МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Запорізький національний технічний університет

МЕТОДИЧНІ ВКАЗІВКИ

до проходження професійного тесту «Загальні поняття мережної взаємодії. Технології та стандарти локальних мереж»
з дисципліни "Комп'ютерні мережі" для бакалаврів спеціальності 123 "Комп'ютерна інженерія", усіх форм навчання.
Частина 2

2017

Методичні вказівки до проходження професійного тесту «Загальні поняття мережної взаємодії. Технології та стандарти локальних мереж» з дисципліни "Комп’ютерні мережі" для бакалаврів спеціальності 123 "Комп'ютерна інженерія", усіх форм навчання. Частина 2 / Укл. Г.Г Киричек. – Запоріжжя: ЗНТУ, 2017. – 38 с.

Укладач:

Г.Г. Киричек, доцент, к.т.н.

Рецензент:

Р.К. Кудерметов, доцент, к.т.н.

Відповідальний за випуск:

Г.Г. Киричек, доцент, к.т.н.

Затверджено
на засіданні кафедри КСМ
Протокол № 2 від 13.09.2017

Затверджено
на засіданні НМК КНТ
Протокол № 2 від 28 .09.2017

ЗМІСТ

1 МЕРЕЖНИЙ РІВЕНЬ. АДРЕСАЦІЯ. МАРШРУТИЗАЦІЯ

Мета: отримати базові навички проведення розрахунків для отримання бажаного розміру MTU при передачі пакетів між мережами, які побудовані за різними технологіями канального рівня. Отримати базові навички налаштування мережних з'єднань з використанням IP-адресації протоколу версії IPv4.

1.1 Максимальна одиниця передачі MTU

MTU (Maximum Transmission Unit - максимальна одиниця передачі). MTU - максимальний розмір пакета даних, який може бути переданий за один фізичний кадр по стеку протоколів TCP/IP. При установці нового з'єднання два віддалених комп'ютера повинні узгодити між собою розмір кадру. Окрім того слідуючи до місця призначення, пакет долає цілий ряд проміжних серверів і маршрутизаторів, настройки MTU яких можуть бути абсолютно різними. Тому занадто великий пакет в мережі фрагментується і заповнюється «повітрям», «баластом», що негативно позначається на ефективності зв'язку. Якщо провайдер має установки MTU = 576, а у вас в Windows задано MTU = 1500, то кожний пакет розбивається на три по 576 байт: 576 + 576 + 576 = 1728 - тобто, 228 байт баласту додаються до кожного пакету. Але навіть якщо провайдер теж має MTU = 1500, то при зв'язку з віддаленим сервером цілком може бути задіяний маршрутизатор з меншим значенням MTU і пакети знову-таки фрагментуватимуться, сповільнюючи передачу даних. Цю ситуацію рятує включена в Windows, за замовчуванням, функція автоматичного визначення MTU - «PMTU Discovery» або «MTU Auto Discovery». Але процедура обчислення MTU для кожного з'єднання вимагає багато часу, що може затримувати роботу при передачі невеликих файлів. Окрім того, у разі неузгодження ваших параметрів з параметрами провайдера, ця функція навряд чи вам допоможе. Звичайно, існують загальноприйняті стандарти для даного параметра, так, наприклад, для Ethernet MTU = 1500 байт, для SLIP - 1006, для

PPPoE -1492, для PPP, тобто модемного зв'язку з Інтернетом - 576. Але на ділі ваш провайдер може вибрати відмінне від цих значень число, виходячи з того, що йому це з якихось причин зручніше. Ми ж в результаті або не завантажуємо свій канал зв'язку повністю, відправляючи кадри меншого розміру, ніж це дозволяє провайдер і сервери, або навпаки, наші установки перевищують необхідне значення, і великі пакети йдуть фрагментованими і це ще більше знижує можливості лінії зв'язку.

Кожен пакет даних в дійсності складається з декількох сегментів - заголовка і фактичних даних.

Формат стандартного заголовку IP-пакету (на прикладі протоколу IPv4) наведено на рисунку 1.1.

Рисунок 1.1 – Структура заголовку IP-пакету

Поле **номер версії** займає 4 біта і ідентифікує версію протоколу IP. Зараз використовується версія 4 (IPv4), але часто зустрічається і нова версія (IPv6).

Довжина заголовка IP-пакета займає 4 біта і вимірюється в 32-бітових словах. Зазвичай заголовок має довжину в 20 байт (п'ять 32-бітових слів). Найбільша довжина заголовка складає 60 байт.

Тип сервісу (Type of Service, ToS) - байт диференційованого обслуговування, або DS-байт. Зберігає ознаки, які відображають вимоги до якості обслуговування пакета. Перші три біти - значення пріоритету пакета: від найнижчого - 0 до найвищого - 7. Наступні три біта - критерій вибору маршруту. Якщо біт D (Delay - затримка) встановлений в 1, то маршрут вибирається з мінімізацією затримки доставки пакету, якщо встановлено в 1 біт T (Throughput - пропускна здатність) - для максимізації пропускної здатності, а біт R (Reliability - надійність) - для максимізації надійності доставки. Решта - два біти, мають нульове значення.

Поле загальної довжини займає 2 байти і характеризує загальну довжину пакета з урахуванням заголовка і поля даних. Максимальна довжина пакета обмежена розрядністю поля, яка визначає цю величину - 65535 байт. Залежить від максимальної довжини пакета протоколу нижнього рівня. Якщо це кадри Ethernet, то вибираються пакети з максимальною довжиною 1500 байт.

Ідентифікатор пакету займає 2 байти і використовується для розпізнавання пакетів, при фрагментації вихідного пакета. Всі фрагменти одного пакету повинні мати однакове значення цього поля.

Флаги займають 3 біта і містять ознаки, пов'язані з фрагментацією. Встановлений в 1 біт DF (Do not Fragment - не фрагментований) забороняє маршрутизатора фрагментувати даний пакет, а встановлений в 1 біт MF (More Fragments - більше фрагментів) говорить про те, що даний пакет є проміжним (не останнім) фрагментом. Біт, який залишився - зарезервований.

Поле зсуву фрагмента займає 13 біт і задає зсув у байтах поля даних цього фрагмента відносно початку поля даних вихідного не фрагментованого пакета. Використовується при складанні/розбиранні фрагментів пакетів. Зміщення повинно бути кратним 8 байтам.

Поле часу життя (Time To Live, TTL) - 1 байт, задає граничний термін, протягом якого пакет може переміщатися по мережі. Час життя пакету вимірюється в секундах і задається джерелом пакетів.

Поле протоколу верхнього рівня - один байт - ідентифікатор, який вказує, якому протоколу верхнього рівня належить інформація, яка розміщена в полі даних пакета. Значення ідентифікаторів для протоколів (RFC 1700), за адресою http://www.iana.org. 6 - в пакеті знаходиться повідомлення TCP, 17 - повідомлення UDP, 1 - повідомлення ICMP.

Контрольна сума заголовка -2 байти, розраховується тільки по заголовку.

Поля IP-адрес джерела і приймача мають довжину - 32 біта.

Поле параметрів - необов'язково і використовується тільки при налагодженні мережі. Так як число підполів в полі параметрів може бути довільним, то в кінці заголовка має бути додано декілька нульових байтів для вирівнювання заголовка пакета по 32-бітній границі.

Та частина пакету, в якій містяться тільки фактичні дані, називається MSS (Maximum Segment Size) - це ще один параметр протоколу TCP, що визначає найбільший сегмент даних TCP, які можуть бути передані за один раз.

Тобто, MTU = MSS + заголовки TCP/IP. У реєстрі MSS задається так: **HKEY_LOCAL_MACHINE \ System \ CurrentControlSet \ Services \ VxD \ MSTCP "DefaultMSS" = "ваше число".**

Для заголовка теж є загальноприйнятий розмір - це 40 байт (20 байт IP і 20 байт TCP), зазвичай MSS = MTU - 40..

З цієї причини у визначенні оптимального розміру MTU є деякі тонкощі.

Давайте на прикладі розглянемо передачу даних при різному розмірі MTU по широкосмугової лінії T1 (пропускна здатність T1 = 1544000 bits/sec), використовуючи наступну формулу: [(MSS + заголовок) * 8 бітів / байт] / [1544000 біт / sec] = затримка на один хоп (на кожен комп'ютер в мережі при передачі нашого пакета).

Використовуючи в цій формулі різні величини MTU, ми можемо обчислити затримку одного пакета. Якщо MTU = 1500, тоді: (1460 + 40) * 8/1544000 = 0.7772 ms. Якщо ж MTU = 576, то: (536 + 40) * 8/1544000 = 0.2924 ms. Припустимо, що при передачі пакету зустрічається 10 серверів (хопов), тоді при MTU = 1500 отримаємо затримку 7.772 ms, а при 576 - 2.924 ms - різниця досить помітна - очевидно, якщо пакети менші за розміром, то вони передаються швидше (через обмеження продуктивності лінії). Однак не все так просто.

