

Передмова

Методичні вказівки призначені для студентів III курсу денної форми навчання. Мета виконання лабораторних робіт – закріплення теоретичного лекційного матеріалу та придбання практичних навичок у використанні мережного емулятора Cisco Packet Tracer 5.3.2 на базі устаткування компанії Cisco System, стандартних мережних утиліт ОС Windows, та розрахунків пропускну здатності і конфігурації локальних мереж. Для досягнення поставленої мети розглянуті основні принципи, методи та можливості технологій комп'ютерних мереж, до яких в першу чергу відносяться: топології мереж, методи фізичної та логічної структуризації за допомогою мережного комунікаційного обладнання, особливості адресації вузлів у мережі, багаторівнева система передачі даних, протоколи комп'ютерних мереж та ін. Методичні вказівки містять приклади конфігурування віртуальних машин мережевого устаткування (маршрутизаторів) компанії Cisco та приклади розрахунку конфігурації локальної мережі Ethernet, які можуть служити базою при виконанні аналогічних завдань лабораторних робіт.

Дисципліна «Комп'ютерні мережі» є однією з основних дисциплін формуючих спеціалістів з напряму підготовки комп'ютерні науки, яка розглядає моделі та методи побудови сучасних локальних і глобальних мереж. Дисципліна викладається у напрямі бакалаврської підготовки «Комп'ютерні науки» і відноситься до циклу професійної та практичної підготовки (цикл В).

Внаслідок вивчення дисципліни студент повинен:

знати: призначення основних мережних утиліт операційних систем сімейства Windows; етапи діагностики мережі; структуру та основні протоколи стека TCP/IP; основи адресації в IP – мережах; архітектури комп'ютерних мереж; принципи структурування та конфігурування мереж; методи передачі дискретних даних на фізичному і каналному рівнях; характеристики ліній зв'язку; принципи стандартизації в комп'ютерних мережах; технології Ethernet, Token Ring, FDDI локальних мереж.

вміти: аналізувати конфігурацію мережі на платформі ОС Windows; одержувати IP – адреси, ім'я домена, імена комп'ютерів, що входять у домен; переглядати і підключати загальні ресурси; визначати причини неполадок у мережі; працювати з емулятором IP-мереж Cisco Packet Tracer 5.3.2; конфігурувати та проводити налаштування комутаційного обладнання та маршрутизаторів Cisco за допомогою команд операційної системи Cisco IOS; розраховувати конфігурацію локальної мережі Ethernet відповідно її фізичного середовища.

Лабораторна робота № 1

Логічна організація комп'ютерних мереж. Робота з мережними утилітами.

1. Мета роботи

Метою лабораторної роботи є ознайомлення студентів з прийомами роботи із сервісними мережними утилітами для отримання даних про організацію мережі, її ресурси та для управління окремими ресурсами, а також ознайомлення з типами IP-адрес та правилами розрахунку VLSM (маски підмережі змінної довжини).

2. Завдання до лабораторної роботи

Лабораторна робота спирається на знання й уміння, отримані при вивченні наступних тем лекційного курсу: „Архітектура складеної мережі”, „Адресація в IP-мережах” і „Багаторівнева структура стека TCP/IP. Протокол IP.” Тому при підготовці до лабораторної роботи рекомендується повторити зазначені розділи дисципліни. Стислий конспект цього теоретичного матеріалу наводиться нижче.

2.1 Типи адрес стека TCP/IP

Кожний комп'ютер у мережі TCP/IP може мати адреси трьох рівнів:

- локальні, або апаратні, адреси, які використовуються для адресації вузлів у межах підмережі;
- мережні, або IP-адреси, які використовуються для однозначної ідентифікації вузлів у межах всієї складеної мережі;
- доменні імена, або DNS-імена - символічні ідентифікатори вузлів, до яких часто звертаються користувачі.

2.1.1 Адресація в протоколі IP. Визначення маски під мережі

IP-адреси використовуються для глобального з'єднання всіх вузлів і мереж в середині Інтернет. Кожному ПК підключеному до мережі необхідно присвоїти IP-адресу. Можливий також варіант під'єднання певної кількості ПК до мережі Інтернет без власного IP-адреса. Суть цих методів полягає у використанні проксі служб та трансляції мережних адрес.

Класи IP-адрес. Адресація (IPv4) припускає використання 32-бітного коду. IP-адресу прийнято записувати у вигляді чотирьох октетів (4 байт), у десятковій системі числення, наприклад:

IP-адреса **192.168.4.25** – це код **11000000 10101000 00000100 00011001**.

Кожне із значень, розділених точками – це 8-ми бітове число, яке може приймати значення від 0 до 255.

Будь-яка IP-адреса складається із двох частин: адреси мережі (**net**) та адреси хоста (**host**). Тут можна провести деяку аналогію з міжміськими телефонними номерами, у яких перші числа вказують на місто, де розміщується абонент, а інших – безпосередньо на самого абонента.

Для виділення з IP-адреси адреси мережі та адреси хоста (вузла) використовується мережева маска (**net mask**) – бітовий шаблон, в якому бітам, що використовуються для адреси мережі, присвоюються значення 1, а бітам адреси хоста – значення 0. Здійснюється це з використанням побітової логічної операції «І» (табл.2.1).

Таблиця 2.1 – Застосування маски

| Біт маски | Біт адреси | Біт результату |
|-----------|------------|----------------|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

Наприклад, маска мережі **255.255.255.0** (**11111111 11111111 11111111 00000000**) визначає, що поле адреси мережі містить 24 біта, а поле адреси хоста – 8 біт. Для наведеного вище прикладу адреси це означає: **192.168.4** – мережна частина (адресу мережі прийнято записувати **192.168.4.0**), а **25** – адреса хоста в цій мережі.

$$192.168.4.25_{10} = 11000000 \ 10101000 \ 00000100 \ 00011001_2$$

$$255.255.255.0_{10} = 11111111 \ 11111111 \ 11111111 \ 00000000_2$$

$$192.168.4.0_{10} = 11000000 \ 10101000 \ 00000100 \ 00000000_2$$

IP-адреса може належати до одного із класів. В залежності від класу адреса може містити різну кількість біт під адресу мережі та різну кількість біт під адресу для хоста. Перший байт (октет) вказує на приналежність IP-адреси до певного класу. Існує 5 класів IP-адрес (рис.2.1).

У таблиці 2.2 наведені діапазони номерів мереж, що відповідають кожному класу мереж. Адреси класу D використовуються для багатоадресної

передачі даних. Перший октет в них може містити значення від 224 до 239. В групових адресах поняття адреси мережі відсутнє. Призначення даних адрес полягає у обміні даними з декількома вузлами при використанні однієї адреси одержувача пакетів. Для того, щоби такий обмін міг відбутися, потрібно об'єднати ці вузли у групу та присвоїти їй групову IP-адресу.

| | | | | | | |
|--------|---|--------------|--------------|--------------|------------------------|----------------|
| Клас А | 0 | Номер мережі | Номер вузла | | | |
| Клас В | 1 | 0 | Номер мережі | Номер вузла | | |
| Клас С | 1 | 1 | 0 | Номер мережі | Номер вузла | |
| Клас D | 1 | 1 | 1 | 0 | Адреса групи multicast | |
| Клас E | 1 | 1 | 1 | 1 | 0 | Зарезервований |

Рис.2.1 – Структура різних класів IP-адрес

Таблиця 2.2 – Характеристики адрес різного класу

| Клас | Найменший номер мережі | Найбільший номер мережі | Максимальне число вузлів у мережі | Маска |
|------|------------------------|-------------------------|-----------------------------------|----------------|
| А | 1.0.0.0 | 126.255.255.255 | $2^{24}-2$ | 255.0. 0.0 |
| В | 128.0.0.0 | 191.255.255.255 | $2^{16}-2$ | 255. 255.0.0 |
| С | 192.0.0.0 | 223.255.255.255 | 2^8-2 | 255. 255.255.0 |
| D | 224.0.0.0 | 239.255.255.255 | групова адресація | |
| E | 240.0.0.0 | 247.255.255.255 | зарезервовано | |

Особливі IP-адреси. У протоколі IP існує кілька угод про особливу інтерпретацію IP-адрес (див. табл.2.3):

– якщо IP-адреса складається тільки із двійкових нулів, то вона позначає адресу того вузла, що згенерував цей пакет. Це спеціальна адреса що вказує на станцію, яка завантажується і яка не знає власного IP адреса. Дана адреса не може бути вказана в пакеті інформації в полі адреси отримувача;

– якщо в полі номера мережі стоять 0, то за замовчуванням вважається, що цей вузол належить тій же самій мережі, що й вузол, який відправив пакет;

– якщо всі двійкові розряди IP-адреси рівні 1, то пакет з такою адресою призначення повинен розсилатися всім вузлам, які перебувають у тій же мережі,

що й джерело цього пакета. Таке розсилання називається обмеженим ширококомовним повідомленням (*limited broadcast*);

– якщо в полі адреси хоста стоять всі 1, то пакет, що має таку адресу розсилається всім вузлам мережі із заданим номером. Таке розсилання називається ширококомовним повідомленням (*broadcast*). Приклад визначення ширококомовної адреси для IP-адреси 192.168.4.25 з маскою 255.255.255.0

$$\begin{aligned}
 192.168.4.25_{10} &= 11000000\ 10101000\ 00000100\ 00011001_2 \\
 !\ 255.255.255.0_{10} &= 00000000\ 00000000\ 00000000\ 11111111_2 \\
 \hline
 192.168.4.0_{10} &= 11000000\ 10101000\ 00000100\ 11111111_2
 \end{aligned}$$

– адреса 127.0.0.1 зарезервована для організації зворотного зв'язку при тестуванні роботи програмного забезпечення вузла без реального відправлення пакета по мережі. Ця адреса має назву *loopback*.

Форма групової IP-адреси – *multicast* – означає, що даний пакет повинен бути доставлений одразу декільком вузлам, які складають групу з номером, зазначеним у полі адреси. Вузли самі ідентифікують себе, тобто визначають, до якій із груп вони ставляться. Той самий вузол може входити в кілька груп. Такі повідомлення на відміну від ширококомовних називаються мультимовним.

Приватні діапазони адрес – використовуються в закритих мережах різного розміру. Використання даних адрес в мережі Інтернет – заборонено.

Таблиця 2.3 – Службові IP-адреси

| Адреси | Призначення |
|---|--|
| 0.0.0.0 | Обмежена адреса відправника |
| 255.255.255.255 | Обмежена ширококомовна адреса |
| X.X.X.255 | Мережева (чи підмережева) ширококомовна адреса |
| 127.X.X.X, де $0 \leq X \leq 255$ | Зарезервовано для програмного інтерфейсу loopback (lo) |
| 10.0.0.0/8 172.16.0.0/12 192.168.0.0/16 | Діапазони приватних (білих) IP адрес |

З ростом мережі Інтернет необхідність у додаткових IP-адресах зростає. Вже зараз тієї кількості IP-адрес, яка закладена в IP версії 4 не вистачає. Тому був розроблений протоколу IP версії 6.

Даний протокол має довжину не 32, а 128 бітів. Записується він у вигляді восьми шістнадцятирозрядних шістнадцяткових чисел, які розділені між собою двокрапками. Приклад:

1. 2100:0:0:0:4D:31AC:12:45
2. 0:0:0:0:0:0:0:1
3. AA0C:0:0:0:36:0:0:1
4. CDCE:0:0:0:FA:0:0:0
5. 3FA:2:17:1EF2:AD:CB:200:11

IP-адреса версії 6 складається з двох частин: адреси мережі, адреси хоста.

Використання підмереж (*subnetting*). Як вже згадувалося вище маска мережі визначає ту частину IP-адреси, яка відноситься до адреси мережі. Маску мережі можна змінювати, змінюючи таким чином, число октетів IP-адреси які відносяться до адреси мережі. На рис. 2.1 зображено мережу з адресою 10.0.0.0 та маскою 255.0.0.0. Дана мережа складається з певної кількості вузлів, кожен з яких фізично належить мережі 10.0.0.0 та формально об'єднаний з певними ПК (кожне об'єднання характеризується видом діяльності робочих станцій, які входять в групу). Мережа складається формально з трьох груп (відділ продажу, маркетингу та технічна група). Обмін даними здійснюється інтенсивно в межах групи і практично не здійснюється між групами. Тим не менше, всі ці групи підключені до єдиної мережі. Ріст числа вузлів такої мережі приводить до погіршення характеристик самої мережі (збільшується кількість колізій). Крім цього, інколи виникають проблеми безпеки при передачі даних в межах групи, оскільки всі ПК підключені до спільної шини.

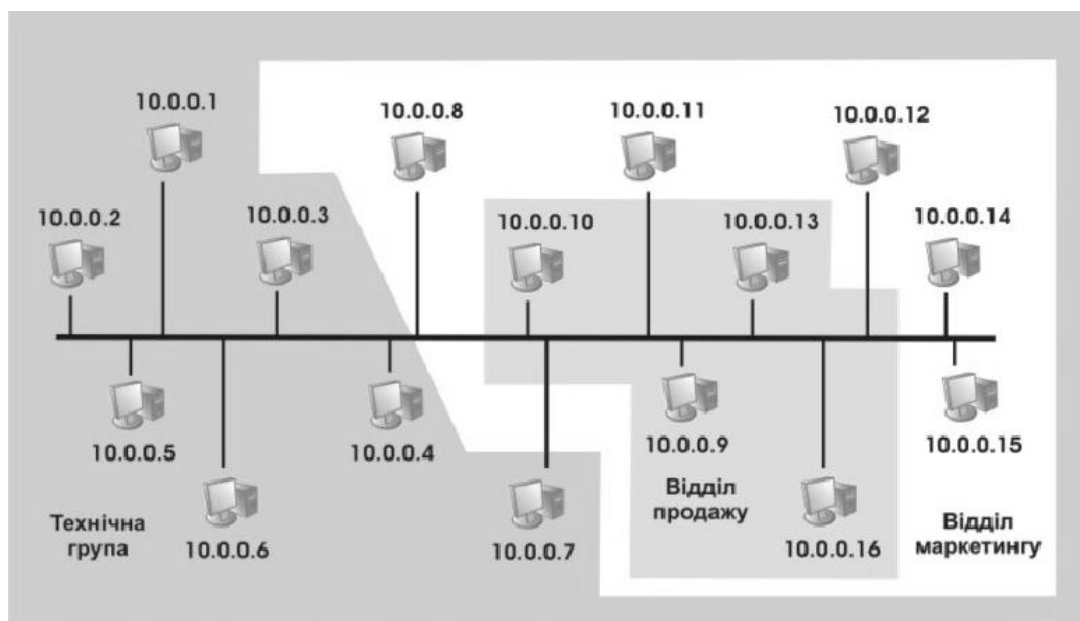


Рисунок 2.1 – Мережа, яку необхідно розділити на підмережі для покращення її характеристик

Для вирішення вище описаних проблем необхідно розбити мережу на окремі підмережі. Всі підмережі, що утворяться об'єднати маршрутизатором. Маршрутизатор необхідний для того щоби забезпечити обмін інформацією між підмережами. Також необхідно буде змінити маску підмереж. На рис. 2.1 зображено мережу, всі вузли якої використовують IP-адреси мережі 10.0.0.0 з маскою 255.0.0.0. Для розбиття мережі на декілька частин необхідно змінити маску. Таким чином ми отримуємо 2 октети для ідентифікації підмережі. Перший з октетів змінювати не можна. Другий октет буде ідентифікувати власне підмережу. Маска для кожної з підмереж буде мати вигляд 255.255.0.0. Відділ маркетингу матиме адресу 10.3.0.0, відділ продажу – 10.2.0.0, технічна група – 10.1.0.0 (рис.2.2).

