МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»

ІНСТИТУТ СПЕЦІАЛЬНОГО ЗВ'ЯЗКУ ТА ЗАХИСТУ ІНФОРМАЦІЇ

ПРОТОКОЛИ МЕРЕЖ ЕЛЕКТРОННИХ КОМУНІКАЦІЙ

Навчальний посібник

Рекомендовано Методичною радою КПІ ім. Ігоря Сікорського
як навчальний посібник для здобувачів освітнього ступеня бакалавра
за освітньою програмою «Безпека державних інформаційних ресурсів»
спеціальності 125 «Кібербезпека та захист інформації»

Укладачі: *Голь Владислав Дмитрович*, канд. техн. наук, проф.
Хахлюк Олексій Анатолійович, канд. техн. наук, доцент
Комонюк Євгеній Леонтійович

Електронне мережеве навчальне видання

В. Д. Голь, О. А. Хахлюк, Є. Л. Комонюк

Київ
КПІ ім. Ігоря Сікорського
2025

Укладачі: *Голь Владислав Дмитрович*, канд. техн. наук, проф.
Хахлюк Олексій Анатолійович, канд. техн. наук, доцент
Комонюк Євгеній Леонтійович

Рецензент *Правило В.В.,* канд. техн. наук, доцент
Перший заступник директора Інституту телекомунікаційних систем

Відповідальний редактор *Голь Владислав Дмитрович*, к.т.н, професор

Голь В.Д.

Протоколи мереж електронних комунікацій [Електронний ресурс] : навч. посіб. для здобувачів освітнього ступеня бакалавра за освіт. програмою «Безпека державних інформаційних ресурсів» спец. 125 «Кібербезпека та захист інформації» / В.Д. Голь, О. А. Хахлюк, Є.Л. Комонюк; ІСЗЗІ КПІ ім. Ігоря Сікорського. – Електрон. текст. дані (1 файл). – Київ: ІСЗЗІ КПІ ім. Ігоря Сікорського, 2025. – 233 с.

Навчальний посібник містить описання основ побудови, структури і архітектури, а також основних протоколів мереж електронних комунікацій.

Навчальний посібник призначений для підготовки та проведення навчальних занять з курсантами (студентами), які навчаються в Інституті за спеціальністю 125 «Кібербезпека та захист інформації», галузі знань 12 «Інформаційні технології»

Обсяг 10,66 авт. арк.

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
проспект Берестейський, 37, м. Київ, 03056
https://kpi.ua
Свідоцтво про внесення до Державного реєстру видавців, виготовлювачів і розповсюджувачів видавничої продукції ДК № 5354 від 25.05.2017 р.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

CDP	–	Cisco Discovery Protocol – закритий протокол другого рівня
STP	–	Spanning Tree Protocol – мережевий протокол, що працює на другому рівні моделі OSI
ACL	–	Access Control List – список прав доступу до об’єкта
NAT	–	Network Address Translation – це механізм у мережах TCP/IP, котрий дозволяє змінювати IP-адресу у заголовку пакету
PAT	–	Port Address Translation – це можливість мережевих пристроїв, яка транслює TCP або UDP зв’язки
AAA	–	Authentication, Authorization, Accounting – використовується для опису процесу надання доступу до обладнання мережі електроних комунікацій та контролю за ним
RADIUS	–	Remote Authentication Dial-In User Service – протокол, що забезпечує реалізацію послуг AAA
TACACS+	–	Terminal Access Controller Access Control System plus – протокол, що забезпечує реалізацію послуг AAA (виробник Cisco)
SSH	–	Secure SHell – мережевий протокол рівня застосунків, що дозволяє проводити віддалене управління вузлом з тунелюванням TCP-з’єднань
ICMP	–	Internet Control Message Protocol – мережевий протокол, що відстежує та повідомляє про наявність зв’язків в мережі з комутацією пакетів
VPN	–	Virtual Private Network – узагальнена назва технології, яка дозволяє створювати віртуальні захищені мережі поверх інших мереж із меншим рівнем довіри
OSI (ВВС)	–	Відкрита система взаємодії
TCP/IP	–	Internet Protocol / Transmission Control Protocol – це систематизований стек протоколів, що поділяється на чотири рівні, які корелюються з еталонною моделлю OSI
МЕК	–	мережа електронних комунікацій
MAC	–	Media Access Control – ідентифікатор обладнання, який однозначно його персоаналізує при взаємодії з середовищем передачі та іншим мережевим устаткуванням
CAM	–	Content Addressable Memory – тип памяті, який порівнює вхідні дані з попередньо завантаженим

		вмістом блоку CAM та генерує заданий результат залежно від типу CAM
ASIC	–	Application-specific integrated circuit – інтегральна схема, спеціалізована для вирішення конкретного завдання
NIC	–	Network interface controller – периферійний пристрій, що дозволяє комп’ютеру взаємодіяти з іншими пристроями мережі
КЦ	–	комутаційний центр
ПЗ	–	Software – сукупність програм системи оброблення інформації та програмних документів, необхідних для експлуатації цих програм
VLSM	–	Variable Length Subnet Mask – технологія, що дозволяє сегментувати адресний простір на частини потрібної розмірності
CIDR	–	Classless Inter-Domain Routing – безкласова міждоменна маршрутизація
ВОЛЗ	–	волоконно-оптична лінія зв’язку
ОК	–	оптичний кабель
АЛ	–	абонентська лінія
КУД	–	кінцеве устаткування даних
АКД	–	апаратура каналу даних
BPDU	–	Bridge Protocol Data Unit – фрейм протоколу управління мережевими мостами, IEEE 802.1d, базується на реалізації протоколу STP
STP	–	Spanning Tree Protocol – алгоритм протоколу сполучного дерева
BID	–	Bridge ID – ідентифікатор моста

ВСТУП

Навчальний посібник призначений для підготовки та проведення занять з навчальної дисципліни «Інформаційно-комунікаційні мережі» для курсантів (слухачів, студентів), які навчаються за спеціальністю 125(F5) «Кібербезпека та захист інформації», галузі знань 12(F) «Інформаційні технології», освітньо-професійної програми бакалаврів «Безпека державних інформаційних ресурсів» денної форми навчання і є навчальною дисципліною обов'язкової професійної та практичної підготовки.

Предметом навчальної дисципліни є вивчення особливостей побудови та оволодіння навиками проектування, експлуатації та аналізу інформаційно-комунікаційних мереж.

Метою навчальної дисципліни є формування у курсантів компетентностей:

– здатність застосовувати знання у практичних ситуаціях;

– знання та розуміння предметної області та розуміння професії.;

– вміння виявляти, ставити та вирішувати проблеми за професійним спрямуванням;

– здатність до використання інформаційно-комунікаційних технологій, сучасних методів і моделей інформаційної безпеки та/або кібербезпеки;

– здатність відновлювати штатне функціонування інформаційних, інформаційно-комунікаційних (автоматизованих) систем після реалізації загроз, здійснення кібератак, збоїв та відмов різних класів та походження.

Згідно з вимогами освітньо-професійної програми курсанти після засвоєння навчальної дисципліни мають продемонструвати такі результати навчання:

знання:

– теоретичних основ побудови інформаційно-комунікаційних мереж;

– перспектив розвитку і напрямків вдосконалення інформаційно-комунікаційних мереж;

– напрямків вдосконалення методів аналізу та розрахунку основних характеристик і параметрів функціонування інформаційно-комунікаційних мереж;

вміння:

– оцінювати ефективність способів побудови мереж та функціонування їх протоколів і процедур управління;

– самостійно робити інформаційний пошук та порівняльний аналіз структур і характеристик мереж;

досвід:

– налаштування систем адресування та маршрутизації мереж;

– проектування та експлуатації локальних мереж органів державного управління.

Науковою основою навчальної дисципліни є теорія побудови телекомунікаційних мереж, теорія реалізації архітектури інформаційних мереж,

теорія проектування (синтезу і аналізу) і розрахунку показників функціонування телекомунікаційних мереж.

Математичною базою навчальної дисципліни є такі розділи курсу математики, як теорія множин, диференційні та інтегральні обчислення, гармонічний спектральний аналіз сигналів, теорія ймовірностей і математична статистика, теорія масового обслуговування та випадкові процеси.

Успішне вирішення завдань навчальної дисципліни базується на знаннях та уміннях, сформованих у закладах середньої освіти.

Компетентності і результати навчання, які надаються навчальною дисципліною «Інформаційно-комунікаційні мережі» в подальшому забезпечують вивчення навчальної дисципліни «Застосування засобів криптографічного захисту інформації».

Змістовно навчальна дисципліна (кредитний модуль) згідно з робочим навчальним планом складається з наступних тем:

Тема 1. **Побудова інформаційно-комунікаційних мереж**

Класифікація інформаційно-комунікаційних мереж. Засоби структуризації мереж. Обладнання передачі сигналів даних. Еталонна модель взаємодії відкритих систем. Базові технології локальних мереж. Адресування в інформаційно-комунікаційних мережах. Засоби ефективного використання пулів IP-адрес. Основи роботи з програмою емуляції мереж CiscoPacketTracer.

Поняття комутації в мережах. Протокол визначення адрес ARP.

Проектування та налаштування мереж передачі даних. Виготовлення кабелю (пач-корд) для з'єднання обладнання локальної мережі. Виконання з'єднання комп'ютерів у локальну мережу та налаштування стеку протоколів TCP/IP.

Тема 2. **Керування в інформаційно-комунікаційних мережах**

Задачі систем управління мережами. Мережеві операційні системи та протоколи управління. Управління файловою системою Cisco. Версії та завантаження операційної системи Cisco IOS. Протокол виявлення сусідніх пристроїв CDP. Отримання інформації про віддалені пристрої. Протокол управляючих повідомлень ICMP. Управляючі утілити в стеці протоколів TCP/IP. Налаштування протоколу CDP. Створення та перевірка telnet/SSH-з'єднання. Використання управляючих повідомлень ICMP.

Протокол взаємодії мереж 2 рівня STP. Моделювання мережі з замкненими контурами у середовищі Packet Tracer. Налаштування мережі 2 рівня з замкненими контурами та резервуванням

Маршрутизація в інформаційно-комунікаційних мережах. Протоколи маршрутизації IP-мереж. Конфігурування інтерфейсів і протоколів маршрутизації. Відомості про використання протоколу SNMP. Протокол синхронізації часу NTP. Налаштування протоколу DHCP. Засоби моніторингу мереж.

Тема 3. **Протоколи реалізації надійних та безпечних мереж**

Модель інформаційної безпеки. Базова архітектура, засоби та протоколи безпеки мереж. Використання списків управління доступом. Протоколи зміни адрес NAT та PAT. Налаштування протоколів управління LAN.

Відновлення роботи Cisco IOS. Протокол резервування першого переходу FHRP. Протоколи комутованої мережі з агрегацією ліній

Моделювання та дослідження роботи протоколів HSRP. Моделювання і дослідження роботи протоколів агрегації ліній. Дослідження процесів відновлення роботи Cisco IOS.

РОЗДІЛ 1 ПОБУДОВА ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ МЕРЕЖ

1.1 Основи побудови інформаційно-комунікаційних мереж

1.1.1 Класифікація інформаційно-комунікаційних мереж

Для класифікації мереж передачі даних використовуються різні ознаки:

1. За розміром охопленої території

Локальна мережа (LAN, Local Area Network) Локальна обчислювальна мережа (ЛОМ) – мережа передачі даних (рисунок 1.1), що покриває зазвичай відносно невелику територію або невелику групу будівель (будинок, офіс, фірму, інститут).

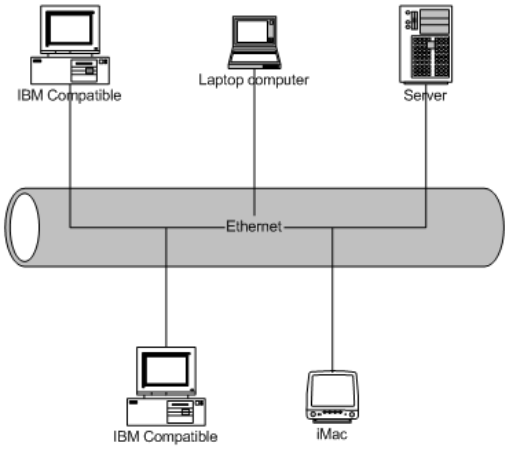


Рисунок 1.1 – Локальна обчислювальна мережа

Глобальна обчислювальна мережа, ГОМ (англ. Wide Area Network, WAN) є мережею передачі даних, що охоплює великі території і що включає десятки і сотні тисяч комп'ютерів. ГОМ служать для об'єднання розрізнених мереж так, щоб користувачі і комп'ютери, де б вони не знаходилися, могли взаємодіяти з усіма іншими учасниками глобальної мережі (рисунок 1.2). Деякі ГОМ побудовані виключно для приватних організацій, інші є засобом комунікації корпоративних ЛОМ з глобальною мережею Інтернет або за допомогою Інтернет з видаленими мережами, що входять до складу корпоративних. Частіше усього ГОМ спирається на виділені лінії, на одному кінці яких маршрутизатор підключається до ЛОМ, а на іншому концентратор зв'язується з іншими частинами ГОМ.

Глобальні мережі відрізняються від локальних тим, що розраховані на необмежене число абонентів і використовують, як правило, не надто якісні канали зв'язку. Правда, зараз вже не можна провести чітку і однозначну межу між локальними і глобальними мережами. Більшість локальних мереж мають вихід в глобальну мережу, але характер переданої інформації, принципи

організації обміну, режими доступу до ресурсів усередині локальної мережі, як правило, сильно відрізняються від тих, що прийнято в глобальній мережі. І хоча усі комп'ютери локальної мережі в даному випадку включені також і в глобальну мережу, специфіку локальної мережі це не відміняє. Можливість виходу в глобальну мережу залишається усього лише одним з ресурсів, що доступний користувачам локальної мережі. Більш того, останнім часом відокремлюють поняття міських обчислювальних мереж (MOM) (з англійської Metropolitan – місто, мегаполіс, Metropolitan Area Network, MAN) – мережі масштабу міста, великого регіону, які займають проміжне місце між мережами LAN та WAN.

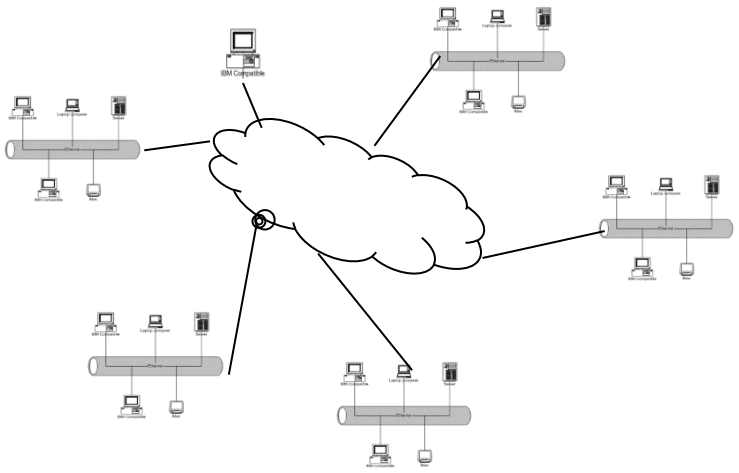


Рисунок 1.2 – Глобальна обчислювальна мережа

2. За типом функціональної взаємодії

Одноранговa архітектура (peer-to-peer architecture) – це концепція мережі, в якій її ресурси розосереджені по усіх системах (рисунок 1.3). Ця архітектура характеризується тим, що в ній усі системи рівноправні.

До одноранговых мереж відносяться малі мережі, де будь-яка робоча станція може виконувати одночасно функції файлового сервера і робочої станції.

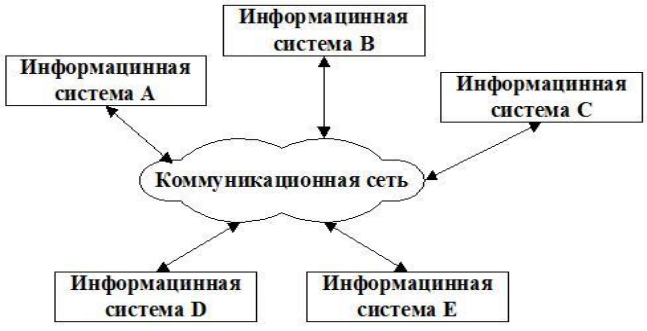


Рисунок 1.3 – Одноранговa архітектура

Мережа на основі сервера (рисунок 1.4). Така мережа характеризується наявністю одного особливого комп'ютера (сервера) та низькі інших (робочих станцій). **Сервер** – це об'єкт, що надає сервіс іншим об'єктам мережі по їх запитах (сервіс – це процес обслуговування клієнтів). На сервері зосереджена переважна більшість ресурсів мережі.

Рисунок 1.4 – Архітектура клієнт-сервер

Віртуальна приватна мережа (VPN – Virtual Private Network) – представляє собою об'єднання кількох окремих комп'ютерів або локальних мереж у віртуальну мережу, яка забезпечує обертання інформації між ними у захищеному режимі навіть при використанні громадської телекомунікаційної інфраструктури (рисунок 1.5). Це досягається шляхом застосування шифрування даних, перевірки їх цілісності та тунелювання (дані однієї мережі передаються через іншу мережу або через безпечний доступ до своєї мережі).

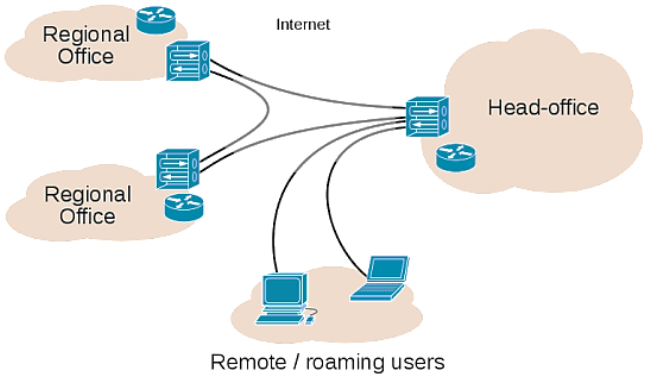


Рисунок 1.5 – Віртуальна приватна мережа (VPN)

3. За типом мережної топології

Під топологією мережі розуміється опис її фізичного розташування, тобто те, як комп'ютери сполучені в мережі один з одним і за допомогою яких пристроїв входять у фізичну топологію.

Існує чотири основні топології:

- Bus (шина);
- Ring (кільце);
- Star (зірка);
- Mesh (осередок).

1.1.2 Обладнання і середовища передачі сигналів даних

Для виконання задач, покладених на інформаційно-комунікаційну мережу, у її склад входять кінцеві пристрої, що одержали назву терміналів чи термінального устаткування, лінії зв'язку і комутаційні центри. Кожний з елементів має чітко визначені функції і при побудові може використовувати різні принципи.

Комутаційний центр це пункт мережі, який забезпечує регенерацію сигналів і (можливо) розподіл інформації в мережі згідно з заданою адресою. Комутаційний центр (КЦ) також називають вузлом (хостом, host) мережі. **Для апаратної реалізації вузлів використовують:**

– *Мережні карти*.

Устаткування, що зв'язує кінцевого користувача з мережею, їх називають також кінцевими вузлами або станціями і реалізується у вигляді плати мережного інтерфейсу (Network Interface Card – NIC) або мережного адаптера (рисунок 1.6).

– *Повторювачі (repeater).*

Представляють собою мережне обладнання (рисунок 1.7), яке функціонує на першому (фізичному) рівні еталонної моделі OSI. Метою використання повторювача є регенерація та синхронізація мережних сигналів на битовом рівні, що дозволяє передавати їх у середовищі на більшу відстань.

– *Концентратори (hub).*

Один з видів мережного обладнання (рисунок 1.8), яке можна встановлювати на рівні доступу мережі Ethernet. На концентраторах є кілька інтерфейсів для підключення вузлів до мережі. Концентратори це просте обладнання, він не в змозі визначити, якому вузлу призначене конкретне повідомлення. Він просто бере електронні сигнали одного інтерфейсу і відтворює (або ретранслює) те саме повідомлення для всіх інших інтерфейсів.

Рисунок 1.6 – Мережний адаптер (NIC)

Рисунок 1.7 –Повторювач (Repeater)

– *Мости (bridge).*

Представляють собою обладнання другого рівня (рисунок 1.9), призначене для створення двох сегментів мережі. Міст збирає інформацію про

те, на який його стороні (порту) знаходиться конкретна MAC-адреса, і приймає рішення щодо пересилання даних на підставі відповідного списку MAC-адрес. Мости здійснюють фільтрацію потоків даних на основі тільки MAC-адреси вузлів. З цієї причини вони можуть швидко пересилати дані будь-яких протоколів мережного рівня.

Рисунок 1.8 – Концентратор (Hub)

Рисунок 1.9 – Міст (Bridge)

– *Комутатори (switch).*

Комутатор з’єднує кілька вузлів з мережею. На відміну від концентратора, комутатор є пристроєм канального рівня (рисунок 1.10) і здійснює передачу повідомлення конкретному вузлу. Коли вузол відправляє повідомлення іншого вузла через комутатор, той приймає, декодує кадри і зчитує фізичну MAC-адресу призначення повідомлення. Далі здійснює відправку на порт відповідний цій адресі.

– *Маршрутизатор (router).*

Являють собою обладнання об’єднаних мереж (рисунок 1.11), які пересилають пакети між мережами на основі адрес третього рівня. Маршрутизатор є високоінтелектуальним пристроєм, що дозволяє також реалізовувати багато додаткових функцій.

Рисунок 1.10 – Комутатори (Switch) Cisco серії Catalyst 6500

Рисунок 1.11 – Маршрутизатор (Router) Cisco 1841

Наведене обладнання є основним, проте є ще величезний перелік спеціалізованого мережного обладнання, типу модеми, брандмауери (firewall), мовні шлюзи і т.д.

1.1.3 Засоби структуризації мереж

У загальному випадку інформаційно-комунікаційну мережу можна представити у вигляді узагальненої структури, що містить 4 макрорівні (рисунок 1.12).

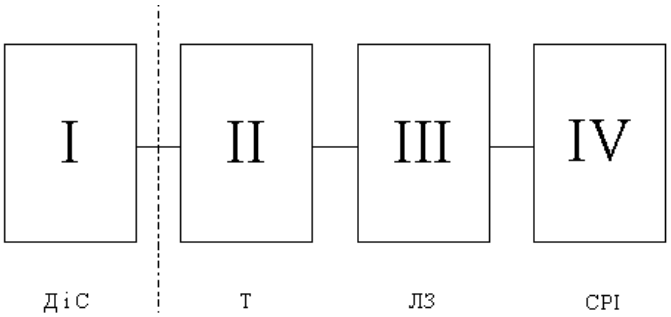


Рисунок 1.12 – Узагальнена структура інформаційно-комунікаційної мережі

Перший рівень – це джерела і споживачі інформації. Вони користуються послугами ІКМ і створюють потоки повідомлень різного виду і призначення. Саме вони визначають вимоги до ІКМ по доставці й обробці інформації з дотриманням визначених якісних і кількісних показників. Це зовнішнє середовище ІКМ.

Другий рівень – це термінали, кінцеві пристрої різного призначення. У загальному випадку вони являють собою інструмент, за допомогою якого джерела і споживачі інформації здійснюють введення/вивід повідомлень. При цьому, під повідомленням розуміється сукупність таких даних, що передані засобами комунікацій і мають ознаки початку і кінця.

Третій рівень – це лінії зв'язку. Вони забезпечують доставку повідомлень на необхідні відстані у вигляді сигналів електрозв'язку.

Четвертий рівень – це системи розподілу інформації (СРІ). Вони забезпечують розподіл повідомлень відповідно до зазначеної у заявці адреси і є ланкою, що поєднує ІКМ та всю комунікаційну систему в цілому. Основним елементом СРІ є різного роду комутаційні центри (КЦ) . Вони багато в чому визначають показники якості функціонування ІКМ.

Оптимальність побудови технічних пристроїв на кожному рівні, можливості їх адаптації до змін внутрішніх і зовнішніх параметрів, надійність і живучість багато в чому визначають можливості інформаційно-комунікаційної мережі.

У локальних мережах з невеликою кількістю комп'ютерів (10÷30) частіше усього використовують технології та структури які забезпечують властивість однорідності, тобто всі комп'ютери в такій мережі мають однакові

права у відношенні доступу до інших комп'ютерів. Така однорідність структури спрощує процедуру нарощування числа комп'ютерів, полегшує обслуговування й експлуатацію мережі.

Однак при побудові великих мереж однорідна структура зв'язків перетворюється з переваги в недолік. У таких мережах використання типових структур породжує різноманітні обмеження, найважливішими з яких є:

– обмеження на довжину зв'язку між вузлами;
– обмеження на кількість вузлів у мережі;
– обмеження на інтенсивність трафіка, що генерують вузли мережі.

Наприклад, технологія *Ethernet* на тонкому коаксіальному кабелі дозволяє використовувати кабель довжиною не більш 185 метрів, до якого можна підключити не більш 30 комп'ютерів. Однак якщо комп'ютери інтенсивно обмінюються інформацією, іноді приходиться знижувати число підключених до кабелю машин до 20, а то і до 10, щоб кожному комп'ютеру діставалася прийнятна частка загальної пропускної здатності мережі.

Для зняття цих обмежень використовуються особливі методи структуризації мережі і спеціальне структуроутворююче устаткування – повторювачі, концентратори, мости, комутатори, маршрутизатори. Такого роду устаткування також називають комунікаційним, маючи на увазі, що з його допомогою окремі сегменти мережі взаємодіють між собою.

Розрізняють:

1. Топологію фізичних зв'язків (фізичну структуризацію мережі). У цьому випадку конфігурація фізичних зв'язків визначається електричними з'єднаннями вузлів та структурними варіантами їх реалізації.

2. Топологію логічних зв'язків (логічну структуризацію мережі). Тут як логічні зв'язки виступають маршрути передачі даних між вузлами мережі, що утворюються шляхом відповідного налаштування комунікаційного устаткування та його типом.

Фізична структуризація мережі

Під **фізичною топологією** розуміється конфігурація зв'язків, утворених окремими ділянками середовища передачі.

Отже фізична структуризація реалізується шляхом вибору певної структури мережі. Вибір варіанту структури ЛОМ визначається в основному оперативними умовами (умовами управління), відношенням між навантаженням і пропускною спроможністю каналів, характеристиками технічних комплексів передачі даних.

Шина

Фізична топологія шина, яку називають також лінійною шиною, складається з єдиного кабелю, до якого приєднані усі комп'ютери сегменту (рисунок 1.13).

Повідомлення посилаються по лінії усім підключеним станціям незалежно від того, хто є одержувачем. Кожен комп'ютер перевіряє кожен пакет в дроті,

щоб визначити одержувача пакету. Якщо пакет призначений для іншої станції, то комп’ютер відкидає його. Якщо пакет призначений цьому комп’ютеру, то він отримає і обробить його.

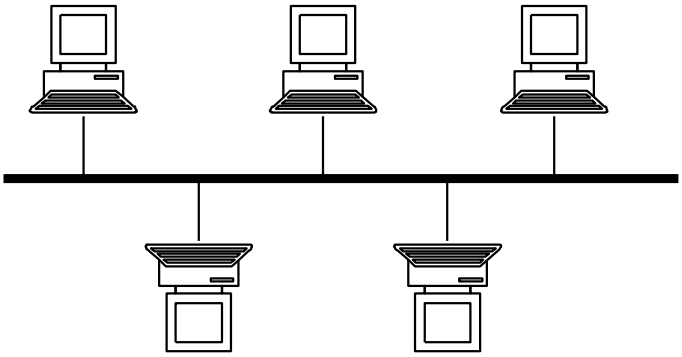


Рисунок 1.13 – Топологія «шина»

Головний кабель шини, відомий як магістраль, має на обох кінцях заглушки (термінатори) для запобігання віддзеркаленню сигналу.

Недоліки:

– важко ізолювати неполадки станції або іншого мережного компонента;

– неполадки в магістральному кабелі можуть привести до виходу з ладу усієї мережі.

Кільце

У фізичній топології «кільце» лінії передачі даних фактично утворюють логічне кільце, до якого підключені усі комп’ютери мережі (рисунок 1.14).

Доступ до носія в кільці здійснюється за допомогою маркерів (token), які пускаються по кругу від станції до станції, даючи їм можливість переслати пакет, якщо це треба. Комп’ютер може посилати дані тільки тоді, коли володіє маркером.

Оскільки кожен комп’ютер при цій топології є частиною кільця, він має можливість пересилати будь-які отримані ним пакети даних, адресовані іншій станції.

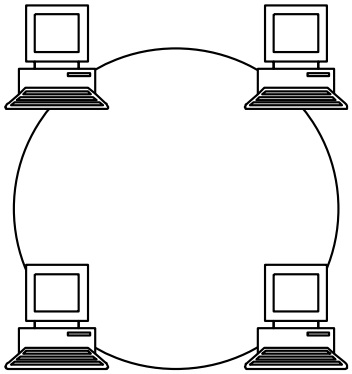


Рисунок 1.14 – Топологія «кільце"

Недоліки:

– неполадки на одній станції можуть привести до відмови усієї мережі;
– при переконфігурації будь-якої частини мережі необхідно тимчасово відключати усю мережу.

Зірка

У топології Star (зірка) усі комп'ютери в мережі сполучені один з одним за допомогою центрального концентратора (рисунок 1.15).

Усі дані, які посилає станція, прямують прямо на концентратор, який пересилає пакет у напрямі одержувача.

У цій топології тільки один комп'ютер може посилати дані в конкретний момент часу. При одночасній спробі двох і більше комп'ютерів переслати дані, усі вони дістануть відмову і будуть вимушені чекати випадковий інтервал часу, щоб спробувати ще раз.

Ці мережі краще масштабуються, чим інші мережі. Неполадки на одній станції не виводять з ладу усю мережу. Наявність центрального концентратора полегшує додавання нового комп'ютера.

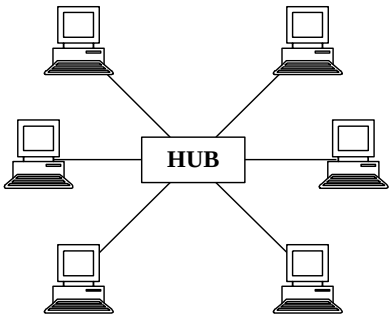


Рисунок 1.15 – Топологія «зірка"

Недоліки:

– вимагає більше кабелю, чим інші топології;
– вихід з ладу концентратора виведе з ладу увесь сегмент мережі.

Комірка

Топологія Mesh (комірка) сполучає усі комп'ютери попарно (рисунок 1.16).

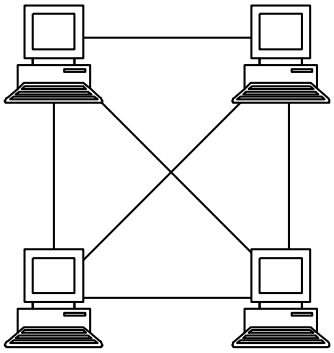


Рисунок 1.16 – Топологія «комірка"

Мережі Mesh використовують значно більшу кількість кабелю, чим інші топології. Ці мережі значно важче встановлювати. Але ці мережі стійкі до збоїв (здатні працювати за наявності ушкоджень).

Пристрої для фізичного з'єднання

1. Найпростіший з комунікаційних пристроїв – повторювач (***repeator***) – використовується для фізичного з'єднання різних сегментів кабелю локальної мережі з метою збільшення загальної довжини мережі. Повторювач передає сигнали, що приходять з одного сегмента мережі, в інший її сегмент (рисунок 1.17). Повторювач дозволяє зняти обмеження на довжину ліній зв'язку за рахунок відновлення переданого сигналу – відновлення його потужності й амплітуди, поліпшення і поновлення фронтів і т.п.

2. Повторювач, що має кілька портів і з'єднує кілька фізичних сегментів, часто називають концентратором (***concentrator***) чи хабом (***hub***). Ці назви (***hub*** – основа, центр діяльності) відбивають той факт, що в даному пристрої зосереджені всі зв'язки між сегментами мережі.

Використання концентраторів характерно практично для всіх базових технологій локальних мереж – ***Ethernet***, ***ArcNet***, ***Token Ring***, **FDDI**, ***Fast Ethernet***, ***Gigabit Ethernet***.

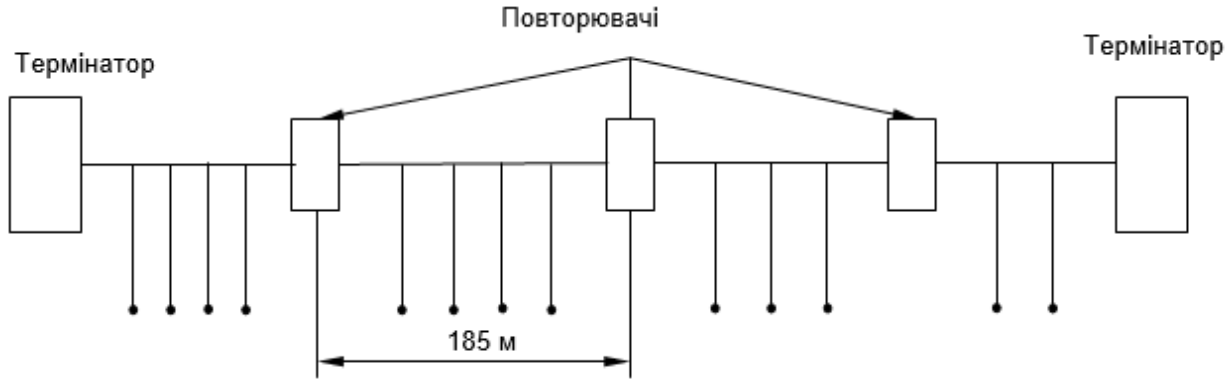


Рисунок 1.17 – Повторювач дозволяє збільшити довжину мережі

Потрібно підкреслити, що в роботі будь-яких концентраторів багато спільного – вони повторюють сигнали, що прийшли з одного з їхніх портів, на інших своїх портах. Різниця полягає в тому, на яких саме портах повторюються вхідні сигнали. Так, концентратор ***Ethernet*** повторює вхідні сигнали на усіх своїх портах, крім того, з якого сигнали надходять (рисунок 1.18).

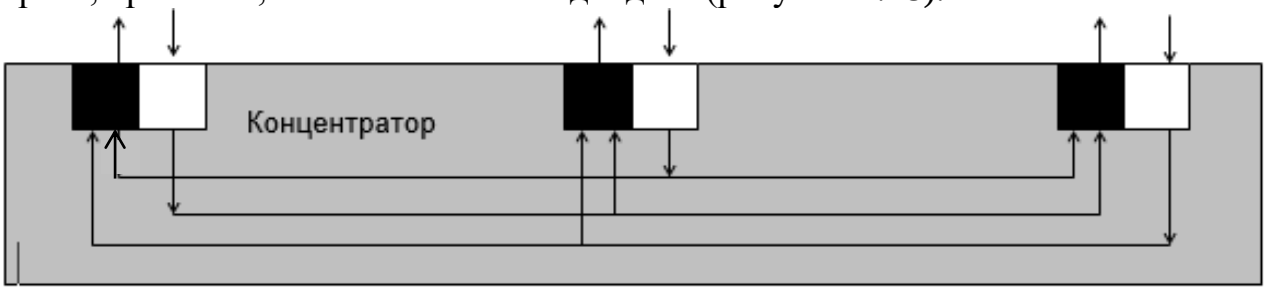


Рисунок 1.18 – Робота концентратора

Фізична структуризація мережі корисна в багатьох відношеннях, однак у ряді випадків, що звичайно відносяться до мереж великого і середнього розміру, без логічної структуризації мережі обійтися неможливо. Найбільш важливою проблемою, не розв'язуваною шляхом фізичної структуризації, залишається проблема перерозподілу переданого трафіка між різними фізичними сегментами мережі.

Логічна структуризація мережі

Логічна структуризація мережі – це процес розбивки мережі на сегменти з локалізованим трафіком.

Для підвищення ефективності роботи мережі необхідно враховувати неоднорідність інформаційних потоків (внутрішній або зовнішній трафік наприклад).

Поширення трафіка, призначеного для комп'ютерів деякого сегмента мережі, тільки в межах цього сегмента, називається локалізацією трафіка.

Для логічної структуризації мережі використовуються комунікаційні пристрої, що можуть аналізувати певні атрибути інформації, адреси призначення, номери портів призначення, види протоколів і т.д.:

- мости;
- комутатори;
- маршрутизатори;
- шлюзи.

Міст (*bridge*) пристрій, що поділяє середовище передачі мережі на частини (які часто називають логічними сегментами), передаючи інформацію з одного сегмента в іншій тільки в тому випадку, якщо адреса комп'ютера призначення належить іншій підмережі. Тим самим міст ізолює трафік однієї підмережі від трафіка іншої, підвищуючи загальну продуктивність передачі даних у мережі. Локалізація трафіка не тільки заощаджує пропускну здатність, але і зменшує можливість несанкціонованого доступу до даних, тому що кадри не виходять за межі свого сегмента, і зловмиснику складніше перехопити їх.

На рисунку 1.19 показана мережа, що була отримана з мережі з центральним концентратором шляхом його заміни на міст. Мережі 1-го і 3-го відділів є окремими логічними сегментами, в середині яких пристрої з'єднані через концентратор. Тобто кожен логічний сегмент, побудований на базі концентратора, має найпростішу фізичну структуру, утворену відрізками кабелю, що зв'язують комп'ютери з інтерфейсами концентратора. Якщо користувач комп'ютера А відправить дані користувачу комп'ютера В, що знаходиться в одному з ним сегменті, то ці дані будуть повторені тільки на тих мережних інтерфейсах, що відзначені на рисунку заштрихованими кружками.

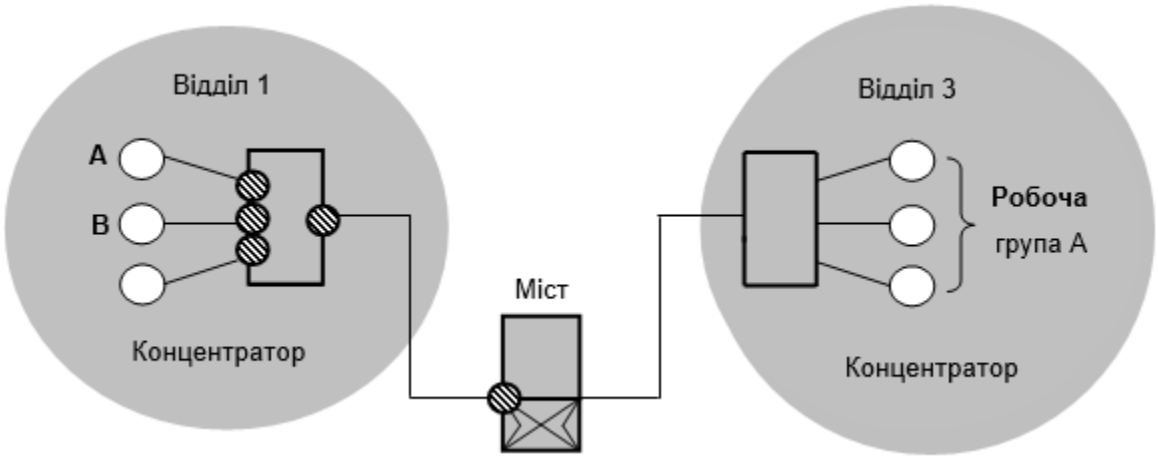


Рисунок 1.19 – Міст Ethernet

Мости використовують для локалізації трафіка апаратні (фізичні, MAC) адреси комп'ютерів. Це ускладнює розпізнавання приналежності того чи іншого комп'ютера до визначеного логічного сегмента – сама адреса не містить подібної інформації.

Тому міст досить спрощено реалізує розподіл мережі на сегменти – він запам'ятовує, через який інтерфейс на нього надійшов кадр даних від кожного комп'ютера мережі, і надалі передає кадри, призначені для даного комп'ютера, на цей інтерфейс. Точної топології зв'язків між логічними сегментами міст не знає. Через це застосування мостів приводить до значних обмежень на конфігурацію зв'язків мережі – сегменти повинні бути з'єднані таким чином, щоб у мережі не утворювалися замкнуті контури.

Комутатор (***switch***) за принципом обробки кадрів від моста практично нічим не відрізняється. Його відмінності полягають в тому, що він має більш ніж два інтерфейси та є свого роду комунікаційним мультипроцесором. Кожен інтерфейс комутатора оснащений спеціалізованою мікросхемою, що обробляє кадри по алгоритму моста незалежно від мікросхем інших інтерфейсів. За рахунок цього загальна продуктивність комутатора звичайно набагато вище продуктивності традиційного моста, що має один процесорний блок. Можна сказати, що комутатори – це мости нового покоління, що обробляють кадри в паралельному режимі.

В комутаторах з'являється також додаткова можливість структуризації на основі **віртуальних мереж VLAN**, тобто обмеження поширення трафіку на основі позначення кадрів інформації спеціальними заголовками – **тегами**. VLAN – це віртуальне відокремлення частини інтерфейсів комутатора, що забезпечує сегментацію та організаційну гнучкість у комутованій мережі. VLAN базуються на логічних зв'язках, а не на фізичних з'єднаннях.

Як показано на рисунку 1.20, VLAN у комутованій мережі дають змогу користувачам різних відділів (Faculty, Student і Guest) підключатися тільки до своєї відокремленої мережі незалежно від фізичного використання спільного комутатора або розташування в кампусній локальній мережі.

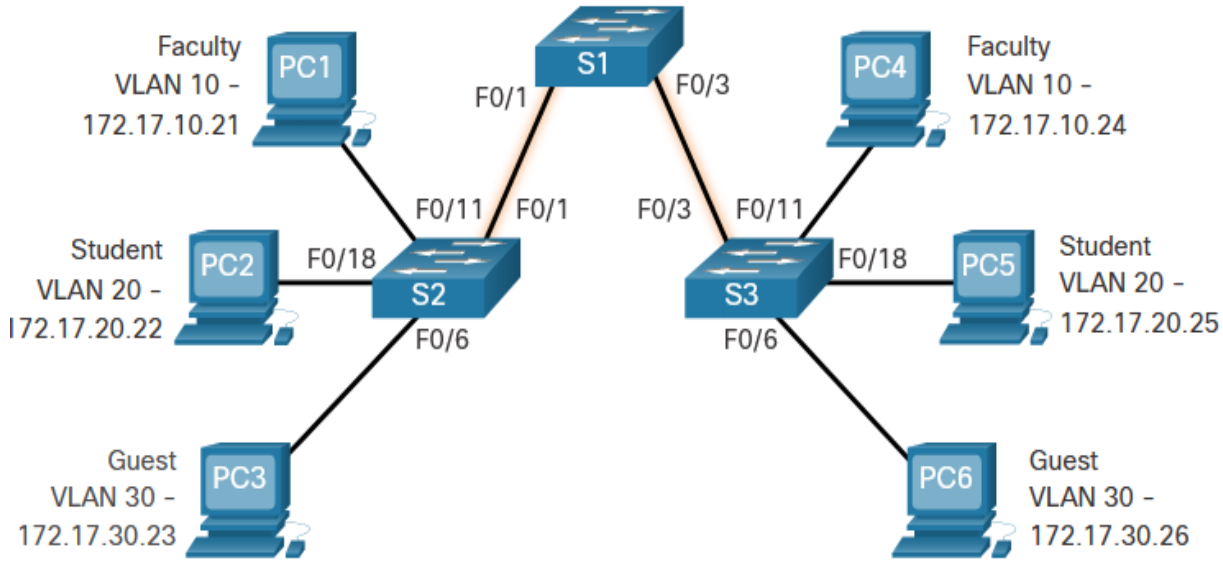


Рисунок 1.20 – Приклад використання VLAN

У таблиці 1.1 наведені переваги проектування мережі з VLAN.

Стандартний заголовок кадру Ethernet не містить інформації про VLAN, якій належить кадр. Тому при передаванні Ethernet-кадрів необхідно додати інформацію про VLAN, до яких належать ці кадри. Цей процес носить назву «тегування» і реалізовується за допомогою заголовка IEEE 802.1Q, описаного у стандарті IEEE 802.1Q. Заголовок 802.1Q містить 4-байтний тег (рисунок 1.21), який додається в оригінальний заголовок Ethernet-кадру, і саме в ньому вказується VLAN, якій належить кадр.

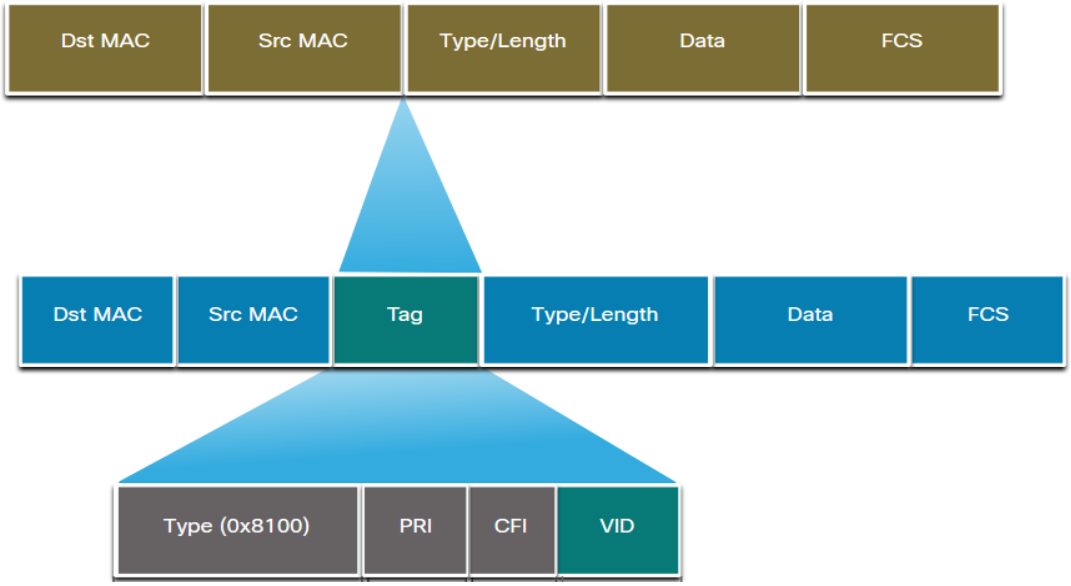


Рисунок 1.21 – Заголовок 802.1Q (VLAN)

Коли комутатор отримує кадр на інтерфейсі, що належить певній VLAN і налаштованому для функціонування в режимі доступу, комутатор додає тег VLAN в заголовок кадру, заново розраховує значення поля FCS і надсилає тегований кадр далі.

Таблиця 1.1 – Переваги проектування мережі з VLAN

Переваги	Опис
Менший широкомовний домен	• Поділ мережі на VLAN зменшує кількість пристроїв у широкомовному домені. • У мережі, що наведена на рисунку, є шість комп'ютерів, але лише три широкомовні домени (Faculty, Student, and Guest).
Покращена безпека	• Спілкуватися один з одним можуть тільки користувачі однієї VLAN. • На рисунку1.20 показано, що мережний трафік підрозділу Faculty (VLAN 10) повністю відокремлений та захищений від користувачів інших VLAN.
Підвищення ефективності IT-інфраструктури	• VLAN спрощують управління мережею, тому що користувачі з однаковими мережними потребами можуть бути налаштовані в одній і тій же VLAN. • Для полегшення ідентифікації VLAN можуть мати назви. • На рисунку VLAN 10 має назву «Faculty», VLAN 20 – «Student», VLAN 30 – «Guest».
Зниження витрат	• VLAN зменшують потребу у високовартісній модернізацій мережі та дають змогу використовувати існуючу пропускну здатність і висхідні канали зв'язку більш ефективно, як наслідок витрати зменшуються.
Краща продуктивність	• Менші широкомовні домени скорочують об'єм непотрібного трафіку у мережі і покращують продуктивність мережі в цілому.
Простіше керування проектами та застосунками	• VLAN об'єднують користувачів та мережні пристрої для забезпечення виробничих вимог або вимог щодо розміщення. • Наявність окремих функцій дозволяє спростити керування проектом або роботу зі спеціалізованим застосунком; прикладом такого застосунку є факультетська платформа для електронного навчання.

Як показано на рисунку 1.21, інформаційне поле керування тегами VLAN складається з полів:

- «Тип» (Type) – 2-байтне значення, яке називається ідентифікатором протоколу тега (TPID, Tag Protocol Identifier). Для технологій Ethernet – це шістнадцяткове значення 0x8100;
- «Пріоритет» (PRI, User Priority) – 3-бітне значення, яке підтримує реалізацію рівня чи служби (клас обслуговування);
- «Ідентифікатор канонічного формату» (CFI, Canonical Format Identifie) – 1-бітний ідентифікатор, який дозволяє передавати кадри технології Token Ring через канали мережі Ethernet;
- «Ідентифікатор VLAN» (VID, VLAN ID, VLAN Identifier) – 12-бітний ідентифікаційний номер VLAN, що може містити до 4096 номерів VLAN.

Використання VLAN було б не можливим без *транкових (магістральних) каналів VLAN*. Транкові канали VLAN змогу пристроям, які підключені до різних комутаторів, але належать одній VLAN, взаємодіяти без передачі трафіку через маршрутизатор. ***Транковий канал*** – це двоточковий канал зв'язку між мережними пристроями, який дає змогу передавати трафік більше ніж однієї

VLAN. Cisco підтримує стандарт IEEE 802.1Q для узгодження магістральних каналів на інтерфейсах технологій Fast Ethernet, Gigabit Ethernet та 10-Gigabit Ethernet. Транковий канал VLAN не належить до певної VLAN – він є каналом передавання трафіку декількох VLAN між комутаторами і маршрутизаторами. Транковий канал також може використовуватися між мережним пристроєм і сервером або іншим пристроєм, який оснащений мережною платою з підтримкою 802.1Q.

На рисунку 1.21 кольором виділені зв'язки між комутаторами S1 та S2, S1 та S3, які встановлені в режим транкового каналу і налаштовані на передачу трафіку, що надходить у мережу від VLAN 10, 20, 30 та, наприклад, 99 (Native VLAN).

В залежності від налаштування та призначення VLAN можуть бути різного типу. Відомі VLAN п'яти наступних типів:

1. VLAN за замовчуванням

VLAN за замовчуванням (Default VLAN) на комутаторі Cisco – це VLAN 1. Слід пам'ятати про VLAN 1 такі важливі факти:

- За замовчуванням всі інтерфейси комутатора належать до VLAN 1.
- За замовчуванням VLAN з нетегованим трафіком (Native VLAN) – це VLAN 1.
- За замовчуванням управлінська VLAN (Management VLAN) – це VLAN 1.
- Не можна перейменувати або видалити VLAN 1.

2. VLAN даних

VLAN даних (Data VLAN) – це VLAN, налаштовані для відокремлення трафіку, що створюється користувачами. Такі VLAN також називають VLAN користувачів, бо саме вони розділяють мережу на групи користувачів або групи пристроїв. Залежно від організаційних вимог сучасна мережа матиме багато VLAN даних. Голосовий трафік і трафік управління мережею не повинні передаватися у VLAN даних.

