МЕХАНІЗМИ КРУГОВОГО ОБСЛУГОВУВАННЯ ЧЕРГ

Содержание

1. Зважений алгоритм кругового обслуговування WRR і модифікований алгоритм зваженого кругового обслуговування MWRR

2. Модифікований алгоритм кругового обслуговування з дефіцитом MDRR

3. Рекомендації щодо вибору стратегії черг

# 1. Зважений алгоритм кругового обслуговування WRR і модифікований алгоритм зваженого кругового обслуговування MWRR

Механізм обслуговування черг має бути максимально наближений до ідеальної моделі планувальника GPS. Механізм кругового обслуговування (Round Robin), що обробляє за цикл один пакет (замість нескінченно малого обсягу даних) з кожної непорожньої черги, є найпростішою реалізацією схеми GPS. В механізмі FQ на основі обчислення порядкового номера пакета імітується робота GPS-сервера, який обслуговує в окремий момент часу 1 байт даних.

Найточніша імітація планувальника GPS досягається в механізмі кругового обслуговування у випадку рівності розміру всіх пакетів. Зважений алгоритм кругового обслуговування (Weighted Round Robin, WRR) є розширенням планувальника кругового обслуговування, відповідно до якого кожному потокові трафіка призначається своя вага. Алгоритм WRR обробляє потік трафіка пропорційно до його ваги. Розглянутий раніше алгоритм CQ можна також розглядати як WRR з додаванням пріоритетної черги.

Найкраще WRR-планувальник узгоджується з механізмом комутації ATM, відповідно до якого пакет подається у вигляді чарунок, а алгоритм WRR використовується для обробки черг, які заповнюються чарунками. По суті, WRR є механізмом кругового обслуговування на основі чарунок, де вага потоку визначає кількість чарунок, які обслуговуються за один цикл. Отже, кожній черзі надається частина смуги пропускання інтерфейса відповідно до ваги потоку трафіка, яка не залежить від розміру пакета.

З метою підтримки пакетів змінного розміру було створено модифікований зважений алгоритм кругового обслуговування (MWRR), що використовує лічильник дефіциту, асоційований з кожною WRR-чергою.

Перед початком обслуговування черг значення відповідних їм лічильників дефіциту встановлюється рівним вазі черги. Обробка пакета починається і продовжується тільки в тому випадку, якщо значення лічильника дефіциту більше нуля. Після обслуговування пакета, що складається з чарунок, значення лічильника дефіциту зменшується на . Коли значення лічильника дефіциту стане менше нуля або рівним нулеві, планувальник переходить до обслуговування наступної черги. У кожному новому циклі значення лічильника дефіциту черги збільшується на вагу черги.



Ефективна ширина смуги пропускання черги - прямо пропорційна її вазі і розраховується за формулою



, (1)



де - ширина смуги пропускання інтерфейса; - загальна кількість активних черг.



Розглянемо роботу алгоритму MWRR на прикладі. Припустимо, що алгоритм MWRR використовується для обслуговування трьох черг (черги з номерами 0, 1,2), вага кожної з яких 2, 3 і 4 відповідно (рис.1).

Кожна черга складається з дев'яти чарунок (рис.1). Чарунки, що є частинами одного пакета, мають однаковий відтінок сірого кольору. Приміром, у черзі 2 знаходяться три пакети, що складаються з двох, трьох і чотирьох чарунок, відповідно.



Рисунок 1 - WRR-черги і відповідні значення їхніх лічильників дефіциту до початку обслуговування

Першою обслуговується черга 0. При ініціалізації лічильника дефіциту присвоюється значення 2, що дорівнює вазі черги. На початку черги 0 знаходиться пакет з чотирьох чарунок. Отже, після обслуговування цього пакета значення лічильника дефіциту дорівнюватиме 2 - 4=-2. Оскільки значення лічильника дефіциту від’ємне, черга 0 не може бути обслугована доти, поки значення лічильника дефіциту не стане більше нуля (рис.2).

Наступною обслуговується черга 1. При ініціалізації лічильника дефіциту йому присвоюється значення 3. У результаті обслуговування пакета з трьох чарунок, що знаходиться на початку черги 1, значення лічильника дефіциту стає рівним 3 - 3=0. Оскільки значення лічильника дефіциту дорівнює нулеві, планувальник алгоритму MWRR приступає до обробки наступної черги (рис.2).