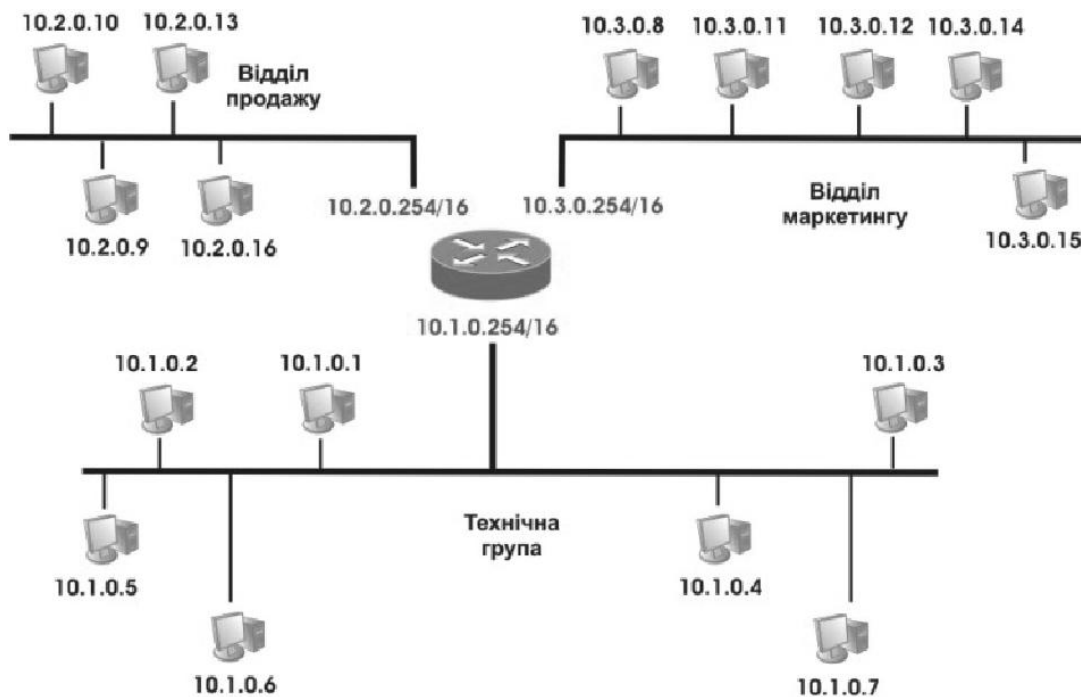


Рисунок 2.2 – Розбиття мережі на підмережі

Вище описаний приклад поділу мережі на окремі підмережі справедливий для адрес мереж класів А та В. Якщо необхідно поділити мережу класу С на підмережі, використовують маску підмережі змінної довжини (VLSM).

Розглянемо приклад поділу мережі на підмережі за допомогою розрахунку маски змінної довжини. Для виділення підмереж у мережі адміністраторові необхідно визначити кількість сегментів і кількість вузлів у кожному сегменті з урахуванням потреб мережі. Наприклад, якщо необхідно розбити мережу 192.168.8. 0 (блок містить 256 адрес) на 6 підмереж з максимальною кількістю вузлів 30 у кожній підмережі, те по-перше, потрібно визначити кількість біт у полі адреси підмережі (3 біти тому що $2^3=8$) і в полі

адреси вузла (5 біт тому що $2^5=32$). Максимальна кількість вузлів у підмережі дорівнює 30-ти, а не 32, тому що коди, що містять всі одиниці і всі нулі, не можуть бути адресою вузла. Визначивши поля адреси підмережі і вузла, запишемо маску підмережі:

11111111 11111111 11111111 11100000 – **255.255. 255. 224**.

Таблиця 2.4 – Адреси підмереж, отримані в результаті застосування маски підмережі

| Адреса мережі | Широкомовний адрес | Адреси хостів |
|------------------------|--------------------|------------------------------------|
| 192. 168.8.0 | 192. 168.8.31 | від 192.168.8.1 до 192.168.8.30 |
| 192. 168.8.32 | 192. 168.8.63 | від 192.168.8.33 до 192.168.8.62 |
| 192. 168.8.64 | 192. 168.8.95 | від 192.168.8.65 до 192.168.8.94 |
| 192. 168.8.96 | 192. 168.8.127 | від 192.168.8.97 до 192.168.8.126 |
| 192. 168.8. 128 | 192. 168.8. 159 | від 192.168.8.129 до 192.168.8.158 |
| 192. 168.8. 160 | 192. 168.8. 191 | від 192.168.8.161 до 192.168.8.190 |
| 192. 168.8. 192 | 192. 168.8. 223 | від 192.168.8.193 до 192.168.8.222 |
| 192. 168.8. 224 | 192. 168.8. 255 | від 192.168.8.225 до 192.168.8.254 |

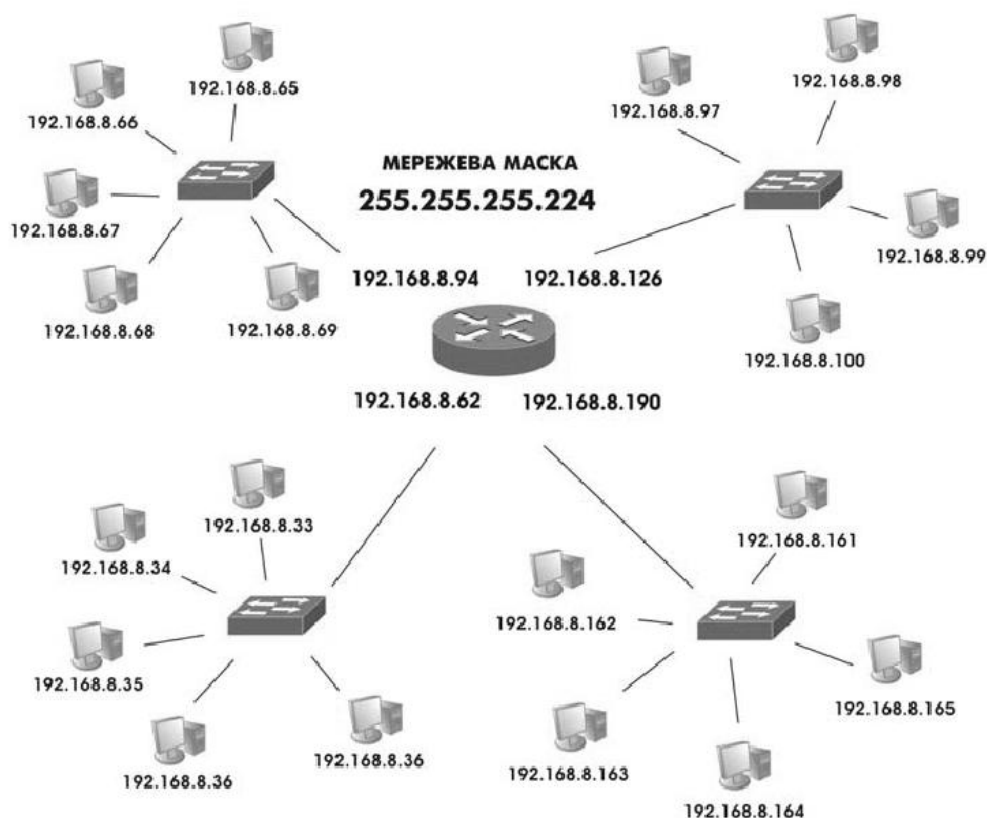


Рисунок 2.3 – Схема мережі 192.168.8.0/24

Призначення IP-адрес. IP-адреса призначається адміністратором під час конфігурування комп'ютерів і маршрутизаторів, при цьому номер мережі може

бути обраний адміністратором довільно, або призначений організацією ICANN. Для отримання власної IP-адреси, необхідно звернутися до Інтернет-провайдера, або на сайт організації ICANN <http://www.icann.org>.

У великих мережах підтримується автоматичний розподіл адрес на основі протоколу *Dynamic Host Configuration Protocol (DHCP)*. Протокол DHCP працює відповідно до моделі клієнт-сервер. Комп'ютер, що є DHCP-клієнтом, посилає в мережу ширококомовний запит на одержання IP-адреси. DHCP-сервер відгукується і посилає повідомлення відповідь, що містить IP-адресу з діапазону вільних для розподілу адрес. Передбачається, що DHCP-клієнт і DHCP-сервер перебувають в одній IP-мережі.

2.1.2 Протоколи перетворення адрес

Фізична адреса. Фізична, або апаратна адреса вузла, визначається технологією, за допомогою якої побудована мережа, до якої входить даний вузол. Для вузлів, що входять у локальні мережі - це MAC-адреса мережевого адаптера або порту маршрутизатора, наприклад, 11-A0-17-3D-BC-01. Ці адреси призначаються виробниками устаткування і є унікальними адресами, тому що управляються централізовано. Для всіх існуючих технологій локальних мереж MAC-адреса має формат 6 байтів: старші 3 байти - ідентифікатор фірми виробника, а молодші 3 байти призначаються унікальним чином самим виробником.

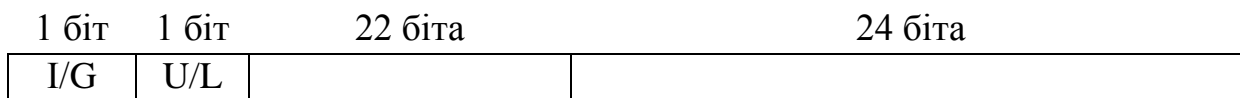


Рисунок 2.4 - Структура MAC-адреси

Крайній лівий біт числа називається ознакою *індивідуальної* або *групової* адреси (I/G). Якщо біт дорівнює 0, то інші біти визначають індивідуальну адресу; значення 1 вказує на те, що інші біти визначають групову адресу. Якщо другий біт (U/L) дорівнює 0, то адреса підмережі є *універсальною*, тобто призначеною комітетом IEEE, у противному випадку адреса є *локальною*.

Протокол ARP. Щоб відправити дейтаграму з одного комп'ютера на інший у локальній або глобальній мережі, відправник повинен знати фізичну адресу одержувача. Повинен існувати механізм перетворення IP-адрес, які задаються додатками, у фізичні адреси устаткування, що з'єднують комп'ютери з мережею. Для рішення цієї проблеми був розроблений протокол перетворення адрес *ARP (Address Resolution Protocol)*. ARP веде таблицю відповідності між IP-адресами і фізичними адресами, яка називається *таблицею APR*. Крім того, ARP підтримує кеш записів - *кеш ARP*.

Алгоритм роботи протоколу ARP:

1. Звичайно пошук починається з кеша ARP, і тільки у випадку невдачі дані шукаються в таблиці ARP. Записи кеша ARP, які були динамічно згенеровані, стають недійсними при закінченні інтервалу тайм-ауту, тоді як на статичні записи тайм-аут не повинен поширюватися. Знищення статичних записів у результаті тайм-ауту є ознакою псування даних у кеші ARP.

2. Якщо в записах таблиці необхідна IP-адреса не знайдена, то вихідний IP-пакет запам'ятовується в буфері, а протокол ARP формує запит (ARP запит), вкладає його в кадр протоколу канального рівня і розсилає ширококомовно.

3. Всі інтерфейси підмережі одержують ARP-запити і порівнюють зазначену там адресу з власною. При збігу вузол чи маршрутизатор формує ARP-відповідь, указуючи в ньому свої IP і MAC адреси та відправляє його по IP-адресі вузла відправника ARP-запиту.

Структура ARP-запита наведена на рис. 2.5. Поле «Тип протоколу» дозволяє використовувати протокол ARP не тільки для протоколу IP, але й для інших мережевих протоколів. У полі коду операції для ARP-запитів вказується значення 1, якщо це запит, і 2, якщо це відповідь.

| | |
|-----------------------------|--------------------------|
| Тип мережі (16 біт) | |
| Тип протоколу (16 біт) | |
| Довжина апаратної адреси | Довжина мережевої адреси |
| Код операції (16 біт) | |
| Апаратна адреса відправника | |
| IP-адреса відправника | |
| Апаратна адреса одержувача | |
| IP-адреса одержувача | |

Рисунок 2.5 – Структура запитів і відповідей ARP

4. Якщо в мережі немає машини із шуканою IP-адресою, то ARP-відповіді не буде. Протокол IP знищує IP-пакети, спрямовані по цій адресі.

5. Якщо відповідність знайдена, то вона записується в ARP-таблицю відповідного інтерфейсу. Новий запис в ARP-таблиці з'являється автоматично, через декілька мілісекунд після того, як модуль ARP проаналізував ARP-відповідь. Крім динамічних записів, побудованих на підставі даних ширококомовних розсилок, ARP-таблиці можуть містити статичні записи, які створюються вручну за допомогою утиліти `arp` і не мають строку старіння по тайм-ауту.

Нижче наведений синтаксис команди `arp`.

```
ARP -s inet_addr eth_addr [if_addr]
ARP -d inet_addr [if_addr]
ARP -a [inet_addr] [-N if_addr]
```

`-a` Відображає поточні ARP-записи, опитуючи поточні дані протоколу. Якщо задано `inet_addr`, то будуть відображені IP і фізична адреса тільки для заданого комп'ютера. Якщо більше одного мережевого інтерфейсу використовують ARP, то будуть відображатися записи для кожної таблиці.

`-g` Теж саме, що й ключ `-a`.

`inet_addr` Визначає IP-адресу.

`-N if_addr` Визначає ARP-записи для заданого в `if_addr` мережевого інтерфейсу.

`-d` Видаляє вузол, що задає `inet_addr`. `inet_addr` може містити символ шаблону `*` для видалення всіх вузлів.

`-s` Додає вузол і зв'язує internet адресу `inet_addr` з фізичною адресою `eth_addr`. Фізична адреса задається 6 байтами (в шістнадцятиричному вигляді), розділеним дефісом. Цей зв'язок є постійним.

`eth_addr` Визначає фізичну адресу.

`if_addr` Якщо параметр заданий, він визначає інтернет-адресу інтерфейсу, чия таблиця перетворення адрес повинна змінитися. Якщо не заданий, - буде використано перший доступний інтерфейс.

Приклад:

Додати статичний запис

```
> arp -s 157.55.85.212 00-aa-00-62-c6-09
```

Використання із ключем `-a` дозволяє побачити поточний стан кеша ARP

```
> arp -a
```

Доменні імена. Доменне, або символічне ім'я, наприклад, `SERV1.IBM.COM` – адреса, що призначається адміністратором і складається з декількох частин, наприклад, імені машини, імені організації, імені домена. Така адреса, названа також DNS-ім'ям, використовується на прикладному рівні, наприклад, у протоколах FTP або telnet. Ієрархія доменних імен аналогічна ієрархії імен файлів, однак запис доменного імені починається із наймолодшої складової, а закінчується найстаршою. Наприклад, в імені `partnering.microsoft.com` складова `partnering` є ім'ям одного з комп'ютерів у домені `microsoft.com`. Сукупність імен, у яких кілька старших складових частин збігаються, утворюють домен (domain) імен. Наприклад, імена `www.chip.kiev.ua`, `www.itc.kiev.ua` і `www.infocity.kiev.ua` входять у домен `kiev.ua`.

Файл HOSTS і служба DNS. Відповідність між доменними іменами і IP-адресами може встановлюватися як засобами локального хосту, так і засобами централізованої служби. У першому випадку, вручну підтримується файл HOSTS, що містить інформацію про те, яка IP-адреса зв'язується з тим або іншим символічним ім'ям. Коли вузлу необхідно знайти інший вузол у мережі, він звертається до локального файлу HOSTS. Синтаксис записів HOSTS можна подивитися у файлі hosts.sam .

У другому випадку використовується спеціальна служба - система доменних імен (Domain Name System, DNS), яка заснована на розподіленій базі відображень «доменне ім'я - IP-адреса». Служба DNS використовує протокол типу «клієнт-сервер». У ньому визначені DNS-сервери і DNS-клієнти, які звертаються до серверів із запитом про дозвіл доменних імен в IP-адресу. Для кожного домена імен створюється свій DNS-сервер. Кожний DNS-сервер крім таблиці відображень імен містить посилання на DNS-сервери своїх піддоменів. Ці посилання зв'язують окремі DNS-сервери в єдину службу DNS.

Мережене програмне забезпечення комп'ютера настроєне таким чином, щоб воно переглядало локальний файл HOSTS перед тим, як звертатися до сервера DNS, тому що вибірка інформації з файлу відбувається набагато швидше, ніж пошук засобами DNS.

Дозвіл імен в NetBIOS. Для спілкування в мережах з вузлами, керованими ОС Windows, вживаються імена комп'ютерів, які визначаються по системі NetBIOS. Варто пам'ятати, що це не теж саме, що DNS імена хостів в IP-мережах.

