

Лабораторна робота №5. Спостереження конвергенції мережі.

Найскладніша проблема в будь-якому протоколі динамічної маршрутизації - кільцеві маршрути. В дистанційно-векторних протоколах маршрутизації вона вирішується за рахунок використання безлічі різноманітних механізмів, час роботи яких, а отже, і час конвергенції мережі, може становити кілька хвилин після відмови якогось каналу. В протоколах з урахуванням стану каналів проблема кільцевих маршрутів вирішується за рахунок того, що кожен маршрутизатор зберігає у себе повну топологічну базу мережі, а також використовує складні математичні алгоритми для її обробки.

Протокол EIGRP виявляє кільцеві маршрути, і відбувається процес їх знищення за рахунок того, що кожен пристрій зберігає у себе деяку базову топологічну інформацію, а завантаження центрального процесора (CPU) пристрою помітно знижується за рахунок того, що така топологічна інформація досить коротка. Коли маршрутизатор виявляє кілька маршрутів до однієї і тієї ж IP-підмережі, він поміщає тільки найкращий маршрут в таблицю маршрутизації. Топологічна інформація про мережу потрібна протоколу EIGRP для того ж, для чого вона потрібна протоколу маршрутизації OSPF: щоб після зміни топології відреагувати на зміну як можна швидше і використовувати новий маршрут, а також уникнути виникнення кільцевих (циклічних) маршрутів в мережі. Фактично протокол маршрутизації EIGRP зберігає записи про наступні транзитних маршрутизаторах і деякі деталі маршрутів до них, але не оперує топологічною інформацією про мережу за найближчими транзитними маршрутизаторами. Менш докладна інформація про топологію мережі не вимагає чогось схожого на складний алгоритм SPF, тому конвергенція відбувається набагато швидше, вимагає менше ресурсів і ймовірність виникнення кільцевих маршрутів за рахунок цього зменшується.

Основними перевагами EIGRP є:

- низьке споживання мережевих ресурсів в режимі нормальної експлуатації (в умовах стабільної мережі передаються тільки пакети "hello")
- при виникненні змін по мережі передаються тільки зміни, що відбулися в маршрутній таблиці, а не вся таблиця цілком; це дозволяє зменшити навантаження на мережу, створювану протоколом маршрутизації
- малий час конвергенції в разі зміни в топології мережі (в окремих випадках збіжність забезпечується майже миттєво)

протокол EIGRP є вдосконаленим протоколом дистанційної-векторної маршрутизації, в якому для розрахунку найкоротшого шляху до кінцевого адресою використовується алгоритм дифузного поновлення (Diffused Update Algorithm - DUAL)..

Основні версії протоколу

Існують дві основні версії протоколу EIGRP - версія 0 і 1. У ранніх версіях програмного забезпечення Cisco IOS (аж до версій 10.3 (11), 11.0 (8) і 11.1 (3)) використовується більш рання версія протоколу EIGRP (з цієї причини деякі пояснення, що містяться в цьому документі, можуть бути не дійсні для цих версій). Ми настійно рекомендуємо

використовувати останню версію EIGRP, оскільки ця версія містить безліч поліпшень, пов'язаних зі стабільністю і продуктивністю.

Основні принципи

При розрахунку найкращого шляху до кінцевого адресою типовий дистанційно-векторний протокол зберігає наступну інформацію: відстань (distance) (сумарна метрика або відстань, наприклад, лічильник переходів) і вектор (наступний перехід). Наприклад, на всіх маршрутизаторах мережі на рис. 1 виконується протокол маршрутної інформації (Routing Information Protocol - RIP). Маршрутизатор 2 вибирає шлях до мережі А, перевіряючи число переходів для кожного наявного шляху.

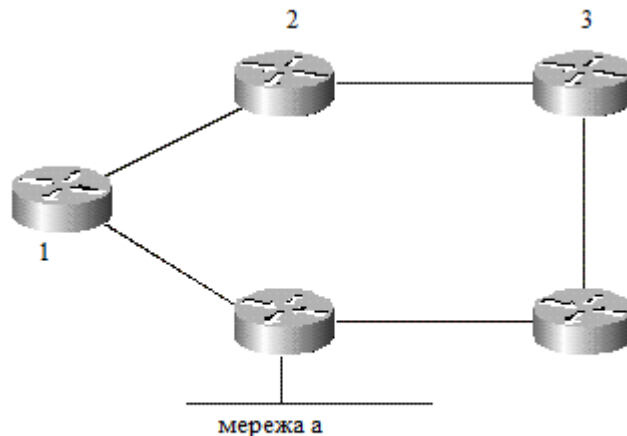


Рис.1

Оскільки довжина шляху, що проходить через Маршрутизатор 3, дорівнює трьом переходам, а довжина шляху через маршрутизатор 1 дорівнює двом переходам, маршрутизатор 2 вибирає шлях, що йде через Маршрутизатор 1, і відкидає інформацію, отриману від Маршрутизатора 3. Якщо маршрут, який пролягає між маршрутизатором 1 і мережею А, порушується, то маршрутизатор 2 втрачає зв'язок з цим пунктом призначення доти, поки цей маршрут не блокуватиме в таблиці маршрутизації по часу простою, яке дорівнює трьом періодам оновлення (90 секунд); а маршрутизатор 3 оголошує маршрут повторно (що відбувається в RIP кожні 30 секунд). Якщо не враховувати період утримання, то маршрутизатора 2 буде потрібно 90-120 секунд, щоб перевести шлях з маршрутизатора 1 на маршрутизатор 3.

Замість того, щоб розраховувати на повні регулярні оновлення для виконання повторної збіжності, EIGRP (замість відкидання даних) будує таблицю топології використовуючи для цього всі оголошення своїх сусідів і виконує збіжність або за допомогою пошуку підходящого маршруту без петель в таблиці топології, або (якщо про такий маршрут нічого невідомо) за допомогою опитування своїх сусідів. Другий маршрутизатор зберігає інформацію, отриману від першого і третього маршрутизаторів. Він вибирає шлях через маршрутизатор 1 як кращий шлях ("наступник"), а шлях через маршрутизатор 3 - як шлях без петель (можливий наступник). Коли маршрут через маршрутизатор 1 стає недоступним, маршрутизатор 2 в пошуках можливого наступника звертається до таблиці топології і відразу ж починає використовувати маршрут через маршрутизатор 3.