Останньою обслуговується черга 2. При ініціалізації лічильника дефіциту йому присвоюється значення 4. У результаті обслуговування пакета з двох чарунок, що знаходиться на початку черги 2, значення лічильника дефіциту стає рівним 4 - 2=2. Оскільки значення лічильника дефіциту більше нуля, MWRR-планувальник обробляє наступний пакет з трьох чарунок. Після обслуговування цього пакета значення лічильника дефіциту черги 2 стає рівним 2 - 3 =-1, як показано на рис.2. Планувальник MWRR приступає до другого циклу обслуговування черги 0. Значення лічильника дефіциту черги 0, встановлене на першому циклі, дорівнює - 2. У результаті збільшення значення лічильника дефіциту на вагу черги воно стає рівним - 2+2=0. Оскільки значення лічильника дефіциту усе ще не більше нуля, планувальник алгоритму MWRR приступає до обробки наступної черги (рис.3).



Рисунок 2 - Стан черг алгоритму MWWR після першого циклу обслуговування



Рисунок 3 - Стан черги алгоритму MWWR після другого циклу обслуговування

Після першого циклу обслуговування значення лічильника дефіциту черги 1 дорівнювало нулеві. В другому циклі обслуговування значення лічильника дефіциту збільшується на вагу черги і стає рівним 0+3=3.

У результаті обслуговування пакета з чотирьох чарунок, що знаходиться на початку черги 1, значення лічильника дефіциту стає рівним 3 - 4 = - 1, як показано на рис.3.

У другому циклі обслуговування значення лічильника дефіциту черги 2 збільшується на її вагу і стає рівним - 1+4=3. У результаті обслуговування пакета з чотирьох чарунок, що знаходиться на початку черги 2, значення лічильника дефіциту стає рівним 3 - 4 = - 1. Оскільки тепер черга 2 стає порожньою, значення її лічильника дефіциту скидається в нуль (рис.3).



Рисунок 4 - Стан черги алгоритму MWWR після третього циклу обслуговування

Планувальник MWRR приступає до третього циклу обслуговування черги 0. Значення лічильника дефіциту збільшується 0+2=2. У результаті обслуговування пакета з двох чарунок, що знаходиться на початку черги 0, значення лічильника дефіциту стає рівним 2 - 2=0. MWRR-планувальник припиняє обробку черги 0 і переходить до черги 1 (рис.4).

Нове значення лічильника дефіциту черги 1 дорівнює - 1+3=2.

У результаті обслуговування пакета з двох чарунок, який знаходиться на початку черги 1, значення лічильника дефіциту стає рівним 2 - 2 = 0. Оскільки тепер черга 1 стає порожньою, планувальник алгоритму MWRR приступає до обробки черги 0, як показано на рис.4.

У четвертому циклі обслуговування черги 0 значення її лічильника дефіциту стає рівним 2. У результаті обслуговування пакета з трьох чарунок, що знаходиться на початку черги 0, значення лічильника дефіциту стає рівним - 1. Оскільки тепер черга 0 стає порожньою, значення її лічильника дефіциту скидається в нуль.

# 2. Модифікований алгоритм кругового обслуговування з дефіцитом MDRR

У маршрутизаторах Cisco серії 12000 використовується модифікований алгоритм кругового обслуговування з дефіцитом (Modified Deficit Round Robin, MDRR). Відповідно до DRR-планувальника кожна черга характеризується пов'язаним з нею квантовим значенням (quantum value) - середньою кількістю байтів, що обслуговуються протягом кожного циклу, - та лічильником дефіциту, початкове значення якого встановлюється рівним квантовому значенню. Кожна непорожня черга обробляється за круговим принципом. Сума розмірів пакетів, які обслуговуються за один цикл, приблизно дорівнює квантовому значенню черги. Обробка пакетів черги реалізується до тих пір, доки значення лічильника дефіциту залишається більше нуля. У результаті обслуговування кожного пакета значення лічильника дефіциту черги зменшується на величину, яка дорівнює розміру пакета в байтах. Коли значення лічильника дефіциту стає менше нуля або рівним нулеві, DRR-планувальник переходить до обробки наступної черги. Слід зазначити, що в кожному новому циклі значення лічильника дефіциту збільшується на величину квантового значення. Як видно, квантове значення в DRR застосовується з тією ж метою, що і вага в MWRR.

Після обслуговування черги значення її лічильника дефіциту є розміром дебету, який залежить від того, чи було перевищено квантове значення черги кількістю байтів, які оброблено протягом попереднього циклу. Обсяг даних, що будуть оброблені в наступному циклі обслуговування черги, розраховується як різниця квантового значення і лічильника дефіциту.