Використовуючи ту ж формулу порахуємо, за який проміжок часу буде переданий файл розміром 1Mb за тією ж широкосмуговою лінією T1. Один мегабайт дорівнює 1024 KB і дорівнює 1048576 байтам. Якщо MTU = 1500, то, як ми з'ясували, затримка на один хоп

складе 0.7772 ms. Скільки при цьому знадобиться послати пакетів? 1 Mb / MSS = 1048576/1460 = 718.2, або всього потрібно 719 ефективних пакетів, щоб передати 1 мегабайт. Далі, множимо 719 пакетів на 0.7772 ms, отримуємо 558.8068 ms, або 5.588 секунд затримки на один хоп. Якщо ж ми передаємо свій файл через 10 хопів (ситуація зустрічаються частіше, ніж коли через один), то отримуємо 55.88 sec - час, який ми (вірніше, провайдер, який має лінію T1) витратили на передачу файлу в 1Mb при ідеальному зв'язку. Якщо ж MTU = 576, тоді: 1 Mb / MSS = 1048576/536 = 1956.3, або потрібно 1957 пакетів, щоб передати 1 мегабайт. Далі, множимо кількість пакетів на затримку кожного з них: 1957 * 0,2924 = 572,2268 ms, або 5,722 секунди на один хоп. Ну і відповідно на 10 хопів доведеться витратити 57,22 секунд. Як бачимо, через те, що при використанні великих пакетів передається менше заголовків, реальна швидкість передачі файлу виходить вище.

Для того, щоб передати 1 мегабайт при використанні MTU = 1500, доводиться пересилати ще й «додаток» заголовків з 28760 байтів, тоді як при використанні MTU = 576 отримуємо аж 1957 * 40 = 78,280 байтів, тобто додаткові 49520 байт заголовків на кожен мегабайт корисної інформації. Для нашої 10-хопової передачі це виливається в зайвих 1,34 секунди при передачі кожного мегабайту навіть при швидкому зв'язку. Ця різниця, можливо, буде ще трохи вище на практиці, оскільки сучасні реалізації TCP / IP використовують більші за розміром заголовки.

Якщо ж провести аналогічні розрахунки для модемного зв'язку на швидкості 33600, то отримаємо, що для передачі мегабайту інформації на відстань одного хопу, тобто безпосередньо вашому провайдеру, буде витрачено в ідеалі 256 секунд при MTU = 1500 і 268 секунд при MTU = 576. Різниця на одному переході 12 секунд або близько 5%! Але не слід забувати, що ці цифри вийдуть за умови відсутності фрагментації пакетів, тобто якщо у провайдера MTU = 1500. Якщо ж це не так, то, зрозуміло, більший, ніж потрібно, пакет буде фрагментуватися - розбиватися на декілька пакетів і навіть розбавлятися «повітрям», і зв'язок погіршиться на 10-50%.

Таким чином, логічним вважається, що великі пакети в підсумку все-таки краще, і якщо провайдер налаштував свої сервери і маршрутизатори на великі пакети, то треба прагнути використовувати це повністю, але не забувати і про те, що в Інтернеті зустрічаються

сервери з MTU = 576 . Тим не менш, якщо чиста продуктивність не є остаточною метою, то менші пакети будуть більш «швидкими», оскільки вони вимагають менше часу для своїх подорожей по мережі. В Інтернеті при проходженні пакетів, ймовірно, зустрінуться різні сервери з різними настройками, але для початку все-таки бажано визначити оптимальне значення MTU при зв'язку тільки з вашим провайдером, оскільки саме воно може надати вирішальне значення при оптимізації вашого доступу в мережу.

При підключенні до Інтернету в термінальному режимі – іноді, при здійсненні реєстрації користувачем, в одному з рядків з'являється рекомендоване значення MTU.

Для ручного визначення MTU досить використовувати стандартну утиліту ping:

PING -f -l yyyy xxx.xxx.xxx.xxx -n 10,

де «xxx.xxx.xxx.xxx» - IP-адреса тестованого сервера, «-l» (англійська літера «ель», а не одиниця), yyyy - розмір буфера відправки (MTU) від 576 до 1492 байт (наприклад 1500 = 1472 + 28, де 1472 - розмір неподільного пакету, 28 = 20 байт заголовок IP + 8 байт ICMP. для Ethernet MTU = 1500 байт, для SLIP - 1006, для PPPoE -1492, для PPP (модемного зв'язку) - 576.

Таким чином можна вручну підібрати найбільш відповідне значення MTU для вашого з'єднання. Припустимо ми визначили, що для нас оптимальним є розмір буфера відправки рівний 1500.

Додаємо необхідні дані до реєстру.

Значення MTU

HKEY_LOCAL_MACHINE\ System \ CurrentControlSet \ Services \ Class \ NetTrans \ 000x "MaxMTU" = "1500"

HKEY_LOCAL_MACHINE\ System \ CurrentControlSet \ Services \ VxD \ MSTCP

MSS (розмір корисних даних) = ab.1 & 6mb.co & m & ab.16mb.com MTU – 40 "DefaultMSS" = "1460"

HKEY_LOCAL_MACHINE\ System \ CurrentControlSet \ Services \ VxD \ MSTCP

Розмір буфера, в якому накопичується вміст області даних (MSS) декількох отриманих пакетів, перш ніж передається далі, наприклад, в браузер. Розмір RWIN обов'язково повинен бути кратний MSS і зазвичай для кращої ефективності модемного з'єднання рекомендується його встановлювати рівним 4-8 MSS. Проте надмірно

великий розмір буфера також небажаний, особливо на поганих лініях - при втраті всього одного пакета в разі збою на лінії буде повторно затребуваний не один втрачений пакет, а всі пакети з цього буфера, що займе деякий час - 1460 * 4 = 5840 "DefaultRcvWindow" = "5840"

1.2 IP-адресація

IP-адреси являють собою основний тип адрес, на підставі яких мережний рівень передає пакети між мережами.

Мережну адресу встановлює користувач (адміністратор) або вона призначається динамічно, протоколом DHCP з діапазону виділених адрес.

Мережна адреса має бути достатньо довгою (в IP-мережах версії IPv4 вона містить 32 біта (4 байти)), що дорівнює 2^{32} = 4294967296 та ієрархічною (на відміну від MAC-адрес інтерфейсів).

4-х байтова адреса може бути представлена у різних системах числення: десятковій, двійковій, шістнадцятковій:

175.100.220.14;

10101111 01100100 11011100 00001110;

A F 6 4 D C 0 E

Незважаючи на те, що в третьому випадку цифр менше, поширеним є десяткове подання, точніше точково-десяткове. Частина адреси (старші розряди) є номером мережі, а інша частина (молодші розряди) - номером вузла в мережі. Виходячи з того, яка частина адреси відноситься до номера мережі, а яка - до номера вузла, адреси діляться на 5 класів: A, B, C, D та E. Для унікальної адресації вузлів використовуються тільки три перших класи адрес.

В адресі класу A старший байт задає адресу мережі, а три молодших байти - адресу вузла (host). 0.x.y.z - 127.x.y.z - мереж 2^7 - 2 (не може бути 0 і 127) = 126 мереж, вузлів 2^{24} = 16 млн.

В адресі класу B два старших байти задають адресу мережі, а два молодших байти - адресу вузла (host). 128.x.y.z - 191.x.y.z - мереж 2^{14} = 16 тис., Вузлів 2^{16} = 64 тис.

В адресі класу C три старших байти задають адресу мережі, а молодший байт - адресу вузла. 192.x.y.z - 223.x.y.z - мереж 2^{21} = 2 млн., Вузлів 2^8 = 256.

Існує також багатоадресний клас D і резервний клас E.

Номер вузла (адреса host) не може складатися тільки з одних одиниць або нулів. Якщо в поле адреси вузла всі нулі, це означає, що задається номер (адреса) мережі або підмережі.

Приватні адреси:

– 10.0.0.0 - 10.255.255.255 / 8;
– 172.16.0.0 - 172.31.255.255 / 12;
– 192.168.0.0 - 192.168.255.255 / 16.

Особливі адреси:

– IP-адреса 0.0.0.0;
– IP-адреси з нульовим номером мережі позначають поточну мережу;
– адреса, яка складається з усіх одиниць, забезпечує широкомовлення в межах поточної (зазвичай локальної) мережі - 255.255.255.255;
– адреси, в яких вказана мережа, але з усіма одиницями в поле номера хоста, забезпечують широкомовлення в межах віддаленої локальної мережі;
– адреси, які мають вид 127.x.y.z зарезервовані для тестування мережного програмного забезпечення методом зворотної передачі;
– 169.254.0.0/16 (169.254.0.1 - 169.254.255.255) - зарезервовані APIPA.

Маски. У класовій адресації маємо недоцільність використання адресного простору для невеликих мереж, тому введене додаткове поле, що має назву - маска мережі.

Маска - 32 бітове число, яке використовується в парі з IP-адресою. Двійковий запис маски містить послідовність одиниць в тих розрядах, які повинні в IP-адресі інтерпретуватися як номер мережі. Оскільки номер мережі - цільна частина адреси, то одиниці в масці повинні представляти безперервну послідовність.