З цих коротких пояснень очевидно, що EIGRP повинен забезпечити наступне:

- систему, при якій EIGRP пересилає оновлення, необхідні в даний момент (це досягається за допомогою виявлення та обслуговування сусідів)
- спосіб, що дозволяє визначити, який з маршрутів, отриманий маршрутизатором, немає петель
- процедуру, що дозволяє видаляти неробочі маршрути з таблиць топології на всіх маршрутизаторах, що знаходяться в мережі
- процедуру опитування сусідів, яка дозволить знайти нові маршрути до кінцевого адресою замість втрачених старих маршрутів

Всі ці вимоги будуть розглянуті далі.

Виявлення та обслуговування сусідів

Для поширення маршрутної інформації по мережі в EIGRP використовується неперіодичне інкрементне оновлення маршрутів. Це означає, що EIGRP пересилає оновлення тільки для змінюються маршрутів і тільки в той момент, коли такі маршрути змінюються.

Основний недолік таких оновлень полягає в тому, що ви можете не впізнати, в який час маршрут, що проходить через сусідній маршрутизатор, став недоступним. Не можна блокувати за часом маршрути, які очікують отримання нової таблиці маршрутизації від сусіда. Для надійної пересилання змін до маршрутної таблиці в EIGRP використовується механізм взаємодії між сусідніми маршрутизаторами (два маршрутизатора стають "сусідами" в тому випадку, якщо кожен з них отримує пакети "hello" від свого сусіда).

EIGRP посилає пакети "hello" кожні 5 секунд (для каналів з високою пропускнуою здатністю) і кожні 60 секунд (для багатоточкових каналів з низькою пропускнуою здатністю).

- п'ятисекундними пакет "hello" використовується:
 - в мережевих середовищах (наприклад, Ethernet, Token Ring і FDDI)
 - в послідовних каналах "точка-точка" (наприклад, виділені лінії, що використовують протоколи PPP або HDLC; підлеглі інтерфейси Frame Relay типу "точка-точка" і підлеглий інтерфейс ATM)
 - багатоточкові лінії з високою пропускнуою здатністю (вище, ніж у лінії T1) (наприклад, ISDN PRI і Frame Relay)
- Шестидесяти секундний пакет "hello ":
 - багатоточкові лінії з пропускнуою здатністю як у T1 або нижче - як у багатоточкових інтерфейсів Frame Relay, багатоточкових інтерфейсів ATM, комутованих віртуальних каналів ATM і базових інтерфейсів обміну ISDN

Частота, з якою EIGRP відправляє пакети "hello", називається hello-інтервалом; цей параметр можна налаштовувати для кожного інтерфейсу окремо за допомогою команди `ip hello-interval eigrp`. Час утримання - час, протягом якого маршрутизатор буде вважати сусідню пристрій чинним, не отримуючи при цьому від нього пакет "hello". Час утримання зазвичай

дорівнює трьом hello-інтервалам (15 і 180 секунд за замовчуванням). Час утримання налаштовується за допомогою команди `ip hold-time eigrp`.

Зверніть увагу на те, що при зміні hello-інтервалу час утримання не коригується автоматично (час утримання необхідно змінити вручну відповідно до новим значенням hello-інтервалу).

Два EIGRP-маршрутизатора можуть стати сусідами навіть в тому випадку, якщо таймер hello і таймер утримання не збігаються. Час утримання вказується в пакетах hello, тому кожен сусід повинен залишатися в робочому стані, навіть якщо інтервал hello і таймер утримання не збігаються.

Величину hello-інтервалу, встановлену на маршрутизаторі, неможливо визначити безпосередньо. Однак цю величину можна з'ясувати з лістингу команди `show ip eigrp neighbor`, виконаної на сусідньому маршрутизаторі.

Якщо лістинг команди `show ip eigrp neighbor` отримано з пристрою Cisco, то в цьому випадку для інтерпретації результатів можна скористатися утилітою Output Interpreter (тільки для зареєстрованих клієнтів). Дана утиліта відображає потенційні проблеми і пропонує способи їх вирішення. Для використання Output Interpreter необхідно включити `JavaScript.router#`

show ip eigrp neighbor

IP-EIGRP neighbors for process 1

H	Address	Interface	Hold	Uptime	SRTT	RTO	Q	Seq	Type
			(sec)	(ms)	Cnt	Num			
1	10.1.1.2	Et1	13	12:00:53	12	300	0	620	
0	10.1.2.2	S0	174	12:00:56	17	200	0	645	

rp-2514aa# show ip eigrp neighbor

IP-EIGRP neighbors for process 1

H	Address	Interface	Hold	Uptime	SRTT	RTO	Q	Seq	Type
			(sec)	(ms)	Cnt	Num			
1	10.1.1.2	Et1	12	12:00:55	12	300	0	620	
0	10.1.2.2	S0	173	12:00:57	17	200	0	645	

rp-2514aa# show ip eigrp neighbor

IP-EIGRP neighbors for process 1

H	Address	Interface	Hold	Uptime	SRTT	RTO	Q	Seq	Type
			(sec)	(ms)	Cnt	Num			
1	10.1.1.2	Et1	11	12:00:56	12	300	0	620	
0	10.1.2.2	S0	172	12:00:58	17	200	0	645	

Значення, вказане в лістингу команди в стовпці "Hold", ніколи не повинно перевищувати значення часу утримання, а також не повинно бути менше різниці значень часу утримання та інтервалу між повідомленнями привітання (якщо, звичайно, пакети вітання не були втрачені). Якщо значення в стовпці "Hold" зазвичай варіюється від 10 до 15 секунд, то інтервал вітання становить 5 секунд, а час утримання - 15 секунд. Якщо значення в стовпці "Hold" зазвичай варіюється від 120 до 180 секунд, то інтервал вітання становить 60 секунд, а час утримання - 180 секунд. Якщо значення не збігаються зі значеннями таймера, заданими за умовчанням, тоді необхідно перевірити відповідний інтерфейс на сусідньому маршрутизаторі (таймер "hello" і таймер інтервалу вітання могли бути змінені вручну).