З метою підвищення ефективності механізму DRR квантове значення має дорівнювати розмірові в байтах пакета максимального обсягу. Це дає гарантію, що DRR-планувальник завжди обслуговуватиме щонайменше один пакет з кожної черги.

Описаний вище узагальнений алгоритм DRR модифікується за допомогою додавання так званої черги з малою затримкою (low-latency queue). Відповідно до алгоритму MDRR усі черги, за винятком черги з малою затримкою, обслуговуються за круговим принципом. Черга з малою затримкою може обслуговуватися в двох режимах: режимі строгого пріоритету і режимі чергування пріоритетів.

У режимі строгого пріоритету (strict priority mode) черга з малою затримкою обслуговується за наявності в ній хоча б одного пакета, що обумовлює мінімально можливу затримку для класу трафіка, який оброблюється за допомогою цієї черги. У той же час слід зазначити, що високопріоритетна черга з малою затримкою здатна на тривалий період часу зайняти всі 100% смуги пропускання інтерфейса і тим самим "придушити" інші MDRR-черги.

У режимі чергування пріоритетів (alternate priority mode) черга з малою затримкою обробляється в проміжках між обробкою інших черг. На додаток до черги з малою затримкою алгоритм MDRR підтримує ще сім черг. Припустимо, що черга з малою затримкою має номер 0. Тоді відповідно до режиму пріоритету, що чергується, черги механізму MDRR обслуговуються в такому порядку: 0, 1, 0, 2, 0, 3, 0, 4, 0, 5, 0, 6, 0,Режим чергування пріоритетів дозволяє зменшити максимальну затримку обслуговування черги 0 із суми квантових значень (що було б у випадку традиційного механізму кругового обслуговування) до максимального квантового значення серед всіх інших черг.

Розглянемо роботу алгоритму MDRR на прикладі. Припустимо, що алгоритм MDRR використовується для обслуговування трьох черг - черги 2, черги 1 і черги 0, - вага яких дорівнює 1, 2 і 1, відповідно (табл.1). Черга 2 є чергою з малою затримкою, яка обслуговується в режимі пріоритету, що чергується. Схематичне зображення черг та відповідні до них значення лічильників дефіциту подані на рис.5.

Таблиця 1 - Вага і квантові значення черг, що обслуговуються за допомогою алгоритму MDRR

|  |  |  |
| --- | --- | --- |
| Номер черги | Вага | Квантове значення = вага MTU (MTU = 1500 байт) |
| Черга 2 | 1 | 1500 |
| Черга 1 | 2 | 3000 |
| Черга 0 | 1 | 1500 |

Коли механізм MDRR сконфігуровано на вихідному інтерфейсі маршрутизатора, як максимальний розмір одиниці передачі інформації (Maximum Transmission Unit, MTU) використовується відповідне значення MTU цього інтерфейса.

Першою обслуговується черга 2. При ініціалізації лічильника дефіциту йому присвоюється значення 1500, яке дорівнює квантовому значенню черги. Черга 2 обслуговується доти, доки відповідне їй значення лічильника дефіциту залишається більше нуля. Після обслуговування пакета значення лічильника дефіциту черги 2 зменшується на величину, яка дорівнює розмірові пакета в байтах. Оскільки поточне значення лічильника дефіциту (1500) більше нуля, планувальник MDRR обслуговує 500-байтовий пакет, що знаходиться на початку черги 2. Нове значення лічильника дефіциту черги 2 стає рівним 1500 - 500=1000. Оскільки нове значення лічильника дефіциту усе ще більше нуля, планувальник MDRR починає обслуговування наступного у черзі пакета. У результаті передачі 1500-байтового пакета значення лічильника дефіциту зменшується до 1000 - 1500=-500, після чого MDRR-планувальник переходить до обслуговування наступної черги. Стан черг і відповідних їм значень лічильників дефіциту після закінчення обслуговування черги 2 схематично подано на рис.6.



Рисунок 5 - MDRR - черги і відповідні значення лічильників дефіциту до початку обслуговування черги 2



Рисунок 6 - Стан черг і відповідних їм значень лічильників дефіциту після першого циклу обслуговування

Оскільки обслуговування черги 2 здійснюється в режимі чергування пріоритетів MDRR-планувальник переходить до обробки іншої черги, номер якої вибирається відповідно до кругового принципу. Припустимо, що підійшла черга обслуговування черги 0. При ініціалізації лічильника дефіциту йому присвоюється значення 1500, яке дорівнює квантовому значенню черги 0. У результаті обслуговування 1500-байтового пакета значення лічильника дефіциту зменшується до 1500 - 1500 = 0. Відповідно до алгоритму MDRR-планувальник завершує обслуговування черги 0. Стан черг і відповідних їм значень лічильників дефіциту після закінчення обслуговування черги 0 схематично подане на рис.