Додаючи кожній IP-адресі маску, можна відмовитися від понять класів адрес і зробити систему адресації більш гнучкою.

Нехай для IP-адреси 129.64.134.5 вказана маска 255.255.128.0. Якщо ігнорувати маску, то відповідно до системи класів 129.64.134.5 відноситься до класу B, тому номер мережі - перші два байти - 129.64.0.0, а номер вузла - 0.0.134.5.

Якщо ж використовувати для визначення границі номеру мережі маску, то 17 послідовних двійкових одиниць у масці 255.255.128.0, «накладені» на IP-адресу, ділять його на наступні дві частини: номер мережі 10000001. 01000000. 1 (129.64.128.0) і номер вузла 0000110. 00000101 (0.0.6.5).

Маршрутизатор, отримавши пакет, з адресою призначення отримує адресу мережі, яку реалізує шляхом логічного множення мережної адреси вузла на маску. Безперервна послідовність одиниць в старших розрядах маски задає число розрядів адреси, які відносяться до номера мережі. Молодші розряди маски, що дорівнюють 0, відповідають розрядам адреси вузла в мережі.

Маски змінної довжини. Використовуючи маски різної довжини для створення підмереж, адміністратор може формувати підмережі різного розміру в межах однієї автономної системи. Таким чином, маски змінної довжини (Variable-Length Subnet Mask - VLSM) дозволяють створювати підмережі різного розміру, при цьому гнучко задаючи границі між полем адреси мережі і полем адреси вузла.

VLSM дають можливість задіяти більше ніж одну маску підмережі в межах виділеного адресного простору мережі.

Наприклад, для формування мереж по 30 вузлів у кожній потрібно 27 розрядів маски, які містять одиниці, а для створення мережі, яка з'єднує пару маршрутизаторів ("точка-точка"), потрібно всього дві адреси, маска повинна мати 30 безперервних одиниць. При цьому частина адресного простору може бути використана для створення мереж по 30 вузлів, а частина незайнятих адрес - для формування пар адрес при створенні зв'язків "точка-точка".

Примітка. При використанні маски в 30 двійкових розрядів два молодших розряди в адресі дозволяють сформувати $2^2 = 4$ адреси, з яких перший потрібен для адресації мережі, другий і третій - для адресації вузлів, а четвертий - в якості широкомовної адреси.

1.3 Приклади вирішення завдань

Розглянемо **практичне завдання з визначення класу мережі та типу адреси**. Для наведених адрес маємо:

– 201.10.255.0 – клас С, адреса мережі;

– 190.195.0.255 – клас B, адреса вузла;
– 9.255.255.255 – клас A, широкомовна адреса;
– 10.10.255.0 – клас A, адреса вузла;
– 134.11.255.255 – клас B, широкомовна адреса;
– 252.250.0.255 – клас E, зарезервована адреса;
– 194.18.144.25 – клас C, адреса вузла;
– 129.77.0.0 – клас B, адреса мережі;
– 237.101.5.0 – клас D, групова адреса;
– 126.0.0.0 – клас A, адреса мережі.

Наведемо приклад **застосування маски для визначення номера мережі і діапазону вузлів** маючи IP-адресу і маску.

Нехай задана IP-адреса: 210.56.78.212. Віднесемо до неї маску 255.255.255.224 (префікс / 27) (в двійковому вигляді). Що тепер номер мережі, а що номер вузла?

Представимо у двійковому вигляді адресу, виконаємо логічне множення на маску, та отримаємо адресу мережі - 210.56.78.192.

Кількість вузлів маємо 2^5 (кількість нулів у масці = 5) = 32 - 2 (адреса мережі і широкомовна адреса) = 30.

Отримаємо діапазон адрес даної мережі:

– адреса першого вузла - 210.56.78.193;
– адреса останнього вузла - 210.56.78.222;
– широкомовна адреса - 210.56.78.223.

У ряді випадків для зручності управління адміністратор може самостійно формувати підмережі всередині виділеного йому адресного простору.

Розглянемо **практичне завдання з формування підмереж із однаковою кількістю вузлів**. Почнемо з формування мереж класу C.

Маємо адресу мережі - 212.24.222.0 / 24.

Розділимо її на 2 підмережі, для цього використовуємо 1 біт маски. Маска збільшиться на 1.

212.24.222.0 | 0000000 / 25 (255.255.255.128)

255.255.255.1 |

Перша підмережа 212.24.222.0 / 25.

Діапазон адрес - 212.24.222.1 - 212.24.222.126, 212.24.222.127 - широкомовна адреса.

Друга підмережа 212.24.222.128 / 25.

Діапазон адрес - 212.24.222.129 - 212.24.222.254, 212.24.222.255 - широкомовна адреса.

Маючи маску 255.255.255.224 (/27) можемо отримати 2^3 підмереж і 2^5-2 вузлів.

Якщо адміністратору виділена адреса мережі класу С (дорівнює 198.11.163.0) і йому необхідно створити 10 комп'ютерних підмереж по 14 вузлів, то для адресації 10 підмереж потрібно 4 розряди адреси (4 біта). У цьому випадку максимально можна бути задати 16 підмереж по 14 вузлів у кожній. З 16 підмереж адміністратор використовує 10, а решта 6 використовуватися не будуть (резервні). У практичних випадках мережі формуються з різною кількістю вузлів в мережі.

Розглянемо **практичне завдання з формування підмереж з використанням масок змінної довжини**.

Дана адреса мережі **200.33.224.0/24** і маска, необхідно сформувати **9** підмереж:

- 3 підмережі на 50 вузлів;
- 3 підмережі на 10 вузлів;
- 1 підмережа на 4 вузла;
- 2 підмережі на 2 вузли.

Після збільшення маски на 2 біти (**200.33.224. |00| 000000**) маємо наступні підмережі:

- **200.33.224.0/26** – перша на 62 вузли (виділяємо на 50 вузлів);
- 200.33.224.1-200.33.224.62 – діапазон;
- 200.33.224.63 – широкомовна адреса;
- **200.33.224.64/26** – друга на 62 вузли (виділяємо на 50 вузлів);
- 200.33.224.65-200.33.224.126 – діапазон;
- 200.33.224.127 – широкомовна адреса;
- **200.33.224.128/26** – третя на 62 вузли (виділяємо на 50 вузлів);
- 200.33.224.129-200.33.224.190 – діапазон;
- 200.33.224.191 – широкомовна адреса;
- **200.33.224.192/26** – підмережа на 62 вузли (цю ділимо далі по 10 вузлів).

Після збільшення маски на 2 біти (**200.33.224. 11 |00| 0000**) маємо наступні підмережі:

- **200.33.224.192/28** – четверта на 16 вузлів (виділяємо на 10 вузлів);

– 200.33.224.193-200.33.224.206 – діапазон;
– 200.33.224.207– широкомовна адреса;
– **200.33.224.208/28** – п’ята на 16 вузлів (на 10 вузлів);
– 200.33.224.209-200.33.224.222 – діапазон;
– 200.33.224.223 – широкомовна адреса;
– **200.33.224.224/28** – шоста на 16 вузлів (на 10 вузлів);
– 200.33.224.225-200.33.224.238 – діапазон;
– 200.33.224.239 – широкомовна адреса;
– **200.33.224.240/28** - підмережа на 14 вузлів (цю ділимо далі по 4 вузли).

Після збільшення маски на 1 біт (**200.33.224. 1111 |0| 000**) маємо наступні підмережі:

– **200.33.224.240/29** – сьома на 6 вузлів (виділяємо на 4 вузли);
– 200.33.224.241-200.33.224.246 – діапазон;
– 200.33.224.247 – широкомовна адреса;
– **200.33.224.248/29** - підмережа на 6 вузлів (цю ділимо далі по 2 вузли).

Після збільшення маски на 1 біт (**200.33.224. 11111 |0| 00**) маємо наступні підмережі:

– **200.33.224.248/30** – восьма на 2 вузли;
– 200.33.224.249-200.33.224.250 – діапазон;
– 200.33.224.251 – широкомовна адреса;
– **200.33.224.252/30** – дев’ята на 2 вузли;
– 200.33.224.253-200.33.224.254 – діапазон;
– 200.33.224.255 – широкомовна адреса.

Таким чином отримали 9 підмереж із різною кількістю вузлів.

1.4 Маршрутизація

1.4.1 Типи маршрутизації

Маршрутизація без таблиць:

– лавинна - кожен роутер передає пакет всім своїм сусідам по всім активним інтерфейсів, крім того, від якого його отримав. Мінус - засмічення мережі надлишкової службовою інформацією;

– маршрутизація, керована подіями - пакет до певної мережі призначення надсилається за маршрутом, вже наводив раніше до успіху (для даної адреси призначення);

– маршрутизація від джерела - відправник поміщає в пакет інформацію про те, які хопи повинен пройти пакет до мережі призначення. Маршрут адміністратор може задавати вручну.