Примітка:

- EIGRP не створює однорангові зв'язку з використанням вторинних адрес. Джерело всього трафіку EIGRP - основний адресу інтерфейсу.
- При налаштуванні EIGRP по мережі Frame Relay з множинним доступом (багатоточковому з'єднанню і ін.) В операторах frame-relay map слід налаштувати ключове слово broadcast. Суміжність двох маршрутизаторів EIGRP не буде створена без ключового слова broadcast. Більш детально див. Розділ "Налаштування та усунення неполадок Frame Relay".
- Кількість сусідніх вузлів, підтримуваних EIGRP, не обмежена. Фактична кількість підтримуваних сусідніх вузлів залежить від характеристик пристрою:
 - обсяг пам'яті
 - обчислювальна потужність
 - обсяг переданої інформації (наприклад, кількість переданих маршрутів)
 - рівень складності топології
 - стабільність мережі

Побудова таблиці топології

Тепер ми знаємо, яким чином маршрутизатори спілкуються між собою. А ось про що вони "кажуть" ?. Звичайно - про таблиці топології. На відміну від протоколів RIP і IGRP, EIGRP не використовує таблицю маршрутизації (або перенаправлення) для зберігання всіх даних, необхідних для його роботи. Натомість EIGRP формує другу таблицю (таблицю топології), на основі якої здійснюється установка маршрутів в таблиці маршрутизації.

Примітка: В Cisco IOS починаючи з версій 12.0T і 12.1 протокол RIP веде свою власну базу даних, звідки маршрути надходять в таблицю маршрутизації.

Щоб дізнатися основний формат таблиці топології на EIGRP-маршрутизатора, виконайте команду show ip eigrp topology. У таблиці топології містяться відомості, необхідні для побудови набору відстаней і векторів для кожної досяжною мережі, включаючи:

- найменша пропускна здатність на маршруті, що йде до даної мережі (дані надаються висхідним сусідом)
- сумарна затримка
- надійність маршруту
- завантаження маршруту
- мінімальний шлях максимального розміру переданого блоку даних (MTU)
- можлива відстань
- оголошене відстань

- джерело маршруту (зовнішні вузли маркуються)

Імовірно й фактичне відстані обговорюватимуться далі в цьому розділі.

Якщо лістинг команди `show ip eigrp topology` отримано з пристрою Cisco, то в цьому випадку для інтерпретації результатів можна скористатися утилітою Output Interpreter (тільки для зареєстрованих клієнтів). Дана утиліта відображає потенційні проблеми і пропонує способи їх вирішення. Для використання Output Interpreter необхідно включити JavaScript.

Метрики EIGRP

Для розрахунку метрик маршрутизації протокол EIGRP використовує мінімальну пропускну здатність маршруту до кінцевої адреси, а також сумарну затримку. Можна також налаштувати й інші метрики. Проте ми не рекомендуємо робити цього, оскільки в цьому випадку у вашій мережі можуть з'явитися петлі по маршрутизації. Метрики пропускну здатності і затримки визначаються на основі значень, встановлених на інтерфейсах маршрутизаторів, які є частиною маршруту до мережі призначення.

На рисунку 2 показано, як маршрутизатор 1 обчислює найкращий шлях в мережу А.

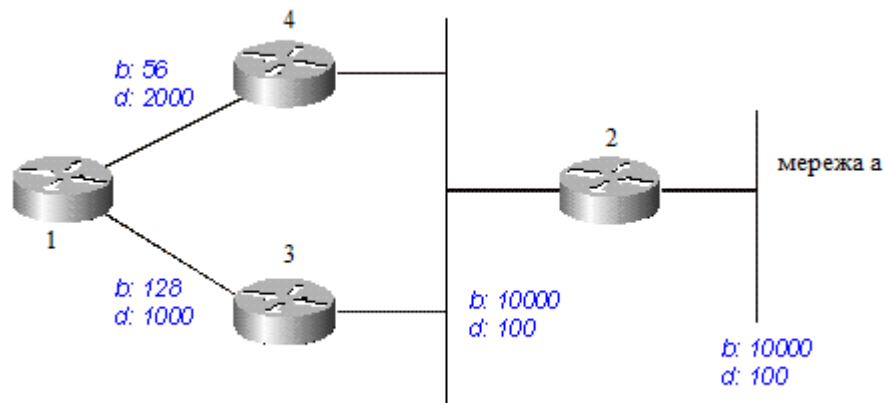


Рис.2.

Розрахунок починається з двох оголошень, що стосуються шуканої мережі: перший маршрут проходить через маршрутизатор 4 (мінімальна пропускну здатність - 56 і сумарна затримка - 2200), а другий - через маршрутизатор 3 (мінімальна пропускну здатність - 128 і сумарна затримка - 1200). Маршрутизатор 1 вибирає маршрут з найменшим значенням метрики.

Тепер обчислимо метрику. EIGRP обчислює сумарну метрику шляхом зважування метрики пропускну здатності і метрики затримки. Для вимірювання пропускну здатності EIGRP використовує наступну формулу:

- $\text{пропуск. способн.} = (10000000 / \text{пропуск. Способн. (I)}) * 256$

де пропускну спроможність (i) - найменша пропускну спроможність усіх вихідних інтерфейсів, що входять до складу маршруту, що веде до мережі призначення (вимірюється в кілобітах).

Для обчислення затримки EIGRP використовує наступну формулу:

- $\text{затримка} = \text{затримка (i)} * 256$

де затримка (i) - сума затримок, які налаштовані на всіх інтерфейсах, що входять до складу маршруту до мережі призначення (вимірюється в десятках мікросекунд). У лістингу команд

show ip eigrp topology і show interface затримка вказана в мікросекундах, тому для використання цих значень у формулі їх необхідно розділити на 10. У цьому документі використовується затримка, налаштована і показана на інтерфейсі.

Зазначені вище значення використовуються для визначення сумарної метрики для шуканої мережі:

- метрика = $[K1 * \text{пропускна здатність} + (K2 * \text{пропускна здатність}) / (256 - \text{навантаження}) + K3 * \text{затримка}] * [K5 / (\text{надійність} + K4)]$

Примітка: значення K слід використовувати тільки після ретельного планування. Неправильні значення K призводять до неможливості побудови зв'язків між сусідніми вузлами, в результаті чого у вашій мережі буде неможливо виконати збіжність.