3. Управлінська VLAN

Управлінська VLAN – це VLAN даних, спеціально налаштована для передавання трафіку управління мережею, зокрема, трафіку протоколів SSH, Telnet, HTTPS, HHTP і SNMP.

4. Голосова VLAN

Для підтримки технології VoIP (Voice over IP) потрібна окрема VLAN, це пов'язане з особливими вимогами до VoIP трафіку з точки зору гарантованої пропускної здатності, уникнення перевантажених ділянок мережі, пріоритетного обслуговування у проміжних вузлах та мінімізації затримки.

5. Native VLAN

Native VLAN – це один з VLAN комутатора, на який покладена функція для обробки нетегованого трафіку. Такий трафік генерується самим комутатором або може надходити з застарілих пристроїв. Трафік згенерований у Native VLAN розміщується у транковому каналі 802.1Q без додавання тегу. Нетегований трафік на виході транкового каналу розподіляється у Native VLAN.

Обмеження, зв'язані з застосуванням мостів і комутаторів – по топології зв'язків, а також ряд інших, – привели до того, що в ряді комунікаційних пристроїв з'явився ще один тип устаткування – **маршрутизатор** (***router***). Маршрутизатори більш надійно і більш ефективно ніж мости, ізолюють трафік окремих частин мережі один від іншого. Маршрутизатори утворюють логічні сегменти за допомогою явної адресації, оскільки використовують не апаратні, а складені числові адреси (мережені). У цих адресах міститься поле номера мережі, так що всі комп'ютери, у яких значення цього поля однакове, належать одному сегменту, який називають в даному випадку підмережею (***subnet***).

Крім локалізації трафіка, маршрутизатори виконують ще багато інших корисних функцій. Так, маршрутизатори можуть працювати в мережі з замкнутими контурами, при цьому вони здійснюють вибір найбільш раціонального маршруту з декількох можливих. Так за наявності між підмережами відділів 1 і 2 прокладено додатковий зв'язок, який може використовуватися для підвищення як продуктивності мережі, так і її надійності (рисунок 1.22).

Іншою дуже важливою функцією маршрутизаторів є їхня здатність зв'язувати в єдину мережу підмережі, побудовані з використанням різних мережних технологій, наприклад *Ethernet* і X.25.

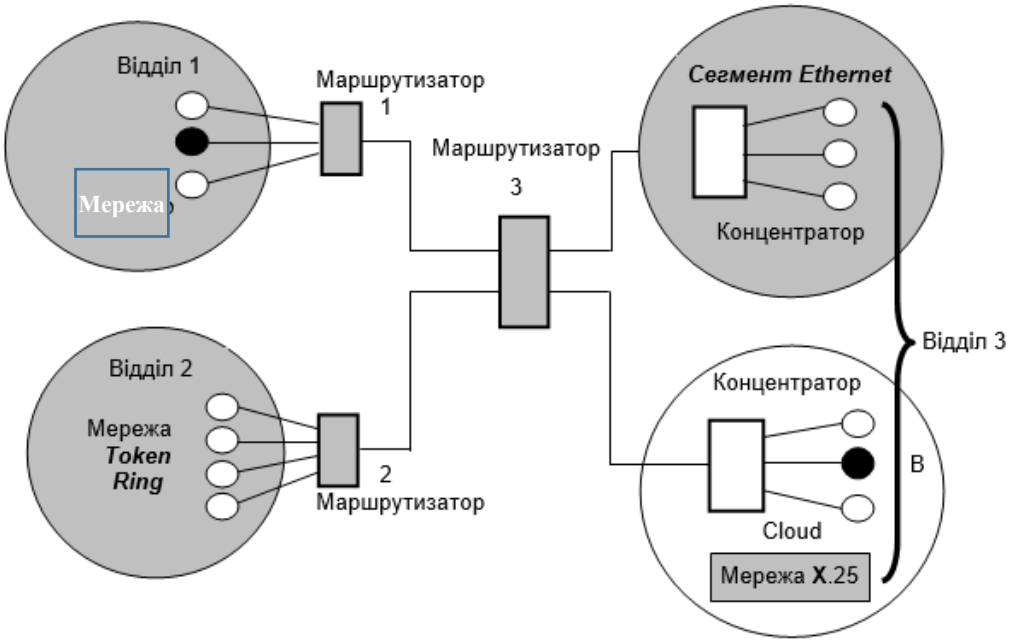


Рисунок 1.22 – Логічна структуризація мережі за допомогою маршрутизаторів

Крім перерахованих пристроїв, окремі частини мережі може з'єднувати **шлюз** (***gateway***). Звичайно основною причиною використання шлюзу в мережі є необхідність об'єднати мережі з різними типами системного і прикладного програмного забезпечення, а не бажання локалізувати трафік. Проте, шлюз забезпечує і локалізацію трафіка як певний побічний ефект.

Великі мережі практично ніколи не будуються без логічної структуризації. Для окремих сегментів і підмереж характерні типові однорідні топології базових

технологій, і для їхнього об’єднання завжди використовується устаткування, що забезпечує локалізацію трафіка: мости, комутатори, маршрутизатори і шлюзи.

1.2 Еталонна модель взаємодії відкритих систем

1.2.1 Характеристика та призначення рівневих протоколів

Обслуговування користувача – це реалізація визначеного набору функцій, що належать до передачі даних і зв’язані з такими елементами:

– мовою (включаються і функції перетворення форматів, трансляції і редагування);

– дисципліною діалогу для керування потоком даних (наприклад, послідовністю роботи, чеканням відповіді);

– керуванням передачею даних, у тому числі керуванням швидкістю передавання даних з урахуванням наявності засобів обробки потоків даних в абонента на обох кінцях лінії і керуванням послідовністю передавання для забезпечення вірогідності даних, що передаються;

– транспортуванням даних (включається проходження сигналів у більш-менш складній мережі між пристроями, кожен з яких має деяку адресу в мережі).

Групові діалоги або сесії можуть здійснюватися паралельно між будь-яким одним елементом та іншими елементами. Для забезпечення керованого використання мережі необхідне введення послуг для користувача з боку мережі. Щоб обслуговувати мережу і керувати нею, потрібні й інші служби мережі, звичайно недоступні і невідомі користувачеві.

Мережна архітектура систем регламентує набір функцій передачі даних, що розподілені по всій мережі, вона також визначає формати і протоколи, які зв’язують ці розподілені функції між собою. Мета створення мережної архітектури полягає в досягненні надійної передачі даних між програмами, операторами, процесами й запам’ятовуючими пристроями, розташованими в будь-якому пункті мережі. Проте з цього не витікає, що всі функції, які виконуються мережею, повністю регламентуються її архітектурою. Доцільніше, щоб деякі функції і (або) адаптери створювалися розробниками апаратури або замовниками. У рамках архітектури в першу чергу уніфікуються ключові формати і протоколи, які потрібні для забезпечення передачі даних.

Архітектура зв’язку в розподілених системах створювалася в два етапи. На першому були розроблені однорідні мережні архітектури, призначені для застосування однотипного устаткування. Наприклад, мережна архітектура SNA (System Network Architecture), створена фірмою IBM, була покладена в основу однієї з перших глобальних обчислювальних мереж. На другому етапі за основу прийнято базову еталонну модель OSI (ВВС – взаємодії відкритих систем), що була першим стандартом, розробленим Міжнародною організацією стандартизації.

Концепції мережних протоколів розвивалися протягом останніх двадцяти років і були спрямовані на забезпечення:

– логічної декомпозиції складної мережі на менші, зрозуміліші частини (рівні); стандартних інтерфейсів між мережними функціями, наприклад стандартних інтерфейсів між модулями програмного забезпечення;

– симетрії стосовно функцій, реалізованих у кожному вузлі мережі. Кожний рівень у деякому вузлі мережі виконує ті ж самі функції, що й аналогічний рівень в іншому вузлі:

– забезпечення засобів передбачення змін і керування змінами, які можуть бути внесені в мережну логіку (програмне забезпечення або мікропрограми);

– реалізація простої стандартної мови комунікації розробників мереж, адміністраторів, фірм-постачальників і користувачів, яка використовується під час обговорення мережних функцій.

Можна зробити висновок, що загальна проблема зв'язку, яка полягає в забезпеченні своєчасної, правильної та розпізнаваної доставки даних кінцевому користувачеві, зайнятому в сеансі зв'язку в мережі або в декількох мережах, поділяється на дві частини. Перша стосується мережі зв'язку: дані, що передаються кінцевому користувачеві з мережі, повинні надійти за призначенням в правильному вигляді та своєчасно. Другу частину проблеми – дані, що надійшли зрештою за призначенням кінцевому користувачеві, повинні розпізнаватись і мати належну форму для їх правильного використання, – можна вирішити введенням протоколів високого рівня. Повна архітектура, орієнтована на кінцевого користувача, містить у собі і мережні протоколи і протоколи високого рівня. Як приклад, на рисунку 1.23 показано схему зв'язку між користувачами А і В через проміжний вузол мережі. До цього вузла можуть бути приєднані кінцеві користувачі, а з ними можуть бути зв'язані протоколи високого рівня, але функцією проміжного вузла є тільки надання відповідних мережних послуг.

У свою чергу, дві групи протоколів – ті, що надають мережні послуги, і протоколи високого рівня – зазвичай поділяються далі на окремі рівні. Кожний рівень вибирається для надання визначеної послуги за змістом названих основних завдань: правильністю і своєчасністю доставлення даних у формі, за якою їх можна розпізнати. Шляхом розробки еталонної моделі взаємодії відкритих систем була побудована концепція, при якій кожний рівень надає послуги вищестоящому рівню.

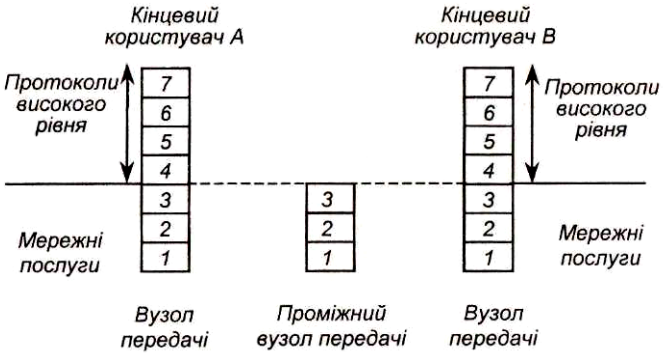


Рисунок 1.23 – Архітектура багаторівневого зв'язку

1.2.2 Рівні еталонної моделі взаємодії відкритих систем

Модель рівневих протоколів взаємодії відкритих систем є семирівневим стандартом. Рівні в мережній моделі, запропонованій ISO, наведено на рисунку 1.24.

Міжнародною організацією стандартизації розроблено базову еталонну модель ВВС для визначення рівневих мереж і рівневих протоколів. Ця модель привернула велику увагу в усьому світі і була реалізована багатьма фірмами – виробниками засобів зв'язку.

Метою моделі ВВС є: стандартизація обміну даними між системами; усунення будь-яких технічних перешкод для зв'язку систем; усунення труднощів «внутрішнього» опису функціонування окремої системи; визначення точок взаємосполучення для обміну інформацією між системами; звуження діапазону можливостей послуг для того, щоб підвищити здатність обміну даними між користувачами без зайвих накладних витрат на переговори і переклад; забезпечення розумної відправної точки відходу від стандартів, якщо вони не задовольняють усіх вимог.

Найнижчий рівень називається *фізичним*. Функції цього рівня забезпечують активізацію, підтримку і деактивізацію фізичного ланцюга між кінцевим устаткуванням даних (КУД) і апаратурою каналу даних (АКД) або фізичним середовищем. Для фізичного рівня опубліковано велику кількість стандартів. Найбільш відомими є RS-232C і інші рекомендації МСЕ.

Канальний рівень (рівень ланки даних) відповідає за передачу даних по каналу. Він забезпечує: синхронізацію даних для розмежування потоку бітів фізичного рівня; вид подання бітів; визначення гарантій прибуття даних в приймальне КУД; керування потоком даних, щоб КУД не перевантажувалися в будь-який момент часу занадто великою кількістю даних.

Одна з найважливіших функцій цього рівня полягає у виявленні помилок передачі і забезпеченні механізму відновлення даних у випадку їх втрати, дублювання або за наявності помилок у даних.

Мережний рівень визначає інтерфейс КУД користувача з мережею пакетної комутації, взаємодію двох пристроїв КУД один з одним у мережі пакетної комутації, а також маршрутизацію в мережі і зв'язок між мережами (інтермережний протокол). Цей рівень детально визначений і має велику кількість функцій. Протоколи X.25 або IP реалізують цей рівень.

Транспортний рівень забезпечує інтерфейс між мережею передачі даних і верхніми трьома рівнями. Саме цей рівень надає користувачеві можливості одержання сервісу визначеної якості (і вартості) від самої мережі (тобто мережного рівня). Він проектується таким чином, щоб відокремити користувача від деяких фізичних і функціональних аспектів пакетної мережі, а також забезпечує наскрізну звітність у мережі.

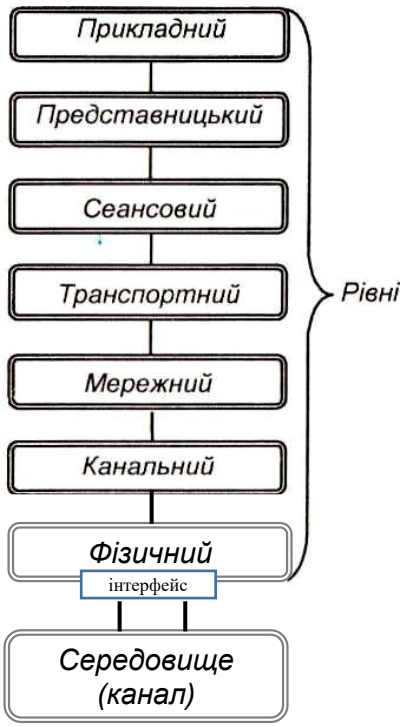


Рисунок 1.24 – Рівні в мережній моделі, запропонованій ISO

Сеансовий рівень служить інтерфейсом користувача з рівнем транспортних послуг. Цей рівень забезпечує засоби організації обміну даними між користувачами. Користувачі можуть вибрати тип синхронізації і керування, що вимагаються від рівня, такі як:

– почерговий (напівдуплексний) або одночасно двоспрямований (дуплексний) діалог;

– точки синхронізації для проміжного контролю і відновлення при передачі файлів;

– аварійне закінчення і рестартування діалогів;

– нормальна і прискорена передача даних.

Сеансовий рівень має спеціальні послуги, примітивні і протокольні блоки даних, що визначені в документах ISO і MCE.

Представницький рівень даних визначає синтаксис даних у моделі. Він не пов'язаний зі значенням або семантикою даних. Його головна роль полягає в тому, щоб приймати типи даних (знак, ціле число) з прикладного рівня і потім узгоджувати з рівнем того ж рангу синтаксичне подання (таке, як телетекс, відеотекс і т. д.). Рівень подання забезпечує відображення даних на віртуальному терміналі, а також надання таких послуг, як дозвіл прийому електронного повідомлення від рівня додаткових програмних продуктів і узгодження з одноранговим рівнем виду подання сторінки (наприклад, для друкарського набору), для прикладного рівня іншого вузла користувача.

Прикладний рівень призначений для підтримки прикладного процесу кінцевого користувача. На відміну від рівня подання даних цей рівень має справу із семантикою даних. Рівень містить сервісні елементи для підтримки

прикладних процесів, таких як керування різноманітними процесами, обмін фінансовими даними, діловими і довідковими даними (наприклад, система обробки повідомлень). Прикладний рівень також підтримує концепції віртуального терміналу і віртуального файлу.

1.2.2 Структура повідомлення у моделі ВВС

Багаторівнева організація керування процесами в мережі породжує необхідність модифікувати на кожному рівні повідомлення стосовно функцій, реалізованих на цьому рівні. Модифікація виконується за схемою взаємодії процесів, поданою на рисунку 1.25. Даним, що передаються у формі повідомлення, надаються заголовок і закінчення, в яких міститься інформація, необхідна для опрацювання повідомлення на відповідному рівні: покажчики типу повідомлення, адреси відправника, одержувача, каналу, порту і т.д.

Заголовок і закінчення називаються обрамленням повідомлення (даних). Повідомлення, сформоване на рівні n+1, при обробці на рівні n супроводжується додатковою інформацією у вигляді заголовка $З_n$ і закінчення $К_n$. Це ж повідомлення, надходячи на нижчий рівень, у черговий раз забезпечується додатковою інформацією – заголовком $З_{n-1}$ і закінченням $К_{n-1}$. У разі передавання від нижчих рівнів до вищих повідомлення звільняється від відповідного обрамлення. Таким чином, кожний рівень оперує власними заголовком і закінченням, а послідовність символів, що знаходиться між ними, розглядається як цільні дані більш високого рівня. За рахунок цього забезпечується незалежність даних, що стосуються різних рівнів керування передачею повідомлень.

Надання повідомленню обрамлення – процедура, аналогічна вкладанню листа в конверт, який використовується у поштовому зв’язку. Всі дані, необхідні для передачі повідомлення, вказуються на конверті. При передачі цього повідомлення на нижчий рівень воно вкладається в новий конверт, забезпечений відповідними даними. Повідомлення, що надходить у приймальну систему, проходить від нижніх рівнів до верхнього. Засіб керування певного рівня оперує даними, зазначеними в обрамленні, нібито як з даними на конверті. При передаванні на вищий рівень повідомлення звільняється від «конверта», внаслідок чого на наступному рівні обробляється черговий (внутрішній) «конверт». Таким чином, кожний рівень керування оперує не з самими повідомленнями, а з «конвертами», в які «упаковані» повідомлення. Тому склад повідомлень, що формуються на верхніх рівнях, ніяк не впливає на функціонування нижніх рівнів керування передачею. Процес передавання повідомлення відбувається послідовно з 7-го рівня на 1-й, а процес прийому – з 1-го на 7-й.

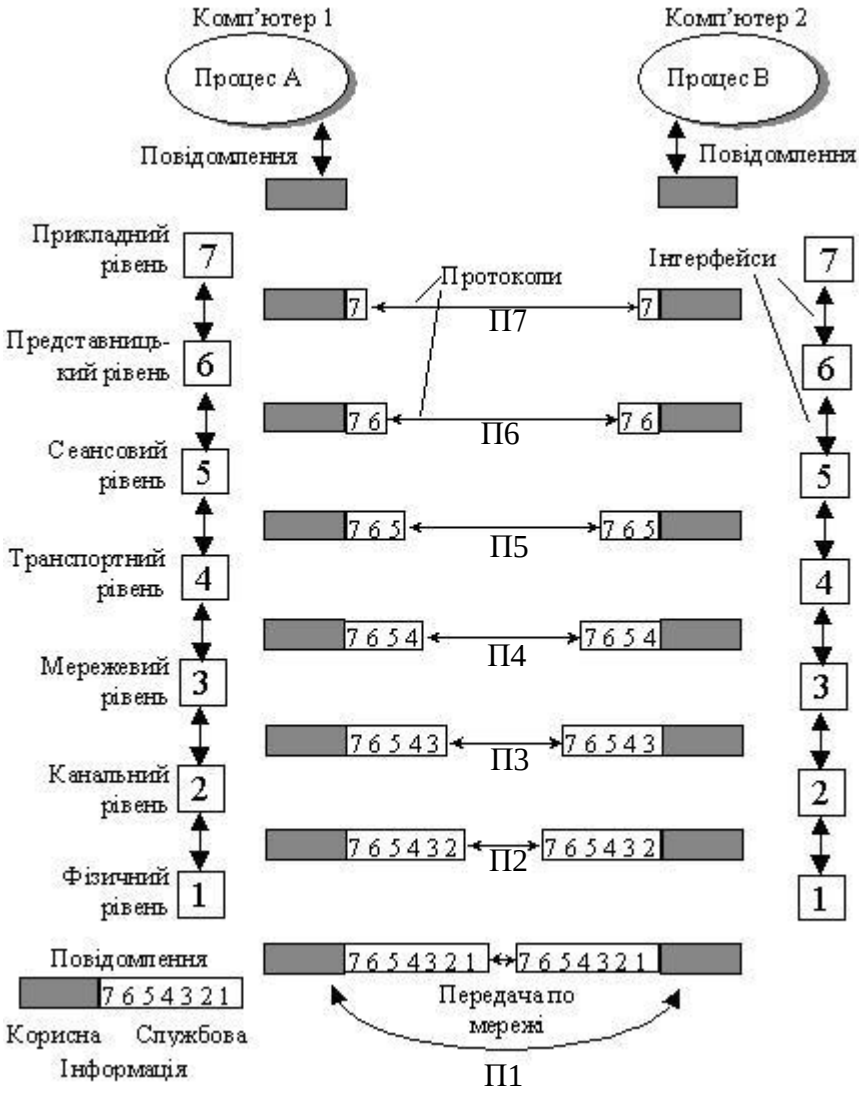


Схема взаємодії процесів на базі мережних протоколів і інтерфейсів

1.2.3 Протоколи в інформаційно-комунікаційних мережах та зв'язок між рівнями

Обмін повідомленнями (даними) допускається тільки між процесами одного рівня. Це означає, що прикладний процес може взаємодіяти тільки з прикладним процесом, а процеси керування передавання повідомлень на рівнях 1, 2,... – тільки з процесами однойменних рівнів. Ця схема взаємодії процесів (рисунок 1.24), як і процедура обрамлення повідомлень, – необхідна умова логічної незалежності рівнів інформаційних мереж.

Прикладний процес у системі А (рівень 7) формує повідомлення прикладного процесу в системі В, пристосовуючись тільки до логіки взаємодії цих двох прикладних процесів, але не до організації мережі. Фактично повідомлення, сформовані процесом у системі А, проходять послідовно через рівні 6, 5,..., 1, зазнаючи процедури послідовного обрамлення, передаються по

каналу зв’язку, і потім через рівні 1, 2,..., 6, на яких з повідомлень послідовно знімається обрамлення, надходять до процесу в системі В повністю розконвертованими. Аналогічно процес керування транспортуванням повідомлень у базову мережу (рівень 4) відправляє власні дані в обрамленні-повідомлення. Всі дані, що знаходяться поза обрамленням, не мають ніякого значення для цього процесу. Таким чином, процеси одного рівня в різних системах обмінюються даними в основному за допомогою заголовків і закінчень повідомлень. Системний процес може послати власне повідомлення іншому процесу такого ж рівня у встановленому порядку. При цьому весь текст повідомлення буде належати однойменному процесові в іншій системі. Такі повідомлення називаються керуючими і використовуються в основному на рівнях 2-5.

Процедура взаємодії процесів однойменних рівнів двох різних систем на основі обміну повідомленнями (даними) називається **протоколом**. Для процесів кожного рівня використовуються протоколи П1, П2,..., П7.

Процедура взаємодії різних (сусідніх) рівнів в одній системі називається **інтерфейсом**.

Описуючи протокол, прийнято виділяти його логічну і процедурну характеристики. Логічна характеристика протоколу – структура (формат) і зміст (семантика) повідомлень. Логічна характеристика задається переліком типів повідомлень та їх змістом. Правила виконання дій, запропонованих протоколом взаємодії, називаються процедурною характеристикою протоколу. Ця характеристика може зображуватися в різній математичній формі: операторними схемами алгоритмів, автоматними моделями, мережами Петрі та ін.

Логіка організації інформаційної мережі найбільшою мірою визначається протоколами, що встановлюють як тип і структуру повідомлень, так і процедури їх обробки – реакцію на вхідні повідомлення і генерацію власних повідомлень. Число рівнів керування і типи використовуваних протоколів багато в чому визначають архітектуру мережі.

Як було відзначено вище, кожний рівень є постачальником сервісу і може складатися з декількох сервісних функцій. Наприклад, один із рівнів може забезпечувати сервісні функції за кодовими перетвореннями.

Функція – це деяка підсистема рівня (деяка реальна підпрограма в якійсь програмі). Кожна підсистема може, складатися з логічних об’єктів – деяких спеціалізованих модулів.

Основна ідея обміну між вузлами полягає в тому, що на кожному рівні додаються послуги цього рівня до послуг, які забезпечуються нижніми рівнями. Отже, верхній рівень, що взаємодіє безпосередньо з додатковим програмним продуктом кінцевого користувача, має повний набір послуг усіх нижніх рівнів. Верхні рівні диктують нижнім, які послуги мають бути викликані.

1.2.4 Розподіл обладнання інформаційно-комунікаційних мереж за моделлю ВВС

Інформаційну мережу за характером взаємодії відкритих систем можна подати у вигляді трьох типів систем: абонентські, адміністративні й асоціативні.

Абонентські системи призначені для обробки прикладних процесів користувачів, тому в області взаємодії вони ніби розсікаються на сім рівнів і на базі комунікаційної підмережі мають структуру, показану на рисунку 1.26. Паралельно в системі реалізується ієрархія протоколів, що підтримує прикладні процеси керування мережею. Ці протоколи можуть бути такими ж, як і в ієрархії, що підтримує прикладні процеси користувачів, але можуть і відрізнятися від них. Усі рівні в системі зв'язані процесом керування системою.

Якщо є така можливість, то слід звільнити користувальницьку ЕОМ абонентської системи від виконання функцій області взаємодії, щоб надати їй можливість ефективно виконувати прикладні процеси. З цією метою абонентську систему поділяють на дві частини (рисунок 1.26): термінальне обладнання і станцію. Термінальне обладнання є основною частиною системи, що виконує прикладні процеси і, якщо є потреба, то і протоколи верхніх рівнів. Станція – допоміжна частина системи, вона реалізує протоколи нижніх або всіх рівнів. За кількістю протоколів, що реалізуються, станцію називають канальною, транспортною або абонентською.

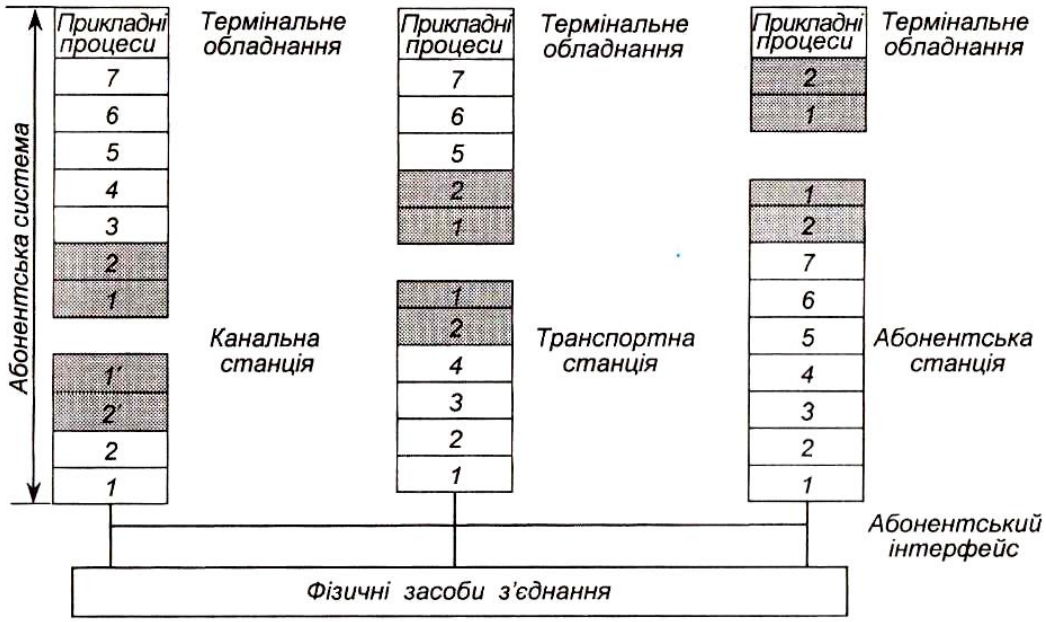
Канальна станція виконує протоколи 1 і 2, транспортна – протоколи 1…4. Абонентська станція реалізує всі сім рівнів області взаємодії відкритих систем.

Рисунок 1.26 – Структура абонентської й адміністративної систем

Станція й термінальне обладнання з'єднуються каналом або шиною. В обох випадках це з'єднання має бути подано спеціальним фізичним (1) і канальним (2) протоколами. Перший з них визначає характеристики каналу (шини), а другий описує процедури керування каналом (шиною) і передавання ними блоків даних. Спеціальні протоколи (1′, 2′) не є стандартними ISO (ВВС).

Вони залежать від конкретних обраних каналів (шин), методів зв'язку зі станціями.

Канальна станція (рисунок 1.27) є найпростішою, тому що реалізує лише протоколи двох рівнів (1, 2) взаємодії.



Поділ абонентської системи на термінальне обладнання та станцію

Проте ця простота вимагає значного завантаження абонентської частини, яка має виконувати функції, що описуються протоколами інших п'яти рівнів.

Ефективною є абонентська станція, яка цілком розвантажує термінальне обладнання від виконання функцій, що забезпечують взаємодію в мережі прикладних процесів. Проте в складному термінальному обладнанні часто працюють кілька комплексів прикладних процесів. Обмін же інформацією між ними відбувається через сеансовий рівень. Тому в тих випадках, коли рівень 5 знаходиться в станції, робота термінального обладнання залежить від надійності, завадостійкості та пропускної спроможності каналу (шини) і станції, що є не завжди прийнятним.

Тому найчастіше використовується транспортна станція. Вона виконує всі функції, пов'язані з передаванням інформації між комплексами термінального обладнання через усю комунікаційну підмережу. Що стосується термінального обладнання, то воно забезпечує виконання прикладних процесів, які підтримуються прикладним, представницьким і сеансовим протоколами.

Абонентські системи є основними компонентами інформаційної мережі. Ці системи будуються на основі різноманітних ЕОМ, що виготовляються виробниками різних країн. Тому для кожного типу комутаційної підмережі розробляється абонентський інтерфейс (рисунок 1.27), що визначає параметри і процедури взаємодії всіх абонентських систем із комунікаційною підмережею.

Адміністративна система має ту ж структуру, що й абонентські (рисунок 1.26 та рисунок 1.27), але замість прикладних процесів користувачів виконуються прикладні процеси керування мережею або її частиною.

Асоціативна система на відміну від абонентської й адміністративної не здійснює обробки інформації для потреб користувачів і керування мережею. Вона призначена для з'єднання в одне ціле частин інформаційних мереж і забезпечення взаємодії цих мереж між собою.

За характеристиками об'єднуваних частин мереж і різних мереж розрізняють чотири типи асоціативних систем (рисунок 1.28). Найскладнішою з них є шлюз. Він забезпечує взаємодію двох або більше інформаційних мереж із різними наборами протоколів семи рівнів. Так, на рисунку 1.28 показано два набори: 1′-7′ і 1"-7", зв'язані спеціальними прикладними процесами. Ці процеси перетворюють один семирівневий набір протоколів в інший, забезпечуючи необхідну взаємодію.

Слід зазначити, що шлюзи найчастіше використовуються в тих випадках, коли потрібно об'єднати інформаційні мережі, створені за різними фірмовими стандартами. Коли ж проектується група мереж відповідно до стандартів ВВС, доцільним є інший підхід: в мережах, що з'єднуються, протоколи рівнів 4…7 слід реалізовувати однаковими. Це дозволяє для з'єднання мереж використовувати не шлюзи, а більш асоціативні системи – маршрутизатори та мости.

Функція маршрутизатора – забезпечення взаємодії комунікаційних підмереж. Останні характеризуються лише трьома рівнями протоколів. Тому логічна структура маршрутизатора має вигляд, показаний на рисунок 1.28. Як бачимо, маршрутизатор не має протоколів рівнів 4…7 і є прозорим для них. Його завданням є перетворення протоколів трьох нижніх рівнів. Іноді в інформаційних мережах маршрутизатори зв'язують частини комунікаційної підмережі, в яких використовуються однакові протоколи рівнів 1…3. Такі маршрутизатори мають назву вузлів комутації пакетів. Перетворення протоколів у них не виконується: мережні процеси здійснюють лише комутацію і маршрутизацію інформації. У з'єднуваних вузлами підмережах має здійснюватися загальна адресація абонентських систем.

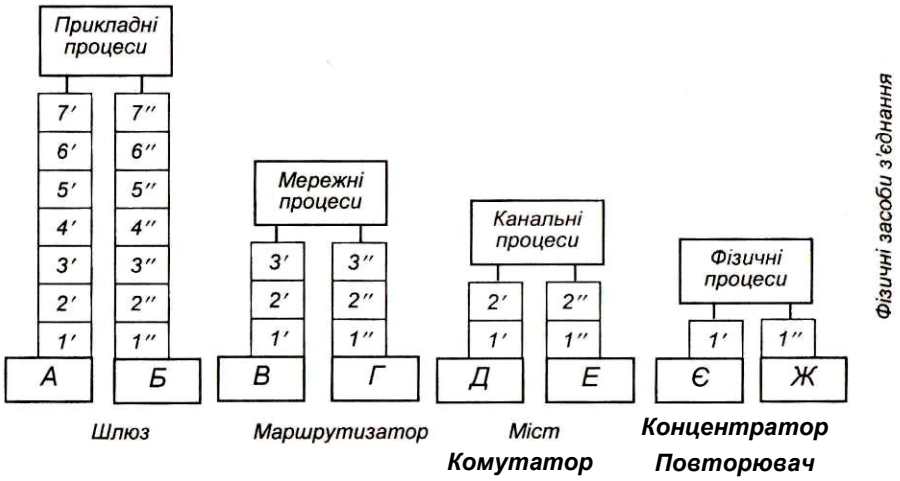


Рисунок 1.28 – Типи асоціативних систем

Мости (рисунок 1.28) призначені для з’єднання частин мереж, різноманітних типів каналів передачі даних. Будь-який канал визначається протоколами рівнів 1 і 2, тому логічна структура моста має дворівневу структуру. Канальні процеси в даному випадку перетворюють протоколи обох рівнів. У разі використання мостів у з’єднуваних підмережах мають бути погоджені структура адреси і розмір кадрів.

Більш складні, або інтелектуальні мости, крім зазначених функцій, виконують також роль фільтрів, що не пропускають пакети, не адресовані іншій частині мережі.

У кожному інтелектуальному мості міститься невеличка база даних, до якої вносяться адреси систем обох підмереж, контрольна сума, призначена для перевірки кадру, що використовується не тільки на вході моста, але й на його виході. Це дозволяє запобігати появі помилок усередині моста. Завдяки простоті виконуваних функцій застосовуються мости з відносно нескладною структурою, вони працюють з високою швидкістю. Формати блоків даних, що передаються, і розміри цих блоків міст не змінює.

Отже, асоціативні системи реалізують міжмережні, мережні, канальні та фізичні процеси. Вони виконують функції, у тому числі й перетворення, потрібні для з’єднання частин мереж або цілих мереж.

Об’єднання мереж здійснюється установленням з’єднання або без нього. Об’єднання з установленням з’єднання дає змогу заздалегідь розподіляти буфери та інші ресурси системи. У цьому випадку забезпечується просте і надійне керування потоками інформації, що проходять з однієї мережі в іншу, а також здійснюються повідомлення про втрату блоків даних і упорядкування послідовностей цих блоків. Проте організація і підтримка міжмережних з’єднань потребує виконання складних протоколів.

Об’єднання мереж без установлення між ними з’єднань характеризується простотою протоколів і високою швидкістю роботи асоціативної системи. Проте цей спосіб не має тих переваг, які притаманні об’єднанням з установленням з’єднання. Тому для забезпечення якісного передавання інформації абонентські системи обох мереж повинні мати потужні версії транспортних протоколів.

Відповідно до наведеного вище при об’єднанні мереж використовуються дві моделі. Перша з них, протокольна, характеризується тим, що у верхній частині мережного рівня розташовується міжмережний протокол. Завдяки його наявності обидві об’єднуючі мережі можуть працювати практично як одна загальна мережа. У цьому випадку асоціативна система забезпечує комутацію інформації, що проходить через систему. Другою моделлю об’єднання мереж є естафетна. У такій моделі міжмережний протокол практично відсутній і передавання керуючої інформації через асоціативну систему не відбувається. Внаслідок цього в кожній мережі повинно проводитися розпізнавання глобальних адрес, на основі якого вибираються маршрути передавання інформації.

1.3 Принципи та технології комутації. Мережне встаткування

1.3.1 Принципи комутації

Комутація є необхідним процесом з'єднання вузлів між собою, що дозволяє скоротити кількість потрібних ліній зв'язку і підвищити завантаженість (ефективність використання) каналів зв'язку. Практично неможливо надати кожній парі вузлів виділену лінію зв'язку, тому в мережах завжди застосовується той або інший спосіб комутації абонентів, що використовує наявні лінії зв'язку для передачі даних від різних вузлів.

Для локальних мереж застосовують комутацію пакетів, коли всі передані користувачем дані розбиваються передавальним вузлом на невеликі частини – пакети (які інкапсулюються в кадри). Кожен пакет оснащується заголовком, в якому вказується службова інформація, зокрема, адреса вузла-одержувача та номер пакету. Передача пакетів по мережі відбувається незалежно один від одного. Комутатори такої мережі мають внутрішню буферну пам'ять для тимчасового зберігання пакетів, що дозволяє згладжувати пульсації трафіка на лініях зв'язку між комутаторами.

Комутатор LAN підтримує таблицю, за допомогою якої визначає, як пересилати трафік через комутатор. Ця таблиця зберігається в асоціативній пам'яті (CAM – Content-addressable memory), яка є особливим типом пам'яті, що використовується у високошвидкісних пошукових застосунках. З цієї причини таблиця MAC-адрес іноді називається CAM-таблицею. Комутатор пересилає трафік на основі номеру вхідного інтерфейсу і MAC-адреси призначення кадру.

Складові принципової схеми комутатора (інтегральні мікросхеми, інші електронні елементи та прикладне програмне забезпечення) забезпечують яке контролює шляхи передавання даних через комутатор.

Щоб комутатор знав, який інтерфейс використовувати для передавання кадру, йому спочатку потрібно дізнатись, які пристрої підключені з боку кожного інтерфейсу. При надходженні кожного кадру Ethernet на комутатор виконується такий двоетапний процес.

Крок 1. Навчання – вивчення MAC-адреси джерела

При отриманні кожного кадру комутатор перевіряє його на наявність нової інформації. Для цього перевіряється MAC-адреса джерела (відправника) кадру та номер інтерфейсу, через який кадр надійшов до комутатора.

- Якщо у таблиці MAC-адрес немає MAC-адреси джерела, то вона та відповідний номер вхідного інтерфейсу додаються до таблиці.
- Якщо MAC-адреса джерела вже існує, комутатор оновлює таймер для цього запису.

За замовчуванням більшість комутаторів зберігають запис у таблиці протягом п'яти хвилин. Якщо MAC-адреса джерела присутня в таблиці, але на іншому інтерфейсу, комутатор розглядає це як новий запис.

Крок 2. Пересилання – перевірка MAC-адреси призначення

Якщо MAC-адреса призначення є адресою індивідуальної розсилки, комутатор буде шукати відповідність між MAC-адресою призначення кадру та записом у його таблиці MAC-адрес:

- Якщо MAC-адреса призначення наявна в таблиці, комутатор відправить кадр через визначений інтерфейс.
- Якщо MAC-адреса призначення відсутня у таблиці, комутатор відправить кадр через усі інтерфейси за винятком вхідного інтерфейсу. Це називається індивідуальна розсилка невідомому отримувачу (unknown unicast).
- Якщо MAC-адреса призначення є широкомовною або адресою групової розсилки, кадр також передається через усі інтерфейси, крім вхідного інтерфейсу.

У сегментах Ethernet влаштованих на основі *концентратора* мережні пристрої змагаються для доступу до спільного середовища. Сегменти мережі, в яких пристрої спільно можуть використовувати смугу пропускання, називаються ***доменами колізій***. Якщо два або більше пристроїв в одному домені колізій одночасно намагаються передавати дані, виникає ***колізія*** – накладання двох та більше кадрів від станцій, що намагаються передати кадр в один і той же момент часу.

У сегментах Ethernet на основі *комутатора* можливість виникнення колізій визначається режимом роботи інтерфейсів комутатора. Коли інтерфейси комутатора працюють у повнодуплексному режимі, доменів колізій не існує. Якщо комутаційний інтерфейс Ethernet працює у напівдуплексному режимі, кожен сегмент перебуває у своєму власному домені колізій.

За замовчуванням інтерфейси комутатора Ethernet мають повнодуплексний режим, якщо суміжний пристрій може також може у ньому працювати, а також найвищу пропускну здатність, доступну для обох пристроїв. Якщо інтерфейс комутатора під’єднаний до такого пристрою, як традиційний концентратор, що працює у напівдуплексному режимі, цей інтерфейс працюватиме в напівдуплексному режимі. У такому випадку, комутаційний інтерфейс буде частиною домену колізій.

Сукупність з’єднаних між собою комутаторів формує єдиний ***широкомовний домен***, до нього належать усі пристрої локальної мережі, які отримують кадри широкомовної розсилки від вузла. Коли комутатор отримує широкомовний кадр, він пересилає кадр з усіх своїх інтерфейсів, за винятком вхідного інтерфейсу, на якому широкомовний кадр був отриманий. Кожен пристрій, під’єднаний до комутатора, отримує копію широкомовного кадру і обробляє її. Тільки пристрій мережного рівня, наприклад, маршрутизатор, здатен розділити широкомовний домен Рівня 2. Маршрутизатори використовуються для сегментації доменів широкомовної розсилки, але вони також відокремлюють домени колізій.

1.3.2 Протоколи та технології комутації

Комутатори дуже швидко приймають рішення щодо пересилання кадрів, оскільки використовують адресацію 2 рівня (канального) – на основі MAC-адрес. Це забезпечується прикладним програмним забезпеченням на спеціалізованих інтегральних схемах (ASIC – application-specific integrated circuit). ASICs скорочують час оброблення кадрів на пристрої і дозволяють пристрою оперувати великою кількістю кадрів без зниження продуктивності.

Комутатори Рівня 2 використовують один із двох способів комутації кадрів:

- **Метод комутації з проміжним збереженням (Store-and-forward switching)** – при використанні цього методу рішення про переадресацію кадру приймається після отримання повного кадру і його перевірки на наявність помилок за допомогою механізму завадостійкого кодування – циклічного надлишкового коду (CRC). Комутація з проміжним збереженням є основним методом комутації локальної мережі Cisco.
- **Метод наскрізної комутації (Cut-through switching)** – при наскрізний комутації процес пересилання починається після визначення MAC-адреси призначення вхідного кадру і вихідного інтерфейсу.

На відміну від наскрізної комутації, комутація із проміжним збереженням має такі дві основні особливості:

- ***Перевірка помилок*** – після отримання всього кадру комутатор порівнює значення перевірочної послідовності кадру (FCS), наведене в останньому полі, з власними розрахунками FCS. Цей процес виявлення помилок, який дозволяє переконатися в тому, що кадр не містить фізичних і канальних помилок. Якщо проблем не виявлено, комутатор пересилає кадр, в іншому випадку кадр відкидається.
- ***Автоматична буферизація*** – процес буферизації на вхідному інтерфейсі, який використовується комутаторами з проміжним збереженням, забезпечує гнучкість для підтримки будь-яких швидкостей Ethernet.

Наприклад, обробка вхідного кадру, що передається через інтерфейс Ethernet 100 Мбіт/с і призначеного для відправки на інтерфейс 1 Гбіт/с, потребує використання комутації з проміжним збереженням. При будь-якій невідповідності у швидкості між вхідним і вихідним інтерфейсами комутатор зберігає весь кадр у буфері, перевіряє FCS, пересилає його в буфер вихідного інтерфейсу і потім відправляє його.

На рисунку 1.29 показано, як приймається рішення на пересилання кадру Ethernet при комутації з проміжним збереженням.

Метод комутації з проміжним збереженням видаляє кадри, які не проходять перевірку FCS. Таким чином, він не пересилає неприпустимі (помилкові) кадри. І навпаки, в режимі наскрізної комутації можливе пересилання невідповідних кадрів, оскільки перевірка FCS не виконується. У разі високого коефіцієнта помилок (неприпустимих кадрів) у мережі, наскрізна

комутація може негативно позначитися на пропускній здатності, наповнюючи її пошкодженими і недійсними кадрами.

Рисунок 1.29 – Комутація з проміжним збереженням

В режимі наскрізної комутації комутатор може прийняти рішення про переадресацію, щойно визначить MAC-адресу призначення кадру в таблиці MAC-адрес, як показано на рисунку 1.30. Для прийняття рішення про пересилання комутатору не потрібно чекати решти частин кадру, що надходить через вхідний інтерфейс. Отже, наскрізна комутація має можливість виконувати швидку комутацію кадрів. Завдяки невеликій затримці наскрізна комутація краще підходить для ресурсномістких застосунків, що виконують високопродуктивні обчислення, для яких затримка між процесами повинна складати не більше 10 мікросекунд.

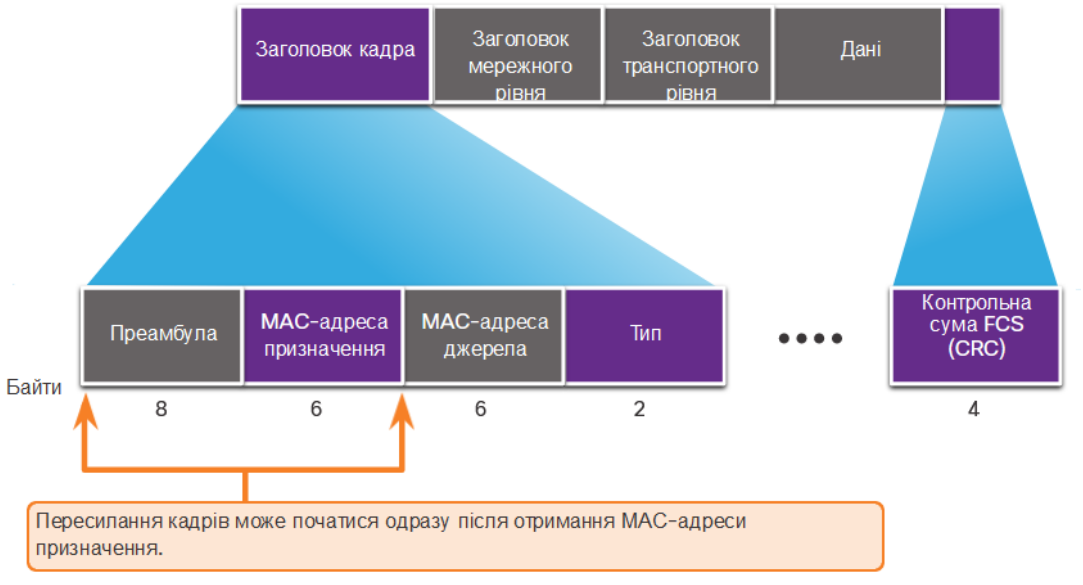


Рисунок 1.30 – Наскрізна комутація

В сучасних комутаторах реалізована удосконалена версія режиму наскрізної комутації, яка передбачає початок пересилання кадру не одразу після визначення MAC-адреси призначення, а після прийому перших 64 байтів кадру. Це незначно збільшує затримку обробки кадру, водночас, кадри, які мають розмір менше 64 байтів (так звані «карликові» кадри), виникають у результаті впливу колізій й складають більшу частину всіх пошкоджених кадрів. Тож відкидання пошкоджених кадрів (менших за 64 байти), значно скорочує кількість неприпустимих кадрів, які пересилаються мережею. Такий режим комутації називають *наскрізною комутацією з фрагментуванням*.

1.3.3 Фізичні адреси

Для ідентифікації абонентів і знаходження місця їх розташування в мережі кожному з них привласнюється ознака – адреса. Під **адресою** розуміється умовний номер, знання якого є достатнім для доставки повідомлення необхідному абонентові.

Сукупність правил присвоєння адрес абонентів, з обліком місць їх розташування й структури мережі називається *способом адресування*.

Існуючий стек протоколів пакетних мереж використовує три типи адрес:

- локальні (називаємі апаратними або фізичними адресами);
- мережні (IP-адреси);
- символьні (доменні імена).

Фізична локальна адреса – це адреса, що використовується для доставки даних в локальних межах (підмережах), які є елементами більш складної мережі. Фізична локальна адреса – це **MAC-адреса** (***Media Access Control***), яка призначається мережним адаптерам і мережним інтерфейсам маршрутизаторів, інтерфейсам ПК і усім інтерфейсам елементів мережі. MAC-адреса назначається виробником обладнання і є унікальною, так як управляється централізовано. Для усіх відомих технологій ЛОМ MAC-адреса (рисунок 1.31) має формат 48 біт (6 октетів), наприклад: 11-A0-17-3D-BC-01.

MAC-адресу можна розділити на дві частини (рисунок 1.31). У першій частині вказується *унікальний ідентифікатор виробника встаткування* (Organizationally Unique Identifier, OUI). Цей унікальний ідентифікатор надається виробникові міжнародною організацією – інститутом IEEE. Останні 24 біта MAC-адреси призначаються безпосередньо самим виробником устаткування й фактично є порядковим номером виробу.

Особливе призначення є у двох бітів (b1 і b0) першого байту MAC-адреси, фактично ці біти розташовані на 7 та 8 позиціях адреси відповідно. Восьмий біт (b0) указує, чи є адреса індивідуальною або груповою (I/G):

- 0 (індивідуальна) – адреса, асоційована з певним мережним пристроєм;
- 1 (групова) – адреса, асоційована з кількома або всіма вузлами даної мережі.

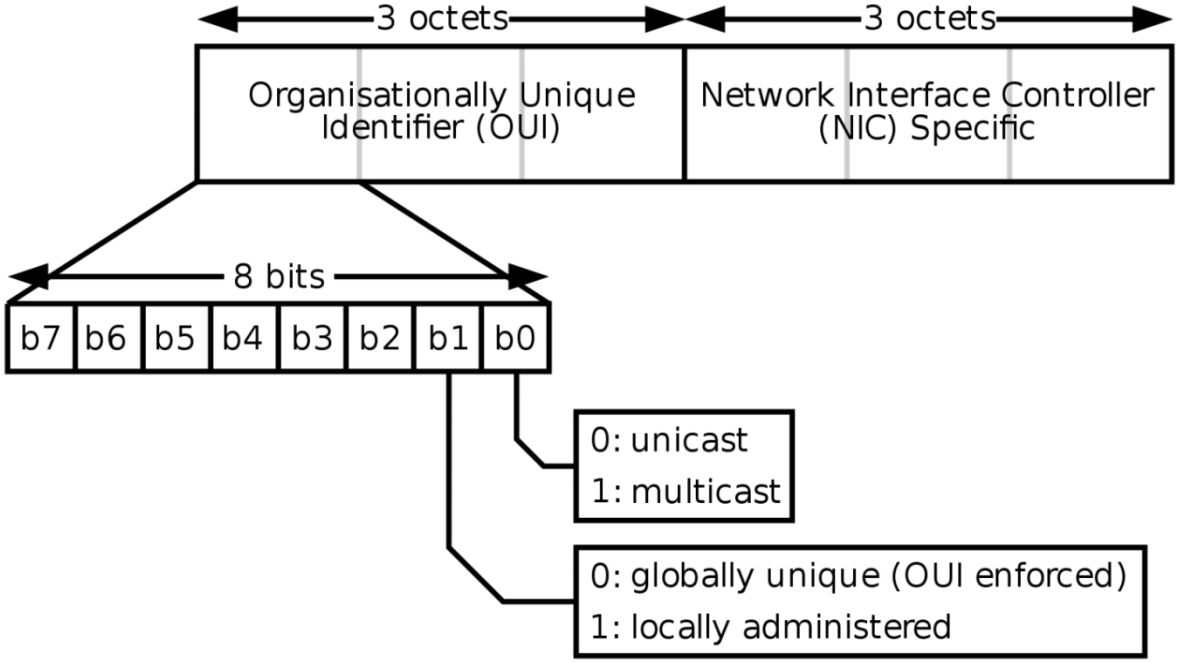


Рисунок 1.31 – Формат MAC-адреси

Існує два види групових адрес:

– *багатоадресна* або *групова* (*multicast*) – адреса, асоційована із групою вузлів мережі. Multicast адреса завжди має у другій позиції непарне шістнадцятьковий символ (1, 3, 5, 7, 9, B, D, F), найчастіше застосовані групові MAC-адреси мають на початку: 01:00:5E, 01:80:C2 та 01:00:0C.