Маршрутизація на основі таблиць:

– статична «фіксована» - таблиці вводяться в пам'ять кожного роутера вручну адміністратором мережі. Всі записи в таблиці мають статус статичних з нескінченний терміном життя; таблиця має, як мінімум п'ять стовпців: адреса мережі - мережа або окремий ip-адреса, куди повинен бути доставлений пакет. Маска мережі - щоб однозначно ідентифікувати сіть. Шлюз - для передачі пакетів з різними адресами призначення. Інтерфейс (номер порту) - ініціалізує інтерфейс, з якого буде відправлений пакет. Метрика - число, що характеризує канал зв'язку. Даний метод зарекомендував себе в невеликих локальних мережах і на магістральних лініях. Коли потрібно доставляти все або більшість пакетів на один вузол, використовується поняття «шлюз». Коли збіг адреси призначення з адресою мережі не відбувається, дані йдуть на дефолтовий шлюз.

– адаптивна (динамічна) - всі зміни конфігурації мережі автоматично вносяться в таблиці роутерів протоколами маршрутизації. У таких таблицях є запис ttl маршруту (час життя в секундах). Якщо після закінчення часу існування маршруту не підтверджується протоколом маршрутизації, то він вважається неробочим.

1.4.2 Протоколи внутрішньої маршрутизації

Для маршрутизації в ip-мережах застосовуються протоколи, в яких маршрут вибирається використовуючи різні варіанти критеріїв, пов'язаних зі зменшенням часу проходження пакету і якістю маршруту (критерії: найкоротша відстань (кількості роутерів на шляху пакета - хопов), пропускна здатність каналів між роутерами,

надійність каналів , латентність (затримка, очікування (доставки даних) - збільшує реальне час відгуку)).

Основою протоколу маршрутизації є алгоритми маршрутизації, які як раз і застосовуються для визначення найкращого шляху пакетів від джерела до приймача. Для подання роботи алгоритмів маршрутизації мережа представляється у вигляді графа. При цьому вузлами графа є маршрутизатори, а ребрами - фізичні лінії зв'язку між ними. Кожній грані відповідає певне число - вартість, що залежить від довжини лінії зв'язку, швидкості передачі даних або фінансової вартості лінії.

Протокол маршрутизації формує в роутерах узгоджені один з одним таблиці маршрутизації, які забезпечують доставку пакета від вихідної мережі в мережу призначення за кінцеве число кроків. Таблиця маршрутизації задає оптимальний шлях для пакета. Таблиця може бути або статична, або динамічна.

Який шлях оптимальний - визначається метрикою - умовна вартість передачі по мережі. Повна вартість маршруту дорівнює сумі метрик мереж, по маршруту. Маршрутизатор вибирає маршрут з найменшою метрикою.

Алгоритми маршрутизації діляться на 2 групи:

– дистанційно-векторні алгоритми (distance vector). Кожен роутер періодично широкомовно розсилає по мережі вектор відстаней від себе до всіх відомих йому мереж. Отримавши від сусіда вектор відстаней до відомих тому мереж, роутер нарощує компоненти вектора на величину відстані від себе до даного сусіда, а так же доповнює вектор інформацією про відомі йому самому інших мережах. Так кожен роутер дізнається інформацію про всіх наявних мережах і про відстані до них. Потім він вибирає з альтернативних маршрутів до кожної мережі той, який має найменшу метрикою. Найближчий роутер, який передав інформацію про даному маршруті, відзначається в таблиці маршрутизації як "next hop". Дистанційно-векторні алгоритми добре працюють тільки в невеликих мережах. Приклад - протокол rip;

– алгоритми стану зв'язків (link state) забезпечують кожен роутер інформацією для побудови точного графа зв'язків мережі. Всі роутери працюють на підставі одного й того ж графа. Приклад - протокол ospf.

Протокол rip (routing information protocol) - протокол маршрутної інформації.

Алгоритм роботи протоколу:

– крок 1 - створення мінімальної таблиці. У початковому стані в кожному роутере програмним забезпеченням стека tcp / ip автоматично створюється мінімальна таблиця маршрутизації, в якій враховуються тільки безпосередньо приєднані мережі.

– крок 2 - розсилка мінімальної таблиці сусідам. Після ініціалізації кожен роутер починає посилати своїм сусідам повідомлення протоколу rip, в яких міститься його мінімальна таблиця. Rip-повідомлення передаються в udp-дейтаграммах і включають два параметра для кожної мережі: ip-адреса мережі та відстань до неї від передавального повідомлення роутера.

– крок 3 - отримання rip-повідомлень від сусідів і обробка отриманої інформації. Після отримання повідомлень від сусідів роутер нарощує кожне отримане поле метрики на одиницю і запам'ятовує, через який порт і від якого роутера отримана нова інформація. Потім роутер порівнює нову інформацію з тієї, яка зберігається в таблиці маршрутизації. Rip заміщає запис про будь-якої мережі тільки в тому випадку, якщо нова інформація має кращу метрику. За наявності кількох рівнозначних за метрикою записів, то в таблиці залишається один запис, яка прийшла в роутер першої.

– крок 4 - розсилка нової таблиці сусідам. Далі - до кроку 3.

Щоб отримувати попередження - маршрут недійсний, rip використовує два механізми:

– закінчення ttl маршруту. При надходженні чергового rip-повідомлення, яке підтверджує справедливість цього запису в таблиці, таймер ttl встановлюється в початковий стан - потім з нього кожну секунду віднімається одиниця. Якщо за час ttl не прийде нове повідомлення про цей маршрут, він позначається як недійсний. У протоколі rip період розсилки - 30 секунд, а ttl маршруту - 180 секунд.

– вказівка нескінченного відстані до недоступної мережі. В rip нескінченним умовно вважається відстань 16 хопов.

Протокол ospf (open shortest path first) - вибір найкоротшого шляху першим. Алгоритм роботи протоколу:

– крок 1 - кожен роутер будує граф зв'язків мережі, в якому вершинами є роутери і ip-мережі, а ребрами - інтерфейси роутерів. Всі

роутери обмінюються зі своїми сусідами тією інформацією про графа мережі, якою вони володіють до даного моменту. Повідомлення, за допомогою яких поширюється топологічна інформація, називаються оголошеннями про стан зв'язків мережі (link state advertisements, lsa). При передачі топологічної інформації роутери її не змінюйте (на відміну від rip). В результаті все роутери мережі розташовують ідентичними відомостями про графа мережі, які зберігаються в базі даних про топологію мережі.

– крок 2 - знаходження оптимальних маршрутів за допомогою отриманого графа. Завдання вирішується за допомогою алгоритму дейкстри. Алгоритм дейкстри обчислює найкоротший шлях між двома точками в мережі, використовуючи граф за методом вузлів і кордонів. При цьому кожен роутер вважає себе центром мережі і шукає оптимальний маршрут до кожної відомої йому мережі. У кожному знайденому маршруті запам'ятовується тільки один крок до наступного роутера. Дані про цей крок потрапляють в таблицю маршрутизації.

Всі протоколи маршрутизації можна розділити на дві великі групи: зовнішні (EGP - Exterior Gateway Protocol) і внутрішні (IGP - Interior Gateway Protocol). Щоб пояснити відмінності між ними - потрібно термін "автономна система".

1.4.3 Протоколи граничної маршрутизації

Маршрутизація Інтернет функціонує в межах автономних систем. АС (домен маршрутизації) - сукупність мереж (група роутерів) під єдиним адміністративним керуванням, що забезпечує загальну політику маршрутизації. Автономні системи з'єднуються зовнішніми шлюзами. Реєстрація АС відбувається централізовано.

Номер АС - 16 розрядів (65535). Інтернет - набір взаємопов'язаних АС, що забезпечує багаторівневий підхід до маршрутизації: маршрут визначається як послідовність АС, потім - як послідовність мереж, а потім - веде до кінцевого вузла.

АС може належати до наступних категорій:

– обмежена (stub) AS автономна система має єдине підключення до однієї зовнішньої автономній системі. У такій автономній системі присутній тільки локальний трафік;

– многоінтерфейсная (multihomed) AS має під'єднання до кількох віддалених автономних систем, однак по ній заборонено проходження транзитного трафіку;

– транзитна (transit) AS має підключення до декількох автономних систем і відповідно до обмежень може пропускати через себе як локальний, так і транзитний трафік.

Загальна топологія Internet складається з транзитних, многоінтерфейсних і обмежених автономних систем. Протокол граничної маршрутизації (BGP - Border Gateway Protocol) - це протокол маршрутизації між автономними системами. Він заснований на методах маршрутизації, які називаються "маршрутизація вектором шляху".

Маршрутизація з використанням вектора шляхів відрізняється і від маршрутизації з використанням вектора довжини маршруту, і від маршрутизації станом лінії. Кожен вхід в таблицю маршрутизації містить мережу пункту призначення, наступний маршрутизатор і шлях до пункту призначення. Шлях зазвичай визначається як впорядкований список автономної системи, який повинен пройти пакет для досягнення пункту призначення.