Ім'я NetBIOS привласнюється комп'ютеру при установці операційної системи, але може бути змінено пізніше. Воно повинне бути унікальним у межах мережі. Щоб визначити ім'я комп'ютера потрібно викликати діалогове вікно «Мережа», клацнувши правою кнопкою миші на значку «Мережеве оточення» і вибравши в контекстному меню команду «Властивість» і вкладку «Ідентифікація». Ім'я хоста настроюється у властивостях протоколу TCP/IP. За замовчуванням ім'я хоста збігається з ім'ям комп'ютера в NetBIOS, тому що це полегшує процес діагностики.

В операційних системах Microsoft використовуються наступні способи перетворення імен NetBIOS в IP-адреси:

– **Кеш імен.** Щоб переглянути вміст кеша імен, введіть команду NBTSTAT-c у режимі командного рядка мережевої операційної системи Microsoft, що використовує TCP/IP.

– **Сервер WINS** - підтримує базу даних з інформацією про зв'язки імен комп'ютерів з IP-адресами. Щоб перетворити ім'я комп'ютера в IP-адресу, клієнт звертається до сервера WINS із запитом.

– **Широкомовне розсилання** – клієнти мережі Microsoft можуть розіслати запит у локальному сегменті мережі, щоб довідатися, чи належить дозволене ім'я цьому сегменту.

– **Файл LMHOSTS** – статичний файл, що містить список IP-адрес із відповідними їм іменами комп'ютерів. Синтаксис записів можна подивитися у файлі lmhosts.sam (sample).

Черговість, у якій клієнтська система намагається дозволити ім'я NetBIOS, залежить від типу вузла NetBIOS. Існують чотири типи вузлів NetBIOS:

1. В-вузол - широкомовні вузли (Broadcast). Дозвіл імен NetBIOS здійснюється тільки за допомогою розсилання запитів у локальному сегменті. Розширені В-вузли шукають інформацію у файлі LMHOSTS, якщо ім'я не було знайдено в локальному сегменті.

2. Р-вузол - дозвіл імен здійснюється крапковим (Point-to-point) звертанням до сервера WINS, широкомовне розсилання в локальному сегменті не використовується.

3. М-вузол - змішані (Mixed) системи спочатку розсилають запит у локальному сегменті, а потім звертаються за дозволом імені до сервера WINS.

4. Н-вузол - гібридні (Hybrid) системи спочатку звертаються за дозволом імені до сервера WINS. Якщо одержати відповідь не вдається, Н-вузол розсилає запит у локальному сегменті.

Всі клієнти NetBIOS спочатку перевіряють зміст кеша імен. За замовчуванням використовуються типи вузлів В и Н. Розширені В-вузли використовуються за замовчуванням для всіх комп'ютерів, на яких не задана адреса сервера WINS. При наявності адреси сервера WINS за замовчуванням використовується Н-вузол.

2.2 Службові файли TCP/IP

Крім файлів HOSTS і LMHOSTS в операційній системі Windows 2000 використовуються ще три файли - NETWORKS, PROTOCOL і SERVICES.

Файл NETWORKS. Файл NETWORKS використовується для ідентифікації мереж, що входять в об'єднану мережу (тобто мережа, що складається з декількох мереж, звичайно зв'язаних через маршрутизатори). У файлі зберігається інформація про відповідність між іменами мереж (ідентифікаторами, що представляють дану мережу) і номерами мереж (мережевими частинами IP-адрес цих мереж). Символьне ім'я мережі не може містити пробілів, символів табуляції й знаків # і повинне бути унікальним в межах файлу NETWORKS. Імена мереж, зазначених у файлі NETWORKS, використовуються в конфігураційних утилітах і командах - наприклад, ім'я мережі може вказуватися замість мережевої адреси. Файл NETWORKS

звичайно редагується мережевими адміністраторами, щоб замість мережесих адрес, що важко запам'ятовуються, у командах і утилітах використовувалися більш зручні символічні імена.

Файл PROTOCOL. Файл PROTOCOL призначений для ідентифікації імен протоколів і відповідних їм номерів. Номер протоколу в сімействі протоколів Інтернету збігається зі значенням ідентифікатора протоколу в заголовку IP.

Файл PROTOCOL доповнює модуль TCP/IP і містить визначення основних протоколів; не змінюйте його вміст без крайньої потреби.

Файл SERVICES. У файлі SERVICES визначаються назви служб і використовувані ними транспортні протоколи й номери портів.

Служби являють собою програми, що працюють на прикладному рівні моделі TCP/IP, - Telnet, FTP, SMTP, SNMP і т.д. Кожна служба працює на базі певного транспортного протоколу (TCP і UDP). Інформація про те, який транспортний протокол використовується тією або іншою службою, зберігається в конфігураційному файлі SERVICES. Деякі служби доступні як через TCP, так і через UDP. У цьому випадку служба представлена у файлі двома записами: для TCP і для UDP.

Файл SERVICES доповнює модуль TCP/IP і містить визначення основних служб TCP/IP; не змінюйте його вміст без крайньої потреби.

2.3 Програми командного рядка TCP/IP

Утиліта hostname. Виводить ім'я локального комп'ютера (хоста). Вона доступна тільки після установки підтримки протоколу TCP/IP. Приклад виклику команди hostname :

```
C:\>hostname  
ws 327b103
```

Утиліта ipconfig. Виводить діагностичну інформацію про конфігурації мережі TCP/IP та дозволяє переглянути поточну конфігурацію IP-адрес комп'ютерів мережі. Синтаксис утиліти ipconfig:

```
ipconfig [/all | /renew [адаптер ] | /release [адаптер ]],
```

де all - виводить відомості про ім'я хоста, DNS (Domain Name Service), тип вузла, IP-маршрутизації та ін. Без цього параметра команда ipconfig виводить тільки IP-адреси, маску підмережі і основний шлюз;

/renew [адаптер] - оновлює параметри конфігурації DHCP. Ця можливість доступна тільки на комп'ютерах, де запущені служби клієнта DHCP. Для завдання адаптера використовується ім'я, виведене командою ipconfig без параметрів;

`/release [адаптер]` - очищає поточну конфігурацію DHCP. Ця можливість відключає TCP/IP на локальних комп'ютерах і доступна тільки на клієнтах DHCP. Для завдання адаптера використовується ім'я, виведене командою `ipconfig` без параметрів. Ця команда часто використовується перед переміщенням комп'ютера в іншу мережу. Після використання утиліти `ipconfig/release`, IP-адреса стає доступна для призначення іншому комп'ютеру.

Запущена без параметрів, команда `ipconfig` виводить повну конфігурацію TCP/IP, включаючи IP адреси й маску підмережі.

Утиліта ping. Утиліта `ping` (Packet Internet Groper) перевіряє з'єднання з віддаленим комп'ютером або комп'ютерами. Вхідними даними для утиліти `ping` є адреса вузла, маршрут до якого підлягає трасуванню. Адреса вузла задається у вигляді IP-адреси або доменної адреси в командному рядку при запуску утиліти. Запити утиліти `ping` передаються по протоколу ICMP (Internet Control Message Protocol). Одержавши такий запит, програмне забезпечення, що реалізує протокол IP в адресата, негайно посилає луну-відповідь. Луни-запити посилають задану кількість разів (ключ `-n`) або за замовчуванням доти, поки користувач не введе команду переривання (`Ctrl+C` або `Del`), після чого виводяться статистичні дані.

Формат команди:

`ping [-t] [-a] [-n число] [-l розмір] [-f] [-i TTL] [-v TOS] [-r число] [-s число] [-j перелік_вузлів] | [-k перелік_вузлів]] [-w інтервал] перелік_розсилки,`

Таблиця 2.5 - Параметри утиліти `ping`

| Ключі | Функції |
|--------|--|
| -t | Відправка пакетів на вказаний вузол до команди переривання. Перегляд статистики та продовження - Control-Break; Закінчення- Control-C. |
| -a | Визначення адресів по іменам вузлів. |
| -n | Число відправляємих запитів. За замовчуванням - 4 |
| -l | Розмір буфера відправки. За замовчуванням – 32 байта, максимум – 65527 |
| -f | Установлення прапору, який забороняє фрагментацію пакету. |
| -i TTL | Задання часу життя пакета(поле "Time To Live"). |
| -v TOS | Задання типу служби(поле "Type Of Service"). |
| -r | Запис маршруту для вказанного числа переходів. |

| Ключі | Функції |
|-------------------|---|
| -s | Штамп часу для вказанного числа переходів. |
| -j перелік вузлів | Вільний вибір маршруту по переліку вузлів. |
| -k перелік вузлів | Жорсткий вибір маршруту по переліку вузлів. |
| -w інтервал | Інтервал очікування кожної відповіді в мілісекундах |
| перелік_розсилки | Список комп'ютерів, яким спрямовуються запити |

На практиці більшість опцій у форматі команди можна опустити, тоді в командному рядку може бути: `ping ім'я вузла`. Зверніть увагу на те, що максимальне значення TTL за замовчуванням приймається рівним 255 вузлів. Отже, щоб визначити кількість вузлів, через які пройшов пакет, треба від 255 відняти отримане значення TTL.

Утиліта `tracert`. Утиліта `tracert` використовується для визначення маршруту пакета, який прямує до вказаного вузла. Утиліта передає ряд даяграм і очікує відповіді на кожну з них. Перед відправленням першої даяграми, значення TTL для неї встановлюється в 1. Перший маршрутизатор, що виявиться на шляху проходження цієї даяграми, зменшить значення TTL на одиницю та, якщо це значення стане рівним 0, поверне помилку ICMP про закінчення TTL. Оскільки повідомлення ICMP передається також у вигляді даяграми IP, то `tracert` може витягти IP-адресу джерела і вивести на екран адресу маршрутизатора. Для наступної даяграми значення TTL буде збільшено на одиницю й т.д., поки не буде отриманий запит від комп'ютера призначення.

Параметри команди наведені нижче:

`tracert [-d] [-h макс_число] [-j перелік_вузлів] [-w інтервал] ім'я`

Таблиця 2.6 - Параметри утиліти `tracert`

| Ключі | Функції |
|-------------------|---|
| -d | Без визначення адреси по іменам вузлів. Використовується для відключення визначення dns-імен по IP-адресах маршрутизаторів. |
| -h макс_число | Максимальне число переходів при пошуку вузла. |
| -j перелік_вузлів | Вільний вибір маршруту за переліком вузлів. |
| -w інтервал | Інтервал очікування кожної відповіді в мілісекундах. |

2.4 Контрольні питання

- 1) Які три типи адреси хоста в мережі TCP/IP ви знаєте?
- 2) Пояснити поняття loopback, broadcast, multicast.
- 3) Що відбудеться при відправленні пакета за адресою 127.0.0.1?
- 4) Яку частку всієї безлічі IP-адрес становлять адреси класу А? Класу В? Класу С?
- 5) Дайте визначення маски підмережі? У яких цілях вона використовується?
- 6) Яким способом може відбуватися дозвіл адрес? Приведіть приклади протоколів дозволу адрес.
- 7) Перелічіть стандартні службові файли TCP/IP. Для чого вони призначені?
- 8) Які функції виконують системні утиліти ping, tracert, ipconfig? Яке призначення має кожний параметр програм?
- 9) Для чого необхідна утиліта hostname?
- 10) Навіщо використовується параметр all в утиліті ipconfig?

2.5 Варіанти завдання

Таблиця 2.7 – Варіанти завдання до лабораторної роботи

| Варіант | IP-адреса | Маска | Завдання |
|---------|---------------|-------|-------------------------------|
| 1 | 194.138.33.0 | /24 | Розбити мережу на 4 підмережі |
| 2 | 192.168.45.0 | /24 | Розбити мережу на 4 підмережі |
| 3 | 82.207.118.0 | /24 | Розбити мережу на 3 підмережі |
| 4 | 113.45.25.0 | /24 | Розбити мережу на 2 підмережі |
| 5 | 164.34.24.0 | /24 | Розбити мережу на 6 підмереж |
| 6 | 155.150.100.0 | /24 | Розбити мережу на 3 підмережі |
| 7 | 164.90.34.0 | /24 | Розбити мережу на 8 підмереж |
| 8 | 197.230.100.0 | /24 | Розбити мережу на 10 підмереж |
| 9 | 87.217.118.0 | /24 | Розбити мережу на 12 підмереж |
| 10 | 182.207.120.0 | /24 | Розбити мережу на 10 підмереж |
| 11 | 178.18.25.0 | /24 | Розбити мережу на 8 підмереж |
| 12 | 112.54.12.0 | /24 | Розбити мережу на 6 підмереж |

2.6 Перелік літератури

1.Олифер В.Г. Компьютерные сети. Принципы, технологии, протоколы: учебник для вузов В.Г. Олифер, Н.А. Олифер. 4-е изд. СПб.: Питер, 2010. 944 с.: ил.

2.Коломоец Г.П. Организация компьютерных сетей: учебное пособие. Запорожье: КПУ, 2012. □ 156 с.

3.Кравец, О.Я. Практикум по вычислительным сетям и телекоммуникациям : учебное пособие / О. Я. Кравец. – Изд. 2-е, перераб. и доп. – Воронеж: Научная книга, 2006. – 156 с.

4.Новиков Ю.В. Основы локальных сетей: курс лекций : учеб. пособие: для студентов вузов, обучающихся по специальностям в обл. информ. технологий . М.: Интернет-ун-т информ. технологий, 2005. □ 360 с.

5. Смирнова Е.В., Пролетарский А.В., И.В. Баскаков, Р.А. Федотов Построение коммутируемых компьютерных сетей: учебное пособие / Е.В. Смирнова и др. – М.: Национальный Открытый Университет «ИНТУИТ»: БИНОМ. Лаборатория знаний, 2011. – 367 с.: ил.

3. Прилади, устаткування та інструменти

Для виконання лабораторної роботи використовуються ПЕОМ, об'єднані в локальну мережу. У якості інструмента для збереження результатів роботи мережевих утиліт може використовуватись будь-який текстовий редактор.

4. Правила техніки безпеки та охорони праці

Правила техніки безпеки при виконанні лабораторної роботи регламентуються: «Правилами техніки безпеки при роботі в комп'ютерній лабораторії».

5. Порядок проведення лабораторної роботи

При проведенні роботи студенти об'єднуються в бригади по дві особи. Для виконання роботи кожен повинен:

1. Відповісти на контрольні питання та пройти усне опитування за теоретичним матеріалом лабораторної роботи, який викладається в п.2;
2. Пройти інструктаж по правилам охорони праці;
3. Отримати варіант завдання у викладача;
4. Запустити комп'ютер, переглянути і занотувати усі функції та

параметри програм ping, tracert, ipconfig.

5. Визначити ім'я локального комп'ютера. Вивести інформацію про конфігурації мережі TCP/IP за допомогою утиліти ipconfig. Проаналізувати отримані результати. Занотувати результати роботи програми та висновки щодо поточної конфігурації мережі;

6. За допомогою команди ping перевірити стан зв'язку з вузлами, зазначеними викладачем. Переглянути маршрути проходження пакетів до даних вузлів мережі за допомогою утиліти tracert;

7. Переглянути стан ARP-кеша за допомогою утиліти arp;

8. Переглянути вміст службових файлів TCP/IP;

9. Розрахувати кількість підмереж згідно варіанту завдання (табл.2.7) і побудувати логічну топологію мережі.

10. Проаналізувати отримані результати;

11. Підготувати і захистити звіт до лабораторної роботи.

6. Оформлення і захист звіту

Підготовлений до захисту звіт до лабораторної роботи повинен містити:

1. Найменування лабораторної роботи.

2. Відомості про виконавця, номер варіанту.

3. Перелік функцій та параметрів утиліт ping, tracert, ipconfig з поясненнями, які виводяться довідковою підсистемою.

4. Перелік IP-адрес та доменних імен локального хоста і вузлів, зазначених викладачем.

5. Результати запуску всіх зазначених у завданні утиліт у мережі (повідомлення утиліт при їх роботі як з усіма окремими параметрами так і з усіма припустимими комбінаціями параметрів).