Оскільки режим чергування пріоритетів припускає постійне переключення MDRR-планувальника між чергою з малою затримкою та іншими чергами, наступною обслуговується черга 2. Нове значення лічильника дефіциту стає рівним - 500+1500=1000. Оскільки значення лічильника дефіциту більше нуля, MDRR-планувальник приступає до обслуговування наступного пакета черги 2.

У результаті передачі 500-байтового пакета значення лічильника дефіциту зменшується до 500. Незважаючи на те що MDRR-планувальник міг би обслужити як мінімум ще один пакет черги 2, це не є можливим, тому що черга порожня. Тому значення лічильника дефіциту черги 2 скидається в нуль. Оскільки порожня черга не обробляється MDRR-планувальником, значення лічильника дефіциту черги 2 залишається рівним нулеві до постановки в цю чергу наступного пакета. Стан черг і відповідних їм значень лічильників дефіциту після закінчення обслуговування черги 2 схематично подано на рис.8.



Рисунок 7 - Стан черг і відповідних їм значень лічильників дефіциту після першого циклу обслуговування черги 0



Рисунок 8 - Стан черг і відповідних їм значень лічильників дефіциту після другого циклу обслуговування черги 2

Наступною обслуговується черга 1, значення лічильника дефіциту якої встановлюється рівним 3000. MDRR-планувальник передає три пакети з черги 1, у результаті чого значення її лічильника дефіциту зменшується до 3000 - 1500 - 500 - 1500=-500. Стан черг і відповідних їм значень лічильників дефіциту після закінчення обслуговування черги 1 схематично подано на рис.9.

У результаті обслуговування двох пакетів черги 0 значення її лічильника дефіциту стає рівним 1500-500-1500=-500. Оскільки черга 0 стає порожньою, відповідне значення лічильника дефіциту скидається в нуль. Стан черг і відповідних їм значень лічильників дефіциту після закінчення обслуговування черги 0 схематично подано на рис.10.



Рисунок 9 - Стан черг і відповідних їм значень лічильників дефіциту після першого циклу обслуговування черги 1



Рисунок 10 - Стан черг і відповідних їм значень лічильників дефіциту після другого циклу обслуговування черги 0

На наступному кроці MDRR-планувальник обробляє останній пакет черги 1. Оскільки черга 1 стає порожньою, відповідне їй значення лічильника дефіциту скидається в нуль.

Модифікований алгоритм кругового обслуговування з дефіцитом MDRR реалізований в маршрутизаторах Cisco серії 12000. Механізм MDRR можна сконфігурувати для обслуговування або черги вихідного інтерфейса (вихідний блок маршрутизатора - ТХ), або черги вхідного інтерфейса (вхідний блок маршрутизатора - RX) при функціонуванні в режимі подачі черг комутаційної матриці маршрутизатора на вихідний інтерфейс.

механізм кругове обслуговування черга

# 3. Рекомендації щодо вибору стратегії черг

Вибір схеми організації черг є одним з важливих кроків у проектуванні мережі і містить у собі такі етапи.

1. Визначити, чи знаходиться в стані перевантаження канал зв'язку з глобальною мережею. У тому випадку, якщо перевантаження немає, то немає й необхідності в додаткових діях. Але якщо спостерігаються періоди значного завантаження каналу, то виникає потреба в реалізації механізмів розподілу ресурсів.

2. На наступному етапі варто з'ясувати, чи потрібний строгий контроль над розподілом трафіка відповідно до пріоритетів і чи є прийнятними автоматичні настройки маршрутизатора. Слід зазначити, що коректне настроювання черг зовсім непроста задача. Для того щоб виконати її з найбільшою ефективністю, адміністратор мережі має проаналізувати типи трафіка, що проходять через даний інтерфейс, з'ясувати, як розрізняти різні потоки трафіку, і визначити відносні пріоритети. Крім того, варто пам'ятати про те, що характер трафіка з часом міняється. А це вимагає регулярного виконання такого аналізу.

1. Стратегія черг базується на проведеному аналізі трафіка і визначенні відносних його пріоритетів (обговорюється на кроці 2).
2. На останньому кроці потрібно визначити, наскільки виділений трафік чутливий до затримок.

На рис.11 подано алгоритм дій адміністратора мережі з метою вибору типу черги.



Рисунок 11 - Алгоритм вибору типу черги.