Автономний граничний маршрутизатор - бере участь в маршрутизації з використанням вектора шляхів, сповіщає про досяжності мереж в їх власній автономній системі для сусідніх автономних прикордонних маршрутизаторів. Концепція оточення тут та ж сама, як у вже розглянутих протоколах RIP і OSPF. Два прикордонних маршрутизатора автономних систем, підключення до тієї ж самої мережі, - сусіди.

Граничний маршрутизатор автономної системи отримує свою інформацію від внутрішнього алгоритму маршрутизації, такого як RIP і OSPF. Кожен маршрутизатор, який отримує вектор шляху, перевіряє, що запропонований шлях узгоджений з його політикою. Якщо політика маршрутизації відповідає записаної в програмі, маршрутизатор оновлює таблиці маршрутизації і модифікує повідомлення, перш ніж послати його до наступного сусідові.

Роутер взаємодіє з іншими роутерами по протоколу BGP тільки в тому випадку, якщо адміністратор явно вказує, що ці роутери є його сусідами. Таким чином, адміністратор може вирішувати, з якими автономними системами він буде обмінюватися трафіком, а з якими ні. Одночасно протоколи RIP і OSPF обмінюються маршрутною інформацією з усіма роутерами, що знаходяться в межах їх безпосередньої досяжності.

Для встановлення сеансу з зазначеними сусідами BGP-роутери використовують протокол TCP (порт 179) з аутентифікацією. Основним повідомленням протоколу BGP є "UPDATE", за допомогою якого роутер повідомляє роутера сусідній автономної системи про досяжності мереж, що належать до його власної автономної системі. "UPDATE" - це тригерні оголошення, яке посилається сусідові тільки тоді, коли в автономній системі що-небудь різко змінюється. В одному повідомленні "UPDATE" можна оголосити про одне новому маршруті або анулювати кілька перестали існувати.

BGP відрізняється від RIP або OSPF тим, що BGP використовує TCP в якості транспортного протоколу.

1.5 Контрольні питання

1. Визначення та процедура обчислення MTU.
2. Різниця між MTU та MSS.
3. Загальноприйняті стандарти MTU для різних технологій канального рівня.
4. Структура заголовку IP-пакету.
5. Ручне визначення MTU. Зміни в реєстрі.

2 МЕРЕЖНІ СЛУЖБИ DHCP ТА DNS

2.1 Протокол динамічної конфігурації хостів

Протокол динамічної конфігурації хостів (Dynamic Host Configuration Protocol, DHCP) автоматизує процес конфігурації мережних інтерфейсів, гарантуючи від дублювання адрес за рахунок централізованого управління їх розподілом. DHCP побудований за схемою клієнт сервер, де DHCP-сервер виділяє мережні адреси і доставляє конфігураційні параметри ПК, які динамічно конфігуруються. Клієнт і сервер можуть погоджувати список необхідних параметрів. ПК не повинна діяти як DHCP-сервер, якщо вона спеціально не налаштована системним адміністратором.

DHCP не може використовуватися для конфігурації маршрутизаторів. Список основних завдань DHCP:

– DHCP є механізм, а не політика і управляється системними адміністраторами, шляхом завдання конфігураційних параметрів;

– клієнти не повинні вимагати ручної конфігурації і повинні читати локальні конфігураційні параметри;

– мережі не вимагають ручної конфігурації для окремих клієнтів. Адміністратор не вводить індивідуальні параметри клієнта;

– DHCP не вимагає окремого сервера для кожної підмережі;

– клієнт DHCP може отримати кілька відгуків на запит конфігураційних параметрів. Для підвищення надійності та швидкодії використовують декілька серверів для перекриття областей мережі;

– DHCP повинен співіснувати з ПК, які сконфігуровані вручну.

DHCP повинен також:

– гарантувати, що будь-яка мережна адреса не буде використовуватися більш ніж одним клієнтом одночасно;

– підтримувати DHCP конфігурацію клієнта при стартовому перезавантаженні DHCP-клієнта - при кожному запиті по мірі можливості, присвоюється один і той же набір конфігураційних параметрів (мережна адреса);

– підтримувати конфігурацію DHCP-клієнта при перезавантаженні сервера (той же набір конфігураційних параметрів);

– дозволяти автоматично отримувати конфігураційні параметри новим клієнтам, щоб уникнути ручної конфігурації;

– підтримувати фіксоване або постійне присвоєння конфігураційних параметрів для заданого клієнта.

Модель DHCP пам'яті характеризується записами ключ-значення для кожного клієнта, де ключ це деякий унікальний ідентифікатор (номер IP-мережі і унікальний ідентифікатор в межах мережі), а значення містить набір конфігураційних параметрів клієнта. Ключ може являти собою пару номер IP-мережі, апаратну адресу.

Повідомлення	**Використання**
DHCPDISCOVER	Клієнт посилає повідомлення широкомовно, щоб виявити доступний сервер
DHCPOFFER	Надсилається сервером клієнтові у відповідь на DHCPDISCOVER і містить пропозицію по конфігураційним параметрам
DHCPREQUEST	Повідомлення клієнта серверу. Робить запит параметрів від одного сервера і відкидає пропозиції інших серверів, підтверджує коректність раніш присвоєної адреси після перезавантаження системи
DHCPACK	Надсилається сервером клієнтові і містить конфігураційні параметри, включаючи присвоєну мережну адресу
DHCPNAK	Надсилається сервером клієнтові, повідомляючи про те, що мережна адреса не коректна (клієнт перемістився в нову підмережу) або час використання адреси клієнтом минув
DHCPDECLINE	Клієнт і сервер виявили, що мережна адреса вже використовується
DHCPRELEASE	Надсилається клієнтом серверу з метою відмови від мережної адреси і анулює час дії адреси
DHCPINFORM	Надсилається клієнтом серверу з проханням про локальні параметри

DHCP може працювати в різних режимах, включаючи:

– ручне призначення статичних адрес - адміністратор, з пулом доступних адрес, постачає DHCP-сервер інформацією про жорстку

відповідність IP-адрес фізичним адресам або іншим ідентифікаторам клієнтських вузлів;

– автоматичне призначення статичних адрес - DHCP-сервер самостійно без втручання адміністратора довільним чином вибирає клієнтові IP-адресу з пулу IP-адрес. Адреса дається клієнту з пулу в постійне користування. При наступних запитах сервер повертає клієнтові ту же IP-адресу;

– автоматичний розподіл динамічних адрес - DHCP-сервер видає клієнту адресу та набір конфігураційних параметрів на обмежений час, термін оренди. Коли DHCP-клієнт видаляється з мережі, IP-адреса автоматично звільняється.

У всіх режимах роботи адміністратор при конфігуруванні DHCP-сервера повідомляє йому один або декілька діапазонів IP-адрес - все адреси належать до однієї мережі - мають одне і те ж значення в поле номера мережі.

2.2 Централізована служба DNS

Централізована служба DNS (Domain Name System - система доменних імен), заснована на розподіленої базі відображень «доменне ім'я - IP-адреса». DNS являє собою, з одного боку, базу даних, розподілену між ієрархічно структурованими серверами імен, з іншого боку, протокол прикладного рівня, який організовує взаємодію між хостами і серверами імен для виконання операцій перетворення.

Протоколу DNS призначено порт з номером 53 і працює він поверх протоколу UDP транспортного рівня. Основні специфікації DNS містяться в документах RFC 1034 і RFC 1035.

Служба DNS використовує в своїй роботі DNS-сервери і DNS-клієнти. DNS-сервери підтримують розподілену базу відображень, а DNS-клієнти звертаються до серверів із запитами про перетворення доменного імені в IP-адресу.

DNS має ієрархічну деревоподібну структуру, яка допускає наявність в імені довільної кількості складових частин.

Ієрархія доменних імен аналогічна ієрархії імен файлів, прийнятої в файлових системах. Дерево імен починається з кореня, що

позначається крапкою (.). Потім слідує старша символьна частина імені, друга за старшинством символьна частина імені і т.п. Молодша частина імені відповідає кінцевому вузлу мережі. Запис доменного імені починається з наймолодшої складової, а закінчується найстаршою. Складові частини доменного імені відокремлюються крапкою. В імені partnering.microsoft.com складова partnering є ім'ям одного з комп'ютерів в домені microsoft.com.

Поділ імені на частини дозволяє розділити адміністративну відповідальність за призначення унікальних імен в межах рівня ієрархії - відповідальність за те, щоб імена із закінченням «ru», мали унікальну наступну вниз по ієрархії частину - все імена типу www.ru, mail.mmt.ru або m2.zil.mmt.ru відрізняються другою, за старшинством, частиною.

