережі, щоб визначити, чи перебуває інтерфейс у стані «up».***
2. ***Кожний маршрутизатор відповідає за виявлення своїх сусідів на безпосередньо підключених мережах, для чого обмінюється з ними пакетами Hello.***
3. ***Кожний маршрутизатор формує Link-State Packet (LSP), що містить стан кожного безпосереднє підключеного каналу. Це виконується для реєстрації всієї доречної інформації про кожного сусіда, включаючи ID сусіда, тип каналу й пропускну здатність.***
4. ***Кожний маршрутизатор лавинним розсиланням (floods) поширює LSP усім сусідам, які зберігають всі отримані LSP у базі даних.*** Сусіди потім поширюють лавиною LSP своїм сусідам, поки всі маршрутизатори в області не одержать LSP. Кожний маршрутизатор запам'ятовує копію кожного LSP, отриманого від сусідів у локальній базі даних.
5. ***Кожний маршрутизатор використовує базу даних, щоб сконструювати повну карту топології й вирахувати найкращий шлях до кожної мережі призначення. Для цього використовується алгоритм SPF.***

Розглянемо цей процес більш докладно.

7.1.4 Вивчення безпосереднє підключених мереж

У топології на рис. 7.4 показані мережні адреси для кожного каналу. Кожний маршрутизатор вивчає свої власні канали, свої безпосередньо підключені мережі. Коли інтерфейс маршрутизатора сформований з IP адресою та маскою підмережі, інтерфейс стає частиною цієї мережі.

Коли Ви правильно конфігуруєте й активуєте інтерфейси, маршрутизатор довідається про свої власні безпосередньо підключені мережі. Незалежно від протоколів маршрутизації, які використовуються, ці безпосередньо підключені мережі стають частиною таблиці маршрутизації. Зараз ми зосередимося на процесі маршрутизації з урахуванням стану каналу з боку R1.

Link - канал

При роботі з протоколами маршрутизації стану каналу, канал (link) - це інтерфейс на маршрутизаторі. Інтерфейс повинен бути належним чином сконфігурований з IP адресою та маскою підмережі, до того ж канал має перебувати в стані «up», перед тим, як протокол маршрутизації з урахуванням стану каналу зможе довідатися про нього. Крім того, інтерфейс повинен бути включений в

один з операторів ***network*** перед тим, як він зможе брати участь у процесі маршрутизації з урахуванням стану каналу.

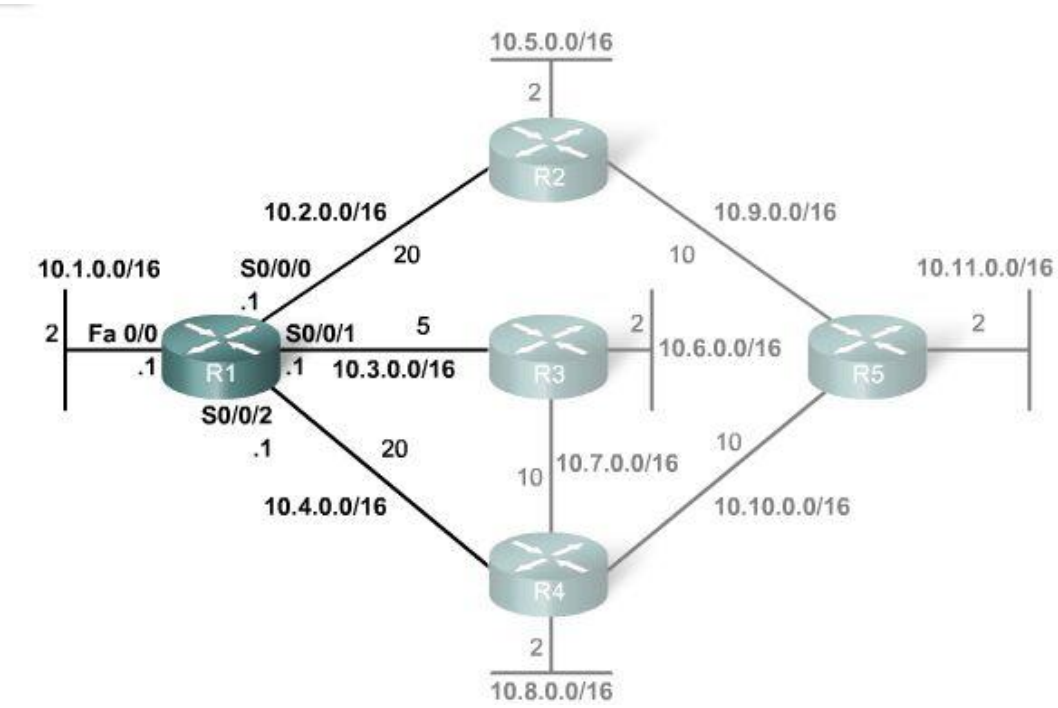


Рис. 7.4 Топологія мережі

На мал. 7.5 показаний R1, та його чотири безпосередньо підключені мережі.

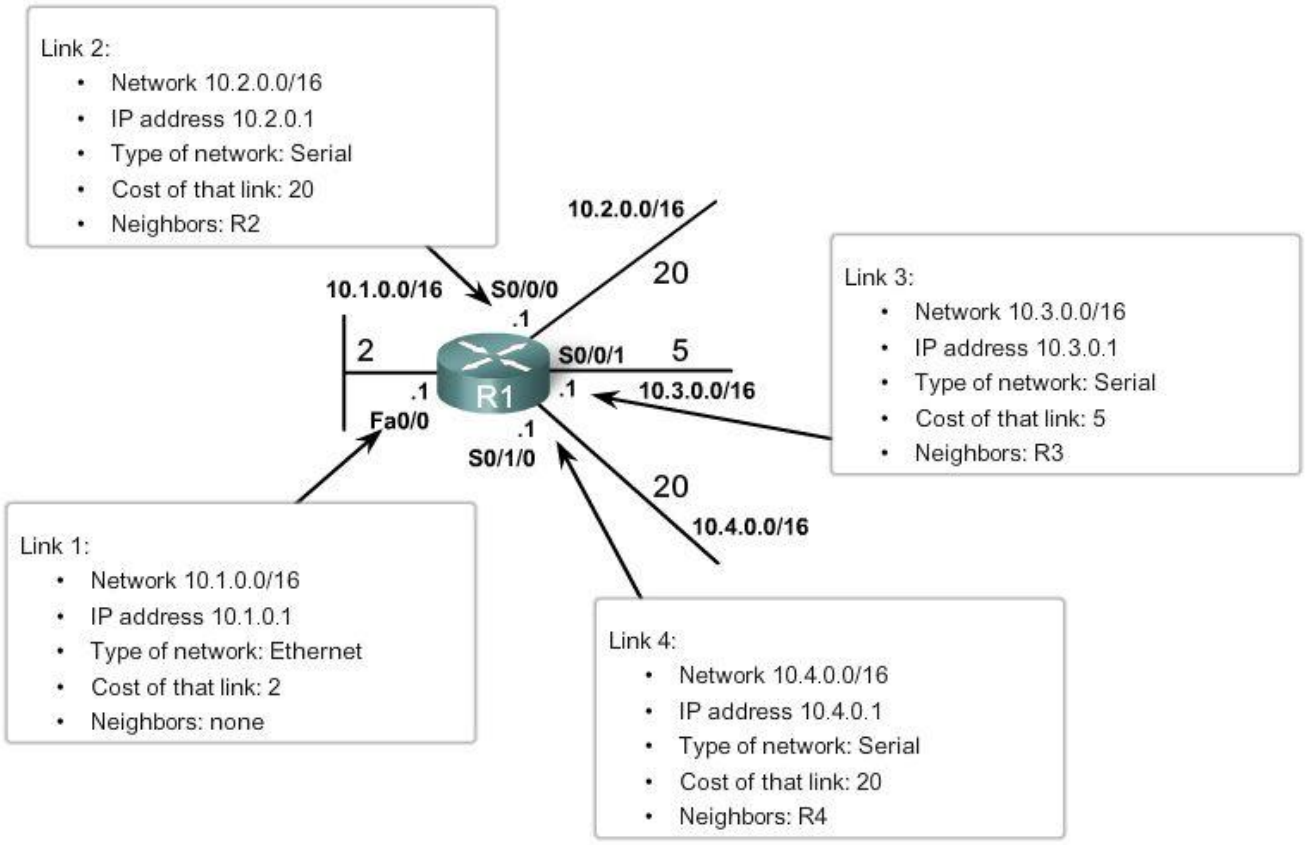


Рис. 7.5 Інформація стану каналу (link state) для R1

Link-State – стан каналу

Інформація про стан цих каналів відома як as ***link-state***. Як видно з рис. 7.5, ця інформація включає:

- ***IP адресу інтерфейсу й маску підмережі.***
- ***Тип мережі, наприклад Ethernet (broadcast) або Serial point-to-point link.***
- ***Вартість цього каналу.***
- ***Усі сусідні маршрутизатори на цьому каналі.***

Примітка: Далі ми побачимо, що Cisco-реалізація OSPF конкретизує вартість каналу, як пропускну здатність вихідного інтерфейсу. Однак, зараз для простоти ми використовуємо довільні значення вартості.

7.1.5 Надсилання Hello пакетів сусідам

Другий крок у процесі маршрутизації стану каналу:

Кожний маршрутизатор відповідає за виявлення своїх сусідів на безпосередньо підключених мережах.

Маршрутизатори, на яких виконуються протоколи маршрутизації з урахуванням стану каналу, використовують Hello протокол, щоб виявити сусідів на своїх каналах. ***Сусід (neighbor)*** – це будь-який інший маршрутизатор, на якому запущений такий самий протокол маршрутизації з урахуванням стану каналу.

На рис. 7.6 R1 посилає Hello пакети по всіх своїх каналах (інтерфейсам), щоб виявити, чи є на них сусіди.

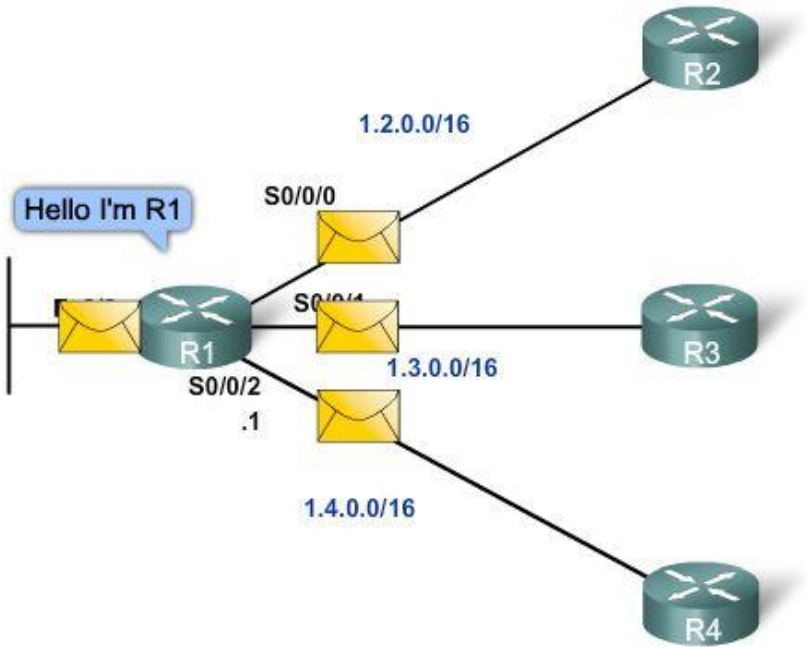


Рис. 7.6 Робота Hello протоколу

R2, R3, і R4 відповідають на Hello пакет своїм власним пакетом Hello, тому що ці маршрутизатори сконфігуровані з таким самим протоколом маршрутизації стану каналу. На інтерфейсі Fastethernet0/0 немає ніяких сусідів, тому R1 не одержить Hello на цьому інтерфейсі і не буде продовжувати виконання основного процесу маршрутизації з урахуванням стану каналу для інтерфейсу Fastethernet0/0.

Подібно EIGRP, коли два маршрутизатори з урахуванням стану каналу довідаються, що вони - сусіди, вони формують суміжність. Між двома суміжними сусідами триває обмін цими маленькими Hello пакетами, які виконують функцію "keepalive" для відстеження стану сусіда. Якщо маршрутизатор перестав одержувати Hello пакети від сусіда, цей сусід розглядається як недосяжний і суміжність розривається. На рис. 7.6 R1 формує суміжність з усіма трьома маршрутизаторами.

7.1.6 Формування пакета стану каналу (LSP)

Ми перебуваємо зараз на третьому кроці процесу маршрутизації з урахуванням стану каналу:

Кожний маршрутизатор формує Link-State Packet (LSP), що містить стан кожного безпосереднє підключеного каналу.

Як тільки маршрутизатор встановив суміжності, він може сформувати свої link-state packets (LSPs), які містять інформацію про стан його каналів. Спрощена версія LSP від R1 показана на рис. 7.7. Згадайте, що містить у собі LSP, використовуючи матеріал розділу 7.1.4.

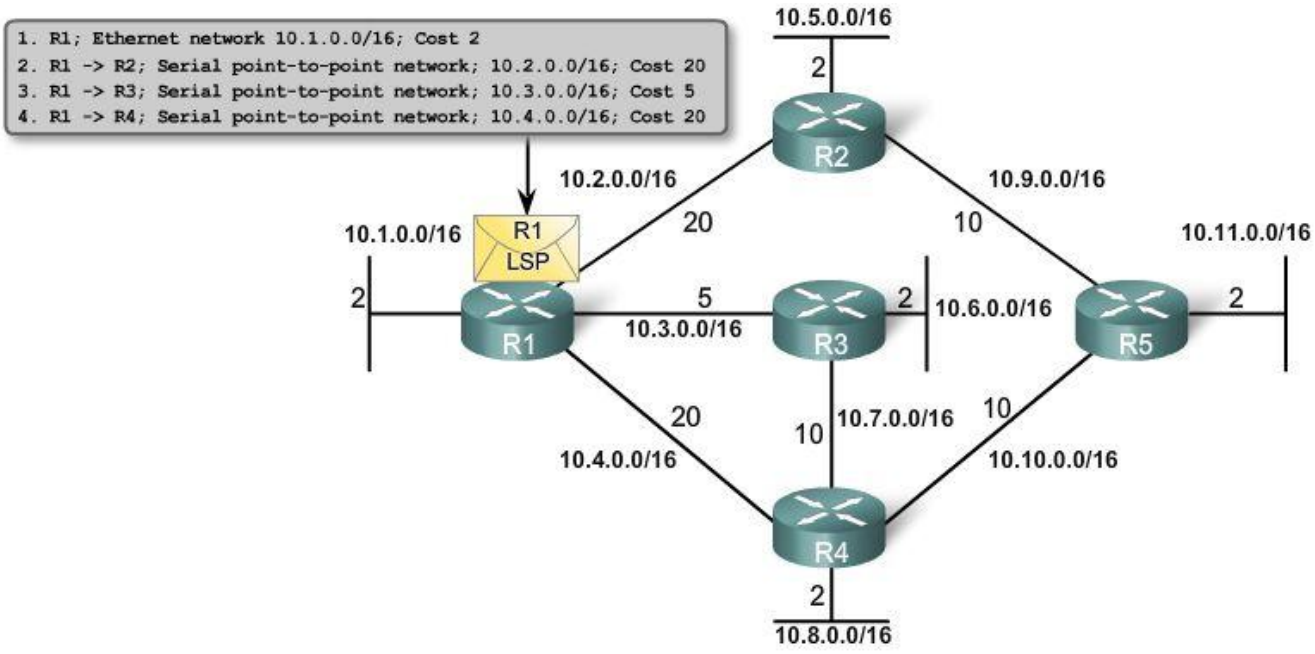


Рис. 7.7 R1 формує LSP

7.1.7 Лавинна передача пакетів стану каналу сусідам

Четвертий крок у процесі маршрутизації з урахуванням стану каналу:

Кожний маршрутизатор лавинним розсиланням (floods) поширює LSP усім сусідам, які зберігають всі отримані LSP у базі даних.

Кожний маршрутизатор поширює лавиною свою інформацію стану каналу всім іншим маршрутизаторам стану каналу в області маршрутизації.

Щораз, коли маршрутизатор одержує LSP від сусіднього маршрутизатора, він негайно розсилає цей LSP на всі інші інтерфейси, окрім інтерфейсу, на який одержав LSP. Цей процес створює ефект лавини LSPs від усіх маршрутизаторів в області маршрутизації.

LSP поширюються лавиною, майже одразу після одержання, без яких-небудь проміжних обчислень. На відміну від дистанційно-векторних протоколів маршрутизації, які повинні спочатку виконувати алгоритм Беллмана-Форда, щоб обробити оновлення маршрутизації перед відсиланням їх до інших маршрутизаторів, протоколи маршрутизації з урахуванням стану каналу проводять обчислення по SPF алгоритму після закінчення лавинного розсилання. У результаті, протоколи маршрутизації з урахуванням стану каналу досягають конвергенції набагато швидше, ніж дистанційно-векторні.

LSP не потрібно посилати періодично. LSP необхідно посилати тільки:

- ***Під час старту маршрутизатора або процесу протоколу маршрутизації на цьому маршрутизаторі.***
- ***Щораз, коли є зміни в топології, у тому числі зміна стану каналу («down» або «up»), у тому числі при встановленні або розриві суміжності із сусідами.***

На додаток до інформації стану каналу, в LSP включається інша інформація - наприклад порядкові номери й інформація про вік (aging), для керування процесом лавинного розсилання. Ця інформація використовується кожним маршрутизатором, щоб визначити, чи одержав він уже LSP від іншого маршрутизатора або LSP має більш нову інформацію, ніж та, що вже міститься в його базі даних стану каналу. Цей процес дозволяє маршрутизатору мати актуальну інформацію в базі даних стану каналу.

7.1.8 Побудова бази даних стани каналу

Останній крок у процесі маршрутизації стану каналу:

Кожний маршрутизатор використовує базу даних, щоб сконструювати повну карту топології й обчислити найкращий шлях до кожної мережі призначення.

Після того, як кожний маршрутизатор поширив свої власні LSPs, викори-

стовуючи процес лавинного розсилання стану каналу, кожний маршрутизатор буде мати LSP від кожного маршрутизатора стану каналу в області маршрутизації. Ці LSPs зберігаються в базі даних стану каналу. Кожний маршрутизатор в області маршрутизації може тепер ***використовувати SPF алгоритм для побудови SPF дерева.***

Кожний маршрутизатор у топології визначає найкоротші шляхи зі своєї власної перспективи.

7.1.9 SPF дерево

Побудова SPF Дерева

Розглянемо більш докладно, як R1 конструює своє SPF дерево. Поточна топологія R1 (рис. 7.8) включає тільки своїх сусідів. Однак, використовуючи інформацію стану каналу від усіх інших маршрутизаторів, R1 може зараз почати конструювати SPF дерево мережі, використовуючи себе у якості кореня дерева.

R1 Link State Database

R1 Links-states:
- Connected to neighbor R2 on network 10.2.0.0/16, cost of 20
- Connected to neighbor R3 on network 10.3.0.0/16, cost of 5
- Connected to neighbor R4 on network 10.4.0.0/16, cost of 20
- Has a network 10.1.0.0/16, cost of 2

LSPs from R2:
- Connected to neighbor R1 on network 10.2.0.0/16, cost of 20
- Connected to neighbor R5 on network 10.9.0.0/16, cost of 10
- Has a network 10.5.0.0/16, cost of 2

LSPs from R3:
- Connected to neighbor R1 on network 10.3.0.0/16, cost of 5
- Connected to neighbor R4 on network 10.7.0.0/16, cost of 10
- Has a network 10.6.0.0/16, cost of 2

LSPs from R4:
- Connected to neighbor R1 on network 10.4.0.0/16, cost of 20
- Connected to neighbor R3 on network 10.7.0.0/16, cost of 10
- Connected to neighbor R5 on network 10.10.0.0/16, cost of 10
- Has a network 10.8.0.0/16, cost of 2

LSPs from R5:
- Connected to neighbor R2 on network 10.9.0.0/16, cost of 10
- Connected to neighbor R4 on network 10.10.0.0/16, cost of 10
- Has a network 10.11.0.0/16, cost of 2

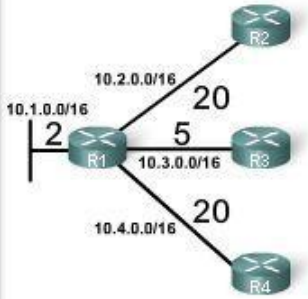


Рис. 7.8 Стани каналів R1

На рис. 7.9 зображені LSPs, які були отримані від R2 і результат їх обробки. R1 може ігнорувати перший LSP, тому що R1 уже знає, що він з'єднаний з R2 по мережі 10.2.0.0/16 з вартістю 20. R1 може використати другий LSP і створити канал від R2 до іншого маршрутизатора, R5, з мережею 10.9.0.0/16 і вартістю 10. Ця інформація додається до SPF дерева. Використовуючи третій LSP, R1 довідається, що R2 має мережу 10.5.0.0/16 з вартістю 2, в якій немає сусідніх маршрутизаторів. Цей канал також додається до R1 SPF дерева.