Примітка: при $K5 = 0$ формула приймає наступний вигляд: метрика = $[k1 * \text{пропускна здатність} + (k2 * \text{пропускна здатність}) / (256 - \text{навантаження}) + k3 * \text{затримка}]$.

За замовчуванням K має наступні значення:

- $K1 = 1$
- $K2 = 0$
- $K3 = 1$
- $K4 = 0$
- $K5 = 0$

Для функціонування за замовчуванням формулу можна спростити наступним чином:

$$\text{metric} = \text{bandwidth} + \text{delay}$$

Маршрутизатори Cisco не виконують розрахунки з плаваючою комою, тому для правильного підрахунку метрики на кожному етапі калькуляції отриманий результат необхідно округляти в меншу сторону до цілого числа. В цьому прикладі загальна вартість маршруту через маршрутизатор 4 дорівнює:

$$\text{minimum bandwidth} = 56k$$

$$\text{total delay} = 100 + 100 + 2000 = 2200$$

$$[(10000000/56) + 2200] \times 256 = (178571 + 2200) \times 256 = 180771 \times 256 = 46277376$$

А общая стоимость маршрута через маршрутизатор 3 равна:

$$\text{minimum bandwidth} = 128k$$

$$\text{total delay} = 100 + 100 + 1000 = 1200$$

$$[(10000000/128) + 1200] \times 256 = (78125 + 1200) \times 256 = 79325 \times 256 = 20307200$$

Таким чином, щоб досягти мережі А, маршрутизатор 1 вибирає маршрут через маршрутизатор 3.

Зверніть увагу на те, що значення пропускної здатності і затримки - це значення, налаштовані на інтерфейсі, через який маршрутизатор досягає наступного сегмента по шляху до мережі призначення. Наприклад, маршрутизатор 2 оголосив про мережу А і вказав затримку, налаштовану на його Ethernet-інтерфейсі. Потім маршрутизатор 4 додав до цього значення затримку, налаштовану на його Ethernet-інтерфейсі, а маршрутизатор 1 додав затримку, налаштовану на його послідовному порті.

Можлива відстань, оголошена відстань і можливий наступник

Можлива відстань - це найкраща метрика по маршруту в мережу призначення, включаючи метрику до сусіда, який оголосив маршрут. Оголошена відстань є сумарною метрикою шляху до мережі призначення, який оголошений висхідним сусідом. Можливий заступник - це маршрут, величина оголошеного відстані якого менше величини можливої відстані (поточний найкращий маршрут). На рис. 3 зображений цей процес:

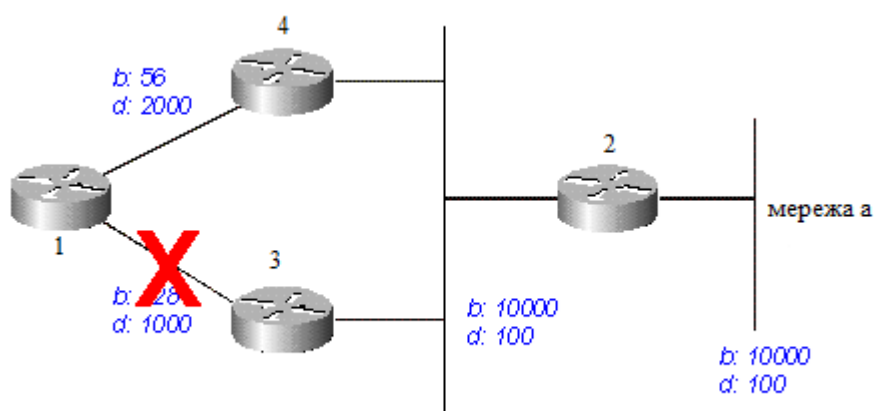


Рис.3

Маршрутизатор 1 визначив, що в мережу А мають два маршрути: один маршрут проходить через маршрутизатор 3, а другий - через маршрутизатор номер 4.

- Вартість маршруту, що проходить через маршрутизатор 4, дорівнює 46277376, а оголошене відстань - 307 200.
- Вартість маршруту, що проходить через маршрутизатор 3, дорівнює 20307200, оголошене відстань дорівнює 307200.

Зверніть увагу, що в кожному випадку EIGRP відраховує оголошене відстань починаючи з маршрутизатора, що оголосив маршрут до шуканої мережі. Іншими словами, оголошене відстань від маршрутизатора 4 - це метрика маршруту до мережі А, який починається з маршрутизатора 4, а оголошене відстань від маршрутизатора 3 - це метрика маршруту до мережі А починаючи з маршрутизатора 3. EIGRP вибирає маршрут через маршрутизатор 3 в якості найкращого і використовує метрику через маршрутизатор 3 як можливу відстань. Оскільки оголошене відстань до цієї мережі через маршрутизатор 4 менше можливої відстані, маршрутизатор 1 вважає шлях через маршрутизатор 4 можливим наступником.

Коли зв'язок між маршрутизаторами 1 і 3 порушується, маршрутизатор 1 перебирає всі відомі йому шляхи до мережі А, і визначає, що у нього є можливий наступник, доступний через маршрутизатор 4. Маршрутизатор 1 звертається до цього маршруту, використовуючи метрику через маршрутизатор 4 в якості нової можливої відстані. Збіжність виконується

миттєво, а оновлення, що надходять до нижчих сусідів, є єдиним трафіком, який генерується протоколом маршрутизації.

Тепер давайте розглянемо більш складний випадок, показаний на рис. 4.