– *широкомовна* (*broadcast*) – адреса, асоційована з усіма вузлами мережі. Її значення – (0x11) FF-FF-FF-FF-FF-FF.

Сьомий біт (b1) MAC-адреси указує, чи є MAC-адреса глобально або локально адмініструєма (U/L):

– 0 (глобально адмініструєма MAC-адреса пристрою) – вона глобально унікальна (адмініструється IEEE) і зазвичай «зашита» в апаратуру;

– 1 (локально адмініструєма MAC-адреса) – вона вибирається довільно й може не містити інформації про виробника даного встаткування (OUI). Деякі виробники мережних адаптерів підтримують можливість змінювати MAC-адреси пристрою.

1.3.4 Протокол визначення адрес ARP

ARP (англ. *Address Resolution Protocol* – протокол визначення адрес) – комунікаційний протокол мереж TCP/IP, призначений для визначення MAC-адреси (адреси канального рівня) вузла, за його IP-адресою.

ARP-таблиця для співставлення адрес

Кожен вузол мережі TCP/IP зберігає у пам’яті таблицю, яка називається ARP-таблицею. Вона містить рядки для кожного відомого вузла мережі, в рядку

містяться IP- та MAC-адреси даного вузла. Якщо потрібно знайти MAC-адресу вузла за відомою IP-адресою, то відбувається пошук запису з цією IP-адресою. Нижче наведено приклад скороченої ARP-таблиці (таблиця 1.2).

Таблиця 1.2 – Приклад ARP-таблиці

Internet Address	Physical Address	Type
223.1.2.1	08-00-39-00-2F-C3	dynamic
224.0.0.252	01-00-5e-00-00-fc	static
255.255.255.255	ff-ff-ff-ff-ff-ff	static

ARP-таблиця необхідна через те, що IP-адреси та MAC-адреси вибираються незалежно, і немає жодного алгоритму для перетворення однієї в іншу. IP-адресу вибирає адміністратор мережі з урахуванням розташування машини у мережі Інтернет. Якщо машину переміщують до іншої частини мережі, то її IP-адреса повинна бути змінена. MAC-адресу вибирає виробник мережного інтерфейсного обладнання з виділеного для нього, згідно з ліцензією, адресного простору. Якщо в машини змінюється мережний адаптер, то змінюється й MAC-адреса.

Порядок визначення адрес

У ході звичайної роботи мережна програма відправляє прикладне повідомлення, користуючись транспортними послугами протоколу TCP. Модуль TCP надсилає відповідне транспортне повідомлення через протокол IP. У результаті складається IP-пакет, який має передаватись канальному протоколу Ethernet. IP-адреса місця призначення відома прикладній програмі, модулеві TCP та модулеві IP. Необхідно на її основі знайти MAC-адресу місця призначення. Для пошуку відповідної MAC-адреси використовується ARP-таблиця. ARP-таблиця заповнюється автоматично модулем ARP по мірі отримання вузлом відповідної інформації.

Запити та відповіді протоколу ARP

Коли в існуючій ARP-таблиці немає запису про відповідність певної пари IP- та MAC-адрес, то у протоколі відбувається наступний процес:

1. Вихідний IP-пакет ставиться в чергу очікування.
2. В мережу передається широкомовний ARP-запит (таблиця 1.3), що містить в якості IP-адреси отримувача, мережеву адресу вузла, до якого потрібно відправити цей пакет.
3. Кожний мережний адаптер приймає широкомовні передачі. Канальні модулі визначають, що це ARP-запит, й транслюють його модулю ARP.
4. Модуль ARP порівнює IP-адресу отримувача, зазначену в запиті, й у випадку збіжності з власною адресою вузла, відправляє відповідь (таблиця 1.4) з зазначенням своєї MAC-адреси безпосередньо на MAC-адресу відправника запиту. У випадку незбіжності IP-адрес – запит ігнорується.
5. Відповідь одержує вузол, що зробив ARP-запит. Мережна карта цього вузла перевіряє поле типу повідомлення в Ethernet-кадрі й передає ARP-пакет

модулю ARP. Модуль ARP аналізує ARP-пакет і додає запис у свою ARP-таблицю.

Таблиця 1.3 – Приклад ARP-запиту

IP-адреса відправника	223.1.2.1
MAC-адреса відправника	08:00:5A:21:A7:22
Шукана IP-адреса	223.1.2.3
Шукана MAC-адреса	<порожньо>

Таблиця 1.4 – Приклад ARP-відповіді

IP-адреса відправника	223.1.2.3
MAC-адреса відправника	08:01:2A:2B:A7:21
IP-адреса автора запиту	223.1.2.1
MAC-адреса автора запиту	08:00:5A:21:A7:22

Якщо в мережі немає вузла із шуканою IP-адресою, то ARP-відповіді не буде й не буде запису в ARP-таблиці. Протокол IP буде відкидати IP-пакети, що направляються на цю адресу. Для протоколів верхнього рівня не існує різниці з якої причини не відправлений пакет: чи це випадок пошкодження мережі Ethernet, чи випадок відсутності вузла із шуканою IP-адресою.

1.4 Базові технології локальних мереж

1.4.1 Принципи CSMA/CD та CSMA/CA

Метод доступу до середовища передавання даних CSMA/CD

Сутність методу CSMA/CD повністю розкрита в його назві: множинний доступ з упізнаванням несучої і виявленням колізій (carrier-sense-multiply-access with collision detection). За цим методом множина комп’ютерів мережі має безпосередній доступ до фізичного середовища, побудованого за технологією загальної шини. При цьому всі комп’ютери мережі одночасно, з урахуванням затримки поширення сигналу по фізичному середовищу, одержують дані, які один з комп’ютерів почав передавати на загальну шину.

На рисунку 1.32 показана взаємодія вузлів в локальній мережі, побудованої за топологією загальної шини, а на рисунку 1.33 наведено часові діаграми передавання комп’ютерами кадрів у фізичне середовище.

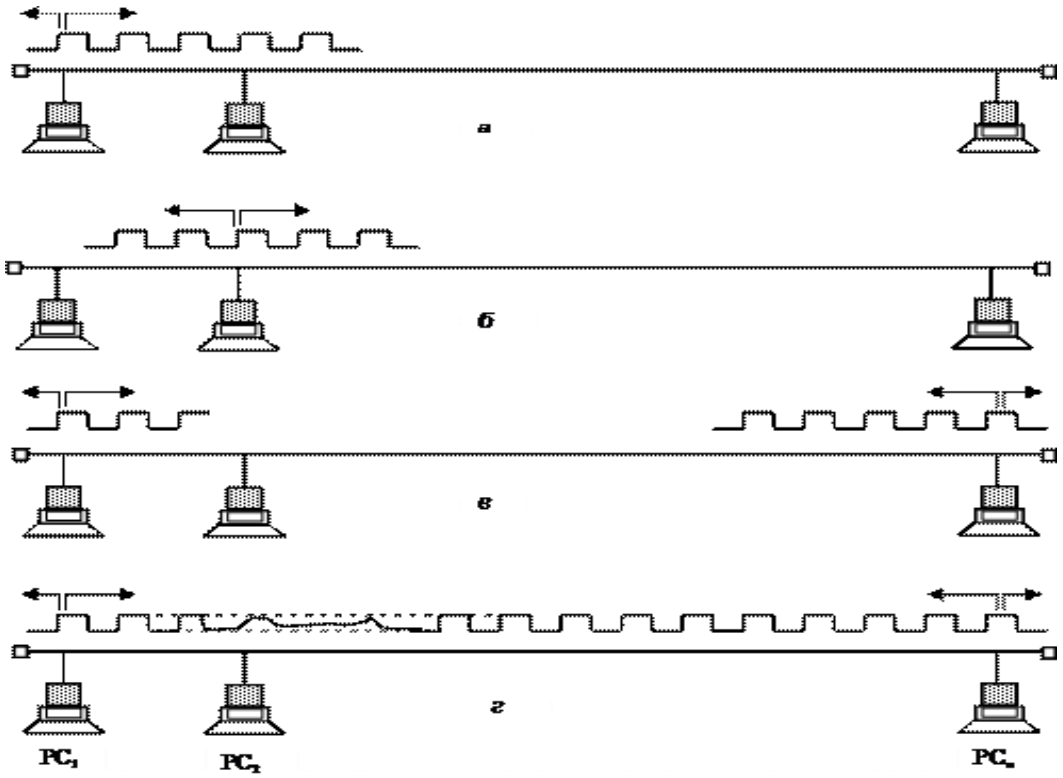


Рисунок 1.32 – Передавання комп’ютерами даних у мережу

Усі дані, які передаються мережею, форматизуються у кадри визначеної структури і забезпечуються унікальною адресою станції призначення. Щоб одержати можливість передавати кадр, станція повинна переконатися, що середовище вільне. Цього досягають прослуховуванням основної гармоніки сигналу, що також називається несучою частотою (carrier-sense, CS). Ознакою незайнятості середовища є відсутність на ній несучої частоти, що за тактової частоти 10 МГц і манчестерського способу кодування залежно від поточної послідовності одиниць і нулів становить 5…10 МГц.

В момент часу t_1 комп’ютер PC_1, прослухавши фізичне середовище і не виявивши несучої частоти, починає в момент часу t_2 передавати на загальну шину кадр даних у вигляді послідовності біт. Дані, закодовані манчестерським кодом, поширюються з певною швидкістю загальною шиною в обидва боки (рисунок 1.32, *а*). Кадр даних завжди супроводжується преамбулою довжиною 7 байт, і прапорцем початку кадру, довжиною 1 байт. Преамбула потрібна для входження приймача в побітову і побайтову синхронізацію із передавачем. Усі комп’ютери, що під’єднані до кабелю, розпізнають факт передачі кадру. Комп’ютер, який розпізнав власну адресу в заголовку кадру, записує його вміст у свій внутрішній буфер, обробляє отримані дані і передає їх нагору протоколам верхніх рівнів стека.

У момент часу t_3 верхні рівні протоколів комп’ютера PC_2 вимагають від його мережевого адаптера передати дані в мережу, але він, прослуховуючи загальну шину, виявив на ній несучу і тому залишився в стані очікування.

У момент часу t_4 комп'ютер PC_1 закінчує передавати кадр даних, і всі комп'ютери мережі витримують технологічну паузу (Inter Packet Gap) $t_{тп}$=96 bt.

У технології Ethernet прийнято всі інтервали вимірювати в бітових інтервалах. Бітовий інтервал позначається як bt і відповідає проміжку часу між появою двох послідовних біт даних у кабелі. Для швидкості 10 Мбіт/с бітовий інтервал дорівнює 0,1 мкс, чи 100 нс, тобто $t_{тп}$=9,6 мкс. Під час технологічної паузи мережеві адаптери передавача і отримувача кадру даних відновлюють свій початковий стан. Міжкадровий інтервал (технологічна пауза) потрібний також для запобігання монопольному захопленню середовища одним комп'ютером.

У момент часу t_5 комп'ютер PC_2, прослухавши фізичне середовище, не виявив несучої частоти і почав передавати свій кадр з послідовністю дій, описаних вище.

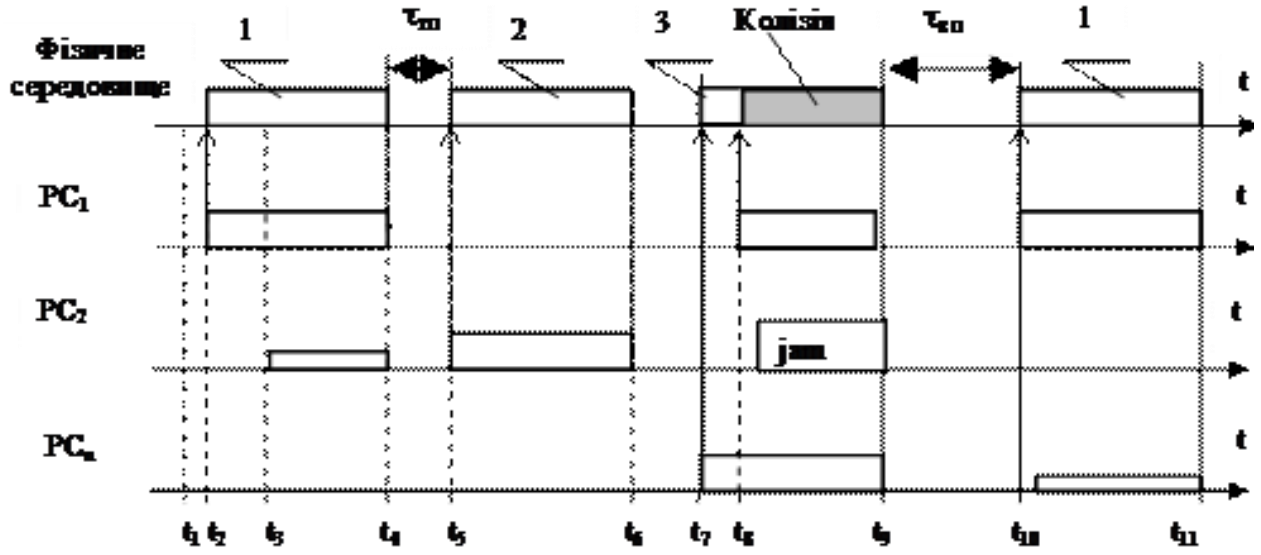


Рисунок 1.33 – Часова діаграма передавання комп'ютерами кадрів у фізичне середовище

У момент часу t_7 дані починає передавати комп'ютер PCn. У цей самий момент часу комп'ютер PC1, не виявивши на загальній шині несучої (сигнали комп'ютера PCn до нього ще не дійшли), розпочинає передавання свого кадру даних. У момент часу t_8 сигнали комп'ютерів PC_n і PC_1 зіштовхуються між собою, що призводить до їх загального спотворення. Це явище має назву «колізія». Щоб коректно обробити колізію, усі комп'ютери одночасно спостерігають за сигналами на кабелі. Першим явище колізії виявляє комп'ютер PC_2, який для її підсилення посилає у фізичне середовище спеціальну *jam*-послідовність довжиною 32 біти.

Виявивши явище колізії, всі комп'ютери мережі припиняють посилання сигналів у фізичне середовище, і настає пауза випадкової довжини, тривалість якої для кожного комп'ютера буде іншою. Після закінчення випадкової паузи комп'ютер може знову спробувати захопити середовище. Випадкову паузу $t_{вп}$ вибирають за таким співвідношенням:

$$t_{вп}=L*512 \text{ bt},$$

де L – ціле число, вибране з рівною ймовірністю з діапазону [0, 2n];

n =1,2, ..., 10 – номер повторної спроби передавання певного кадру.

Після 10-ї спроби інтервал, з якого вибирається пауза, не збільшується. Отже, випадкова пауза може набувати значення від 0 до 52,4 мс. Якщо 16 послідовних спроб передачі кадру викликають колізію, то передавач повинен припинити спроби і відкинути цей кадр.

У випадку, наведеному на рисунку 1.33, тривалість випадкової паузи комп'ютера PC_1 виявилася коротшою, ніж технологічна пауза комп'ютера PC_n і в момент t_{10} він розпочинає передавання свого кадру даних, а PC_n залишається у стадії очікування.

Тобто, метод CSMA/CD не гарантує вузлам мережі доступ до фізичного середовища. Ймовірність успішного одержання вузлом у своє розпорядження фізичного середовища залежить від завантаженості мережі Ethernet.

Carrier Sense Multiple Access With Collision Avoidance (**CSMA/CA**, «множинний доступ з контролем несучої й запобіганням колізій» або «багатостанційний доступ з контролем несучої й запобіганням конфліктів» – це мережевий протокол, у спрощеному варіанті якого:

- використовується схема прослуховування несучої хвилі;
- станція, яка збирається почати передачу, посилає *jam*-послідовність (сигнал затору);
- після тривалого очікування всіх станцій, які можуть послати *jam*-послідовність, станція починає передачу кадру;
- якщо під час передачі станція виявляє *jam*-послідовність від іншої станції, вона зупиняє передачу на відрізок часу випадкової довжини й потім повторює спробу за вище описаним алгоритмом

CSMA/CA відрізняється від CSMA/CD тим, що колізіям піддаються не пакети даних, а тільки jam-сигнали. Звідси й назва «Collision Avoidance» – запобігання колізій (саме пакетів даних).

Запобігання колізій використовується для того, щоб поліпшити продуктивність CSMA, віддавши мережу єдиному передавальному пристрою. Поліпшення продуктивності досягається за рахунок зниження ймовірності колізій і повторних спроб передачі. Але очікування *jam*-послідовності створює додаткові затримки. Запобігання колізій корисно на практиці в тих ситуаціях, коли своєчасне виявлення колізії неможливо – наприклад, при використанні радіопередавачів (WiFi).

1.4.2 Структура кадру Ethernet

Формат кадрів технології Ethernet описаний стандартами IEEE 802.2 (кадр LLC) та IEEE 802.3 (кадр MAC). Проте з урахуванням історії розвитку технології сьогодні існує чотири версії кадрів Ethernet, які мають основні та альтернативні назви і розпізнаються комутаційним обладнанням локальних мереж:

- кадр 802.3/LLC (802.3/802.2 або Novell 802/2);
- кадр RAV 802.3 (Novell 802.3);

- кадр Ethernet DIX (Ethernet II);
- кадр Ethernet SNAP.

Формат кадру 802.3/LLC, визначений стандартом IEEE 802, наведено на рисунку 1.34. Заголовок кадру 802.3/LLC є результатом об'єднання заголовків LLC і MAC, визначених в стандартах IEEE 802.3 і IEEE 802.2.

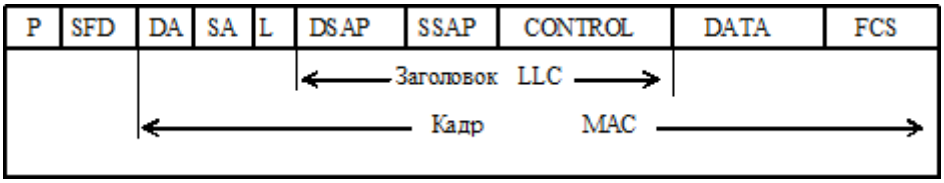


Рисунок 1.34 – Формат кадру 802.3/LLC

Стандарт IEEE 802.3 визначає такі поля кадру:

- P – преамбула (7 байт);
- SFD – прапорець початку кадру (1 байт);
- DA – адреса призначення кадру (6 байт);
- SA – адреса відправника кадру (6 байт);
- L – вказівник довжини або типу поля даних (2 байти);
- DATA – поле даних (від 46 до 1500 байт (разом з заголовком LLC));
- FSC – контрольна сума (4 байти).

Преамбула довжиною сім синхронізуючих байт 10101010 призначена для входження адаптерів комп'ютерів, під'єднаних до мережі, в синхронізацію з адаптером відправника пакета. Для манчестерського коду ця комбінація у фізичному середовищі є періодичним сигналом частотою 5 МГц.

Прапорець початку кадру являє собою двійковий код 10101011. Поява цієї комбінації вказує мережевим адаптерам вузлів мережі, які прослуховують розподілене фізичне середовище, що наступним байтом буде перший байт заголовка кадру, тобто адреса одержувача пакета.

Адреса призначення кадру може бути груповою (широкомовною) або індивідуальною, вона має довжину 6 байтів. Широкомовна адреса складається з усіх одиниць (FF-FF-FF-FF-FF-FF), а індивідуальна адреса отримувача пакета – це, як правило, MAC-адреса його мережевого адаптера, ознакою індивідуальності адреси є наявність «0» у восьмому розряді. *Адреса відправника* кадру – це його MAC-адреса довжиною 6 байт, яка також завжди має «0» у восьмому розряді.

Вказівник довжини визначає розмір поля даних у кадрі.

Поле даних призначене для розміщення і передавання мережею пакетів вищих рівнів. У керуючих і деяких ненумерованих кадрах воно може бути відсутнім, тоді в полі довжини передається тип кадру (наприклад для ARP воно має значення 0806).

Заголовок LLC (3 (4) байти) разом з полем DATA (1497 (1496) байт) входить в поле даних кадру MAC, максимальний розмір якого може становити 1500 байт. Якщо поле даних менше за 46 байт, то воно доповнюється до 46 байт полем заповнення. Мінімальний розмір поля даних 46 байт обумовлений

вимогами коректного визначення явища колізії, яке може спостерігатися в мережі, коли декілька вузлів одночасно передають дані у фізичне середовище.

Поле FSC вказує контрольну суму кадру, обчислену за певним алгоритмом (завадостійкий циклічний код). Отримувач кадру використовує контрольну суму для контролю за спотворенням даних в усіх полях від DA до DATA.

Формати кадрів RAV 802.3, Ethernet DIX та Ethernet SNAP, які не увійшли в стандарт IEEE 802, але розпізнаються комутаційним обладнанням локальних мереж, мають деякі відмінності від формату кадру 802.3/LLC. Так, у кадрів RAV 802.3 і Ethernet DIX відсутній заголовок LLC і максимальна довжина поля DATA становить 1500 байт, кадр Ethernet SNAP, крім заголовку LLC, має два додаткові поля довжиною 5 байт, а довжина поля DATA становить 1492 байти.

1.4.3 Особливості сімейства технологій Ethernet

Технологія Ethernet

Ethernet – архітектура мереж, що ґрунтується на логічній топології шини, з розподіленим середовищем передавання, методом доступу до середовища передавання CSMA/CD, за фізичною реалізацією розрізняють:

10BASE5 – Thick («товстий») Ethernet;

10BASE2 – Thin («тонкий») Ethernet;

10BASET – Twisted-pair Ethernet (Ethernet на витій парі);

10BASE-F – мережа на оптоволоконному кабелі.

Перший елемент в умовному позначенні архітектури – швидкість передавання 10 Мбіт/с; другий елемент позначає спосіб передавання: Base – пряме немодульоване передавання; третій елемент – середовище передавання (5 та 2 –коаксіальний кабель).

Ethernet на коаксіальному кабелі

Товстий Ethernet

Класичний варіант використовує товстий коаксіальний кабель RG-11 з посрібненим центральним проводом та подвійним екрануванням. Кабель має хвильовий опір 50 Ом, мале затухання та високий ступінь захисту від зовнішніх впливів. На кінцях кабелю встановлюються 50-Омні опори (термінатори), один з яких заземлюється. Для включення вузла на кабель встановлюється трансивер MAU (активний пристрій з живленням 12В), який може підключатись через Т-конектор або шляхом проколювання кабелю («вампір»).

Жорсткість кабелів створює додаткові експлуатаційні труднощі. Вартість устаткування та складність монтажу не сприяють широкому використанню цієї архітектури. Іноді «товстий» Ethernet використовують для прокладання базових (хребтових, Backbone) сегментів у процесі побудови кампусних мереж. Основні характеристики наведені в таблиці 1.5.

Тонкий Ethernet

Вважається застарілим і в сучасних комп’ютерних мережах застосовується дуже рідко, але його використання може бути виправданим у приміщеннях з великою електромагнітною завадою.

Кабель має хвильовий опір 50 Ом, середні затухання та ступінь захисту від зовнішніх впливів. Включення вузла, що завжди супроводжується розрізанням кабелю, може здійснюватися через Т-конектор або Т-подібне відгалуження від Т-конектора, яке не може перевищувати 10 см. Таке обмеження створює експлуатаційні труднощі. Відсутність контакту в будь-якому місці сегмента (дуже поширена несправність) виводить з ладу роботу всієї мережі. Оптимальний спосіб використання – для прокладання базової мережі між кабельними центрами. Основні характеристики наведені в таблиці 1.5.

Таблиця 1.5 Параметри специфікацій фізичного рівня технології Ethernet

	10Base–5	10Base–2	10Base–T	10Base–F
Тип кабелю	Товстий коаксіальний кабель RG–8 або RG–11	Тонкий коаксіальний кабель RG–58	Неекранована скручена пара UTP категорій 3, 4, 5	Багато-модовий ВОК
1. Максимальна довжина сегмента, м	500	185	100	2000
2. Максимальна віддаль між вузлами мережі (з використанням повторювачів), м	2500	925	500	2500 (2740 для 10Base–FB)
3. Максимальне число станцій в сегменті	100	30	1024	1024
4. Максимальне число повторювачів між станціями мережі	4	4	4	4 (5 для 10Base–FB)

Сучасні локальні комп’ютерні мережі майже не використовують технології, для яких фізичним носієм є коаксіальний кабель, але з розповсюдженням відеоспостереження з використанням IP-камер ця технологія отримала нове життя.

Ethernet на витій парі

Стандарт Ethernet 10BaseT використовує дві неекрановані виті пари UTP (Unshielded Twisted Pair) 3, 4 або 5 категорій. Для зв’язку між вузлами мережі необхідними є дві виті пари дротів: одна – для передавання, інша – для приймання інформації. Звичайно, замість двох кабелів по одній парі витих дротів у кожній використовують один кабель з чотирма парами дротів. Окрім економії та технічних переваг, це створює можливість переходу на більш швидкісні мережні архітектури без заміни самого кабелю.

Фізична топологія – зірка: кожен вузол мережі з’єднується зі своїм портом кабельного центру кабельним променем, що не повинен перевищувати довжини 100 м. На кінцях кабелю за допомогою спеціального обтискаючого інструмента встановлюються 8-контактні роз’єми RJ-45. Найпоширенішими є 8-ми та 16-ти портові кабельні центри, що комплектуються зовнішніми адаптерами електромережі. Основні характеристики наведені в таблиці 1.5.

Технологія Fast Ethernet

Технологію Fast Ethernet було розроблено на початку 90-х років некомерційним об'єднанням Fast Ethernet Alliance, в яке ввійшло декілька провідних фірм та інститутів в галузі технології Ethernet. Метою розробки було створення нової недорогої технології локальних мереж, яка за великої швидкості передавання даних зберегла би особливості Ethernet, зокрема її ефективність за співвідношенням ціна/якість. У 1995 році комітет IEEE 802 затвердив цю технологію стандартом 802.3u.

Технологія Fast Ethernet має такі характерні особливості:

- Швидкість передавання даних – 100 Мбіт/с.
- Метод доступу – CSMA/CD.
- Фізична топологія – ієрархічне дерево.
- Специфікації фізичного рівня:
  - 100Base–TX – дві скручені пари UTP 5-ї категорії або STP;
  - 100Base–FX – багатомодовий оптоволоконний кабель;
  - 100Base–T4 – чотири виті пари UTP 3-ї категорії;
- Максимальний діаметр мережі – 200 м.

Стандарти 100Base–TX/FX можуть працювати як в напівдуплексному, так і в повнодуплексному режимах.

Технологія Fast Ethernet використовує метод доступу CSMA/CD з тими самими алгоритмами і часовими співвідношеннями в біт інтервалах, що і технологія Ethernet. За технологією Fast Ethernet міжкадровий інтервал дорівнює 0,96 мкс, а 1bt = 10 ns.

Протоколи канального рівня мережі Fast Ethernet збігаються з протоколами канального рівня мережі Ethernet, а протоколи фізичного рівня визначаються специфікаціями фізичного середовища технології Fast Ethernet .

У табл. 1.6 наведено параметри фізичних інтерфейсів стандарту Fast Ethernet**.**

Таблиця 1.6 Фізичні інтерфейси стандарту Fast Ethernet (IEEE 802.3u) та їхні основні характеристики

Фізичний інтерфейс	**100Base-FX**	**100Base-TX**	**100Base-T4*****
Порт пристрою	Duplex SC	RJ-45	RJ-45
Середовище передачі	Оптичне волокно	Вита пара UTP Cat.5 (5e)	Вита пара UTP Cat. 3, 4, 5
Сигнальна схема	4B/5B	4B/5B	8B/6T
Бітове кодування	NRZI	MLT-3	NRZI
Кількість витих пар/волокон	2 волокна	2 виті пари	4 виті пари
Протяжність сегмента*	До 412 м (БМВ), до 2 км, дуплекс (БМВ), до 100 км (ОМВ)*	До 100 м	До 100 м

Технологія Gigabit Ethernet

У середині 90-х років назріла необхідність переведення перевантажених швидкодіючими серверами магістралей локальних мереж на більші швидкості передавання даних. Швидкості 100 Мбіт/сек, яку забезпечувала технологія Fast Ethernet, було вже недостатньо. Більшу швидкість пропонували комутатори технології ATM, але відсутність засобів міграції в інші технології LAN стримувало її впровадження. Технологію Gigabit Ethernet було розроблено у 1997 році об'єднанням Gigabit Ethernet Alliance, в яке увійшли провідні фірми та інститути в галузі LAN. Метою розробки було переведення магістралей Ethernet і Fast Ethernet на вищу швидкість передавання даних. У 1998 році комітет IEEE 802 затвердив цю технологію стандартом 802.3z.

Технологія має такі характерні особливості:

- Швидкість передавання даних на верхніх рівнях мережі – 1000 Мбіт/с.
- Збережені формати кадрів Ethernet.
- Збережений метод доступу до розподіленого середовища CSMA/CD.
- Фізична топологія – ієрархічне дерево, побудоване на одному концентраторі з діаметром до 200 м.
- Використання комутаторів з повнодуплексним режимом передавання даних.
- Фізичне середовище – оптоволоконний кабель, кручена пара UTP 5-ї категорії, подвійний коаксіал.

Технологія Gigabit Ethernet зберігає сумісність з технологіями Ethernet і Fast Ethernet. Вона використовує ті самі формати кадрів, працює в напівдуплексному і повнодуплексному режимах, підтримує на розподілених середовищах метод доступу CSMA/CD. Стандарт 802.3z використовує метод логічного кодування 8B/10B. За тактової частоти передавання сигналів у лінію 1000 МГц цей код забезпечує корисну швидкість передавання даних 800 Мбіт/с. Очікується, що найближчим часом технологія забезпечить тактову частоту 1,25 ГГц, що дозволить досягнути корисної швидкості передавання даних 1000 Мбіт/с.

Для збереження максимального діаметра 200 м мережі на одному концентраторі у напівдуплексному режимі передавання даних необхідно було збільшити час подвійного обороту сигналів, який для надійного розпізнавання колізії не повинен перевищувати часу передавання кадру мінімальної довжини. Для цього мінімальний розмір кадру було збільшено з 64 байт (без преамбули) до 512 байт (4096 біт). Це дозволило збільшити час подвійного обороту сигналу до 4096 bt. Для забезпечення мінімальної довжини кадру 512 байт поле даних доповнюється 448 байтами заборонених символів коду 8B/10B. За потреби передати декілька коротких кадрів (наприклад, квитанцій) станціям дозволено їх передавати підряд загальною довжиною до 8192 байт без міжкадрових інтервалів і доповнення до 512 байт.

З використанням комутаторів знімаються обмеження на загальну довжину мережі Gigabit Ethernet, але залишаються обмеження на довжини сементів, які з'єднують вузли мережі. Ці довжини визначаються специфікацією конкретного фізичного середовища. При побудові мережі на комутаторах протокол Gigabit Ethernet працює в повнодуплексному режимі.

Стандарти Gigabit Ethernet використовують чотири специфікації, які описують побудову мережі з використанням різного фізичного середовища.

Специфікація 1000Base-SX описує побудову мережі Gigabit Ethernet на багатомодових оптоволоконних кабелях діаметром 62,5/125 і 50/125 мікрон з максимальними довжинами сегментів відповідно 260 і 550 м. При цьому використовують короткохвильові лазерні передавачі, які генерують промені в діапазоні 850 нм.

Специфікація 1000Base-LX описує побудову мережі Gigabit Ethernet на оптоволоконних кабелях з використанням довгохвильових лазерних передавачів, які генерують промені в діапазоні 1300 нм. При цьому можуть використовуватися такі кабелі:

- багатомодове оптоволокно діаметром 62,5/125 і 50/125 мікрон з максимальними довжинами сегментів відповідно 440 і 550 м.
- одномодове оптоволокно діаметром 8,3/125 мікрон з максимальною довжиною сегментів до 5000 м.

Специфікація 1000Base-CX описує побудову мережі Gigabit Ethernet на двох і чотирьох коаксіальних кабелях у напівдуплексному і повнодуплексному режимах передавання даних. Для цієї специфікації почали випускати спеціальний кабель, який містить чотири коаксіальні пари і за зовнішнім виглядом, діаметром і гнучкістю нагадує кабель категорії 5. Максимальна довжина сегмента на цьому кабелі становить лише 25 м, і тому його використовують, як правило, для з'єднання вузлів мережі у межах однієї кімнати.

Специфікація 1000Base-T описана стандартом 802.3ab, розробленим спеціально створеним підкомітетом IEEE. Ця специфікація використовує для передавання даних чотири скручені пари UTP категорії 5 з максимальною довжиною сегмента до 100 м, якими одночасно передаються електричні сигнали. При цьому застосовують метод кодування PAM–5, який використовує п'ять рівнів електричного сигналу: –2, –1, 0, +1, +2. Код PAM–5 дозволяє передати за один такт 2,5 біти, що з врахуванням одночасного передавання сигналів за чотирма парами забезпечує швидкість 1000 Мбіт/с.

Технологія Gigabit Ethernet дозволяє використовувати в одному домені колізій тільки один концентратор. Використання комутаторів знімає обмеження на діаметр мережі загалом.

Стандарти 802.3z рекомендують використовувати технологію Gigabit Ethernet для побудови магістралей локальних мереж Ethernet та Fast Ethernet.

1.4.4 Особливості технологій TokenRing та FDDI

Технологія Token Ring

Технологія створена фірмою IBM в 1984 році, з використанням централізованого управління. Старша станція видає в мережу маркер, який переміщується по кільцю і надає права передачі кадру іншим станціям (рисунок 1.35).

Стандарт описано в рекомендації IEEE 802.4 – ***Token Bus***, 802.5 – ***Token Ring***. Реалізація технології має більшу вартість ніж Ethernet, так як додавання одного вузла в цій мережі в 2…4 рази дорожче, а ніж 1-го вузла в мережі Ethernet. Ця мережа використовує спеціальний концентратор **MSAU**, який має внутрішню розводку для Token Ring. Швидкість 4 ,16 і більше Мбіт/с. Сучасні мережі забезпечують швидкість до 100 Мбіт/с.

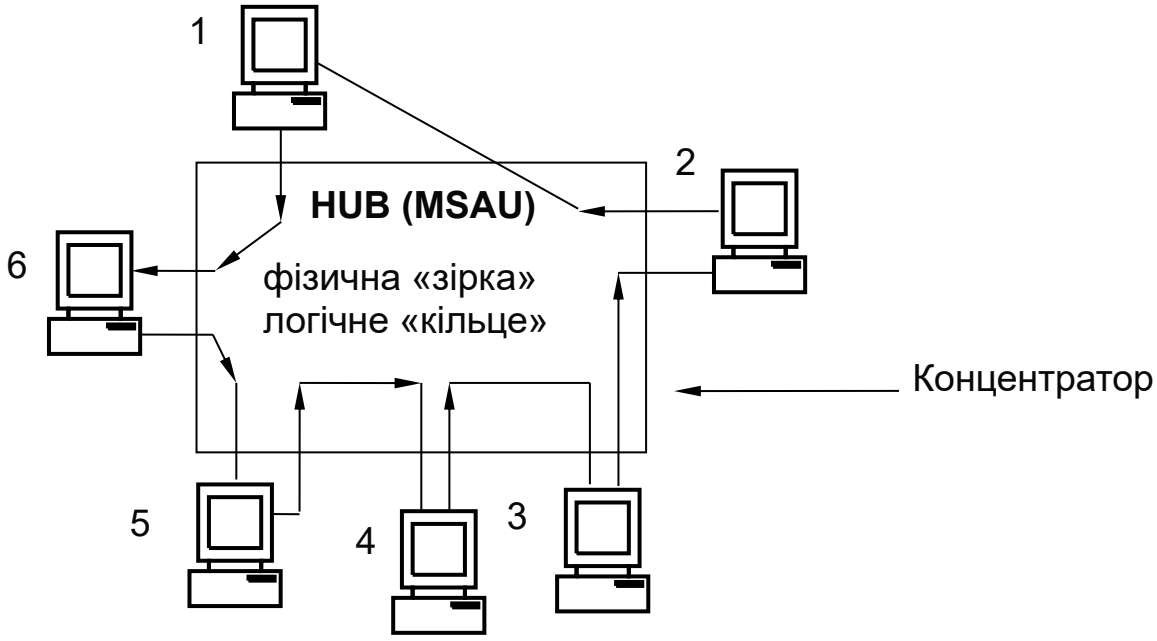


Рисунок 1.35 – Логічна топологія локальної мережі за технологією Token Ring

Старша станція – це станція із максимальним номером MAC-адреси. Вона кожні 3 секунди генерує спеціальний кадр (маркер) своєї появи в мережі. Якщо спеціальний сигнал відсутній >7 с, то здійснюється пошук нової старшої станції.

Доступ: спеціальне повідомлення («маркер») – постійно циркулює по логічному кільцю в одному напрямі. Коли маркер проходить через вузол, готовий до передачі даних в мережу, то цей вузол *захоплює маркер* і приєднує до нього дані (підготовлені до передачі) і передає їх в кільце. Повідомлення продовжує свою подорож по кільцю. Та станція, у якої номер (MAC-адрес) співпадає із MAC-адресою, вказаною в кадрі, отримує кадр і відправляє цей кадр в кільце вже з позначкою його прийому. Станція відправник побачить в кадрі позначку і вилучить кадр.

В локальних мережах за технологією Token Ring не існує умов для виникнення колізій і необхідності повторної передачі інформації, як наслідок – ефективність використання каналу буде більшою. Водночас час потрібний на очікування маркеру для початку передачі призводить до зростання затримки сигналів.

Основні технічні параметри:

- Швидкість передачі даних: стандартні значення становлять 4 Мбіт/с або 16 Мбіт/с. Пізніші модифікації підтримували до 100 Мбіт/с (High-Speed Token Ring).
- Метод доступу: маркерний (Token Passing). Станція може передавати дані лише тоді, коли утримує спеціальний службовий кадр – маркер.
- Топологія: логічне кільце, що фізично реалізується як «зірка» за допомогою концентраторів MAU (Multistation Access Unit).
- Тип середовища (кабелі):
  - Екранована вита пара (STP) типів 1, 2 або 6.
  - Неекранована вита пара (UTP).
  - Оптоволоконний кабель.
- Кількість вузлів у мережі: до 260 станцій при використанні STP та до 72 при використанні UTP.
- Адресація: використовуються 48-бітні MAC-адреси (індивідуальні, групові або функціональні).
- Максимальна довжина кадру: при швидкості 4 Мбіт/с — близько 4500 байт; при 16 Мбіт/с — до 18000 байт.

Технологія FDDI

Технологію FDDI було розроблено і стандартизовано інститутом ANSI у 1988 році з метою збільшення швидкості передавання даних та надійності LAN. Це перша технологія локальних мереж, в якій для передавання даних почали використовувати волоконно-оптичні кабелі. Fider Distributed Data Interface (FDDI) в перекладі означає – оптоволоконний інтерфейс розподілених даних.

Характерними особливостями технології FDDI є:

- Швидкість передавання даних – 100 Мбіт/с.
- Метод доступу до фізичного середовища – метод маркерного кільця.
- Топологія – подвійне кільце.
- Основне фізичне середовище – волоконно-оптичний кабель.
- Максимальна довжина мережі – 200 км (2 × 100 км).
- Максимальне число вузлів – 500.
- Відновлення роботи мережі шляхом її внутрішньої реконструкції за допомогою стандартних процедур.

Мережа FDDI будується на основі двох кілець з волоконно-оптичного кабелю, до яких під’єднуються робочі станції (рисунок 1.36). Одне з кілець є основним, а друге – резервним. У нормальному режимі роботи для передавання даних використовують основне (первинне) кільце. Резервне (вторинне) кільце мережа використовує у випадку обриву основного кабелю або при виході з ладу

однієї з робочих станцій. Використання двох кілець підвищує надійність роботи мережі FDDI та забезпечує автоматичне відновлення її роботи за допомогою стандартних процедур.

Особливості методу доступу до фізичного середовища

Метод маркерного кільця, який використовує технологія FDDI, передбачає циркулювання кільцем кадру певного формату, який називається *маркером.* Максимально допустимий час обертання маркера кільцем встановлюють під час ініціалізації мережі. Кожна станція мережі слідкує за реальним часом обертання маркера і порівнює його з максимально допустимим. Право на передавання своїх кадрів надається тільки тій станції, яка отримала маркер.

Стандарти технології FDDI дозволяють передавати як синхронні, так і асинхронні пакети. Формат і тип пакета визначається канальним рівнем. Розмір пакета технології FDDI становить 4500 байт.

Як правило, синхронні пакети є пакетами реального часу, які необхідно передавати в мережу невеликими порціями відразу ж після їх отримання. Тому станція, яка отримала маркер і має для передачі синхронні кадри, відразу ж передає їх в мережу протягом фіксованого часу.

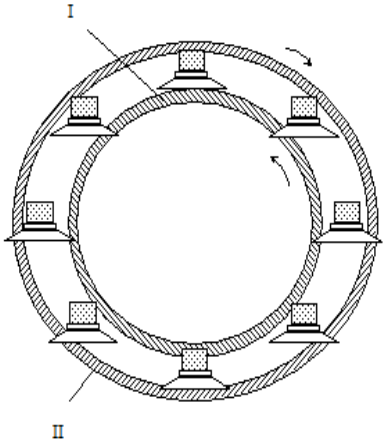


Рисунок 1.36 – Основна топологія мережі FFDI

Асинхронні пакети мають менший пріоритет, ніж синхронні. Вони не вимагають негайного передавання в мережу і тому їх можна передавати рідше і великими порціями. Право на передавання асинхронних кадрів станція отримує тільки тоді, коли реальний час обертання маркера, який вона отримала, менший за максимально допустимий. Асинхронні кадри станція може передавати протягом різниці цих інтервалів часу.

Якщо ж кільце перевантажене і реальний час обертання маркера більший за максимально допустимий, то станції при першому проходженні маркера пропускають його. Маркер швидко обертається кільцем і під час його повторного проходження станції отримують право на передавання своїх асинхронних кадрів. Такий механізм передавання технологією FDDI асинхронних кадрів є адаптивним і добре регулює завантаження мережі.

Отже, метод маркерного кільця регулює завантаження мережі, гальмуючи передавання асинхронних кадрів, забезпечує високий пріоритет синхронних кадрів і гарантує кожній станції доступ до фізичного середовища.

Висока надійність мережі підтримується за рахунок слідкування робочими станціями і концентраторами за часовими інтервалами циркуляції маркера та кадрів, а також наявністю фізичного з'єднання між портами. За виявлення відхилення від норми вони починають процес повторної ініціалізації мережі, а потім її реконфігурації.

Специфікації фізичного середовища технології FDDI

Технологія FDDI для побудови мережі використовує три стандарти: PMD, SMF-PMD і TP-PMD.

Стандарт PMD характерний для перших версій FDDI і передбачає використання багатомодового оптоволоконного середовища на основі кабелю діаметром 62.5/125 мікрон з максимальною віддаллю між вузлами до 2 км.

Стандарт SMF-PMD описав побудову мережі на базі одномодового волоконно-оптичного кабелю діаметром 8/125 мкм з лазерним передавачем сигналів. Це дозволило збільшити довжину сегмента до 40 км, а на особливо якісних кабелях – до 60 км.

Під час передавання даних волоконно-оптичними кабелями технологія FDDI використовує логічне кодування даних 4B/5B, відповідно до якого отримані від верхніх рівнів 4-бітні символи перекодовуються канальним рівнем у 5-бітні. Для забезпечення швидкості передавання даних 100 Мбіт/с фізичний рівень передає біти у лінію зв'язку з частотою 125 МГц.

У 1994 році ANSI випустив стандарт TP-PMD на основі неекранованих скручених пар 5-ї категорії з довжиною сегмента до 100 м. На ринку цей стандарт отримав назву CDDI – розподілений інтерфейс передавання даних кабельними лініями.

1.4.5 Середовища передачі даних для локальних мереж

А. Бездротові мережі передачі даних

Бездротові середовища передачі дозволяють передавати двійкові дані, кодуючи їх в електромагнітні сигнали мікрохвильового і радіодіапазону.

З усіх середовищ передачі бездротове забезпечує найбільшу мобільність, тому кількість пристроїв, що підтримують такі підключення, зростає з кожним днем. У міру збільшення пропускної спроможності бездротове підключення завойовує все більшу популярність в корпоративних мережах.

Використання бездротового середовища має певні особливості, які необхідно враховувати, а саме:

- **Зона покриття**. Бездротові технології передачі даних добре працюють на відкритих просторах. Проте деякі будівельні матеріали, що використовують при зведенні будівель і споруд, а також умови місцевості можуть обмежувати зону покриття.
- **Завади.** Якість бездротових з'єднань сприйнятлива до завад і може погіршуватися при роботі таких звичайних пристроїв, як бездротові телефони, деякі типи флуоресцентних ламп, мікрохвильові печі, а також під впливом інших бездротових комунікацій.

- **Безпека.** Для доступу до середовища бездротового підключення не вимагається підключатися до фізичних кабелів. Тому доступ до цього середовища можуть діставати несанкціоновані користувачі і пристрої. Отже, головним аспектом адміністрування бездротової мережі є безпека.
- **Загальне середовище передачі.** Мережі WLAN працюють в напівдуплексному режимі, що означає, що в кожен момент часу передачу або прийом може здійснювати тільки один пристрій. Середовище передачі загальне для усіх бездротових користувачів. Чим більше користувачів одночасно підключаються до WLAN, тим менша пропускна спроможність надається кожному з них.

Стандарти IEEE і галузеві стандарти бездротової передачі даних охоплюють як канальний, так і фізичний рівні (таблиця 1.7).

У кожному із стандартів, наведених у таблиці 1.7 специфікації фізичного рівня визначають наступні характеристики:

- Кодування даних за допомогою радіосигналів.
- Частота і потужність передачі.
- Вимоги до прийому і декодування сигналів.
- Проектування і зведення антен.

Таблиця 1.7 – Основні стандарти бездротових локальних мереж

Номер стандарту	**Комерційна назва технології**	**Опис**
1	2	3
IEEE 802.11	Wi-Fi	Набір стандартів для комунікації в бездротовій локальній мережі: **IEEE 802.11** – початковий стандарт, заснований на бездротовій передачі даних у діапазоні 2,4 ГГц. Підтримує обмін даними зі швидкістю до 1…2 Мбіт/с, передбачає два типи модуляції – DSSS і FHSS. **IEEE 802.11a** – стандарт, заснований на передачі даних в діапазоні 5 ГГц. Діапазон роздільний на три непересічні піддіапазони. Максимальна швидкість обміну даними становить 54 Мбіт/с, при цьому доступні також швидкості 48, 36, 24, 18, 12, 9 і 6 Мбіт/с. **IEEE 802.11b** – стандарт, заснований на передачі даних в діапазоні 2,4 ГГц. У всьому діапазоні існує три непересічні канали, тобто на одній території, не впливаючи один на одного, можуть працювати три різні бездротові мережі. Застосований метод модуляції DSSS. Максимальна швидкість роботи становить 11 Мбіт/с, при цьому доступні також

1	2	3
		швидкості 5,5, 2 та 1 Мбіт/с. Має покращену версію (b+), відрізняється від оригінального варіанту модуляцією PBCC (Packet Binary Convolutional Coding), подвоєною максимальною швидкістю (до 22 Мбіт/с). **IEEE 802.11g** – стандарт, заснований на передачі даних в діапазоні 2,4 ГГц. Діапазон аналогічний стандарту b. Для збільшення швидкості обміну даними застосований метод модуляції з ортогональним частотним мультиплексуванням (OFDM, *Ortogonal Frequency Division Multiplexing*), а також метод двійкового пакетного згорткового кодування PBCC (Packet Binary Convolutional Coding). **IEEE 802.11n (Wi-Fi 4)** – стандарт, що забезпечує передачу даних в діапазонах 2,4 ГГц і 5 ГГц. Стандарт 802.11n значно перевищує за швидкістю обміну даними попередні стандарти, внаслідок підтримки протоколу MIMO (multiple-input multiple-output). Теоретична швидкість може складати 150 Мбіт/с, зворотно сумісний з 802.11a, 802.11b і 802.11g. **IEEE 802.11ac (Wi-Fi 5)** – стандарт бездротових локальних мереж Wi-Fi на частотах 5…6 ГГц, передбачає використання до 8 антен MU-MIMO та розширення каналу до 80 або 160 МГц. Швидкість обміну даними може бути більшою за 1 Гбіт/с (до 6 Гбіт/с 8x MU-MIMO). **IEEE 802.11ax (Wi-Fi 6 (6E))** – новий стандарт бездротових локальних мереж, має за мету забезпечити пропускну спроможність близько 10 Гбіт/с. Робочі діапазони стандарту 5 ГГц і 2,4 ГГц (може включати додаткові смуги частот в діапазонах від 1 до 7 ГГц).
IEEE 802.15	Bluetooth	Інтерфейс Bluetooth дає змогу передавати як голос (зі швидкістю 64 кбіт/с), так і дані. Для передачі даних можуть бути використані асиметричний (721 кбіт/с в одному напрямку і 57,6 кбіт/с в іншому) та симетричний (432,6 кбіт/с в обох напрямках) методи. Частота 2,4 ГГц, дальність зв’язку у межах 10 або 100 метрів. З’єднання в межах 10 метрів дає змогу зберегти низьке енергоспоживання, компактний розмір і досить невисоку вартість компонентів.