Сукупність імен, у яких декілька старших складових частин збігаються, утворюють домен імен (domain). Імена www1.zil.mmt.ru, ftp.zil.mmt.ru, yandex.ru і s1.mgu.ru входять в домен ru - всі вони мають одну загальну старшу частину з ім'ям ru. Іншим прикладом є домен mgu.ru. У нього входять імена s1.mgu.ru, s2.mgu.ru і m.mgu.ru. Цей домен утворюють імена, у яких дві старші частини завжди рівні mgu.ru. Адміністратор домену mgu.ru несе відповідальність за унікальність імен наступного рівня, включених у домен (s1, s2 і m).

У доменній системі імен розрізняють короткі імена, відносні імена і повні доменні імена. Коротке ім'я - ім'я кінцевого вузла мережі: хоста або порту маршрутизатора. Коротке ім'я - це лист дерева імен. Відносне ім'я - ім'я, яке починається з деякого рівня ієрархії, але не з самого верхнього. www1.zil - це відносне ім'я. Повне доменне ім'я (Fully Qualified Domain Name, FQDN) включає складові всіх рівнів ієрархії, починаючи від короткого імені і закінчуючи кореневою точкою: www1.zil.mmt.ru.

Кореневий домен управляється центральними органами Інтернету IANA і InterNIC. Домени верхнього рівня призначаються для кожної країни, а також для різних типів організацій. Імена цих доменів повинні слідувати міжнародному стандарту ISO 3166. Для позначення країн використовуються три або дволітерні абревіатури, наприклад ru (Росія), uk (Велика Британія), fi (Фінляндія), us (США), а для різних типів організацій: com - комерційні організації (microsoft.com); edu - освітні організації (mit.edu); gov - урядові

організації (nsf.gov); org - некомерційні організації (fidonet.org); net - мережеві організації (nsf.net).

Кожен домен адмініструє конкретна організація, яка зазвичай розбиває свій домен на піддомени і передає функції адміністрування цих піддоменів іншим організаціям.

Примітка. Комп'ютери входять в домен у відповідності зі своїми складовими іменами - вони можуть мати незалежні один від одного IP-адреси, що належать різним мережам і підмережам. Наприклад, в домен mgu.ru можуть входити хости з адресами 132.13.34.15, 201.22.100.33 і 14.0.0.6.

Кожен DNS-сервер окрім таблиці відображень імен містить посилання на DNS-сервери своїх піддоменів. Вони пов'язують окремі DNS-сервери в єдину службу DNS. Посилання являють собою IP-адреси відповідних серверів. Для обслуговування кореневого домену виділено декілька дублюючих один одного DNS-серверів.

Процедура дозволу DNS-імені аналогічна процедурі пошуку файлової системою адреси файлу по його символьному імені. Істотною відмінністю файлової системи від служби DNS є те, що перша розташована на одному комп'ютері, а друга є розподіленою.

Існує дві основні схеми дозволу DNS-імен:

У першому варіанті роботу з пошуку IP-адреси координує DNS-клієнт. DNS-клієнт звертається до кореневого DNS-сервера із зазначенням повного доменного імені. DNS-сервер відповідає клієнту, вказуючи адресу наступного DNS-сервера, який обслуговує домен верхнього рівня, заданий в наступній старшій частині імені, яке запитується. DNS-клієнт робить запит наступного DNS-сервера, який відсилає його до DNS-сервера потрібного піддомену і т.п., Поки не буде знайдений DNS-сервер, в якому зберігається відповідність імені, яке запитується IP-адресі. Цей сервер і дає остаточну відповідь клієнту.

Така процедура дозволу імені називається нерекурсивною, коли клієнт сам ітеративно виконує послідовність запитів до різних серверів імен. Це завантажує клієнта складною роботою тому застосовується рідко.

У другому варіанті реалізується рекурсивна процедура. DNS-клієнт запитує локальний DNS-сервер - сервер, який обслуговує піддомен, якому належить ім'я клієнта.

Далі можливі два варіанти дій:

– локальний DNS-сервер знає відповідь і відразу повертає значення клієнту (коли ви запросили ім'я входить в той же піддомен, що і ім'я клієнта або, коли сервер визначав відповідність для іншого клієнта і зберіг його в кеші);

– локальний сервер не знає відповідь і виконує ітеративні запити до кореневого сервера так само, як це робив клієнт в попередньому варіанті, а отримавши відповідь, передає її клієнту, який чекає її від свого локального DNS-сервера.

Записи DNS або ресурсні записи (Resource Records, RR) - одиниці зберігання і передачі інформації в DNS. Кожна ресурсна запис складається з наступних полів:

– ім'я (NAME) - доменне ім'я, до якого прив'язана або якому «належить» дана ресурсна запис;

– TTL (Time To Live) - допустимий час зберігання даної ресурсної записи в кеші невідповідального DNS-сервера;

– тип (TYPE) ресурсної записи - визначає формат і призначення даної ресурсної записи;

– клас (CLASS) ресурсної записи; теоретично вважається, що DNS може використовуватися не тільки з TCP / IP, але і з іншими типами мереж, код в поле клас визначає тип мережі;

– довжина поля даних (RDLEN);

– поле даних (RDATA) формат і зміст залежать від типу запису;

Найбільш важливі типи DNS-записів:

– запис A (address record) або запис адреси пов'язує ім'я хоста з IP-адресою. Наприклад, запит A-записи на ім'я referrals.icann.org поверне його IP адресу - 192.0.34.164;

– запис AAAA (IPv6 address record) пов'язує ім'я хоста з адресою протоколу IPv6. Наприклад, запит AAAA-запису на ім'я K.ROOT-SERVERS.NET поверне його IPv6 адресу - 2001: 7fd :: 1;

– запис CNAME (canonical name record) або канонічний запис імені (псевдонім) використовується для перенаправлення на інше ім'я;

– запис MX (mail exchange) або поштовий обмінник вказує сервер (и) обміну поштою для даного домену;

– запис NS (name server) вказує на DNS-сервер для домену;

– запис SOA (Start of Authority) або початковий запис зони вказує, на якому сервері зберігається еталонна інформація про домен, містить контактну інформацію особи, відповідальної за дану зону;

– запис PTR (pointer) або запис покажчика пов'язує IP хоста з його канонічним ім'ям. Запит в домені in-addr.arpa на IP хоста в reverse формі поверне ім'я (FQDN) даного хоста (див. Зворотний DNS-запит). Наприклад для IP адреси 192.0.34.164: запит записи PTR 164.34.0.192.in-addr.arpa поверне його канонічне ім'я referrals.icann.org. З метою зменшення обсягу небажаної кореспонденції (спаму) багато сервери-одержувачі електронної пошти можуть перевіряти наявність PTR запису для хоста, з якого відбувається відправка. В цьому випадку PTR запис для IP-адреси повинен відповідати імені відправляючого поштового сервера, яким він представляється в процесі SMTP сесії.

in-addr.arpa

DNS-запис in-addr.arpa виглядає так:

56.34.12.10.in-addr.arpa. IN PTR host1.example.net.

Це означає, що IP-адреса 10.12.34.56 відповідає імені хоста host1.example.net.

DNS-запис ip6.arpa виглядає так:

5.4.3.2.1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.8.b.d.0.1.0.0.2.ip6.arpa. IN PTR host1.example.net.

Це означає, що IPv6-адреса 2001: 0db8 :: 1: 2345 відповідає імені хоста host1.example.net.

2.3 Контрольні питання

1. Визначення та основні завдання DHCP.
2. Види повідомлень DHCP та їх значення.
3. Режими роботи DHCP.
4. Правила ієрархії доменних імен. Приклади формування імен.
5. Основні схеми дозволу DNS-імен. Кроки реалізації.
6. Навести приклади ресурсних записів.

3 МЕРЕЖІ VLAN

3.1 Загальні поняття

VLAN - логічна група вузлів мережі, трафік якої, включаючи і широкомовний, на канальному рівні повністю ізольований від інших вузлів мережі. Передача кадрів між різними VLAN на підставі MAC-адреси неможлива незалежно від типу адреси - унікального, групового або широкомовного. Переваги VLAN: надають ефективний спосіб групування користувачів в віртуальні групи, незважаючи на фізичне розміщення в мережі; забезпечують контроль широкомовних повідомлень, що збільшує смугу пропускання, доступну для користувача; дозволяють підвищити безпеку мережі, визначивши за допомогою фільтрів політику взаємодії користувачів з різних віртуальних мереж. Навести приклад ефективності використання логічної сегментації мереж за допомогою VLAN можна при вирішенні типової задачі організації доступу в Інтернет співробітникам офісу. Трафік кожного відділу є ізольованим. Кожна кімната це окрема робоча група з невеликою кількістю співробітників. При стандартному підході фізичної сегментації трафіку в кожну кімнату встановлюють окремий комутатор, який підключають до маршрутизатора. Маршрутизатор при цьому повинен мати достатню кількість портів для підключення фізичних сегментів мережі. Рішення погано масштабоване і дороге, при збільшенні кількості відділів збільшується кількість комутаторів, портів маршрутизатора і кабелю. При VLAN - не потрібно підключати користувачів одного відділу до окремого комутатора. Комутатор - виконує логічну сегментацію програмно, що дозволяє підключати користувачів з різних сегментів.