6. Вміст ARP-кеша і службових файлів TCP/IP.

7. Таблиця розрахунку кількості підмереж згідно варіанту завдання (табл.2.7) і логічна топологія мережі.

8. Висновки за результатами роботи.

Лабораторна робота № 2

Устаткування локальних мереж. Знайомство з програмним емулятором Cisco Packet Tracer

1. Мета роботи

Здобути практичні навички роботи з мережною операційною системою комутаційного обладнання та маршрутизаторів Cisco IOS.

2. Завдання до лабораторної роботи

Лабораторна робота спирається на знання й уміння, отримані при вивченні наступних тем лекційного курсу: „Фізична і логічна структуризація мережі за допомогою різних типів комунікаційного обладнання” і „Принципи роботи комутаторів”. Тому при підготовці до лабораторної роботи рекомендується повторити зазначені розділи дисципліни. Стислий конспект цього теоретичного матеріалу наводиться нижче.

2.1 Компоненти устаткування Cisco

Склад внутрішніх компонентів Cisco в певній мірі залежить від призначення устаткування, потужності блоку живлення, конструкції та складу модулів. Всі устаткування практично завжди мають деякі основні компоненти. Зокрема, будь-який маршрутизатор або комутатор можна розглядати як спеціалізований комп'ютер, в якому аналогічні компоненти можна використовувати для тієї ж мети. Устаткування Cisco можуть включати не тільки внутрішні компоненти, але і зовнішні, склад яких також залежить від моделі устаткування.

Внутрішні компоненти. До числа найбільш використовуваних компонентів відносяться модулі оперативної пам'яті (флеш-пам'ять, ПЗП), процесор, об'єднувальна плата и енергонезалежний ОЗП (рис.2.1).

Оперативна пам'ять. Моделі DRAM (Dynamic RAM – динамічний ОЗП) застосовуються в устаткуваннях Cisco з тією ж метою, що і в персональному комп'ютері: в якості оперативної пам'яті. Оперативна пам'ять в маршрутизаторах Cisco має наступні характеристики:

- енергозалежна;
- пересписувана;
- об'єм від 16 до 512 Мбайт;
- і функції:

- зберігає таблицю маршрутизації (routing table);
- містить ARP кеш;
- буферизує пакети;
- під час роботи маршрутизатора містить файл робочої конфігурації (running-config file).

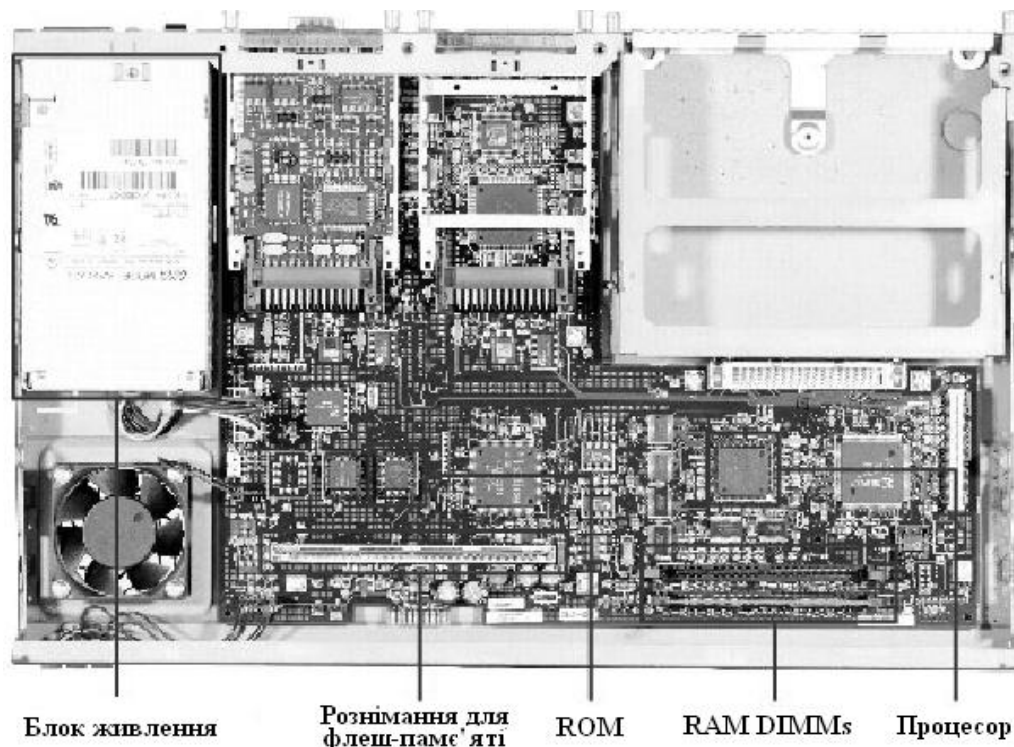


Рисунок 2.1 – Внутрішні компоненти Cisco Router 2600

На маршрутизаторах Cisco виконується високопродуктивна операційна система IOS (Cisco Internetworking Operating System), створена на базі ОС UNIX, яка фізично розміщена в енергонезалежній пам'яті маршрутизатора (FLASH).

Флеш-пам'ять. Флеш-пам'ять в маршрутизаторах Cisco використовується приблизно з тією ж метою, що і жорсткий диск на комп'ютері. Флеш-пам'ять має наступні характеристики:

- енергонезалежна;
- пересписувана;
- об'єм від 8 до 128 Мбайт;
- і функції:
- зберігає образ або образи Cisco IOS;
- зберігає файли конфігурації.

Постійний запам'ятовуючий пристрій. Постійний запам'ятовуючий пристрій призначений тільки для читання. Для переходу на нову версію потрібно замінити мікросхему ПЗП, котрий має наступні характеристики:

- енергонезалежний;
- не пересписуваний;

і функції:

- зберігає спрощену (резервну) версію Cisco IOS, призначену для використання у тому випадку, якщо всі інші способи загрузки устаткування не вдаються;

- містить код функції ROM Monitor, який застосовується у тому випадку, якщо програмне забезпечення Cisco IOS, яке знаходиться на флеш-пам'яті, спотворено і не завантажується. А також він служить для діагностики та перенастроювання конфігурації на низькому рівні (наприклад, в тому випадку, якщо хтось змінив пароль, виключив тим самим доступ мережевого адміністратора до маршрутизатора).

Енергонезалежний ОЗП. Енергонезалежний ОЗП (NVRAM) має наступні характеристики:

- енергонезалежний;
- пересписуваний;
- об'єм від 32 до 256 Кбайт;

і функції:

- вказує шлях до образу Cisco IOS і файлу пускової конфігурації;
- зберігає файл пускової конфігурації (startup-config file).

Процесор. Процесор в устаткуваннях Cisco служать тієї ж меті, що і в ПК: він є «мозком» устаткування. В більшості устаткування Cisco програмне забезпечення виконує багато обчислень, і для цього використовується процесор. В комутаторах процесор – це не такий важливий елемент, як в маршрутизаторах, тому що загальна частина обчислень виконується комутатором за допомогою спеціалізованих апаратних компонентів, що називаються модулями ASIC.

Об'єднувальна плата. Об'єднувальну плату можна порівняти з магістраллю, по якій ідуть всі взаємодії всередині мереженого устаткування. Її продуктивність має велике значення в комутаторах і інших устаткуваннях з високою густиною розміщення інтерфейсів.

Зовнішні компоненти. До зовнішніх компонентів відноситься консольний інтерфейс, допоміжний (AUX) інтерфейс, інтерфейси Ethernet, послідовні інтерфейси і слоти PCMCIA (рис.2.2).

Консольний інтерфейс. Консольний інтерфейс використовується для введення до системи Cisco IOS первісної інформації про конфігурацію і є окремим розніманням (connector) RJ-45. Консольний інтерфейс – це

низькошвидкісний асинхронний послідовний інтерфейс, який має особливе розташування виводів, і встановлює певні вимоги до типу кабелю, який повинен використовуватися для підключення станції керування (рис.2.3). Консольні кабелі зазвичай поставляються разом з маршрутизатором.

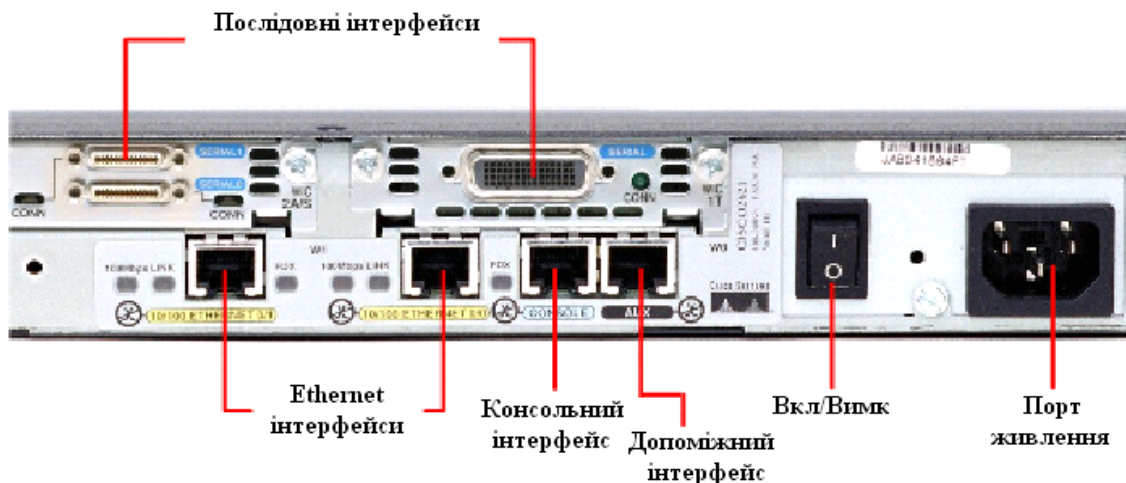


Рисунок 2.2 – Внутрішні компоненти Cisco Router 2600

Допоміжний інтерфейс. Допоміжний інтерфейс (AUX) - це ще один низькошвидкісний асинхронний послідовний інтерфейс, який звичайно використовується для підключення модему, що дозволяє здійснювати дистанційне адміністрування.

Ethernet інтерфейси. Ethernet інтерфейс – це інтерфейс, який дозволяє підключити дане мережеве устаткування к Ethernet мережі.

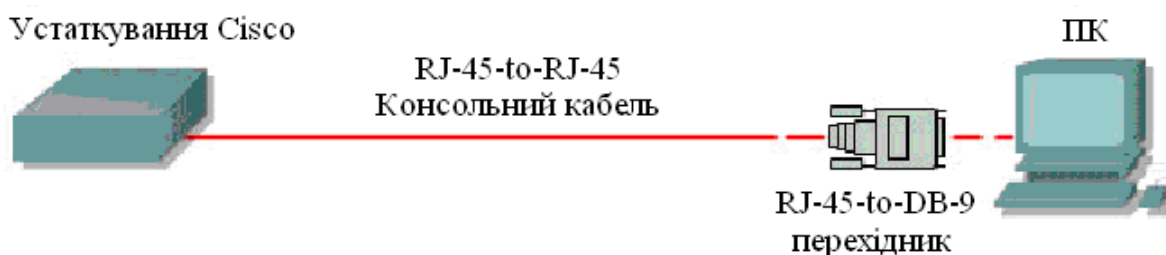


Рисунок 2.3 – Схема підключення до консольного інтерфейсу

Послідовний інтерфейс. Існує шість загальних специфікацій послідовного підключення: EIA/TIA-232, X.21, V.35, EIA/TIA-449, EIA-530 и HSSI. Послідовний інтерфейс призначений для підключення DCE1 и DTE2 устаткування.

¹ DCE – Data Communications Equipment – Апаратура передачі даних. Як правило це модем (модуль даних або модулятор/демодулятор пакетів на боці мережі каналу зв'язку),

2.2 Основні відомості про операційну систему Cisco IOS

На маршрутизаторах Cisco виконується високопродуктивна операційна система IOS (Cisco Internetworking Operating System), створена на базі ОС UNIX, яка фізично розміщена в енергонезалежній пам'яті маршрутизатора (FLASH).

Процес ініціалізації маршрутизатора виконується в наступній послідовності:

- POST (Power On Self Test) – тестування обладнання після включення живлення.
- Bootstrap IOS – програма завантаження основного IOS.
- Cisco IOS – основна операційна система маршрутизатора.
- Файл конфігурації із NVRAM. Виконуються команди, які зберігаються в цьому файлі.

Після автоперевірки включення живлення в процесі ініціалізації маршрутизатора відбуваються наступні події (рис.2.5):

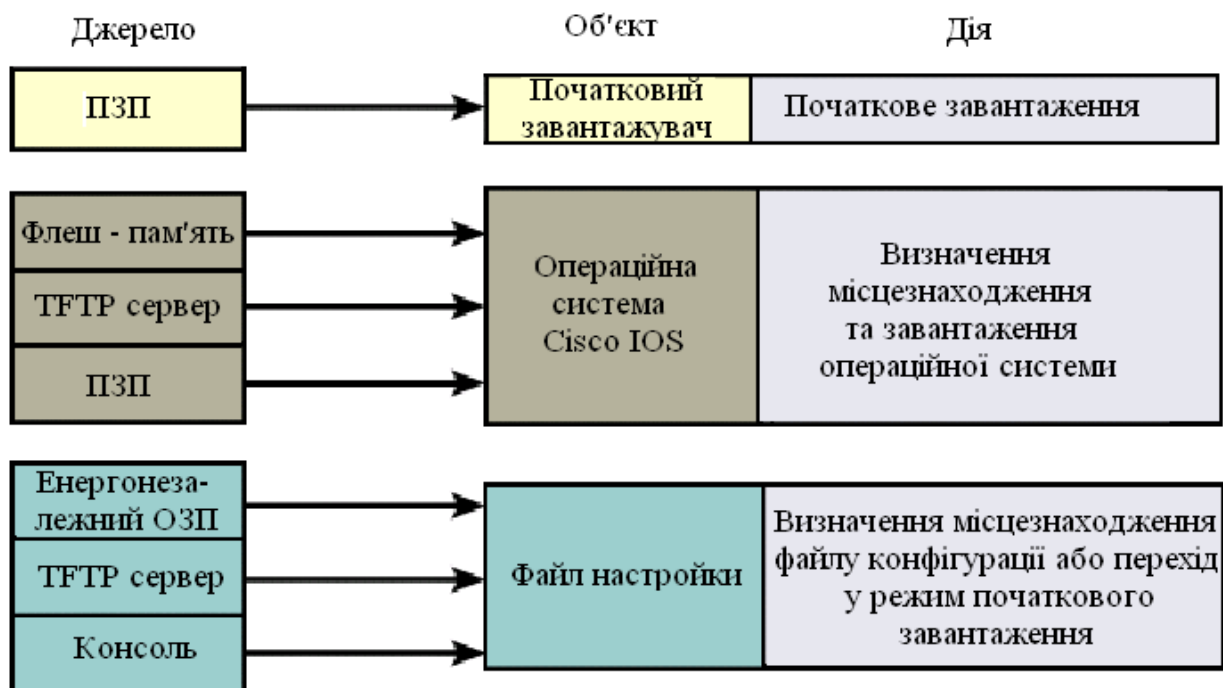


Рисунок 2.5 – Послідовність завантаження Cisco IOS

призначений для забезпечення сумісності двійкових даних, що передаються послідовно від джерела або передавача, з каналом зв'язку.

² DTE – Data Terminal Equipment – Термінальне устаткування. Апаратура користувача лінії зв'язку, яка виробляє дані для передачі лінією зв'язку и підключається безпосередньо к апаратурі передачі даних DCE. Це, наприклад, комп'ютери, комутатори і маршрутизатори.

Підключення до маршрутизатора здійснюється програмою TELNET до IP-адреси будь-якого з його інтерфейсів або при посередництві будь-якої іншої термінальної програми через консольний порт маршрутизатора CON, або додатковий порт AUX. Останньому способу слід надати перевагу, оскільки в процесі конфігурування маршрутизатора можуть змінюватися параметри IP – інтерфейсів, що може призвести до втрати з'єднання через TELNET. Окрім того, з міркувань безпеки доступ до маршрутизатора через TELNET слід заборонити.