R1 Link State Database

R1 Links-states:
- Connected to neighbor R2 on network 10.2.0.0/16, cost of 20
- Connected to neighbor R3 on network 10.3.0.0/16, cost of 5
- Connected to neighbor R4 on network 10.4.0.0/16, cost of 20
- Has a network 10.1.0.0/16, cost of 2

LSPs from R2:
- Connected to neighbor R1 on network 10.2.0.0/16, cost of 20
- Connected to neighbor R5 on network 10.9.0.0/16, cost of 10
- Has a network 10.5.0.0/16, cost of 2

LSPs from R3:
- Connected to neighbor R1 on network 10.3.0.0/16, cost of 5
- Connected to neighbor R4 on network 10.7.0.0/16, cost of 10
- Has a network 10.6.0.0/16, cost of 2

LSPs from R4:
- Connected to neighbor R1 on network 10.4.0.0/16, cost of 20
- Connected to neighbor R3 on network 10.7.0.0/16, cost of 10
- Connected to neighbor R5 on network 10.10.0.0/16, cost of 10
- Has a network 10.8.0.0/16, cost of 2

LSPs from R5:
- Connected to neighbor R2 on network 10.9.0.0/16, cost of 10
- Connected to neighbor R4 on network 10.10.0.0/16, cost of 10
- Has a network 10.11.0.0/16, cost of 2

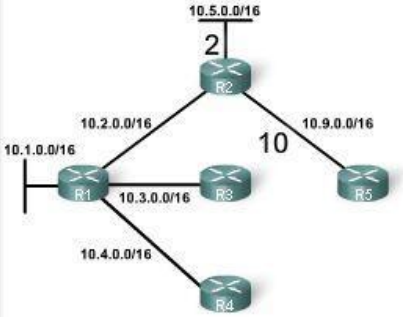


Рис. 7.9 Обробка LSPs, що отримані від R2

На рис. 7.10 зображено, як алгоритм SPF обробляє LSPs, які були отримані від R3 і результат їх обробки. R1 може пропустити перший LSP, тому що R1 уже знає, що він з'єднаний з R3 по мережі 10.3.0.0/16 з вартістю 5. R1 може використати другий LSP і створити канал від R3 до маршрутизатора R4, з мережею 10.7.0.0/16 і вартістю 10. Ця інформація додана до SPF. Використовуючи третій LSP, R1 довідався, що R3 має мережу 10.6.0.0/16 з вартістю 2 і без сусідів. Це посилання також додане до R1 SPF дерева.

R1 Link State Database

R1 Links-states:
- Connected to neighbor R2 on network 10.2.0.0/16, cost of 20
- Connected to neighbor R3 on network 10.3.0.0/16, cost of 5
- Connected to neighbor R4 on network 10.4.0.0/16, cost of 20
- Has a network 10.1.0.0/16, cost of 2

LSPs from R2:
- Connected to neighbor R1 on network 10.2.0.0/16, cost of 20
- Connected to neighbor R5 on network 10.9.0.0/16, cost of 10
- Has a network 10.5.0.0/16, cost of 2

LSPs from R3:
- Connected to neighbor R1 on network 10.3.0.0/16, cost of 5
- Connected to neighbor R4 on network 10.7.0.0/16, cost of 10
- Has a network 10.6.0.0/16, cost of 2

LSPs from R4:
- Connected to neighbor R1 on network 10.4.0.0/16, cost of 20
- Connected to neighbor R3 on network 10.7.0.0/16, cost of 10
- Connected to neighbor R5 on network 10.10.0.0/16, cost of 10
- Has a network 10.8.0.0/16, cost of 2

LSPs from R5:
- Connected to neighbor R2 on network 10.9.0.0/16, cost of 10
- Connected to neighbor R4 on network 10.10.0.0/16, cost of 10
- Has a network 10.11.0.0/16, cost of 2

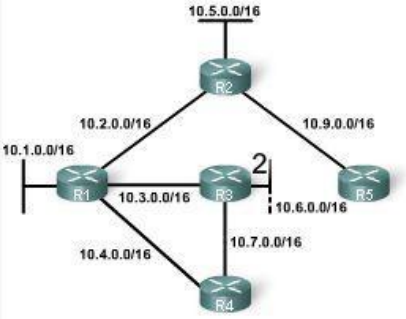


Рис. 7.10 Алгоритм SPF обробляє LSP, отримані від R3

На рис. 7.11 зображені LSPs, які були отримані від R4 і результат їх обробки. R1 може проігнорувати перший LSP, тому що R1 уже знає, що він з'єднаний з R4 по мережі 10.4.0.0/16 з вартістю 20. R1 може також проігнорувати другий LSP, тому що SPF вже довідався про мережу 10.6.0.0/16 з вартістю 10 від R3.

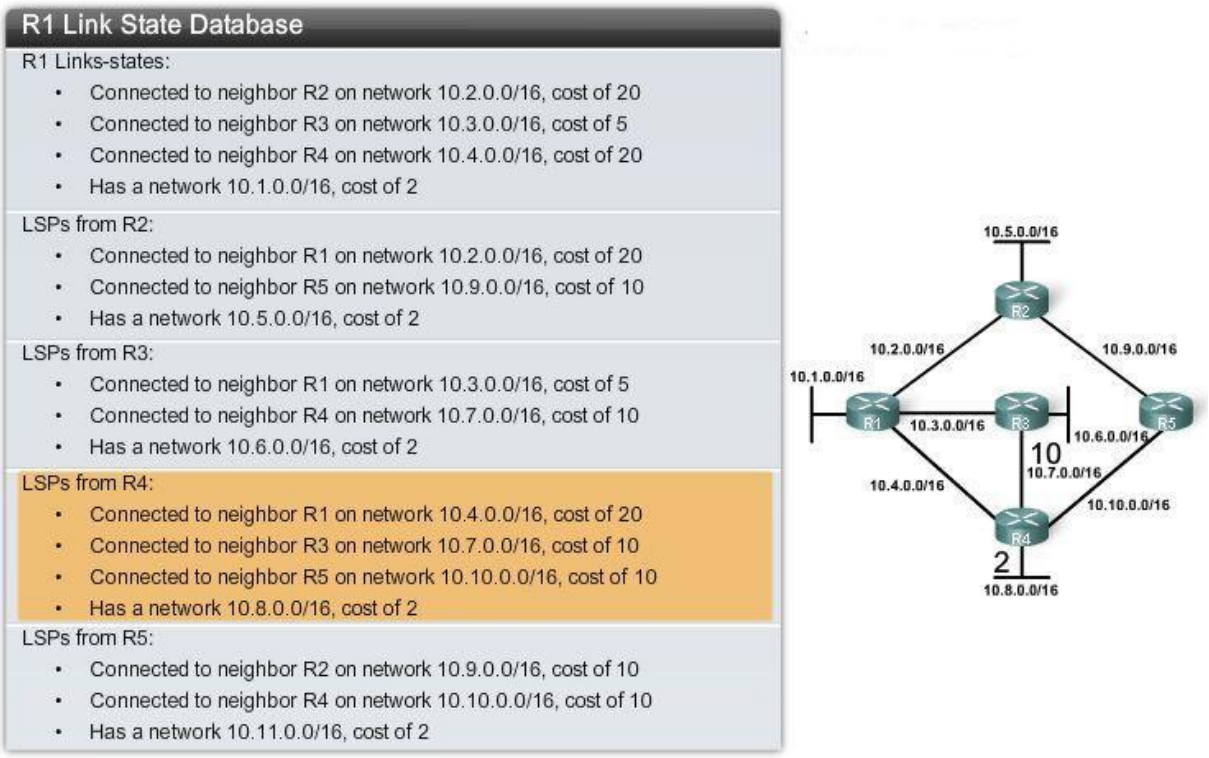


Рис. 7.11 Обробка LSPs, що отримані від R4

Однак, R1 використовує третій LSP, щоб створити канал від R4 до маршрутизатора R5, з мережею 10.10.0.0/16 і вартістю 10. Ця інформація додана до SPF дерева. Використовуючи четвертий LSP, R1 довідається, що R4 має мережу 10.8.0.0/16 з вартістю 2 і без сусідів. Це посилання також додане до R1 SPF дерева.

На рис. 7.12 зображено, як алгоритм SPF обробляє останні LSPs, які були отримані від R5. R1 може ігнорувати перші два LSPs (для мереж 10.9.0.0/16 і 10.10.0.0/16), оскільки SPF вже раніше довідався про ці канали й додав їх до SPF дерева. R1 може обробити третій LSP, щоб вивчити, що R5 має мережу 10.11.0.0/16 з вартістю 2 і без сусідів. Це посилання додане до SPF дерева для R1.

Визначення найкоротшого шляху (Shortest Path)

Оскільки всі LSP були оброблені SPF алгоритмом, R1 зараз сконструював повне SPF дерево. 10.4.0.0/16 і 10.9.0.0/16 канали не використовуються для досягнення інших мереж, тому що існують шляхи більш низької вартості. Однак ці мережі все ще існують як частина SPF дерева і використовуються, щоб дістатися пристроїв, розташованих у цих мережах.

R1 Link State Database

R1 Links-states:
- Connected to neighbor R2 on network 10.2.0.0/16, cost of 20
- Connected to neighbor R3 on network 10.3.0.0/16, cost of 5
- Connected to neighbor R4 on network 10.4.0.0/16, cost of 20
- Has a network 10.1.0.0/16, cost of 2

LSPs from R2:
- Connected to neighbor R1 on network 10.2.0.0/16, cost of 20
- Connected to neighbor R5 on network 10.9.0.0/16, cost of 10
- Has a network 10.5.0.0/16, cost of 2

LSPs from R3:
- Connected to neighbor R1 on network 10.3.0.0/16, cost of 5
- Connected to neighbor R4 on network 10.7.0.0/16, cost of 10
- Has a network 10.6.0.0/16, cost of 2

LSPs from R4:
- Connected to neighbor R1 on network 10.4.0.0/16, cost of 20
- Connected to neighbor R3 on network 10.7.0.0/16, cost of 10
- Connected to neighbor R5 on network 10.10.0.0/16, cost of 10
- Has a network 10.8.0.0/16, cost of 2

LSPs from R5:
- Connected to neighbor R2 on network 10.9.0.0/16, cost of 10
- Connected to neighbor R4 on network 10.10.0.0/16, cost of 10
- Has a network 10.11.0.0/16, cost of 2

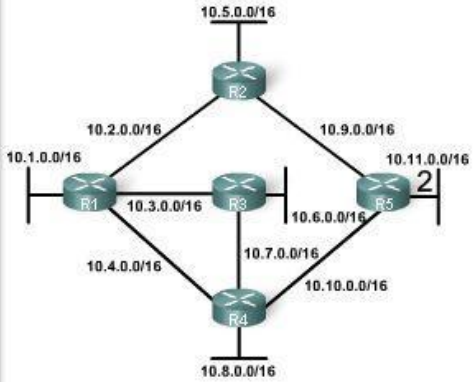
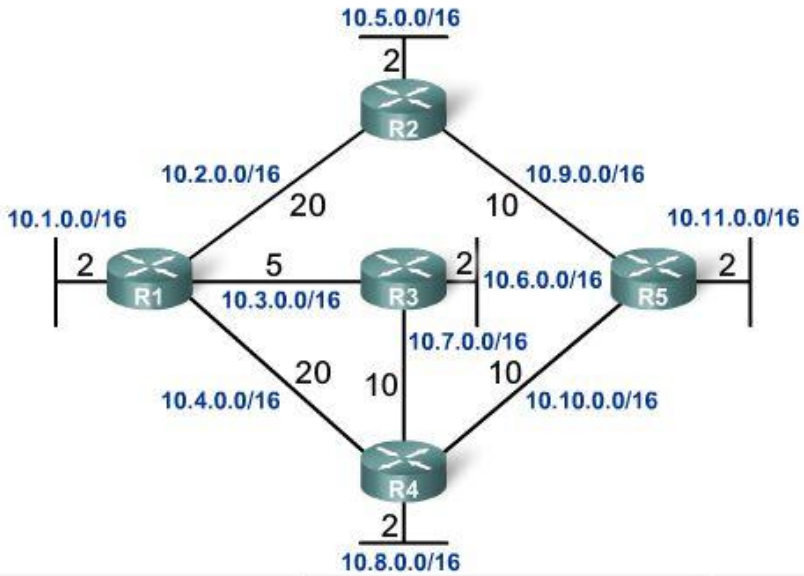


Рис. 7.12 Алгоритм SPF обробляє LSP, отримані від R5

На рис. 7.13 показано SPF дерево для R1. По цьому дереву визначені найкоротші шляхи до кожної мережі. Наприклад:

мережа 10.5.0.0/16 доступна через R2 serial 0/0/0 з вартістю 22



Destination	Shortest Path	Cost
R2 LAN	R1 to R2	22
R3 LAN	R1 to R3	7
R4 LAN	R1 to R3 to R4	17
R5 LAN	R1 to R3 to R4 to R5	27

Рис. 7.13 SPF дерево для R1

Кожний маршрутизатор конструює своє власне SPF дерево, незалежно від усіх інших маршрутизаторів.

Щоб гарантувати правильну маршрутизацію, бази даних стану каналу (link-state databases), які використовуються при побудові SPF дерев, повинні бути ідентичні на всіх маршрутизаторах.

Генерація таблиці маршрутизації з SPF дерева

Найкоротші шляхи, визначені SPF алгоритмом, додаються до таблиці маршрутизації. Таблиця маршрутизації буде також включати всі безпосередньо підключені мережі й маршрути, вивчені з інших джерел, наприклад статичні маршрути.

7.2 Впровадження протоколів маршрутизації з урахуванням стану каналу

7.2.1 Переваги протоколів маршрутизації з урахуванням стану каналу

У порівнянні з дистанційно-векторними, протоколи маршрутизації з урахуванням стану каналу мають низку переваг.

Будують топологічну карту

Протоколи маршрутизації з урахуванням стану каналу створюють топологічну карту, або SPF дерево мережної топології. Дистанційно-векторні протоколи маршрутизації не мають топологічної карти мережі. Маршрутизатори, що реалізують дистанційно-векторний протокол маршрутизації, мають тільки список мереж, який включає вартість (відстань) і next-hop маршрутизатор (напрямок) до мережі. Оскільки протоколи маршрутизації з урахуванням стану каналу обмінюються станами каналів, SPF алгоритм може побудувати SPF дерево мережі. ***Використовуючи SPF дерево, кожний маршрутизатор може незалежно визначити найкоротший шлях до кожної мережі.***

Швидка конвергенція

Одержуючи Link-state Packet (LSP), протоколи стану каналу негайно лавиною розсилають LSP на всі інтерфейси за винятком інтерфейсу, з якого LSP був отриманий. Маршрутизатору, що використовує дистанційно-векторний протокол маршрутизації, потрібно обробити кожне оновлення маршрутизації й модифікувати свою таблицю маршрутизації перед поширенням оновлення на всі інші інтерфейси, навіть якщо використовуються миттєві (triggered) оновлення. Більш швидка конвергенція досягається протоколами маршрутизації стану каналу. EIGRP – це скоріше виняток.

Оновлення, керовані подіями

Після початкового лавинного обміну LSP, протоколи маршрутизації з урахуванням стану каналу ***посилають LSP тільки при змінах у топології.***

LSP містить тільки інформацію про канал, який був порушений змінами. На відміну від деяких дистанційно-векторних протоколів маршрутизації, протоколи з урахуванням стану каналу не посилають періодичні оновлення.

Примітка: Маршрутизатори OSPF наводнюють лавинним розсиланням свої власні канали кожні 30 хвилин. Це відомо як параноїдальне оновлення й обговорюється в наступній Темі. Також, слід зазначити, що не всі дистанційно-векторні протоколи маршрутизації посилають періодичні оновлення. RIP і IGRP посилають періодичні оновлення; однак, EIGRP цього не робить.

Ієрархічне проектування

Протоколи маршрутизації стану каналу, такі, як наприклад OSPF і IS-IS використовують концепцію областей. Безліч областей приводить до ієрархічного проектування мереж. Це дозволяє використовувати кращу агрегацію маршрутів і ізолювати проблеми, що з'являються межами області.

7.2.2 Вимоги протоколів маршрутизації з урахуванням стану каналу

Сучасні протоколи маршрутизації з урахуванням стану каналу проектують так, щоб мінімізувати витрати пам'яті, процесорного часу і пропускної здатності. ***Використання й конфігурування множинних областей можуть скоротити розміри баз даних стану каналу. Множинні області можуть також обмежити кількість інформації про стан каналів, яка поширюється лавиною в області маршрутизації,*** і дозволяють посилати LSP тільки тим маршрутизаторам, яким вони потрібні.

Наприклад, коли є зміни в топології, тільки маршрутизатори області, що піддалася змінам, одержують LSP і виконують SPF алгоритм. Це може допомогти ізолювати нестабільний канал певною областю у домені маршрутизації. На рис. 7.14 показано три окремі домени (області) маршрутизації: Area 1, Area 0, і Area 51. Якщо мережа в Area 51 виходить із ладу, LSP з інформацією про цей канал лавиною поширюється тільки до інших маршрутизаторів у цій області. Тільки маршрутизаторам в Area 51 потрібно буде модифікувати їхні бази даних стану каналу, виконувати повторно SPF алгоритм, створювати нове SPF дерево, і модифікувати свої таблиці маршрутизації. Маршрутизатори в інших областях довідаються, що цей маршрут вийшов з ладу, але це буде зроблено LSP пакетом такого типу, який не змусить їх виконувати повторно SPF алгоритм. Маршрутизатори в інших областях зможуть модифікувати свої таблиці маршрутизації безпосередньо.

Вимоги до пам'яті

Протоколи маршрутизації з урахуванням стану каналу зазвичай вимагають більше ресурсів маршрутизатора, ніж дистанційно-векторні протоколи маршрутизації. ***Вимоги до пам'яті обумовлені використанням бази даних стану каналу й створенням SPF дерева.***

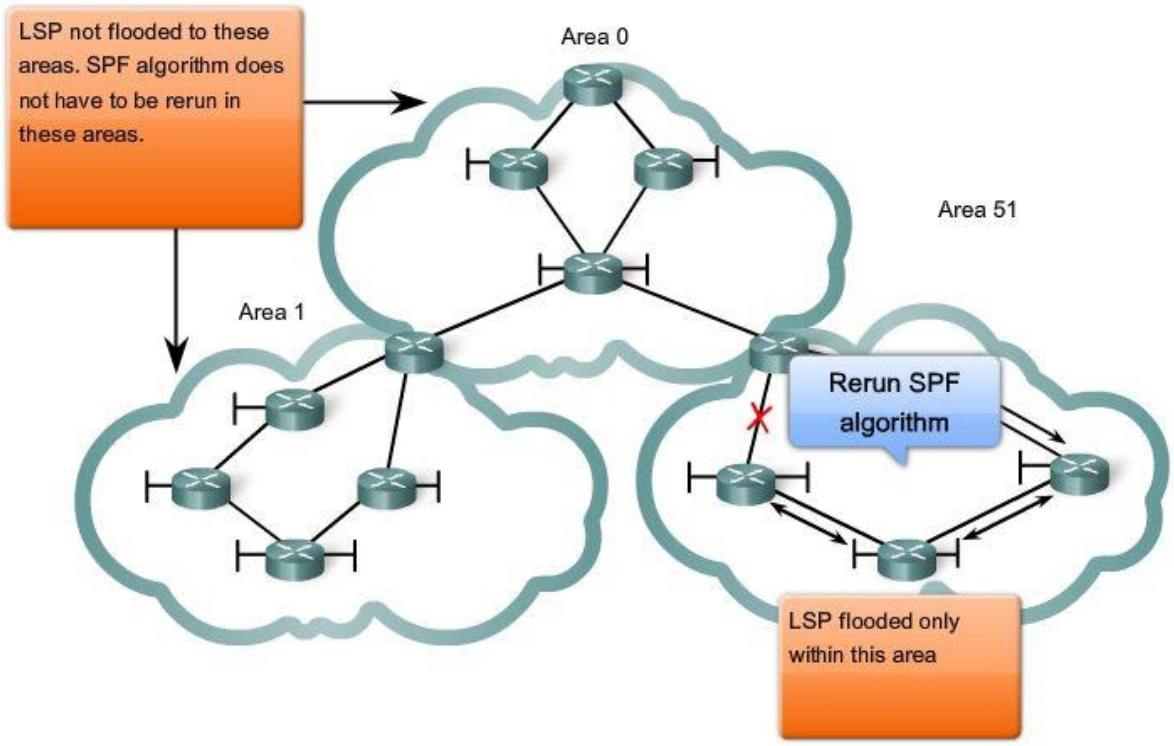


Рис. 7.14 Множинні області

Вимоги до процесора

Протоколи маршрутизації з урахуванням стану каналу можуть також вимагати більше процесорного часу, ніж дистанційно-векторні. Алгоритм SPF вимагає більше процесорного часу, ніж алгоритми дистанційно-векторні, як наприклад Беллмана-Форда, тому що протоколи маршрутизації стану каналу будують повну карту топології.