1	2	3
		Малопотужний передавач споживає всього 0,3 мА в режимі standby і в середньому 30 мА під час обміну інформацією. Передбачене шифрування даних, що передаються з використанням ключа ефективної довжини від 8 до 128 біт і можливістю вибору односторонньої або двосторонньої аутентифікації. Працює за принципом FHSS (англ. Frequency-hopping spread spectrum), псевдовипадковий алгоритм стрибкоподібної перебудови передбачає зміну частоти 1600 разів за секунду, за шаблоном (pattern), складеному з 79 підчастот. Спільно працювати можуть тільки пристрої, які мають однаковий шаблон передачі. Основним структурним елементом мережі Bluetooth є так звана «пікомережа» (piconet) – сукупність від 2 до 8 пристроїв, що працюють на одному і тому ж шаблоні. У кожній пікомережі один пристрій працює як активний (master), а інші як пасивні (slave). Активний пристрій визначає шаблон, на якому працюватимуть усі пасивні пристрої його пікомережі та синхронізує її роботу. Частотний діапазон Bluetooth в більшості країн вільний від ліцензування.
IEEE 802.16	WiMAX	Стандарт бездротового зв’язку, що забезпечує широкосмуговий зв’язок на значні відстані зі високою швидкістю. У загальному вигляді мережі WiMAX складаються з базових і абонентських станцій, а також обладнання, що зв’язує базові станції між собою, з постачальником сервісів і з Інтернетом. Для з’єднання базової станції з абонентською використовується високочастотний діапазон радіохвиль від 1,5 до 11 ГГц. Швидкість обміну даними може досягати 70 Мбіт/с, при цьому не вимагається забезпечення прямої видимості між базовою станцією і приймачем. Між базовими станціями встановлюються з’єднання (прямої видимості), що використовують діапазон частот від 10 до 66 ГГц, швидкість обміну даними може досягати 120 Мбіт/с. Наразі поширені WiMAX-системи, засновані на версіях 802.16d і 802.16e цього стандарту, які між собою практично несумісні.

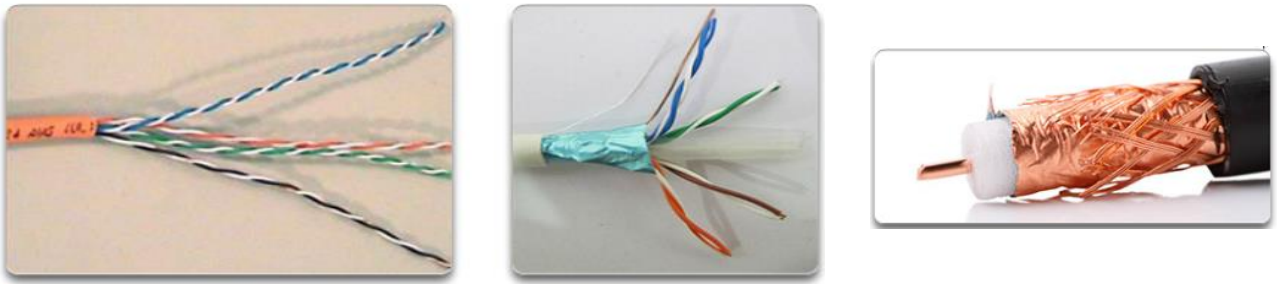
1	2	3
		802.16d (802.16-2004, фіксований WiMAX). Використовується ортогональне частотне мультиплексування (OFDM), підтримується фіксований доступ у зонах з наявністю або відсутністю прямої видимості. Користувацькі пристрої являють собою стаціонарні модеми для встановлення поза й всередині приміщень, а також PCMCIA-карти для ноутбуків. У більшості країн під цю технологію відведені діапазони 3,5 та 5 ГГц. *802.16e* (802.16-2005, мобільний WiMAX). Оптимізована для підтримки мобільних користувачів версія підтримує низку специфічних функцій, таких як хендовер, «idle mode» та роумінг. Застосовується масштабований OFDM-доступ (SOFDMA), можлива робота при наявності або відсутності прямої видимості. Частотні діапазони, що плануються для мереж Mobile WiMAX, такі: 2,3; 2,5; 3,4…3,8 ГГц.

Б. Кабелі передачі даних з металевими дротами

Середовища передачі з металевими дротами здійснюють передачу даних у вигляді електричних імпульсів, маніпульованих, як правило, за напругою.

Для побудови мереж використовується два основні типи мідних кабелів (рисунок 1.37).

- коаксіальний кабель;
- вита (кручена) пара неекранована (UTP) або екранована (STP).



а) UTP б) STP в) коаксіальний

Рисунок 1.37 – Основні типи кабелів з мідними дротами

Ці кабелі використовуються для з’єднання вузлів локальної мережі і підключення пристроїв мережевої інфраструктури, таких як комутатори, маршрутизатори і бездротові точки доступу. У стандартах фізичного рівня описані вимоги до кабелів для кожного з типів з’єднання і пристроїв, що відповідають їм.

Мідні кабелі, що використовуються в мережах, мають невисоку вартість, просто монтуються і досить ефективні за своїми електричними параметрами. Однак при передачі сигналів по мідних кабелях є обмеження по дальності і швидкості передачі.

Приймач в мережевому інтерфейсі цільового пристрою повинен отримати такий сигнал, який можна легко декодувати для відновлення відправленого сигналу. Проте чим більше дальність передачі сигналу, тим сильніше він спотворюється. Це відбувається загасання (attenuation), тобто зменшення величини сигналу при його розповсюдженні кабелем. Тому для усіх середовищ передачі даних на основі мідних кабелів в стандартах встановлені суворі обмеження на дальність передачі.

Часові характеристики і значення напруги електричних імпульсів також схильні до впливу наступних джерел перешкод.

- *Електромагнітні завади (ЕМЗ) або радіочастотні перешкоди (РЧП).* Сигнали ЕМЗ і РЧП можуть спотворювати і порушувати сигнали даних, що передаються по мідному кабелю. Потенційними джерелами ЕМЗ і РЧП є джерела радіочастотного випромінювання і електромагнітні пристрої, наприклад флуоресцентні лампи або електродвигуни.
- *Перехресні перешкоди (взаємні наведення).* Це перешкоди, викликані дією електричних або магнітних полів сигналу одного кабелю (пари) на сигнал сусіднього кабелю (пари). При передачі мовних сигналів без кодування перехресні перешкоди можуть призводити до часткової чутності сторонньої розмови у сусідньому каналі. Причина цього в тому, що при проходженні електричного струму по дроту навколо нього створюється слабке кругове магнітне поле, яке може впливати на сусідній дріт.

Найпростіший ***коаксіальний кабель*** складається з мідної жили (core), ізоляції, що її оточує, екрану у вигляді металевого обплетення та зовнішньої оболонки. Якщо кабель, крім металевого обплетення, має і шар фольги, він називається кабелем із подвійною екранізацією. *Екран* (shield) захищає дані, що передаються по кабелю, поглинаючи зовнішні електромагнітні сигнали, тобто шумові сигнали, які діють як завади. Отже, екран зменшує вплив завад щодо спотворення даних.

Мідна центральна жила може бути одним суцільним дротом або пучком дротів. Жила оточена *ізоляційним шаром*, який відокремлює її від металевого обплетення. Центральна жила та металеве обплетення не повинні стикатися, інакше відбудеться коротке замикання, перешкоди проникнуть у жилу, і дані руйнуються. Зовні кабель покритий непровідним шаром – з гуми, тефлону чи пластику для захисту від впливу навколишнього середовища. Загасання сигналу в коаксіальному кабелі менше, ніж у кручений парі.

Існує два типи коаксіальних кабелів: «тонкий» і «товстий».

Вибір того чи іншого типу кабелю залежить від потреб конкретної мережі.

Тонкий коаксіальний кабель – кабель діаметром близько 0,5 см (близько 0,25 дюймів). Він простий у застосуванні та придатний практично для будь-якого типу мережі. Підключається безпосередньо до плат мережного адаптера

комп’ютерів. Тонкий коаксіальний кабель здатний передавати сигнал на відстань до 185 м (близько 607 футів) без його помітного спотворення, викликаного згасанням. Тонкий коаксіальний кабель відноситься до групи, яка називається сімейством RG-58, його хвильовий опір ((impedance), опір змінному струму) дорівнює 50 Ом.

Товстий (thick) коаксіальний кабель відносно жорсткий кабель з діаметром близько 1 см (близько 0,5 дюймів). Іноді його називають «стандартний Ethernet», оскільки він був першим типом кабелю, який застосовується в Ethernet. Мідна жила цього кабелю товстіша, ніж у тонкого коаксіального кабелю. Чим товстіше центральна жила кабелю, тим більшу відстань здатний подолати сигнал. Тому товстий коаксіальний кабель передає сигнали до 500 м (близько 1640 футів), отже його використовують як основний кабель для прокладання магістралі (backbone), що з’єднує кілька невеликих мереж, побудованих на тонкому коаксіальному кабелі.

Товстий кабель важко гнути, тому складніше встановлювати його, він більш дорогий, передає сигнали на великі відстані. Як правило, що товстіший кабель, то складніше з ним працювати. Тонкий коаксіальний кабель гнучкий, простий в установці та відносно недорогий.

Для з’єднання коаксіальних кабелів використовують з’єднувач типу BNC (рисунок 1.38 а) – електричний роз’єм з байонетною фіксацією. У роз’ємах BNC різної конструкції центральна жила та обплетення коаксіального кабелю можуть фіксуватися трьома способами: пайкою, накруткою, обтиском деталей роз’єму на кабелі.

а. BNC-конектор

б. F-конектор

в. N-конектор

Рисунок 1.38 – Типи коаксіальних з’єднувачів

Також широке застосування мають з’єднувачі з простішою конструкцією та різьбовою фіксацією, а саме F-конектори (рисунок 1.38 б) та N-конектори (рисунок 1.38 в).

Найпоширенішим мідним кабелем для побудови локальних мереж є сьогодні є ***вита (кручена) пара*** – чотири пари мідних або мідно-алюмінієвих провідників діаметром 0,52 мм. Вони залишаються основним типом кабелів на відстані до 100 метрів – прості в монтажі та обслуговуванні, надійні та дуже економічні.

Наразі відрізняють 8 категорій крученої пари (рисунок 1.39):

Категорія 1 (Cat1). Одна кручена пара проводів Олександра Белла. Використовується лише в аналоговій телефонії.

Категорія 2 (Cat2). Двопарний кабель, розроблений для мереж Arcnet та TokenRing та забезпечує швидкість передачі до 4 Мбіт/с. Знято з виробництва на початку 2000-х років.

Категорія 3 (Cat3). Перший кабель на 4 пари. Створено для мереж Ethernet 10Base-T. Знято з виробництва у 2000-ні роки.

Категорія 4 (Cat4). Кабель на 4 пари для мереж Token Ring, 10/100Base-T. Знято з виробництва, але зустрічається на старих мережах.

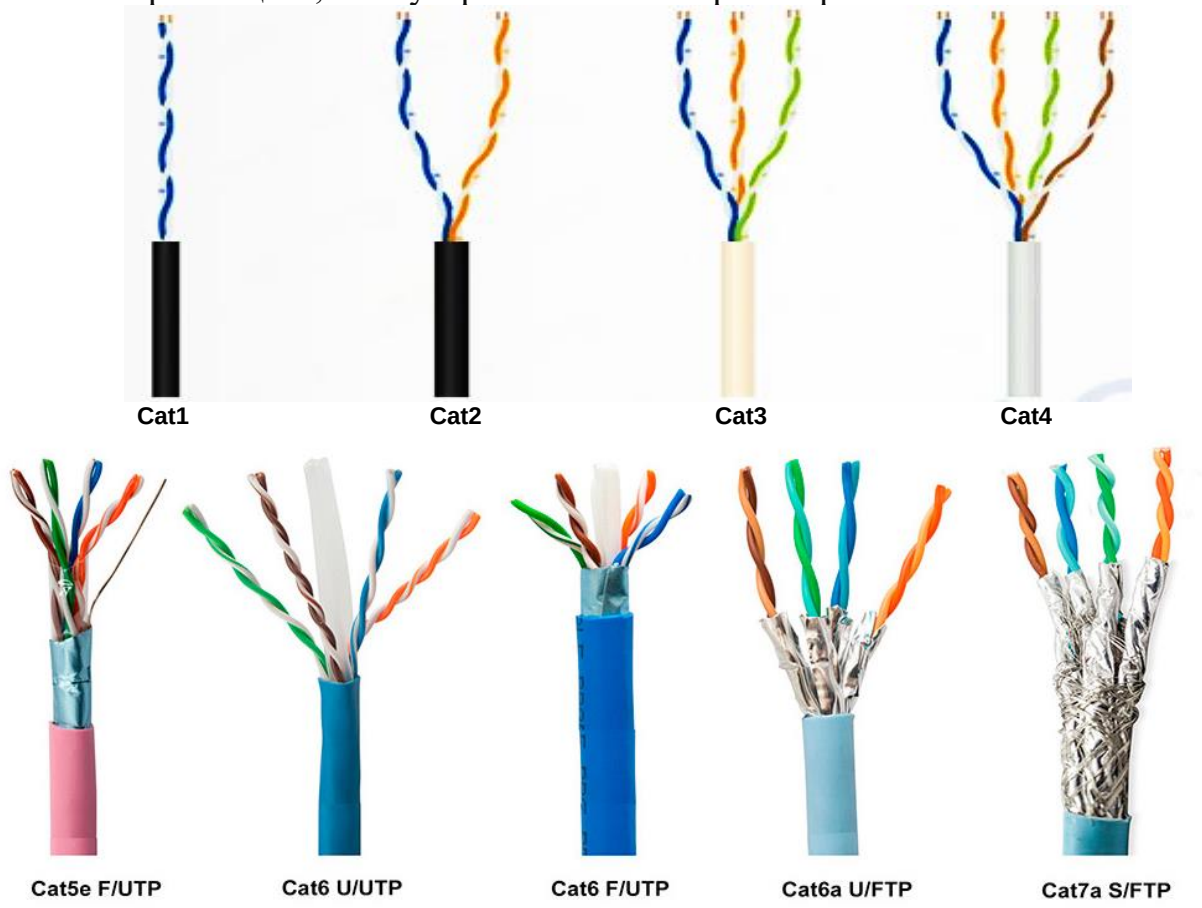


Рисунок 1.39 – Категорії кабелів типу вита пара

Категорія 5 (Cat5). Перший кабель здатний передавати інформацію на швидкості до 100 Мбіт/с. Майже повністю витіснений наступником – кабелем категорії 5e (Cat5e), який здатний передавати дані швидкості до 1 Гбіт/с.

Категорія 6 (Cat6). Представлена у 2002 році. Пропускна здатність крученої пари 10 Гбіт/с на відстань 25 м. Модифікація стандарту Cat6 – категорія 6A (Cat6a), представлена у 2008 році, зберігає пропускну здатність крученої пари 10 Гбіт/с на дистанціях до 100 метрів.

Категорія 7A (Cat7a). Глибока модернізація Cat7 призначена для роботи з 25 GE (GigabitEthernet). Пропускна здатність цього кабелю також дозволяє передавати сигнал 40GE, але лише на дистанцію до 15 метрів.

Категорія 8 (Cat8). Найновіший стандарт, представлений у 2016 році. Цей кабель на чотири пари може передати 40GE-сигнал, на відстань до 42 метрів.

Cat8 ділиться на 2 категорії: Cat8.1 – стандартизована для роботи з конекторами типу RJ-45 та назад сумісний з кабелями Cat6A. Cat8.2 – призначена для конекторів типу TERA (розробка Siemens Company), GG45 (розробка Nexans) та ARJ-45 (розробка Bei Fuse Ltd). Дані конектори є пропрієтарними і перспективи їх застосування не визначені.

Крім категорій, мідні кабелі розрізняють по конструкції. Виділяють такі різновиди кабелю:

- UTP – кабель у простій оболонці, без броні або захисного екрану (неекранована кручена пара). Зазвичай прокладається усередині приміщень.
- FTP – екранована кручена пара (екран із фольги).
- STP – тут у захисний екран поміщена кожна пара проводів і між двома оболонками прокладено броню із дротяної сітки.
- S/FTP він же SSTP – кабель із подвійним екрануванням. Перший обплітає кожну пару окремо, другий охоплює весь пучок.
- U/STP - аналог STP, але без зовнішньої броні.
- SFTP – ця екранована кручена пара має найбільш товстий кабель з усіх. Має три екрани: внутрішній, що охоплює парні жили та два зовнішні. Один із фольги, інший із дротяної сітки.

Головними перевагами мідного кабелю в порівнянні з оптичним – це дешевизна та легкість розгортання. Все, що потрібно для армування – інструмент для обтиску крученої пари (кримпер або кліщі для обтиску крученої пари) і конектор типу 8P8C (RJ45). При цьому монтажнику не потрібне спеціальне спорядження та підготовка. Прокладання мідного кабелю всередині приміщення не вимагає застосування захисних чохлів або гофротруби як оптичних патчкордів. А якщо роз’єм забрудниться, його можна легко очистити або встановити новий.

В. Волоконно-оптичні кабелі передачі даних

Конструкція

Волоконно-оптичний кабель (також відомий як оптоволоконний) призначений передачі сигналів зв’язку за допомогою світлового потоку. Основною його відмінністю є використання світлових імпульсів, що генеруються в оптичному модулі та надходять до приймача на іншому кінці волокна. Завдяки своїй структурі волокно забезпечує передавання світлових імпульсів з мінімальним загасанням, тож оптоволоконні лінії здатні забезпечити передавання сигналів на значні відстані з високою швидкістю.

Це відносно надійний (захищений) спосіб передачі, оскільки за відсутності електричних сигналів назовні з волокна нічого не випромінюється, а фізичне підключення до волокна без втрати сигналу на прийомі не можливе. Отже, практично оптоволоконний кабель не можна розкрити та перехопити дані.

Оптичне волокно – дуже тонкий скляний циліндр, що має назву серцевина (core), покритий шаром скла, який називають оболонкою, з іншим, більшим ніж у серцевини, коефіцієнтом заломлення. Існують оптоволокна із пластику, вони

простіше у використанні, але передають світлові імпульси на менші відстані, порівняно зі скляним оптоволокном. Скляне оптоволокно – це гнучка, дуже тонка і прозора нитка з хімічно чистого скла завтовшки дещо більше людського волосся. Кожне скляне оптоволокно, як правило, передає сигнали лише одному напрямку, тому для дуплексної роботи потрібний кабель з двох волокон з окремими конекторами. Одне служить передачі, інше – прийому.

Для передачі по оптоволоконному кабелю біти кодуються за допомогою світлових імпульсів. Оптоволоконний кабель діє як світлопровід, або «оптичний хвилевід», що забезпечує передачу світлового сигналу між двома кінцями кабелю з мінімальними втратами.

Оптичне волокно складається з (рисунок 1.40):

• серцевини,
• оптичної оболонки,
• захисного покриття,
• буферного покриття.

Промені світла поширюються в серцевині, не виходячи за нього межі, оскільки відбиваються від покриваючого шару оболонки, якій має відмінний показник заломлення. Для реалізації цього ефекту формуються два шари з кварцового скла з різними показниками заломлення.

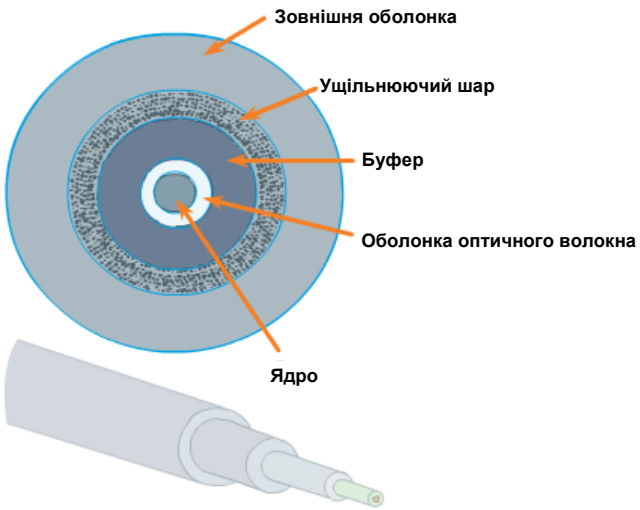


Рисунок 1.40 – Конструкція оптичного волокна

Одним з проблемних питань використання оптичних волокон є реалізація їх з’єднання. Для цього потрібні або спеціальні пристрої – оптоволоконні роз’єми або спеціальне обладнання – зварювальний пристрій (рисунок 1.41).

Рисунок 1.41 – Обладнання для з’єднання оптичних волокон

Жорсткість волокон збільшена покриттям із пластику, а міцність – волокнами із кевлару. Кевларові волокна розташовуються між волокнами, укладеними у пластик.

Конструкція волокно-оптичного кабелю представлена на рисунку 1.42.

Рисунок 1.42 – Конструкція оптоволоконного кабелю

1. *Несучий сердечник (1)* – встановлюється для натягу оптоволоконного кабелю, як правило, виконується з металу або склопластику і сприймає на себе всю вагу волоконно-оптичної лінії при підвішуванні, оскільки власне оптичне волокно не має достатньої міцності на розрив. Також він центрує всю конструкцію довкола себе. Може виготовлятися як із захисною оболонкою, так і без неї.
2. *Оптичне волокно (2)* – основний елемент сердечника, призначений безпосередньо передачі світлового сигналу. В одному кабелі зазвичай міститься від 2 до 250 волокон. Кожне з них покривається спеціальним лаком, який забезпечує достатню міцність волокну та додатково запобігає поширенню світла за його межі.
3. *Трубчасті модулі (3)* – призначені для захисту волокон від механічних пошкоджень та маркування їх окремих груп. У складі кабелю може бути один або кілька таких модулів (рисунок 1.43), всередині вони заповнюються спеціальним захисним шаром з гідрофобного наповнювача. Чим більше волокон входить до складу волоконно-оптичного кабелю, тим більш актуальним є використання декількох модулів.
4. *Плівка з гідрофобним наповнювачем (4)* – виступає в ролі однієї або кількох захисних оболонок для всього пучка волоконно-оптичного кабелю, їх кількість

визначається конструкцією всього сердечника та умовами його експлуатації. Призначена для зниження тертя всередині та запобігання проникненню вологи усередину. Як правило, усередині волоконно-оптичного кабелю вона додатково стягується нитками.

5. *Шар діелектричного матеріалу (5)* – з поліетилену, але в інших моделях може застосовуватися і ізоляція полівінілхлориду. Також виконує функцію захисту волоконно-оптичних ліній від вологи.

6. *Шар броні (6)* – забезпечує достатню механічну міцність при будь-якій прокладці. Його основне завдання: запобігти порушенню цілісності волокон різальними предметами, у процесі перетирання або гризунами. Може виконуватись металевим дротом, скловолокном або кевларом. Броньований кабель може використовуватися для зовнішньої прокладки, у шахтах та колодязях та під землею.

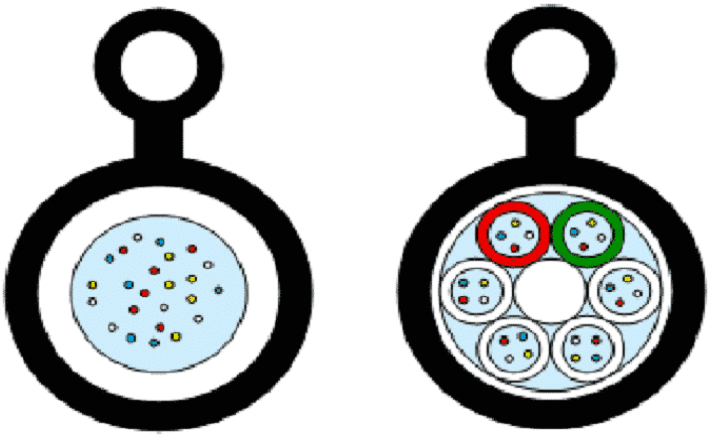


Рисунок 1.43 – Приклад одномодульного та багатомодульного кабелю

7. *Зовнішня оболонка (7)* – основний елемент волоконно-оптичного кабелю, що запобігає руйнівному впливу зовнішніх факторів на лінію. Зовнішній шар виконується із поліетилену або іншого герметичного діелектрика. Дозволяє використовувати оптоволоконну продукцію як для повітряної прокладки, так і для кабельних каналів. Додатковою функцією діелектричної оболонки є захист впливу електричної напруги в аварійних ситуаціях.

Класифікація

Волоконно-оптичний кабель, залежно від обраного критерію, буде поділятися на різні категорії. Так, *за матеріалом виготовлення* виділяють два типи оптоволоконних виробів:

GOF-волокно – оптичне волокно, виготовлене зі скла (перша буква абревіатури походить від англійського glass);

POF-волокно – моделі з полімерним провідним елементом (перша буква абревіатури походить від англійського plastic).

Залежно *від величини серцевини* у волоконно-оптичному каналі по відношенню до демпфуючого шару виділяють ***одномодові*** та ***багатомодові*** волокна (рисунок 1.43). Вони відрізняються за кількістю типів хвиль (мод), що проводяться, від чого і походить їх назва.

Одномодове волокно (рисунок 1.44) характеризується відносно невеликим діаметром провідника сердечника – 9 мкм. Такий розмір пропускає лише один тип хвилі каналом. В одномодовій конструкції сигнал не спотворюється внаслідок міжмодової дисперсії і може передаватися на більші відстані.

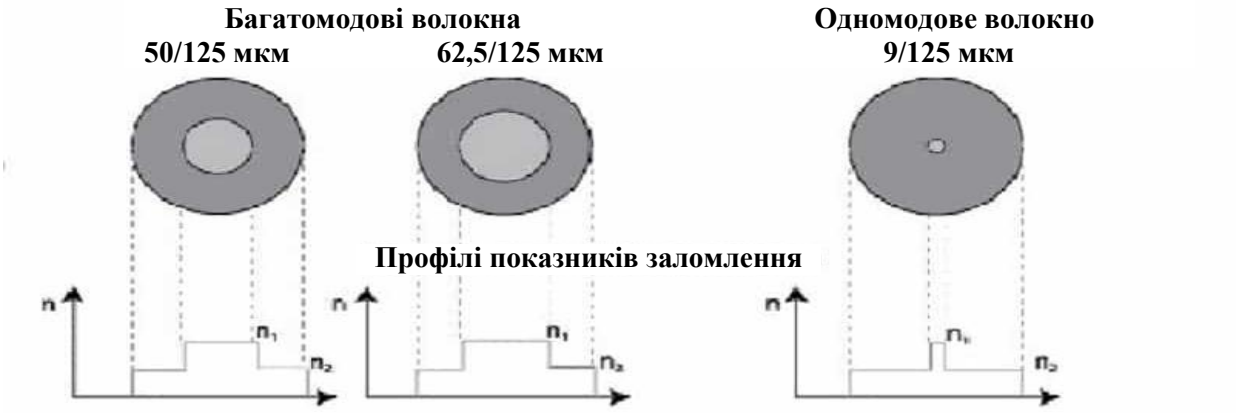


Рисунок 1.43. Одномодове та багатомодові волокна у перерізі

Рисунок 1.44. Просування моди в одномодовому волокні

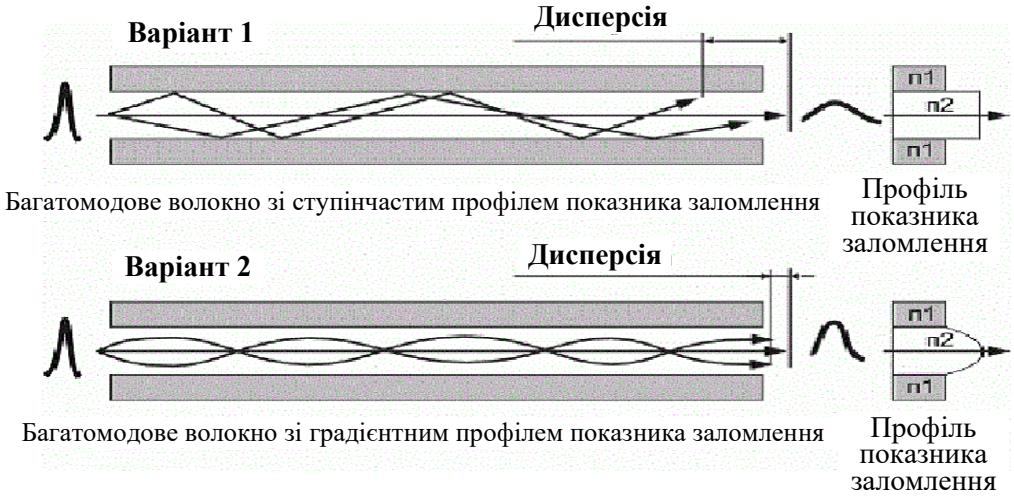


Рисунок 1.45 – Просування мод в багатомодовому волокні

Недоліком багатомодового волокна є спотворення сигналу внаслідок міжмодової дисперсії, але разом з тим багатомодове волокно більш дешеве у виробництві, порівняно з одномодовим, а також висуває менші вимоги до іншого обладнання, як то оптичні випромінювачі (працює на звичайних світлодіодах, а не на лазері), оптичні з'єднувачі, тощо.

Технічні характеристики

При монтажі та під час експлуатації важливо враховувати основні параметри кабельно-провідникової продукції, які визначають номінальні умови для нормальної роботи системи. Волоконно-оптичні кабелі мають такі характеристики:

– Мінімальний радіус вигину становить не менше 20 діаметрів виробу.

– Межі робочих температур – від – 60 до + 70°С. Слід зазначити, що монтажні роботи волоконно-оптичних ліній можуть виконуватись при температурі навколишнього середовища не менше – 10°С без втрати характеристик основних елементів волоконно-оптичного кабелю.

– Електричний опір становить щонайменше 2000 МОм. Ізоляція витримує вплив підвищеної напруги в 20 кВ змінного та 10 кВ постійного струму.

– Ізоляція здатна короткочасно витримати струм розтікання до 105 кА, та його тривалість допускається трохи більше 60 мс.

– Максимальне зусилля на розтягування, залежно від конкретної моделі, становить від 4 до 42 кН.

Переваги та недоліки

Волоконно-оптичні лінії, в порівнянні з лініями на мідних кабеляях, мають ряд вагомих **переваг**:

– Відсутність впливу електромагнітного випромінювання від сусідніх джерел, завдяки чому в ньому не виникають перешкоди.

– Завдяки відсутності електричної напруги в сполучних кабелях здійснюється гальванічна розв’язка між джерелом і приймачем.

– Висока швидкість та велика пропускна здатність, у порівнянні з мідними варіантами.

– Велика відстань передачі сигналу, внаслідок невеликого коефіцієнту загасання сигналу.

– Краща безпека даних через неможливість безконтактного знімання даних та складності несанкціонованого підключення до лінії.

– Відсутня зацікавленість вандалів до розкрадання волоконно-оптичної продукції.

Поряд з перевагами оптоволокно має і деякі вади. До **недоліків** волоконно-оптичних ліній відносять:

– Крихкість самого матеріалу волокна, через що їх легко поламати.
– Для передачі і перетворення сигналів, що передаються у волоконно-оптичних каналах в оптичному діапазоні, необхідно спеціальне обладнання.
– У разі пошкодження оптоволоконного кабелю для з’єднання місця розриву необхідне спеціальне обладнання, приладдя та кваліфікований персонал.
– Більша вартість як самого кабелю (внаслідок маркетингового попиту), так і додаткових пристроїв.
– Монтажні роботи можуть виконувати лише кваліфікований, спеціально навчений персонал.

1.5 Адресування в інформаційно-комунікаційних мережах

1.5.1 Мережеві адреси (IPv4)

IP-адреса або **адреса третього рівня** – це логічна адреса, яка не прив'язується до конкретної апаратури (мережній карті, інтерфейсу і т.д.) і призначається адміністратором мережі. Протокол IP версії 4 (**IPv4**) – використовує 32-бітні адреси (при цьому можна адресувати $2^{32} \approx$ 4,3 млрд. пристроїв)

Класова адресація IPv4

Хронологічно першим методом поділу IP-адрес є так звана класова модель IP-адресації, яка частково розв'язала проблему нераціонального використання адресного простору. Згідно із цією моделлю, увесь простір IP-адрес ділиться на 5 класів залежно від значення перших чотирьох біт IPv4-адреси. Класам привласнені імена від A до E (таблиця 1.7).

Перші 3 класи A, B и C використовуються для індивідуальної (unicast) адресації мереж і вузлів, клас D – для багатоадресного або групового (multicast) розсилання, клас E зарезервований для експериментів. Класи A, B и C мають різну довжину мережної частини адреси.

Для мереж класу A (рисунок 1.46) під ідентифікатор мережі приділяється 1 байт (перший октет), 3 байти (3 октети), що залишилися використовуються для ідентифікатора вузла, причому старший (лівий) біт ідентифікатора мережі завжди рівний 0.

Таблиця 1.7 – Класи IPv4-адрес

Клас	**Перші біти**	**Найменший номер мережі**	**Найбільший номер мережі**	**Максимальне число вузлів у мережі**
A	0	1.0.0.0	126.0.0.0	2^{24} (поле 3 байти)
B	10	128.0. 0.0	191. 255.0.0	2^{16} (поле 2 байти)
C	110	192.0. 0.0	223. 255.255.0	2^{8} (поле 1 байт)
D	1110	224.0. 0.0	239. 255.255.255	Групові адреси
E	11110	240.0. 0.0	247. 255.255.255	Зарезервований

Оскільки перший біт ідентифікатора мережі завжди дорівнює нулю, те 7 біт, що залишилися дозволяють адресувати 128 (2^7) різних мереж. Однак через те, що адреси 0.0.0.0 і 127.0.0.0 є спеціальними IPv4-адресами, кількість доступних мереж класу A дорівнює 126. У кожній мережі класу A можна адресувати до 16 777 214 (2^{24}-2) вузлів. Дві адреси віднімаються внаслідок того, що вони використовуються в спеціальних цілях і не можуть бути призначені пристрою (перша адреса – адреса мережі, остання адреса – широкомовна адреса).

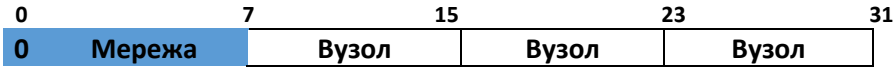


Рисунок 1.46 – Формат IPv4-адреси класу A

Мережі класу B (рисунок 1.47) визначаються значеннями 10 у двох старших бітах адреси. Перші 2 байти в адресі використовуються для ідентифікатора мережі, 2 байти, що залишилися, – для ідентифікатора вузла. У результаті кількість доступних мереж класу B становить 16 384 (2^{14}) з кількістю вузлів у кожній мережі рівним 65 534 (2^{16}-2).

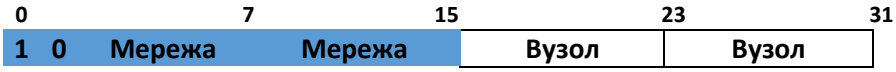


Рисунок 1.47 – Формат IPv4-адреси класу B

Для мереж класу C (рисунок 1.48) під ідентифікатор мережі приділяється 3 байти, а під ідентифікатор вузла тільки 1 байт. Три старші біти першого октету завжди мають значення 110, дозволяючи визначити, що адреса відноситься саме до класу C. Таким чином, можна сформувати 2 097 152 (2^{21}) мереж, у кожній з яких можуть перебувати 254 (2^{8}-2) вузли.

Рисунок 1.48 – Формат IPv4-адреси класу C

Мережі класу D (рисунок 1.49) визначаються значеннями 1110 у перших чотирьох бітах адреси, інші біти використовуються для адресації багатоадресної групи. Адресний простір класу D зарезервований для групового розсилання й використовується для адресації групи вузлів. Ідентифікаторів мереж і вузлів в IPv4-адресі класу D не виділяють.

Рисунок 1.49 – Формат IPv4-адреси класу D

Мережі класу E (рисунок 1.50) є експериментальними й у цей час не використовуються. Адреси в цьому класі визначаються значеннями 1111 у перших чотирьох бітах.

Рисунок 1.50 – Формат IPv4-адреси класу E

Часткові й публічні адреси IPv4

У мережі Інтернет ідентифікація пристроїв здійснюється унікальними IPv4-адресами, які не повинні повторюватися в глобальній мережі. Такі IPv4-адреси називаються *публічними* адресами. Однак число публічних адрес обмежене, тому в кожному із класів IP-мереж визначений так званий частковий простір IP-адрес. Часткові IPv4-адреси призначені для використання в локальних комп'ютерних мережах і не маршрутизуються в Інтернет. Для локальних мереж, не підключених до мережі Інтернет, можна використовувати будь-які можливі адреси, унікальні в межах даної мережі.

Публічні адреси перебувають у межах від 1.0.0.1 до 223.255.255.254 за винятком часткових адрес IPv4. Адресний простір часткових IPv4-адрес складається з 3 блоків:

- 10.0.0.0 … 10.255.255.255 (в класі A);
- 172.16.0.0 … 172.31.255.255 (в класі B);
- 192.168.0.0 … 192.168.255.255 (в класі C).

Крім цього визначені IPv4-адреси (таблиця 1.8), які мають спеціальне призначення (спеціальні адреси).

Таблиця 1.8 – Спеціальні IPv4-адреси

Ідентифікатор мережі (ІМ)	**Ідентифікатор вузла (ІВ)**	**Опис**
Усі «0»	Усі «0»	0.0.0.0 – не відома IPv4-адреса. Використовується, наприклад, коли пристрій намагається одержати IPv4-адресу за допомогою протоколу DHCP
Усі «0»	ІВ	Вузол призначення належить тій же мережі, що й вузол-відправник, наприклад, 0.0.0.25
ІМ	Усі «0»	Адреса мережі, наприклад, 175.11.0.0
ІМ	Усі «1»	Обмежена широкомовна адреса (у межах даної IP-мережі), наприклад, 192.168.100.255
Усі «1»	Усі «1»	255.255.255.255 – «глобальна» широкомовна адреса
169.254.x.y	ІВ	Адреси, що автоматично призначаються, за відсутності відповіді від сервера DHCP
127.0.0.1		Адреса інтерфейсу зворотної петлі (loopback), призначений для тестування встаткування без реального відправлення пакета

Застосування IP-адресування на основі класового розподілу обмежує можливості раціонального використання всього переліку (пула) адрес в кожній мережі, тому в подальшому стали використовувати два додаткових варіанти IPv4: на основі підмереж та основі безкласового адресування.

1.5.2 Мережеві адреси (IPv6)

Протокол IPv6 – це нова версія протоколу IP, яка розроблена в якості спадкоємця IPv4 і покликана розв'язати проблему вичерпання адресного простору. На відміну від адреси IPv4, яка має довжину 32 біта, розмір адреси IPv6 становить 128 біт, що дозволяє адресувати приблизно $3{,}4 \times 10^{38}$ інтерфейсів пристроїв. Адреса IPv6 відображається як вісім груп по чотири шістнадцяткові цифри, розділені двокрапкою. Наприклад (рисунок 1.51), 2001:0DB8:AC10:FE01:0018:8BFF:FED8:E3E0.

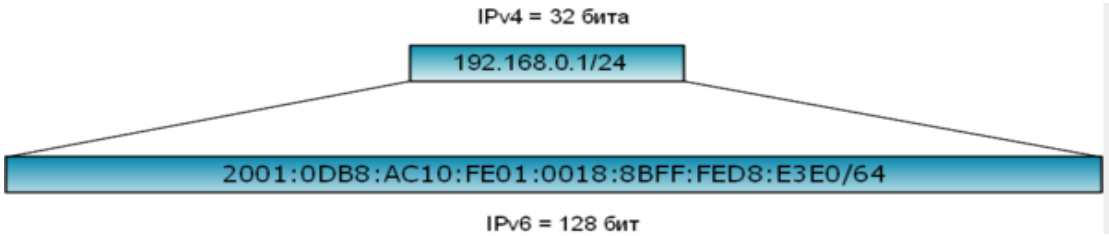


Рисунок 1.51 – Порівняння адрес IPv4 і IPv6

Існує два способи, які дозволяють скоротити запис IPv6-адреси:

- нулі на початку групи можна прибрати (не писати);
- одна або кілька послідовних груп, що складаються з нулів, можуть бути замінені позначкою «**::**», але тільки один раз на адресу.

Для наведеної нижче адреси цифри, виділені жирним шрифтом, представляють позиції, у яких адреса може бути скорочена:

2001:1000:**0000**:**0000**:**0000**:ABCD:**0000**:**0001**

Варіанти можливих скорочень:

- 2001:1000::ABCD:0:0001;
- 2001:1000::ABCD:0:1.

IPv6-адреса складається із двох логічних частин – *префікса* (Prefix) і *ідентифікатора інтерфейсу* (Interface ID) (рисунок 1.52).

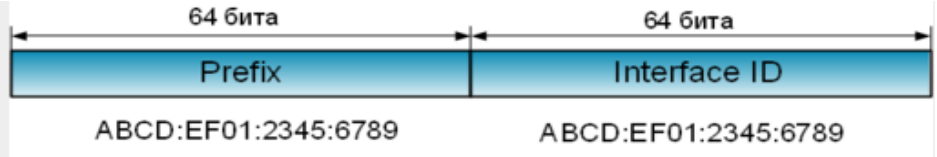


Рисунок 1.52 – Структура IPv6-адреси

***Префікс* (Prefix)** – перші 64 біти адреси, це частина адреси, яка відведена під ідентифікатор мережі/підмережі (аналог ідентифікатора мережі в IPv4). Префікс для адреси IPv6 визначається у вигляді *адреса IPv6/довжина префікса*. Якщо довжина префіксу має значення 64, значить розбивка на підмережі не передбачена, якщо менше наприклад 48, то 48 біт відведені під номер мережі, а інша частина префіксу (64-48=16 біт) для номера підмережі.

Розглянемо для прикладу IPv6-адресу: 2001:0f68:0000:0000:0000:0000:1986:69af/48. Оскільки префікс (/48) указує на перші 48 біт, то 2001:0f68:0000 є номером мережі (Global Routing Prefix), а наступне поле, 0000, указує на ідентифікатор під мережі (Subnet ID).

Ідентифікатор інтерфейсу **(Interface ID)** – останні 64 біта IPv6-адреси використовуються для ідентифікації інтерфейсу в сегменті мережі (аналог ідентифікатора вузла в IPv4). Він повинен бути унікальним усередині мережі/підмережі.

Адресний простір протоколу IPv6 розділений на три типи адрес:

- індивідуальні (unicast) адреси
- багатоадресні (multicast) адреси;
- альтернативні (anycast) адреси.

Індивідуальні адреси ідентифікують один інтерфейс пристрою. Пакети, відправлені на цю адресу, доставляються тільки на цей інтерфейс.

Типи Unicast адрес:

– *Глобальні*

Відповідають публічним IPv4 адресам. Можуть перебувати в будь-якому не зайнятому діапазоні. У цей час регіональні Інтернет-регістратори розподіляють блок адрес 2000::/3 (з 2000:: по 3FFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF).

– *Link-Local*

Адреси мережі, які призначені тільки для комунікацій в межах одного сегмента місцевої мережі або магістральної лінії. Вони дозволяють звертатися до хостів, не використовуючи загальний префікс адреси. Маршрутизатори не відправлятимуть пакети з адресами link-local.

Адреси link-local часто використовуються для автоматичного конфігурування мережної адреси, у випадках, коли зовнішні джерела інформації про адресу мережі недоступні.

– *Unique-Local*

RFC 4193, відповідають частковим IP-адресам, якими у версії IPv4 були 10.0.0.0/8, 172.16.0.0/12 і 192.168.0.0/16. Починаються із цифр FC00 і FD00.

Групові адреси IPv6, подібно однойменним адресам IPv4, визначають групу інтерфейсів. Пакети, що посилають на цю адресу, доставляються всім інтерфейсам – учасникам групи розсилання.

Альтернативні адреси дозволяють адресувати групу інтерфейсів (звичайно приналежних різним вузлам). Однак на відміну від багатоадресних адрес, пакети, передані на альтернативну адресу, доставляються на один з інтерфейсів (звичайно «найближчий», згідно з метрикою маршрутизації), обумовлених цією адресою.

Широкомовні адреси (Broadcast), які використовуються в IPv4, в IPv6 відсутні, що сприяє зменшенню мережного трафіка й зниженню навантаження на більшість систем. Широкомовні адреси замінені багатоадресними.

Серед IPv6-адрес також передбачені зарезервовані адреси (таблиця 1.9), які мають спеціальне призначення (спеціальні адреси).

Таблиця 1.9 – Спеціальні адреси IPv6

IPv6 адреса	Довжина префіксу (бітів)	Опис	Примітки
::	128	Невизначена адреса	див. 0.0.0.0 в IPv4
::1	128	loopback адреса	див. 127.0.0.1 в IPv4
::ffff: xx.xx.xx.xx	96	Адреса IPv4, що відображена на IPv6	Нижні 32 біти – це адреса IPv4. Для хостів, що не підтримують IPv6
2001:db8::	32	Документування	Зарезервовано для прикладів в документації в rfc3849
fe80:: – febf::	10	link-local	Аналог 169.254.0.0/16 в IPv4
fec0:: — feff::	10	site-local	Відмічений як застарілий в rfc3879
fc00::	7	Unique Local Unicast	Аналог 169.254.0.0/16 в IPv4
ffxx::	8	multicast	Мультімовлення для всіх маршрутизаторів ff02 :: 2 Мультімовлення для всіх вузлів ff02 :: 1

1.5.3 Символьні адреси

Символьні адреси, які в *Internet* називають доменними іменами застосовують для зручності роботи користувача у клієнтських програмах перегляду web-сторінок. Символьні адреси зазвичай несуть якесь смислове навантаження, їх набагато легше запам’ятовувати. Такі адреси формуються за ієрархічною ознакою: складові повної символьної адреси в мережі розділяються крапкою і перераховуються в наступному порядку:

– спочатку ім’я кінцевого вузла, наприклад ***home***;
– потім ім’я групи вузлів, наприклад ***managers***;
– потім ім’я більшої групи, наприклад ***company***;
– і так до самого вищого рівня, наприклад ***ua***.

Наявність символьної адреси не змінює алгоритм передачі інформації і взаємодії вузлів – реальна взаємодія вузлів все одно відбувається за IP-адресами. Тобто кожній символьній адресі співставляється певна IP-адреса. Проте між символьною адресою (доменним іменем) і IP-адресою вузла нема ніякої алгоритмічної (логічної) відповідності, тому необхідні додаткові таблиці або

служби, щоб вузол однозначно визначався, як за доменним іменем, так і за IP-адресою.

В мережах на основі стеку TCP/IP використовується спеціальна служба ***Domain Name System*** (**DNS**), яка установлює цю відповідність на основі створюємих адміністраторами мережі *таблиць відповідності*. Тому доменні імена також називають DNS-імена.

Система DNS є ієрархічної й розподіленою. Не існує єдиної бази даних, що зберігає інформацію про всі імена та відповідні їм IP-адреси і інші записи. DNS – це мільйони баз даних, кожна з яких містить інформацію про конкретний домен. Ієрархію DNS можна побачити в доменному імені, наприклад, в імені веб-сайту. Візьмемо, наприклад, сайт – **www.cip.gov.ua**. Це ім'я складається із трьох частин, розділених крапками. Точніше чотирьох, оскільки, формально говорячи, повне доменне ім'я завжди закінчується крапкою, що позначає так званий кореневий домен, або кореневу зону DNS. Отже:

Коренева зона	Містить інформацію про всі домени найвищого (верхнього) рівня, які сформовані за організаційним або географічним ознаками: net, com, org, gb, li, ua, su і т.д. Точніше, інформацію про сервери, що обслуговують ці домени
ua	Домен **ua**, що містить інформацію про всі піддомени, зареєстровані у ньому, наприклад, **gov**. Знову ж, цей домен містить адреси серверів, у яких можна одержати додаткову інформацію про вміст піддоменів
gov	Піддомен **gov**, що містить інформацію про всі піддомени наступного, зареєстровані у ньому, зокрема **cip**. Знову ж, цей домен містить адреси серверів, у яких можна одержати додаткову інформацію про вміст піддоменів.
cip	Піддомен **cip**, що містить інформацію про всі служби або файли, а також імена серверів, зареєстрованих безпосередньо в цьому домені, зокрема **www.cip.gov.ua**
www	Послуга веб-сервера: для неї у піддомені **cip.gov.ua** зберігається відповідна йому IP-адреса

Таким чином, трансляція імені www.cip.gov.ua у відповідну йому IP-адресу буде відбуватися в кілька етапів (рисунок 1.53). Спочатку будуть запитані сервери, що обслуговують кореневу зону. Ці сервери нічого не знають про існування доменів gov чи cip і тим більш адреси www.cip.gov.ua. Але вони повідомлять, як можна зв'язатися із серверами, що обслуговують домен наступного рівня ua (операції 2 та 3). Від них можна довідатися адреси серверів домену gov (операції 4 та 5). Від наступних – домену cip, які, у свою чергу, дадуть відповідь на запит про IP-адресу сервера www.cip.gov.ua.

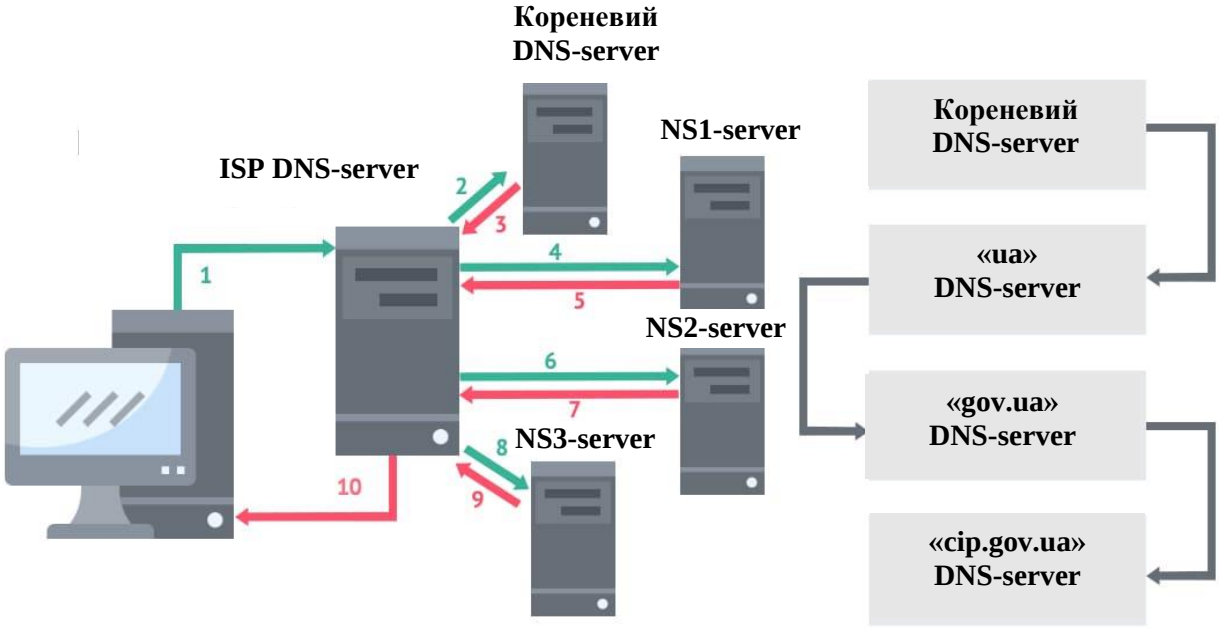


Рисунок 1.53 – Процес трансляції імен в DNS

Процедуру послідовної взаємодії з DNS-серверами різних рівнів ієрархії реалізує DNS-сервер провайдера послуг Інтернету (ISP DNS-server), IP-адресу якого комп'ютер користувача, як правило, отримує під час роботи протоколу DHCP. Також цей DNS-сервер зберігає певну кількість IP-адрес для найбільш часто запитуваних доменних імен, так званий DNS-cash.