Аварійне відключення оператора від консолі не реєструється маршрутизатором і сеанс залишається в тому ж стані. При повторному підключенні оператор опиниться в тому ж самому контексті, з якого відбулося аварійне відключення (якщо не спрацював автоматичний вихід по таймеру неактивності). Навпаки, при втраті TELNET-з'єднання маршрутизатор закриває сеанс роботи оператора.

При першому завантаженні IOS намагається завантажити конфігурацію з глобальної мережі. При невдалому завершенні цієї процедури IOS пропонує здійснити початкове конфігурування маршрутизатора за допомогою програми SETUP. Програма SETUP пропонує встановити деякі основні глобальні параметри конфігурації маршрутизатора шляхом діалогу питання-відповідь. До початкового конфігурування маршрутизатора відносяться наступні дії:

- Завдання імені маршрутизатора (за замовчуванням пропонується “Router”).
- Завдання пароля enable secret.
- Завдання пароля enable password.
- Завдання пароля віртуального терміналу.
- Конфігурування протоколів SNMP, IP, протоколів маршрутизації RIP IGRP.
- Конфігурування інтерфейсів.

Кожен з наведених вище етапів, запропонованих програмою SETUP, може бути проігнорований, а необхідні конфігураційні параметри можуть встановлюватися без посередництва програми SETUP за допомогою відповідних команд Cisco IOS. Крім цього, запуск програми SETUP є можливим в довільний момент з привілейованого режиму.

Правила роботи з командним рядком Cisco IOS. Взаємодія з системою Cisco IOS відбувається при посередництві інтерфейсу командного рядка (Command Line Interface, CLI). В загальному випадку формат команди виглядає наступним чином:

Команда [параметри або опції]

Параметри або опції, залежно від команди, можуть бути обов'язковим, необов'язковими або відсутніми взагалі. Для орієнтування в системі команд в Cisco IOS передбачена залежна від контексту система допомоги.

Допомога може знадобитися при необхідності отримання переліку команд, які розпочинаються попередньо введеною послідовністю символів. В цьому випадку пропонується завершити введену послідовність символом “?” (знак питання) – у відповідь Cisco IOS надасть перелік команд, які починаються шуканою послідовністю символів. Наступний приклад демонструє використання допомоги слова:

```
Router# co?  
configure connect copy
```

Допомога синтаксису дозволяє отримати перелік допустимих ключових слів та команд даного контексту або перелік допустимих параметрів команди. Для використання допомоги синтаксису пропонується одразу після ключового слова через пробіл ввести символ “?” (знак питання). В результаті буде видано перелік можливих команд чи параметрів команди.

У випадку введення невірної команди (помилка в слові, недопустима в даному контексті команда або невірно заданий параметр) Cisco IOS видасть відповідне повідомлення і вказівку імовірного місцезнаходження помилки в командному рядку. Ключове слово або невірний параметр в цьому випадку позначаються символом “^” (тильда). Наступний приклад демонструє реакцію системи на невірно введенне ключове слово “Ethernet”.

```
Router(config)#interface ethernat  
^Invalid input detected at '^' marker
```

Команди та ключові слова можна скорочувати до мінімально можливого – необхідно набрати кількість символів, яка є достатньою для однозначного трактування ключового слова чи команди. Якщо введена послідовність недостатня для однозначного трактування команди чи ключового слова – реакцією Cisco IOS на спробу виконати таку команду буде повідомлення, типу:

```
cisco(config)# Ambiguous command: "i"
```

Автозавершення – клавішею TAB можна завершити ввід команди, якщо кількість попередньо набраних символів команди задовольняє вище наведеної умові.

Для усунення необхідності повторного набору команд передбачено буфер історії команд, який надає можливість повторного використання введених раніше команд.

Контексти Cisco IOS. При роботі з командним рядком Cisco IOS передбачено декілька контекстів (режимів вводу команд). Поточний контекст ідентифікується символом запрошення вводу команди, який виводиться вслід за іменем маршрутизатора, наприклад Router> - контекст користувача; Router# - контекст адміністратора. Замість сигнатури "Router" виводиться назва маршрутизатора, якщо вона була наперед визначена за допомогою відповідної команди.

Контекст користувача – відкривається при підключенні до маршрутизатора і допускає виконання лише обмеженого набору основних контрольних команд, що не впливають на конфігурацію маршрутизатора. Якщо на протязі тривалого часу відсутні будь-які дії в контексті адміністратора, Cisco IOS автоматично переходить в контекст користувача.

Контекст адміністратора – відкривається командою **enable**, поданої в контексті користувача. Контекст адміністратора надає доступ до всіх без винятку команд (команди, що дозволяють отримати повну інформацію про конфігурацію маршрутизатора та його поточний стан, команди переходу в режим конфігурування, команди збереження та завантаження конфігурації). Зворотній перехід до контексту користувача відбувається по команді **disable** або по закінченні встановленого часу неактивності.

Контексти користувача та адміністратора можуть бути захищені паролями з метою запобігання несанкціонованого доступу незареєстрованих операторів, тому при вході до одного з цих контекстів може відбуватися запит пароля (Password:). При вводі пароля останній із міркувань безпеки на екрані терміналу не відображається. При роботі через сеанс TELNET пароль передається мережею у відкритому форматі. TELNET не вживає жодних засобів по забезпеченню захисту пароля від можливого перехоплення. Завершення сеансу роботи відбувається по команді **exit**.

Команди Cisco IOS чітко структуровані і доступні в різних контекстах і для успішної роботи з системою команд важливим є розуміння того, в якому контексті які команди є доступними. Для спрощення орієнтування в ієрархії команд вигляд рядка запрошення має унікальний вигляд. На рис.2.6 наведена проста схематична діаграма деяких контекстів Cisco IOS.

Кожна команда доступна лише на певному рівні ієрархії CLI (в певному контексті CLI). Наприклад, команди конфігурації не будуть доступними, поки інтерфейс не буде переведено на рівень глобального конфігурування командою **configure**.

В табл. 2.1 наведено перелік можливих контекстів та доступних команд системи команд.

Вихід з контексту глобального конфігурування до контексту адміністратора, а також вихід з будь-якого контексту до контексту верхнього рівня виконується командою **exit**. Комбінація CTRL+Z приводить до переходу в

контекст адміністратора з будь-якого підконтексту, до цього ж приводить команда end будь-якого підконтексту.

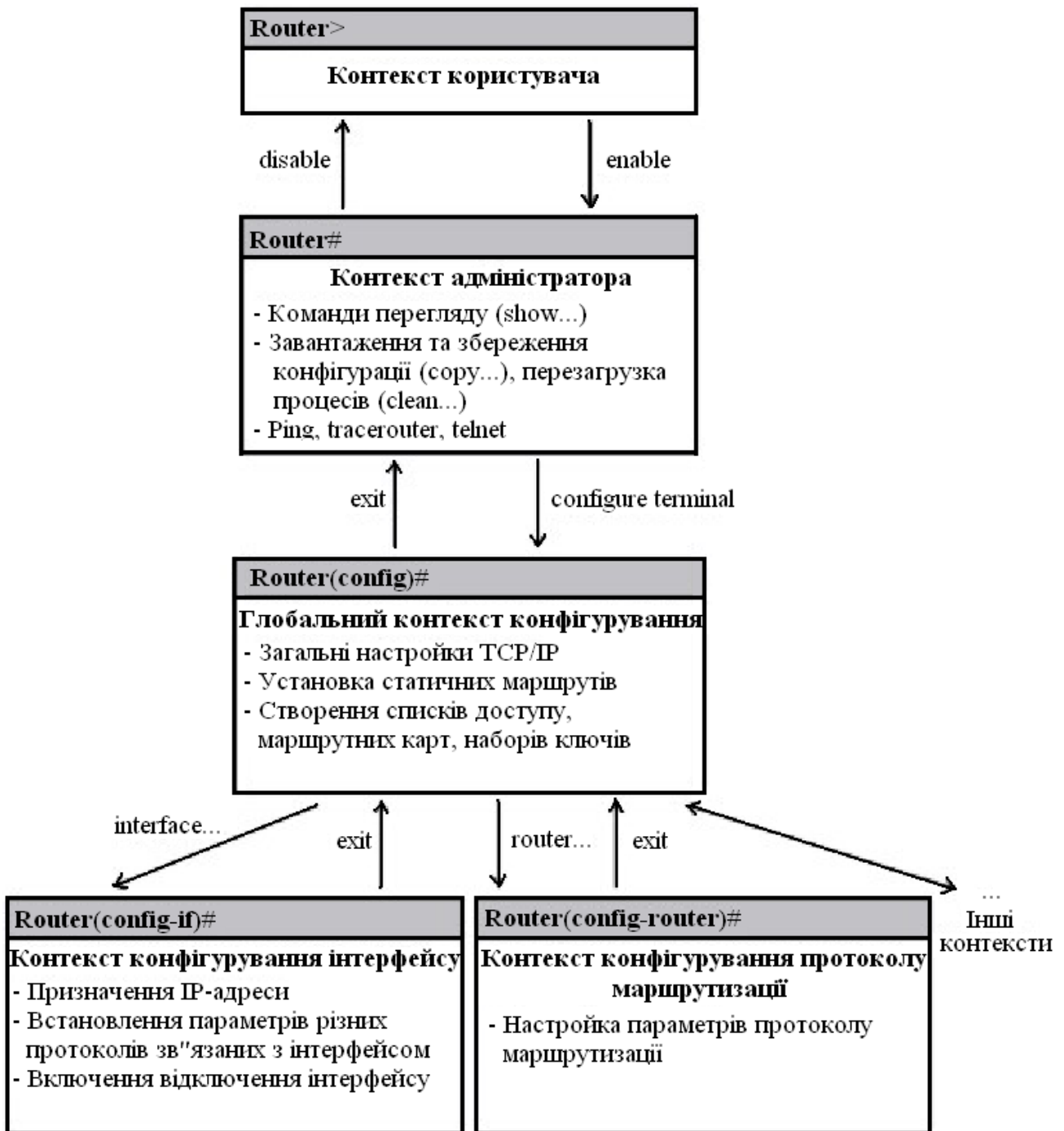


Рисунок 2.6 – Схематична ієрархія команд Cisco IOS

Відміна дії будь-якої команди реалізована за допомогою т.з. "негативних" команд – команд, яким передає префікс no, наприклад:

Router(config-if)#shutdown – виключає інтерфейс

Router(config-if)#no shutdown – включає інтерфейс.

Інколи при введенні негативних команд є потреба у вказуванні параметрів команд, дії яких вони відмінюють.

Таблиця 2.1 – Контексти та доступні команди системи команд Cisco IOS

| Контексти | Опис |
|------------------------|--|
| Router> | Режим користувача |
| Router# | Привілейований режим |
| Router(config)# | Режим глобального конфігурування |
| Router(config-if)# | Режим конфігурування інтерфейсу (контекст обраного інтерфейсу) |
| Router(config-router)# | Режим конфігурування маршрутизації |
| Router(config-line)# | Режим конфігурування віртуального терміналу |

Контекст адміністратора. Команди конфігурування дозволяють маніпулювати поточним режимом роботи маршрутизатора шляхом зміни значень параметрів, які зберігаються в файлі конфігурації.

Маршрутизатор Cisco зберігає конфігурацію в двох копіях – файл поточної конфігурації (running-config) в RAM та файл стартової конфігурації (startup-config) в NVRAM. Файли конфігурації є текстовими файлами, що містять секції, кожна з яких відповідає одній із підсистем маршрутизатора; в секціях прописуються значення конкретних параметрів відповідних підсистем. При завантаженні Cisco IOS зчитує команди конфігурації з файлу startup-config (в NVRAM) до файлу running-config (в RAM). Поточна конфігурація є активною у процесі функціонування маршрутизатора.

Всі команди вступають в дію одразу ж після їх введення і прописуються до файлу поточної конфігурації (running-config) в RAM. Деякі настройки маршрутизатора та його окремих підсистем мають значення за замовчуванням. До файлу конфігурації прописуються лише ті значення параметрів, які відрізняються від значень, прийнятих за замовчуванням.

Контекст адміністратора містить команди перегляду файлів поточної та стартової конфігурації:

show running-config [options] – перегляд файлу поточної конфігурації;

show startup-config [options] – перегляд файлу стартової конфігурації.

Параметри [options] дозволяють керувати процесом виводу і дозволяють, наприклад, здійснювати вивід не всього файлу, а вмісту деякої окремої його секції.

Якщо маршрутизатор втратить управління і буде перезавантажений, всі зміни, зафіксовані в running-config буде втрачено, якщо їх попередньо не було

збережено до файлу стартової конфігурації (startup-config) в NVRAM. Для збереження змін у файлі стартової конфігурації слід користуватися командою:

Router# copy running-config startup-config

Конфігурація маршрутизатора може зберігатися на TFTP – сервері і завантажуватися з нього. Для цього необхідно вказувати IP – адресу TFTP – сервера та назву файлу, під якою буде збережено файл конфігурації. Команда збереження на TFTP має вигляд:

copy <файл-джерело>TFTP://<IP – адреса TFTP >/[<назва файлу>]

Якщо параметр <назва файлу> не буде вказано, Cisco IOS запропонує вказати його значення в процесі діалогу.

При збереженні однієї конфігурації поверх іншої можливі два варіанти: перезапис і злиття. При перезапису стара конфігурація попередньо видаляється, а при злитті – команди нової конфігурації дописуються до старої так, ніби вони вводилися вручну. При злитті конфігурацій можлива низка побічних ефектів, що має особливе значення при злитті списків доступу, оскільки порядок запису рядків списків має суттєве значення. Злиття може змінити цей порядок і суттєво спотворити роботу маршрутизатора.

Примусове перезавантаження маршрутизатора здійснюється командою:

reload

Якщо на момент перезавантаження виявлено факт попередньої зміни файлу поточної конфігурації running-config, Cisco IOS запропонує варіанти його збереження в файлі startup-config (або відмова від збереження).

Контекст глобального конфігурування. Перехід до контексту глобального конфігурування здійснюється з контексту адміністратора командою configure:

з терміналу: configure terminal;

з NVRAM: configure memory;

з мережі: configure network.

В контексті глобального конфігурування виконуються команди, які впливають на функціонування системи в цілому, а також команди переходу до контекстів конфігурування конкретних підсистем маршрутизатора. Контекст глобального конфігурування ідентифікується рядком запиту (config)# і допускає виконання наступних команд:

1) hostname <назва маршрутизатора> - встановлює назву маршрутизатора замість назви за замовчуванням "Router".

2) [no] enable password <пароль> - команда парольного доступу до контексту адміністратора, який буде запитуватися під час виконання команди enable. Пароль прописується до файлу поточної конфігурації і зберігається там

у відкритому (нешифрованому) вигляді. При відсутності цього пароля переключення до привілейованого режиму можна здійснити лише при використанні консолі, а з віртуального терміналу буде доступний лише контекст користувача.

3) [no] enable secret <пароль> - команда, за своєю дією аналогічна попередньо описаній, однак пароль зберігається в зашифрованому MD5 – алгоритмом вигляді і має вищий пріоритет виконання.

4) [no] ip domain-lookup – дозволити/заборонити звернення до DNS(Domain Name Service).

5) [no] cdp run – дозволяє/забороняє використання протоколу CDP (Cisco Discovery Protocol) виявлення безпосередньо підключеної апаратури Cisco, тобто доступної на канальному рівні. Протокол з періодичністю 60 с. опитує порти маршрутизатора на предмет наявності апаратури Cisco і заносить інформацію про виявлені пристрої до бази даних. Маршрутизатори до безпосередньо приєднаних мереж заносяться до таблиці маршрутизації автоматично одразу ж після конфігурування інтерфейсу, при умові, що цей інтерфейс працездатний (line protocol up). Для формування додаткових статичних маршрутів призначена команда:

б) [no] ip route <dest.address><dest.mask><next-hop>[options]

<destination address> - адреса цільової мережі

<destination mask> - маска цільової мережі

<next-hop> - адреса сусіднього маршрутизатора

< options> - додаткові параметри, наприклад – параметри метрики

В якості параметра <next-hop> можна вказувати:

– безпосередню адресу сусіднього (доступного на канальному рівні) маршрутизатора;

– адресу віддаленої мережі або віддаленого хоста (опосередкована маршрутизація);

– локальний інтерфейс.