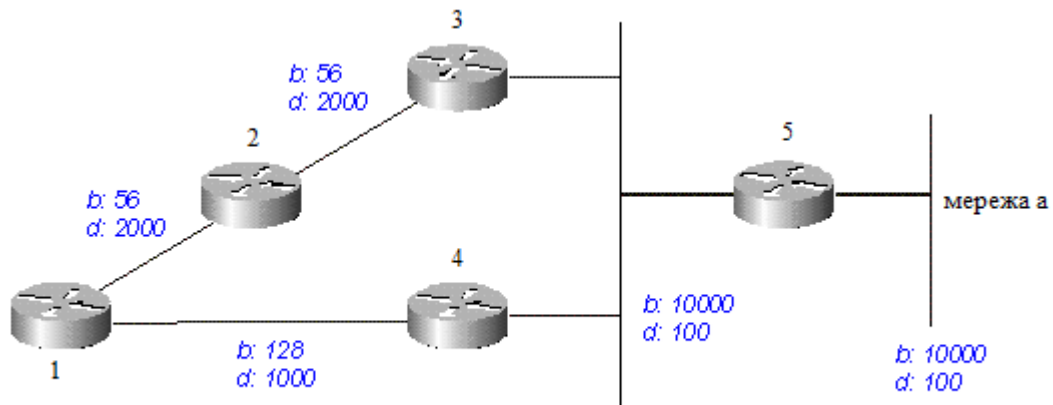


Рис.4

Від маршрутизатора 1 в мережу А ведуть два маршрути: один маршрут проходить через маршрутизатор і має метрику 46789376, а другий через маршрутизатор 4 і має метрику 20307200. В якості маршруту в мережу А маршрутизатор 1 вибере найменшу з двох метрик, яка стане можливим відстанню. Тепер розглянемо маршрут, що йде через маршрутизатор 2. Нам необхідно з'ясувати, чи підходить цей маршрут в якості можливого наступника. Оголошене відстань від другого маршрутизатора одно 46277376. Це значення вище значення можливої відстані, тому цей шлях не є можливим наступником. Якби ми подивилися таблицю топології маршрутизатора 1 (за допомогою команди `show ip eigrp topology`), то ми б побачили тільки одну запис для мережі А - через маршрутизатор 4. (В дійсності в таблиці топології маршрутизатора 1 мають два записи, але можливим наступником може бути тільки одна з них, інша ж запис в лістингу команди `show ip eigrp topology` відображено не буде; всі маршрути, які не є можливими наступниками, можна подивитися скориставшись командою `show ip eigrp topology all-links`).

Припустимо, що з'єднання між маршрутизаторами 3 і 4 було порушено. Виявивши, що єдиний шлях до мережі А втрачений, маршрутизатор 1 починає опитувати всіх своїх сусідів на предмет маршруту в мережу А. Оскільки маршрутизатор 2 може запропонувати маршрут в мережу А, він відгукнеться на запит, ініційований маршрутизатором 1. Оскільки найкращий маршрут, що йде через маршрутизатор 4, був втрачений, маршрутизатор 1 приймає маршрут, який йде в мережу А через маршрутизатор 2.

Визначення беспетлевого шляху

Яким чином протокол EIGRP використовує поняття можливої відстані, оголошеного відстані і можливого наступника для того, щоб визначити, чи є даний шлях дійсним і беспетлевого? На малюнку 4а показано, що маршрутизатор 3 виконує оцінку маршрутів в мережу А. Оскільки функція "розділений горизонт" відключена (наприклад, це необхідно при роботі з багатоточковими інтерфейсами Frame Relay), маршрутизатор 3 вкаже в мережу А три маршрути: через маршрутизатор 4, через маршрутизатор 2 (шлях проходить через маршрутизатор два, один, три і чотири) і через маршрутизатор 1 (шлях проходить через маршрутизатор один, два, три, чотири).

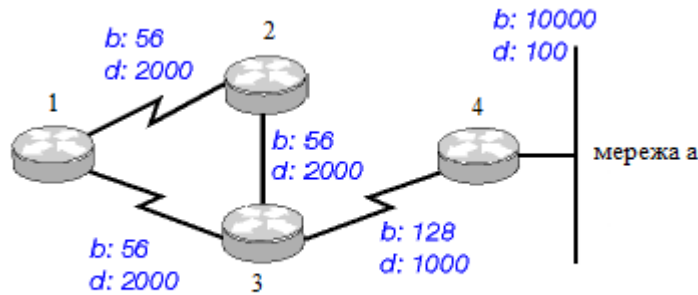


Рис.4

Якщо маршрутизатор прийме всі ці маршрути, то утворюється маршрутна петля. Маршрутизатор 3 вважає, що він зможе потрапити в мережу А через маршрутизатор 2, однак перш ніж потрапити в мережу А, шлях, що йде через маршрутизатор 2, проходить через маршрутизатор 3. Якщо з'єднання між маршрутизатором 4 і маршрутизатором 3 буде порушено, тоді маршрутизатор 3 буде вважати, що він зможе досягти мережі А використовуючи один з решти маршрутів, але через дію правил для визначення можливих наступників він ніколи не використовуватиме ці маршрути в якості альтернативних. Щоб це зрозуміти, звернемося до метриці:

- сумарна метрика в мережу А через маршрутизатор 4: 20281600
- сумарна метрика в мережу А через маршрутизатор 3: 47019776
- сумарна метрика в мережу А через маршрутизатор 1: 47019776

Оскільки шлях через маршрутизатор 4 володіє найкращою метрикою, маршрутизатор 3 встановлює цей маршрут в таблиці перенаправлень і використовує метрику 20281600 в якості можливого відстані в мережу А. Потім маршрутизатор 3 обчислює оголошене відстань в мережу А для шляхів, що йдуть через маршрутизатори 2 і 1: 47019776 відповідає шляху через маршрутизатор 2, а 47019776 відповідає шляху через маршрутизатор 1. Оскільки значення цих метрик перевищує значення можливої відстані, маршрутизатор 3 не призначила жоден з цих маршрутів в якості можливих наступників, провідних в мережу А.

Припустимо, що зв'язок між маршрутизаторами 3 і 4 була порушена. Маршрутизатор 3 запрошувати альтернативний маршрут до мережі А у всіх сусідніх вузлів. Маршрутизатор 2 отримує запит і, оскільки цей запит надходить від наступника, виконує пошук всіх записів про можливих наступників в таблиці топології. Єдина така запис в таблиці топології належить маршрутизатора 1 (оголошене відстань цьому записі дорівнює значенню останньої відомої метриці через маршрутизатор 3). Оскільки оголошене відстань через маршрутизатор 1 більше ніж останнє відоме можливу відстань, маршрутизатор 2 позначає цей маршрут як недосяжний і починає опитувати своїх сусідів (в цьому випадку опитується тільки маршрутизатор 1) на предмет шляху до мережі А.