Така архітектура DNS дозволяє розподілити навантаження й відповідальність за роботу системи між адміністраторами окремих доменів. До їхніх завдань входить забезпечення нормальної продуктивності при відповіді на запити до певної зони, підтримка унікальності імен у рамках зони, а також повідомлення адміністраторові батьківської зони про зміни в складі серверів, що обслуговують зону.

1.6 Засоби ефективного використання пулів IPv4-адрес

1.6.1 Поняття маски IPv4-адреси. Адресування на основі підмереж

Використання класового підходу IPv4-адресування мереж призводить до досить неефективного використання адресного ресурсу. Навіть для мереж класу C кількість вузлів в одному широкомовному домені складає 254, це занадто багато, тож частина адрес може лишатися не використаною. Для зменшення розміру мереж потрібні додаткові методи, які дозволяли б збільшити частину адреси, яка вказує на ідентифікатор підмережі, і, відповідно, зменшити частину, яка – на ідентифікатор вузла. Для цього використовують бітову маску (bit mask), яка дозволяє відокремити довільну частину IP-адреси для адресного простору ідентифікаторів вузлів. Така бітова маска називається ***маскою підмережі*** (subnet mask).

Маска підмережі (рисунок 1.54) – це 32-бітне число, двійковий запис якого містить одиниці в тих розрядах, які в IPv4-адресі повинні визначатися як ідентифікатор мережі, і нулі в тих розрядах, які в IPv4-адресі повинні визначатися як ідентифікатор вузла. Оскільки ідентифікатор мережі є цільною частиною IPv4-адреси, послідовність одиниць у масці підмережі повинна бути також безперервної.

Рисунок 1.54 – Використання маски підмережі

Щоб одержати адресу мережі, знаючи IPv4-адресу й маску підмережі, необхідно застосувати до них операцію *логічного множення («І»)* (рисунок 1.55). Інакше кажучи, у тих позиціях IPv4-адреси, у яких в масці підмережі є двійкові 1, перебуває ідентифікатор мережі, а де двійкові 0 – ідентифікатор вузла.

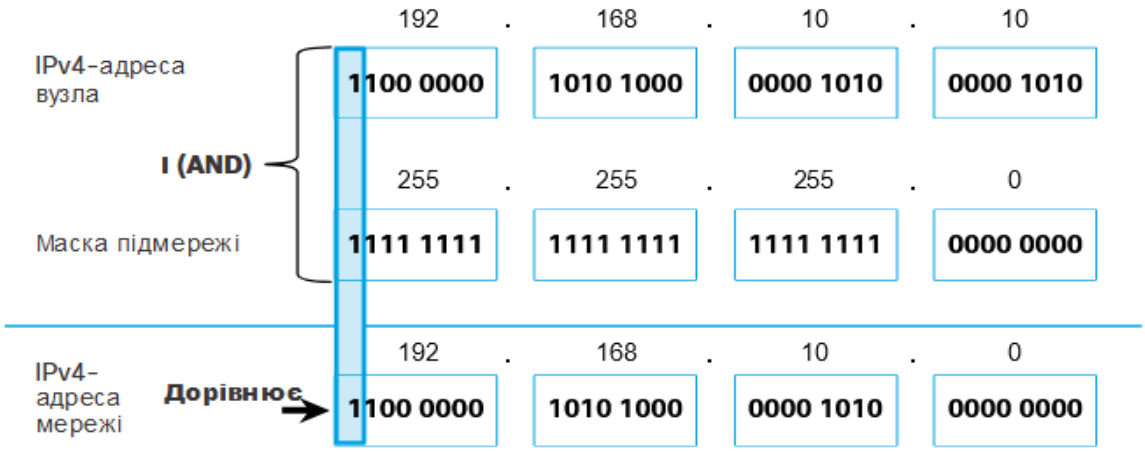


Рисунок 1.55 – Одержання адреси мережі з IP-адреси й маски підмережі

Для мереж класу A, B и C визначені фіксовані маски підмережі, які жорстко визначають кількість можливих IPv4-адрес. Технологія поділу мережі дає можливість створювати більше число мереж з меншою кількістю вузлів у них, що дозволяє ефективно використовувати адресний простір.

Для більш ефективного використання адресного простору були внесені зміни в існуючу класову систему адресації. В RFC 950 була описана процедура розбивки мереж на підмережі, і в структуру IPv4-адреси був доданий ще один рівень ієрархії – *підмереж* (subnetwork). Поява ще одного рівня ієрархії (рисунок

1.56) не змінило самої IPv4-адреси, вона залишилася 32-розрядною, а частина адреси, яка раніше відводилася під ідентифікатор вузла, була розділена на 2 частині – ідентифікатор підмережі й ідентифікатор вузла.

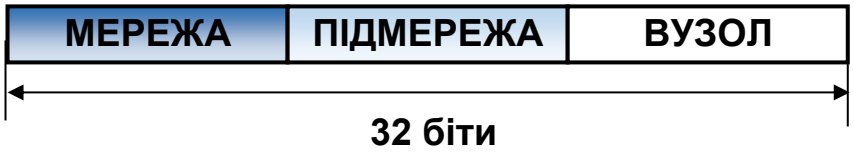


Рисунок 1.56 – Трьохрівнева ієрархія IP-адреси

Для обчислення кількості підмереж використовується формула 2^s, де s – кількість біт, зайнятих під ідентифікатор мережі із частини, відведеної під ідентифікатор вузла. Кількість вузлів у кожної підмережі обчислюється по формулі $2^n–2$, де n – кількість біт, що залишилися в частини, що ідентифікує вузол, а дві адреси – для адреси підмережі (перша) й для широкомовної адреси (остання) – в кожній отриманій підмережі зарезервовані.

Розбивка однієї великої мережі на більш дрібні (рисунок 1.57) дозволяє:

– раціонально використовувати адресний простір (тобто виділити для сегмента мережі блок адрес не цілком класу A, B або C, а тільки частину класової мережі);

– підвищити безпеку й керованість мережі (за рахунок зменшення розмірів сегментів і ізоляції трафіка сегментів один одного).

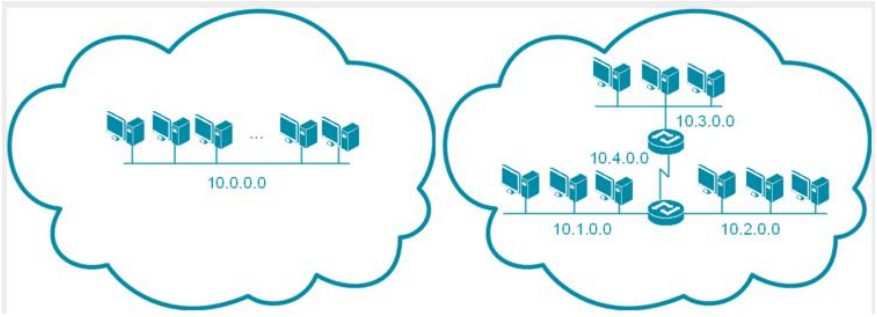


Рисунок 1.57 – Розбивка мережі на підмережі

Наприклад, організації необхідно розбити мережу 192.168.1.0 на 20 підмереж по 6 комп'ютерів у кожній. Для початку необхідно визначити, до якого класу відноситься адреса. 192.168.1.0 – це клас C, відповідно, стандартна маска підмережі для класу C дорівнює 255.255.255.0 і під ідентифікатор вузла відведений весь четвертий октет. Потім визначається кількість біт четвертого октету, потрібних для формування 20 підмереж. Оскільки знайти число, при якому ступінь 2 буде дорівнювати 20 неможливо, вибираємо найближче більше число $2^5 = 32$. Таким чином, 5 перших біт 4-го октету будуть використані для

ідентифікації підмережі, а 3 біта, що залишилися – для ідентифікації вузлів у них (рисунок 1.58).

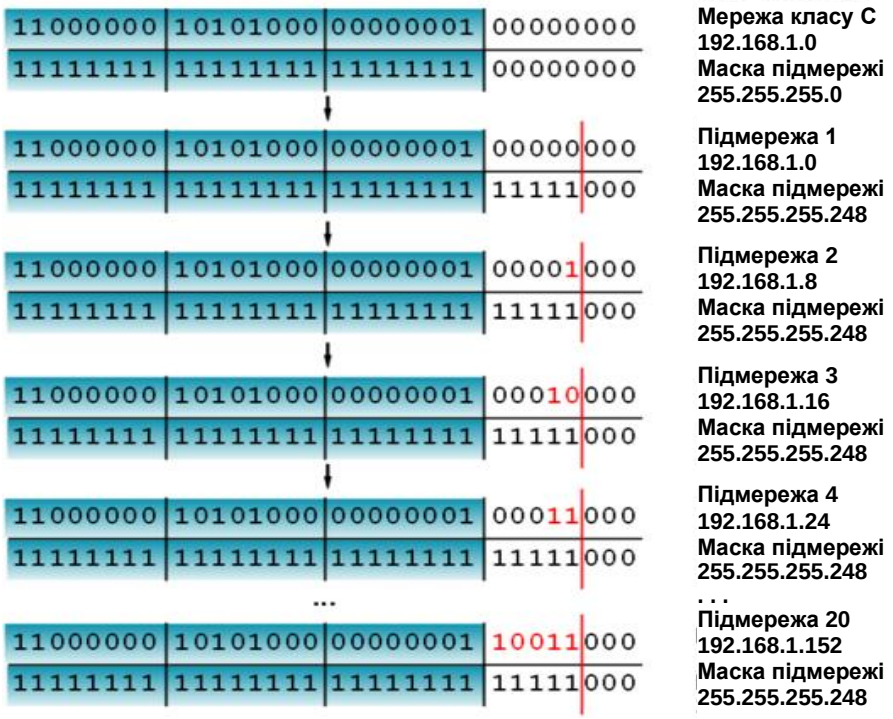


Рисунок 1.58 – Приклад розбивки мережі 192.168.1.0 на підмережі

Щоб уникнути проблем з адресацією й маршрутизацією всі мережні пристрої TCP/IP в одному сегменті мережі повинні використовувати ту саму маску підмережі.

1.6.2 Безкласове адресування

Класова модель IPv4-адресації виявилася нераціональної з погляду ефективного використання адресного простору. Навіть при використанні підмереж, у випадку класової адресації мережу можна було розбити тільки на підмережі однакового розміру. При цьому, якщо обрана маска підмережі забезпечує потрібну кількість підмереж, можливо, що припустимої кількості вузлів для кожної підмережі буде недостатньо або, навпаки, частина адрес не буде використана. Наприклад, велика кількість вузлів є надлишковою для підмережі, яка зв'язує два маршрутизатори по каналу «точка-точка». У цьому випадку необхідно всього дві IPv4-адреси для адресації інтерфейсів сусідніх маршрутизаторів. Таким чином, розбивка мережі на підмережі різного розміру дозволила б раціонально використовувати адресний простір.

Поступово з ростом мережі Інтернет відбулася відмова від класової схеми, і була прийнята *безкласова модель IPv4-адресації*, у якій відсутня прив'язка до класу мережі й масці підмережі за замовчуванням. Безкласова адресація використовує *маски підмережі змінної довжини* (Variable Length Subnet Mask, VLSM) і *технологію безкласової міждоменої маршрутизації*

(Classless Inter Domain Routing, CIDR). Термін «маска змінної довжини» означає, що мережа може бути розбита на підмережі з різними масками підмережі. Основна ідея застосування VLSM полягає в тому, що можна розбити мережу на підмережі, потім підмережі розбити ще на підмережі точно в такий же спосіб, як була розбита первісна мережа. Тобто мережа може бути розбита на підмережі різних розмірів, з різними масками. Маски підмережі є основою методу безкласової маршрутизації й записуються у вигляді нотації «IP-адреса/довжина префіксу». Число після «/» означає кількість одиничних розрядів у масці підмережі. Наприклад, мережна адреса 192.168.1.8 з маскою підмережі 255.255.255.248 також може бути записана, як 192.168.1.8/29. Число 29 указує, що в масці підмережі 255.255.255.248 є 29 одиничних бітів.

Розподіл мережі на підмережі з використанням масок змінної довжини аналогічно традиційному розподілу на підмережі. Розглянемо приклад, показаний на рисунку 1.59.

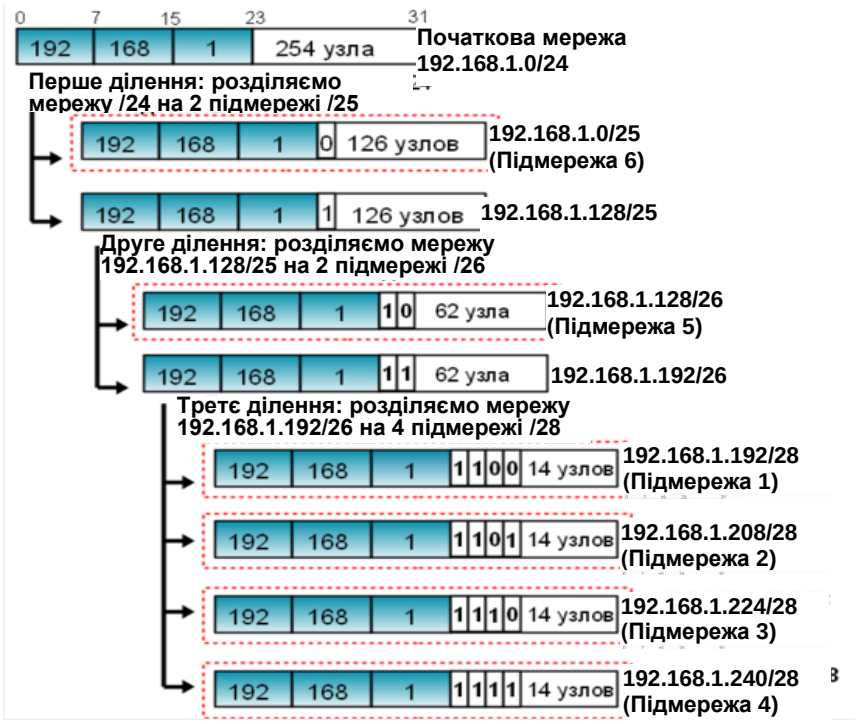


Рисунок 1.59 – Приклад розбивки мережі 192.168.1.0/24 на підмережі за допомогою VLSM

В даному прикладі для організації виділена мережа класу С 192.168.1.0/24. Потрібно розділити її на 6 підмереж. У підмережах 1, 2, 3 і 4 повинне бути 10 вузлів, в 5-й підмережі – 50 вузлів, в 6-й підмережі – 100. Теоретично для мережі 192.168.1.0/24 припустима кількість вузлів дорівнює 254, і розбити таку мережу на підмережі з необхідною кількістю вузлів без використання VLSM неможливо.

Спочатку необхідно розділити мережу 192.168.1.0/24 на дві підмережі. Для цього з 4-го октету необхідно зайняти 1 біт для ідентифікатора підмережі, таким чином, для ідентифікації вузлів залишиться 7 біт. У підсумку виходить дві

підмережі 192.168.1.0/25 і 192.168.1.128/25, у кожній з яких може бути по 126 (2^7–2) вузлів. Першу з них залишимо, тому що потрібно, щоб в 6-й підмережі було 100 вузлів, а другу розділимо ще на дві підмережі. Для цього заберемо 1 біт з тих 7 біт відведених під ідентифікатор вузла, що залишилися. Таким чином, виходить дві підмережі 192.168.1.128/26 і 192.168.1.192/26, у кожній з яких припустима кількість вузлів дорівнює 62 (2^6–2). Першу підмережу необхідно залишити для 5-й підмережі, в якій повинне бути 50 вузлів, а із другої підмережі сформувати ще чотири підмережі. Для цього займемо ще 2 біта з 6 біт, відведених під ідентифікатор вузла, що залишилися. У результаті одержимо чотири підмережі з 14 (2^4–2) вузлами в кожній, що дозволить адресувати необхідну кількість вузлів, необхідних для підмереж 1, 2, 3 і 4.

1.6.3 Порядок роботи протоколу DHCP

DHCP (англ. *Dynamic Host Configuration Protocol* – протокол динамічної конфігурації вузла) – це стандартний протокол прикладного рівня, який дозволяє комп'ютерам автоматично отримувати IP-адресу та інші параметри, необхідні для роботи в мережі. Для цього комп'ютер звертається до DHCP-сервера. Мережний адміністратор може задати діапазон адрес, які будуть розподілені між комп'ютерами. Це дозволяє уникнути ручного налаштування комп'ютерів мережі й зменшує кількість помилок. Протокол DHCP використовується в більшості великих мереж TCP/IP.

Крім IP-адреси, DHCP також може повідомляти клієнтові додаткові параметри, необхідні для нормальної роботи в мережі. Ці параметри називають опціями DHCP. Список стандартних опцій можна знайти в RFC 2132. Деякими з найбільш часто використовуваних опцій є:

- IP-адреса маршрутизатора за замовчуванням;
- маска підмережі;
- адреси серверів DNS;
- ім'я домену DNS.

Деякі постачальники програмного забезпечення можуть визначати власні, додаткові опції DHCP.

Протокол DHCP працює за схемою клієнт-сервер. Під час запуску системи комп'ютер, який є DHCP-клієнтом, відправляє в мережу запит на отримання IP-адреси. DHCP-сервер відповідає і відправляє повідомлення-відповідь, яка містить IP-адресу і деякі інші конфігураційні параметри. При цьому сервер DHCP може працювати в різних режимах, включаючи:

1. **Динамічний розподіл** – адміністратор присвоює IP-діапазон адрес на сервері DHCP. Кожен клієнтський вузол на етапі ініціалізації в мережі повинен запросити IP-адресу від DHCP-сервера за принципом «оренди». DHCP-сервер надає клієнтському вузлу першу з вільних IP-адрес. Коли закінчується термін оренди, якщо вона не буде продовжена за ініціативою клієнтського вузла, DHCP-сервер має право повернути адресу і призначити її на інші комп'ютери.

2. **Автоматичне виділення** – аналогічно попередньому режиму, DHCP-сервер призначатиме клієнтському вузлу, за його запитом, вільну IP-адресу з діапазону, встановленого адміністратором. Основна відмінність від режиму динамічного розподілу полягатиме в тому, що сервер зберігає записи минулих розподілів IP-адрес і намагається привласнити ту саме адресу тому ж вузлу для наступних мережних підключень.

3. **Статичний розподіл** – DHCP-сервер робить призначення IP-адрес виключно на основі таблиці MAC-адрес, які зазвичай заповнені вручну адміністратором мережі. Отже, якщо MAC-адреса вузла не зазначена в таблиці, йому не буде призначена мережна адреса.

DHCPv4

Протокол DHCP побудований так, що клієнт може звертатися із запитом відразу до декількох серверів.

Клієнт DHCP, що потребує адресу, відправляє широкомовний кадр запиту *DHCPDISCOVER* в пошуках сервера. Кадр містить MAC-адресу клієнта, відправника запиту. Потім один або кілька DHCP-серверів розглядають запит і відправляють у відповідь одноадресний (безпосередньо на адресу клієнта) кадр пропозиції *DHCPOFFER*, що містить пропоновану IP-адресу і «час оренди».

Клієнт обирає адресу з отриманих кадрів DHCPOFFER. Вибір клієнта залежить від його налаштувань – наприклад, він може вибрати адресу з найбільшим часом оренди. За результатом вибору клієнт відправляє кадр *DHCPREQUEST* з адресою вибраного сервера. Кадр DHCPREQUEST також має широкомовну MAC-адресу розсилки, з метою проінформувати всі DHCP-сервери, від яких надійшли пропозиції. Інші сервери переглядають пакет DHCPREQUEST і розуміють, що їх пропозиція була відкинута. Таким чином, вони знають, що запропоновані ними IP-адреси вільні для призначення іншим клієнтам.

Обраний сервер посилає підтвердження – одноадресний кадр *DHCPACK* і процес узгодження завершується. Пакет DHCPACK містить узгоджену IP-адресу, час оренди та інші мережеві параметри, що пропонує DHCP-сервер. Сервер позначає виділену адресу як зайняту, до закінчення терміну оренди цю адресу не можна буде присвоїти іншому клієнту. Клієнт конфігурує свої налаштування відповідно до отриманих даних і може працювати в мережі.

У разі якщо сервер не може прийняти конфігурацію, він посилає пакет DHCPNAK (відмова в підтвердженні), що змушує клієнта почати процес узгодження заново. За наявності в мережі двох DHCP-серверів з різними конфігураціями, результат вибору клієнтом певного з них не є прогнозованим.

Якщо DHCP-сервер, розташований на віддаленому (рисунок 1.60) маршрутизаторі (R3), в іншій мережі і централізовано видає адреси в усі локальні мережі (LAN_1 і LAN_2), то необхідна конфігурація агентів DHCP-relay на маршрутизаторах, до яких підключені ці локальні мережі (R1 і R2). Необхідність DHCP-relay пов’язана з неможливістю трансляції широкомовних кадрів за межі локальної мережі (широкомовного домену), а сутність полягає в пересиланні

вмісту широкомовного кадру від клієнта одноадресним пакетом до DHCP-сервера в іншій мережі.

Всі повідомлення за протоколом DHCP, що формує клієнт, на транспортному рівні адресовані на порт 67(UDP), а сервер відповідає на порт 68 (UDP).

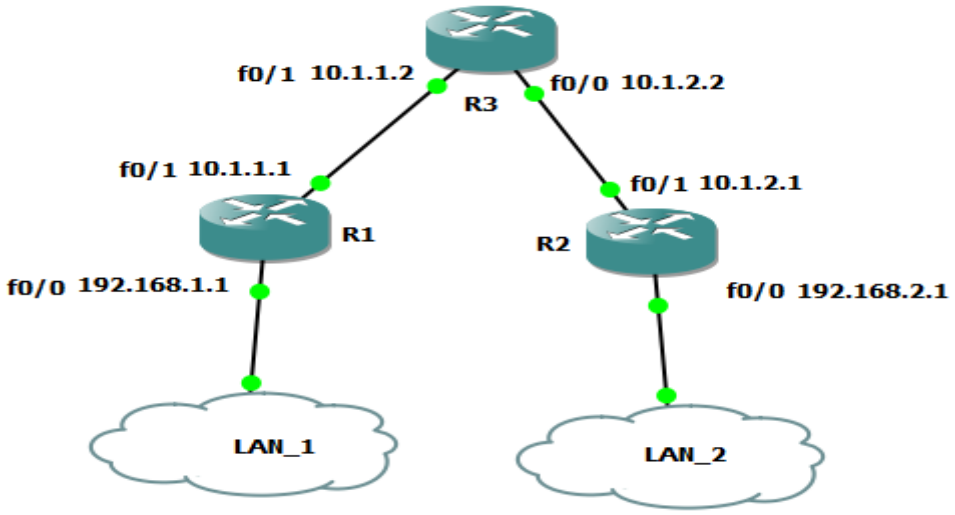


Рисунок 1.60 – DHCP-сервер, розташований на віддаленому маршрутизаторі

DHCPv6 (Dynamic host configuration protocol for IPv6)

В мережах, де використовується IP-адресування версії 6 (IPv6), а маршрутизатори підтримують IPv6-маршрутизацію механізми отримання вузлами мережевих параметрів носять більш складний і варіативний характер.

Можливі три варіанти взаємодії між IPv6-вузлом, IPv6-маршрутизатором і власне DHCPv6-сервером:

1. ***SLAAC (Stateless Address Autoconfiguration, Автоматична конфігурація адреси без збереження стану)*** – це спосіб, який дозволяє вузлу отримати для своєї мережі – префікс, його довжину і адресу шлюзу за замовченням. Ця інформація надходить від маршрутизатора IPv6, до якого підключена дана мережа, без допомоги DHCPv6-сервера. Передавання та отримання необхідної інформації здійснюється за допомогою повідомлень «Оголошення маршрутизатора IPv6» (Router Advertisement – RA).

IPv6-маршрутизатори періодично відправляють повідомлення RA всім вузлам у мережі під управлінням IPv6. За замовчуванням маршрутизатори Cisco відправляють такі повідомлення кожні 200 секунд на групову адресу IPv6-вузлів ff02::1, в подальшому ці повідомлення служать підтвердженням працездатності маршрутизатора IPv6. Ключовими позиціями повідомлення RA, що визначають режим взаємодії SLAAC, є значення «прапорців» у заголовку: A=1, M=0, O=0. Ця комбінація означає, що дане повідомлення RA містить значення префіксу мережі, довжину цього префіксу і IPv6-адресу шлюзу за замовчуванням, які вузол має використовувати для формування своїх мережевих параметрів. Ніяких інших параметрів вузол отримати не зможе, оскільки DHCPv6-сервер в такому режимі взаємодії – відсутній.

Фактично для роботи в мережі вузлу залишається сформувати частину своєї IPv6-адреси, яка буде нести інформацію про ідентифікатор інтерфейсу, довжиною 64 біти. Ця процедура здійснюється або шляхом реалізації процедури EUI-64 (рисунок 1.61) – на основі MAC-адреси відповідного мережевого інтерфейсу, або генерацією випадковим чином.

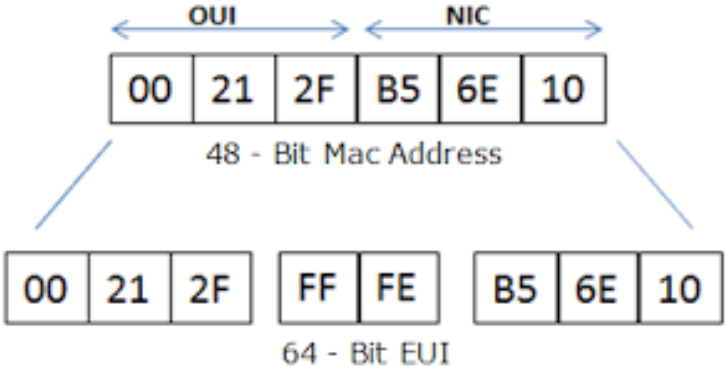


Рисунок 1.61 – Пояснення процедури EUI-64

IPv6-вузол, який щойно включився до роботи в мережі, не має необхідності чекати наступного повідомлення RA, він відправляє повідомлення «Запит маршрутизатора IPv6» (Router Solicitation, RS), який використовує адресу: ff02::2 – групової передачі всім IPv6-маршрутизаторам. За наявності в мережі відповідного маршрутизатора, здатного надати потрібну інформацію, він сформує відповідь у вигляді повідомлення RA.

Слід зауважити, що IPv6-маршрутизація не включена за замовчуванням, отже щоб маршрутизатор працював як IPv6-маршрутизатор, необхідно використовувати команду глобальної конфігурації: ***ipv6 unicast-routing***. Зазначені повідомлення (RA та RS) не є повідомленнями протоколу DHCPv6, вони належать до протоколу ICMPv6. Зазначений протокол (ICMPv6) має багато додаткових функцій порівняно з ICMPv4, які перекривають роботу деяких окремих протоколів (наприклад, ARP).

Сформовані в результаті процедури SLAAC вузлом мережеві параметри остаточно стають діючими після перевірки можливого дублювання адрес. Дублювання може виникнути внаслідок некоректного використання однакових MAC-адрес, або генерації двох однакових випадкових IPv6-адрес. Перевірка здійснюється шляхом групового розсилання (ff02::1) вузлом повідомлення ICMPv6 «Запит сусіда» (Neighbor Solicitation, NS). Якщо будь-який вузол мережі визначає, що IPv6-адреса відправника повідомлення NS збігається з його адресою, то він у відповідь формує повідомлення ICMPv6 «Оголошення сусіда» (Neighbor Advertisement, NA). Наявність NA визначає вузлу необхідність сгенерувати новий ідентифікатор інтерфейсу, відсутність – можливість використання наявних мережевих параметрів.

2. ***SLAAC разом з stateless DHCPv6*** – це спосіб, який додатково до попереднього дозволяє вузлу, крім отримання від маршрутизатора IPv6 основних мережевих параметрів, взаємодіяти з власне DHCPv6-сервером для одержання додаткових параметрів (адреси DNS-сервера, імені домену та інших).

Даний режим взаємодії можливий, якщо у заголовку повідомлення RA встановлені такі значення «прапорців»: A=1, M=0, O=1.

Перший етап взаємодії з вузла з IPv6-маршрутизатором відбувається абсолютно аналогічно режиму SLAAC і, за допомогою повідомлень RS та RA вузол отримує префікс мережі, його довжину і адресу шлюзу за замовченням. Проте наявність у заголовку RA «прапорця» O=1 означає для вузла можливість звернення до DHCPv6-сервера за додатковими параметрами.

Взаємодія вузла з DHCPv6-сервером при цьому обмежується двома кроками:

I крок. Багатоадресне повідомлення Information-Request (запит інформації від Клієнта до Серверу): клієнт розсилає це повідомлення, щоб знайти доступні DHCPv6-сервери в мережі, він не запитує основні мережеві параметри, оскільки вже отримав їх від маршрутизатора. Повідомлення містить перелік потрібних параметрів.

II крок. Одноадресне повідомлення Reply (відповідь від Серверу до Клієнта): сервер повідомляє про свою наявність та пропонує запитані параметри конфігурації.

Наявність в назві режиму слова *stateless* визначає, що DHCPv6-сервер в даному випадку працює без відстеження стану адреси, лише для передачі додаткових параметрів. Отже, IPv6-адреса вузла генерується самим вузлом і, аналогічно режиму SLAAC, має бути перевірена за допомогою повідомлень NS та NA на дублювання.

3. ***Stateful DHCPv6 (DHCPv6 з відстеженням стану адреси)*** – в даному режимі у заголовку повідомлення RA встановлені значення «прапорців»: A=0, M=1, O=1. Це означає для вузла заборону на використання інформації з повідомлення RA, а необхідність звернення для отримання всіх мережевих і додаткових параметрів від DHCPv6-сервера.

Взаємодія вузла з DHCPv6-сервером також (як і у DHCPv4) відбувається у чотири кроки:

I крок. Багатоадресне повідомлення Solicit (пошук від Клієнта до Серверу): клієнт розсилає це повідомлення, щоб знайти доступні DHCPv6-сервери в мережі.

II крок. Одноадресне повідомлення Advertise (оголошення від Серверу до Клієнта): сервер повідомляє про свою наявність та пропонує параметри конфігурації.

III крок. Багатоадресне повідомлення Request (запит від Клієнта до Серверу): клієнт повідомляє про прийняття і запитує підтвердження на використання запропонованих параметрів.

IV крок. Одноадресне повідомлення Reply (відповідь від Серверу до Клієнта): сервер підтверджує призначення параметрів і налаштувань.

Наведені повідомлення протоколе DHCPv6, які сформовані клієнтом, на транспортному рівні адресовані на порт 546 (UDP), а сервер відповідає на порт 547 (UDP).

В цьому випадку DHCPv6-сервер працює як типовий DHCP-сервер, який фіксує всі дані які виділяє. Тож відстеження IPv6-адрес, щоб не призначати одну й ту саму IPv6-адресу на декількох пристроях, відбувається у самому DHCPv6-сервері.

1.6.4 Налаштування протоколу DHCP

IP-адреси, які буде роздавати DHCP-сервер, що реалізований на вузлі у вигляді спеціалізованого програмного забезпечення, зберігаються в DHCP пулі (pool). Приклад налаштування такого пулу адрес показаний на рисунку 1.62. Потрібно вказати назву пулу (Pool Name), порт за умовчанням (Default Gateway), IP-адресу DNS-сервера, початкову адресу пулу, маску підмережі та максимальну кількість адрес, які можна роздати.

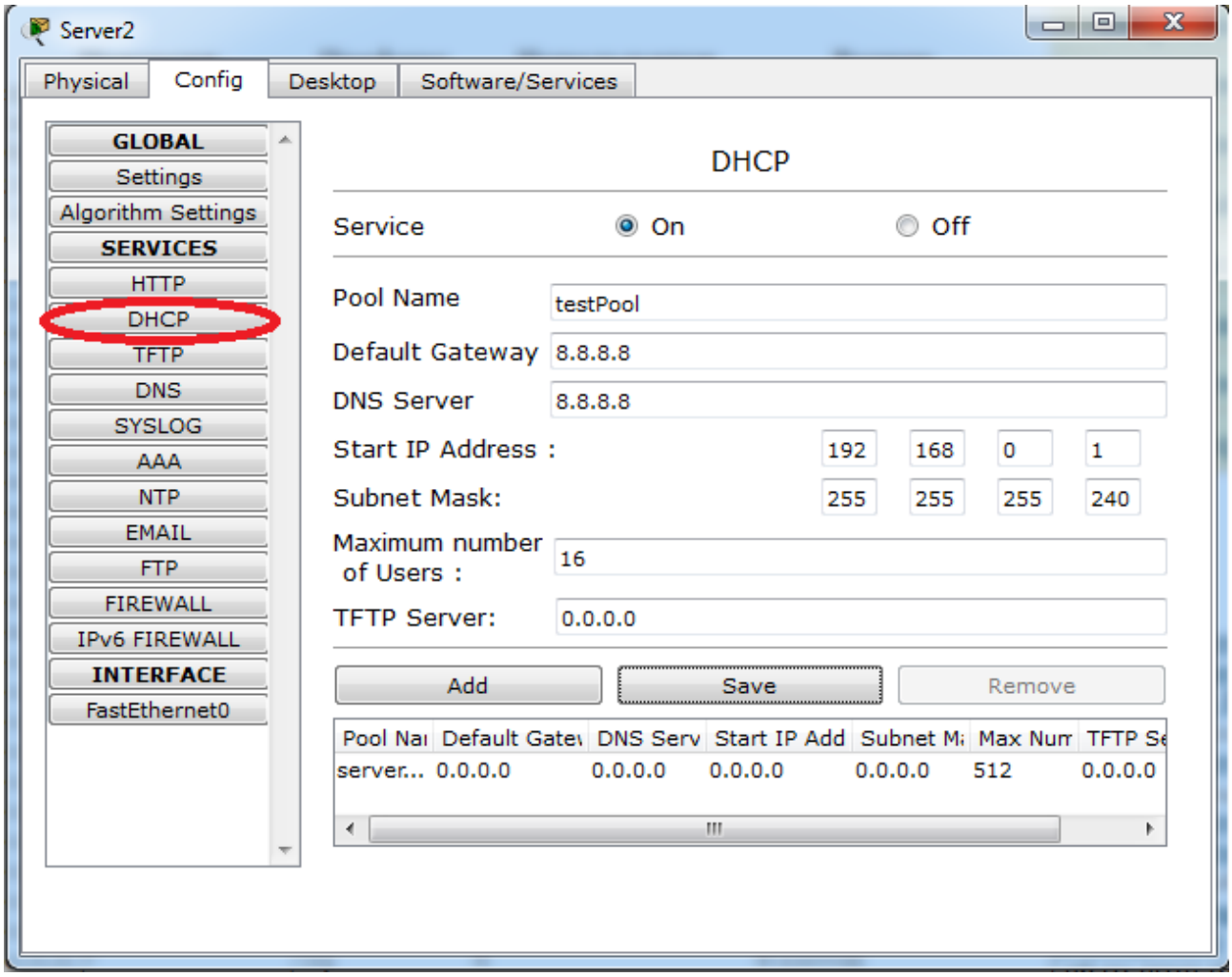


Рисунок 1.62 – Приклад налаштування DHCP пулу

Маршрутизатори і комутатори також можуть виконувати роль DHCP-сервера. Налаштування протоколу DHCP на цих пристроях виконується за допомогою наступних команд:

- *ip dhcp pool [ім'я пулу]* – створення пулу з вказаним ім'ям;

- *network [маска підмережі] [IP-адреса підмережі]* – перелік IP-адрес, що використовуватиметься для розповсюдження з вказаними параметрами;
- *default-router [IP-адреса інтерфейсу маршрутизатора в підмережі]* – визначає інтерфейс вихідного шлюзу за замовчанням для даного пулу;
- *dns-server [IP-адреса DNS-сервера]* – встановлює адресу DNS-сервера для даного пулу;
- *interface [ID інтерфейсу маршрутизатора]* – перехід у режим конфігурування інтерфейсу;
- *ip address dhcp* – включення режиму налаштування параметрів інтерфейсу від DHCP;

Приклад:

Router(config)#ip dhcp pool routerPool
Router(dhcp-config)#network 192.168.0.0 255.255.255.240
Router(dhcp-config)#default-router 8.8.8.8
Router(dhcp-config)#dns-server 8.8.8.8
Router(dhcp-config)#exit
Router(config)# interface fastethernet 0/1
Router(config-if)# ip address dhcp

Часто необхідно видалити з пулу адреси які вже закріплені за інтерфейсами роутера чи кінцевого пристрою. Для цього використовується наступна команда:

- *ip dhcp excluded-address [IP-адреса, яку потрібно виключити з пулу]* – виключення IP-адреси з пулу.

Приклад:

Router(config)#ip dhcp excluded-address 8.8.8.1

1.7. Основи роботи з програмою емуляції мереж CiscoPacketTracer

1.7.1. Головне вікно програми та його можливості

Cisco Packet Tracer – це потужний навчальний емулятор мережевої інфраструктури, призначений для моделювання, проектування, конфігурування та дослідження переважної більшості мережевих технологій та протоколів. Він дозволяє створювати віртуальні топології з обладнанням Cisco, вивчати протоколи та відпрацьовувати навички адміністрування без реального обладнання.

Після запуску програми Packet Tracer відкривається головне вікно програми (рисунок 1.63). Воно містить 6 основних меню, 4 з яких використовуються найбільш часто.

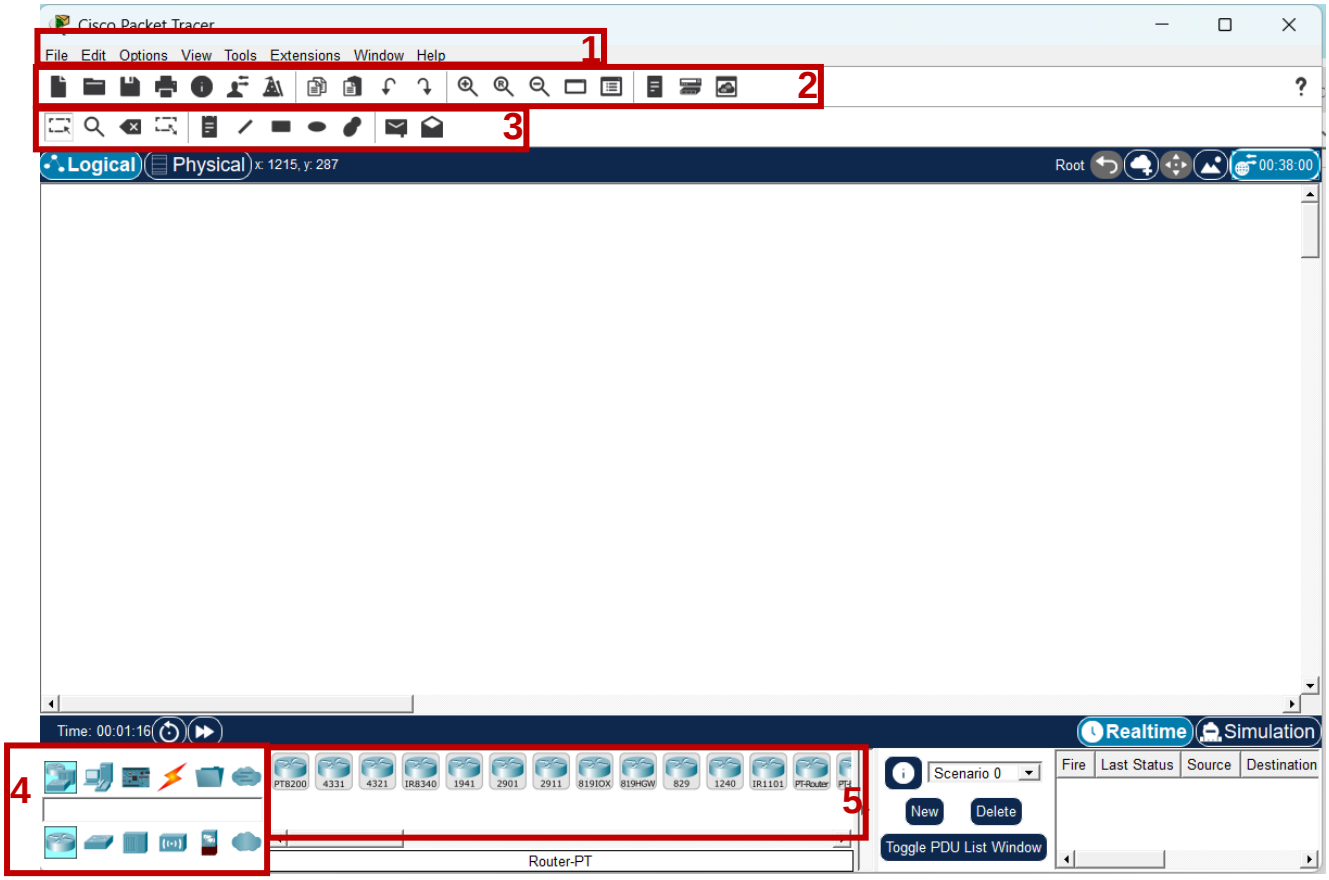


Рисунок 1.63 – Головне вікно програми Cisco Packet Tracer

Головне меню програми містить такі розділи меню (1):

- *Файл (File)* – містить операції відкриття/збереження документів;
- *Виправлення (Edit)* – стандартні операції копіювати/вирізати/скасувати/повторити;
- *Налаштування (options)* – додаткові налаштування програми;
- *Вид (View)* – масштаб робочої області і панелі інструментів;
- *Інструменти (Tools)* – колірна палітра і кастомізація кінцевих пристроїв;
- *Розширення (Extention)* – майстер проектів;
- *Допомога (Help)* – інформація про програму, довідка;

Панель інструментів (2) містить досить впізнавані піктограми інструментів, які переважно дублюють основні розділи головного меню. Кожен з розташованих в даному меню інструментів активується кліком мишки на відповідній піктограмі, а також клавіатурою.

Додаткова панель інструментів (3) містить піктограми для вибору інструментів для виконання дій з об'єктами у робочому полі. Вони переважно мають стандартний для своїх операційних систем вигляд і призначення. Особливими є інструменти: Inspect ([I]), який нагадує лупу (рисунок 1.64), але використовується для перегляду специфічних параметрів мережевих пристроїв (таблиці ARP, таблиці маршрутизації, таблиці NAT і інші) та Add Simple PDU ([P]) і Add Complex PDU ([C]) призначені для емуляції відправки з подальшим відстеженням довільного пакету даних всередині проекту (рисунок 1.65).

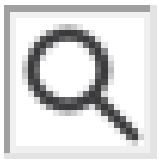


Рисунок 1.64 – Піктограма інструменту Inspect

Рисунок 1.65 – Піктограми інструментів Add Simple PDU і Add Complex PDU

Дві нижніх панелі компонентів (4 і 5) знаходяться в лівому кутку програми є найбільш використовуваними. Панель (4) дозволяє вибрати тип пристроїв, а панель (5) безпосередньо сам пристрій. Існує можливість використання таких «Мережевих пристроїв» (Network Devices):

1. Маршрутизатори Cisco.
2. Комутатори Cisco 2 та 3 рівня.
3. Концентратор Cisco, повторювач та дільник сигналу для коаксіальних кабелів.
4. Бездротові пристрої.
5. Брандмауери.
6. Пристрої емуляції WAN-мереж.

Кнопка «Кінцеві пристрої» (End Devices) дозволяє обрати велике різноманіття відповідних мережевих пристроїв, а також кінцеві пристрої для систем IoT (Internet of Things).

Кнопка «З'єднання» (Connections, рисунок 1.66) дозволяє обрати типи підключень (кабелів) для з'єднання обладнання проєкту. Packet Tracer підтримує широкий діапазон мережевих з'єднань, але кожен тип кабелю може бути з'єднаний лише з певними типами інтерфейсів.

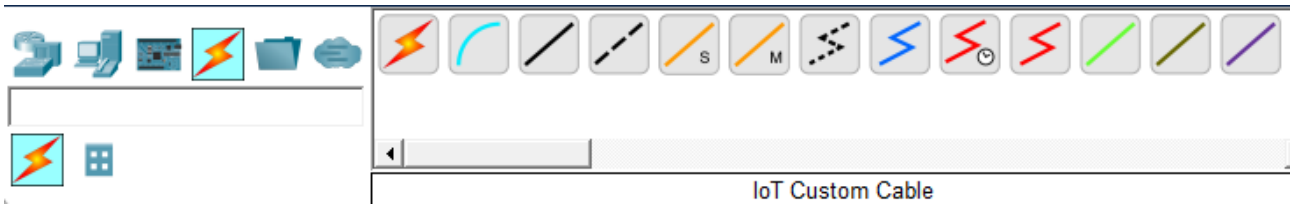


Рисунок 1.66 – Типи підключень (кабелів) доступних в Packet Tracer

Можливе використання таких підключень (кабелів) обладнання:

1. Auto – автоматичне визначення типу з'єднання (такої можливості в реальному обладнанні не існує, тому краще цей варіант виключити із застосування).

2. Console – з'єднання за допомогою консольного кабелю (COM-порт (RS-232) на робочому терміналі (комп'ютері) та вхід Console на пристроях Cisco).

Після з’єднання треба запустити довільний термінальний додаток (TeraTerm, Putty тощо), в Packet Tracer це емулюється додатком Terminal. Треба виконати деякі вимоги для роботи консольного сеансу з терміналом: швидкість з’єднання з обох сторін повинна бути однакова, має бути 7 біт даних (або 8 біт) для обох сторін, контроль парності повинен бути однаковий, має бути 1 або 2 степових біта (але вони не обов’язково повинні бути однаковими), а потік даних може бути чим завгодно для обох сторін.

3. Copper Straight-Through – з’єднання за допомогою «прямого» кабелю типу вита пара. Цей тип кабелю є стандартним середовищем передачі Ethernet для з’єднання пристроїв, які функціонують на різних рівнях OSI. Він може з’єднуватися з наступними типами портів: мідний 10 Мбіт/с (Ethernet), мідний 100 Мбіт/с (FastEthernet) і мідний 1000 Мбіт/с (GigabitEthernet).

4. Copper Cross-Over – з’єднання за допомогою перехресного (крос, cross) кабелю типу вита пара. Цей тип кабелю є середовищем передачі Ethernet для з’єднання пристроїв, які функціонують на однакових рівнях OSI. Він може бути з’єднуватися з аналогічними типами портів: мідний 10 Мбіт/с (Ethernet), мідний 100 Мбіт/с (FastEthernet) і мідний 1000 Мбіт/с (GigabitEthernet).

5. Single Mode Fiber – з’єднання за допомогою одномодового оптичного волокна. Оптичне середовище використовується для з’єднання між оптичними портами (SFP), які призначені для роботи з одномодовим оптичним середовищем.

6. Multi Mode Fiber – з’єднання за допомогою багатомодового оптичного волокна. Оптичне середовище використовується для з’єднання між оптичними портами (SFP), які призначені для роботи з багатомодовим оптичним середовищем.

7. Phone – з’єднання за допомогою телефонної лінії. З’єднання через телефонну лінію може бути здійснено тільки між пристроями, що мають модемні порти. Стандартне подання модемного з’єднання – це кінцевий пристрій (наприклад, комп’ютер), який додзвонюється в мережеву хмару.

8. Coaxial – з’єднання за допомогою коаксіального кабелю. Коаксіальне середовище використовується для з’єднання між коаксіальними портами, такі як кабельний модем, з’єднаний з хмарою Packet Tracer.

9. Serial DCE і Serial DTE – послідовні канали зв’язку. З’єднання через послідовні порти, часто використовуються для зв’язків мереж WAN. Один кабель має з одного боку роз’єм DCE, а з іншого – DTE. Сторона DCE підключається в бік обладнання мережі WAN – модем або маршрутизатор постачальника мережевих послуг, а сторона DTE в бік кінцевого обладання. Для настройки таких з’єднань необхідно встановити синхронізацію зі стороні DCE-пристрою. Синхронізація DTE виконується автоматично відповідно налаштувань зустрічного DCE. Сторону DCE можна визначити по маленькій іконці «годинника» поруч з портом. При виборі типу з’єднання Serial DCE, перший пристрій, до якого застосовується з’єднання, стає DCE пристроєм, а другий – автоматично стане стороною DTE. Можливо і зворотне розташування сторін, якщо обраний тип з’єднання Serial DTE.

10. Octal і IoT Custom Cable – з’єднання, що використовуються для реалізації систем та мереж IoT.

11. USB – кабель для підключення обладнання з портами USB.

Мережа створюється шляхом вибору необхідних активних компонентів мережі у панелі компонентів та розміщенні їх на робочому полі, а також з’єднання компонентів кабелями. Кабелі беруться також з панелі компонентів.

Для вибору компоненту необхідно: а) клацнути по ньому лівою кнопкою миші у панелі компонентів, б) клацнути на робочому полі, на місце необхідного розташування компоненту.

Для з’єднання кабелем необхідно: а) клацнути на зображення кабелю, б) клацнути на зображення компоненту, до якого необхідно підключити кабель – з’явиться зображення вільних інтерфейсів компоненту, в) клацнути зображення з’єднуваного інтерфейсу, г) аналогічним чином підключити другий кінець кабелю до необхідного інтерфейсу другого компоненту.

На робочому полі компоненти мережі можуть вільно переміщуватися та можуть бути видалені.

1.7.2 Основні команди та режими операційної системі Cisco IOS

Адміністрування пристроями Cisco здійснюється переважно через інтерфейс командного рядка (CLI). Доступ до командного рядка адміністратор може отримати трьома варіантами:

1. Консольний доступ (Console) – це пряме фізичне підключення до пристрою, яке зазвичай використовується для початкового налаштування або відновлення доступу, коли мережа недоступна.

Потребує прямого з’єднання консольним кабелем (rollover, блакитного кольору) спеціального інтерфейсу (RJ-45) на обладнанні Cisco з написом «Console» з COM-портом (RS-232) комп’ютера або з USB через перехідник та використання термінальної програми.

2. Віддалений доступ через мережу (VTY)

Дозволяє керувати пристроєм через IP-мережу з будь-якої точки. Для цього на пристрої мають бути налаштовані IP-адреса та віртуальні лінії (line VTY). При цьому можливе використання протоколів Telnet або SSH (Secure Shell), обидва забезпечують віддалений доступ мережевих пристроїв через текстовий термінальний інтерфейс, створюючи можливість працювати в його командному рядку. Проте SSH (порт 22(TCP)) на відміну від Telnet (порт 23(TCP)) забезпечує шифрування даних та паролів при передачі, що робить його більш захищеним і безпечним.