Опосередкований маршрут вказує на запис в таблиці маршрутизації, в якому знаходиться прямий маршрут. Дозволяється формувати таким чином послідовності маршрутов будь-якої довжини. Локальний інтерфейс рекомендується вказувати лише для двоточкових інтерфейсів.

Статичні маршрути фіксуються в файлі стартової конфігурації, а до таблиці маршрутизації піднімаються тільки за умовою досяжності вказаного в них маршруту.

Для регулювання пріоритетами маршрутів слід користуватися параметром <адміндістанція>, який може приймати значення від 0 до 255. Рівень пріоритету зворотно пропорційний значенню адміністративної дистанції і в таблицю маршрутизації з усіх активних маршрутів, що ведуть до даного

префікса піднімається лише маршрут з найменшим значенням адміндістанції. За замовчуванням адміндістанція статичних маршрутів рівна 1.

Нульове значення зарезервоване системою Cisco IOS і не може бути використане в явному вигляді, однак неявно нульову адміндістанцію мають також маршрути до безпосередньо приєднаних мереж. Маршрути, які в якості адміндістанції містять значення 255, до таблиці маршрутів не піднімаються.

7) [no] ip default network <адреса віддаленої мережі> - дозволяє вказати маршрут за замовчуванням, відмінний від стандартного. Параметр <адреса віддаленої мережі > повинен бути статично описаний в таблиці маршрутизації. Можливим є визначення декількох маршрутів за замовчуванням – в цьому випадку при обранні маршруту Cisco IOS користується значенням адміністративної дистанції та метричною інформацією. Маршрути за замовчуванням в таблиці маршрутизації позначаються символом "*".

Наведений нижче приклад демонструє використання маршруту в мережу 10.0.0.0 в якості маршруту за замовчуванням:

```
ip route 10.0.0.0 255.0.0.0 10.108.3.4
ip default-network 10.0.0.0
```

Cisco IOS має за замовчуванням зарезервований маршрут для використання його в якості маршруту за замовчуванням – 0.0.0.0/0. Cisco IOS надає можливість активації (деактивації) цього маршруту командою:

8) [no] ip classless

Команда ip classless активує зарезервований Cisco IOS маршрут за замовчуванням (0.0.0.0/0), а команда no ip classless деактивує цей маршрут. За замовчуванням цей маршрут активований, але не описаний.

Лінії керування. Налаштування ліній керування маршрутизаторів здійснюється окремо для кожної лінії в контексті обраної лінії, перехід до якого здійснюється з контексту глобального конфігурування командою:

line [aux | console | tty | vty] line-number [ending-line-number]

Дана команда приводить до зміни поточного контексту на контекст обраної лінії керування, який ідентифікується зміною рядка запрошення на (config-line)#

В якості параметрів команди вказуються:

– aux – додатковий EIA/TIA-232 DTE – порт. Повинен задаватися, як відносна лінія 0. Додатковий порт може використовуватися для підтримки модема та асинхронних зв'язків;

– con – консольна термінальна лінія (DTE);

– tty – стандартна асинхронна лінія;

– vty – віртуальна термінальна лінія, що використовується для віддаленого доступу до консолі;

– `line-number` – відносний номер останньої лінії (при конфігуруванні декількох ліній одночасно).

За замовчуванням маршрутизатор виводить діагностичні повідомлення тільки на консоль, а перенаправлення таких повідомлень на обрану лінію керування реалізується командою: **terminal monitor**.

Дія цієї команди відміняється командою: **no monitor**.

За замовчуванням маршрутизатор виводить системні повідомлення поверх вводу оператора і для продовження вводу оператор повинен пам'ятати, в якому місці його перервали. Для того, щоб дозволити після виводу кожного системного повідомлення вивід частини попередньо введеного оператором рядка, слід використовувати команду:

logging synchronous

Якщо по завершенню деякого проміжку часу (інтервал неактивності) спостерігається відсутність вводу з терміналу, Cisco IOS розриває поточну сесію. Інтервал неактивності встановлюється командою:

exec timeout <хвилини>[<секунди>]

Будь-яка команда X, введена в контексті оператора або адміністратора, сприймається маршрутизатором, як команда `telnet X`. Це приводить до того, що будь-який помилковий ввід примушує маршрутизатор опитувати сервер DNS для перетворення помилково введеного рядка в IP – адресу, що зумовлює затримки в роботі оператора. Уникнути таких затримок допомагає команда:

transport preffered none

З міркувань безпеки, доступ до маршрутизатора через віртуальний термінал слід обмежити за допомогою пароля. Встановити пароль можна командою:

[no] password <текст пароля>

Активація запиту пароля при в ході в Cisco IOS через віртуальний термінал виконується командою:

[no] login

При відсутності парольного захисту контексту адміністратора використовується пароль захисту лінії CON, якщо цей пароль встановлено.

Слід зауважити, що в робочому режимі з міркувань безпеки віртуальні термінали потрібно заблокувати, а доступ до маршрутизатора здійснювати лише по консольній лінії або через термінальний сервер.

Конфігурування інтерфейсів. Конфігурування інтерфейсів здійснюється окремо для кожного інтерфейсу в контексті обраного інтерфейсу, перехід до якого здійснюється командою контексту глобального конфігурування:

interface <тип><номер>

В якості параметру <тип> допускаються наступні слова: Ethernet, Fast Ethernet, Serial, Loopback, Null.

Вказана команда приводить до зміни поточного контексту на контекст конфігурування обраного інтерфейсу (config-if#).

На інтерфейсах Ethernet, окрім встановлення IP – адреси, як правило більше нічого робити не потрібно, однак Fast Ethernet може потребувати деяких примусових налаштувань дуплексного режиму або встановлення фіксованої швидкості (за замовчуванням ці параметри встановлюються шляхом переговорів, однак в окремих випадках переговори можуть не дати необхідних результатів).

Послідовні інтерфейси за замовчуванням на фізичному рівні є інтерфейсами DTE, а на канальному рівні – інтерфейсами HDLC (фірмову модифікацію Cisco HDLC). Якщо інтерфейс переведено в режим DCE, для нього слід задавати тактову частоту синхронізації передачі даних.

Для надання фізичному інтерфейсу IP – адреси слід використовувати команду:

ip address <IP-address><address-mask>

де <IP-address> - IP – адреса інтерфейсу;

<address-mask> - маска підмережі.

В деяких випадках може бути необхідність встановлення ширини смуги пропускання командою:

bandwidth <ширина-смуги-пропускання, кБіт/с>

За замовчуванням bandwidth може мати наступні значення:

- для Ethernet 10000;
- для Fast Ethernet 100000;
- для Serial 1544.

Слід зауважити, що значення параметра bandwidth не впливає на фізичну швидкість передачі, а використовується деякими протоколами маршрутизації для оцінки маршруту.

Тип середовища передачі вказується командою:

media-type <тип-середовища-передачі>

Параметр <тип-середовища-передачі> може приймати значення:

- для Ethernet "10BASE-T";
- для Fast Ethernet "100BASE-T", "100BASE-TX".

Для послідовних інтерфейсів, які використовують функції DCE, необхідно вказати фізичну швидкість передачі даних. Це можна зробити командою:

clock rate <фізична-швидкість-передачі, кБіт/с>

Параметр <фізична-швидкість-передачі, кБіт/с> може приймати фіксовані значення, перелік яких можна попередньо проглянути, ввівши `clock rate?`.

Для послідовного інтерфейсу, що виконує функцію DTE також може бути вказаний цей параметр, однак він буде проігнорований Cisco IOS і жодного впливу на роботу інтерфейсу не матиме, оскільки обладнання DTE запозичує цей параметр від DCE.

За замовчуванням фізичні інтерфейси виключені (неактивні – `administratively down`). Для їх активації використовується команда:

[no] shutdown

Ця команда переводить інтерфейс до стану `manual up`. Якщо зовнішнє обладнання вимкнено, то Cisco IOS автоматично переведе фізичний інтерфейс до стану `manual down`, а при активізації зовнішнього обладнання фізичний інтерфейс підніметься до стану `manual up` автоматично.

Для послідовних інтерфейсів іноді виникає необхідність використовувати протокол каналного рівня, відмінний від протоколу за замовчуванням (HDLC). Cisco IOS надає можливість вказати тип використовуваного протоколу командою:

encapsulation <протокол>

Параметр <протокол> може приймати фіксовані значення, для яких Cisco IOS передбачені зарезервовані ключові слова, наприклад PPP, Frame-Relay і т.п. Повний перелік значень параметра <протокол> доступний для перегляду командою `encapsulation?`.

Логічні інтерфейси конфігуруються командами, які наведені вище, за винятком того, що параметри `media-type`, `clock rate` та операція `[no] shutdown` для них не мають сенсу.

2.3 Програма Packet Tracer 5.3.2

Cisco Packet Tracer – емулятор мережі передачі даних, який випускається компанією Cisco System. Програмні продукти Packet Tracer надають можливість створювати мережеві топології із широкого спектру маршрутизаторів і комутаторів Cisco, робочих станцій та мережевих з'єднань типу Ethernet, Serial, ISDN, Frame Relay. Ця функція може бути виконана як для навчання, так і для роботи. Наприклад, щоб провести настройку мережі ще на етапі планування або щоб створити копію робочій мережі з метою усунування недоліків.

Загальний вигляд програми представлений на рис.2.7

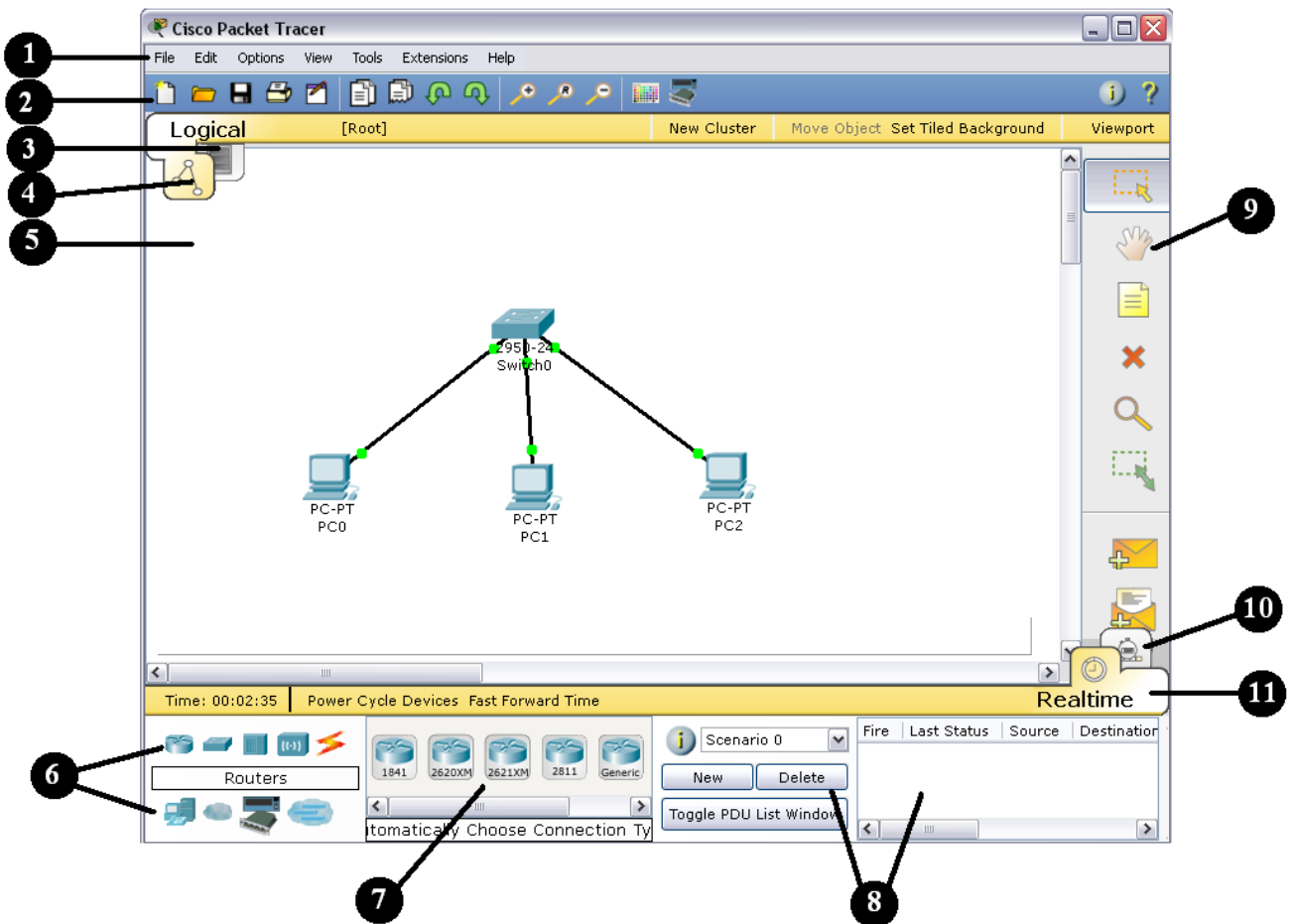


Рисунок 2.7 – Інтерфейс програми Packet Tracer

Робоча область вікна програми складається з наступних елементів:

1. **Menu Bar** – головне меню програми. Дозволяє детально налаштувати роботу програми. Панель містить меню File, Edit, Options, View, Tools, Extensions, Help.

2. **Menu Tool Bar** – піктографічне меню, містить графічні зображення ярликів для доступу к командам меню File, Edit, View і Tools, а також кнопку Network Information.

3. **Logical/Physical Workspace and Navigator Bar** – панель, яка надає можливість перемикає робочу область: фізичну чи логічну, а також дозволяє пересуватися між рівнями кластера.

4. **Logical Workspace and Navigator Bar** – режим побудови логічної топології мережі.

5. **Workspace** – область, в якій відбувається створення мережі, проводяться спостереження за симуляцією і проглядається різна інформація і статистика.

6. **Network Component Box** – це область, в якій вибираються устаткування і зв'язки для розміщення їх на робочому просторі. Вона містить області Device –Type Selection і Device-Specific Selection. Область Device –Type Selection містить доступні типи пристроїв і зв'язків, а область Device-Specific Selection змінюється в залежності від обраного пристрою.

7. **Device-Specific Selection** – область використовується для вибору конкретних устаткувань і з'єднань, необхідних для побудови в робочому просторі мережі. Вибір класу пристрою, яке буде елементом фізичної або логічної топології.

8. **User Created Packet Window** – вікно керує пакетами, які були створені в мережі під час симуляції сценарію.

9. **Common Tools Bar** – панель піктограм, яка забезпечує доступ до найбільш використовуваних інструментів програми: Select, Move Layout, Place Note, Delete, Inspect, Add Simple PDU і Add Complex PDU.

10. **Simulation Bar** – пакетний аналізатор, містить кнопки Play Control і перемикач Event List.

11. **Realtime Bar** – панель для роботи в режимі реального часу, містить кнопки, що відносяться до Power Cycle Devices.

Для створення топології необхідно вибрати устаткування з панелі Network Component, а далі з панелі Device–Type Selection вибрати тип обраного устаткування. Після цього потрібно натиснути ліву кнопку миші в полі робочої області програми (Workspace). Також можна витягнути пристрій прямо з області Device–Type Selection, але при цьому буде обрана модель пристрою за замовчанням.

Для швидкого створення декількох екземплярів одного і того ж пристрою потрібно натиснути кнопку Ctrl і разом з нею натиснути на пристрій, який вже знаходиться в області Workspace. Після цього можна декілька разів натискати на робочій області для додавання копій пристрою.

В Packet Tracer представлені наступні типи устаткування:

- маршрутизатори;
- комутатори (в тому числі і мости);
- хаби і повторювачі;

- ПК, сервери, принтери, IP – телефони;
- бездротові пристрої: точки доступу і бездротовий маршрутизатор;
- інші пристрої – хмара, DSL – модем і кабельний модем.

Додамо необхідні елементи в робочу область програми так, як показано на рис.2.8.