Вимоги по пропускній здатності

Лавинне поширення LSP може несприятливо позначитися на доступній пропускній здатності мережі. Це повинно відбуватися тільки під час початкового запуску маршрутизаторів, але може також бути проблемою в нестабільних мережах.

7.2.3 Порівняння протоколів маршрутизації стану каналу

Сьогодні для IP маршрутизації використовуються два протоколи з урахуванням стану каналу:

- Open Shortest Path First (OSPF)
- Intermediate System-to-Intermediate System (IS-IS)

OSPF

OSPF проектував IETF OSPF Working Group, яка усе ще існує сьогодні. Розробка OSPF почалася в 1987 році і використовується дві поточні версії:

- Ospfv2: OSPF для Ipv4 мереж (RFC 1247 і RFC 2328)
- Ospfv3: OSPF для Ipv6 мереж (RFC 2740)

OSPF буде описаний в наступній Темі, за винятком множинних областей.

IS-IS

IS-IS був розроблений ISO (International Organization for Standardization) і описаний в ISO 10589. Перша реалізація цього протоколу маршрутизації була розроблена DEC (Digital Equipment Corporation) і відома як Decnet Phase V.

IS-IS спочатку проектувався для стека протоколів OSI, а не для стека протоколів TCP/IP. Пізніше, інтегрований IS-IS, включав підтримку IP мереж. Хоча IS-IS відомий як протокол маршрутизації, який використовується більшістю провайдерів послуг Інтернет, сьогодні IS-IS починають використовувати в мережах великих підприємств.

OSPF і IS-IS мають як багато загальних рис, так і багато відмінностей. Обоє протоколів забезпечують необхідну функціональність маршрутизації.

7.3 Висновки

7.3.1 Резюме

Протоколи маршрутизації з урахуванням стану каналу також відомі як протоколи пошуку найкоротшого шляху й побудовані навколо SPF алгоритму Дейкстри.

Для IP існує два протоколи маршрутизації з урахуванням стану каналу: OSPF (Open Shortest Path First) і IS-IS (INTERMEDIATE-SYSTEM-TO-INTERMEDIATE-SYSTEM).

Процес маршрутизації з урахуванням стану каналу можна описати в такий спосіб:

1. Кожний маршрутизатор вивчає свої власні канали, свої безпосередньо підключені мережі.
2. Кожний маршрутизатор відповідає за обмін пакетами Hello зі своїми сусідам на безпосередньо підключених мережах.
3. Кожний маршрутизатор формує Link-State Packet (LSP), що містить стан кожного безпосереднє підключеного каналу.
4. Кожний маршрутизатор робить лавинне розсилання LSP усім своїм сусідам, які зберігають всі отримані LSP у базі даних.
5. Кожний маршрутизатор використовує базу даних стану каналу, щоб сконструювати повну карту топології й обчислити найкращий шлях до кожної мережі призначення.

Канал (link) - це інтерфейс на маршрутизаторі. ***Інформація стану каналу про цей інтерфейс включає IP-адресу, маску підмережі, тип мережі, вартість, пов'язану з каналом, і всі сусідні маршрутизатори на цьому каналі.***

Кожний маршрутизатор визначає стан своїх власних каналів і робить ла-

винне розсилання інформації до всіх інших маршрутизаторів в області. ***В результаті, кожний маршрутизатор будує базу даних стану каналу (LSDB), яка містить інформацію про стан каналів від усіх інших маршрутизаторів. Усі маршрутизатори будуть мати ідентичні LSDB.*** Використовуючи інформацію з LSDB, кожний маршрутизатор буде виконувати SPF алгоритм. Алгоритм SPF побудує SPF дерево, з маршрутизатором у корені. Оскільки кожний канал з'єднаний з іншими каналами, SPF дерево буде побудовано. Після побудови SPF дерева маршрутизатор може визначити по ньому свій власний кращий шлях до кожної мережі в дереві. Ця інформація про кращі шляхи потім зберігається в таблиці маршрутизації маршрутизатора.

Протоколи маршрутизації стану каналу будують локальну карту топології мережі, яка дозволяє кожному маршрутизатору визначити кращий шлях до заданої мережі. Новий LSP посилається тільки, коли є зміни в топології. Коли канал додається, видаляється або змінюється, маршрутизатор виконує лавинне розсилання нового LSP до всіх інших маршрутизаторів. Коли маршрутизатор одержує новий LSP, він модифікує LSDB, виконує повторно SPF алгоритм, створює нове SPF дерево, і модифікує свою таблицю маршрутизації.

Протоколи маршрутизації з урахуванням стану каналу мають тенденцію до більш швидкого часу конвергенції, ніж дистанційно-векторні. Винятком є EIGRP. Однак, протоколи маршрутизації з урахуванням стану каналу висувають більше вимог до пам'яті та процесорної обробки, що не є проблемою для більш нових, сучасних маршрутизаторів.

У наступній, останній Темі Ви вивчите протокол маршрутизації з урахуванням стану каналу OSPF.

7.3.2 Питання для самоперевірки

1. Чому дистанційно-векторні протоколи маршрутизації порівнюють із дорожніми покажчиками?
2. Чому протоколи з урахуванням стану каналу подібні використанню карти доріг?
3. Який алгоритм використовують протоколи маршрутизації з урахуванням стану каналу?
4. Що таке link ?
5. Що таке link-state?
6. Що називається сусідом і як відбувається виявлення сусідів?
7. Що таке лавинне розсилання стану каналу? Що є результатом такого розсилання?
8. Де зберігаються LSP і як вони використовуються?
9. Чи посилають періодичні оновлення протоколи маршрутизації з урахуванням стану каналу?

10. Які переваги протоколів маршрутизації стану каналу в порівнянні з дистанційно-векторними?
11. Які вимоги протоколів маршрутизації з урахуванням стану каналу і як ці вимоги можна мінімізувати?
12. Які два протоколи з урахуванням стану каналу використовуються сьогодні для IP маршрутизації?

7.3.3 Матеріали для самостійного поглибленого вивчення теми

Розуміння SPF алгоритму не є занадто складним. Існують книги й on-line ресурси, які пояснюють алгоритм Дейкстри та його використання в комп'ютерних мережах. Корисним буде розглянути наступні джерела, представлені англійською мовою:

- Interconnections, Bridges, Routers, Switches, and Internetworking Protocols, by Radia Perlman
- Cisco IP Routing, by Alex Zinin
- Routing the Internet, by Christian Huitema

Тема 8. OSPF

Ви навчитеся:

- Описувати основні можливості OSPF.
- Застосовувати основні команди для конфігурування OSPF.
- Описувати, модифікувати й вираховувати метрику, яка застосовуэться OSPF.
- Описувати процес вибору Designated Router/Backup Designated Router (BR/BDR) у мережах множинного доступу.
- Застосовувати команду ***default-information originate*** для конфігурування й поширення маршруту за замовчуванням в OSPF.

Протокол вибору найкоротшого маршруту OSPF - протокол маршрутизації з урахуванням стану каналу, який розроблявся як заміна дистанційно-векторного протоколу маршрутизації RIP. RIP був прийнятним протоколом маршрутизації під час створення мережі Інтернет, але те, що він покладається на число переходів, як на єдину метрику для вибору кращого маршруту швидко стало неприйнятним у великих мережах, для яких потрібне більш стійке рішення маршрутизації. OSPF - безкласовий протокол маршрутизації, який використовує концепцію областей для масштабованості. RFC 2328 визначає метрику OSPF як довільну величину, яка зветься вартістю. Cisco IOS у якості показника вартості OSPF використовує пропускну здатність.

Головні переваги OSPF над RIP - його швидка конвергенція і його масштабованість для великих мереж. У цій Темі Ви будете вивчати базову OSPF реалізацію й конфігурацію для однієї області.

8.1 Вступ до OSPF

8.1.1 Історія OSPF

Розробка OSPF почалася в 1987 році Internet Engineering Task Force (IETF) OSPF Working Group. У той час Інтернет був у значній мірі академічною й дослідницькою мережею, що була заснована Американським урядом.

В 1989 році специфікація для OSPFv1 була опублікована в RFC 1131. Було написано 2 реалізації: одна працювала на маршрутизаторах, а інша на робочих станціях Unix. Остання реалізація пізніше стала широко розповсюдженим Unix процесом, відомим, як GATED. OSPFv1 був експериментальним протоколом маршрутизації і ніколи не розгортався.

В 1991 році OSPFv2 був уведений в RFC 1247. OSPFv2 запропонував істотні технічні вдосконалення порівняно з OSPFv1. У той же час, ISO працювала над власним протоколом маршрутизації з урахуванням стану каналу, Intermediate System-to-Intermediate System (IS-IS). IETF обрав OSPF як свій рекомендований IGP (протокол внутрішнього шлюзу).

В 1998 році специфікація OSPFv2 була модифікована в RFC 2328, який є на сьогодні поточним RFC для OSPF.

В 1999 році OSPFv3 для Ipv6 був опублікований в RFC 2740.

8.1.2 Інкапсуляція OSPF повідомлення

Дані OSPF повідомлення инкапсулюються в пакет. Поле даних OSPF повідомлення може включати один з п'яти типів пакета OSPF.

Результат інкапсуляції показаний на рис. 8.1.

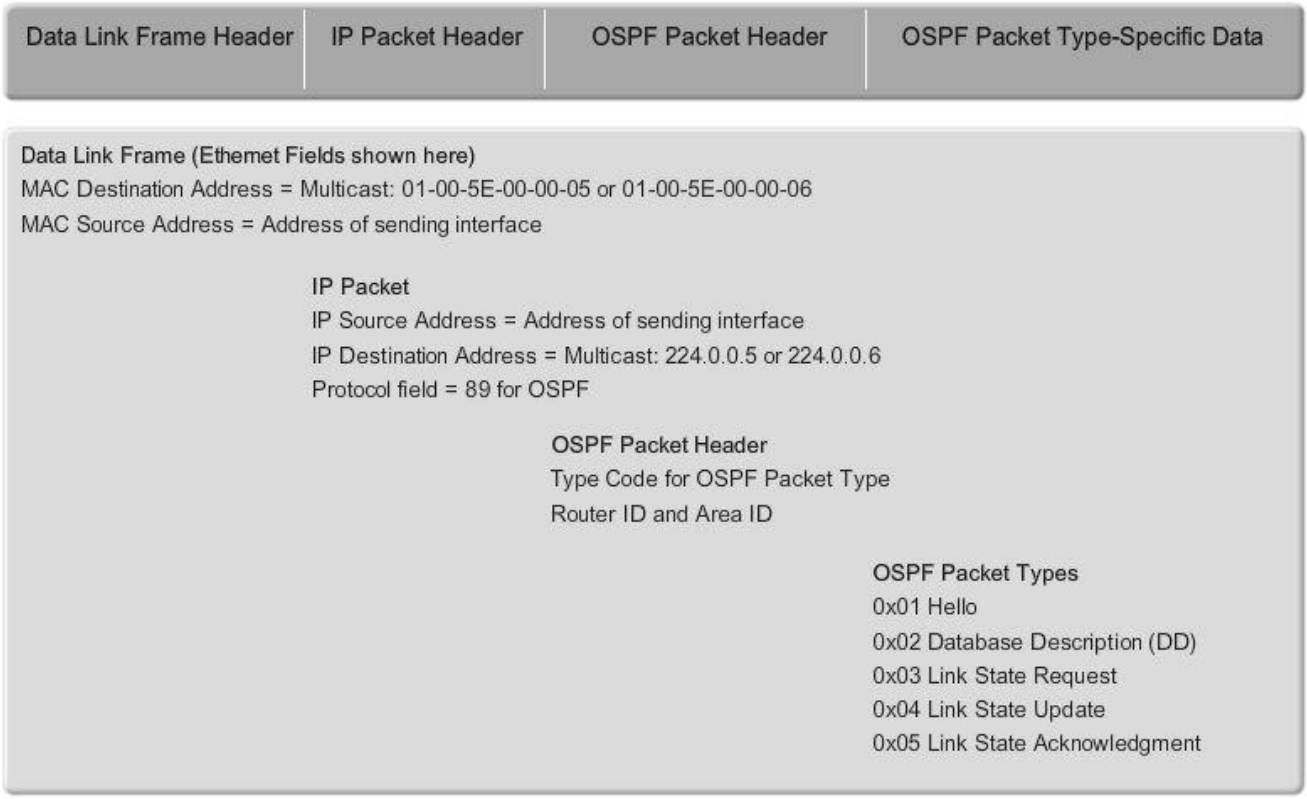


Рис. 8.1 Инкапсулюване OSPF повідомлення

Заголовок OSPF пакета входить до складу кожного пакета OSPF, незалежно від його типу. Заголовок OSPF пакета й специфічні для типу пакета дані потім инкапсуюються в IP пакет. У заголовку IP пакета, поле протоколу встановлене в 89, щоб позначити протокол OSPF, і адреса призначення встановлено в один із двох адрес групового розсилання: 224.0.0.5 або 224.0.0.6. Якщо OSPF пакет инкапсульовано у фрейм Ethernet, MAC адреса призначення - також адреса групового розсилання: 01-00-5E-00-00-05 або 01-00-5E-00-00-06.

8.1.3 Типи OSPF пакетів

У Темі 8 було введено поняття Link-State Packets (LSPs). Всього існує п'ять різних типів OSPF LSPs. Кожний пакет служить певній меті в процесі OSPF маршрутизації:

1. Hello - Hello пакети використовуються для встановлення і підтримки суміжності з іншими маршрутизаторами OSPF.
2. DBD - Database Description (DBD), пакет опису бази даних, містить ско-

рочений список бази даних стану каналу, що надсилає маршрутизатор. Маршрутизатори, що одержали, використовують його для порівняння з локальною базою даних стану каналу.

3. LSR - маршрутизатори, що одержують, можуть потім запросити більше інформації про будь-який запис в DBD шляхом відсилання Link-State Request (LSR) – запиту стану каналу.
4. LSU - Link-State Update (LSU), пакети оновлення стану каналу використовуються, щоб відповісти на LSRs, а також анонсувати нову інформацію. LSUs містять сім різних типів Link-State Advertisements (Lsas), анонсів стану каналу.
5. LSAck - Коли LSU отриманий, маршрутизатор посилає Link-State Acknowledgement (LSAck), щоб підтвердити одержання LSU.

8.1.4 Протокол Hello

На рис. 8.2 показаний заголовок OSPF пакета та Hello пакет.

OSPF пакета типу 1 - це Hello пакет. Hello пакет використовується для того, щоб:

- Виявити OSPF сусідів і встановити з ними суміжності.
- Анонсувати параметри, які два маршрутизатори повинні погодити, щоб стати сусідами.
- Вибрати відзначений маршрутизатор (Designated Router - DR) - і резервний відзначений маршрутизатор (Backup Designated Router - BDR) у мережах множинного доступу, подібних до Ethernet і Frame Relay.

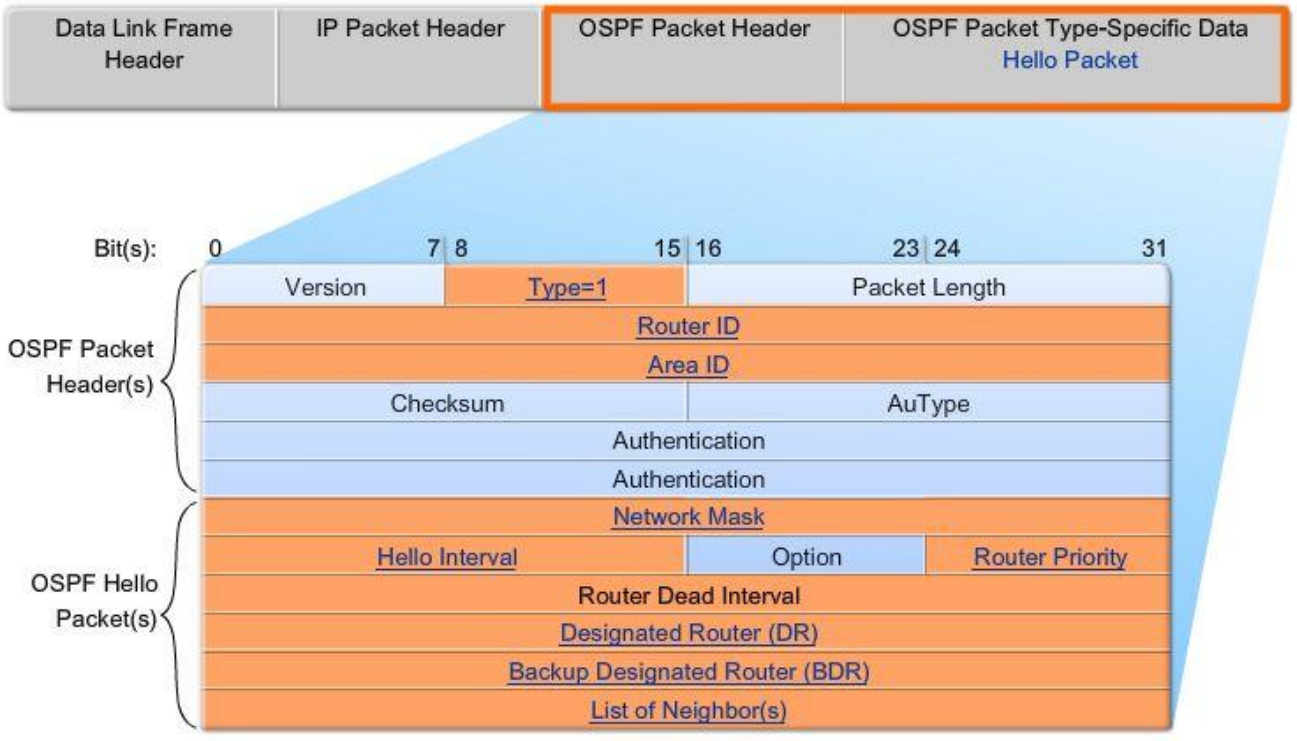


Рис.8.2 Формат OSPF повідомлення

Важливі поля, показані на рис. 8.2:

- ***Type:*** Тип OSPF пакета: Hello (1), DD (2), LS Request (3), LS Update (4), LS ACK (5).
- ***Router ID***: ID маршрутизатора, що породжує цю інформацію.
- ***Area ID***: область, з якої посланий пакет.
- ***Network Mask***: Маска підмережі, пов'язана з інтерфейсом, що відсилає.
- ***Hello Interval***: інтервал у секундах між надсиланням Hello пакетів маршрутизатором, що відсилає.
- ***Router Priority***: Використовується при виборах DR/BDR
- ***Designated Router (DR):*** Router ID відзначеного маршрутизатора, якщо він обраний.
- ***Backup Designated Router (BDR)***: Router ID резервного відзначеного маршрутизатора, якщо він обраний.
- ***List of Neighbors:*** список OSPF Router ID сусідніх маршрутизаторів

Визначення сусідів

Перш ніж OSPF маршрутизатор зможе лавиною поширювати свої стани каналу до інших маршрутизаторів, він повинен визначити, чи є OSPF сусіди на кожному з його каналів. На рис. 8.3, OSPF маршрутизатори посилають Hello пакети на всі OSPF-дозволені інтерфейси, щоб визначити, чи є сусіди на цих каналах. В OSPF Hello включається інформація про OSPF Router ID маршрутизатора, що надіслав даний пакет Hello. Одержання OSPF Hello пакета на інтерфейсі підтверджує для маршрутизатора що на цьому каналі є інший маршрутизатор OSPF. Тоді OSPF встановлює суміжність із сусідом. Наприклад, на рис. 8.3 R1 встановлює суміжність із R2 та R3.

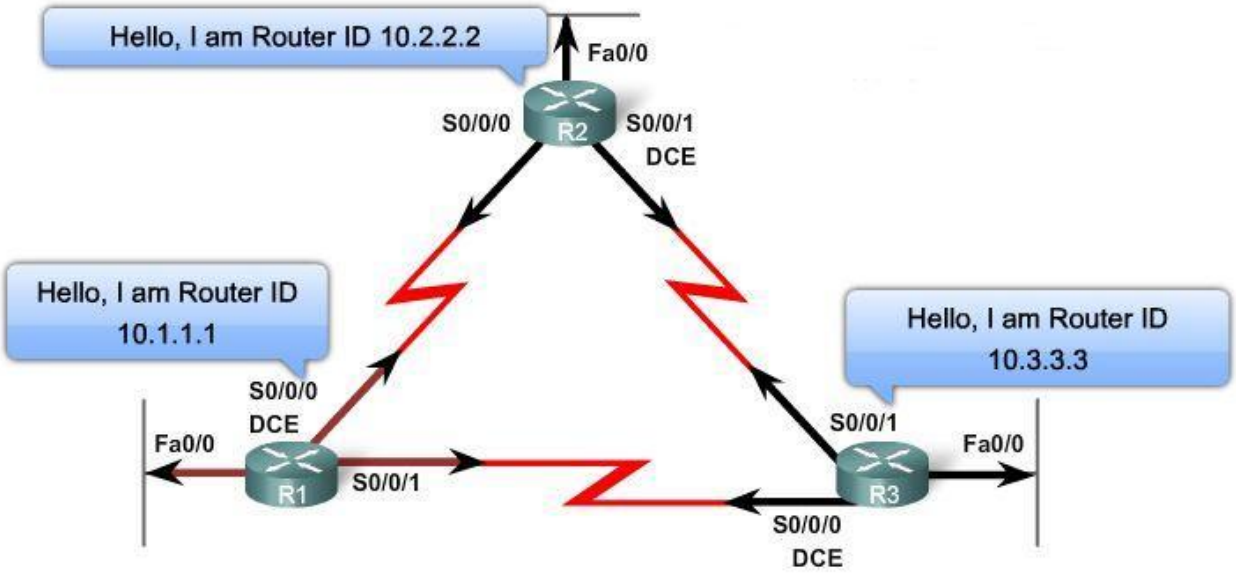


Рис.8.3 Hello протокол

OSPF Hello інтервал і Dead інтервал

Перед тим, як два маршрутизатори зможуть сформувати OSPF суміжність, вони повинні ***домовитися про три значення: Hello interval, Dead interval, і тип мережі.*** OSPF Hello interval показує, як часто маршрутизатор OSPF передає Hello пакети. За замовчуванням, OSPF Hello пакети надсилаються кожні 10 секунд на мережах множинного доступу та двокрапкових сегментах і кожні 30 секунд на не широкомовних (NBMA) сегментах множинного доступу (Frame Relay, X.25, ATM).