Маршрутизатор 3 відправляє маршрутизатора 1 запит щодо мережі А. Маршрутизатор 1 виконує пошук в своїй таблиці топології і виявляє, що тільки другий єдиний маршрут до мережі А проходить через маршрутизатор 2 з оголошеним відстанню, рівним останнім відомим можливого віддалі через маршрутизатор 3. Оскільки оголошене відстань через маршрутизатор 2 знову-таки перевищує останнє відоме можливу відстань, цей маршрут не є

можливим наступником. Маршрутизатор 1 позначає цей маршрут як недосяжний і запитує в єдиного, що залишився сусіда (маршрутизатор 2) шлях до мережі А.

Намагаючись знайти маршрут в мережу А маршрутизатор 3 опитав усіх своїх сусідів. В свою чергу маршрутизатори 1 і 2 позначили маршрут як недосяжний і опитали своїх сусідів, щоб знайти шлях в мережу А. При отриманні запиту від маршрутизатора 1 маршрутизатор 2 виконує пошук по своїй таблиці топології, в ході якого виявляє, що пункт призначення позначений як недосяжний. Маршрутизатор 2 відповідає маршрутизатору 1, що мережа А недоступна. Коли маршрутизатор 1 отримує запит від маршрутизатора 2, він також посилає назад відповідь про те, що мережа А недоступна. Оскільки маршрутизатори 1 і 2 встановили, що мережа А чи не доступна, вони відповідають на початковий запит маршрутизатора 3. Збіжність закінчена, і всі маршрути переходять в пасивний стан.

Розщеплений горизонт і зворотня заборона

В попередньому прикладі ми обговорили, що функція "розщеплений горизонт" була вимкнена (ми хотіли показати, яким чином EIGRP використовує можливу відстань і оголошену відстань, щоб визначити, чи є маршрут петлею). Проте в деяких випадках EIGRP використовує розщеплений горизонт, щоб запобігти виникненню петель маршрутизації. Перш ніж почати розглядати особливості використання розщепленого горизонту, необхідно пояснити, що це таке і як це працює. Правило розщепленого горизонту виглядає наступним чином:

- ніколи не оголошувати маршрут через інтерфейс, за допомогою якого маршрутизатор дізнався про маршрут.

Наведемо приклад. На рис. 4а показано, що якщо маршрутизатор 1 підключений до маршрутизаторів 2 і 3 через єдиний багатоточковий інтерфейс (наприклад, Frame Relay) і якщо при цьому маршрутизатор 1 дізнається про мережу А через маршрутизатор 2, то він не буде використовувати той же самий інтерфейс, щоб оголосити маршрутизатору 3 маршрут в мережу А. маршрутизатор 1 припускає, що маршрутизатор 3 отримає інформацію про мережу А безпосередньо від маршрутизатора 2.

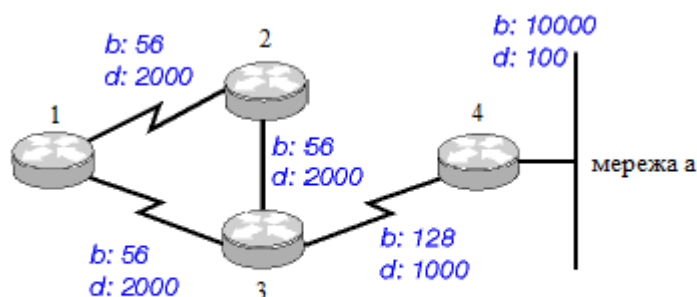


рис.4а

Зворотний заборона - це ще один спосіб запобігти появі петель. Правило зворотного заборони виглядає наступним чином:

- якщо інформація про маршрут надійшла через який-небудь інтерфейс, то цей маршрут необхідно оголосити недосяжним через той же самий інтерфейс.

Припустимо, що на маршрутизаторах, показаних на малюнку 4а, включена функція зворотного заборони .. Коли маршрутизатор 1 дізнається від маршрутизатора 2 про мережу А, він оголошує маршрутизаторам 2 і 3, що мережа А недосяжна через його канал. У разі якщо маршрутизатор 3 вказує якоїсь шлях в мережу А через маршрутизатор 1, він повинен видалити його, оскільки було оголошено про те, що мережа А недосяжна через цей шлях. EIGRP використовує два цих правила, щоб запобігти виникненню петель маршрутизації.

EIGRP використовує розщеплений горизонт і оголошує маршрут недосяжним в наступних випадках:

- якщо два маршрутизатора працюють в режимі ініціалізації (тобто виконують первинний обмін таблицями топології)
- якщо оголошується про зміну в таблиці топології
- при відправці запиту

Тепер розглянемо кожен з цих випадків.

режим ініціалізації

Коли два маршрутизатора вперше стають "сусідами", вони, перебуваючи в режимі ініціалізації, обмінюються таблицями топології. Кожен запис у таблиці, яку маршрутизатор прийняв в режимі ініціалізації, повторно оголошується (приймаючим маршрутизатором) новому сусідові, при цьому до цього запису застосовується максимальна метрика (це так званий "заборонений маршрут" - poison route, букв. Отруєний маршрут).

Зміни в таблиці топології

На рис. 5 в маршрутизаторі 1 використовується дисперсія для балансування трафіку, призначеного для мережі А на відрізку між двома послідовними каналами (канал в 56 Кбіт між маршрутизаторами 2 і 4 і канал в 128 Кбіт між маршрутизаторами 3 і 4). (Докладно про дисперсії см. Балансування навантаження, розділ, в якому обговорюється дисперсія).