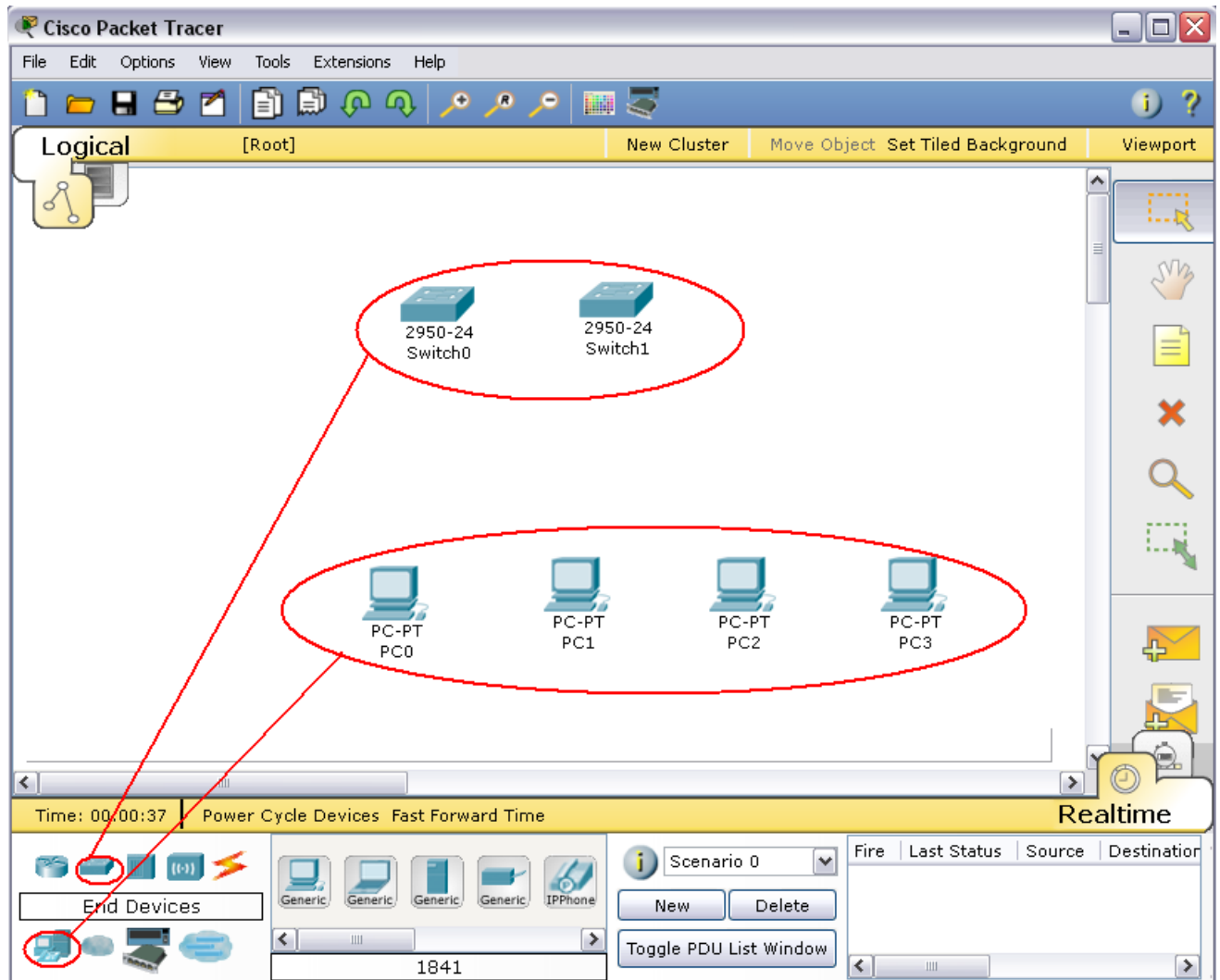


Рисунок 2.8 – Додавання елементів мережі

При додаванні кожного елемента користувач має можливість дати йому ім'я і установити параметри. Для цього необхідно натиснути на потрібний елемент лівою кнопкою миші (ЛКМ) і в діалоговому вікні устаткування перейти до вкладки **Config**.

Діалогове вікно властивостей кожного елемента має дві вкладки:

- Physical – містить графічний інтерфейс устаткування і дозволяє симулювати роботу з ним на фізичному рівні.

– Config – містить всі необхідні параметри для настройки устаткування і має зручний для цього інтерфейс.

Також в залежності від устаткування, властивості можуть мати додаткову вкладку для керування роботою обраного елемента: Desktop (якщо обране кінцеве устаткування) або CLI (якщо обраний маршрутизатор) і т.п.

Для видалення непотрібних устаткувань з робочої області програми використовується кнопка Delete (Del).

Додані елементи треба зв'язати за допомогою з'єднувальних зв'язків. Для цього необхідно вибрати вкладку Connections з панелі Network Component Box. Стануть доступними всі можливі типи з'єднань між устаткуваннями. Далі вибирається відповідний тип кабелю. Вказівник миші зміниться на курсор “connection” (має вигляд рознімання). Слід натиснути на першому пристрої і вибрати відповідний інтерфейс, к якому треба виконати з'єднання, а далі натиснути на другий пристрій, виконавши ту ж операцію. Можна також



з'єднати за допомогою **Automatically Choose Connection Type**  (автоматично з'єднує елементи в мережі). Між пристроями з'явиться кабельне з'єднання, а індикатори на кожному кінці покажуть статус з'єднання (для інтерфейсів які мають індикатор).









Рисунок 2.9 – Типи кабелю, що підтримуються в Packet Tracer

Packet Tracer підтримує широкий діапазон мережевих з'єднань. Вони описані в табл.2.2. Кожний тип кабелю може бути з'єднаний лише з певними типами інтерфейсів.

Таблиця 2.2 – Типи кабелю в Packet Tracer

| Тип кабелю | Опис |
|--|--|
|  <p>Console</p> | <p>Консольне з'єднання може бути виконане між ПК і маршрутизаторами або комутаторами. Для цього повинні виконуватися деякі вимоги для роботи консольного сеансу з ПК: швидкість з'єднання з обох сторін повинна бути однаковою, 7 біт даних (або 8 біт) для обох сторін, однаковий контроль парності, 1 або 2 стопових біта (але вони не обов'язково повинні</p> |

| | |
|---|--|
| | бути однаковими), а потік даних може бути будь-яким для обох сторін. |
|  Copper Straight - through | Цей тип кабелю є стандартним середовищем передачі Ethernet для з'єднання пристроїв. Він повинен бути з'єднаний з наступними типами портів: мідний 10 Мбіт/с (Ethernet), мідний 100 Мбіт/с (Fast Ethernet) і мідний 1000 Мбіт/с (Gigabit Ethernet). Використовується для з'єднання типу ПК-комутатор, маршрутизатор-комутатор). |
|  Copper Cross - over | Цей тип кабелю є середовищем передачі Ethernet для з'єднання пристроїв. Використовується для з'єднання типу ПК-ПК, комутатор-комутатор, маршрутизатор-маршрутизатор, маршрутизатор-ПК). |
|  Fiber | Оптоволоконне середовище використовується для з'єднання між оптичними портами (100 Мбіт/с або 1000 Мбіт/с). |
|  Phone | З'єднання через телефонну лінію може бути здійснено тільки між пристроями, які мають модемні порти. |
|  Coaxial | Коаксіальне середовище використовується для з'єднання між коаксіальними портами, такими як кабельний модем, з'єднаний з хмарою Packet Tracer. |
|  Serial DCE and DTE | З'єднання через послідовні порти, часто використовуються для зв'язку WAN. Для настройки таких з'єднань необхідно встановити синхронізацію на боці DCE – устаткування. Синхронізація DTE виконується за вибором. Сторону DCE можна визначити по маленькому малюнку «годинника» поряд з портом. При виборі типу з'єднання Serial DCE, перший пристрій, до якого застосовується з'єднання, становиться DCE – устаткуванням, а другий – автоматично стане стороною DTE. Можливе і зворотне розташування сторін, якщо обраний тип з'єднання Serial DTE. |

Найбільш часто будемо використовувати два типи кабелю: прямий (Copper Straight-through) і перехресний кабель (Copper Cross – over). Щоб

визначити тип кабелю RJ-45, треба положити два кінця кабелю разом, щоб побачити різнокольорові дроти, як це показано на рис. 2.10. На кінці кожного є вісім різнокольорових смужок або контактів. Якщо порядок слідкування кольорових контактів співпадає, то такий кабель називається прямим (рис.2.5).

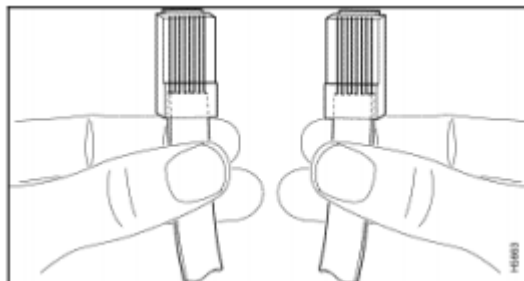
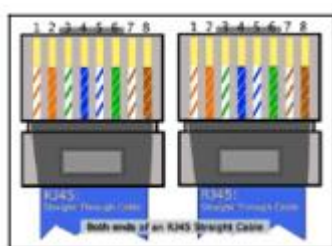
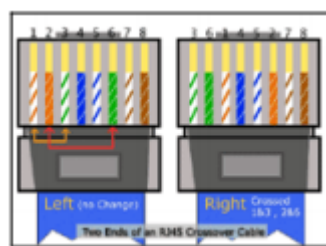


Рисунок 2.10 – Визначення типу кабелю RJ-45

Для того, щоб визначити який кабель слід використати для з'єднання, розділимо всі устаткування на два типи. Тип 1: мережеві адаптери комп'ютерів (LAN або Ethernet), WAN – порт маршрутизатора, рознімання Ethernet різних устаткувань (телевізори, тюнери та інш.). Тип 2: LAN- порти маршрутизаторів, LAN- порти ADSL модемів, всі порти концентраторів і комутаторів. При з'єднанні між собою двох пристроїв одного типу (наприклад, комп'ютер – комп'ютер або комутатор – комутатор) потрібен перехресний кабель, а при з'єднанні між собою двох пристроїв різного типу – прямий кабель (наприклад, комп'ютер – концентратор або комп'ютер – комутатор).



а) прямий кабель



б) перехресний кабель

Рисунок 2.11 – Вигляд прямого та перехресного кабелю

Після створення мережі її треба зберегти, вибравши пункт меню File->Save або іконку Save на панелі Main Tool Bar. Файл з збереженою топологією має розширення *.pkt.

Packet Tracer надає можливість симулювати роботу с інтерфейсом командного рядка (ІКР) операційної системи IOS, встановленої на всіх комутаторах і маршрутизаторах компанії Cisco.

Підключившись до устаткування, можна працювати з ним так, як за консоллю реального пристрою. Стимулятор забезпечує підтримку практично усіх команд, що доступні на реальних пристроях.

Підключення до ІКР комутаторів або маршрутизаторів можна провести, клацнувши на необхідний пристрій і переключившись в вікно властивостей до вкладки CLI.

Для симуляції роботи командної строки на кінцевому устаткуванні (комп'ютері) необхідно во властивостях вибрати вкладку Desktop, а далі натиснути на ярлик Command Prompt.

Робота з файлами в Packet Tracer. Програма Packet Tracer дозволяє користувачеві зберігати конфігурацію деяких пристроїв, таких як маршрутизатори або комутатори в текстових файлах. Для цього необхідно перейти до властивостей даного пристрою і у вкладці Config натиснути на кнопку “Export...” для експорту конфігурації Startup Config або Running Config. Відкриється діалогове вікно для збереження необхідної конфігурації в файл, який буде мати розширення *.txt. Текст файлу з конфігурацією пристрою running-config.txt (ім'я за замовчуванням) представляється аналогічним до тексту інформації, отриманому при використанні команди show running в IOS пристроях.

Слід відмітити, що конфігурація кожного устаткування зберігається в окремому текстовому файлі. Користувач також має можливість змінювати конфігурацію в збереженому файлі вручну за допомогою довільного текстового редактору. Для надання устаткуванню збережених або відредагованих налаштувань треба в вкладці Config натиснути кнопку “Load...” для завантаження необхідної конфігурації Startup Config або кнопку “Merge...” для завантаження конфігурації Running Config.

2.4 Хід роботи

1. Додати на робочу область програми 2 комутатора Switch-PT. За замовчуванням вони мають ім'я – Switch0 і Switch1.
2. Додати 4 комп'ютера з іменами за замовчуванням PC0, PC1, PC2, PC3.
3. З'єднати устаткування в мережу Ethernet, як показано на рис.2.12.

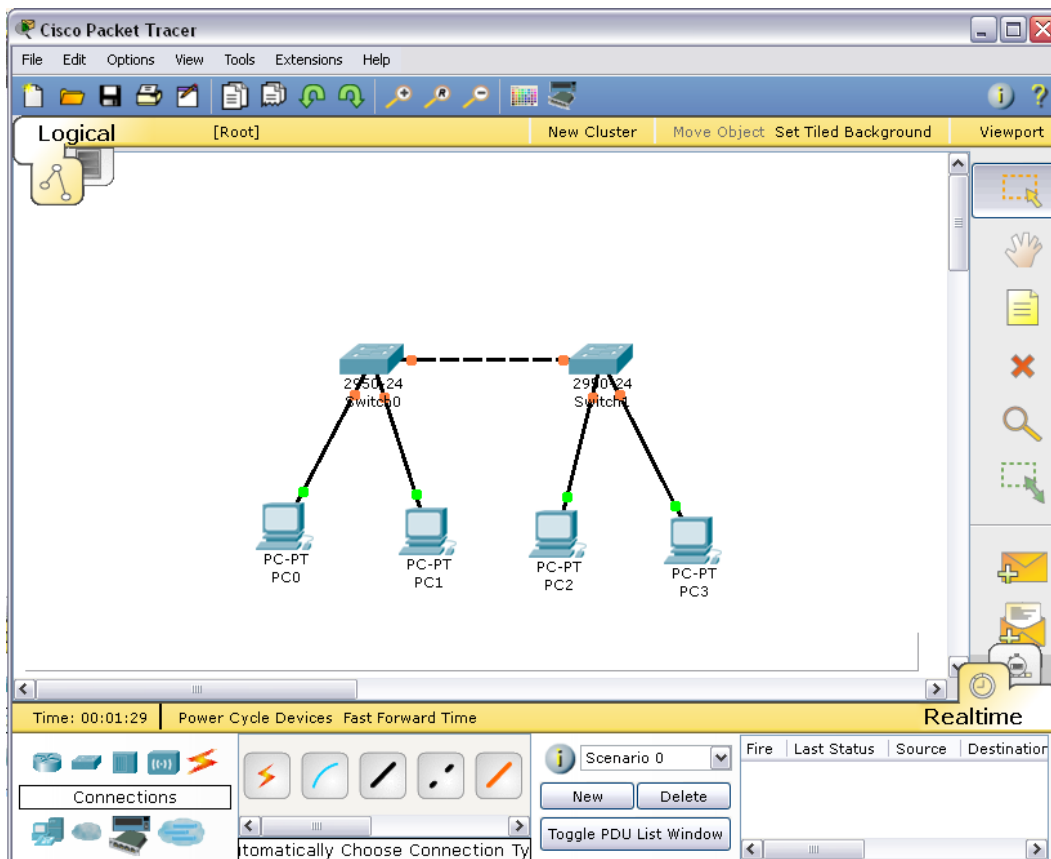


Рисунок 2.12 – Фізична топологія мережі для симуляції

4. Зберегти створену топологію, натиснувши кнопку Save (в меню File->Save).

5. Відкрити властивості устаткування PC0 натиснув на його зображенні. Перейти до вкладки Desktop і виконати симуляцію роботи run натиснувши Command Prompt.

6. Перелік команд можна отримати, якщо ввести ? і натиснути Enter. Для конфігурування комп'ютера слід скористатися командой ipconfig з командного рядка, наприклад, ipconfig 192.168.1.2 255.255.255.0

IP адресу і маску також можна вводити в зручному графічному інтерфейсі устаткування (рис.2.13). Поле DEFAULT GATEWAY – адреса шлюзу не важна, тому що мережа, що створюється не потребує маршрутизації.