3.2 Типи VLAN

На комутаторах реалізовані наступні типи VLAN:

– на основі портів (Port-based VLAN) - порт комутатора призначається в VLAN і пристрій, підключений до порту, перебуває в призначеній VLAN;

– на основі MAC-адрес (MAC-based VLAN) - членство в VLAN засноване на MAC-адресі пристрою. На комутаторі створюється прив'язка MAC пристроїв до VLAN;

– на основі стандарту IEEE 802.1Q - поле про належність до VLAN інтегрується в кадр Ethernet. Перевага у сучасній функціональності та використання VLAN в межах всієї мережі; використання в мережі з обладнанням різних виробників. Цей тип VLAN використовується частіше інших;

– на основі IEEE802.1ad (Q-in-Q VLAN) - провайдерські мости;

– на основі портів і протоколів IEEE 802.1v - тип протоколу для членства в VLAN;

– асиметричні - клієнти різних VLAN взаємодіють з розподіленими пристроями (серверами), без підтримки 802.1Q, через один фізичний канал зв'язку з комутатором, не вимагаючи використання зовнішнього маршрутизатора.

VLAN на основі портів (Port-based VLAN) - кожен порт призначається в VLAN, незалежно який користувач або комп'ютер підключений до порту - всі вони члени цієї VLAN. Конфігурація портів статична - змінюється вручну. Основні характеристики:

– застосовуються в межах одного комутатора - організація декількох робочих груп (технічний відділ і відділ продажів);

– не вимагають від адміністратора великого обсягу ручної роботи - всім портам одного VLAN, привласнюють однаковий ідентифікатор VID (VLAN ID);

– дають можливість зміни логічної топології без фізичного переміщення станцій - зміна порту з однієї VLAN (технічного відділу) на іншу (відділу продажів), і ПК отримує можливість спільно використовувати ресурси нової VLAN - забезпечують гнучкість при переміщеннях і нарощуванні мережі;

– кожен порт може входити тільки в одну VLAN;

– об'єднання VLAN як всередині комутатора, так і між комутаторами через мережний рівень. Один з портів кожної VLAN підключається до інтерфейсу маршрутизатора для пересилання кадрів з однієї підмережі (VLAN) в іншу (IP-адреси підмереж різні).

Недолік - один порт кожної VLAN потрібно підключати до маршрутизатора. Призводить до додаткових витрат на покупку кабелів і маршрутизаторів.

VLAN на основі MAC-адрес включає багато ручних операцій по маркуванню MAC-адрес на кожному комутаторі. VLAN на базі MAC - дозволяють фізично переміщати ПК (включаючи до будь-якого порту комутатора), залишаючи ПК у тому ж широкомовному домені без змін в настройках конфігурації.

VLAN на основі стандарту IEEE 802.1Q використовують поля кадру для інформування про приналежність до VLAN при переміщенні по мережі. З точки зору гнучкості налаштувань, VLAN IEEE 802.1Q краще рішення. Його переваги:

– гнучкість і зручність при налаштуванні та внесенні змін, створення необхідних комбінацій VLAN в межах 1 комутатора і у всій мережі, побудованій на комутаторах з підтримкою 802.1Q - інформація про VLAN поширюється через безліч 802.1Q - сумісних комутаторів по одному фізичному з'єднанню;

– дозволяє активізувати алгоритм сполучного дерева на всіх портах і працювати в звичайному режимі. Це дозволяє автоматично визначати деревовидну конфігурацію зв'язків при довільному поєднанні портів між собою;

– здатність додавати і витягувати теги з заголовків кадрів дозволяє використовувати комутатори та мережеві пристрої, які не підтримують 802.1Q;

– пристрої різних виробників, що підтримують 802.1Q, можуть працювати разом, незалежно від фірмового рішення;

– для зв'язку підмереж на мережному рівні використовується маршрутизатор або комутатор рівня 3. Для організації доступу до сервера з різних VLAN, маршрутизатор не потрібно. Треба включити порт підключення серверу, в усі підмережі, а адаптер сервера повинен підтримувати IEEE 802.1Q.

Будь-який порт комутатора може бути налаштований як tagged (маркований) або як untagged (немаркований) - untagging дозволяє працювати з мережними пристроями VLAN, які не розуміють тегів у заголовку кадра Ethernet. Функція tagging дозволяє налаштовувати VLAN між декількома комутаторами, які підтримують IEEE 802.1Q.

Тег VLAN IEEE 802.1Q. До кадру Ethernet додані 32 біта, які збільшують розмір до 1522 байт. Перші 2 байта (поле Tag Protocol Identifier, TPID) з фіксованим значенням 0x8100 визначають, що кадр містить тег 802.1Q. Решта 2 байта містять:

– Priority (Пріоритет) - 3 біта поля пріоритету дозволяють кодувати до восьми рівнів пріоритету (7 - вищий), які використовуються в стандарті 802.1Q;

– Canonical Format Indicator (CFI) - 1 біт індикатора канонічного формату зарезервований для позначення кадрів мереж інших типів (Token Ring, FDDI), які передаються через Ethernet;

– VID (VLAN ID) - 12-бітний ідентифікатор визначає, якій VLAN належить трафік. Можна задати 4094 VLAN (VID 0 і 4095 зарезервовані).

Кожен фізичний порт комутатора має ідентифікатор VLAN (PVID) - для визначення в яку VLAN комутатор направить вхідний немаркований кадр з підключеного до порту сегмента коли кадр потрібно передати на інший порт (всередині комутатора в заголовки всіх немаркованих кадрів додається ідентифікатор VID, рівний PVID порту, на який вони прийняті). Цей механізм дозволяє одночасно існувати в одній мережі пристроїв з підтримкою і без підтримки стандарту IEEE 802.1Q. Комутатори, які підтримують стандарт 802.1Q зберігають таблицю, яка пов'язує PVID з ідентифікаторами VID мережі. Кожен порт комутатора має тільки один PVID і стільки VID, скільки підтримує дана модель комутатора. Правила вхідного трафіку (ingress rules) дозволяють класифікувати одержувані кадри щодо належності до VLAN. Правила просування між портами (forwarding rules) дозволяють приймати рішення про просування або відкидання кадру. Правила вихідного трафіку (egress rules) дозволяють приймати рішення про збереження або видалення в заголовку кадра тега 802.1Q перед його передачею.

3.3 Контрольні питання

1. Визначення та приклад організації VLAN.
2. VLAN на основі портів.
3. VLAN на основі MAC-адрес.
4. VLAN на основі стандарту IEEE 802.1Q.
5. Тег IEEE 802.1Q.

4 ТРАНСПОРТНИЙ РІВЕНЬ. ПРОТОКОЛИ

4.1 Порти додатків

Головне завдання транспортного рівня полягає в передачі даних між прикладними процесами. Його вирішують протокол управління передачею (Transmission Control Protocol, TCP) (RFC 793), і протокол користувацьких дейтаграм (User Datagram Protocol, UDP), описаний в RFC 768.

Протоколи TCP і UDP забезпечують інтерфейс з вище розташованим прикладним рівнем, передаючи дані, які надходять на вхідний інтерфейс хоста, відповідному додатку, використовуючи концепції «порт» і «сокет».

При доставці пакета на мережний інтерфейс одержувача, дані направляються конкретному процесу-одержувачу, а пакети, які надходять в мережу від різних додатків, обробляються загальним для них протоколом IP. Тому в стеці передбачено метод «збирання» пакетів від різних додатків для передачі IP протоколу мережного рівня. Прийом даних протоколами TCP і UDP, які надходять від різних прикладних служб - мультиплексування. Зворотна процедура - розподілу пакетів, які надходять від мережного рівня - демультиплексування (рис.4.1).

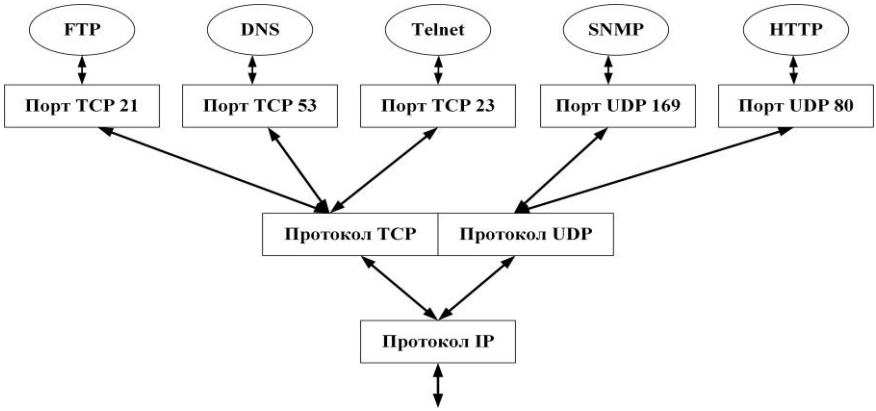

Рисунок 4.1 – Мультиплексування і демультиплексування на транспортному рівні

При виконанні прикладних процесів, маємо декілька точок входу - адреси призначення для пакетів даних.