3. Допоміжний порт (AUX)

Використовується для віддаленого доступу до пристрою «поза смугою» (out-of-band), зазвичай через модем і телефонну лінію. Дозволяє отримати доступ до консолі, якщо основна мережа вийшла з ладу, але потребує підключення телефонної лінії через модем до порту AUX на обладнанні Cisco.

Після отримання доступу до командного рядка пристрою Cisco адміністратор починає управляти IOS, яке може перебувати в наступних режимах:

- **Користувацький режим (User EXEC Mode)**

Ознакою режиму є позначка «>» після назви пристрою. Режим вмикається одразу після підключення до командного рядка і дозволяє лише переглядати базову інформацію про стан пристрою та зв'язки з сусідніми пристроями (наприклад, *ping*, *traceroute*).

- **Привілейований режим (Privileged EXEC Mode)**

Ознакою режиму є позначка «#» після назви пристрою. Перехід в режим відбувається з користувацького режиму виконанням команди входу: *enable*. Режим дозволяє переглядати детальну статистику та всі налаштування, керувати файлами (копіювання, видалення), робити налаштування, які не стосуються основних функцій пристрою та зберігати поточні налаштування (конфігурацію).

- **Режим глобальної конфігурації (Global Configuration Mode)**

Ознакою режиму є слово config в дужках та позначка «#» після назви пристрою ((config)#). Перехід в режим відбувається з привілейованого режиму виконанням команди входу: *configure terminal*. Режим дозволяє робити всі налаштування, які впливають на весь пристрій загалом (як то назва пристрою, банери, паролі тощо).

- **Режими часткових налаштувань конфігурації**

Ознакою режимів є додаткове слово (наприклад -line, -router, -if тощо) після слова config в дужках та позначка «#» після назви пристрою ((config-if)#). Режими дозволяють робити налаштування, які впливають лише на конкретний інтерфейс, лінію або протокол. Перехід в режими відбувається з режиму глобальної конфігурації виконанням команд, які містять назву конкретного інтерфейсу, лінії або протоколу.

Для повернення з вищого режиму в попередній використовується команда *exit* (в певних режимах є додаткові команди з аналогічною дією, як то *disable*, *logout*, *end*, але *exit* є універсальною). Можливе пряме повернення з будь-якого нижчого режиму до привілейованого натисненням комбінації клавіш *<Ctrl>+<Z>*.

Для збереження поточної конфігурації в привілейованому режимі виконується команда *copy running-config startup-config*.

1.8 Основи забезпечення роботи локальних мереж

1.8.1 Виготовлення кабелю (пач-корд) для з'єднання обладнання локальної мережі

Path-cord – це невеликий відрізок кабелю «вита пара» довжиною від 1 до 10 м, на обох кінцях якого змонтований роз'єм RJ-45 (8P8C). Цей кабель утворює ділянку локальної мережі від гнізда мережного адаптера на комп'ютері до найближчої розетки або інтерфейсу мережевого пристрою. Для виготовлення

одного екземпляру Path-cord потрібні: відрізок кабелю, два роз'єми RJ-45 і обтискний інструмент для цих роз'ємів.

Обтискний інструмент

Обтискний інструмент (ключ) даного типу відрізняє наявність спеціального вирізу у формі роз'єму RJ-45 (8P8C) і оснащений ріжучою кромкою для рівної обрізки кабелю «вита пара» (рисунок 1.67). Як правило, обтискний ключ одночасно має можливість обробки роз'ємів типу RJ-11 (6P4C), що використовується в телефонних мережах.

Рисунок 1.67 – Обтискний ключ для роз'ємів RJ-45 (RJ-11)

Роз'єм RJ-45 (8P8C)

Роз'єми RJ-45 представляють собою порожнистий прозорий пластиковий корпус з фіксуючим замком, всередині якого розташовані вісім рухомих металевих контактів. У новому роз'ємі контакти виходять за межі корпусу, після обтиску вони вдавлюються всередину, прорізаючи зовнішній ізолюючий шар на дротах, розташованих усередині кабелю «вита пара», і замикаючись на дротову жилу (рисунок 1.68).

Порядок виконання монтажу

Для того щоб змонтувати роз'єм RJ-45 на кабель «вита пара», потрібно виконати наступні операції:

1. Видалити верхній захисний шар кабелю на відстань 0,5 дюйма (12,5 мм). Як правило, обтискний інструмент має спеціальну ріжучу кромку та обмежувач на цю відстань, що дозволяє точно виконати зазначену процедуру, не вивіряючи необхідний розмір лінійкою.

2. Акуратно розплести звиті пари дротів. Зачищати їх ізоляцію до дротової жили не потрібно.

3. Розташувати дроти «витої пари» в порядку, відповідному обраної схеми закладення кабелю. Всього для восьмижильного кабелю існує дві можливі схеми закладення: EIA/TIA-568A, E1A/T1A-568B (рисунок 1.69).

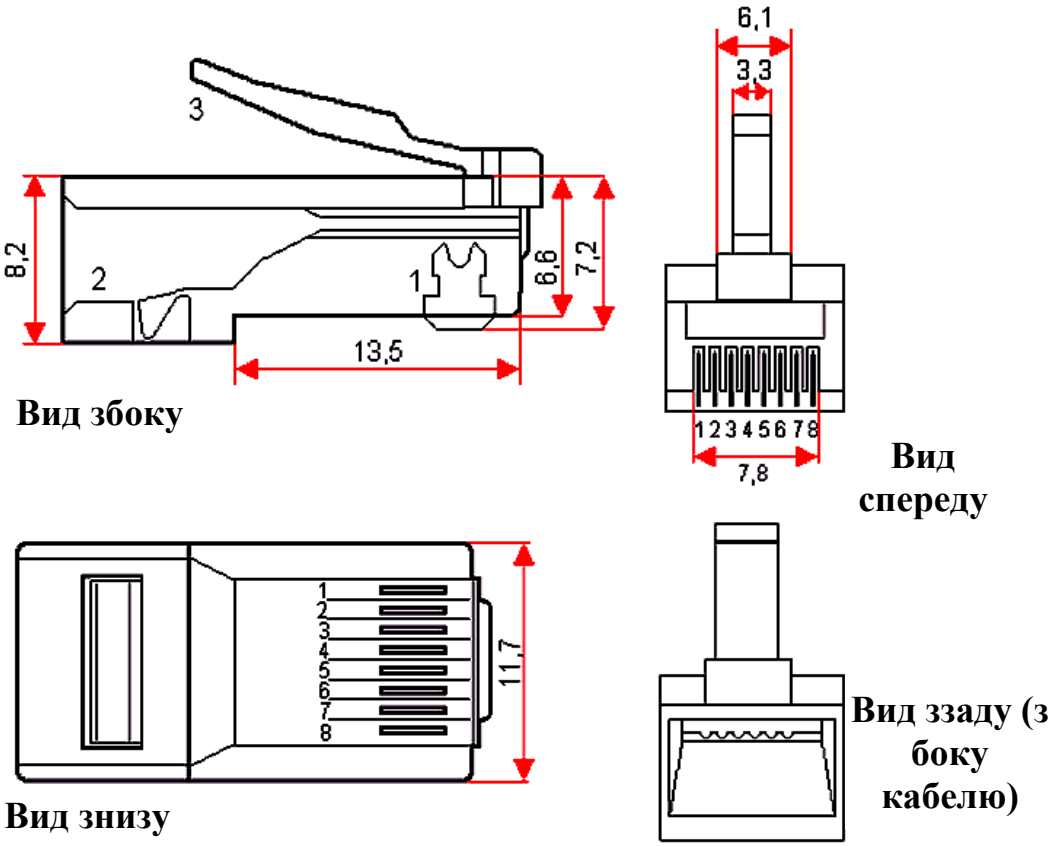


Рисунок 1.68 – Роз'єм RJ-45: 1 – контакти; 2 – тримач кабелю; 3 – замок роз'єму

Зазначені схеми в цілому ідентичні, проте слід розуміти, що на обох кінцях кабелю схема має бути однаковою коли потрібно виготовити «прямий» кабель для з'єднання комп'ютера до будь-якого пристрою (комутатор, концентратор і т.д.), або протилежною, коли виготовляють кабель типу «крос» (Cross-Over), що призначена для прямого з'єднання двох однакових пристрої. Вибір конкретного порядку розміщення дротів залежить від використовуваної у локальній мережі схеми закладення. Якщо мережа створюється заново, потрібно обрати будь-яку з цих двох схем і надалі дотримуватися саме її.

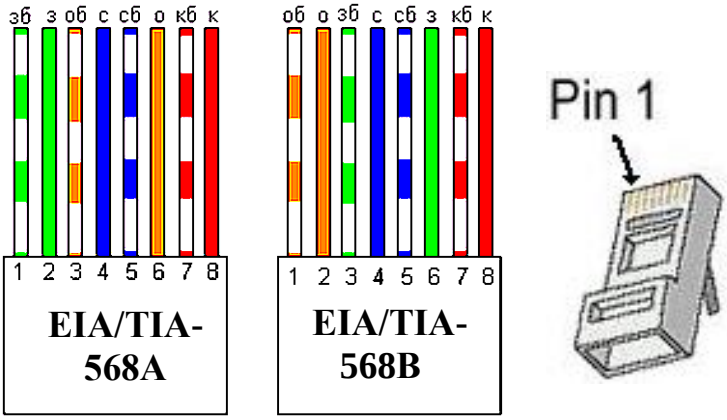


Рисунок 1.69 – Схеми закладення восьмижильного кабелю «вита пара» у роз'єм

4. Розташувавши дроти відповідним чином, необхідно взяти роз'єм RJ-45 контактами до себе, розмістивши тильною стороною до кабелю так, щоб кріплення замку було на протилежній від кабелю стороні роз'єму, і до упору насунути його на виступаючі з кабелю дроти (рисунок 1.70).

5. Вставити роз'єм з кабелем в отвір, розташований на робочій поверхні обтискного інструменту, і сильним, але плавним натисканням на ручки обтиснути кабель. При цьому контакти, що виступають з корпусу роз'єму, і тримач кабелю повинні повністю втопитися всередину роз'єму.

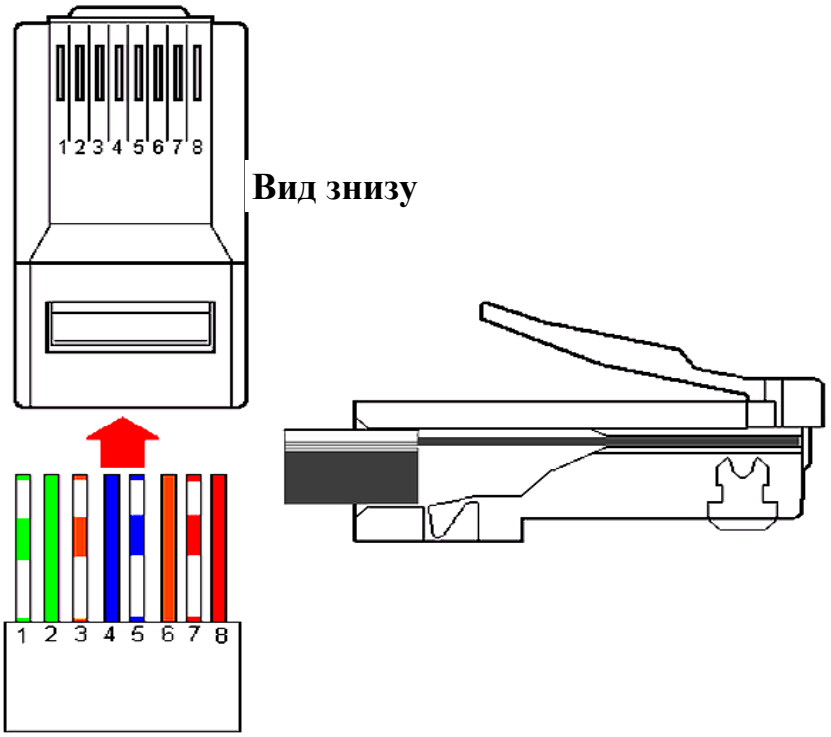


Рисунок 1.70 – Розташування кабелю «вита пара» в роз'ємі перед обтиском

1.8.2 Виконання з'єднання комп'ютерів у локальну мережу та налаштування стеку протоколів TCP/IP

Для організації роботи найпростішої локальної мережі необхідно з'єднати виходи мережевих карт комп'ютерів з входами комутатора та налаштувати IP-адреси комп'ютерів. Призначення IP-адрес здійснюється в меню: Панель управління/Мережа і Інтернет/Властивості протоколу Інтернет версії IPv4 локального підключення (рисунок 1.71). Перевірка налаштувань адресування забезпечується виконанням в командному рядку комп'ютера команди: **ipconfig /all**.

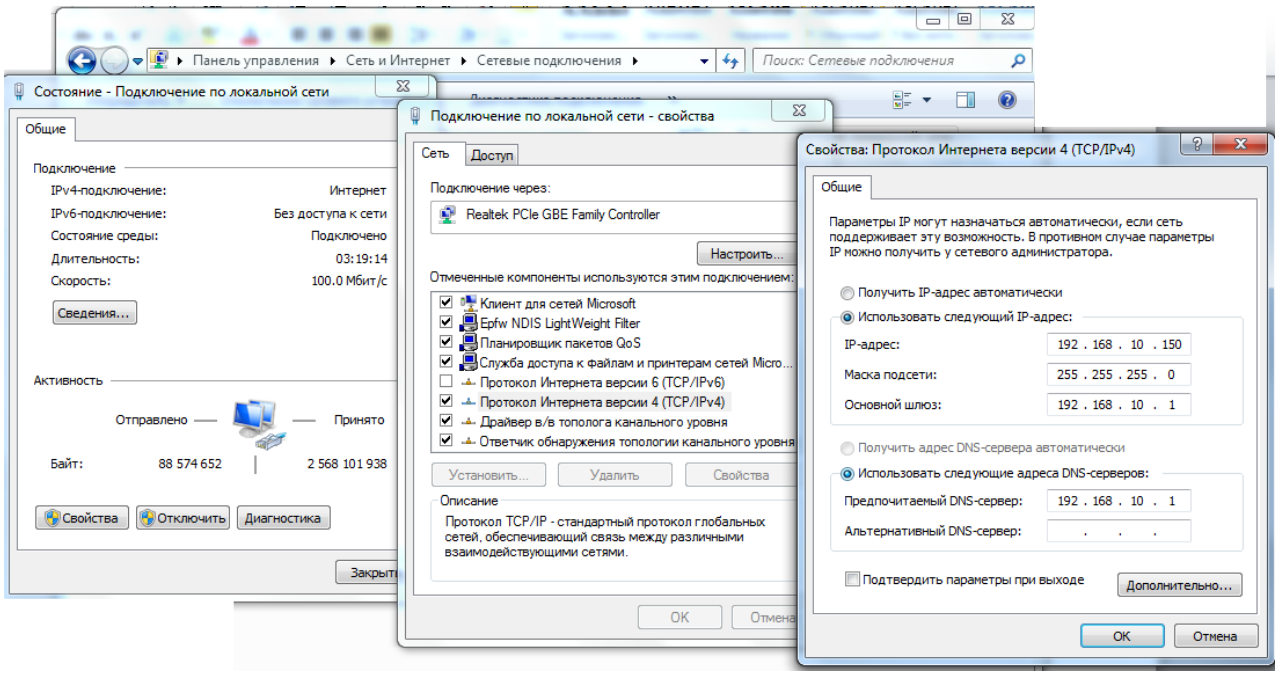


Рисунок 1.71 – Призначення IP-адреси комп'ютеру

Контрольні питання до розділу 1

1. Навести схему інформаційної мережі. Пояснити призначення складових.
2. Пояснити поняття фізичної структуризації локальних мереж (призначення, способи реалізації, обладнання)
3. Пояснити поняття логічної структуризації локальних мереж (призначення, способи реалізації, обладнання)
4. Перерахувати та стисло охарактеризувати варіанти середовищ передачі інформації для побудови локальних мереж.
5. Охарактеризувати та пояснити особливості технології сімейства Ethernet. Навести та пояснити формат кадру Ethernet.
6. Охарактеризувати та пояснити особливості технологій FDDI та Token Ring.
7. Охарактеризувати множинний доступ з опитуванням каналу і виявленням конфліктів - CSMA/CD. Особливості CSMA/CA.
8. Навести та описати рівні еталонної моделі ВВС (OSI). Пояснити розподіл мережевого обладнання за моделлю.
9. Описати порядок створення структури повідомлення за моделлю ВВС (OSI). Охарактеризувати поняття «протоколів» та «інтерфейсів» в моделі.
10. Пояснити призначення, можливості та описати режими роботи операційної системи IOS.
11. Перелічити та пояснити зміст задач та функцій управління інформаційно-комунікаційними мережами. Охарактеризувати основні протоколи управління мережами.
12. Перелічити та охарактеризувати типи мобільності, що має підтримувати мережа IP-телефонії.

13. Проаналізувати та порівняти різні методи кодування мовної інформації в IP-телефонії.
14. Навести особливості (переваги) використання IP-телефонії. Охарактеризувати відмінності трьох різних термінів для позначення технології передачі мови по мережах з пакетною комутацією на базі протоколу IP.
15. Охарактеризувати види з’єднань у мережі IP-телефонії.
16. Навести та проаналізувати перелік основних засобів захисту інформації в мережах
17. Навести та проаналізувати перелік основних загроз безпеці інформації в мережах

РОЗДІЛ 2
КЕРУВАННЯ В ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ МЕРЕЖАХ

2.1 Принципи керування в інформаційно-комунікаційних мережах

2.1.1 Задачі систем управління мережами

Система мережного управління виконує завдання з надання адміністратору інформації про структуру мережі з усіма зв'язками та інформацію про стан керованих об'єктів (кінцевих вузлів, ліній зв'язку, інтерфейсів комунікаційної апаратури). Для рішення цієї задачі існують спеціалізовані пакети керуючого програмного забезпечення (ПЗ), як правило, орієнтовані на роботу з обладнанням одного виробника.

Це, наприклад, Transcend для обладнання 3Com, Optivity для Bay Networks, HP Open View для Hewelett-Packard, Spectrum для Cabletron, Nways Manager для IBM, LanDesk (управління робочими станціями) для Intel. Часто задачі управління вирішують в обмеженому обсязі, охоплюючи тільки комунікаційне обладнання, і не завжди в повному обсязі. Скорочення обсягу звичайно обумовлюється досить високою вартістю засобів управління.

Задачі мережного управління:

– *Вимір параметрів*, що визначають пропускну здатність, час відгуку, завантаження ліній і т.п. Система стежить за значеннями параметрів, визначає їхні середні (нормальні) значення, а по досягненні якими-небудь критичними параметрами заданих порогів генерує попередження чи автоматично виконує керуючі дії. Крім спостереження за мережею, у цій області управління можливо й активний вплив на мережу за допомогою симуляторів (генераторів трафіку). Це допоможе оцінити поводження мережі при підвищенні навантаження (наприклад, при плануванні її розширення) і почати необхідні заходи для забезпечення прийнятної продуктивності.

– *Відстеження інформації* про апаратні та програмні складові елементів вузлів мережі, включаючи дані про тип і версію продуктів, їх налаштування і т.п. Сюди ж відноситься управління функціями пристроїв у цілому і конфігурування їхніх інтерфейсів.

– *Облік трафіку* окремих вузлів і їхніх груп як для ефективного розподілу ресурсів мережі, так і з метою визначення розміру оплати за надані послуги.

– *Виявлення і реєстрація відмов* для повідомлення адміністратора і відновлення працездатності. З появою симптомів відмови, система повинна по можливості ізолювати проблемну ділянку, задіяти резервні елементи, запротоколювати події, відновити вихідну конфігурацію після усунення відмови.

– *Управління доступом* до мережних ресурсів: авторизація користувачів, захист від несанкціонованого доступу, запобігання блокування мережі (навмисного і ненавмисного).

Для виконання цих завдань система управління виконує наступні ***функції***:

– керування конфігурацією;
– керування якістю роботи;
– керування усуненням несправностей;
– керування розрахунками;
– керування безпекою.

Керування конфігурацією передбачає:

– збір, ведення й відображення інформації про мережні елементи (їх типи, місцезнаходження, ідентифікатори й т.п.);
– уведення елементів в експлуатацію і їх вивід з експлуатації;
– установлення й зміна логічних з'єднань між елементами.

Керування якістю роботи включає:

– контроль і підтримку на необхідному рівні основних характеристик мережі;
– збір, обробку, реєстрацію, зберігання й відображення статистичних даних про роботу мережі і її елементів;
– виявлення тенденцій у їхній поведінці й попередження про можливі порушення в роботі.

Керування усуненням несправностей забезпечує:

– виявлення й локалізацію несправностей у мережі;
– реєстрацію несправностей і доведення відповідної до інформації до обслуговуючого персоналу;
– видачу рекомендацій з усунення несправностей.

Керування розрахунками:

– здійснює контроль над ступенем використання мережних ресурсів;
– підтримує функції по нарахуванню оплати за це використання.

Керування безпекою забезпечує:

– захист мережі від несанкціонованого доступу;
– обмеження доступу за допомогою паролів;
– видачу сигналів тривоги при спробах несанкціонованого доступу;
– відключення небажаних користувачів;
– криптографічний захист інформації керування.

Принципи побудови системи управління

Відносно систем управління мережами найбільш розробленим і ефективним для створення багаторівневої ієрархічної системи є стандарт Telecomunication Management Network (TMN), розроблений сумісно ITU-T (міжнародний союз електрозв'язку), ISO, ANSI і ETSI.

Незважаючи на те, що цей стандарт початково призначався для телекомунікаційних мереж (мереж зв'язку), орієнтація на використання загальних принципів зробила його корисним для побудови будь-якої інтегрованої системи управління мережами.

Стандарт TMN складається з великої кількості рекомендацій ITU-T і інших організацій стандартизації, а основні принципи моделі TMN приведені в рекомендації M.3010.

Модель TMN має 4 рівні:

1. Нижній рівень – **управління елементами мережі** (Network Element Layer – NEL) складається з окремих пристроїв мережі: маршрутизаторів, підсилювачів, концентраторів, мультиплексорів, комутаторів, кінцевого обладнання і інших. Ці елементи мають вбудовані засоби для підтримки управління: датчики, інтерфейси управління. На сьогодні є пристрої, які крім загального стану контролюють ще й трафік, що проходить через елемент мережі. Подібні пристрої використовуються в технології FDDI, ISDN, Frame Relay, SDH.

Ці протоколи не маючи спеціального блоку управління зобов'язують пристрій підтримувати функції управління. Якщо ж немає вбудованої функції управління тоді пристрої мережі мають окремі блоки управління, що підтримують один із двох найпоширеніших протоколів управління – SNMP (Simple Network Management Protocol) чи CMIP (Common Management Information Protocol) – інформаційний протокол загального управління.

Ці протоколи відносяться до прикладного рівня (7-го) моделі взаємодії відкритих систем (OSI).

2. **Рівень управління мережею** (Network Management Layer) – координує роботу елементарних систем управління елементами мережі. Дозволяє:

- контролювати конфігурацію каналів зв'язку,
- забезпечувати адресування та маршрутизацію,
- узгоджувати роботу транспортних підмереж різних технологій і т.д.

3. **Рівень управління послугами** (Servise Management Layer) – контроль і управління транспортними послугами і інформаційними послугами, що надаються користувачам. А також контроль результатів якості обслуговування (швидкість передачі в Frame Relay, пульсація пакетів і т.і.).

4. **Рівень адміністративного управління** (Buisiness Management Layer) – питання довгострокового планування мереж зі врахуванням фінансового і організаційного аспектів діяльності мережі.

Архітектура системи управління мережею зв'язку

Розподілена природа мереж обумовлює застосування моделі «менеджер-агент» для побудови системи управління (рисунок 2.1). *Менеджер* представляє собою програмно-апаратний засіб, що збирає інформацію від агентів і виконує її обробку для надання адміністратору мережі. На підставі цієї інформації адміністратор за допомогою менеджера може здійснювати деякі керуючі впливи на об'єкти мережі. Управління може бути тією чи іншою мірою автоматизовано.

Агенти розташовуються в керуємих елементах мережі. Вони безпосередньо взаємодіють з керованими об'єктами й обслуговують базу даних керованих параметрів (за якими ведеться спостереження). *Інформаційні бази управління* MIB (Management Information Base) містять списки керованих параметрів і їхнього значення, агенти відповідають за відповідність баз реальним станам об'єктів. Менеджер може в будь-який момент запросити інформацію про стан об'єкта, виконуючи операцію читання й агент у відповідь на цей запит зобов'язаний передати уміст усієї бази чи її частини. Операція запису в базу, якщо вона дозволена, змушує агента виконувати керуючі впливи на об'єкт. Опитування стану виконуються з ініціативи менеджера регулярно (полінг) чи епізодично, наприклад, по запиту адміністратора. Агенту можуть надаватися можливості й асинхронного повідомлення менеджера про настання яких-небудь подій. Для цього використовуються спеціальні повідомлення, названі *перериваннями* чи *тривогами* (trap, alert), що посилаються на адресу менеджера.

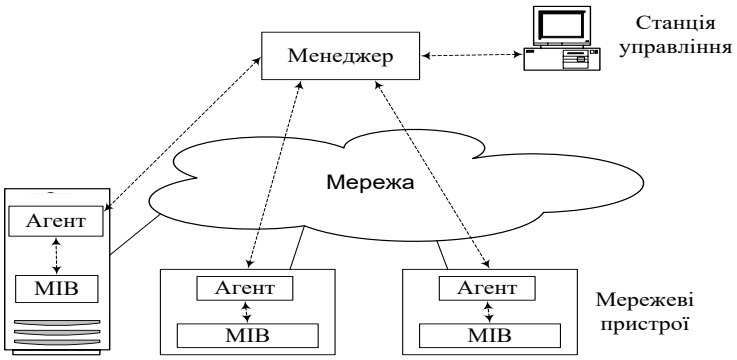


Рисунок 2.1 – Структура систем управління мережею

2.1.2 Протокол TFTP. Збереження та відновлення конфігураційних файлів та операційної системи на TFTP-сервері

Для збереження та відновлення конфігураційних файлів, а при необхідності – образів операційної системи, використовується протокол TFTP (Trivial File Transfer Protocol – спрощений протокол передачі файлів). Він є дуже простим і підтримується будь-яким обладнанням, в тому числі програмою початкового завантаження обладнання Cisco. Обмін відбувається з мережним вузлом, на якому встановлено програмне забезпечення TFTP-сервера.

TFTP – це простий протокол, розроблений таким чином, щоб поміщатися в постійному запам'ятовуючому пристрої і бути використаним тільки в процесі завантаження бездискових систем. Він використовує невелику кількість форматів повідомлень (п'ять) і протокол із зупинкою і очікуванням підтвердження (stop-and-wait).

Обмін між клієнтом і сервером починається з того, що клієнт запитує сервер або прочитати, або записати файл для клієнта. У стандартному варіанті завантаження бездискової системи перший запит – це запит на читання (RRQ). На рисунку 2.2 показаний формат п'яти повідомлень TFTP (коди операцій 1 і 2 мають однаковий формат).

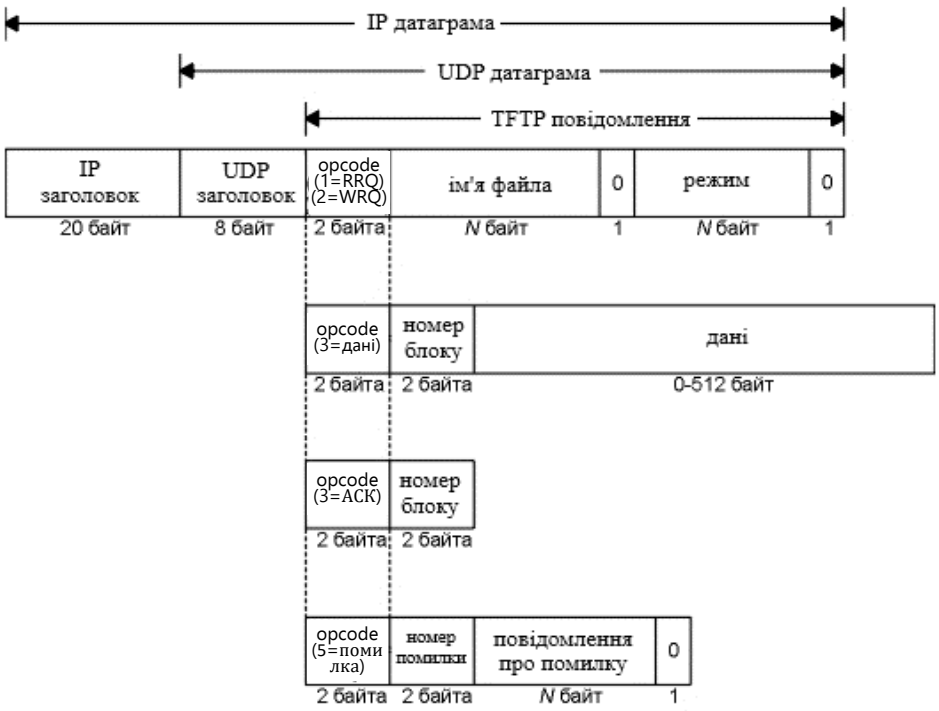


Рисунок 2.2 – Формат п’яти TFTP повідомлень (opcode – код операції)

У TFTP існує 2 режими передачі:

– **netascii** – файл перед передачею перекодовується в ASCII.

– **octet** – файл передається без змін.

Кожен пакет даних містить номер блоку (block number), який потім використовується в пакеті підтвердження. Як приклад скажімо, що коли необхідно здійснити читання файлу, клієнт посилає запит на читання (RRQ), вказуючи ім’я файлу і режим. Якщо файл може бути прочитаний клієнтом, сервер відповідає пакетом даних з номером блоку рівним 1. Клієнт посилає підтвердження (ACK) на номер блоку 1. Сервер відповідає наступним пакетом даних з номером блоку рівним 2. Клієнт підтверджує номер блоку 2. Це триває доти , поки файл не буде переданий. Кожен пакет даних містить 512 байт даних, за винятком останнього пакета, який містить від 0 до 511 байт даних. Коли клієнт отримує пакет даних, який містить менше ніж 512 байт, він вважає, що отримав останній пакет.

У разі запиту на запис (WRQ) клієнт посилає WRQ, вказуючи ім’я файлу і режим. Якщо файл може бути записаний клієнтом, сервер відповідає підтвердженням (ACK) з номером блоку рівним 0. Клієнт посилає перші 512 байт файлу з номером блоку рівним 1, сервер відповідає ACK з номером блоку рівним 1. Цей тип передачі даних і називається протоколом із зупинкою і очікуванням підтвердження, він використовується тільки в простих протоколах, таких як TFTP.

Останній тип TFTP повідомлень це повідомлення про помилки, код операції (opcode) дорівнює 5. Це якраз те, чим сервер відповідає в тому випадку, якщо запит на читання або запис не може бути оброблений. Помилки читання

або запису в перебігу передачі файлу також призводять до того, що відправляється повідомлення про помилку, при цьому передача припиняється. Номер помилки (error number) містить цифровий код помилки, за яким слід повідомлення про помилку в ASCII форматі, яке може містити додаткову інформацію надану операційною системою.

Так як TFTP використовує ненадійний UDP, то саме від TFTP залежить, як будуть опрацьовані втрачені і дубльовані пакети. У разі втрати пакету, відправник відпрацьовує тайм-аут і здійснює повторну передачу.

TFTP пакети не містять ніяких даних про ім'я користувача або пароль. Це пролом в секретності характерна для TFTP. Так як TFTP був розроблений для використання в процесі завантаження, він не надає можливості передати ім'я користувача і пароль.

Ця характеристика TFTP була використана багатьма хакерами, щоб отримати копії файлу паролів з Unix і потім розшифрувати паролі. Щоб запобігти подібному доступу, більшість TFTP серверів в даний час регламентують, які файли можуть бути отримані з використанням TFTP (як правило, файли з директорії /tftpboot в Unix системах). Ця директорія містить тільки завантажувальні файли, необхідні бездисковим системам.

Щоб дозволити кільком клієнтам завантажуватися одночасно, TFTP сервер надає кілька форм одночасної роботи. Так як UDP не надає унікального з'єднання між клієнтом і сервером (як це робить TCP), TFTP сервер створює новий UDP порт для кожного клієнта. Це дозволяє різним клієнтам видавати датаграми, які будуть демультіплексіровать UDP модулем сервера, на основі номерів портів призначення, замість того щоб це робив сам сервер.

Використання TFTP для резервного копіювання конфігурації

Копії файлів конфігурації можуть зберігатися як резервні файли на випадок виникнення проблем. Файли конфігурації можуть зберігатися на сервері TFTP або на USB-накопичувачі. Файл конфігурації також повинен бути включений в мережеву документацію.

Щоб зберегти поточну конфігурацію або конфігурацію запуску на TFTP-сервері, використовують команду **copy running-config tftp** або **copy startup-config tftp** в послідовності, яка показана на рисунку 2.3.

```
R1# copy running-config tftp
Remote host []?192.168.10.254
Name of the configuration file to write[R1-config]? R1-Jan-2019
Write file R1-Jan-2019 to 192.168.10.254? [confirm]
Writing R1-Jan-2019 !!!!!! [OK]
```

Рисунок 2.3 – Використання команди **copy running-config tftp**

Дії для резервного копіювання працює конфігурації на TFTP-сервер:

Крок 1. Ввести команду **copy running-config tftp**.

Крок 2. Ввести IP-адресу хоста, на якому буде зберігатися файл конфігурації.

Крок 3. Ввести ім’я, яке потрібно присвоїти файлу конфігурації.

Крок 4. Натиснути Enter, щоб підтвердити кожен вибір.

Щоб відновити поточну або початкову конфігурацію з TFTP-сервера, використовують команду **copy tftp running-config** або **copy tftp startup-config**. Для відновлення конфігурації з TFTP-сервера необхідно:

Крок 1. Ввести команду **copy tftp running-config**.

Крок 2. Ввести IP-адресу хоста, на якому зберігається файл конфігурації.

Крок 3. Ввести ім’я, яке потрібно присвоїти файлу конфігурації.

Крок 4. Натиснути Enter, щоб підтвердити кожен вибір.

Деякі моделі маршрутизаторів Cisco підтримують USB-накопичувачі. Функція USB-флеш-пам’яті забезпечує можливість додаткового зберігання і додатковий завантажувальний пристрій. Образи Cisco IOS, файли конфігурації та інші файли можна копіювати на флеш-пам’ять Cisco USB або з неї з тієї ж надійністю, що і при зберіганні та отриманні файлів з використанням карти Compact Flash. Крім того, модульні інтегровані сервісні маршрутизатори можуть завантажувати будь-який образ програмного забезпечення Cisco IOS, збережений на флеш-пам’яті USB. В ідеалі USB-накопичувач може зберігати кілька копій Cisco IOS і кілька конфігурацій маршрутизатора.

Команду **dir** використовують для перегляду змісту флеш-накопичувача USB, як показано на рисунку 2.4.

```
Router# dir usbflash0:
Directory of usbflash0:/
1 -rw- 30125020 Dec 22 2032 05:31:32 +00:00 c3825-entservicesk9-mz.123-14.T
63158272 bytes total (33033216 bytes free)
```

Рисунок 2.4 – Використання команди **dir** для перегляду змісту флеш-накопичувача USB

Під час резервного копіювання на USB-порт рекомендується виконати команду **show file systems**, щоб переконатися в наявності USB-накопичувача і підтвердити його ім’я. В останньому рядку виведення відображається порт USB і ім’я: «usbflash0:».

Команда **copy running-config usbflash0:/** застосовується щоб скопіювати файл конфігурації на USB-накопичувач. Обов’язково використовувати ім’я флешки, як зазначено в файлової системі. Коса риска є необов’язковою, але вказує на кореневої каталог флеш-накопичувача USB. IOS запросить ім’я файлу. Якщо файл вже існує на USB-накопичувачі, маршрутизатор запропонує перезаписати, можливо і копіювання на флеш-накопичувач USB без раніше створеного файлу, із запитом імені.

Команду **dir** використовують щоб переглянути файл на USB-накопичувачі, і команду more, щоб переглянути вміст, як показано на рисунку 2.5.

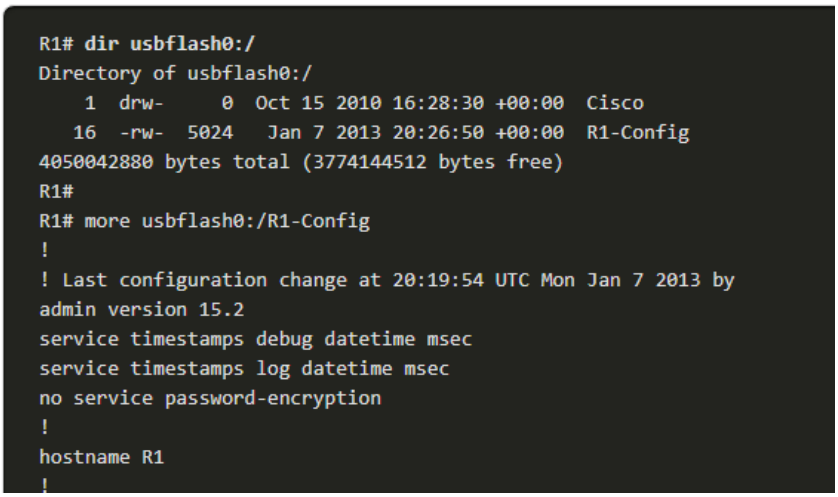


Рисунок 2.5 – Використання команди **dir** для перегляду файлів на USB-накопичувачі

Щоб скопіювати файл з іменем R1-Config назад і, наприклад, відновити робочу конфігурацію, треба використати команду **copy usbflash0:/ R1-Config running-config**. Також існує можливість відредагувати збережений на USB-накопичувачі файл за допомогою текстового редактора.

Процедури відновлення пароля

Паролі на пристроях використовуються для запобігання несанкціонованому доступу. Для зашифрованих паролів, таких як включення секретних паролів, паролі повинні бути замінені після відновлення. Детальна процедура відновлення пароля залежить від пристрою. Однак всі процедури відновлення пароля засновані на тому ж принципі:

Крок 1. Увійти в режим ROMMON.

Крок 2. Змінити регістр конфігурації.

Крок 3. Скопіювати початковий конфігураційний файл в поточний.

Крок 4. Змінити пароль.

Крок 5. Зберегти running-config як новий startup-config.

Крок 6. Перезавантажити пристрій.

Для відновлення пароля потрібно консольний доступ до пристрою через програмне забезпечення терміналу або емулятора терміналу на ПК. Налаштування терміналу для доступу до пристрою: 9600 біт, без перевірки парності, 8 біт даних, 1 стоповий біт, немає управління потоком.

Крок 1. Увійти в режим ROMMON.

За допомогою доступу до консолі користувач може отримати доступ до режиму ROMMON, використовуючи комбінацію переривання під час процесу завантаження або витягуючи зовнішню флеш-пам’ять, коли пристрій вимкнений. У разі успіху з’явиться запрошення rommon 1>, як показано на рисунку 2.6.

Комбінація переривання для PuTTY: Ctrl + Break. Список стандартних комбінацій клавіш переривання для інших емуляторів терміналу і операційних систем можна знайти в Інтернеті.

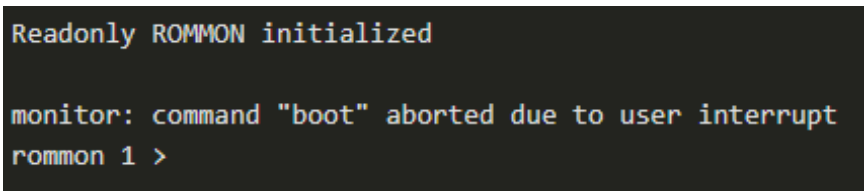


Рисунок 2.6 – Варіант доступу до режиму ROMMON

Крок 2. Змінити регістр конфігурації.

Програмне забезпечення ROMMON підтримує деякі основні команди, такі як **confreg**. Команда confreg 0x2142 дозволяє користувачеві встановити регістр конфігурації на 0x2142 (рисунок 2.7). З регістром конфігурації 0x2142 пристрій ігноруватиме файл конфігурації запуску при запуску. Файл конфігурації запуску, де знаходяться невідомі паролі. Щоб перезавантажити пристрій після установки регістра конфігурації в 0x2142, необхідно ввести в командному рядку **reset**.

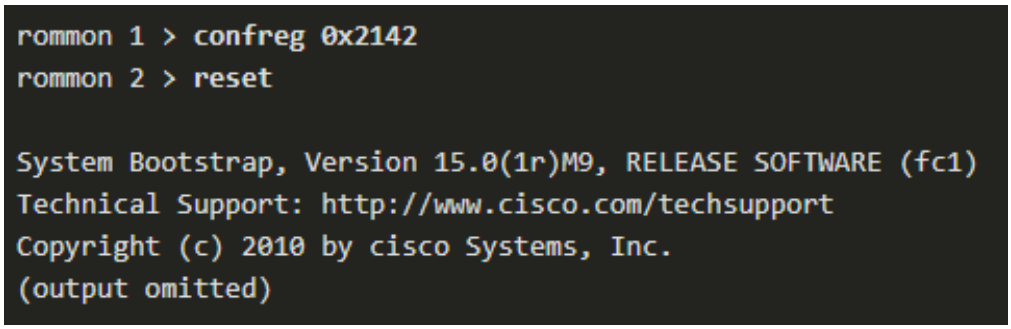


Рисунок 2.7 – Використання команди confreg 0x2142

Крок 3. Скопіювати стартову конфігурацію startup-config в поточну конфігурацію running-config.

Якщо потрібно зберегти налаштування, які прописані у стартовій конфігурації, то після завершення перезавантаження пристрою треба скопіювати початкову конфігурацію в поточну конфігурацію за допомогою команди **copy startup-config running-config**, як показано на рисунку 2.8. При цьому запрошення маршрутизатора може змінитися на ім’я, яке записано в стартовій конфігурації (у прикладі на R1#).

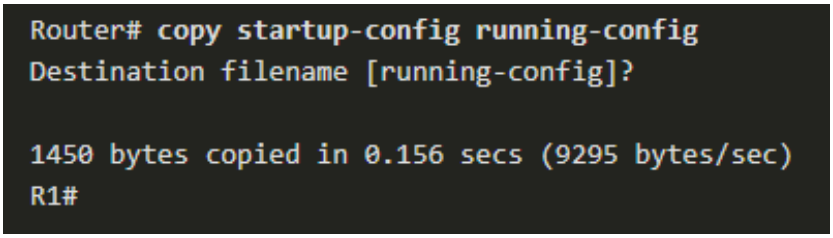


Рисунок 2.8 – Використання команди **copy startup-config running-config**

Якщо потреби у збереженні стартових налаштувань не існує, то можна виконати команду **copy running-config startup-config**, яка замініть вихідну конфігурацію запуску на порожню.

Крок 4. Змінити пароль.

Перебуваючи в привілейованому режимі EXEC, можна налаштувати всі необхідні паролі, як показано на рисунку 2.9.

```
R1# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)# enable secret cisco
```

Рисунок 2.9 – Варіант налаштування нового паролю

Крок 5. Збереження running-config як нового startup-config.

Після настройки нових паролів треба змінити регістр конфігурації назад на 0x2102 за допомогою команди **config-register 0x2102** в режимі глобальної конфігурації і зберегти running-config в startup-config, як показано на рисунку 2.10.

```
R1(config)# config-register 0x2102
R1(config)# end
R1# copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
R1#
```

Рисунок 2.10 – Використання команди **config-register**

Крок 6. Перезавантаження пристрою.

Тепер пристрій використовуватиме нові налаштовані паролі для аутентифікації і його можна перезавантажити командою **reload**. Бажано попередньо використати команди **show startup-config**, щоб переконатися, що всі зміни збережені коректно.

TFTP-сервери як резервне сховище

У міру зростання мережі образи програмного забезпечення Cisco IOS і файли конфігурації можуть зберігатися на центральному сервері TFTP, як показано на рисунку 2.11. Це допомагає контролювати кількість образів IOS і версій цих образів IOS, а також файлів конфігурації, які необхідно підтримувати.

Рисунок 2.11 – Зберігання файлів конфігурації на центральному сервері TFTP

Корпоративні мережі зазвичай охоплюють широкі області та містять кілька маршрутизаторів. Для таких мереж рекомендується зберігати резервну

копію образу програмного забезпечення Cisco IOS на випадок, якщо образ системи на маршрутизаторі буде пошкоджений або випадково видалений.

Використання мережевого TFTP-сервера дозволяє завантажувати образ або його конфігурацію по мережі. Мережевим TFTP-сервером може бути інший маршрутизатор, робоча станція або хост-система.

Щоб забезпечити мінімальний час простою мережі, необхідно мати процедури для резервного копіювання образів Cisco IOS. Це дозволяє адміністратору швидко скопіювати зображення назад на маршрутизатор в разі пошкодження або стирання образу.

Розглянемо випадок коли адміністратор мережі хоче створити резервну копію поточного файлу образу маршрутизатора (isr4200-universalk9_ias.16.09.04.SPA.bin) на сервері TFTP за адресою 172.16.1.100.

Крок 1. Перевірка зв'язку з TFTP-сервером.

Переконатися, що є доступ до мережевого TFTP-сервера. Виконати ping TFTP-серверу для перевірки підключення.

Крок 2. Перевірка розміру образу у флеш-пам'яті.

Переконатися, що на сервері TFTP достатньо дискового простору для розміщення образу програмного забезпечення Cisco IOS. Використати команду **show flash0:** на маршрутизаторі, щоб визначити розмір файлу образу Cisco IOS.

Крок 3. Копіювання образу на TFTP-сервер.

Скопіювати образ на TFTP-сервер за допомогою команди **copy flash: tftp:**. Після введення команди адміністратору пропонується ввести ім'я вихідного файлу, IP-адресу віддаленого хоста і ім'я кінцевого файлу. Якщо варіант назви (адреси), що пропонується операційною системою, влаштовує адміністратора, то досить натиснути Enter, щоб його прийняти. Передача почнеться (рисунок 2.12) і її процес відображатиметься послідовними знаками оклику.

```
R1# copy flash: tftp:
Source filename []? isr4200-universalk9_ias.16.09.04.SPA.bin
Address or name of remote host []? 172.16.1.100
Destination filename [isr4200-universalk9_ias.16.09.04.SPA.bin]?
Writing isr4200-universalk9_ias.16.09.04.SPA.bin...
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
(output omitted)
517153193 bytes copied in 863.468 secs (269058 bytes/sec)
```

Рисунок 2.12 – Використання команди **copy source-url destination-url**

Команда завантаження системи

Щоб визначити маршрутизатору необхідність використання для завантаження нового образу IOS, який був збережений у його флеш-пам'яті (особливо, якщо файл попереднього образ також залишився у флеш-пам'яті), необхідно налаштувати маршрутизатор на використання нового способу завантаження за допомогою команди **boot system**, як показано на рисунку 2.13. Далі треба зберегти конфігурацію і перезавантажити маршрутизатор – він має завантажитися з новим образом.

```
R1# configure terminal
R1(config)# boot system flash0:isr4200-universalk9_ias.16.09.04.SPA.bin
R1(config)# exit
R1# copy running-config startup-config
R1# reload
```

Рисунок 2.13 – Використання команди **boot system**

Під час запуску програма початкового завантаження аналізує файл конфігурації запуску в NVRAM, відшуковує команди завантажувальної системи, які вказують ім’я та розташування образу програмного забезпечення Cisco IOS для завантаження. Якщо кілька команд системи завантаження введені послідовно, то це забезпечує кілька різних варіантів завантаження, що будуть реалізовані за неможливості попереднього, для підвищення відмовостійкості мережі.

Якщо в конфігурації немає команд завантаження системи, то маршрутизатор за замовчуванням знаходить перший дійсний образ Cisco IOS у флеш-пам’яті і запускає його. Після завантаження маршрутизатора, щоб переконатися, що новий образ завантажений, можна використати команду **show version**.

2.1.3 Управління файловою системою Cisco

Файлова система Cisco IOS (IFS) дозволяє адміністратору переміщатися по різних каталогах (директоріях) і перебирати файли в каталозі. Адміністратор також може створювати підкаталоги у флеш-пам’яті або на диску. Доступність каталогу залежить від типів пристрою та пам’яті.

Команда **show file systems** надає інформацію (рисунок 2.14), таку як обсяг загальної і вільної пам’яті, тип файлової системи та її дозволи. Дозволи включають тільки читання (ro), тільки запис (wo) або читання і запис (rw). Права доступу відображаються в стовпці «Прапори» вихідних даних команди.

Якщо певна файлова системи (на рисунку це флеш-пам’ять – flash) позначена зірочкою, то це вказує, що вона є поточною файловою системою за замовчуванням. Завантажувальний образ IOS знаходиться у флеш-пам’яті; тому символ (bootflash) додається до списку флеш-пам’яті, вказуючи, що це завантажувальний диск.

Вміст поточної файлової системи відображається командою **dir (directory)**. На рисунку 2.15 показаний результат виконання цієї команди, оскільки flash є поточною файлової системою за замовчуванням, команда dir виводить вміст саме flash-пам’яті.

Кілька файлів відображаються у переліку флеш-пам’яті, останній з наведеного списку – це файл образу Cisco IOS, з якого виконане завантаження системи.

```
Router# show file systems
File Systems:
      Size(b)       Free(b)      Type  Flags  Prefixes
            -             -    opaque     rw   system:
            -             -    opaque     rw   tmpsys:
*   7194652672    6294822912      disk     rw   bootflash: flash:
    256589824     256573440      disk     rw   usb0:
   1804468224    1723789312      disk     ro   webui:
            -             -    opaque     rw   null:
            -             -    opaque     ro   tar:
            -             -   network     rw   tftp:
            -             -    opaque     wo   syslog:
     33554432      33539983     nvram     rw   nvram:
            -             -   network     rw   rcp:
            -             -   network     rw   ftp:
            -             -   network     rw   http:
            -             -   network     rw   scp:
            -             -   network     rw   sftp:
            -             -   network     rw   https:
            -             -    opaque     ro   cns:
Router#
```

Рисунок 2.14 – Використання команди **show file systems**

```
Router# dir
Directory of bootflash:/
   11  drwx           16384   Aug 2 2019 04:15:13 +00:00  lost+found
370945  drwx            4096   Oct 3 2019 15:12:10 +00:00  .installer
338689  drwx            4096   Aug 2 2019 04:15:55 +00:00  .ssh
217729  drwx            4096   Aug 2 2019 04:17:59 +00:00  core
379009  drwx            4096  Sep 26 2019 15:54:10 +00:00  .prst_sync
80641  drwx             4096   Aug 2 2019 04:16:09 +00:00  .rollback_timer
161281  drwx            4096   Aug 2 2019 04:16:11 +00:00  gs_script
112897  drwx          102400   Oct 3 2019 15:23:07 +00:00  tracelogs
362881  drwx            4096  Aug 23 2019 17:19:54 +00:00  .dbpersist
298369  drwx            4096   Aug 2 2019 04:16:41 +00:00  virtual-instance
   12  -rw-              30   Oct 3 2019 15:14:11 +00:00  throughput_monitor_params
 8065  drwx            4096   Aug 2 2019 04:17:55 +00:00  onep
   13  -rw-              34   Oct 3 2019 15:19:30 +00:00  pnp-tech-time
249985  drwx            4096  Aug 20 2019 17:40:11 +00:00  Archives
   14  -rw-           65037   Oct 3 2019 15:19:42 +00:00  pnp-tech-discovery-summary
   17  -rw-         5032908  Sep 19 2019 14:16:23 +00:00  isr4200_4300_rommon_1612_1r_SPA.pkg
   18  -rw-       517153193  Sep 21 2019 04:24:04 +00:00  isr4200-universalk9_ias.16.09.04.SPA.bin
7194652672 bytes total (6294822912 bytes free)
Router#
```

Рисунок 2.15 – Відображення файлової системи flash
за допомогою команди **dir**

Щоб переглянути вміст NVRAM, потрібно змінити поточну файлову систему за замовчуванням за допомогою команди **cd** (змінити каталог), як показано на рисунку 2.16. Команда робочого каталогу – **pwd** перевіряє, що йде перегляд саме каталогу NVRAM. Далі командою **dir** можна вивести вміст NVRAM: в списку кілька файлів конфігурації, з них саме файл startup-config представляє конфігурацію запуску.