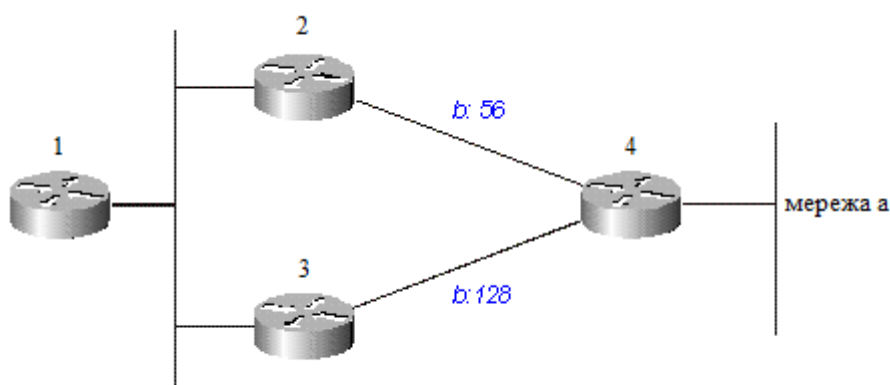


Рис.5

Маршрутизатор 2 розглядає шлях через маршрутизатор 3 в якості можливого наступника. Якщо канал між маршрутизаторами 2 і 4 буде порушений, то маршрутизатор 2 виконає збіжність по маршруту, що проходить через маршрутизатор 3. Оскільки правило розщепленого горизонту свідчить, що не можна оголошувати маршрут через інтерфейс, за допомогою якого були отримані відомості про цей маршрут, маршрутизатор 2 зазвичай не передає оновлення. Однак це залишає маршрутизатор 1 з неприпустимою записом в таблиці

топології. Коли маршрутизатор змінює свою таблицю топології таким чином, що при цьому відбувається зміна інтерфейсу, через який маршрутизатор з'єднується з мережею, то він відключає розщеплений горизонт, а виправлення міняють всі маршрути, які виходять із інтерфейсів, в зворотному напрямку. В цьому випадку маршрутизатор 2 відключає розщеплений горизонт для даного маршруту і оголошує мережу А недоступною. При отриманні цього оголошення Маршрутизатор 1 видаляє зі своєї таблиці маршрутизації свій маршрут до мережі А, що йде через маршрутизатор 2.

Запити.

Запити призводять до виникнення розщепленого горизонту тільки тоді, коли маршрутизатор отримує запит або оновлення від наступника, якого він використовує для одержувача в запиті. Розглянемо мережу, показану на рис. 6.

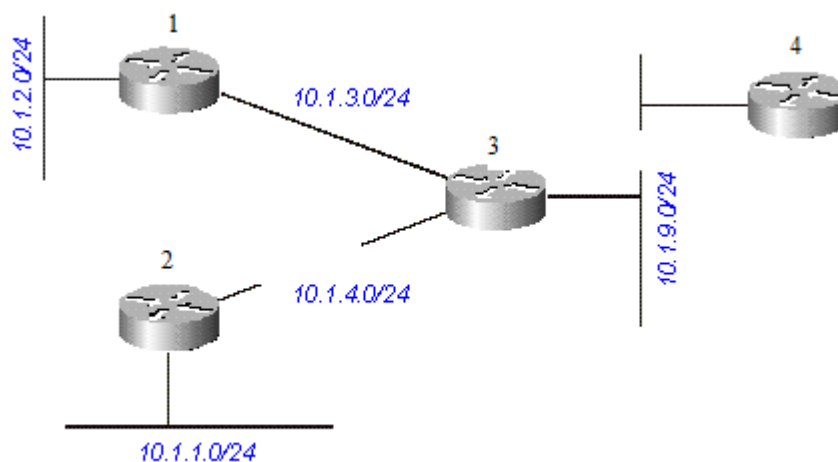


Рис.6

Маршрутизатор 3 отримує від маршрутизатора 4 запит про мережу 10.1.2.0/24 (якої він досягає через маршрутизатор 1). Якщо у маршрутизатора 3 відсутній наступник для даного пункту призначення (з причини перемикання лінії або через інший тимчасової ситуації в мережі), цей маршрутизатор відправляє запит кожному зі своїх сусідів (в даному випадку це маршрутизатори 1, 2 і 4). Однак якщо маршрутизатор 3 отримає від маршрутизатора 1 запит або оновлення (наприклад, зміна метрики) для мережі 10.1.2.0/24, той він не відправлятиме запит назад маршрутизатора 1, оскільки останній є його наступником на маршруті до цієї мережі. Замість цього він відправить запити тільки маршрутизаторам 2 і 4.

Затримка в активних маршрутах

При певних обставинах може пройти велику кількість часу, перш ніж на запит буде отримана відповідь. "Мовчання" може бути настільки тривалим, що ініціював запит маршрутизатор може припинити очікування і скинути з'єднання з "мовчазним" маршрутизатором, при цьому відбувається фактичний перезапуск сеансу зв'язку з сусіднім вузлом. Така подія називається "затор на активному маршруті" (stuck in active - SIA). Найбільш прості випадки SIA виникають, якщо потрібно занадто багато часу для досягнення запитом іншого кінця мережі, а також для зворотного проходження відгуку. Наприклад, на рис. 7 маршрутизатор 1 записує велику число маршрутів SIA від маршрутизатора 2.

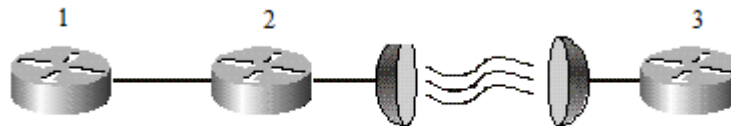


Рис.7

Після проведення перевірок виявляється, що проблема зводиться до наявності затримки в супутниковому каналі між маршрутизаторами 2 і 3. Існують два можливих рішення проблеми цього типу. Перше - збільшити інтервал, протягом якого маршрутизатор буде очікувати відповіді на запит, перш ніж оголосити маршрут маршрутом SIA. Цю настройку можна змінити за допомогою команди `timers active-time`.

Однак оптимальне рішення проблеми полягає в тому, щоб перекомпонувати мережу таким чином, щоб зменшити діапазон запитів (щоб по супутниковому каналу передавалося мінімальна кількість запитів). Це питання розглядається в розділі Діапазон запитів. Проблема діапазону запитів не є поширеною причиною виникнення SIA маршрутів. Значно частіше деякі маршрутизатори в мережі не можуть відповісти на запит по одній з наступних причин:

- маршрутизатор сильно завантажений роботою і не може відповісти на запит (зазвичай це пов'язано з сильною завантаженням ЦП)
- у маршрутизатора є труднощі при роботі з пам'яттю, і він не може виділити пам'ять для обробки запиту або для формування пакета відгуку
- канал між двома маршрутизаторами не забезпечує достатньої якості (це значить, що по такому каналу і раніше можна передавати пакети в кількості, достатній для підтримання зв'язку "сусід-сусід", однак при цьому деякі запити або відгуки все-таки губляться при пересиланні від одного маршрутизатора до іншого)
- односпрямовані канали (тобто канали, в яких через збій трафік може передаватися тільки в одному напрямку)

Усунення SIA маршрутів

Процес усунення маршрутів SIA зазвичай складається з трьох етапів:

1. Спочатку необхідно встановити маршрути, про які система зразу в раз повідомляє як про маршрутах SIA.
2. Потім необхідно встановити маршрутизатор, який зразу в раз не відповідає на запити на даних маршрутах.
3. Потім необхідно визначити причину, по якій маршрутизатор не отримує запитів або не відповідає на них.