У більшості випадків, OSPF Hello packets розсилаються груповою розсилкою на адресу, зарезервовану для AllOSPFrouters - 224.0.0.5. Використання групової адреси дозволяє пристрою ігнорувати пакет, якщо його інтерфейс не має дозволу приймати OSPF пакети. Це зберігає процесорний час на не-OSPF пристроях.

Dead interval - період, виражений у секундах, протягом якого маршрутизатор буде очікувати одержання Hello пакета перед оголошенням сусіда як такого, що перейшов у стан "down." Cisco використовує значення за замовчуванням у чотири рази більше, ніж Hello інтервал. Для множинного доступу й двокрапкових сегментів, цей період становить 40 секунд. Для мереж NBMA, Dead interval становить 120 секунд.

Якщо Dead interval минає до того, як маршрутизатор одержить Hello пакет, OSPF вилучить сусіда зі своєї бази даних стану каналу. Маршрутизатор лавиною поширює інформацію про стан каналу сусіда у стані "down" на всі OSPF припустимі інтерфейси.

Вибори DR і BDR

Щоб зменшити OSPF трафік на мережах множинного доступу, OSPF вибирає відзначений маршрутизатор (DR) і резервний відзначений маршрутизатор (BDR). DR відповідає за оновлення всіх інших OSPF маршрутизаторів (називаються DRothers), коли в мережі множинного доступу з'являються зміни. BDR відслідковує DR і сам стає DR, якщо поточний DR виходить з ладу.

На рис. 8.3, R1, R2, і R3 підключені через двокрапкові канали. Тому, ніяких виборів DR/BDR не відбувається.

8.1.5 OSPF оновлення стану каналу

Link-state updates (LSUs) - пакети, які використовуються для OSPF маршрутних оновлень. Пакет LSU може містити 11 різних типів Link-State Advertisements (LSAs).. LSU містить один або більше LSAs і кожен з цих двох термінів може використовуватися відносно інформації стану каналу, яка розповсюджується маршрутизаторами OSPF.

8.1.6 OSPF алгоритм

Кожний маршрутизатор OSPF підтримує базу даних стану каналу, що мі-

стить LSAs, отримані від усіх інших маршрутизаторів. Як тільки маршрутизатор одержав усі LSAs і побудував свою локальну базу даних стану каналу, OSPF використовує алгоритм Дейкстри SPF, щоб побудувати SPF дерево. Дерево SPF потім використовується, щоб заповнити IP таблицю маршрутизації кращими шляхами до кожної мережі.

8.1.7 Адміністративна відстань

Адміністративна відстань (AD) - це ступінь надійності (або перевага) джерела маршруту. OSPF має задану за замовчуванням ***адміністративну відстань 110.*** Як видно з рис. 8.4 OSPF надається більше переваги у порівнянні з IS-IS і RIP.

Route Source	Administrative Distance
Connected	0
Static	1
EIGRP summary route	5
External BGP	20
Internal EIGRP	90
IGRP	100
OSPF	110
IS-IS	115
RIP	120
External EIGRP	170
Internal BGP	200

Рис. 8.4 Адміністративні відстані, прийняті за замовчуванням

8.1.8 Аутентифікація

Подібно іншим протоколам маршрутизації, OSPF може бути сформований для аутентифікації.

Це - гарна практика, щоб засвідчити передану маршрутну інформацію. Протоколи RIPv2, EIGRP, OSPF, IS-IS, і BGP можуть бути сконфігуровані таким чином, щоб шифрувати й засвідчувати свою маршрутну інформацію. Ця практика гарантує, що маршрутизатори приймуть маршрутну інформацію тільки від інших маршрутизаторів, які були сконфігуровані з таким самим паролем або аутентифікаційною інформацією.

Примітка: Аутентифікація не кодує таблицю маршрутизації маршрутизатора.

8.2 Базова конфігурація OSPF

8.2.1 Лабораторна топологія

На рис. 8.5 показана топологія для вивчення даної теми. Зверніть увагу,

що адресна схема (рис. 8.6) є несуміжною. OSPF - безкласовий протокол маршрутизації, тому проблем із несуміжними мережами виникнути не повинно. Також зверніть увагу, що в топології є три серійні канали різних пропускних здатностей і що кожний маршрутизатор має безліч шляхів до кожної віддаленої мережі. У даний момент на всіх серійних каналах встановлена пропускна здатність 1544kbps за замовчуванням.

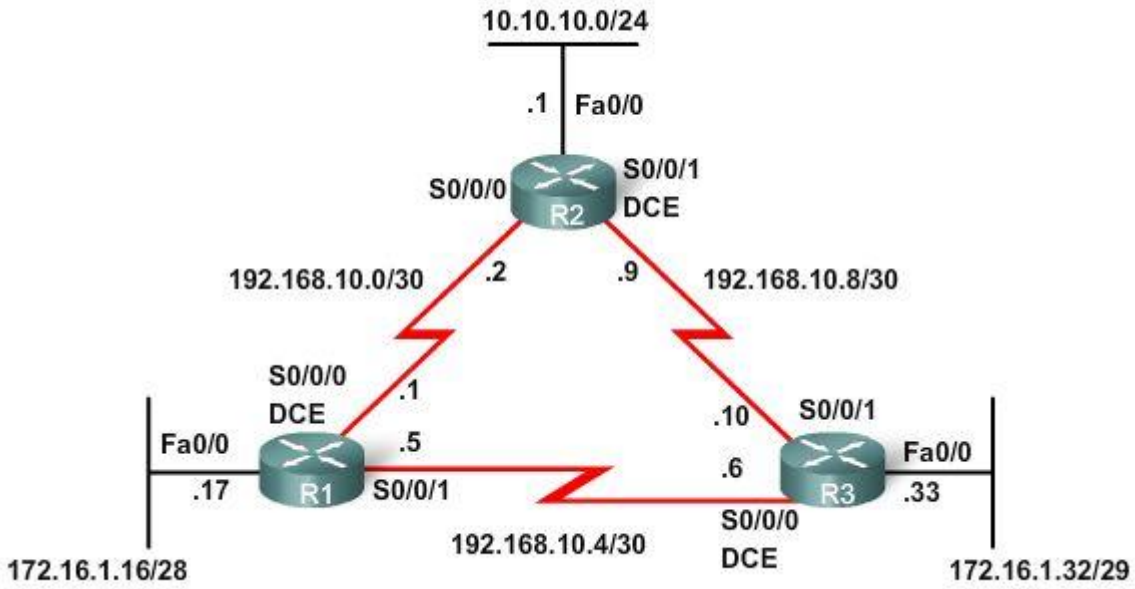


Рис.8.5 Топологія для розгляду

Device	Interface	IP Address	Subnet Mask
R1	Fa0/0	172.16.1.17	255.255.255.240
	S0/0/0	192.168.10.1	255.255.255.252
	S0/0/1	192.168.10.5	255.255.255.252
R2	Fa0/0	10.10.10.1	255.255.255.0
	S0/0/0	192.168.10.2	255.255.255.252
	S0/0/1	192.168.10.9	255.255.255.252
R3	Fa0/0	172.16.1.33	255.255.255.248
	S0/0/0	192.168.10.6	255.255.255.252
	S0/0/1	192.168.10.10	255.255.255.252

Рис. 8.6 Адресна схема

8.2.2 Команда router ospf

OSPF дозволяється на маршрутизаторі командою ***router ospf process-id y*** режимі глобальної конфігурації***. process-id*** є номером між 1 і 65535 і вибирається мережним адміністратором. ***process-id*** локально важливий, тобто він не повинен узгоджуватися з іншими маршрутизаторами, щоб встановити OSPF суміжність. Це відрізняється від EIGRP. EIGRP process ID, або номер автономної системи обов'язково повинен співпадати у двох сусідів EIGRP, щоб вони могли

сформувати суміжність.

У нашій топології, ми допускаємо OSPF на всіх трьох маршрутизаторах, використовуючи той самий process ID = 1.

*R1(config)#router **ospf 1***
R1(config-router)#

8.2.3 Команда network

Команда ***network,*** що використовується для OSPF, має таке ж функціональне навантаження, як при використанні з іншими IGP протоколами маршрутизації:

Будь-які інтерфейси на маршрутизаторі, які збігаються з мережною адресою в команді ***network***, одержують можливість посилати й одержувати OSPF пакети. Ця мережа (або підмережа) увійде в OSPF оновлення маршрутизації.

Команда ***network*** використовується в режимі конфігурації маршрутизації.

*Router(config-router)#**network** network-address wildcard-mask **area** area-id*

Команда ***network*** OSPF використовує комбінацію мережної адреси й ***wildcard-mask***, подібно EIGRP. На відміну від EIGRP, однак, в OSPF наявність ***wildcard mask*** є необхідною. Мережна адреса поряд з wildcard-mask використовується, щоб конкретизувати інтерфейс або діапазон інтерфейсів, які будуть допущені для OSPF, використовуючи дану команду ***network***.

Як і при роботі з EIGRP, wildcard mask може бути сформована шляхом інвертування маски підмережі. Наприклад, на R1 Fastethernet 0/0 інтерфейс належить мережі 172.16.1.16/28. Маска підмережі для цього інтерфейсу /28 або 255.255.255.240. Інвертування цієї маски дає wildcard mask.

```
  255.255.255.255
- 255.255.255.240
  ---------------
    0.   0.   0.  15        Wildcard mask
```

Примітка: Подібно EIGRP, деякі версії IOS дозволяють вам просто ввести маску підмережі замість wildcard mask. IOS тоді перетворює маску підмережі в wildcard mask формат.

Область ***area-id*** посилається на область OSPF. Область OSPF – це група маршрутизаторів, які обмінюються інформацією про стан каналу. Усі маршрутизатори в одній області OSPF повинні мати однакові бази даних стану каналу. Це досягається тим, що маршрутизатори лавиною розсилають свої стани каналу до всіх інших маршрутизаторів в області. У даній темі всі маршрутизатори OSPF будуть сформовані у межах однієї, єдиної області OSPF.

Мережа OSPF також може бути сконфігурована для безлічі областей. Є кілька переваг у тому, щоб конфігурувати великі OSPF мережі як множинні області. Це дає нам більш маленькі бази даних стану каналу й ізолювання про-

блем мережної нестабільності в межах області.

Якщо всі маршрутизатори перебувають у межах однієї області OSPF, ***network*** команди мають бути сформовані з однаковим ***area-id*** на всіх маршрутизаторах. Можна використовувати будь-який ***area-id***, але гарною практикою є для єдиної області OSPF використовувати ***area-id 0***. Ця угода дозволяє легше перейти на множинні області OSPF, де область 0 стає опорною (backbone) областю.

На рис. 8.7 показані команди ***network*** для всіх трьох маршрутизаторів, що дозволяють OSPF на всіх інтерфейсах. Усі маршрутизатори повинні зараз успішно пінгувати усі мережі.

```
R1(config)#router ospf 1
R1(config-router)#network 172.16.1.16 0.0.0.15 area 0
R1(config-router)#network 192.168.10.0 0.0.0.3 area 0
R1(config-router)#network 192.168.10.4 0.0.0.3 area 0
```

```
R2(config)#router ospf 1
R2(config-router)#network 10.10.10.0 0.0.0.255 area 0
R2(config-router)#network 192.168.10.0 0.0.0.3 area 0
R2(config-router)#network 192.168.10.8 0.0.0.3 area 0
```

```
R3(config)#router ospf 1
R3(config-router)#network 172.16.1.32 0.0.0.7 area 0
R3(config-router)#network 192.168.10.4 0.0.0.3 area 0
R3(config-router)#network 192.168.10.8 0.0.0.3 area 0
```

Рис. 8.7 Конфігурування підмереж OSPF

8.2.4 OSPF Router ID

Визначення Router ID

OSPF ***router ID*** використовується, щоб унікально ідентифікувати кожний маршрутизатор в OSPF домені маршрутизації. ***router ID*** - це просто IP-адреса. Маршрутизатори Cisco встановлюють ***router ID***, ґрунтуючись на трьох критеріях з наступною перевагою:

1. Використовується IP адреса, сформована командою OSPF ***router-id*** .
2. Якщо ***router-id*** не сформований, маршрутизатор вибирає саму старшу IP адресу кожного з loopback інтерфейсів.
3. Якщо немає сконфігурованих loopback інтерфейсів, маршрутизатор вибирає саму старшу активну IP адресу кожного з його фізичних інтерфейсів.

Сама старша активна IP адреса

Якщо маршрутизатор OSPF не сконфігурований командою OSPF ***router-id*** і немає сконфігурованих ***loopback*** інтерфейсів, OSPF ***router ID*** буде самою

старшою активною IP адресою на кожному з інтерфейсів маршрутизатора. Інтерфейс не обов'язково повинен бути дозволений для OSPF, тобто мав бути використаний в одній з команд OSPF ***network***. Однак, інтерфейс повинен бути активний - він повинен перебувати в стані «up» .

Використовуючи критерії, описані вище, спробуйте визначити ***router ID*** для R1, R2, і R3 по рис. 8.5. Оскільки ми не сконфігурували явно ***router ID*** або ***loopback*** інтерфейси на всіх трьох маршрутизаторах, ***router ID*** для кожного маршрутизатора визначається за третім критерієм: сама старша активна IP адреса кожного з фізичних інтерфейсів маршрутизатора.

Перевірити свої результати можна за допомогою команди ***show ip protocols*** (рис. 8.8). Крім того, для перевірки ***router ID*** можна використовувати команди ***show ip ospf*** і ***show ip ospf interface.***

```
R1#show ip protocols
Routing Protocol is "ospf 1"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Router ID 192.168.10.5
  Number of areas in this router is 1. 1 normal 0 stub 0 nssa
***output omitted***

R2#show ip protocols
Routing Protocol is "ospf 1"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Router ID 192.168.10.9
  Number of areas in this router is 1. 1 normal 0 stub 0 nssa
***output omitted***

R3#show ip protocols
Routing Protocol is "ospf 1"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Router ID 192.168.10.10
  Number of areas in this router is 1. 1 normal 0 stub 0 nssa
***output omitted***
```

Рис. 8.8 Перевірка router ID за допомогою команди show ip protocols

Loopback Адреса

Якщо команда OSPF ***router-id*** не використовується, а loopback інтерфейси сформовані, OSPF вибере саму старшу IP-адресу кожного з його loopback интерфейсів. loopback є віртуальним інтерфейсом і автоматично перебуває в стані «up», якщо сконфігурований. Ви вже знаєте команди для конфігурування loopback інтерфейсу:

*Router(config)#**interface loopback** number*
*Router(config-if)#**ip address** ip-address subnet-mask*

На рис. 8.9 зображена топологія, в якій усі три маршрутизатори були сконфігуровані з loopback адресами, щоб представляти OSPF ***router Ids***. Перевага використання loopback інтерфейсу в тому, що на відміну від фізичних інтерфейсів, він не може вийти з ладу. Тому, використовуючи loopback адреси для ***router ID,*** ми забезпечуємо стабільність процесу OSPF.

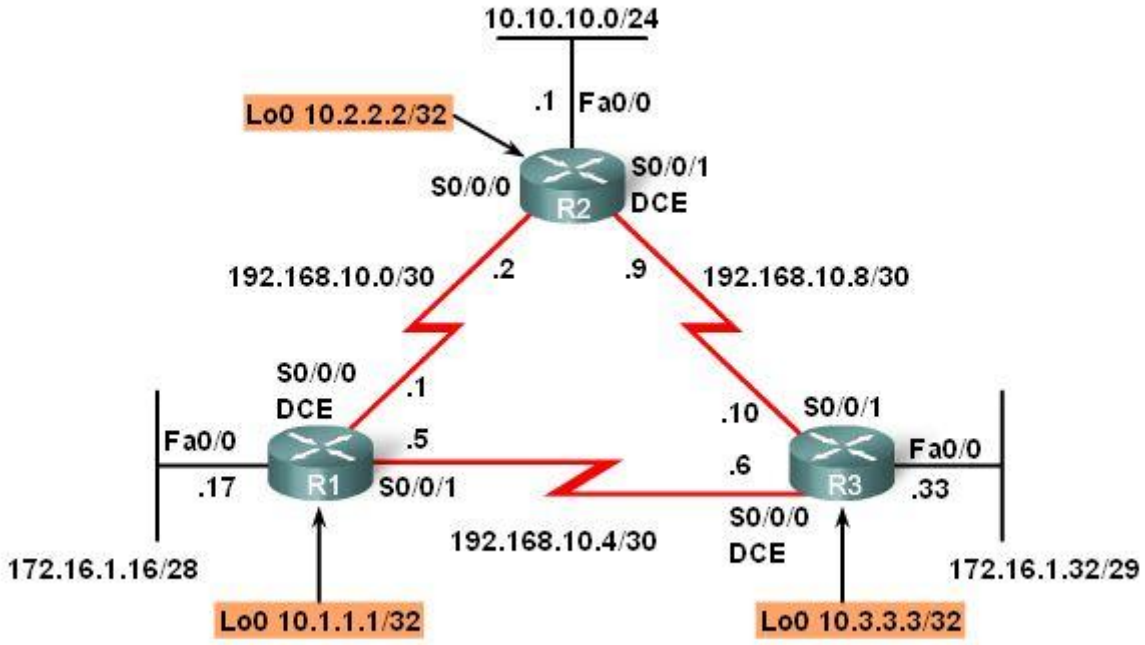


Рис. 8.9 Топологія з loopback інтерфейсами

OSPF router-id команда

OSPF команда ***router-id*** була введена в IOS 12.0(T) і має пріоритет над IP-адресами loopback і фізичних інтерфейсів при визначення ***router ID***. Синтаксис команди:

*Router(config)#**router ospf** process-id*
*Router(config-router)**#router-id** ip-address*

Зміна Router ID

router ID вибирається, коли OSPF сконфігурований першою OSPF командою ***network***. Якщо OSPF команда ***router-id*** або loopback адреса сформовані після OSPF команди ***network***, ***router ID*** буде отриманий від інтерфейсу із самою старшою активною IP-адресою.

router ID може бути змінений шляхом перезавантаження маршрутизатора або наступною командою:

*Router#clear **ip ospf process***

Дубльовані Router IDs

Коли два маршрутизатори мають однакові ***router ID*** в OSPF домені, маршрутизація може на функціонувати належним чином. Якщо ***router ID*** однакові на двох сусідніх маршрутизаторах, суміжність не буде встановлена. Якщо відбувається дублювання OSPF ***router IDs***, IOS буде відображати подібне повідомлення:

% OSPF-4-DUP_RTRID1: Detected router with duplicate router ID

Щоб виправити цю проблему, конфігуруйте всі маршрутизатори таким

чином, щоб кожний мав унікальний OSPF ***router ID***.

8.2.5 Перевірка OSPF

Команда ***show ip ospf neighbor*** може бути використана, щоб перевірити й знайти несправності в установці відносин між OSPF сусідами. Для кожного сусіда ця команда відображає наступну інформацію (див. рис. 8.10):

- ***Neighbor ID*** - router ID сусіднього маршрутизатора.
- ***Pri*** - OSPF пріоритет інтерфейсу.
- ***State*** - OSPF стан інтерфейсу. FULL означає, що маршрутизатор і його сусід мають ідентичні OSPF бази даних стану каналу.
- ***Dead Time*** – кількість часу, що залишилася, впродовж якого маршрутизатор буде очікувати одержання OSPF пакета від сусіда, перед тим, як оголосити сусіда таким, що перейшов у стан «down». Це значення відновлюється при одержанні інтерфейсом Hello пакета.
- ***Address*** - IP адреса інтерфейсу сусіда, з яким цей маршрутизатор безпосередньо з'єднаний.
- ***Interface*** - інтерфейс, на якому даний маршрутизатор сформував суміжність із сусідом.

```
R1#show ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
10.3.3.3          1   FULL/  -        00:00:30    192.168.10.6    Serial0/0/1
10.2.2.2          1   FULL/  -        00:00:33    192.168.10.2    Serial0/0/0
```

Рис. 8.10 Перевірка сумідностей із сусідами

При пошуку несправностей у мережі OSPF, команда ***show ip ospf neighbor*** може бути використана, щоб перевірити, що маршрутизатор сформував суміжності із сусідніми маршрутизаторами. Якщо ***router ID*** сусіднього маршрутизатора не відображається, або якщо стан не показаний як FULL, два маршрутизатори не сформували суміжність OSPF. Якщо два маршрутизатори не встановлюють суміжність, обмін інформацією про стан каналу проводитися не буде. Неповні бази даних стану каналу можуть привести до неправильних SPF дерев і таблиць маршрутизації. Маршрути до мереж призначення можуть не існувати, або бути не самими оптимальними.