Аналогічним способом слід налаштувати кожний комп'ютер, надавши їм IP-адреси з табл.2.3.

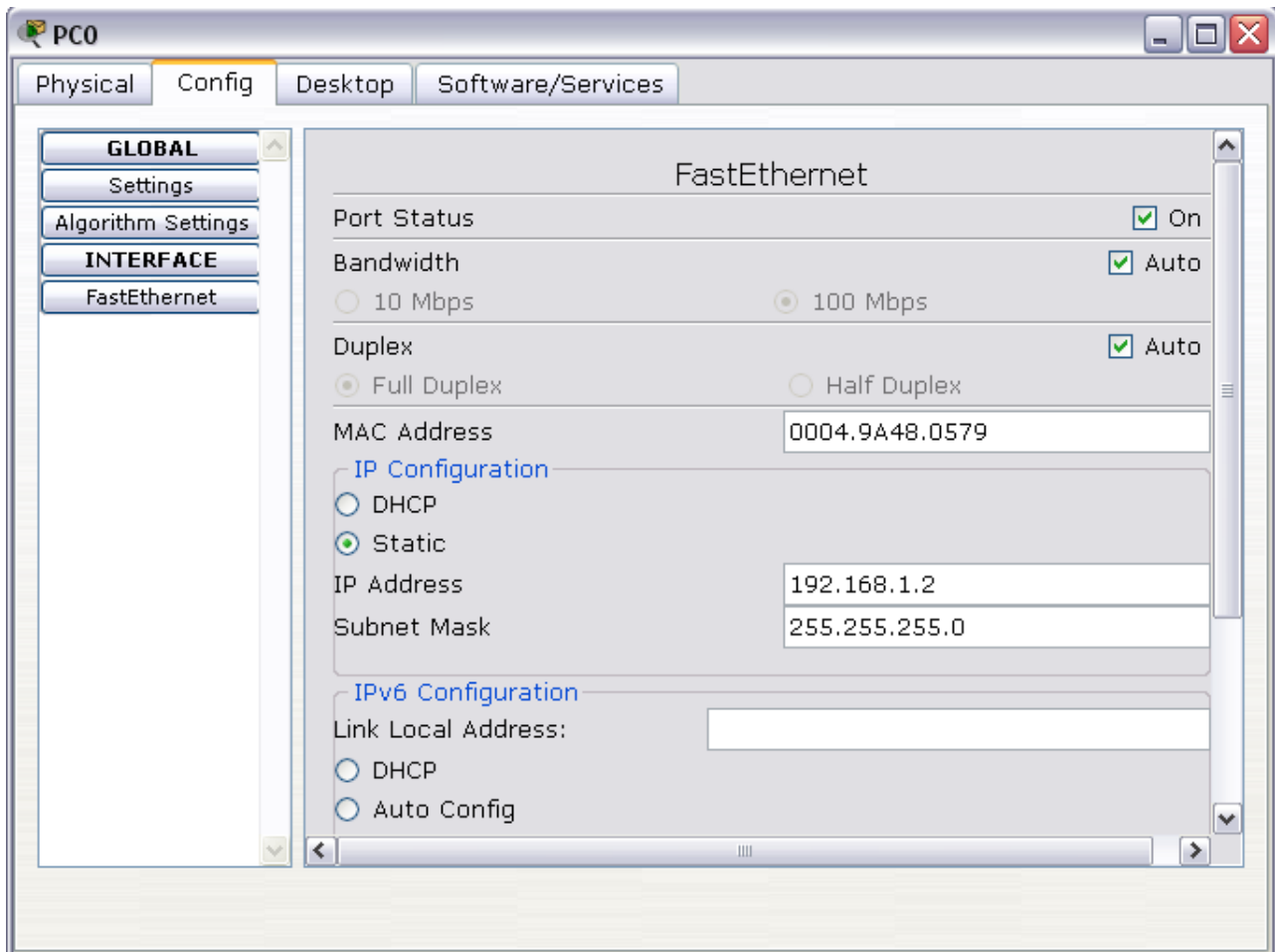


Рисунок 2.13 – Вкладка Config робочої станції PC0

Таблиця 2.3 – Перелік IP-адрес для конфігурації мережі

| Устаткування | IP ADDRESS | SUBNET MASK |
|--------------|-------------|---------------|
| PC0 | 192.168.1.2 | 255.255.255.0 |
| PC1 | 192.168.1.3 | 255.255.255.0 |
| PC2 | 192.168.1.4 | 255.255.255.0 |
| PC3 | 192.168.1.5 | 255.255.255.0 |

7. На кожному комп'ютері переглянути назначені адреси командою ipconfig без параметрів.

8. Якщо всі пункти виконані вірно, то можна пропінгувати будь-який комп'ютер з будь-якого іншого комп'ютера. Наприклад, з комп'ютера PC3 виконати пінгування до комп'ютера PC0. Звіт про виконання команди ping наведений на рис.2.14.

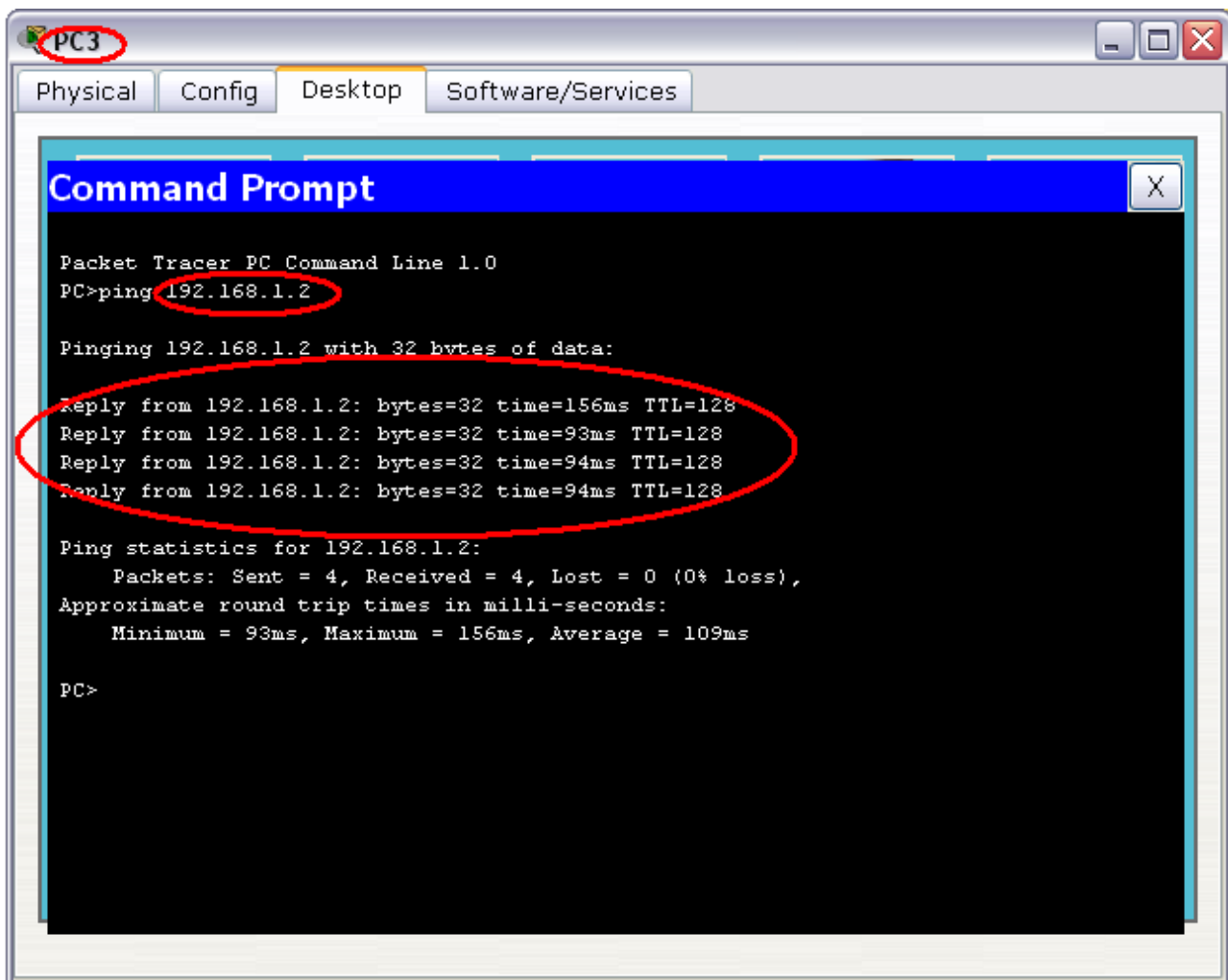


Рисунок 2.14 – Звіт про виконання команди ping між вузлами PC3 і PC0

2.5 Контрольні питання

6. Які типи мережевих пристроїв і з'єднань можна використовувати в Packet Tracer?
6. Яким способом можна перейти до інтерфейсу командного рядка устаткування?
6. Як конфігурувати пристрої з іншого комп'ютера?
6. Як додати в топологію і налаштувати нове устаткування?
6. Як зберегти конфігурацію устаткування в *.txt файл?

2.6 Перелік літератури

1. Паркер Т., К. Сиян TCP/IP. Для професіоналов. 3-е изд. - СПб.: Питер, 2004. - 859 с.: ил.

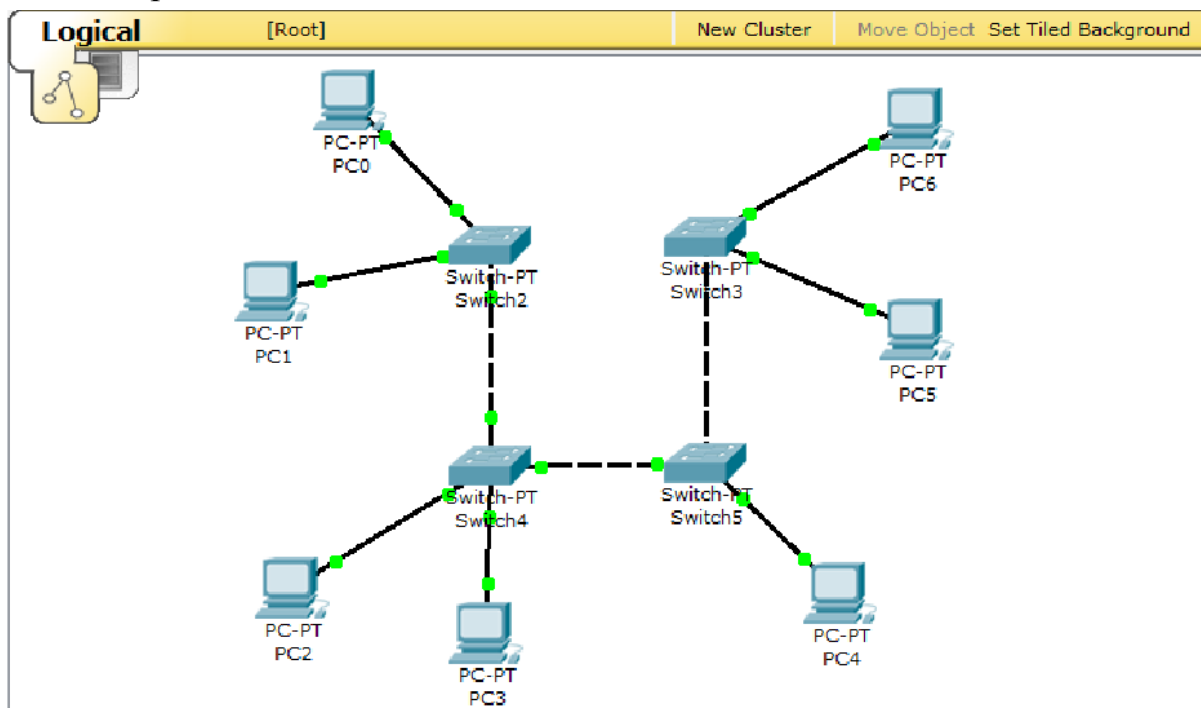
2. Снейдер И. Эффективное программирование TCP/IP. Библиотека программиста - СПб.: Питер, 2002. - 320 с.: ил.

3. Шамис В.А. Borland C++ Builder 6. Для профессионалов. - СПб.: Питер, 2004. - 798 с.: ил.

4. Олифер В.Г., Олифер Н.А. Компьютерные сети. Принципы, технологии, протоколы. – СПб.: Питер, 2001. – 672 с., ил.

2.7 Варіанти завдань для самостійної роботи

1. Створити топологію



4. Призначте комп'ютерам адреси згідно варіанту (v=1-12). Наприклад, для варіанту 7 (v=7) і комп'ютер PC1 має IP ADDRESS 7.1.1.1

Таблиця 2.4 – Варіанти для конфігурування комп'ютерів мережі

| Устаткування | IP ADDRESS | SUBNET MASK |
|--------------|------------|-------------|
| PC0 | v.1.1.1 | 255.0.0.0 |
| PC1 | v.1.1.2 | 255.0.0.0 |
| PC2 | v.1.1.3 | 255.0.0.0 |
| PC3 | v.1.1.4 | 255.0.0.0 |
| PC4 | v.1.1.5 | 255.0.0.0 |
| PC5 | v.1.1.6 | 255.0.0.0 |
| PC6 | v.1.1.7 | 255.0.0.0 |

3. Призначте комп'ютерам різні ім'я.
4. Якщо все буде зроблено вірно, то стане можливим пропінгувати будь-який комп'ютер з іншого.

3. Прилади, устаткування та інструменти

Для виконання лабораторної роботи використовуються ПЕОМ, об'єднані в локальну мережу, програмний емулятор IP-мереж Cisco Packet Tracer.

4. Правила техніки безпеки та охорони праці

Правила техніки безпеки при виконанні лабораторної роботи регламентуються: «Правилами техніки безпеки при роботі в комп'ютерній лабораторії».

6. Порядок проведення лабораторної роботи

Для виконання роботи кожен повинен:

1. Вивчити теоретичну частину лабораторної роботи.
2. Відповісти на контрольні запитання.
3. Виконати в Packet Tracer практичну частину лабораторної роботи.
4. Отримати у викладача варіант і виконати в Packet Tracer завдання для самостійної роботи.
5. Продемонструвати викладачу результати виконання завдання для самостійної роботи. Показати звіт про виконання команди ping з будь-якого комп'ютера на інший.
6. Оформити звіт.
7. Захистити звіт.

6. Оформлення та захист звіту

Звіт готується в електронному вигляді і роздруковується. Підготовлений до захисту звіт до лабораторної роботи повинен містити:

1. Найменування лабораторної роботи.
2. Відомості про виконавця, номер варіанту.
3. Завдання до лабораторної роботи.
4. Тема і мета першого завдання
5. Скріншот побудованої топології

6. Скріншот виконання команди ping згідно варіанту
7. Висновок за результатами роботи

Таблиця 2.5 – Варіанти виконання команди ping для перевірки працездатності мережі

| Варіант v | Ping з вузла | Ping до вузла | Варіант v | Ping з вузла | Ping до вузла |
|-----------|--------------|---------------|-----------|--------------|---------------|
| 1 | PC0 | PC5 | 7 | PC6 | PC4 |
| 2 | PC1 | PC6 | 8 | PC0 | PC5 |
| 3 | PC2 | PC0 | 9 | PC1 | PC6 |
| 4 | PC3 | PC1 | 10 | PC2 | PC0 |
| 5 | PC4 | PC2 | 11 | PC3 | PC1 |
| 6 | PC5 | PC3 | 12 | PC4 | PC2 |

МЕТОДИЧНІ ВКАЗІВКИ

до виконання лабораторних робіт
з дисципліни “Комп’ютерні мережі”
частина I

для студентів IV курсу денної форми навчання
Напрямок підготовки – комп’ютерні науки, спеціальність
7.080.401 “Інформаційні управляючі системи та технології”.

Викладач: доц. Кузніченко С.Д.

Одеський державний екологічний університет,
65016, м. Одеса, вул. Львівська, 15