Конфігурацію можна формувати безпосередньо в пристрої, а можна скопіювати з файлу, а потім додати в пристрій. При завантаженні IOS виконує кожен рядок тексту конфігурації у вигляді команди. Файл конфігурації може попередньо редагуватися в текстовому редакторі. Наприклад, щоб представити паролі в незашифрованому вигляді, а текст, який не є командою – вилучити. Перед вставкою конфігурації необхідно увійти в режим глобальної конфігурації.

```
Router#
Router# cd nvram:
Router# pwd
nvram:/
Router# dir
Directory of nvram:/
32769  -rw-            1024                    startup-config
32770  ----              61                    private-config
32771  -rw-            1024                    underlying-config
    1  ----               4                    private-KS1
    2  -rw-            2945                    cwmp_inventory
    5  ----             447                    persistent-data
    6  -rw-            1237                    ISR4221-2x1GE_0_0_0
    8  -rw-              17                    ecfm_ieee_mib
    9  -rw-               0                    ifIndex-table
   10  -rw-            1431                    NIM-2T_0_1_0
   12  -rw-             820                    IOS-Self-Sig#1.cer
   13  -rw-             820                    IOS-Self-Sig#2.cer
33554432 bytes total (33539983 bytes free)
Router#
```

Рисунок 2.16 – Відображення вмісту NVRAM за допомогою команди **cd**

2.1.4 Версії та завантаження операційної системи Cisco IOS

В основі більшості технологій корпорації Cisco лежить спеціалізована операційна система – міжмережева операційна система корпорації Cisco IOS (Internetwork Operating System), яка являє собою специфічне програмне забезпечення для керування функціями маршрутизації та комутації.

Маршрутизатор має доступ до енергозалежної або незалежній пам'яті. Енергозалежною пам'яті для збереження даних потрібне постійне живлення. При виключенні електроживлення маршрутизатора або при його перезапуску вміст цієї пам'яті втрачається. Незалежна пам'ять зберігає дані навіть при перезавантаженні пристрою. У зв'язку з цим в маршрутизаторі Cisco використовується чотири типи пам'яті (рисунок 2.17).

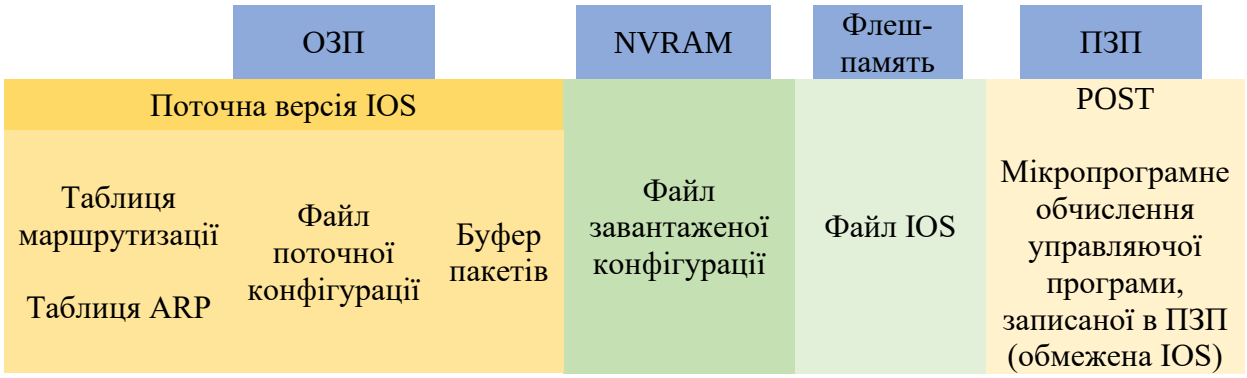


Рисунок 2.17 – Типи пам'яті пристроїв Cisco

Оперативний запам'ятовуючий пристрій (ОЗП, RAM). Це енергозалежна пам'ять, яка використовується в обладнанні Cisco для зберігання додатків, процесів і даних, щопотребують обробки центральним процесором. Пристрої

Cisco використовують швидкий тип ОЗП, який називають синхронним динамічним ОЗП (SDRAM).

Постійний запам'ятовуючий пристрій (ПЗП, ROM). Ця енергонезалежна пам'ять використовується для зберігання важливих інструкцій по експлуатації, програми самоперевірки та обмеженої версії IOS для запуску.Фактично ПЗП – це вбудована в мікросхему мікропрограма всередині маршрутизатора, яка може бути змінена тільки компанією Cisco.

Постійний перепрограмований запам'ятовуючий пристрій (ППЗП, NVRAM). Ця енергонезалежна пам'ять використовується як місце постійного зберігання файлу завантажувального конфігурації (startup-config), який може бути змінений під чса роботи пристрою.

Флеш-пам'ять (Flash). Це енергонезалежна пам'ять пристрою, що використовується в якості місця постійного зберігання образу IOS і інших системних файлів, таких як файли журналів, файли голосового конфігурації, HTML файли, конфігурації резервного копіювання та багато іншого. При перезавантаженні маршрутизатора образ IOS розгортається з флеш-пам'яті в ОЗП.

Залежно від завдань, що має виконувати обладнання Cisco у мережі, на ньому може розгортатися операційна система з певними можливостями. На рисунку 2.18 показаний приклад позначення файлу образу Cisco IOS, яке має декілька полів в імені, які описують призначення (можливості) цього образу IOS:

– **Апаратна платформа (Hardware Platform).** Перша частина імені файлу вказує апаратну платформу, для якої призначена ця версія (C2960 – для комутаторів серії Catalyst 2960; c1900 – для маршрутизаторів ISR серії 1900; тощо).

Рисунок 2.18 – Прийняті корпорацією Cisco позначення імені файлу образу IOS

– **Набір функцій (Feature Set).** Друга частина імені файлу характеризує різні функції, які реалізує цей файл. Користувач може вибрати будь-які набори функцій, які упаковані в образи програмного забезпечення. Кожен набір функцій містить певну підмножину з усього набору функцій програмного забезпечення Cisco IOS (universal – універсальний образ (функції активуються ліцензіями);

lanbase – базовий набір функцій рівня L2; ipbase – підтримка базової IP-маршрутизації; тощо).

– **Шифрування (Encryption).** Містить позначення рівня можливої криптографії, використовуються ідентифікатори k8/k9 (k8 – вказує на 64-бітове шифрування або шифрування з меншою кількістю бітів; k9 – вказує на шифрування з кількістю бітів більше за 64).

– **Тип пам'яті та формат стиснення файлу (Memory Location and Compression).** Четверта частина імені файлу IOS вказує формат його стиснення та де він має зберігатися. Зазвичай це дволітерний код, де перша літера – місце зберігання, друга – метод стиснення (перша літера: f – Flash, виконується безпосередньо з флеш-пам'яті; m – RAM, образ копіюється та виконується в оперативній пам'яті; r – ROM, виконується з постійної пам'яті; l – Relocatable, образ може змінювати своє положення в пам'яті під час роботи; друга літера: z – стиснуто алгоритмом zip; x – алгоритмом mzip; w – алгоритмом STAC).

– **Версія і випуск (Version Number & Release).** У п'ятій частині імені файлу вказується номер версії і випуску. У міру того, як розробляються нові версії IOS Cisco, номер версії зростає (M – стабільні версії; T – нові версії, служать для тестування технологій; E – спеціальні, корпоративні версії; S – версії для обладнання операторів зв'язку).

– **Розширення файлу (File extension).** Частина після крапки визначає тип файлу образу (.bin – двійковий виконуваний файл, який безпосередньо завантажується пристроєм; .tar – архів, який може містити образ .bin та файли веб-інтерфейсу (HTTP-файли керування)).

При виборі нової версії операційної системи Cisco IOS варто перевірити вимоги до обсягу оперативної та Flash-пам'яті. Зазвичай, чим новіша версія операційної системи, тим більше функцій до неї включено, отже, тим більших апаратних ресурсів, зокрема, пам'яті вона вимагатиме.

Для перевірки поточної версії операційної системи та вільної Flash-пам'яті можна використовувати команду ***show version***.

Після включення мережеве обладнання Cisco проходить наступні стадії завантаження:

– По-перше, пристрій завантажує програму самотестування (POST), що зберігається в ПЗУ. POST перевіряє підсистеми процесора, тестує оперативну динамічну пам'ять (DRAM) і частину флеш-пристрою, яка складає файлову систему флеш-пам'яті.

– Після цього на пристрої запускається програмне забезпечення початкового завантажувача. Початковий завантажувач – це невелика програма, яка зберігається в ПЗУ і запускається відразу після успішного завершення перевірки POST. Початковий завантажувач виконує низькорівневу ініціалізацію процесора, ініціалізує регістри процесора, які контролюють фізичну пам'ять, обсяг пам'яті і швидкість.

– Потім початковий завантажувач запускає файлову систему флеш-пам'яті на материнській платі, знаходить і завантажує образ операційної системи IOS за замовчуванням і передає їй управління пристроєм.

– Розгорнута операційна система IOS знаходить файл стартової конфігурації і завантажує його.

.

2.2 Методи отримання інформації про пристрої мережі і помилки

2.2.1 Протокол виявлення сусідніх пристроїв CDP

Протокол CDP (Cisco Discovery Protocol) – протокол виявлення устаткування Cisco, інструмент мережевого адміністратора, який допомогає побудувати базову схему структури мережі. Протокол показує інформацію виключно про безпосередньо підключені до цього вузла сусідні пристрої, але він все одно є дуже потужним засобом налаштування мережі. Протокол працює на канальному рівні, який об'єднує фізичне середовище передачі даних нижнього рівня з протоколами верхніх мережних рівнів, він використовується для отримання інформації про сусідні мережеві пристрої обладнання Cisco. Отримувана інформація включає типи підключених пристроїв, інтерфейси вузла, до яких сусідні пристрої підключені, інтерфейси, що використовуються сусідніми пристроями для створення з'єднань, а також моделі пристроїв.

Протокол CDP не залежить від середовища передачі і від протоколів, працює з будь-яким устаткуванням корпорації Cisco і в якості своєї основи використовує протокол доступу до підмережі (SNAP – Subnetwork Access Protocol). CDP є власним протоколом мережних облаштувань Cisco і працює тільки з мережними пристроями, випущеними компанією Cisco. Самою останньою версією протоколу CDP є версія 2 (CDPv2).

Протокол CDP запускається автоматично при завантаженні устаткування Cisco і дозволяє мережному пристрою знаходити сусідні вузли, на яких також запущений протокол CDP.

2.2.2 Налаштування протоколу CDP

Кожен пристрій з налагодженим протоколом CDP періодично відправляє повідомлення, також відомі як анонси (advertisement), усім сусіднім пристроям. За замовчуванням анонси розсилаються кожні 60 секунд.

Мережний пристрій, який розуміє анонси, які він періодично отримує від сусідніх пристроїв, зберігає отриману інформацію у CDP-таблиці. Якщо пристрій тричі не надіслав анонс (при значеннях за замовчуванням – 3 хвилини (180 секунд)), він видаляється з таблиці.

Основним завданням протоколу CDP є отримання даних про платформи сусідніх пристроїв і виконувані ними протоколи. CDP-фрейм може бути невеликим, проте містити масу корисної інформації про сусідні маршрутизатори і комутатори.

На рисунку 2.19 показано, як протокол CDP дозволяє адміністраторові отримати корисну інформацію про систему. Адміністратор може подивитися результати цього обміну CDP-інформацією за допомогою консолі, приєднаної до

локального маршрутизатора. Для відображення інформації про мережі, безпосередньо приєднані до маршрутизатора, можна скористатися командою **show cdp neighbors**.

```
Router#show cdp neighbors
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater, P - Phone
Device ID    Local Intrfce   Holdtme    Capability   Platform    Port ID
Switch       Fas 0/0          153            S       2960        Gig 0/1
Router       Fas 0/1          138            R       C2800       Fas 0/0
```

Рисунок 2.19 – Інформація протоколу CDP

Блоки інформації про кожен сусідній пристрій включають такі дані:

– ідентифікатор пристрою;
– номер і тип локального інтерфейсу;
– час утримання інформації;
– можливості пристрою;
– платформу;
– ідентифікатор інтерфейсу сусіда;
– доменне ім’я VTP (тільки у разі використання протоколу CDPv2);
– номер власної мережі VLAN (тільки у разі використання протоколу CDPv2);
– інформацію про наявність дуплексного з’єднання (тільки у разі використання протоколу CDPv2).

Для відображення усієї інформації, що міститься в повідомленнях використання протоколу CDP, можна використати попередню команду з додатковим ключем **show cdp meighbors detail**, як показано на рисунку 2.20.

Для включення протоколу CDP, отримання і відстежування CDP-інформації використовуються команди:

– **cdp run** в режимі глобальної конфігурації – включає протокол CDP в маршрутизаторі
– **cdp enable** в режим налаштування інтерфейсу – дозволяє використання протоколу CDP на інтерфейсі
– **clear cdp counters** в привілейованому режимі – скидає усі лічильники переданих даних в початковий стан.

Для відключення протоколу CDP, після того, як він був дозволений, слід використати команди **no cdp run** в режимі глобальної конфігурації або **no cdp enable** у відповідному режимі конфігурації інтерфейсу.

Аналогом протоколу CDP є ***Link Layer Discovery Protocol (LLDP)***, якій може підтримуватися на обладнанні інших виробників і виконувати схожі функції.

```
Router#show cdp neighbors detail

Device ID: Switch
Entry address(es):
  IP address : 192.168.10.10
Platform: cisco 2960, Capabilities: Switch
Interface: FastEthernet0/0, Port ID (outgoing port): GigabitEthernet0/1
Holdtime: 139

Version :
Cisco IOS Software, C2960 Software (C2960-LANBASE-M), Version 12.2(25)FX, RELEASE SOFTWARE (fc1)
Copyright (c) 1986-2005 by Cisco Systems, Inc.
Compiled Wed 12-Oct-05 22:05 by pt_team

advertisement version: 2
Duplex: full
---------------------------

Device ID: Router
Entry address(es):
  IP address : 11.1.1.2
Platform: cisco C2800, Capabilities: Router
Interface: FastEthernet0/1, Port ID (outgoing port): FastEthernet0/0
Holdtime: 124

Version :
Cisco IOS Software, 2800 Software (C2800NM-ADVIPSERVICESK9-M), Version 12.4(15)T1, RELEASE SOFTWARE (fc2)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2007 by Cisco Systems, Inc.
Compiled Wed 18-Jul-07 06:21 by pt_rel_team

advertisement version: 2
Duplex: full
```

Рисунок 2.20 – Розширена інформація протоколу CDP

2.2.3 Короткі відомості про інші протоколи управління мережами

Протокол **SNMP** (Simple Network Management Protocol) – це прикладний протокол для керування та моніторингу мережевих пристроїв в IP-мережах. Архітектура протоколу базується на моделі «менеджер–агент». Менеджер (станція керування мережею, NMS) ініціює запити, тоді як агент (програмний модуль на мережевому пристрої) надає дані про стан об’єкта.

Інформаційна структура SNMP описується базою керуючої інформації (MIB – Management Information Base), яка є ієрархічною базою даних об’єктів. Кожен параметр (наприклад, завантаження процесора або статус інтерфейсу) має свій унікальний ідентифікатор об’єкта (OID).

Версії протоколу SNMPv1 та SNMPv2c використовують текстові рядки (communities) для автентифікації, що не забезпечує належної безпеки. SNMPv3 є сучасним стандартом, оскільки впроваджує криптографічний захист (шифрування) та цілісність повідомлень.

Протокол також підтримує механізм Traps, який на відміну від стандартного опитування агента менеджером, дозволяє агенту самостійно сповіщати менеджера про критичні події в реальному часі, що мінімізує навантаження на канал зв'язку.

Протокол **Telnet** (Teletype Network) – це мережевий протокол прикладного рівня, розроблений для забезпечення двостороннього інтерактивного текстового зв'язку через віртуальний термінал. Використовуючи клієнт-серверну архітектуру, Telnet дозволяє користувачу ініціювати сесію віддаленого керування операційною системою пристрою (зокрема Cisco IOS) через командний інтерфейс (CLI).

Telnet розглядається як протокол з високим рівнем уразливості, основним його недоліком є передача всієї інформації, включаючи дані автентифікації (логіни та паролі), у відкритому текстовому вигляді. Це робить сесію вразливою до атак типу «людина посередині» (Man-in-the-Middle) та перехоплення пакетів. У сучасних корпоративних мережах використання Telnet обмежене виключно ізольованими сегментами для налагодження обладнання.

Протокол **SSH** (Secure Shell) – це протокол прикладного рівня, призначений для безпечного віддаленого доступу та передачі даних. Він був розроблений як захищена заміна для протоколу Telnet. SSH забезпечує конфіденційність і цілісність даних шляхом використання сильних методів шифрування та автентифікації.

Функціонування SSH базується на використанні криптографії з відкритим ключем для встановлення безпечного з'єднання. Протокол забезпечує:

– Автентифікацію: перевірку справжності клієнта та сервера.

– Шифрування: захист даних від несанкціонованого перегляду під час передачі.

– Цілісність: гарантування того, що пакет не був змінений у дорозі.

Протокол **ICMP** (Internet Control Message Protocol) – це допоміжний протокол стеку TCP/IP, який використовується для передачі діагностичних повідомлень та інформації про помилки. На відміну від TCP/UDP, ICMP не використовується безпосередньо для обміну даними користувачів, а слугує інструментом зворотного зв'язку між вузлами мережі.

– Утиліта Ping: використовує типи повідомлень Echo Request та Echo Reply. Вона дозволяє визначити доступність віддаленого вузла на рівні IP та виміряти кругову затримку (RTT). Відсутність відповіді може свідчити як про несправність вузла, так і про блокування ICMP-трафіку міжмережевим екраном.

– Утиліта Traceroute: використовується для візуалізації маршруту слідування пакета. Принцип роботи базується на маніпуляції полем TTL (Time To Live) в IP-заголовку. Послідовно збільшуючи TTL від 1, утиліта змушує кожен проміжний маршрутизатор відкидати пакет і повертати повідомлення ICMP внаслідок Time Exceeded. Це дозволяє ідентифікувати всі хопи (вузли) на шляху до цільового пристрою та виявляти затримки на конкретних ділянках мережі.

2.3 Протокол взаємодії мереж 2 рівня STP

2.3.1 Побудова та недоліки топології на комутаторах з резервними каналами зв'язку

Резервування є важливою частиною ієрархічної конструкції й запобігає виникненню перебоїв в наданні сервісів користувачам. Для резервування потрібне додавання надмірних фізичних шляхів, а також логічне резервування.

Трирівнева ієрархічна модель мережі (рисунок 2.21), яка містить ядро, рівні розподілу і доступу з надлишковістю, дозволяє усунути точку відмови в мережі за рахунок альтернативних фізичних каналів для передачі даних по мережі.

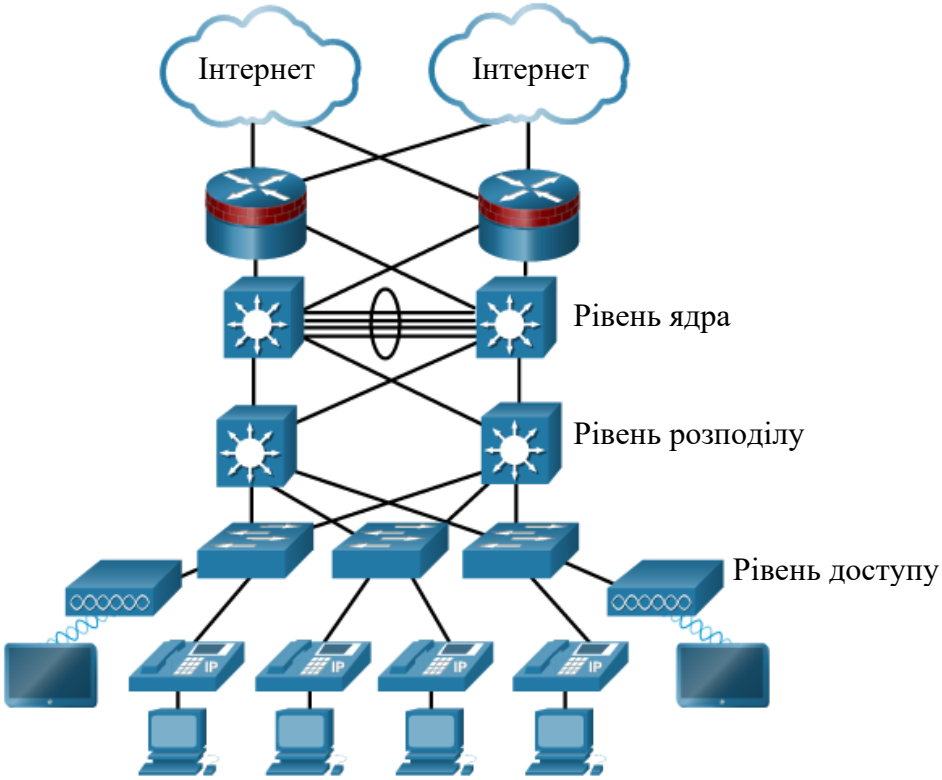


Рисунок 2.21 – Ієрархічна модель трьохрівневої мережі

В комутованій мережі (рисунок 2.22) наявність надлишковості дозволяє створювати альтернативні шляхи передачі інформації:

– PC1 взаємодіє з PC4 через надлишкову топологію мережі, використовуючи Магістраль 1.

– При розриві мережного каналу між комутаторами S1 і S2 шлях між комп'ютерами PC1 і PC4 автоматично коригується певним протоколом для компенсації цього розриву.

– При відновленні мережного з'єднання між S1 і S2 цей шлях коригується протоколом заново, і трафік на PC4 направляється безпосередньо від S2 до S1.

Внаслідок роботи комутаторів, особливо в процесі отримання даних і пересилання, можуть виникати логічні петлі 2-го рівня. Отже, при наявності декількох шляхів між двома пристроями 2-го рівня виникає петля. Петля 2-го рівня може призвести до трьох основних проблем, а саме:

Нестабільність бази даних MAC-адрес

Нестабільності бази даних MAC-адрес можуть статися внаслідок пересилання широкомовних кадрів.

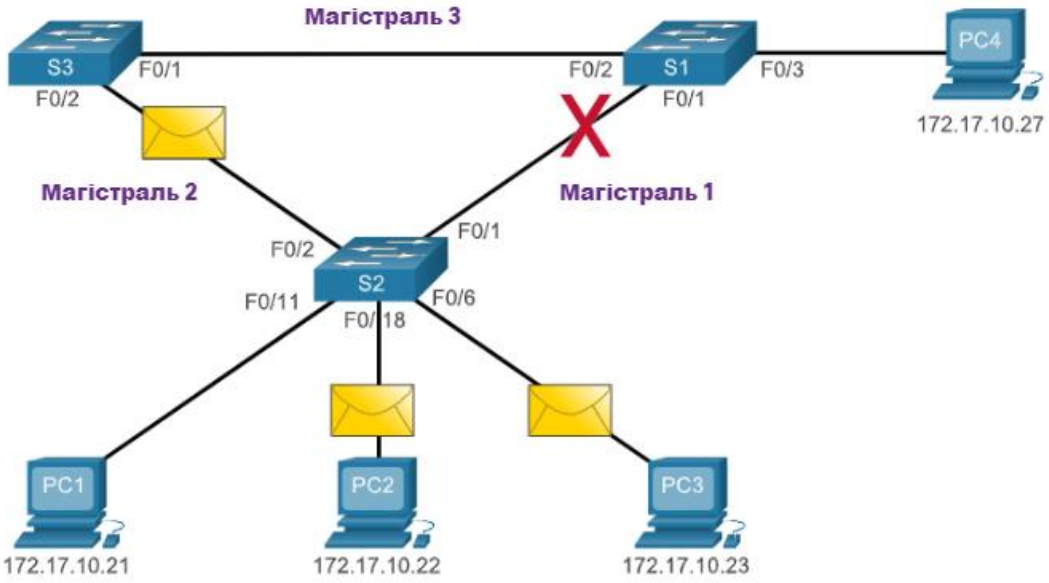


Рисунок 2.22 – Мережа 2-го рівня з надлишковістю

Широкомовні кадри пересилаються з усіх інтерфейсів комутатора, за винятком вхідного інтерфейсу. Це гарантує, що всі пристрої в домені широкомовної розсилки можуть отримати кадр. При наявності більш одного шляху, з якого пересилається кадр, може виникнути нескінченна петля. При появі петлі виникає можливість постійної зміни таблиці MAC-адрес на комутаторі оновленнями з кадрів широкомовної розсилки, що й призводить до нестабільності бази даних MAC-адрес.

Наприклад, комп'ютер PC1 відправляє широкомовний кадр на S2. S2 приймає широкомовний кадр на інтерфейс F0/11. Коли S2 приймає широкомовний кадр, він оновлює свою таблицю MAC-адрес, щоб зареєструвати доступність PC1 на інтерфейсі F0/11.

Оскільки цей кадр широкомовний, S2 пересилає кадр з усіх інтерфейсів, включаючи Магістраль 1 і Магістраль 2. При надходженні кадру широкомовної розсилки на комутатори S3 і S1 ці комутатори оновлюють свої таблиці MAC-адрес, вказуючи, що комп'ютер PC1 доступний з інтерфейсу F0/1 комутатора S1 і з інтерфейсу F0/2 комутатора S3.

Оскільки цей кадр є широкомовним, S3 і S1 пересилають кадр з усіх інтерфейсів, за винятком вихідного вхідного інтерфейсу. S3 відправляє широкомовний кадр з PC1 на S1. S1 відправляє широкомовний кадр з PC1 на S3.

Всі комутатори оновлюють свою таблицю MAC-адрес з урахуванням неправильного інтерфейсу PC1.

Кожен комутатор пересилає кадр широкомовної розсилки з усіх своїх інтерфейсів, за винятком вхідного інтерфейсу, внаслідок чого обидва комутатора пересилають кадр на комутатор S2.

При отриманні комутатором S2 кадрів широкомовної розсилки з S3 і S1 таблиця MAC-адрес оновлюється останнім записом, отриманої з інших двох комутаторів.

Цей процес повторюється кожного разу заново доти, поки петля не буде розірвана фізичним розривом або вимиканням живлення одного з комутаторів у петлі. При цьому створюється високе навантаження на центральні процесори (ЦП) на всіх комутаторах, що беруть участь в петлі. Оскільки між усіма комутаторами в петлі постійно передаються одні і ті ж кадри, ЦП комутатора доводиться обробляти великий обсяг даних. При цьому знижується продуктивність комутатора при надходженні іншого трафіку.

Вузол, який бере участь в мережній петлі, недоступний для інших вузлів в мережі. Крім того, внаслідок постійних змін в таблиці MAC-адрес комутатор не знає, з якого інтерфейсу слід пересилати кадри одноадресної розсилки. У вищевказаному прикладі для PC1 перераховані неправильні інтерфейси. Будь-якій одноадресний кадр, призначений для PC1, пересилається в мережі по петлі, подібно широкомовною кадрам. Через зростаючу кількість кадрів, які формують петлю в мережі, в кінцевому підсумку утворюється широкомовний шторм.

Широкомовний шторм

Широкомовний шторм виникає в разі, коли в петлю на другому рівні потрапляє стільки кадрів широкомовної розсилки, що при цьому споживається вся доступна смуга пропускання. Відповідно, для легітимного трафіку немає доступної смуги пропускання, і мережа стає недоступною для обміну даними. Це призводить до відмови в обслуговуванні (DoS).

Широкомовний шторм (рисунок 2.23) неминучий в мережі, де виникла петля. Чим більше пристроїв відправляє широкомовне розсилання по мережі, тим більше трафіку потрапляє в цикл і споживає ресурси. В кінцевому рахунку це створює широкомовний шторм, що призводить до збоїв в мережі:

– PC1 передає кадр широкомовної розсилки в мережу, де виникла петля.

– Кадр широкомовної розсилки циклічно передається між усіма з’єднаними між собою комутаторами в мережі.

– PC4 також передає кадр широкомовної розсилки в мережу, де виникла петля.

– Кадр широкомовної розсилки PC4 потрапляє в петлю між усіма взаємно з’єднаними комутаторами, подібно кадру широкомовної розсилки PC1.

– Чим більше пристроїв відправляє широкомовне розсилання по мережі, тим більше трафіку потрапляє в цикл і споживає ресурси. В кінцевому рахунку це створює широкомовний шторм, що призводить до збоїв в мережі.

– Коли мережу повністю насичена трафіком широкомовної розсилки, який циклічно передається між комутаторами, новий трафік відкидається комутатором, оскільки він не в змозі його обробити.

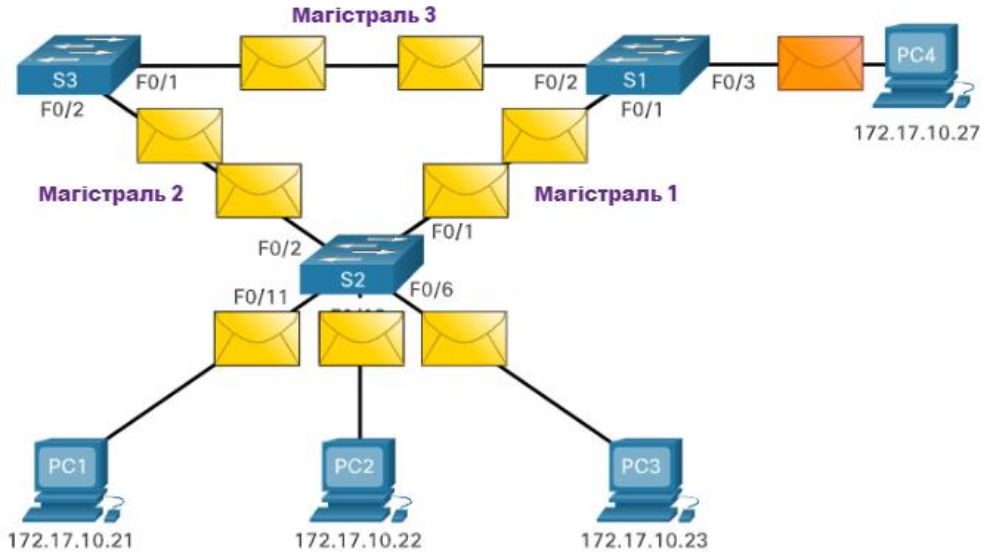


Рисунок 2.23 – Широкомовний шторм

Широкомовні шторми також мають і ряд інших наслідків. Оскільки широкомовний трафік пересилається з кожного інтерфейсу на комутаторі, всім підключеним до нього пристроям необхідно обробляти весь цей широкомовний трафік, обсяг якого в мережу з петлею весь час зростає. Це може викликати порушення функціонування кінцевого пристрою, якщо він не зможе обробляти таке велике навантаження трафіку на своїй мережній інтерфейсній платі (NIC).

Широкомовний шторм може виникнути за кілька секунд, так як підключення до мережі пристрою регулярно відправляють кадри широкомовної розсилки, наприклад запити ARP. В результаті при виникненні петлі мережа 2-го рівня швидко стане непрацездатною.

Дубльовані одноадресні кадри

Кадри широкомовної розсилки є не єдиним типом кадрів, на які впливає виникнення петель. Невідомі одноадресні кадри, відправлені в циклічну мережу, можуть призводити до появи дубльованих кадрів, що надходять на цільовий пристрій. Невідомий одноадресний кадр з комутатора формується, коли у комутатора немає MAC-адреси призначення в таблиці MAC-адрес, і він повинен переслати цей кадр зі всіх своїх інтерфейсів, за винятком вхідного інтерфейсу:

– PC1 відправляє кадр одноадресної розсилки, призначений для PC4.

– У таблиці MAC-адрес на комутаторі S2 немає запису PC4. У спробі знайти PC4 він розсилає невідомий одноадресний кадр з усіх своїх інтерфейсів, за винятком інтерфейсу, який прийняв цей трафік.

– Кадр надходить на комутатори S1 і S3.

– На комутаторі S1 є запис MAC-адреси PC4, тому він пересилає кадр на PC4.

– На комутаторі S3 є запис PC4 в таблиці MAC-адрес, тому він пересилає одноадресний кадр з інтерфейсу Trunk3 на S1.

– S1 приймає дубльований кадр і відправляє його на PC4.

– Таким чином, PC4 приймає два однакових кадри.

Більшість протоколів верхнього рівня не призначені для розпізнавання дубльованих передач. Як правило, протоколи, що використовують механізм нумерації послідовності, припускають, що стався збій передачі, і номер послідовності переходить в інший сеанс обміну даними. Інші протоколи намагаються перекласти дубльовану передачу на відповідний протокол верхнього рівня для обробки з можливим відхиленням.

Протоколи LAN 2-го рівня, наприклад Ethernet, не включають в себе механізм для розпізнавання і видалення нескінченних петльових кадрів. Деякі протоколи 3-го рівня використовують механізми часу життя (TTL), які обмежують кількість спроб повторної передачі пакетів мережними пристроями 3-го рівня. Пристрої 2-го рівня не мають такого механізму, тому продовжують повторно передавати петльовий трафік необмежено довго. Для вирішення таких проблем був розроблений протокол **STP** (Spanning Tree Protocol) – механізм запобігання петель 2-го рівня.

Протокол spanning-tree за замовчуванням включений на комутаторах Cisco, запобігаючи, таким чином, виникненню петель 2-го рівня. Щоб уникнути виникнення петель 2-го рівня потрібно керувати маршрутами. Вибрати оптимальні маршрути, а альтернативні маршрути – заблокувати, проте вони повинні бути негайно доступні в разі збою основного маршруту. Протокол STP використовується для створення єдиного шляху через мережу 2-го рівня.

Термін Spanning Tree Protocol (протокол spanning-tree) і абревіатура STP відповідають протоколу 802.1D. У новітній документації IEEE по протоколу сполучного дерева (IEEE-802-1D-2004) зазначено: «STP тепер замінений протоколом Rapid Spanning Tree Protocol (RSTP)».

2.3.2 Призначення та логіка роботи протоколу STP

Протокол STP забезпечує наявність тільки одного логічного шляху між усіма вузлами призначення в мережі шляхом навмисного блокування резервних шляхів, які могли б викликати петлю.

Порт вважається заблокованим (рисунок 2.24), коли заблокована відправка і прийом даних на цей інтерфейс, за виключенням кадрів, які використовуються самим протоколом STP. Такі кадри називають BPDU (Bridge Protocol Data Unit) – це службові кадри даних, якими обмінюються комутатори для функціонування протоколів сімейства STP. Важливо, що всі фізичні шляхи, як і раніше використовуються для забезпечення надлишковості, однак логічно певні з цих шляхів відключені з метою запобігання петель. Проте коли буде необхідно компенсувати несправність мережевого кабелю або комутатора, протокол STP повторно розрахує шляхи і зніміть блокування з певних інтерфейсів, щоб дозволити активацію резервного шляху.

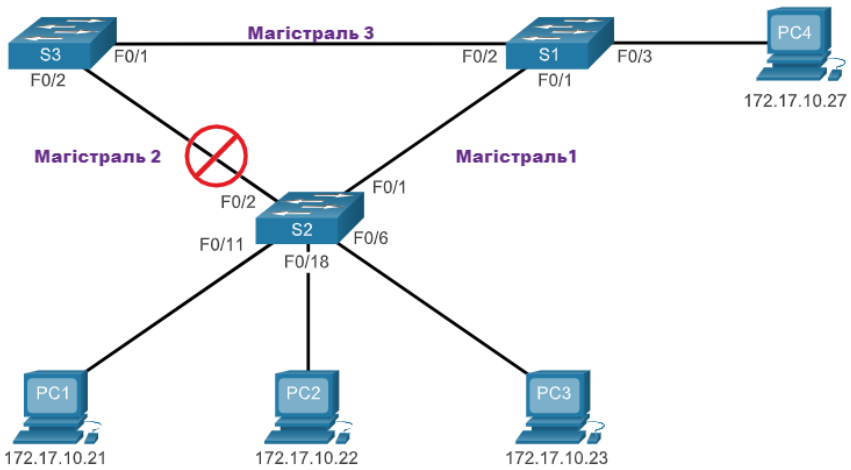


Рисунок 2.24 – Логіка роботи протоколу STP

У розглянутому прикладі протокол STP включений на всіх комутаторах:

– PC1 відправляє широкомовне розсилання в мережу.

– S2 налаштований з використанням протоколу STP, і для інтерфейсу F0/2 задано стан блокування. Стан блокування не дозволяє інтерфейсам пересилати призначені для користувача дані, що запобігає виникненню петлі. S2 пересилає кадр широкомовної розсилки з усіх інтерфейсів комутатора, за винятком інтерфейсу джерела – PC1 і інтерфейсу F0/2.

– S1 приймає кадр широкомовної розсилки і пересилає його з усіх інтерфейсів комутатора крім того, звідки він надходить, тобто до PC4 і S3. S3 пересилає кадр з інтерфейсу F0/2, але S2 не пропускає цей кадр. Виникнення петлі 2-го рівня попереджено.

Ролі інтерфейсів

У протоколах IEEE 802.1D STP і RSTP для визначення інтерфейсів комутатора в мережі, які мають бути переведені в стан блокування для запобігання виникненню петель, використовується алгоритм протоколу сполучного дерева (STA). STA призначає один з комутаторів як кореневий міст і використовує його в якості точки прив'язки для розрахунку всіх шляхів.

Визначивши найкращі шляхи до кореневого мосту для кожного комутатора, алгоритм STA призначає ролі інтерфейсам комутаторів. Ролі інтерфейсів описують їх зв'язок з кореневим мостом в мережі, а також вказують, чи дозволено для них пересилання трафіку:

– *Кореневі інтерфейси* – інтерфейси комутатора, найближчі до кореневого мосту з точки зору загальної вартості маршруту до нього. На рисунку 2.24 кореневим інтерфейсом, обраним STP на S2, буде F0/1 – канал між S2 і S1. Кореневим інтерфейсом, обраним STP на S3, буде F0/1 – канал між S3 і S1. Кореневі інтерфейси вибираються для кожного комутатора окремо.

– *Призначені інтерфейси* – все некореневі інтерфейси, яким, проте, дозволено пересилати трафік по мережі. На рисунку 2.24 інтерфейси комутатора S1 (F0/1 і F0/2) є призначеними інтерфейсами. На комутаторі S2 інтерфейс F0/2 також налаштований як призначений інтерфейс. Призначені інтерфейси

вибираються для кожного сегмента на основі вартості кожного інтерфейсу з обох сторін сегмента та сукупну вартість, обчисленої протоколом STP для маршруту від цього інтерфейсу до кореневого мосту. Якщо на одному кінці сегмента знаходиться кореневий інтерфейс, на іншому кінці буде призначений інтерфейс. Всі інтерфейси на кореневому мосту є призначеними інтерфейсами.

– *Альтернативні і резервні інтерфейси* – знаходяться в стані відхилення або блокування для запобігання петель. На рисунку 2.24 STA налаштував порт F0/2 на комутаторі S3 в ролі альтернативного інтерфейсу. Цей інтерфейс на комутаторі S3 знаходиться в стані блокування. Альтернативні інтерфейси вибираються тільки на каналах, де ні з одного з кінців не має кореневого інтерфейсу. Блокується тільки один кінець сегмента, це дозволяє при необхідності швидше перейти в стан пересилки.

– *Відключені інтерфейси* – відключеним називається комутаційний інтерфейс, живлення якого відключено адміністратором.

Кореневий міст

Як показано на рисунку 2.25, протокол spanning-tree для кожної комутованої мережі LAN або кожного домену широкомовного розсилання визначає комутатор, який виконує функцію кореневого моста (на рисунку 2.25 це комутатор S1). Кореневий міст служить точкою прив'язки для всіх розрахунків протоколу spanning-tree, дозволяючи визначити надлишкові шляхи, які слід заблокувати.

Кореневий міст обирається за допомогою спеціального процесу вибору. Всі комутатори, які беруть участь в STP, обмінюються кадрами BPDU для визначення комутатора з найменшим ідентифікатором моста (BID) в мережі. Комутатор з найменшим значенням BID автоматично стає кореневим мостом для розрахунків STA.

На рисунку 2.26 показані поля BID. Ідентифікатор BID складається з значення *пріоритету, розширеного ідентифікатора системи і MAC-адреси комутатора*.

Пріоритет моста являє собою значення, яке можна використовувати для визначення комутатора, що стане кореневим мостом. Комутатор з найнижчим пріоритетом, який має найменше значення BID, стає кореневим мостом. Наприклад, щоб призначити в якості кореневого моста конкретний комутатор, то для нього слід задати більш низьке значення пріоритету, ніж для інших комутаторів в мережі. Значення пріоритету за замовчуванням для всіх комутаторів Cisco дорівнює значенню 32768. Значення варіюються в діапазоні від 0 до 61440 з кроком в 4096. Допустимі значення пріоритету: 0, 4096, 8192, 12288, 16384, 20480, 24576, 28672, 32768, 36864, 40960, 45056, 49152, 53248, 57344 і 61440. Всі інші значення відхиляються. Пріоритет моста 0 має перевагу в порівнянні з усіма іншими значеннями.

Відомості про мережу VLAN включені в кадр BPDU за допомогою ***розширеного ідентифікатора системи***. Як показано на рисунку 2.26, в ідентифікаторі моста після 4 бітів пріоритету моста, розміщені 12 бітів – для розширеного ідентифікатора системи, який дорівнює номеру мережі VLAN, що

бере участь в конкретному процесі STP. Отже, оскільки крайні праві 12 бітів резервуються для ідентифікатора мережі VLAN, а крайні ліві 4 біти – для пріоритету моста, стає зрозумілим чому значення пріоритету моста можна налаштувати тільки кратним 4096 або 2^{12}. Якщо крайні ліві біти будуть рівні 0001, то пріоритет моста має значення 4096. Якщо крайні ліві біти рівні 1111, пріоритет моста має значення 61440 (= 15 × 4096).

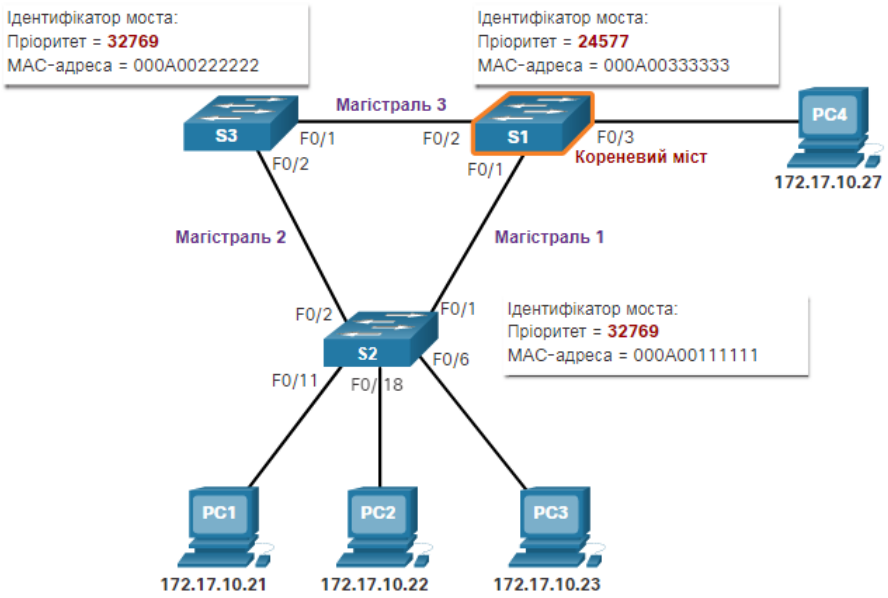


Рисунок 2.25 – Визначення кореневого мосту

Рисунок 2.26 – Ідентифікатор моста (BID)

Всі комутатори в домені широкомовної розсилки беруть участь в процесі вибору. Після завантаження комутатора вони починають розсилати кадри BPDU *з інтервалом у дві секунди*. **BPDU** – це кадр повідомлень протоколу STP (рисунок 2.27), яким обмінюються комутатори для забезпечення роботи даного протоколу.

Кадр BPDU містить 12 спеціальних полів, що містять інформацію про шляхи і пріоритеті, які використовуються для визначення кореневого моста і вартості шляхів до нього.

– У перших чотирьох полях вказані протокол, версія, тип повідомлення і прапори стану.

– Наступні два поля служать для ідентифікації комутатора, якого відправник кадру вважає кореневим мостом і вартості шляху до нього від відправника.

– Далі два поля, що ідентифікують сам комутатор-відправник та його інтерфейс, через який відбулась відправка кадру.

– Останні чотири поля все є полями таймерів, що визначають частоту відправлення повідомлень BPDU і тривалість зберігання інформації, отриманої через процес BPDU.

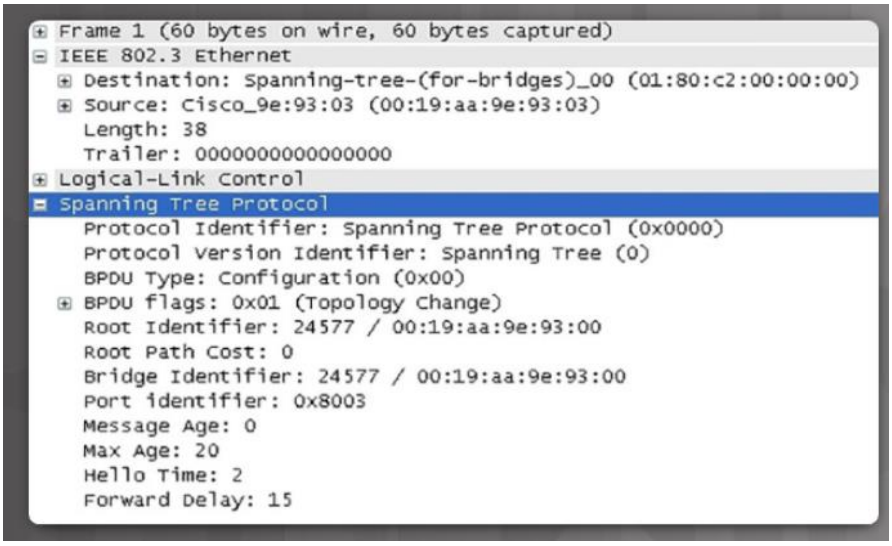


Рисунок 2.27 – Приклад кадру BPDU

Кадр BPDU, показаний на рисунку 2.27, був перехоплений за допомогою програми Wireshark. У цьому прикладі кадр BPDU містить більшу кількість полів, ніж описано вище. Повідомлення BPDU при передачі по мережі інкапсулюється в кадр Ethernet, де в заголовку вказуються адреси джерела і призначення кадру BPDU. Кадр на рисунку містить MAC-адресу призначення 01:80:C2:00:00:00, яка є адресою групової розсилки для групи протоколу spanning-tree. Коли адресується кадр з цим MAC-адресою, кожен комутатор, на якому налаштований протокол сполучного дерева, приймає і зчитує інформацію з кадру. Всі інші пристрої в мережі нехтують цим кадром.

Кожен комутатор в домені широкомовної розсилки спочатку передбачає, що він є кореневим мостом. Надалі комутатори пересилають свої кадри BPDU, а суміжні комутатори в домені широкомовної розсилки зчитують з них дані про ідентифікатор кореневого моста відправника. Якщо ідентифікатор кореневого моста отриманого кадру BPDU має менше значення, ніж ідентифікатор кореневого моста на приймаючому комутаторі, то в цьому випадку комутатор-приймач оновлює в себе ідентифікатор кореневого моста, вказуючи в якості кореневого моста комутатор, зазначений в кадрі BPDU. Цей комутатор не обов’язково розташований поблизу, це може бути будь-який інший комутатор в домені широкомовної розсилки. Потім комутатор пересилає нові кадри BPDU з меншим значенням ідентифікатора кореневого моста на інші суміжні комутатори. Поступово комутатор з найменшим значенням ідентифікатора BID визначається як кореневий міст для протоколу spanning-tree певного домену широкомовної розсилки.

На рисунку 2.25 комутатор S1 має більш низьке значення пріоритету, ніж інші комутатори. Тому він є кращим кореневим мостом для даного примірника протоколу сполучного дерева.

Якщо все комутатори налаштовані з однаковим пріоритетом, як у випадку, коли всі комутатори зберігають конфігурацію за замовчуванням з пріоритетом 32768 – MAC-адреса стає визначальним фактором в тому, який комутатор стане кореневим мостом, як показано на рисунку 2.28.

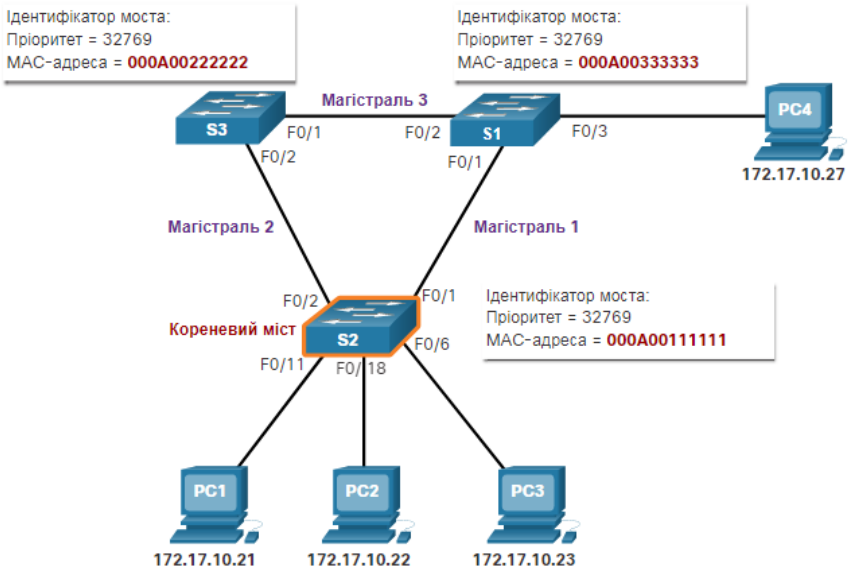


Рисунок 2.28 – Визначення кореневого мосту за MAC-адресою

У цьому прикладі для всіх комутаторів використовується значення 32769. Це значення засноване на значенні пріоритету за умовчанням 32768 і номері мережі VLAN 1, що застосована в кожному з комутаторів (32768 + 1). В такому випадку, комутатор з найнижчим значенням MAC-адреси вважається кращим кореневим мостом. У цьому прикладі S2 має найменше значення MAC-адреси і буде призначений кореневим мостом для протоколу spanning-tree цього домену.