Примітка: На мережах множинного доступу, як наприклад Ethernet, два суміжні маршрутизатори можуть мати стан, відображений як 2WAY.

Два маршрутизатори можуть не сформувати суміжність OSPF, якщо:

- ***Маски підмережі не співпадають, тоді маршрутизатори перебувають у різних мережах.***
- ***OSPF Hello або Dead таймери не збігаються.***
- ***Типи мережі не співпадають.***

- ***Маршрутизатори належать до різних областей OSPF.***
- ***Була використана помилкова або некоректна OSPF команда network.***

Інші потужні команди OSPF для пошуку несправностей:

show ip protocols
show ip ospf
show ip ospf interface

Як показано на рис. 8.11, команда ***show ip protocols*** - швидкий спосіб перевірити важливу інформацію про конфігурацію OSPF, включаючи OSPF ***process ID***, ***router ID***, мережі, які анонсуються маршрутизатором, сусіди, від яких маршрутизатор одержує оновлення й задана за замовчуванням адміністративна відстань (110 для OSPF).

```
R1#show ip protocols
Routing Protocol is "ospf 1"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Router ID 10.1.1.1
  Number of areas in this router is 1. 1 normal 0 stub 0 nssa
  Maximum path: 4
  Routing for Networks:
    172.16.1.16 0.0.0.15 area 0
    192.168.10.0 0.0.0.3 area 0
    192.168.10.4 0.0.0.3 area 0
 Reference bandwidth unit is 100 mbps
  Routing Information Sources:
    Gateway         Distance      Last Update
    10.2.2.2              110      11:29:29
    10.3.3.3              110      11:29:29
  Distance: (default is 110)
```

Рис. 8.11 Команда ***show ip protocols***

Команда ***show ip ospf*** також може використовуватися для визначення OSPF ***process ID*** і ***router ID.*** Крім того, вона відображає інформацію OSPF області, наприклад, коли останній раз виконувався алгоритм SPF. Як можна бачити із прикладу на рис. 8.12, OSPF - дуже стійкий протокол маршрутизації. Єдина OSPF- подія, у якій R1 брав участь за останні 9,5 годин – посилка невеликих Hello пакетів до його сусідів.

Команда показує також важливу інформацію SPF алгоритму, куди входить список SPF затримок:

Initial SPF schedule delay 5000 msecs
Minimum hold time between two consecutive Spfs 10000 msecs
Maximum wait time between two consecutive Spfs 10000 msecs

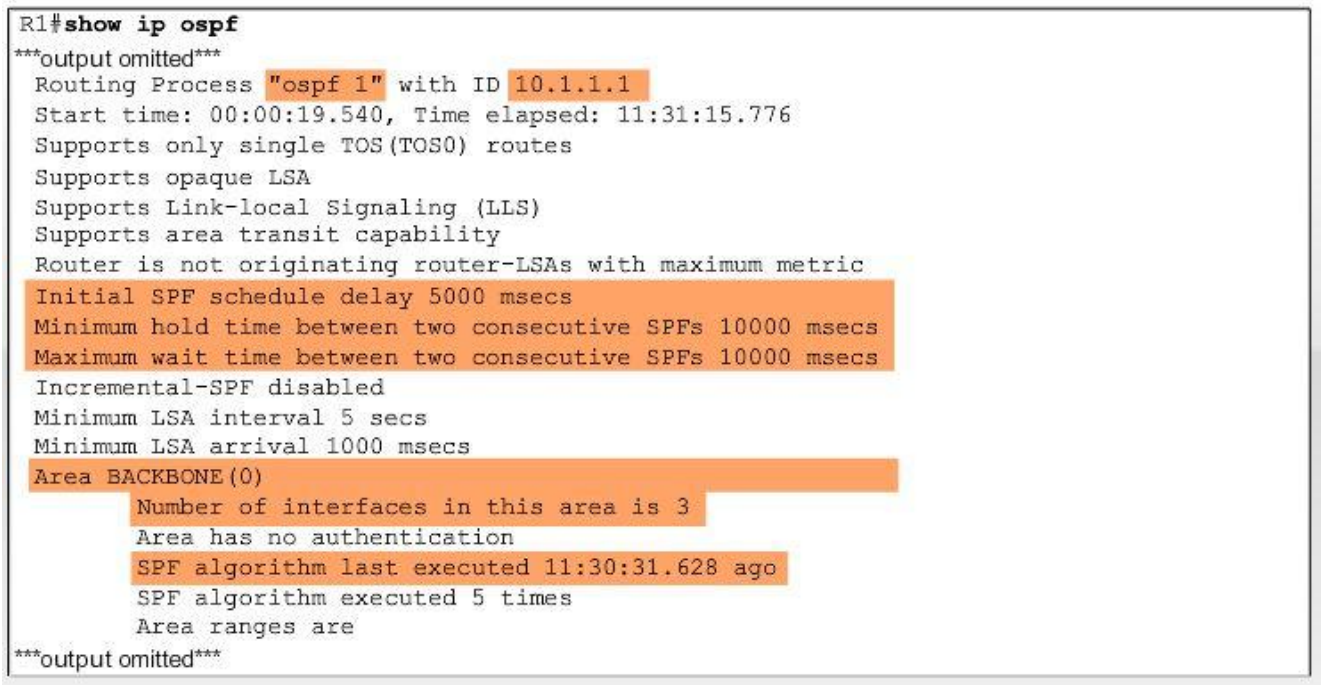


Рис. 8.12 Команда ***show ip ospf***

При одержанні маршрутизатором нової інформації про топологію (додавання, видалення або модифікація каналу), маршрутизатор повинен виконати повторно SPF алгоритм, створити нове SPF дерево, і модифікувати таблицю маршрутизації. Алгоритм SPF інтенсивно використовує процесорний час і час обчислень залежить від розміру області. Розмір області визначається числом маршрутизаторів і розміром бази даних стану каналу.

Мережа, яка циклічно переходить зі стану «up» в «down» і назад, називається flapping link. flapping link може змусити OSPF маршрутизатори в області постійно перераховувати алгоритм SPF, перешкоджаючи правильній конвергенції. Щоб мінімізувати цю проблему, маршрутизатор очікує 5 секунд (5000 мс) після одержання LSU перед виконанням алгоритму SPF . Це відомо як ***SPF schedule delay***. Для того, щоб запобігти постійному виконанню алгоритму SPF маршрутизатором, є додатково ***Hold Time*** 10 секунд (10000 мс). Маршрутизатор очікує 10 секунд після виконання алгоритму SPF перед повторним виконанням цього алгоритму.

Найшвидший спосіб перевірити Hello і Dead інтервали - використовувати команду ***show ip ospf interface***. Як показано на рис. 8.13, додаючи ім'я інтерфейсу й номер до команди, можна добитися відображення інформації тільки про один інтерфейс. Ці значення інтервалів включаються в OSPF Hello пакети, які пересилаються між сусідами. OSPF може мати різні Hello і Dead інтервали на різних інтерфейсах, але для того, щоб два маршрутизатори OSPF стали сусідами, їх OSPF Hello і Dead інтервали повинні бути ідентичні.

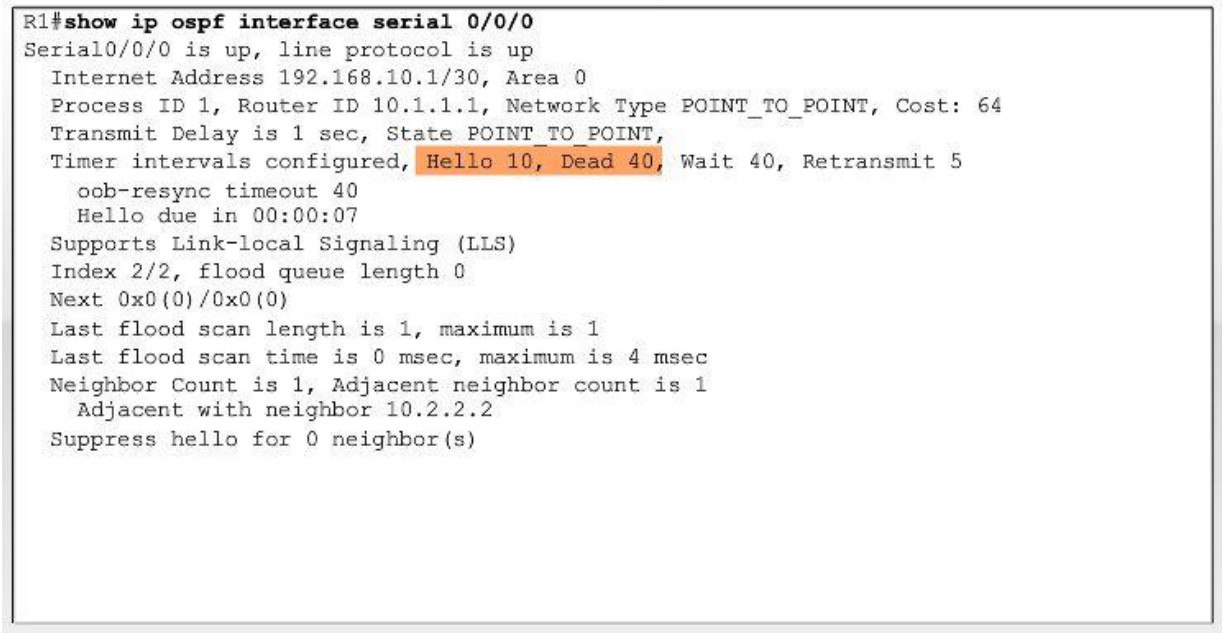


Рис. 8.13 Команда ***show ip ospf interface***

8.2.6 Дослідження таблиці маршрутизації

Як відомо, найшвидший спосіб перевірити конвергенцію OSPF - подивитися таблицю маршрутизації для кожного маршрутизатора в топології.

Команда ***show ip route*** може використовуватися, щоб перевірити, що OSPF посилає й одержує маршрути через OSPF. Літера **O** на початку кожного маршруту вказує, що джерело маршруту - OSPF. Слід звернути увагу на дві чіткі відмінності в OSPF таблиці маршрутизації, у порівнянні з таблицями маршрутизації, які розглядалися в попередніх главах (див. рис. 8.14). Для початку, помітимо, що кожний маршрутизатор має чотири безпосередньо підключені мережі, тому що loopback інтерфейс вважається четвертою мережею. Ці loopback інтерфейси не анонсуються в OSPF. Тому, кожний маршрутизатор складає список семи відомих мереж. По-друге, на відміну від RIPv2 і EIGRP, OSPF не виконує автоматичну сумаризацію по класових мережних границях. OSPF протокол дійсно безкласовий.

8.3 Метрика OSPF

8.3.1 Метрика OSPF

Метрика OSPF називається вартістю (cost). В RFC 2328 написано: "Вартість асоціюється з кожним вихідним інтерфейсом маршрутизатора. Ця вартість конфігурується адміністратором системи. Чим нижче вартість, тем імовірніше, що інтерфейс буде використаний для передачі трафіка."

Зверніть увагу, що RFC 2328 не конкретизує, які значення потрібно використовувати, щоб визначити вартість.

```
R1#show ip route

Codes: D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
***output omitted***

Gateway of last resort is not set

     192.168.10.0/30 is subnetted, 3 subnets
C       192.168.10.0 is directly connected, Serial0/0/0
C       192.168.10.4 is directly connected, Serial0/0/1
O       192.168.10.8 [110/128] via 192.168.10.6, 14:27:57, Serial0/0/1
                     [110/128] via 192.168.10.2, 14:27:57, Serial0/0/0
     172.16.0.0/16 is variably subnetted, 2 subnets, 2 masks
O       172.16.1.32/29 [110/65] via 192.168.10.6, 14:27:57, Serial0/0/1
C       172.16.1.16/28 is directly connected, FastEthernet0/0
     10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
O       10.10.10.0/24 [110/65] via 192.168.10.2, 14:27:57, Serial0/0/0
C       10.1.1.1/32 is directly connected, Loopback0
```

Рис.8.14 Таблиця маршрутизації R1

Cisco IOS використовує в якості вартості накопичені пропускні здатності вихідних інтерфейсів від маршрутизатора до мережі призначення. На кожному маршрутизаторі вартість інтерфейсу розраховується як 10^8, поділене на пропускну здатність у біт/сек. 10^8 відома як еталонна пропускна здатність (reference bandwidth). Ділення 10^8 на пропускну здатність інтерфейсу призводить до того, що інтерфейси з вищими значеннями пропускної здатності будуть мати більш низьку розрахункову вартість. Помніть, що маршрут з найнижчою вартістю - маршрут, якому надається перевага (наприклад, для RIP, 3 переходи краще, ніж 10). Рис. 8.15 показує задані за замовчуванням метрики OSPF для декількох видів інтерфейсів.

Interface Type	10^8/ bps = Cost
Fast Ethernet and faster	10^8/100,000,000 bps = 1
Ethernet	10^8/10,000,000 bps = 10
E1	10^8/2,048,000 bps = 48
T1	10^8/1,544,000 bps = 64
128 kbps	10^8/128,000 bps = 781
64 kbps	10^8/64,000 bps = 1562
56 kbps	10^8/56,000 bps = 1785

Рис. 8.15 Величини вартості Cisco OSPF

Еталонна пропускна здатність (Reference Bandwidth)

Значення еталонної пропускної здатності за замовчуванням 10^8, 100,000,000 біт/с або 100 Мбіт/с. В результаті інтерфейси із пропускною здатністю 100 Мбіт/с і вище будуть мати однакову OSPF вартість 1. Еталонна пропускна здатність може бути змінена для адаптації до мереж з каналами, більш

швидкими ніж 100,000,000 біт/с (100 Мбіт/с), використовуючи команду OSPF ***auto-cost reference-bandwidth***. Коли ця команда необхідна, рекомендується використовувати її на всіх маршрутизаторах, щоб маршрутні метрики OSPF залишалися несуперечливими.

Накопичені вартості OSPF

Вартість OSPF маршруту – накопичена величина від одного маршрутизатора до мережі призначення. Наприклад, на рис. 8.16, таблиця маршрутизації на R1 показує вартість 65 для досягнення мережі 10.10.10.0/24 на R2. Оскільки 10.10.10.0/24 прикріплена до інтерфейсу Fastethernet, R2 призначає вартість рівну 1 для 10.10.10.0/24. Потім R1 додає додаткове значення вартості 64, щоб послати дані через канал T1 (заданий за замовчуванням) між R1 і R2.

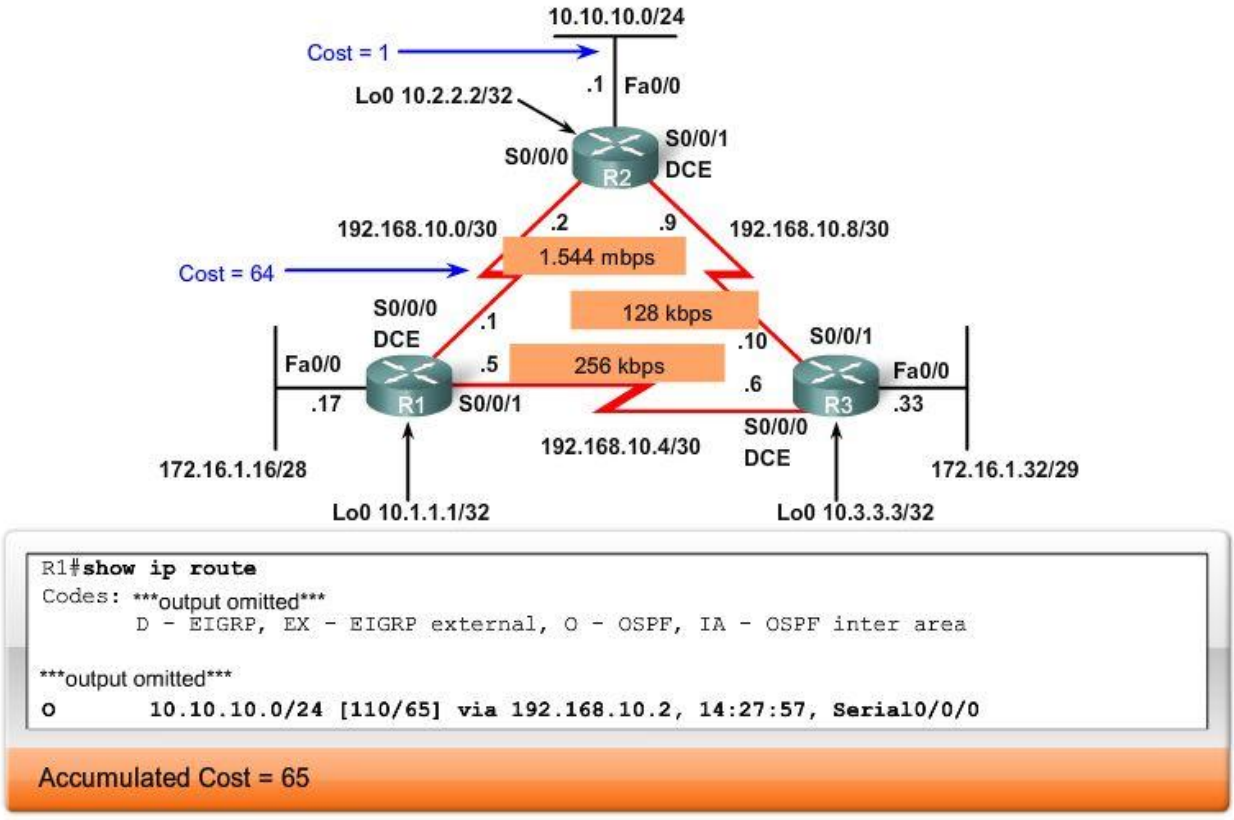


Рис. 8.16 Накопичена вартість OSPF

Задана за замовчуванням пропускна здатність на серійних інтерфейсах

Можна використовувати команду ***show interface***, щоб переглянути значення пропускної здатності, зіставлене з інтерфейсом. На маршрутизаторах Cisco значення пропускної здатності на багатьох серійних інтерфейсах за замовчуванням встановлене в T1 (1.544 Мбіт/с). Однак, деякі серійні інтерфейси, можуть приймати значення за замовчуванням 128 Кбіт/с. Тому, завжди слід перевіряти значення за замовчуванням командою ***show interface***.

Пам’ятайте, це значення пропускної здатності фактично не впливає на швидкість каналу; це значення використовується деякими протоколами маршрутизації, щоб розрахувати метрику. Найчастіше, на серійних інтерфейсах реа-

льна швидкість каналу відрізняється від заданої за замовчуванням пропускної здатності. Важливо, щоб значення пропускної здатності відображало реальну швидкість каналу, тоді таблиця маршрутизації буде мати точну інформацію про найкращі шляхи. Наприклад, ви оплачуєте частину підключення T1 від провайдера, одну чверть від повного підключення T1 (384 Кбіт/с). Однак, для цілей протоколу маршрутизації, IOS призначить значення пропускної здатності T1, навіть якщо інтерфейс фактично тільки пересилає й одержує одну чверть повного підключення T1 (384 Кбіт/с).

Розрахункова OSPF вартість інтерфейсу може бути перевірена командою ***show ip ospf interface*** (див. рис. 8.17).

```
R1#show ip ospf interface serial 0/0/0
Serial0/0/0 is up, line protocol is up
  Internet Address 192.168.10.1/30, Area 0
  Process ID 1, Router ID 10.1.1.1, Network Type POINT_TO_POINT, Cost: 64
  <output omitted>
```

Рис. 8.17 Перевіряємо OSPF вартість інтерфейсу

8.3.2 Зміна вартості каналу

Коли серійний інтерфейс фактично не працює на заданій за замовчуванням швидкості T1, інтерфейс вимагає модифікації вручну. Обидві сторони каналу повинні бути сконфігуровані з однаковими значеннями. Обидві команди ***bandwidth interface*** і ***ip ospf cost interface*** досягають поставленої мети - при визначенні кращого маршруту, OSPF буде використовувати точне значення.

Команда bandwidth

Команда ***bandwidth*** використовується, щоб змінити значення пропускної здатності, яке використовується IOS під час підрахунку метрики - вартості OSPF.