Перший крок виконується відносно просто. Якщо у вашій системі ведеться облік консольних повідомлень, то при швидкому перегляді журналу можна визначити, які маршрути найчастіше позначаються як маршрути SIA. Другий крок більш складний. Для збору цієї інформації необхідно використовувати команду `show ip eigrp topology active`:

Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,
r - Reply status

A 10.2.4.0/24, 0 successors, FD is 512640000, Q
1 replies, active 00:00:01, query-origin: Local origin
via 10.1.2.2 (Infinity/Infinity), Serial1
1 replies, active 00:00:01, query-origin: Local origin
via 10.1.3.2 (Infinity/Infinity), r, Serial3
Remaining replies:
via 10.1.1.2, r, Serial0

Всі сусідні вузли, для яких відображена буква R, ще тільки повинні відповісти (активний таймер показує час, протягом якого маршрут є активним). Зверніть увагу на те, що сусідні вузли можуть бути відсутні в розділі "Remaining replies" (очікувані відгуки). Ці вузли можуть з'явитися серед інших RDB. Особливу увагу необхідно приділяти маршрутам, які були активні протягом певного часу (зазвичай 2-3 хвилини) і відповідь від яких тільки очікується. Виконайте цю команду кілька разів, щоб з'ясувати, які сусіди не відповідають на запити або які інтерфейси мають багато пропущених запитів. Перевірте цей сусідній вузол, щоб дізнатися, чи знаходиться він постійно в режимі очікування відгуків від своїх сусідів. Повторюйте цю процедуру доти, поки не знайдете маршрутизатор, який з раз у раз не відповідає на запити. Можна також пошукати неполадки, що відносяться до з'єднання з цим сусідом, до пам'яті, а також неполадки, пов'язані із завантаженням ЦП та іншими проблемами, наявними у даного сусіда.

Якщо ви стикаєтеся з ситуацією, коли на вашу думку причиною неполадок є діапазон запитів, кращим рішенням проблеми буде зменшити діапазон запиту (а не збільшувати значення таймера SIA).

Перерозподіл

Даний розділ містить опис різних випадків перерозподілу. Будь ласка, пам'ятайте про те, що в прикладах, розглянутих нижче, вказаний мінімум, який необхідний для настройки перерозподілу. Перерозподіл пов'язано з потенційними проблемами, такими як: неоптимальна маршрутизація, освіта петель або повільна збіжність. Щоб уникнути подібних проблем, зверніться до розділу "Як уникнути проблем при перерозподілі" в документі "Перерозподіл протоколів маршрутизації".

Завдання на лабораторну роботу.

- підключення та налаштування з'єднань WAN;
- настройка EIGRP для оголошення конкретних мереж;
- дослідження конвергенції мережі з вікна інтерфейсу командного рядка при відключенні і повторному включенні інтерфейсу;
- перевірка пакетів EIGRP в режимі моделювання процесу конвергенції мережі.

Вихідні дані

Вам пропонується топологія з уже налаштованими вузлами HQ, Branch1, Branch2 і Branch3. В неї додано новий частково налаштований маршрутизатор (New_Branch). Маршрутизатор New_Branch потрібно підключити до HQ і Branch1, закінчити настройку нового маршрутизатора і перевірити конвергенцію мережі.

Крок 1. Підключення та настройка з'єднання WAN для маршрутизатора New_Branch

- a. Підключіть інтерфейс S0 / 0/0 маршрутизатора New_Branch до інтерфейсу S0 / 1/1 маршрутизатора HQ (DCE)
- b. Підключіть інтерфейс S0 / 0/1 маршрутизатора New_Branch до інтерфейсу S0 / 1/1 маршрутизатора Branch1 (DCE)
- c. Налаштуйте інтерфейс S0 / 0/0, використовуючи IP-адресу 172.16.3.218/30
- d. Налаштуйте інтерфейс S0 / 0/1, використовуючи IP-адресу 172.16.3.221/30

Крок 2. Налаштування EIGRP для оголошення конкретної мережі на маршрутизаторі New_Branch

- Налаштуйте на маршрутизаторі New_Branch протокол EIGRP і виберіть номер анонімної системи 3
- Оголосіть прямо підключення мережі

Крок 3. Спостереження за конвергенцією мережі в режимі реального часу

- У вікні CLI для маршрутизатора New_Branch можна спостерігати за конвергенцією в режимі реального часу. В процесі конвергенції буде видно, як EIGRP створює суміжності.
- Після завершення конвергенції відключіть інтерфейс S0 / 0/0 маршрутизатора New_Branch.
- Простежте за змінами мережі
- Знову включите інтерфейс S0/0/0

Крок 4. Спостереження за конвергенцією мережі в режимі моделювання

- Перейдіть в режим моделювання
- Налаштуйте фільтри списку подій так, щоб відображались тільки пакети EIGRP
- Відкрийте вікно CLI для маршрутизатора New_Branch
- Вимкніть інтерфейс S0/0/1
- Натисніть кнопку Auto Capture / Play, щоб почати моделювання
- Знову відкрийте вікно CLI і подивіться на результати
- Почекайте трохи і ще раз натисніть кнопку Auto Capture / Play, щоб призупинити моделювання
- Перевірте деякі пакети в списку Event List

Крок 5. Перевірка результату включення інтерфейсу

- Повторно запустіть моделювання, натиснувши кнопку Auto Capture / Play.
- Увімкніть інтерфейс S0 / 0/1 і простежте за ходом конвергенції за допомогою інтерфейсу командного рядка, списку подій і топології
- Зупиніть моделювання

Контрольні питання

1. Що відображалось в інтерфейсі командного рядка при конвергенції протоколу EIGRP?
2. Що сталося з пакетами EIGRP в новій локальній мережі при відключенні каналу WAN, яка зв'язує HQ і New_Branch?