Для кожної програми ведуться дві черги: черга пакетів, які надходять до даного додатку з мережі, і черга пакетів, які відправляються даним додатком в мережу. Пакети, що надходять на транспортний рівень, організовуються операційною системою (ОС) у вигляді безлічі черг до точок входу різних прикладних процесів. Системні черзі при цьому називаються портами (порти програми не плутати з портами обладнання). Вхідна і вихідна черги однієї програми розглядаються як один порт. Для ідентифікації портів їм присвоюють номери, які використовуються для адресації додатків.

За загальнодоступними службами (FTP, telnet, HTTP, TFTP, DNS) закріплюються стандартні, призначені номера - добре відомі (well-known) номера портів (стандарти RFC1700, RFC3232) - вони унікальні в межах Інтернет і призначаються додатків централізовано з діапазону від 0 до 1023. Номер 21 закріплений за службою віддаленого доступу до файлів FTP, а 23 - за службою віддаленого управління telnet. Для інших додатків номера портів призначаються розробниками додатків або ОС локально при надходженні запиту від програми. ОС веде список зайнятих і вільних номерів портів. Під час отримання запиту від програми, яка виконується на даному комп'ютері, ОС виділяє йому перший вільний номер, що має динамічне значення у діапазоні від 1024 до 65535.

Мережні додатки адресуються із зазначенням призначеного їм номера порту. По завершенні роботи, виділений з додатком локальний номер порту повертається в список вільних. Динамічні номера - унікальні в межах кожного комп'ютера, але звичайним є збіг номерів портів додатків, які виконуються на різних комп'ютерах. Додатки, що передають дані по протоколу UDP, отримують номери, під назвою UDP-порти. Додаткам, які звертаються до протоколу TCP, виділяються TCP-порти.

Якщо номери TCP- і UDP-портів збігаються, вони ідентифікують різні додатки. Одному з додатків може бути призначений TCP-порт 1750, іншому - UDP-порт 1750. Якщо стандартний додаток (наприклад DNS) звертається до протоколу TCP або UDP йому, для зручності, призначають однакові номери TCP та UDP-портів (53).

4.2 Протокол UDP

Протокол UDP - дейтаграмний протокол, реалізує сервіс по можливості, не гарантуючи доставку своїх повідомлень, не компенсує ненадійність протоколу IP. Одиниця даних протоколу UDP - UDP-дейтаграма яка переносить окреме призначене для користувача повідомлення, це призводить до обмеження: довжина дейтаграми UDP не може перевищувати довжини поля даних протоколу IP - обмеження на розмір кадру технології канального рівня. При переповненні UDP-буфера дані додатки відкидаються.

Заголовок UDP, складається з чотирьох 2-байтових полів, які включають порти відправника і одержувача, контрольну суму і довжину дейтаграми.

Приклад заголовка UDP: Source Port = 0x0035 Destination Port = 0x0411 Total length = 132 (0x84) bytes Checksum = 0x5333.

У UDP-дейтаграми в полі даних, довжина якого дорівнює (132 - 8) байт, вміщено повідомлення DNS-сервера - номер порту джерела (Source Port = 0x0035) з номером DNS - 53. UDP є простим протоколом. Функції зводяться до мультиплексування та демультиплексування даних між мережним і прикладним рівнями.

При вирішенні завдання демультиплексувания протоколом UDP кадри, які несуть UDP-дейтаграми, прибувають на мережний інтерфейс хоста та послідовно обробляються протоколами стека і надходять в розпорядження протоколу UDP. UDP, виконуючи демультиплексування, витягує з заголовка номер порту призначення і передає дані на порт, який відповідає додатку. Рішення непрацездатно в ситуації, коли на кінцевому вузлі виконується декілька копій одного і того ж додатку.

Наприклад, нехай запущені два DNS-сервера і обидва використовують для передачі своїх повідомлень протокол UDP з номером порта 53. У кожного DNS-сервера є свої клієнти, власні бази даних, власні налаштування. Коли на мережний інтерфейс комп'ютера прийде запит від DNS-клієнта та в UDP-дейтаграмі буде вказано номер порту 53, він в рівній мірі відноситься до обох DNS-серверів. Для зняття неоднозначності для різних копій одного додатку, який встановлено на одному комп'ютері, їм привласнюють різні IP-адреси.

Якщо DNS-сервер 1 має адресу IP1, то DNS2 буде мати адресу IP2. Однозначно прикладний процес в мережі буде визначати пара (IP-адреса, номер порту UDP), яка має назву UDP-сокету (socket).

Протокол UDP виконує демультиплексування на основі сокетів.

4.3 Протокол TCP

Інформація, яка надходить до протоколу TCP від протоколів прикладного рівня, розглядається як неструктурований потік байт. Для передачі на мережний рівень з буфера «вирізається» безперервна частина даних - сегмент (одиниця переданих даних (поле даних і заголовок TCP) або окремо поле даних) і забезпечується заголовком.

Заголовок TCP-сегмента містить більше полів, ніж UDP:

– порт джерела - 2 байта і ідентифікує процес-відправник;

– порт приймача - 2 байта і ідентифікує процес-одержувач;

– послідовний номер - 4 байта являє собою номер байта, який визначає зміщення сегмента щодо потоку даних, що відправляються (номер першого байта даних в сегменті);

– підтверджений номер - 4 байта. містить максимальний номер байта в отриманому сегменті, збільшений на 1. Використовується в якості квитанції. Якщо встановлено контрольний біт АСК, то це поле містить наступний номер черги, який відправник даного сегменту бажає отримати в зворотному напрямку;

– довжина заголовка (hlen) займає 4 біта - довжина заголовка TCP-сегмента, вимірюється в 32-бітових словах. Довжина заголовка не фіксована і може змінюватися в залежності від значень, встановлених в поле параметрів;

– резерв (reserved) займає 6 біт;

– кодові біти (code bits) займає 6 біт є службовою інформацією про тип даного сегмента. Позитивне значення сигналізується установкою бітів в одиницю:

1) URG - термінове повідомлення;

2) АСК - квитанція на прийнятий сегмент;

3) PSH - запит на відправку повідомлення без очікування заповнення буфера (протокол TCP може вичікувати заповнення

буфера перед відправкою сегмента, але якщо термінова передача, додаток повідомляє про це TCP за допомогою цього біта);

4) RST - запит на відновлення з'єднання;

5) SYN - синхронізація лічильників даних при встановленні з'єднання;

6) FIN - ознака досягнення передаючою стороною останнього байту в потоці переданих даних;

– вікно - 2 байта - кількість байт даних, очікуваних відправником даного сегмента, починаючи з байту, номер якого вказаний у полі підтвердженого номера;

– контрольна сума (checksum) займає 2 байта;

– покажчик терміновості (urgent pointer) займає 2 байта і вказує на кінець даних, які потрібно терміново прийняти, незважаючи на переповнення буфера;

– параметри (options) мають змінну довжину і можуть бути відсутніми;

– заповнювач (padding) може мати змінну довжину. Фіктивне поле, яке використовується для доведення розміру заголовка до цілого числа 32-бітових слів.

Відмінність TCP від UDP у забезпеченні надійної доставки повідомлень, використовуючи в якості основи ненадійний дейтаграмний протокол IP. Протокольні модулі TCP забезпечують надійний обмін даними шляхом встановлення логічних з'єднань. Завдяки ним TCP стежить, щоб передані сегменти не були втрачені або продубльовані. Логічне TCP-з'єднання однозначно ідентифікується парою сокетів.

4.4 Контрольні питання

1. Процедури мультиплексування та демультиплексування на транспортному рівні.
2. Структура UDP протоколу. Формат заголовку.
3. Структура TCP протоколу. Формат заголовку.
4. Порти та сокети. Визначення та приклад застосування.
5. Логічне TCP-з'єднання. Визначення та приклад застосування.

РЕКОМЕНДОВАНА ЛІТЕРАТУРА

1. Одом, Уэнделл. Официальное руководство Cisco по подготовке к сертификационным экзаменам CCENT/CCNA ICND1 100-101 / У. Одом. – акад. изд.: Пер. с англ. – М.; ООО “И. Д. Вильямс”, 2015. – 912 с. – ISBN 978-5-8459-1906-9.

2. Одом, Уэнделл. Официальное руководство Cisco по подготовке к сертификационным экзаменам CCNA ICND2 200-101: маршрутизация и коммутация / У. Одом. – акад. изд.: Пер. с англ. – М.; ООО “И. Д. Вильямс”, 2015. – 736 с. – ISBN 978-5-8459-1907-6.

3. Олифер В. Г. Компьютерные сети. Принципы, технологии, протоколы / В.Г. Олифер, Н.А.Олифер. // Учебник для вузов. – 5-е изд. – СПб.: Питер, 2016. – 992с.: ил.

4. Таненбаум Э. Компьютерные сети / Э. Таненбаум, Д.Уэзеролл. – 5-е изд. – СПб.: Питер, 2012. – 960 с.