Синтаксис команди :

*Router(config-if)#**bandwidth** bandwidth-kbps*

На рис. 8.18 показане виконання ***bandwidth*** команди, і як результат зміна вартості серійного інтерфейсу.

```
R1(config)#inter serial 0/0/0
R1(config-if)#bandwidth 64
R1(config-if)#inter serial 0/0/1
R1(config-if)#bandwidth 256
R1(config-if)#end
R1#show ip ospf interface serial 0/0/0
Serial0/0 is up, line protocol is up
  Internet Address 192.168.10.1/30, Area 0
  Process ID 1, Router ID 10.1.1.1, Network Type POINT_TO_POINT, Cost: 1562
  Transmit Delay is 1 sec, State POINT_TO_POINT,
***output omitted***
```

$10^8/64{,}000$ bps = 1562

Рис. 8.18 Команда bandwidth

Команда ip ospf cost

Альтернативний метод - використовувати команду ***ip ospf cost***, яка дозволяє нам безпосередньо вказати вартість інтерфейсу (мал. 8.19). Наприклад, на R1 ми могли б сконфігурувати Serial 0/0/0 наступною командою:

*R1(config)#**interface serial 0/0/0***
*R1(config-if)#**ip ospf cost 1562***

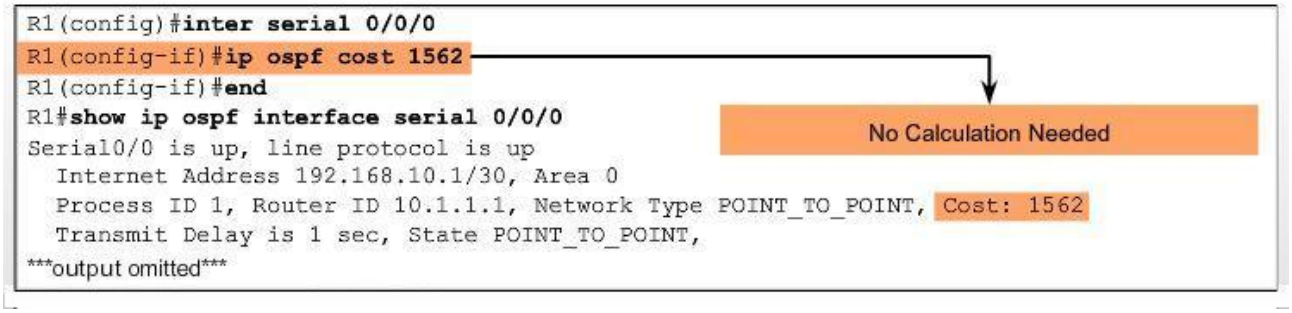


Рис. 8.19 Команда ip ospf cost

Порівняння команди bandwidth і команди ip ospf cost

Команда ***ip ospf cost*** корисна в змішаних середовищах, де маршрутизатори від різних виробників по різному обчислюють вартість OSPF. Основна різниця між двома командами та, що команда ***bandwidth*** використовує результати обчислення, щоб визначити вартість каналу. Команда ***ip ospf cost*** обходить це обчислення, безпосередньо встановлюючи вартість каналу в певне значення.

8.4 OSPF і мережі множинного доступу

8.4.1 Проблеми в мережах множинного доступу

Мережа множинного доступу - це мережа більш ніж із двома пристроями в одному загальному середовищі. У верхній частині рис. 8.20, Ethernet LAN, яка приєднана до R1 розширено, щоб показати можливі пристрої, які, можуть бути прикріплені до мережі 172.16.1.16/28. Ethernet LAN - приклад широкомовної мережі множинного доступу. Ці мережі називаються широкомовними, тому що всі пристрої мережі бачать усі широкомовні фрейми. Це мережі множинного доступу, тому що є безліч хостів, принтерів, маршрутизаторів, і інших пристроїв в одній і тій же мережі.

На контрасті, у двоточковій мережі є тільки два пристрої, по одному з кожної сторони. WAN канал між R1 і R3 - приклад двоточкового каналу, показаний на нижній частині рис. 8.20. багатоточкова

OSPF визначає п'ять типів мереж:

- Point-to-point – двоточкові.
- Broadcast Multiaccess - широкомовні множинного доступу.
- Nonbroadcast Multiaccess (NBMA) нешироковомовні множинного доступу.
- Point-to-multipoint – багатоточкові.
- Virtual links – віртуальні канали.

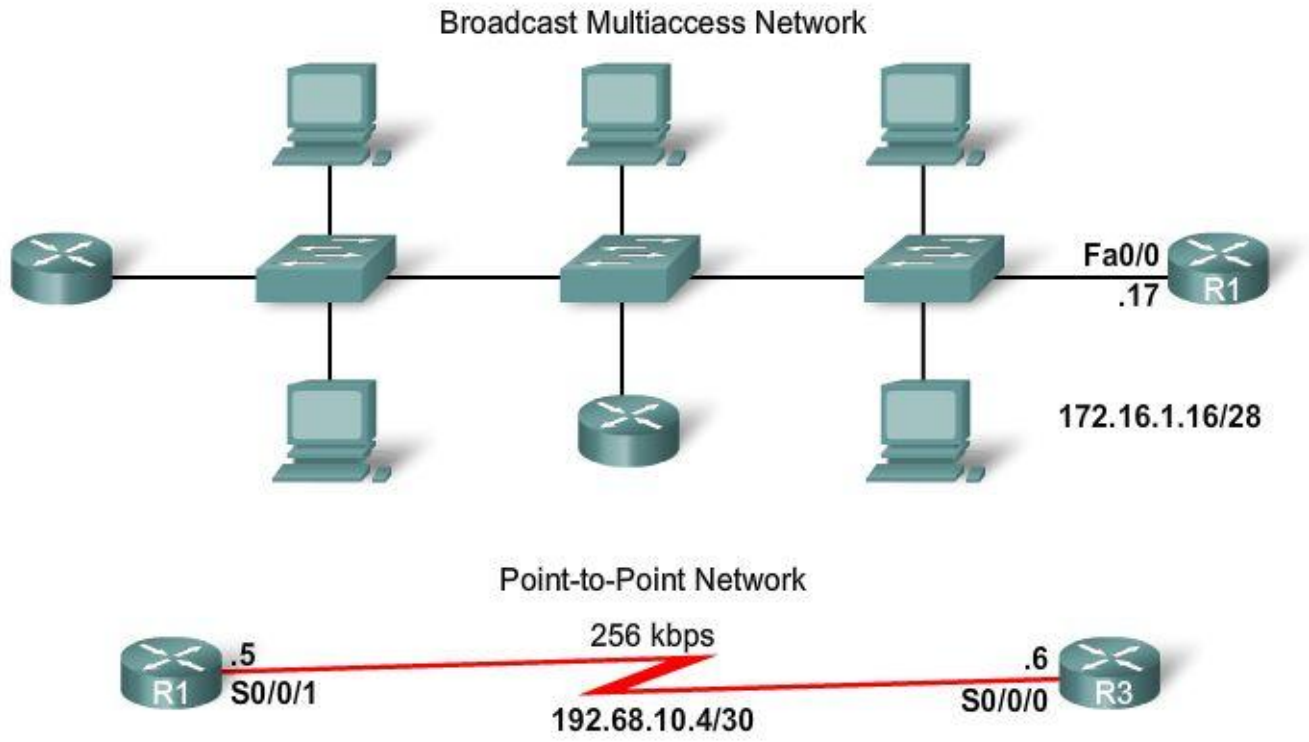


Рис. 8.20 Мережі множинного доступу та двоточкові мережі

NBMA і point-to-multi-point мережі містять у собі Frame Relay, ATM, і X.25 мережі. Віртуальні канали - спеціальний вид каналу, який може використовуватися в множинних областях OSPF.

Мережі множинного доступу можуть створити дві проблеми для OSPF, які мають відношення до лавинного розсилання LSAs:

1. Створення множинних суміжностей, одна суміжність для кожної пари маршрутизаторів.
2. Велике лавинне розсилання LSAs (Link-State Advertisements).

Множинні суміжності

Створення суміжності між кожною парою маршрутизаторів у мережі створило б непотрібне число суміжностей. Це приводило б до надмірного числа LSAs, що передаються між маршрутизаторами в мережі.

Щоб зрозуміти проблему із множинними суміжностями, нам потрібно вивчити формулу. ***Для будь-якого числа маршрутизаторів (позначене як n) у мережі множинного доступу, буде n (n - 1) / 2 суміжностей***. При збільшенні числа маршрутизаторів число суміжностей різко збільшується, як показано на рис. 8.21.

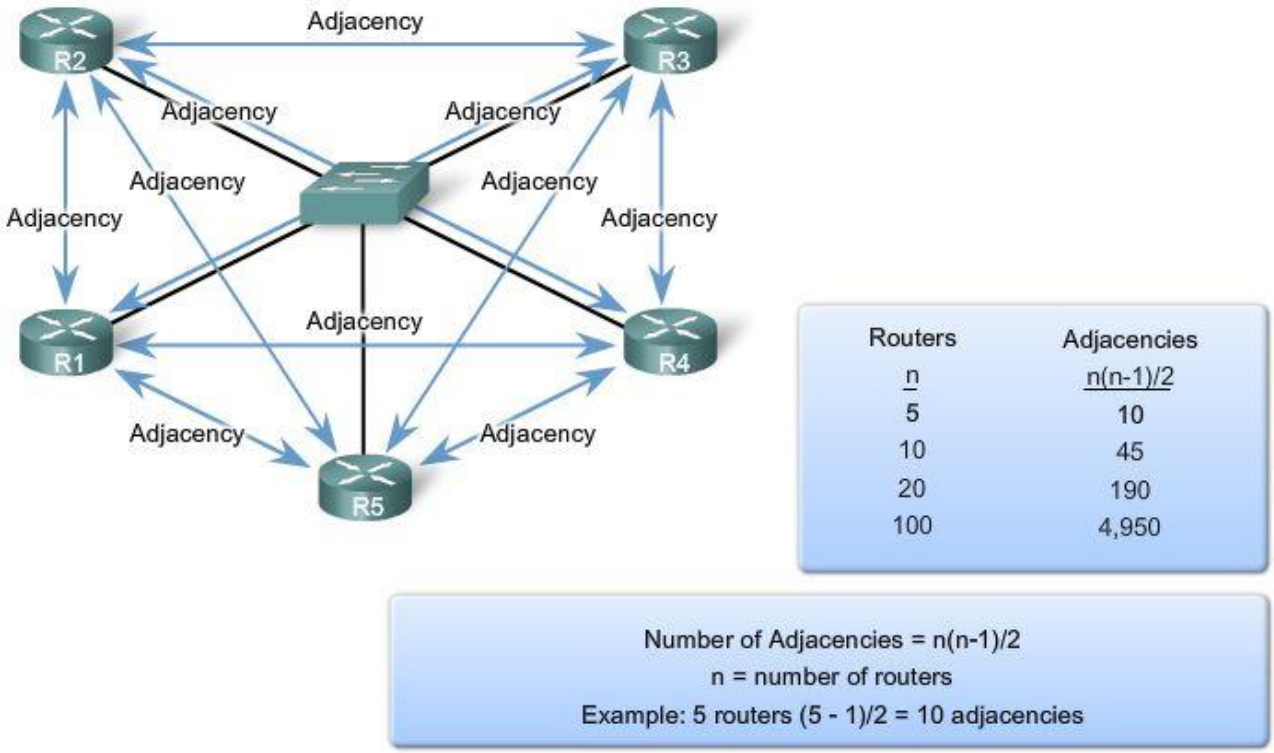


Рис. 8.21 Число суміжностей росте експоненційно

Лавинне розсилання LSA

Маршрутизатори стану каналу лавиною розсилають свої пакети стану каналу, коли OSPF ініціалізується або, коли є зміни в топології.

У мережі множинного доступу ця лавина може стати надмірною. На рис. 8.22, R2 посилає LSA. Ця подія змушує кожен інший маршрутизатор також послати LSA. Для кожного отриманого LSA потрібне підтвердження. Якби кожний маршрутизатор у мережі множинного доступу виконував лавинне розсилання й відсилав підтвердження на всі прийняті LSA від усіх інших маршрутизаторів у мережі, мережний трафік став би хаотичним.

Розв'язок: Відзначений маршрутизатор (Designated Router)

Розв'язок по керуванню числом суміжностей і лавинним розсиланням LSAs у мережах множинного доступу – відзначений маршрутизатор (DR).

У мережах множинного доступу, OSPF вибирає відзначений маршрутизатор (DR), який буде місцем збору й поширення LSAs, посланих і отриманих. Резервний відзначений маршрутизатор (BDR) обирається на випадок збою відзначеного маршрутизатора. Усі інші маршрутизатори стають DROthers (це означає, що маршрутизатор не є ні DR ні BDR).

Маршрутизатори в мережі множинного доступу вибирають DR і BDR. DROthers тільки формують повні суміжності з DR і BDR у мережі. Це означає, що замість лавинного розсилання LSAs усім маршрутизаторам у мережі, DROthers тільки відправляють свої LSAs до DR і BDR, використовуючи групо-

ву адресу 224.0.0.6 (AllDRrouters - Усі DR маршрутизатори). На рис. 8.23 R1 відправляє LSAs до DR. BDR також прослуховує їх. DR відповідає за відправлення LSAs від R1 до всіх інших маршрутизаторів (рис. 8.24). DR використовує групову адресу 224.0.0.5 (AllOSPFrouters - Усі маршрутизатори OSPF). У результаті тільки один маршрутизатор займається лавинним розсиланням усіх LSAs у мережах множинного доступу.

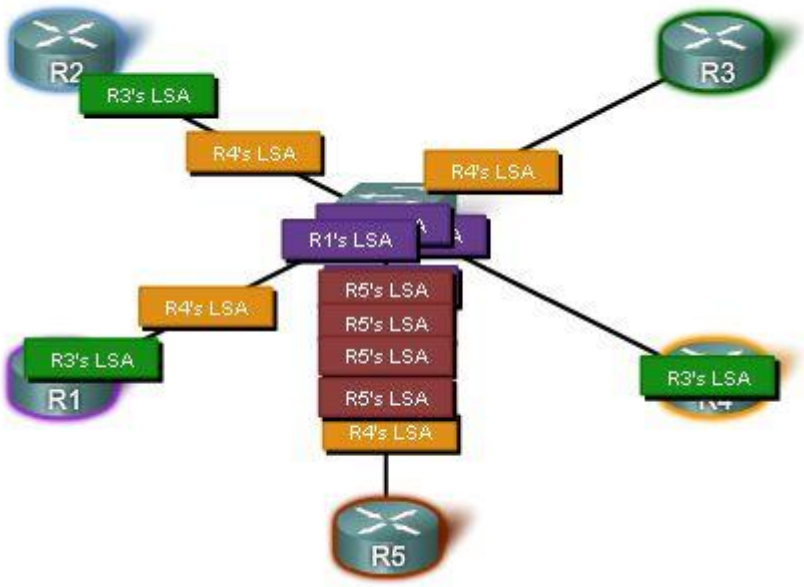


Рис. 8.22 Лавинне розсилання LSA

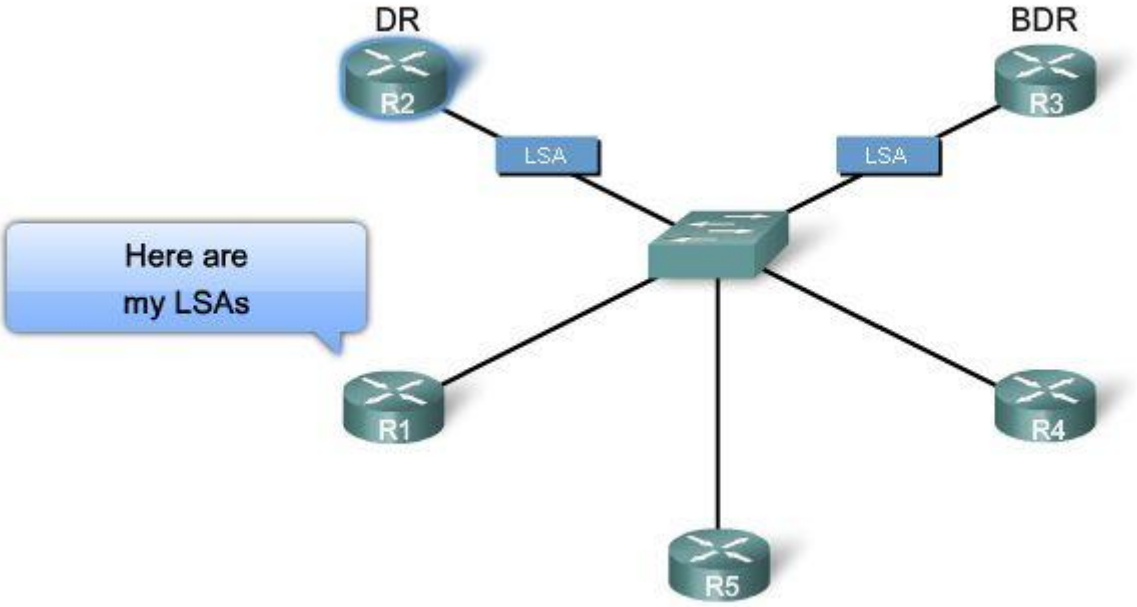


Рис. 8.23 Суміжності формуються тільки з DR і BDR

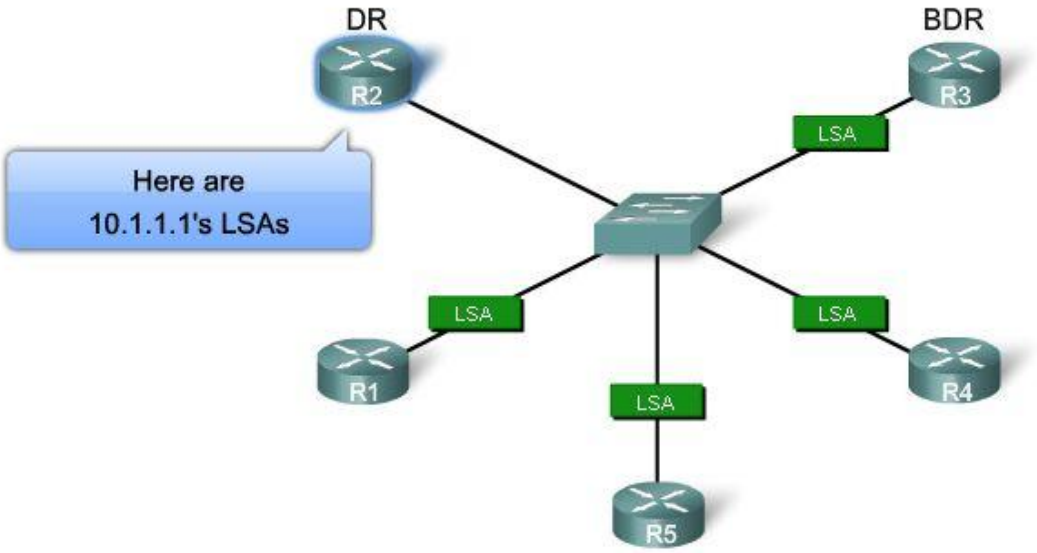


Рис. 8.24 DR розсилає LSAs усім іншим маршрутизаторам

8.4.2 Процес вибору DR і BDR

Зміна топології

Вибори DR/BDR не проводяться у двоточкових мережах. Тому, у стандартній топології трьох маршрутизаторів (рис 8.5) не потрібно вибирати DR і BDR, оскільки канали між маршрутизаторами не є мережами множинного доступу.

Тому для обговорення вибору DR і BDR, ми будемо використовувати топологію множинного доступу, показану на рис. 8.25. У цій новій топології, ми маємо три маршрутизатори, що поділяють загальну мережу множинного доступу Ethernet, 192.168.1.0/24. Кожний маршрутизатор сформований з IP адресою на інтерфейсі та loopback адресою для router ID.

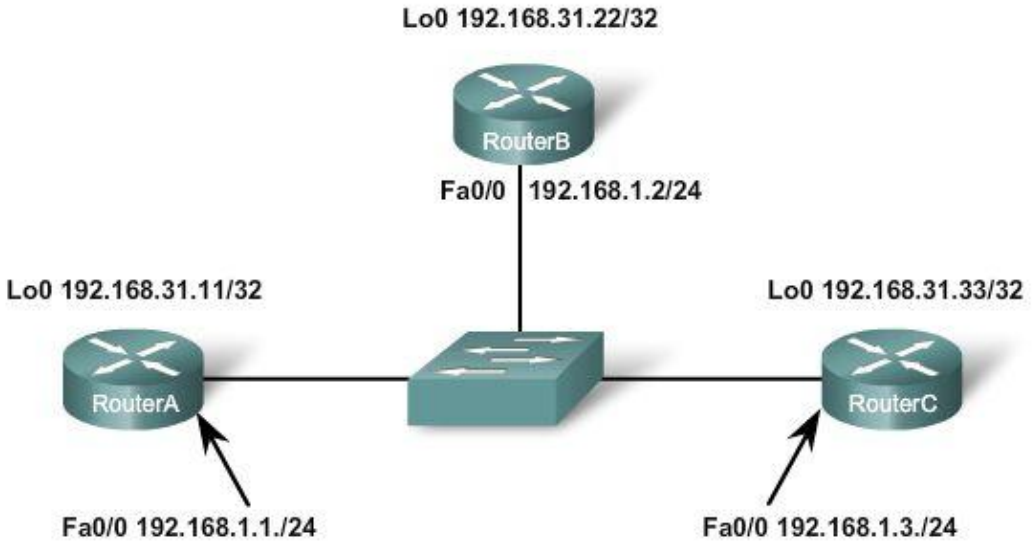


Рис. 8.25 Топологія із множинним доступом

Вибори DR/BDR

Як вибираються DR і BDR? При цьому застосовуються наступні критерії:

1. DR: Маршрутизатор з найвищим пріоритетом OSPF інтерфейсу .
2. BDR: Маршрутизатор із другим за значенням найвищим пріоритетом OSPF інтерфейсу.
3. Якщо пріоритети OSPF інтерфейсів рівні, використовується найбільше значення ***router ID***.

У цьому прикладі, заданий за замовчуванням пріоритет OSPF інтерфейсу дорівнює 1. У результаті, ґрунтуючись на критеріях, перерахованих вище, щоб вибрати DR і BDR використовується OSPF ***router ID***. Як видно на рис. 8.26, RouterC стає DR і RouterB, із другим найвищим router ID, стає BDR. Оскільки RouterA не обраний ні як DR ні як BDR, він стає DROther.

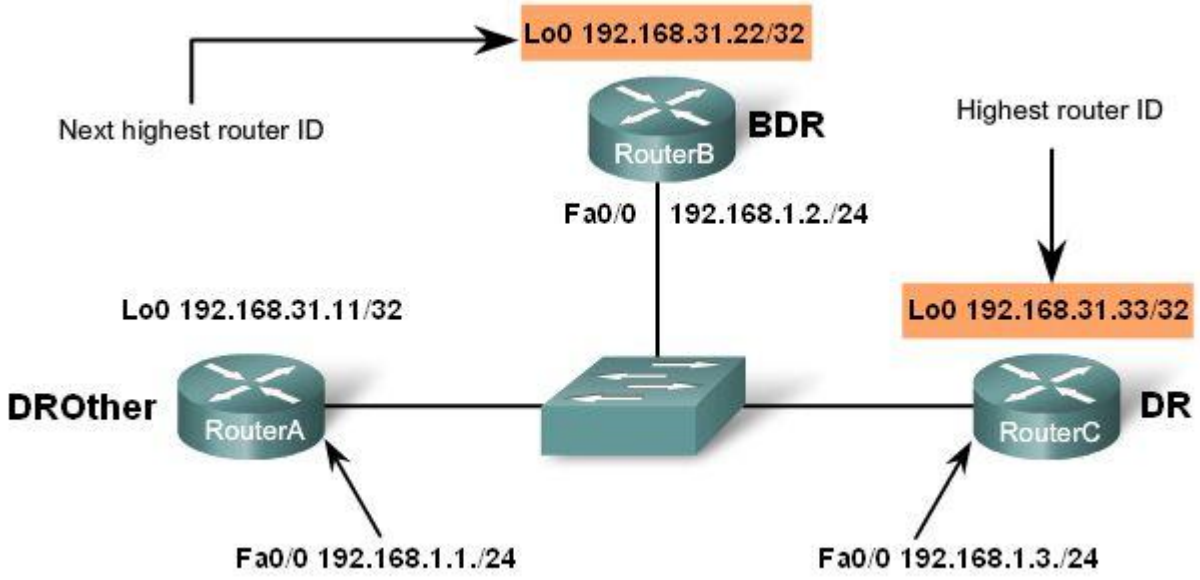


Рис. 8.26 Вибори DR/BDR

DROthers не тільки формує FULL суміжності з DR і BDR, але усе ще формує сусідню суміжність із будь-яким DROthers у мережі. Це означає, що всі маршрутизатори DROther у мережі множинного доступу усе ще одержують Hello пакети від усіх інших маршрутизаторів DRQther. Таким чином, вони інформовані про всі маршрутизатори в мережі. Коли два маршрутизатори DROther формують суміжність, стан відображається як 2WAY.

На рис. 8.27 команда ***show ip ospf neighbor*** відображає сусідню суміжність кожного маршрутизатора в мережі множинного доступу. Зверніть увагу на інформацію RouterA, де зазначено, що DR - RouterC з ***router ID*** 192.168.31.33 і що BDR - RouterB з ***router ID*** 192.168.31.22.

Оскільки що RouterA показує обох своїх сусідів як DR і BDR, то сам RouterA - DROther. Це можна перевірити, використовуючи на RouterA команду

show ip ospf interface fastethernet 0/0, як показано на рис. 8.28. Ця команда покаже DR, BDR, або DROTHER стан даного маршрутизатора, поряд з router ID DR і BDR у цій мережі множинного доступу.

```
RouterA#show ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
192.168.31.33     1   FULL/DR         00:00:39    192.168.1.3     FastEthernet0/0
192.168.31.22     1   FULL/BDR        00:00:36    192.168.1.2     FastEthernet0/0

RouterB#show ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
192.168.31.33     1   FULL/DR         00:00:34    192.168.1.3     FastEthernet0/0
192.168.31.11     1   FULL/DROTHER    00:00:38    192.168.1.1     FastEthernet0/0

RouterC#show ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
192.168.31.22     1   FULL/BDR        00:00:35    192.168.1.2     FastEthernet0
192.168.31.11     1   FULL/DROTHER    00:00:32    192.168.1.1     FastEthernet0
```

Priority is equal at the default value of 1.

Рис. 8.27 Команда show ip ospf neighbor

```
RouterA#show ip ospf interface fastethernet 0/0
FastEthernet0/0 is up, line protocol is up
  Internet Address 192.168.1.1/24, Area 0
  Process ID 1, Router ID 192.168.31.11, Network Type BROADCAST, Cost: 1
  Transmit Delay is 1 sec, State DROTHER, Priority 1
  Designated Router (ID) 192.168.31.33, Interface address 192.168.1.3
  Backup Designated router (ID) 192.168.31.22, Interface address 192.168.1.2
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:06
  Supports Link-local Signaling (LLS)
  Index 1/1, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 0, maximum is 1
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 2, Adjacent neighbor count is 2
    Adjacent with neighbor 192.168.31.22  (Backup Designated Router)
    Adjacent with neighbor 192.168.31.33  (Designated Router)
  Suppress hello for 0 neighbor(s)
```

Рис. 8.28 Команда show ip ospf interface

Часові параметри вибору DR/BDR

Процес вибору DR і BDR починається, як тільки перший маршрутизатор з дозволеним OSPF інтерфейсом стає активним у мережі множинного доступу. Це може відбутися, коли маршрутизатори включили, або коли команда OSPF ***network*** для цього інтерфейсу сформована. Процес вибору займає кілька се-

кунд. Якщо не всі маршрутизатори в мережі множинного доступу закінчили завантаження, можливо, що маршрутизатор з найнижчим ***router ID*** стане DR. Це може бути просто маршрутизатор, у якого завантаження зайняло менше часу.

Коли DR обраний, він залишається DR, поки не відбудеться одна з наступних подій:

- DR вийде з ладу.
- Збій OSPF процесу на DR.
- Збій інтерфейсу множинного доступу на DR.

На малюнках 8.29 – 8.32 показані приклади таких збоїв.

Якщо DR виходить із ладу, BDR виконує роль DR і вибори проводяться для нового BDR. На рис. 8.29, RouterC зазнає невдачі й колишній BDR, RouterB, стає DR. Єдиний доступний маршрутизатор, щоб бути BDR - RouterA.

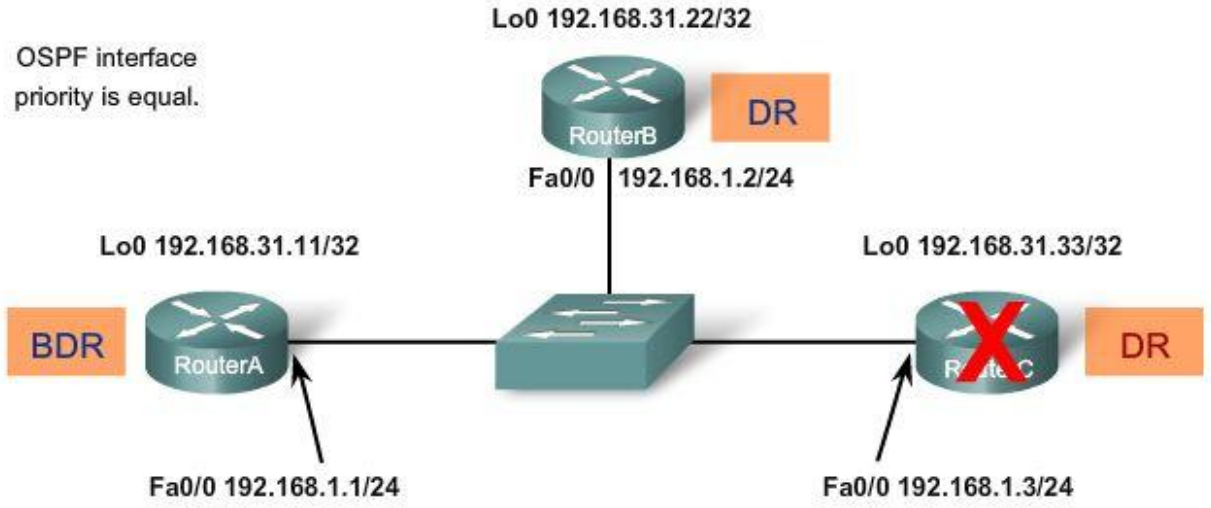


Рис. 8.29 З ладу вийшов DR

RouterD приєднався до мережі – рис. 8.30. ***Якщо новий маршрутизатор входить у мережу після того, як DR і BDR були обрані, він не стане DR або BDR, навіть якщо він має більш високий пріоритет OSPF інтерфейсу або router ID, ніж поточні DR і BDR***. Новий маршрутизатор може бути обраний BDR, якщо поточний DR або BDR зазнають збою. Якщо поточний DR зазнає невдачі, BDR стане DR, і новий маршрутизатор може бути обраний як новий BDR. Після того, як новий маршрутизатор стає BDR, якщо DR зазнає невдачі, то новий маршрутизатор стане DR. Тобто поточні DR і BDR повинні обоє потерпіти крах перед тим, як новий маршрутизатор зможе бути обраний DR або BDR.

Колишній DR повертається. Попередній DR не відновлює стан DR, якщо він повертається в мережу. На рис. 8.31, RouterC закінчив перезавантаження і стає DROther, навіть якщо його ***router ID***, 192.168.31.33, вище, ніж у поточних DR і BDR.

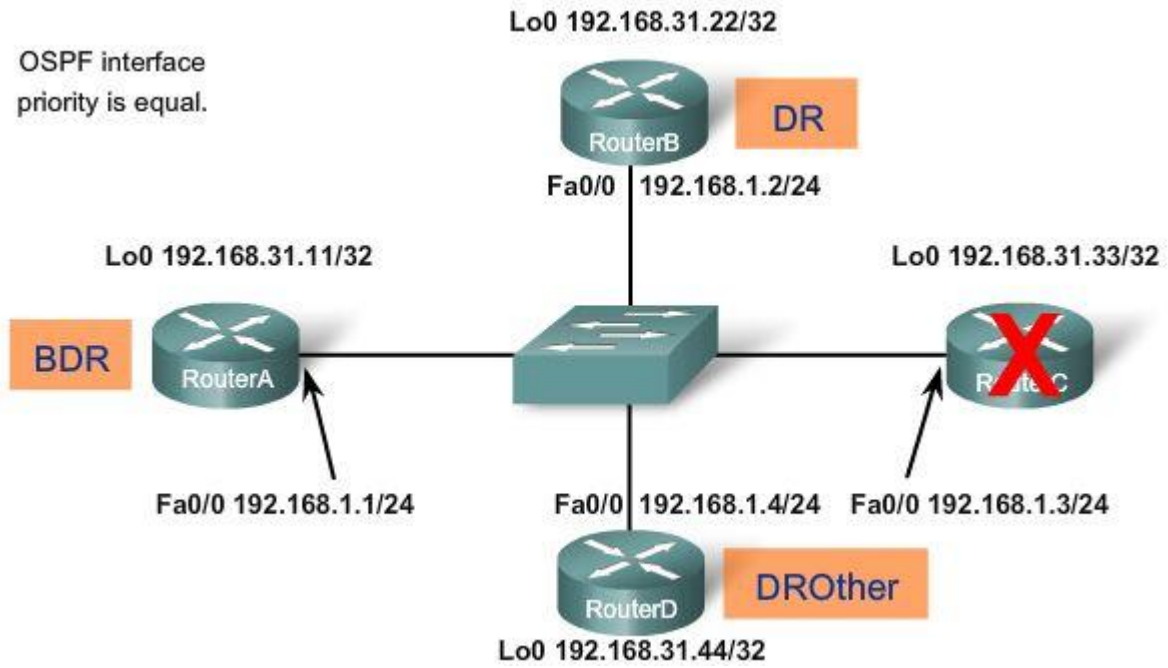


Рис. 8.30 RouterD приєднався до мережі

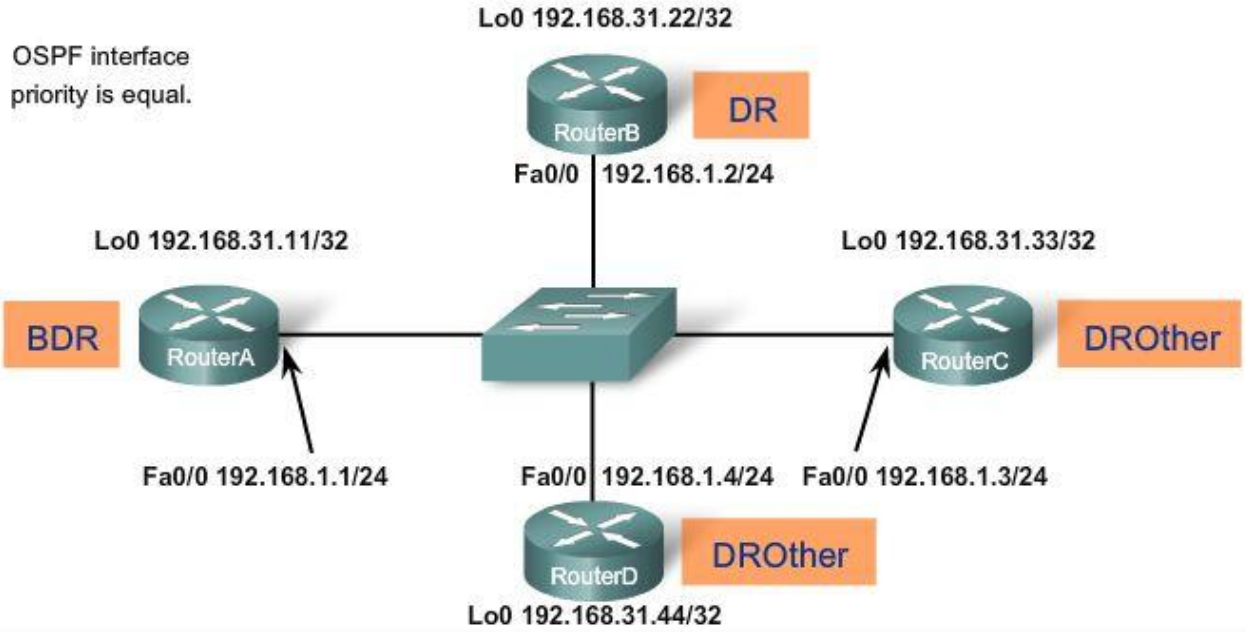


Рис. 8.31 DR, який повернувся, вже не зможе зайняти своє місце

Якщо BDR виходить із ладу, вибір проводиться серед DROthers, щоб визначити, який маршрутизатор буде новим BDR. На рис. 8.32, BDR маршрутизатор відмовив. Вибір робиться між RouterC і RouterD. RouterD виграє з більш високим ***router ID***.

Як бути впевненим у тому, що маршрутизатори, які ви прагли б бачити як DR або BDR виграють вибори? Завантажте DR першим, потім BDR, а потім завантажте всі інші маршрутизатори, або виконайте ***shutdown*** інтерфейсу на всіх маршрутизаторах, а потім ***no shutdown*** на DR, потім на BDR, а потім на всіх інших маршрутизаторах.

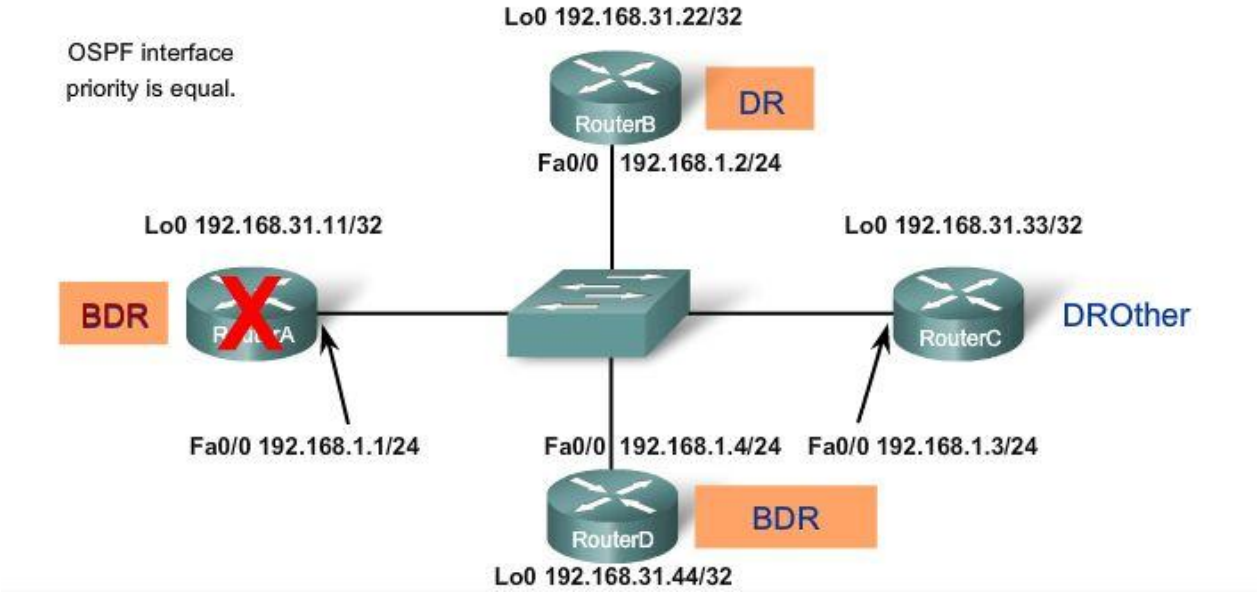


Рис. 8.32 З ладу вийшов BDR

Однак, як ви, можливо, уже здогадалися, ми можемо змінити пріоритет OSPF інтерфейсу, щоб краще контролювати вибори DR/BDR.

8.4.3 Пріоритет OSPF інтерфейсу

Оскільки DR стає центральним місцем збору й поширення LSAs, важливо, щоб цей маршрутизатор мав потужний процесор і досить пам'яті, щоб виконувати роботу DR. Замість того, щоб залежати від router ID при ухваленні рішення, які маршрутизатори вибираються DR і BDR, краще керувати вибором цих маршрутизаторів за допомогою команди ***ip ospf priority*** .

*Router(config-if)#**ip ospf priority {0 - 255}***

У попередньому прикладі пріоритет OSPF був однаковим у зв'язку з тим, що значення пріоритету за замовчуванням дорівнює 1 для всіх інтерфейсів маршрутизатора. Тому, router ID визначав DR і BDR. Але, якщо ви змінюєте значення пріоритету до більш високого ніж 1, маршрутизатор з найвищим пріоритетом стане DR і маршрутизатор з наступним найвищим пріоритетом стане BDR. Значення 0 робить маршрутизатор непридатним стати DR або BDR.

Оскільки пріоритети – значення, які встановлюються для інтерфейсів, вони забезпечують краще керування OSPF мережами множинного доступу. Вони також дозволяють маршрутизатору бути DR в одній мережі і DROther в іншій.

Пріоритет OSPF інтерфейсу можна переглянути, використовуючи команду ***show ip ospf*** . На рис. 8.34 видно, що пріоритет на RouterA має значення за замовчуванням 1.

Після зміни пріоритетів на інтерфейсах потрібно виконати команди shutdown і no shutdown. Відбудуться нові вибори DR і BDR з урахуванням установлених пріоритетів інтерфейсів.

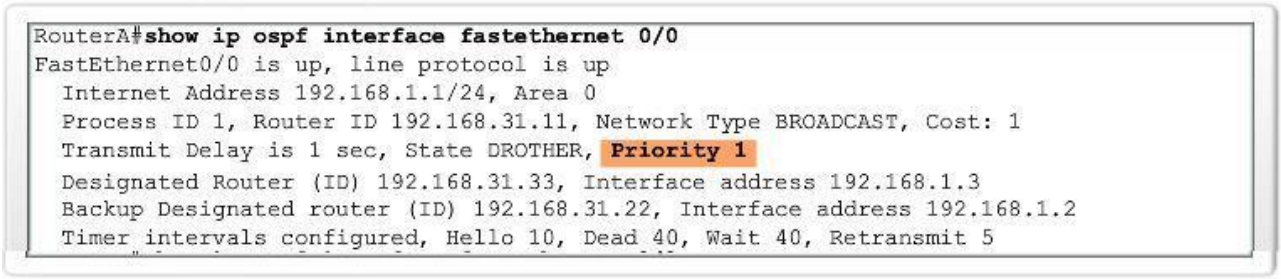


Рис.8.34 Перевірка значення пріоритету інтерфейсу

8.5 Висновки

8.5.1 Резюме

OSPF (Open Shortest Path First) – безкласовий протокол маршрутизації з урахуванням стану каналу. Поточна версія OSPF для IPv4 - OSPFv2, яка була введена в RFC 1247 і вдосконалена в RFC 2328. В 1999 році OSPFv3 для IPv6 був опублікований в RFC 2740.

OSPF за замовчуванням має адміністративну відстань 110, і відзначається в таблиці маршрутизації як джерело маршруту кодом O. OSPF дозволяється до виконання командою ***ospf process-id*** у режимі глобальної конфігурації. Значення ***process-id*** має локальну значимість, це означає, що не потрібно збігу ***process-id*** на маршрутизаторах для того, щоб вони могли встановити між собою суміжність.

Команди ***network,*** які використовуються з OSPF, виконують ті ж функції, що й при використанні з іншими IGP протоколами маршрутизації, але мають злегка одмінний синтаксис:

*Router(config-router)#**network network-address wildcard-mask area area-id***

wildcard-mask – це інвертована маска підмережі, а area-id повинно бути встановлене в 0.

OSPF не використовує протокол транспортного рівня, OSPF пакети посилають безпосередньо через IP. OSPF Hello пакет використовується OSPF для встановлення суміжностей із сусідами. За замовчуванням, OSPF Hello пакети посилаються кожні 10 секунд на сегментах множинного доступу то двоточкових сегментах (point-to-point) і кожні 30 секунд на не широкомовних сегментах множинного доступу (NBMA) (Frame Relay, X.25, ATM). Dead interval - це період часу, впродовж якого OSPF маршрутизатор буде очікувати, перш ніж обірве суміжність із сусідом. Dead інтервал за замовчуванням в 4 рази більше, ніж Hello interval.

Для того, щоб маршрутизатори могли сформувати суміжності, мають збігатися їх Hello інтервали, Dead інтервали, типи мереж і маски підмереж. Команда ***show ip ospf neighbors*** може бути використана для перевірки OSPF суміжності.

жностей.

OSPF ***router ID*** використовується, щоб унікально ідентифікувати кожний маршрутизатор в OSPF домені маршрутизації. Cisco маршрутизатори призначають ***router ID,*** ґрунтуючись на трьох критеріях з наступною перевагою:

1. Використовується IP адреса, сформована командою OSPF ***router-id*** .
2. Якщо ***router-id*** не сформовано, маршрутизатор вибирає саму старшу IP адресу з loopback інтерфейсів.
3. Якщо немає сконфігурованих loopback інтерфейсів, маршрутизатор вибирає самий старшу активну IP адресу з його фізичних інтерфейсів.

RFC 2328 не вказує які величини потрібно використовувати, щоб визначити вартість. Cisco IOS використовує накопичені пропускні здатності вихідних інтерфейсів маршрутизатора до мережі призначення, як вартості.

Мережі множинного доступу можуть створювати дві проблеми для OSPF, які пов'язані з лавинним розсиланням LSAs, включаючи створення множинних суміжностей – одна суміжність на кожну пару маршрутизаторів, і надлишкове лавинне розсилання LSAs (Link-State Advertisements). OSPF вибирає DR (Designated Router), щоб він працював як місце збору й поширення LSAs у мережі множинного доступу. BDR (Backup Designated Router) вибирається, щоб прийняти на себе роль DR, якщо DR вийде з ладу. Усі інші маршрутизатори відомі як DROthers. Усі маршрутизатори посилають свої LSAs до DR, який потім поширює LSA усім іншим маршрутизаторам у мережі множинного доступу.

Маршрутизатор з найвищим ***router ID*** стає DR, а маршрутизатор з наступним найбільшим router ID стає BDR. Це може бути змінене командою зміни пріоритету інтерфейсу ***ip ospf priority***. За замовчуванням у мережах множинного доступу OSPF пріоритет дорівнює 1 для всіх інтерфейсів. Значення пріоритету "0" означає, що маршрутизатор непридатний стати DR або BDR.

Поширення маршруту за замовчуванням в OSPF подібне до виконання даної операції в RIP, для цього використовується команда ***default-information originate***.

Команда ***show ip protocols*** використовується для перевірки важливої конфігураційної інформації OSPF, включаючи OSPF ***process ID***, ***router ID*** і мереж, які маршрутизатор анонсує.

8.5.2 Питання для самоперевірки

1. Чи повинні в команді ***router ospf*** на всіх маршрутизаторах збігатися ***router-id*** ?
2. По наданій конфігурації визначте Router ID для RouterA.

```
RouterA(config)#interface serial 0/0/0
RouterA(config-if)#ip add 192.168.2.1 255.255.255.252
RouterA(config)#interface loopback 0
RouterA(config-if)#ip add 10.1.1.1 255.255.255.255
RouterA(config)#router ospf 1
RouterA(config-if)#network 192.168.2.0 0.0.0.3 area 0
```

3. Яка команда може бути використана для визначення величини пропускної здатності інтерфейсу, яка використовується при розрахунках вартості OSPF?
4. Яка команда може бути використана для зміни OSPF вартості інтерфейсу без зміни значення пропускної здатності інтерфейсу?
5. Яке значення за замовчуванням Hello інтервалу для Ethernet мереж і мереж типу point-to-point? Який Hello інтервал для NBMA мереж?
6. Які величини повинні збігатися у двох маршрутизаторів, щоб вони могли сформувати OSPF суміжність?
7. Які проблеми допомагають розв'язати вибори DR і BDR?
8. Як відбуваються вибори DR і BDR?
9. Якщо DR вийшов з ладу, як визначається новий DR?
10. Що відбудеться, якщо в мережу, де вже обрані DR і BDR, додати маршрутизатор з більш високим пріоритетом?
11. Що означає, якщо пріоритет OSPF інтерфейсу дорівнює 0?
12. Яка команда повинна використовуватися для поширення маршруту за замовчуванням з використанням OSPF?

8.5.3 Матеріали для самостійного поглибленого вивчення теми

Можна повністю або частково вивчити RFC 2328 - поточний RFC для OSPF version 2. Цей документ завжди доступний на веб-сайті www.ietf.org (http://www.ietf.org/rfc/rfc2328.txt).

Справжні переваги використання OSPF у великих мережах можна побачити, вивчивши множинні області OSPF. Для цього можна використати наступні ресурси:

- Routing TCP/IP, Volume I, by Jeff Doyle and Jennifer Carroll
- OSPF, Anatomy of an Internet Routing Protocol, by John Moy

ЛАБОРАТОРНИЙ ПРАКТИКУМ

Виконання лабораторних робіт супроводжує весь теоретичний матеріал. Виконання кожної лабораторної роботи бажано проводити паралельно з вивченням теоретичного матеріалу.

Більшість робіт проводиться з використанням програмного засобу для проектування, моделювання та дослідження комп'ютерних мереж Packet Tracer. Оскільки роботи базуються на симуляторах, бажано встановити Packet Tracer останньої версії.

До кожної роботи додається ретельна інструкція та перевірочний бланк. Необхідно довести кожну роботу до змістовного завершення, про що можна отримати повідомлення у перевірочному бланку. Якщо під час виконання роботи було допущено помилки, вони будуть відображатися у перевірочному бланку.

Найбільш важливі роботи, а саме конфігурування протоколів динамічної маршрутизації RIPv2 та OSPF, потрібно продублювати на реальному обладнані Cisco.

Хоча в електронному посібнику лабораторні роботи наводяться до будь якого теоретичного матеріалу, серед них можна виділити найважливіші. Виконання нижче зазначених лабораторних робіт є обов'язковим для успішного засвоєння матеріалу курсу:

1. Використання CDP для виявлення мереж.
2. Конфігурація статичних маршрутів.
3. Розподіл мереж на підмережі за сценарієм.
4. Основи конфігурування RIPv1.
5. Пошук несправностей у маршрутизації RIPv1.
6. Агрегація маршрутів.
7. Основи конфігурування RIPv2.
8. Пошук несправностей у маршрутизації RIPv2.
9. Основи конфігурування OSPF.

ГЛОСАРІЙ

AD – Administrative Distance – Адміністративна відстань – величина, яка характеризує надійність джерела інформації про маршрут.

Adjacent neighbor – суміжний пристрій –так називають два безпосередньо з'єднаних маршрутизатори, які обмінюються маршрутною інформацією

AS –Autonomous System – автономна система, набір мереж, які знаходяться у одному адміністративному домені.

CDP – Cisco Discovery Protocol – протокол, який використовується пристроєм для отримання інформації про сусідні пристрої.

CIDR – Classless InterDomain Routing – безкласова між доменна маршрутизація – технологія, яка дозволяє виконувати агрегацію маршрутів, за рахунок чого зменшується об'єм таблиць маршрутизації.

Convergence – конвергенція – здатність групи пристроїв, які використовують один протокол маршрутизації, погоджувати інформацію про топологію мережі, після змін в топології.

Dynamic Routing – динамічна маршрутизація – різновид мережної маршрутизації, яка автоматично враховує зміни мережної топології і характер трафіку.

EGP – Exterior Gateway Protocol – протокол зовнішнього шлюзу, протокол маршрутизації, який призначено для використання між автономними системами.

EIGRP – Enhanced Interior Gateway Routing Protocol – вдосконалений протокол маршрутизації внутрішнього шлюзу, розроблений Cisco.

IGP – Interior Gateway Protocol – протокол внутрішнього шлюзу, протокол маршрутизації, який призначено для використання в автономній системі.

IGRP – Interior Gateway Routing Protocol – протокол маршрутизації внутрішнього шлюзу, розроблений Cisco.

Global configuration mode – режим глобальної конфігурації – використовується для введення команд, які вносять зміни до глобальної конфігурації маршрутизатора.

Keepalive – тестовий пакет – повідомлення, яке відправляється одним мережним пристроєм та сигналізує іншому пристрою, про працездатність віртуального каналу між ними.

Metric –метрика – числове значення, яке виробляється алгоритмом маршрутизації для кожного маршруту в мережі. Чим менше метрика, тим більше переваги надається маршруту.

NVRAM – Non-Volatile RAM – енергонезалежна пам'ять

OSPF – Open Short Path First – відкритий протокол пошуку найкоротшого шляху – використовує алгоритм маршрутизації з урахуванням стану каналу.

Ping - Packet Internet Groper – програма, яка надсилає ехо-запит і отримує на нього ехо-відгук за протоколом ICMP. Часто використовується у IP мережах для виявлення досяжності пристрою.

Propagation delay – затримка розповсюдження - час, який потрібен даним, аби пройти від відправника до адресата, тобто час на передачу сигналу.

RIP – Routing Information Protocol – протокол маршрутної інформації, протокол внутрішнього шлюзу, який у якості метрики використовує кількість переходів.

Router – маршрутизатор, це пристрій Мережного рівня, який використовує одну чи декілька метрик для виявлення оптимального шляху доставки даних.

Routing – маршрутизація – процес знаходження маршруту до адресату.

Routing protocol – протокол маршрутизації – це протокол, який забезпечує маршрутизацію, за рахунок реалізації деякого алгоритму маршрутизації.

Routing table – таблиця маршрутизації – перелік відомих маршрутизатору маршрутів та відповідних до них інтерфейсів.

Routing updates – оновлення маршрутизації – це повідомлення, які містять інформацію про досяжність мережі та відповідну оцінку маршруту. Вони розсилаються між маршрутизаторами, зазвичай періодично.

SPF algorithm – Shortest Path First algorithm – алгоритм вибору найкоротшого маршруту. Це обчислення, які виконуються над базою даних з метою отримання SPF дерева.

Split Horizon – розщеплення обрію – механізм маршрутизації, за допомогою якого уникають появи петель маршрутизації. Розсилка інформації не відбувається через той інтерфейс маршрутизатора, через який вона була отримана.

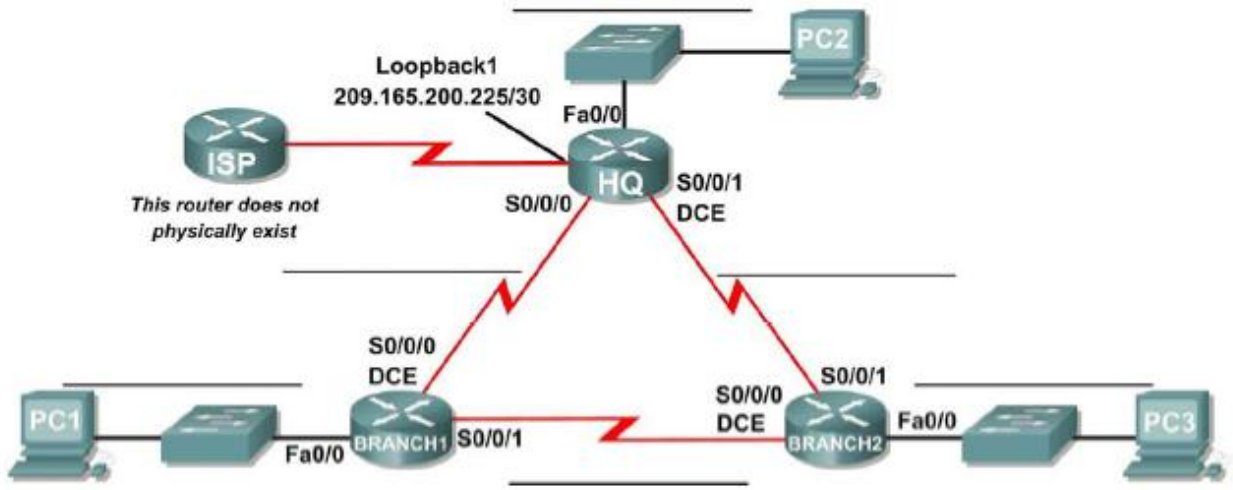
Static Routing – статична маршрутизація – процес визначення та конфігурування маршрутів вручну.

Triggered updates – миттєві оновлення – це повідомлення про зміни маршрутизації, які надсилаються у разі змін топології мережі, не чекаючи поки мине таймер анонсів.

WAN – Wide-Area-Network - розподілена мережа, передачі даних, яка охоплює значний географічний простір.

ЗАВДАННЯ ДО КОНТРОЛЬНОЇ РОБОТИ

Для топології



- Розробити адресну схему VLSM за своїм варіантом.
- У Packet Tracer побудувати топологію, застосувати розроблену адресну схему.
- Конфігурувати маршрутизацію OSPF у середині автономної області.
- Сформувати статичні маршрути для зв’язку з маршрутизатором провайдера.
- Провести перерозподіл статичного маршруту за замовченням засобами OSPF.
- Перевірити наявність зв’язку в системі.

Оформлення контрольної роботи

Необхідно надати розрахунки за своїм варіантом адресної схеми по VLSM розподілу на підмережі, саму адресну схему, а також для кожного маршрутизатора роздруковані з Packet Tracer файл running-config та таблицю маршрутизації.

Варіанти завдань для розробки адресної схеми

№ вар	Адрес сети для LAN	HQ LAN	Branch1 LAN	Branch 2 LAN	Адрес сети для WAN	Адрес сети для loopback ISP
1	172.16.0.0/16	500	200	100	192.168.1.16/28	209.165.200.224/30
2	172.16.32.0/20	500	200	100	192.168.1.32/28	209.165.200.224/30
3	172.16.64.0/19	500	200	100	192.168.2.16/28	209.165.200.224/30
4	172.16.64.0/18	500	200	100	192.168.2.32/28	209.165.200.224/30
5	172.17.0.0/16	250	200	150	192.168.3.16/28	209.165.200.224/30
6	172.17.0.0/18	250	200	150	192.168.3.32/28	209.165.200.224/30
7	172.17.0.0/20	250	200	150	192.168.4.16/28	209.165.200.224/30
8	172.17.64.0/18	250	200	150	192.168.4.32/28	209.165.200.224/30
9	172.18.0.0/16	100	140	210	192.168.10.16/28	209.165.200.224/30
10	172.18.64.0/18	100	140	210	192.168.10.32/28	209.165.200.224/30
11	172.18.64.0/20	100	140	210	192.168.12.16/28	209.165.200.224/30
12	172.18.64.0/19	100	140	210	192.168.12.32/28	209.165.200.224/30
13	172.20.0.0/16	400	100	300	192.168.12.48/28	209.165.200.224/30
14	172.20.128.0/18	400	100	300	192.168.12.64/28	209.165.200.224/30
15	172.20.64.0/20	400	100	300	192.168.16.16/28	209.165.200.224/30
16	172.20.64.0/19	400	100	300	192.168.16.32/28	209.165.200.224/30
17	172.21.0.0/16	300	100	130	192.168.16.48/28	209.165.200.224/30
18	172.21.32.0/21	50	100	100	192.168.16.64/28	209.165.200.224/30
19	172.21.48.0/22	50	100	100	192.168.32.16/28	209.165.200.224/30
20	172.21.32.0/22	50	100	100	192.168.32.32/28	209.165.200.224/30
21	172.21.64.0/20	50	100	100	192.168.32.48/28	209.165.200.224/30

ПИТАННЯ З ПІДГОТОВКИ ДО ІСПИТУ

1. Апаратні компоненти маршрутизатора.
2. Процес завантаження маршрутизатора.
3. Призначення маршрутизатора.
4. Міжмережна операційна система.
5. Формування таблиці маршрутизації.
6. Принципи таблиці маршрутизації
7. Використання CDP для виявлення мереж
8. Конфігурація статичних маршрутів
9. Історія та перспективи протоколів динамічної маршрутизації.
10. Класифікація протоколів динамічної маршрутизації.
11. Поняття метрики.
12. Адміністративна відстань.
13. Характеристики дистанційно-векторних протоколів маршрутизації.
14. Поняття конвергенції.
15. Заходи для підтримки таблиці маршрутизації дистанційно-векторними протоколами маршрутизації.
16. Визначення петлі маршрутизації та причини їх появи.
17. Механізми запобігання петель маршрутизації у дистанційно-векторних протоколах маршрутизації.
18. Сучасні дистанційно-векторні протоколи маршрутизації.
19. Основні характеристики RIPv1.
20. Основи конфігурування RIPv1.
21. Засоби перевірки і пошуку несправностей в RIPv1.
22. Автоматична сумаризація, її переваги та недоліки.
23. Поширення маршруту за замовченням засобами RIPv1.
24. Класова і безкласова маршрутизація.
25. CIDR та супермаршрути.
26. Обмеження RIPv2.
27. RIPv2 і VLSM.
28. RIPv2 і CIDR.
29. Перевірка і пошук несправностей RIPv2.
30. SPF алгоритм.

31. Процес маршрутизації з урахуванням стану каналу.
32. Переваги протоколів маршрутизації з урахуванням стану каналу.
33. Вимоги протоколів маршрутизації з урахуванням стану каналу.
34. Історія OSPF.
35. Типи OSPF пакетів.
36. \Базова конфігурація OSPF.
37. Визначення OSPF Router ID.
38. Метрика OSPF.
39. OSPF і мережі множинного доступу.
40. Тонке настроювання OSPF.

ЛІТЕРАТУРА

1. Програма мережної академії Cisco CCNA Exploration 4.0 «Routing Protocols and Concepts» – електронний курс, 2009р.
2. Англо-український тлумачний словник з обчислювальної техніки, Інтернету і програмування. – Вид.1. – К.: Видавничий дім «СофтПрес», 2005. – 552с.
3. RFC 791 "Internet Protocol," http://www.ietf.org/rfc/rfc791.txt
4. RFC 1058 “Routing Information Protocol," http://www.ietf.org/rfc/rfc1058.txt
5. "ISC Domain Survey: Number of Internet Hosts," https://www.isc.org/solutions/survey/history
6. "Internet Multicast Addresses," http://www.iana.org/assignments/multicast-addresses
7. "A Brief History of the Internet," http://www.isoc.org/internet/history/brief.shtml
8. "Internet Protocol v4 Address Space," http://www.iana.org/assignments/ipv4-address-space
9. RFC 1519 "Classless Inter-Domain Routing (CIDR): an Address Assignment and Aggregation Strategy," http://www.ietf.org/rfc/rfc1519.txt
10. RFC 1723 “RIP Version 2,” http://www.ietf.org/rfc/rfc1723.txt
11. RFC 2328 "OSPF Version 2," http://www.ietf.org/rfc/rfc2328.txt

Методичне видання

Н. П. Полякова
доцент

МЕРЕЖНА МАРШРУТИЗАЦІЯ ТА КОМУТАЦІЯ

Навчально-методичний посібник

для студентів ЗДІА
спеціальності 121 «Інженерія програмного забезпечення»
денної та заочної форм навчання

Підписано до друку 04.07.2018р. Формат 60х84 1/32. Папір офсетний.
Умовн. друк. арк. 8,9. Наклад 1 прим. Ціна 52,39 грн.
Внутрішній договір № 159/18

Запорізька державна інженерна академія
Свідоцтво про внесення до Державного реєстру суб'єктів
видавничої справи ДК № 2958 від 03.09.2007 р.

Віддруковано друкарнею
Запорізької державної інженерної академії
з оригінал-макету авторів

69006, м. Запоріжжя, пр. Соборний, 226
ЗДІА