Кореневий міст вибирається для кожного екземпляра протоколу spanning-tree. Можлива наявність декількох окремих кореневих мостів для різних наборів мереж VLAN. Якщо всі порти на всіх комутаторах є учасниками мережі VLAN 1, значить, існує тільки один екземпляр протоколу spanning-tree. Розширений ідентифікатор системи включає в себе ідентифікатор мережі VLAN і бере участь в тому, як визначаються примірники протоколу сполучного дерева.

Вартість шляхів до кореневого моста

Після визначення кореневого моста для примірника протоколу spanning-tree, STA починає процес визначення *оптимальних шляхів до кореневого моста* від всіх некореневих комутаторів в домені широкомовної розсилки.

Отримані алгоритмом STA значення вартості шляхів в подальшому використовуються для визначення інтерфейсів, що підлягають блокуванню. Поки STA визначає оптимальні шляхи до кореневого моста для всіх інтерфейсів комутаторів мережі пересилання трафіку у цьому домені широкомовної розсилки заблоковане.

Інформацію про вартість шляху до кореневого мосту називають вартістю внутрішнього кореневого шляху. Вона дорівнює сумі вартості окремих портів на шляху від даного комутатора до кореневого мосту.

Комутатори відправляють повідомлення BPDU, які включають в себе вартість кореневого шляху. Це вартість шляху від комутатора-відправника до кореневого мосту. Коли комутатор отримує блок BPDU, він додає вартість свого вхідного порту до отриманого значення для визначення своєї вартості для отриманого внутрішнього кореневого шляху. Якщо для вибору є кілька шляхів, то шляхи з найменшою вартістю стають кращими, а всі інші надлишкові шляхи блокуються.

Вартість портів за замовчуванням визначається швидкістю роботи інтерфейсу, стандартні значення наведені в таблиці 2.1.

Таблиця 2.1 – Стандартні значення вартості портів для STP

Швидкість каналу	**Вартість STP, IEEE 802.1D-1998**	**Вартість RSTP, IEEE 802.1W-2004**
10 Гбіт/с	2	2000
1 Гбіт/с	4	20 000
100 Мбіт/с	19	200 000
10 Мбіт/с	100	2 000 000

Хоча значення вартості шляху за замовчуванням пов’язано з швидкістю роботи інтерфейсу комутатора, значення вартості порту можна налаштувати. Можливість настройки окремих портів надає адміністратору необхідну гнучкість при контролі шляхів протоколу spanning-tree до кореневого мосту.

Щоб налаштувати вартість порту інтерфейсу, треба ввести команду *spanning-tree cost* ***value*** в режимі конфігурування інтерфейсу. Це значення може перебувати в діапазоні між 1 і 200 000 000 (наприклад, ***spanning-tree cost 25***). Щоб повернути вартість порту до значення за замовчуванням, треба ввести в тому ж режимі команду ***no spanning-tree cost.***

У прикладі на рисунку 2.29 вартість внутрішнього кореневого шляху від S2 до кореневого моста S1 по шляху 1 дорівнює 19 (на основі зазначеної IEEE вартості індивідуального порту), а вартість внутрішнього кореневого шляху для шляху 2 дорівнює 38. Оскільки загальна вартість шляху 1 до кореневого мосту нижче, саме цей шлях є кращим. Протокол STP блокує резервний шлях, щоб уникнути формування петлі.

Щоб перевірити порт і вартість внутрішнього кореневого шляху до кореневого мосту використовують команду *show spanning-tree*. Поле вартості (Cost) у верхній частині вихідних даних являє собою вартість внутрішнього кореневого шляху – загальну вартість шляху до кореневого мосту. Це значення змінюється в залежності від кількості портів комутатора, які необхідно пройти на шляху до кореневого мосту. У вихідних даних все інтерфейси також визначені зі значенням вартості окремого порту 19.

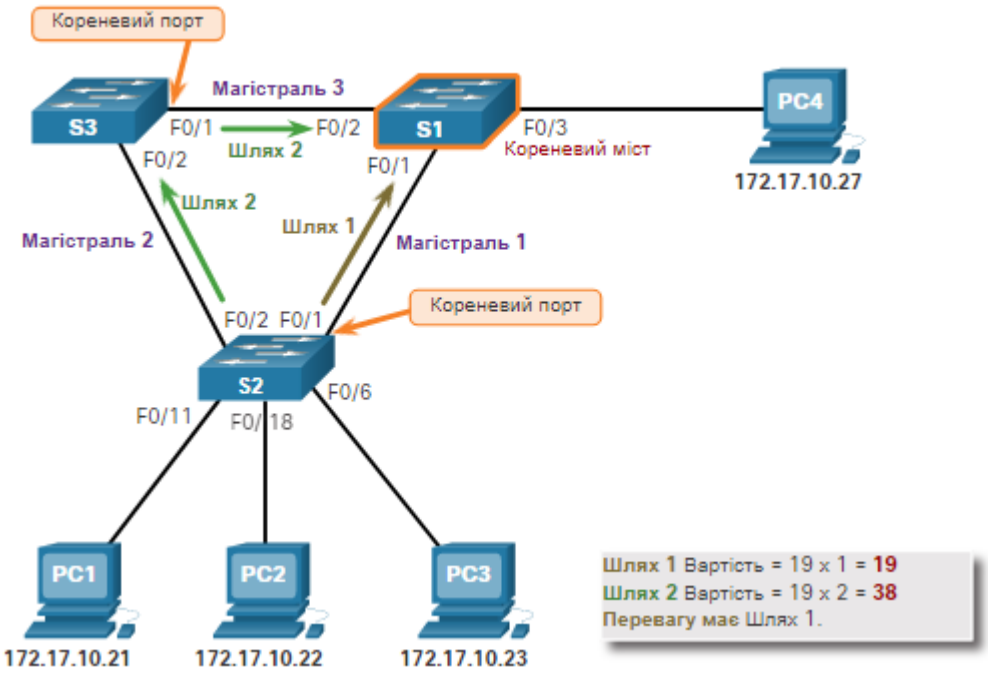


Рисунок 2.29 – Визначення вартості шляхів

На кореневому мосту не буде ніяких кореневих портів – всі порти на кореневому мосту будуть призначеними. Для комутатора, який не є кореневим мостом топології мережі, буде визначено лише один кореневий порт (рисунок 2.30)

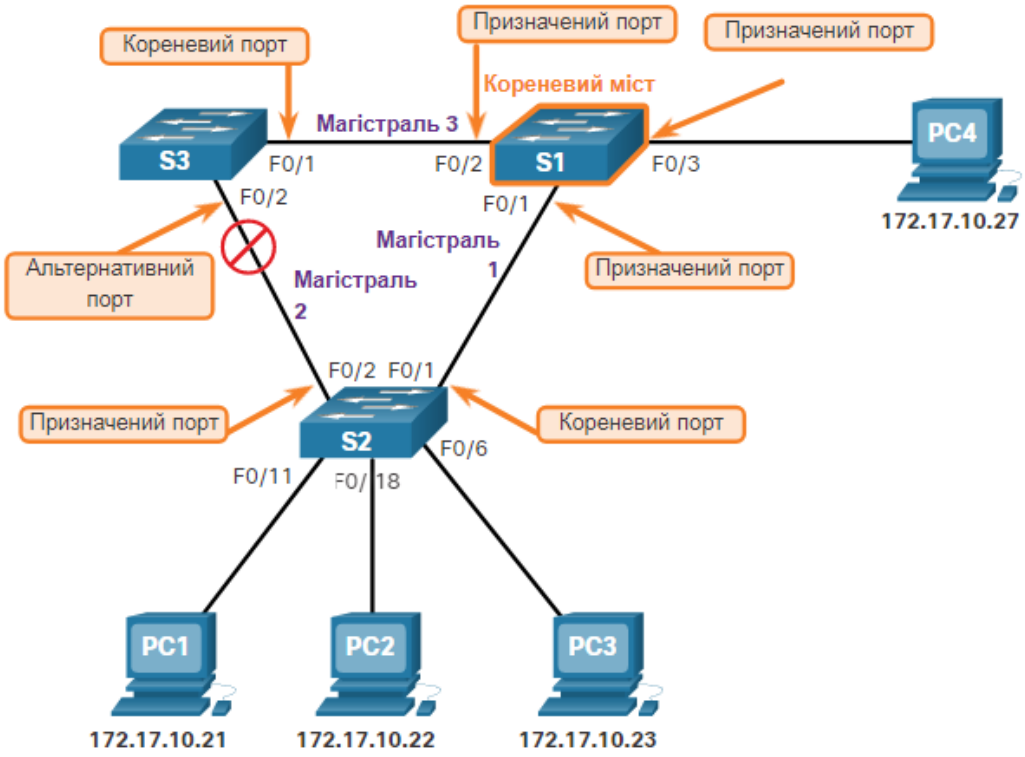


Рисунок 2.30 – Розподіл функцій портів

2.3.3 Сучасні версії протоколу STP, їхні особливості

З моменту створення вихідного стандарту IEEE 802.1D було розроблено кілька різновидів протоколів STP.

До різновидів протоколів STP відносяться наступні:

– STP. Це початкова версія IEEE 802.1D (802.1D-1998 і більш рання), яка запобігає формуванню петель в топології мережі з резервними каналами. Загальний протокол spanning-tree (CST): передбачає використання тільки одного примірника протоколу spanning-tree для всієї мережі з мостовим з'єднанням незалежно від кількості мереж VLAN.

– Швидкий протокол STP (RSTP) або IEEE 802.1w: доопрацьований протокол STP, який забезпечує більш швидке сходження, ніж протокол STP. 802.1D-2004: оновлена версія стандарту STP, в яку входить IEEE 802.1w.

– PVST+ є вдосконаленим протоколом компанії Cisco, в якому для кожного окремого VLAN використовується окремий екземпляр RSTP. Розглянутий варіант протоколу spanning-tree підтримує PortFast, UplinkFast, BackboneFast, BPDU guard, BPDU filter, root guard і loop guard.

– Rapid PVST+ – вдосконалений корпорацією Cisco протокол RSTP, який використовує PVST +. Rapid PVST + надає окремий екземпляр 802.1w для кожної мережі VLAN. Кожен окремий екземпляр підтримує функції PortFast, BPDU guard, BPDU filter, root guard і loop guard.

– Протокол множинного єднального дерева (Multiple Spanning Tree Protocol, MSTP) – це стандарт IEEE, натхненний ранньою фірмовою версією протоколу, Multiple Instance STP (MISTP), розробленого Cisco. MSTP поєднує декілька VLAN у одному екземплярі єднального дерева.

– Множинне єднальне дерево (Multiple Spanning Tree, MST) — це реалізація MSTP компанії Cisco, яка надає до 16 екземплярів RSTP і поєднує декілька VLAN з однаковою фізичною і логічною топологією у спільному екземплярі RSTP. Кожен екземпляр підтримує PortFast, захист BPDU, фільтр BPDU, захист кореня і захист від петель.

Мережному фахівцю, який відповідає за адміністрування комутаторів, треба прийняти рішення щодо того, який тип протоколу STP необхідно реалізувати. Деякі характеристики версій протоколу STP наведені в таблиці 2.2.

Таблиця 2.2 – Характеристики версій протоколу STP

Протокол	**Стандарт**	**Потрібні ресурси**	**Конвергенція**	**Розрахунок дерева**
STP	802.1D	Низкі	Повільна	Одне
PVST+	Cisco	Високі	Повільна	На VLAN
RSTP	802.1w	Середні	Висока	На VLAN
Rapid PVST+	Cisco	Дуже високі	Висока	На VLAN
MSTP	802.1s, Cisco	Середні	Висока	На екземпляр (декілька VLAN)

2.3.4 Налаштування протоколу STP

За замовченням протокол STP на комутаторах Cisco включений і має параметри, наведені в таблиці 2.3.

Таблиця 2.3 – Параметри протоколу STP за замовченням

Функція	**Налаштування**
Стан	Включений для VLAN 1
Версія	PVST+
Пріоритет комутатора	32768
Пріоритет дерева (на інтерфейсі)	128
Вартість інтерфейсів	1 Гбіт/с – 4; 100 мбіт/с – 19; 10 мбіт/с – 100
Пріоритет VLAN	128
Вартість портів VLAN	1 Гбіт/с – 4; 100 мбіт/с – 19; 10 мбіт/с – 100
Таймери	Привітання – 2 с; Час затримки – 15 с Час старіння – 20 с; Лічильник утримання – 6 BPDU

Якщо адміністратор планує налаштувати окремий комутатор в якості кореневого моста, значення пріоритету моста за замовченням необхідно скоригувати таким чином, щоб воно було нижче значень пріоритету моста для всіх інших комутаторів в мережі. Існує два різних методи налаштування значення пріоритету моста для комутаторів Cisco.

Метод 1

Для забезпечення отримання комутатором найменшого значення пріоритету моста застосовують команду ***spanning-tree vlan*** *vlan-id* ***root primary*** в режимі глобальної настройки. Пріоритет комутатора налаштовується за використанням попередньо певного значення 24576 або найбільшого значення, кратного 4096, яке менше найнижчого значення пріоритету моста, виявленого в мережі.

Якщо потрібний альтернативний кореневої міст, застосовують команду ***spanning-tree vlan*** *vlan-id* ***root secondary*** режиму глобальної настройки. Ця команда задає для комутатора попередньо певне значення пріоритету 28672. Таким чином, альтернативний комутатор стає кореневим мостом в разі відмови основного кореневого моста. При цьому мається на увазі, що для інших комутаторів в мережі визначено значення пріоритету за умовчанням 32768.

На рисунку 2.31 комутатор S1 призначений в якості основного кореневого моста за допомогою команди spanning-tree vlan 1 root primary, а комутатор S2 в якості допоміжного кореневого моста за допомогою команди spanning-tree vlan 1 root secondary.

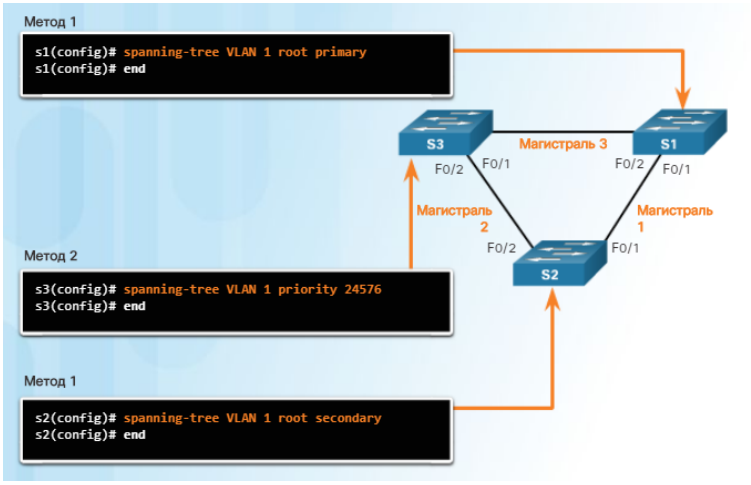


Рисунок 2.31 – Призначення кореневих функцій в мережі

Метод 2

Інший спосіб настройки значення пріоритету моста – використовувати команду режиму глобальної настройки ***spanning-tree vlan*** *vlan-id* ***priority*** *value*. Ця команда забезпечує більш ретельний контроль значення пріоритету моста. Значення пріоритету налаштовується за кроком в 4096 в діапазоні від 0 до 61440.

У прикладі (рисунок 2.31) комутатору S3 присвоєно значення пріоритету моста 24576 за допомогою команди spanning-tree vlan 1 priority 24576.

Щоб перевірити пріоритет моста для комутатора, застосовують команду ***show spanning-tree***. На рисунку 2.32 для комутатора заданий пріоритет 24576 і він призначений в якості кореневого моста для примірника протоколу spanning-tree.

```
S3# show spanning-tree
VLAN0001
  Spanning tree enabled protocol ieee
  Root ID     Priority    24577
              Address     000A.0033.3333
              This bridge is the root
              Hello Time  2 sec Max Age 20 sec Forward Delay 15 sec
  Bridge ID   Priority    24577  (priority  24576  sys-id-ext  1)
              Address     000A.0033.3333
              Hello Time  2 sec Max Age 20 sec Forward Delay 15 sec
              Aging Time  300

Interface     Role    Sts    Cost     Prio.Nbr   Type
------------  -----   ----   -----    ---------- ------
Fa0/1         Desg    FWD    4        128.1      p2p
Fa0/2         Desg    FWD    4        128.2      p2p
```

Рисунок 2.32 – Перегляд налаштувань STP

Обидва методи доцільно застосовувати разом з командами, які додатково до визначення кореневого моста, дозволяють вплинути на вартість шляхів (інтерфейсів) до кореневого моста. Такі команди впливають на визначення інтерфейсів, які блокуються для переведення їх в альтернативний режим роботи. До переліку таких команд віднесені команди, що встановлюють вартість інтерфейсів (наведені в розділі 2.3.2) та пріоритет інтерфейсу, що за замовченням

дорівнює 128, але може буде змінений в межах від 0 до 240 командою в режимі конфігурування інтерфейсу: **spanning-tree vlan 1 port-priority *valye***.

2.3.5 Налаштування інтерфейсів комутатора у режими portfast і BPDU guard

PortFast є функцією Cisco для середовищ PVST+. Якщо порт комутатора налаштований за допомогою функції PortFast, то такий порт відразу переходить зі стану блокування в стан пересилання, минаючи стандартні стани протоколу STP 802.1D (стану прослуховування та завантаження даних). Замість того, щоб очікувати сходження протоколу STP IEEE 802.1D в кожній мережі VLAN, PortFast можна використовувати на портах доступу для забезпечення негайного підключення цих пристроїв до мережі. Портами доступу є порти, підключені до однієї робочої станції або сервера.

Оскільки PortFast призначений для мінімізації часу очікування портами доступу сходження протоколу spanning-tree, цю функцію рекомендується використовувати тільки на портах доступу. Якщо функція PortFast включена на порту, підключеному до іншого комутатора, виникне ризик виникнення петлі протоколу spanning-tree.

У допустимої конфігурації PortFast прийом кадрів BPDU ніколи не допускається, оскільки це вказувало б на те, що до порту підключений інший міст або комутатор, а це може привести до виникнення петлі протоколу spanning-tree. Комутатори Cisco підтримують функцію BPDU guard. Коли функція BPDU guard включена, при отриманні блоку BPDU вона переводить порт в стан errdisabled (error-disabled – відключення через помилку). Це дозволяє виключити порт. Функція BPDU guard забезпечує безпечне реагування на неприпустимі конфігурації.

Технологію Cisco PortFast особливо рекомендується використовувати для DHCP. Без PortFast комп’ютер може відправити запит DHCP до переходу порту в стан пересилання, тобто до наявності умов отримання придатних для використання IP-адреси та інші даних. Оскільки PortFast відразу ж змінює стан на стан пересилання, комп’ютер завжди має умови для отримання придатної для використання IP-адреси (за наявності в мережі правильно налаштованого DHCP-сервера і інших умов для обміну даними між комп’ютером і DHCP-сервером).

Щоб налаштувати на порту комутатора PortFast, виконують команду режиму конфігурації інтерфейсу ***spanning-tree portfast*** на кожному інтерфейсі, для якого потрібно включити PortFast. Команда глобального режиму конфігурації ***spanning-tree portfast default*** використовується для включення PortFast на всіх нетранкових інтерфейсах.

Щоб налаштувати BPDU guard на порту доступу 2-го рівня, використовують команду режиму конфігурації інтерфейсу ***spanning-tree bpduguard enable***. Команда глобального режиму конфігурації ***spanning-tree portfast bpduguard default*** включає BPDU guard на всіх портах з підтримкою PortFast.

Щоб перевірити, чи включений PortFast і BPDU для порту комутатора, використовують команду **show running-config**.

2.4 Маршрутизація в інформаційно-комунікаційних мережах

2.4.1 Задачі та види маршрутизації в інформаційно-комунікаційних мережах

Процес вибору маршруту (шляху) переміщення повідомлень (пакетів) від джерела до адресату називають **маршрутизацією**.

Своєчасність і надійність доставки повідомлень (пакетів), а також завантаження комутаційних центрів (маршрутизаторів) в значній мірі визначаються ефективністю використовуваних алгоритмів маршрутизації.

В мережах з комутацією каналів алгоритм маршрутизації впливає лише на стадії установки з'єднання при виборі шляху (маршруту). В мережах з комутацією пакетів алгоритм маршрутизації може визначати шлях (маршрут) для кожного повідомлення (пакету) окремо, або ж серії повідомлень (пакетів).

По принципу пересування (передачі) повідомлень по маршруту від джерела до адресату виділяють методи **направленої** та **ненаправленої** маршрутизації.

Направлена маршрутизація – постійно забезпечує вибір оптимального маршруту у відповідності до прийнятого критерію передачі. З цією метою використовуються таблиці маршрутів, які формуються відповідними протоколами маршрутизації (динамічні маршрути), та за рахунок статичних маршрутів, які створює адміністратор мережі.

В свою чергу, *по способу формування таблиць маршрутів*, методи направленої маршрутизації діляться на **детерміновані (статичні)** та **адаптивні (динамічні)**.

Детермінований спосіб передбачає, що комутаційні центри мають фіксовані (статичні) таблиці маршрутів і передбачають єдиний шлях передачі для кожного напрямку зв'язку і не використовують який-небудь інший (на даний час) шлях передачі. Цей спосіб підходить для мереж з невеликим навантаженням.

Адаптивний спосіб надає можливість динамічної зміни шляху передачі в кожному комутаційному центрі (маршрутизаторі) в залежності від його завантаження і стану зв'язків.

Інформація, що необхідна для вибору шляху (маршруту) передачі в певний момент часу знаходиться:

– в попередньо складеній таблиці маршрутів комутаційного центру (маршрутизатору);

– відомостях про технічний стан вихідних каналів;

– довжинах черг, що очікують передачі на кожному з каналів.

Вибір оптимального маршруту здійснюється за допомогою розрахунків, що задовольняють умові передачі за визначеним критерієм.

Адаптивний спосіб може бути реалізований у вигляді:

– *локальної адаптивної маршрутизації*, яка для вибору шляху (маршруту) в кожному комутаційному центрі (маршрутизаторі) використовує лише відомості, що доступні на його рівні;

– *розподіленої адаптивної маршрутизації*, при якій маршрутна інформація формується та розподіляється пристроєм, що реалізує функції централізованої маршрутизації.

Ненаправлена маршрутизація – немає систематичного алгоритму просування (передачі) повідомлення (пакету) по мережі. Такий спосіб маршрутизації може використовуватись в умовах відсутності даних про топологію мережі зв'язку. При цьому в комутаційних центрах відсутні таблиці маршрутів. *По способу формування маршрутів* методи ненаправленої маршрутизації діляться на **лавинні** або **випадкові**.

При **лавинному** способі повідомлення із комутаційних центрів (маршрутизаторів) передаються по всім напрямам підключеним до них, окрім каналу (інтерфейсу), по якому ці повідомлення надійшли на нього.

При **випадкової** маршрутизації повідомлення може бути передане на будь-який з вихідних каналів (інтерфейсів). При ненаправлених способах маршрутизації не потрібно знати топологію мережі. Таку маршрутизацію доцільно застосовувати в мережах з невеликим навантаженням і коли необхідна стійка робота мережі при виході з ладу різних її елементів.

В більшості випадків основним критерієм маршрутизації є використання найкоротших шляхів. Задачу про знаходження найкоротшого шляху можна сформулювати в наступному вигляді. Дано зв'язаний граф мережі G, у якому кожній дузі (ребру) приписано вагу, пропорційну до її (його) довжини. Потрібно знайти шлях μ_{st} між заданими вершинами s та *t*, який має мінімально можливу довжину, тобто:

$$l = \sum_{(i,j)\in\mu_{st}} m_{ij} \rightarrow \min(\text{на } \mu')$$

де μ' – множина всіх можливих шляхів з s до *t*.

Процедуру маршрутизації доцільно розрізняти в межах однієї локальної мережі і міжмережну. Для реалізації процесу маршрутизації в локальній мережі важлива функція покладена на протокол ARP (розділ 1.3.4).

Статична маршрутизація має ряд переваг в порівнянні динамічної маршрутизацією, в тому числі:

– Статичні маршрути не оголошуються в мережі, що підвищує їх безпеку;

– Статичні маршрути створюють менше навантаження на пропускну здатність, ніж протоколи динамічної маршрутизації, і для розрахунку і для передачі даних про маршрути не використовуються ресурси центрального процесора.

– Шлях, який використовується статичним маршрутом для відправки даних, буде відомий адміністраторові.

Наступний перехід може бути визначений за IP-адресою, інтерфейсу виходу або по обох параметрах разом (рисунок 2.33). В залежності від того, як визначений наступний перехід, створюється один з трьох наступних типів статичних маршрутів.

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

Рисунок 2.33 – Команда налаштування статичного маршруту

– *Маршрут наступного переходу*. Вказується лише IP-адреса наступного маршрутизатора.

– *Безпосередньо підключений статичний маршрут.* Вказується лише вихідний інтерфейс маршрутизатора.

– *Повністю заданий статичний маршрут.* Визначено і IP-адресу наступного маршрутизатора і вихідний інтерфейс в його бік від даного маршрутизатора.

Існує різновид статичних маршрутів – ***статичний маршрут «за замовчуванням»*** – маршрут, що визначає IP-адресу шлюзу, на який маршрутизатор відправляє всі IP-пакети, для яких у нього немає точного динамічного або статичного маршруту (рисунок 2.36).

```
Router(config)#ip route 0.0.0.0 0.0.0.0 {ip-address | exit-intf}
```

Рисунок 2.34 – Налаштування маршруту «за замовчуванням»

Адміністративна відстань для статичного маршруту дорівнює 1, тобто статичні маршрути мають високий рівень довіри і, як наслідок, завжди будуть у пріоритеті по відношенню до маршрутів сформованих динамічним способом. Проте досить часто постає завдання щодо використання статичного маршруту у якості резерву для динамічних або забезпечення резервування статичних маршрутів іншими статичними. В таких випадках потрібні статичні маршрути з іншими, більшими за 1 значеннями адміністративної відстані. Статичні маршрути, які мають адміністративну відстань більшу за 1 називають ***«плаваючими статичними маршрутами»***. Значення адміністративної відстані такого резервного статичного маршруту обирається більше, ніж адміністративна відстань іншого (основного) статичного маршруту або динамічних маршрутів. Такий статичний маршрут «плаває», тобто існує, але не використовується, коли активний маршрут з меншою адміністративною відстанню (рисунок 2.35).

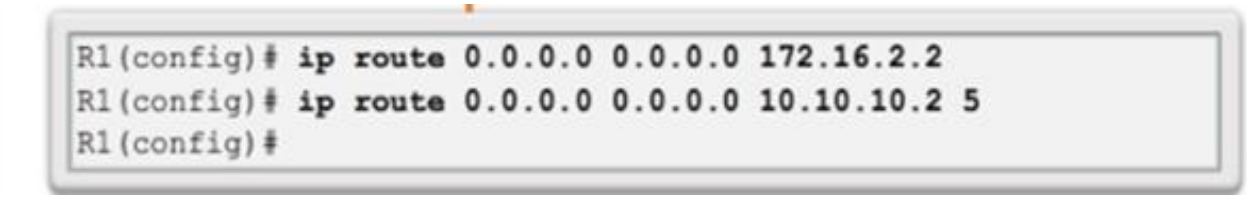


Рисунок 2.35 – Налаштування «плаваючого» маршруту «за замовченням»

2.4.2 Таблиці маршрутизації

Таблиця маршрутизації є для маршрутизатора керівним документом, яким чином діяти з певними пакетами. що надходять на нього, і зберігає інформацію про:

- ***Маршрути із прямим підключенням*** – це маршрути, що прямо надходять із активних інтерфейсів маршрутизатора (підключені до нього).
- ***Вилучені маршрути*** – це маршрути до вилучених мереж, підключених до інших маршрутизаторів.

Таблиця маршрутизації являє собою файл даних в ОЗП, що зберігає інструкції для маршрутизатора –куди конкретно відправити пакет, на який наступний перехід на шляху до пункту призначення, щоб врешті він досягнув конкретного місця призначення.

На маршрутизаторі Cisco IOS команда **show ip route** може бути використана для відображення таблиці IPv4-маршрутизації (рисунок 2.36).

У таблиці маршрутизації можуть бути такі види записів:

– ***локальні маршрути*** **–** додаються, коли інтерфейс настроєний і активний, указує адресу, призначену інтерфейсу маршрутизатора (**L**).

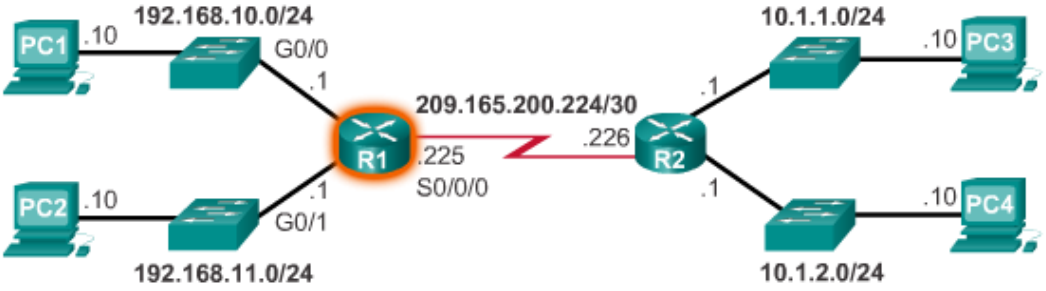
– ***прямі підключення*** **–** визначає мережу із прямим підключенням (**C**).

– ***статичні маршрути*** – додаються, коли маршрут настроєний вручну адміністратором (**S**).

– ***динамічні маршрути*** **–** додаються, коли визначені мережі й реалізуються протоколи маршрутизації, які одержують інформацію про мережу динамічно (**R**, **D**, **O**…).

Кожний запис в таблиці містить наступні відомості (рисунок 2.37):

- **Джерело маршруту** – визначає спосіб одержання інформації про маршрут.
- **Мережа призначення** – визначає адресу мережі, до якої веде цей маршрут.
- **Адміністративна відстань** – визначає надійність джерела інформації про маршрут. Низькі значення вказують на краще джерело інформації про маршрут (таблиця 2.7).
- **Метрика** – визначає умовну «вартість, яку треба сплатити» або «відстань, яку треба подолати» пакету, щоб досягти мережі призначення маршруту. Низькі значення вказують на кращі маршрути.




```
Gateway of last resort is not set
     10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
D       10.1.1.0/24 [90/2170112] via 209.165.200.226, 00:00:05,
        Serial0/0/0
D       10.1.2.0/24 [90/2170112] via 209.165.200.226, 00:00:05,
        Serial0/0/0
     192.168.10.0/24 is variably subnetted, 2 subnets, 3 masks
C       192.168.10.0/24 is directly connected, GigabitEthernet0/0
L       192.168.10.1/32 is directly connected, GigabitEthernet0/0
     192.168.11.0/24 is variably subnetted, 2 subnets, 3 masks
C       192.168.11.0/24 is directly connected, GigabitEthernet0/1
L       192.168.11.1/32 is directly connected, GigabitEthernet0/1
```

Рисунок 2.36 – Таблиця маршрутизації

- **Наступний перехід** – визначає IPv4-адресу наступного (сусіднього) маршрутизатора, на який слід переслати пакет для досягнення мережі призначення маршруту.
- **Часова мітка маршруту** – визначає кількість часу, що пройшов з тих пор, як даний маршрут був створений або підтверджений.
- **Вихідний інтерфейс** – визначає вихідний інтерфейс для відправлення пакету до наступного маршрутизатора (переходу).

Таблиця 2.7 – Припустимі значення адміністративної відстані

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

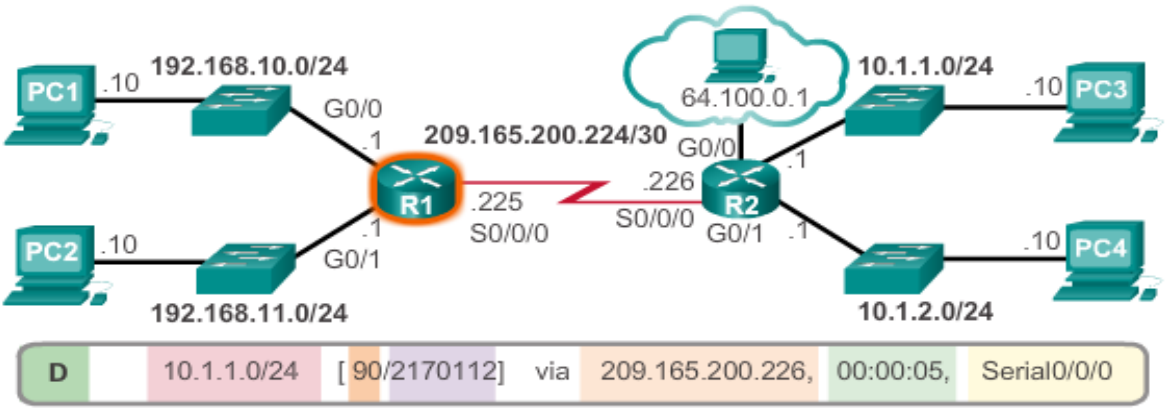


Рисунок 2.37 – Записи в таблиці маршрутизації

2.4.3 Огляд основних протоколів динамічної маршрутизації

Завданням протоколів динамічної маршрутизації є забезпечення обміну між всіма маршрутизаторами інформацією, яка надасть можливість розрахувати (визначити) маршрути до будь-якої мережі, та підтримання цієї інформації в актуальному стані при змінах в мережі. Водночас самі протоколи маршрутизації не забезпечують безпосередньо передачу інформації, вони лише формують маршрути, якими протокол IP має здійснювати передавання.

Протокол маршрутизації RIP

RIP (Routing Information Protocol) – один з перших протоколів внутрішньої маршрутизації, що застосовувався в Інтернеті ще з 1982 р. Перша версія протоколу RIP описана в специфікації RFC 1058, а друга (RIP ver. 2) – в специфікації RFC 2453. В цьому протоколі метрикою є кількість пересилань між вузлами (*hops*). Ця метрика не забезпечує облік пропускної здатності, надійності та завантаженості трактів передачі, а також характеристик трафіка користувачів.

RIP – це протокол дистанційно-векторної маршрутизації, що ґрунтується на використанні вектору відстаней (*Distance Vector*). Він не дозволяє забезпечити функціонування широкомасштабних мереж через обмеженість числа пересилань (*hops*) до 15.

Робота протоколу RIP заснована на широкомовному розсиланні повідомлень про коректування маршрутів. Періодичність розсилання оновлень – 30 с, розсилка відбувається на групову IP-адресу 224.0.0.9. Розсилаються повні копії таблиць маршрутизації кожного маршрутизатора незалежно від того, змінені вони чи ні.

В наслідок своєї недосконалості, протокол RIP отримав найбільше (найгірше з усіх протоколів динамічної маршрутизації) значення адміністративної відстані – 120. Водночас RIP досить простий у налаштуваннях і знаходить своє застосування у невеликих мережах.

Протокол маршрутизації OSPF

Протокол OSPF (Open Shortest Path First) належить до класу протоколів стану каналів (Link State Protocol). Дослівний переклад – першим обирається найкоротший шлях (використовує алгоритм Дейкстри).

В OSPF підтримуються метрика, яка враховує пропускну здатність каналів, що складають маршрут. Метрика каналу визначається як кількість секунд необхідних для передачі 100 Мбіт інформації (1 – мінімальне значення, всі значення менше 1 округлюються до 1). Завдяки метриці адміністративна відстань OSPF дорівнює 110.

Повідомлення OSPF інкапсулюється прямо в IP пакет (поле даних), тобто протоколи транспортного рівня не використовуються.

Протокол OSPF може обслуговувати мережі будь-якого розміру, для зменшення вимог до ресурсів маршрутизаторів, протокол передбачає розділення мережі на окремі зони (area). Обов'язкова наявність зони 0 (магістральної), всі інші зони (за наявності) мають прив'язуватися до магістральної. При цьому лише маршрутизатори зони 0 мають зберігати інформацію про всі маршрути мережі, маршрутизаторам в інших зонах достатньо маршрутів всередині зони, а всі інші напрями буде обробляти один маршрутизатор зони, який називають прикордонним. Проте кожний маршрутизатор OSPF зберігає в пам'яті повну структуру мережі (принаймні зони, якщо це звичайний маршрутизатор не із зони 0), що дозволяє їм швидко відновлювати маршрути (без взаємодії з іншими маршрутизаторами) у випадку їх зникнення.

Якщо маршрутизатори OSPF з'єднані в єдиному широкомовному домені, то для зменшення службової (маршрутної) інформації, що там пересилається, між ними відбувається перерозподіл функцій. Один з маршрутизаторів стає обраним (за максимальним значенням Router ID), інші маршрутизатори направляють маршрутні повідомлення на групову IP-адресу 224.0.0.6, яку прослуховує лише обраний маршрутизатор. Після обробки отриманої інформації, обраний маршрутизатор розповсюджує її для всіх маршрутизаторів домену на групову IP-адресу 224.0.0.5.

Маршрутизатори OSPF використовують таки типи повідомлень:

1. **Hello** – для виявлення сусідніх маршрутизаторів з OSPF, встановлення та підтримки відносин суміжності (кожні 10 с).
2. **Database Description** – опис вмісту бази даних Link-State маршрутизатора (при початку взаємодії з сусідами).
3. **Link-State Request** – запит конкретної інформації в сусіда після обробки його Database Description.
4. **Link-State Update** – оновлення даних про стан каналів надсилається у відповідь на Link-State Request або коли в мережі відбуваються зміни.
5. **Link-State Acknowledgment** – підтвердження отримання Link-State Update та Database Description.

Протокол пограничного шлюзу BGP

Завдання BGP – побудова маршрутів між мережами, що належать різним автономним системам (АС). АС – це відокремлений сегмент мережі під окремим

адмініструванням, які мають свій номер (виданий міжнародною адміністрацією), і належать операторам (провайдерам) першого (глобального) або другого (регіонального) рівня.

Протокол BGP на основі інформації, отриманої від різних маршрутизаторів, вибудовує граф АС з усіма зв'язками між вузлами. Такий граф іноді називають деревом. Протокол є основним для маршрутизації в мережі Internet. Для протоколу BGP граф мережі Internet складається з з'єднаних автономних систем, де кожній автономній системі відповідає свій унікальний номер. З'єднання між двома автономними системами формує шлях, а інформація про сукупність шляхів від одного вузла в автономній системі до вузла в іншій автономній системі становить маршрут. Протокол BGP активно використовує інформацію про маршрути до заданого пункту призначення, що дозволяє уникнути утворення петель маршрутизації між доменами.

В основі протоколу BGP лежать оголошення про маршрути. Оголошення про маршрути посилає один рівноправний BGP-вузол іншому рівноправному BGP-вузлу для з'єднання «точка-точка». Оголошення складається з адреси мережі і набору атрибутів, асоційованих з маршрутом до цієї мережі. Двома найбільш важливими атрибутами є опис маршруту (список усіх автономних систем на шляху до зазначеної мережі) і ідентифікатор наступного маршрутизатора на шляху до зазначеної мережі призначення. Таким чином, формується ієрархічна схема маршрутизації, що зв'язує різні вузли й автономні системи в єдину мережу.

Очевидно, що BGP-маршрутизатори, що перебувають в одній АС, також повинні обмінюватися між собою маршрутною інформацією. Це необхідно для погодженого відбору зовнішніх маршрутів відповідно до політики даною АС і для передачі транзитних маршрутів через автономну систему. Такий обмін проводиться також по протоколу BGP, який у цьому випадку часто називається *IBGP* (*Internal BGP-внутрішній протокол* BGP), відповідно, протокол обміну маршрутами між маршрутизаторами різних АС називають зовнішнім протоколом BGP і позначають *EBGP* (*external BGP*).

Особливістю протоколу BGP є пряма вказівка адміністратора на встановлення сусідських відносин з іншими маршрутизаторами, і, як наслідок, відсутність групової розсилки маршрутних повідомлень. Проте всі повідомлення BGP передаються з використанням послуг транспортного протоколу TCP з портом 179.

Метрика протоколу досить складна і включає 7 різних атрибутів, проте деякі виробники мережевого обладнання збільшують їх до 9 і, навіть, 13. При цьому адміністратор має повноваження впливати на більшість з цих атрибутів. Така складна і багатофакторна метрика позитивно впливає на якість і надійність вибору маршрутів, що відображається на адміністративній відстані протоколу BGP, яка дорівнює 70.

Маршрутизатори BGP використовують таки типи повідомлень:

1. **OPEN** – встановлення сесії і узгодження параметрів з сусідами (відправляється один раз на початку роботи).

2. **KEEPALIVE** – підтвердження активного з'єднання з сусідом (відправляється кожні 60 с)
3. **UPDATE** – передача маршрутної інформації: містить нові маршрути з їхніми атрибутами або список відкликаних маршрутів, які більше не доступні (відправляється за необхідності – у випадку наявності змін).
4. **NOTIFICATION** – повідомлення про помилку та про закриття сесії TCP (відправляється лише у разі виникнення помилки).

Протокол маршрутизації EIGRP

Протокол EIGRP (*Enhanced Interior Gateway Routing Protocol*) – це пропрієтарний (Cisco) протокол маршрутизації, що підтримується виключно обладнанням Cisco. *EIGRP* – дистанційно-векторний протокол маршрутизації, що був оптимізований для зменшення нестабільності маршрутизації після змін топології мережі, уникнення проблеми зациклення маршрутів та більш ефективного і економного використання потужностей маршрутизатора. Роутери, що підтримують протокол *EIGRP* також підтримують і попередню версію (*IGRP*) та перетворюють маршрутну інформацію для *IGRP*-сусідів з 32-бітної метрики *EIGRP* у 24-бітну метрику стандарту *IGRP*. Алгоритм визначення маршруту базується на алгоритмі Дейкстри пошуку в глибину на графі. *EIGRP* обчислює і враховує 5 параметрів для кожної ділянки маршруту між вузлами мережі:

- *Total Delay* – загальна затримка передачі (з точністю до мікросекунди);
- *Minimum Bandwidth* – мінімальна пропускна спроможність (в кбіт/с);
- *Reliability* – надійнсть (оцінка від 1 до 255; 255 найбільш надійно);
- *Load* – завантаженість (оцінка від 1 до 255; 255 найбільш завантажено);
- *Maximum Transmission Unit* (*MTU*) – максимальний розмір блоку, що можливо передати по ділянці маршруту.

Формула обчислення оцінки ділянки:

$$\left(K_1 * BW + \left(\frac{K_2 * BW}{256 - Load}\right) + K_3 * Delay\right) * \left(\frac{K_5}{reliability + K_4}\right) * 256 \quad ,$$

де K_1, K_2, K_3, K_4 і K_5 – коефіцієнти, що визначаються вручну користувачем для зміни пріоритетів обчислення оцінок. За замовчанням всі змінні дорівнюють 1.

На маршрутизаторах *Cisco Interface Bandwidth* (пропускна спроможність інтерфейсу) є налаштованим параметром, що задається користувачем. Аналогічно *Interface Delay* (затримка інтерфейсу) є конфігурованим статичним параметром.

За своїм алгоритмом протокол *EIGRP* дуже подібний до протоколу *OSPF* за виключенням всього, що стосується зонування. Адміністративна відстань протоколу *EIGRP*, завдяки 5-тифакторній метриці, краще за OSPF і складає 90, проте відсутність зонування накладає певні обмеження на розмірність мережі – протокол рекомендований для мереж середнього розміру.

EIGRP використовує 5 типів повідомлень:

1. **Hello** – маршрутизатори використовують для виявлення сусідів. Пакети відправляються multicast на адресу 224.0.0.10 (один раз на 5 с) і не вимагають підтвердження про отримання.

2. **Update** – містить інформацію про зміну маршрутів. Вони відправляються тільки до маршрутизаторів, яких стосується оновлення. Ці пакети можуть бути відправлені конкретному маршрутизатору (unicast) або групі маршрутизаторів (*multicast*). Отримання update-пакета підтверджується відправкою *ACK*. Спочатку відправляються повні оновлення, в які включені всі маршрути. Після того як обмін маршрутами завершився, оновлення відправляються тільки якщо є зміни маршрутів.
3. **Query** – відправляється сусідам, коли маршрутизатор не має резервного маршрутизатора (можливого наступника) для певного маршруту, з метою запиту такої інформації в них. Зазвичай *query*-пакети відправляються груповою розсилкою, але можуть й індивідуальною. Отримання *query*-пакета підтверджується відправкою *ACK* одержувачем пакета.
4. **Reply** – відправляється у відповідь на *query*-пакет. *Reply*-пакети відправляються *unicast* до того, хто відправив *query*-пакет. Отримання *reply*-пакета підтверджується відправкою *ACK*.
5. **ACK** – пакет, який підтверджує отримання пакетів *update*, *query*, *reply*. *ACK*-пакети відправляються *unicast* і містять в собі *acknowledgment number*. Фактично це *hello*-пакети, які не передають даних.

Головною особливістю та перевагою протоколу *EIGRP* є висока ступінь збіжності під час оновлення топології, яка реалізується завдяки застосуванню ***алгоритму дифузного поновлення*** (*DUAL*) *DUAL* – це алгоритм оновлення, вибору і супроводу маршрутів, він виконує всі розрахунки маршрутів *EIGRP* і відповідає за вибір *наступника* і *можливого наступника*, а також передачу *Hello*-повідомлень, оновлень, запитів, відповідей і підтверджень.

Наступник – це маршрутизатор, що знаходиться за адресою наступного транзитного переходу, який має кращий маршрут до одержувача в порівнянні з локальним маршрутизатором. Кожен маршрут, перерахований в таблиці маршрутизації, має, щонайменше, по одному наступнику. Якщо вибрано кілька наступників, маршрутизатор *EIGRP* розподіляє навантаження між наступниками. Список наступників знаходиться і в таблиці маршрутизації, і в таблиці топології.

Можливий наступник – це маршрутизатор, що знаходиться за адресою наступного транзитного переходу, який не має найкращого маршруту до одержувача, але все ж ближче до одержувача, ніж локальний маршрутизатор. Можливі наступники можуть розглядатися як альтернативні маршрути на той випадок, якщо маршрут через поточного наступника стане недійсним. Можливі наступники перераховуються тільки в таблиці топології.

Крім того існує протокол динамічної маршрутизації **IS-IS**, але він не знаходить наразі широкого застосування.

2.4.4 Налаштування протоколів маршрутизації

Налаштування всіх протоколів маршрутизації розпочинається з їх включення командою:

router [назва та (за виключенням *RIP*) ідентифікатор протоколу**]**, дана команда вводиться в режимі глобального конфігурування *CiscoIOS* й одночасно переводить її в режим часткового конфігурування протоколу.

Далі прописуються мережі, які даний маршрутизатор буде анонсувати. До них відносять всі мережі, що напряму підключені, до даного маршрутизатора:

network [опис мережі**].**

Доцільно також оголосити пасивні інтерфейси маршрутизатора, тобто такі інтерфейси, до яких напряму підключена мережа, яку треба анонсувати, але в якій немає маршрутизаторів, тобто до неї не варто відправляти маршрутні повідомлення:

passive-interface [назва та номер інтерфейсу].

Можливі інші додаткові команди, наприклад:

version 2 – для включення версії 2 протоколу *RIP*;

no auto-summary – для включення режиму роботи з безкласовими мережами, доступна в кількох протоколах;

router-id [числове значення ідентифікатора] – для призначення ідентифікатора маршрутизатора в протоколі *OSPF*;

neighbor [IP-адреса інтерфейсу сусіднього маршрутизатору] **remote-as** [числове значення ідентифікатора автономної системи] – для визначення маршрутизаторів-сусідів в протоколі BGP.

2.5 Протоколи управління інформаційно-комунікаційними мережами та порядок їх застосування

2.5.1 Відомості про використання протоколу SNMP

SNMP (*Simple Network Management Protocol*, простий протокол керування мережею) – це протокол керування мережами на основі архітектури TCP/IP, який забезпечує керування й контроль за пристроями й застосунками в інформаційно-комунікаційній мережі шляхом обміну керівною інформацією між агентами, що розташовуються на мережних пристроях, і менеджерами, розташованими на станціях керування. SNMP визначає мережу як сукупність мережних керівних станцій й елементів мережі (головні машини, шлюзи, маршрутизатори, термінальні сервери), які спільно забезпечують адміністративні зв’язки між мережними керівними станціями й мережними агентами.

Зазвичай при використанні SNMP присутні керовані та керівні системи. До складу керованої системи входить компонент, який називається *агентом*, який відправляє звіти керівній системі. По суті SNMP агенти це спеціалізоване програмне забезпечення, яке постійно зберігає (оновлює) управлінську інформацію. Вона зберігається у вигляді множини змінних, таких як «вільна пам’ять», «ім’я системи», «кількість процесів, що працюють», «стан інтерфейсу» тощо).

Змінні, доступні через SNMP, організовані в ієрархії. Ці ієрархії та інші метадані (такі як тип і опис змінної) описуються базами інформації керування (Management Information Bases, MIBs). MIB використовують ієрархічний адресний простір імен, що містить унікальний ідентифікатор об'єкта (object identifier, OID). Тобто, кожен унікальний OID визначає деяку змінну, яка може бути прочитана чи встановлена через SNMP.

Керівна система може отримати достовірну інформацію через операції запиту GET, GETNEXT і GETBULK. Агент може самостійно без запиту надсилати дані, використовуючи операції інформування TRAP або INFORM. Керівні системи можуть також відправляти конфігураційні оновлення або контрольні запити, використовуючи операцію SET для безпосереднього управління системою. Операції конфігурування та управління використовуються тільки тоді, коли потрібні зміни, наприклад у мережній інфраструктурі. Операції моніторингу зазвичай виконуються на регулярній основі.

Ієрархія MIB може бути зображена як «дерево» з безіменним коренем, рівні якого приписані різними організаціями. На найвищому рівні MIB – OID-и належать різним організаціям, що займаються стандартизацією, в той час як на нижчих рівнях OID-и виділяються асоційованими організаціями. Ця модель забезпечує управління на всіх шарах мережної моделі OSI, адже MIB можуть бути визначені для будь-яких типів даних і операцій. Ідентифікатор об'єкта (OID) унікально ідентифікує керований об'єкт в ієрархії MIB (рисунок 2.38).

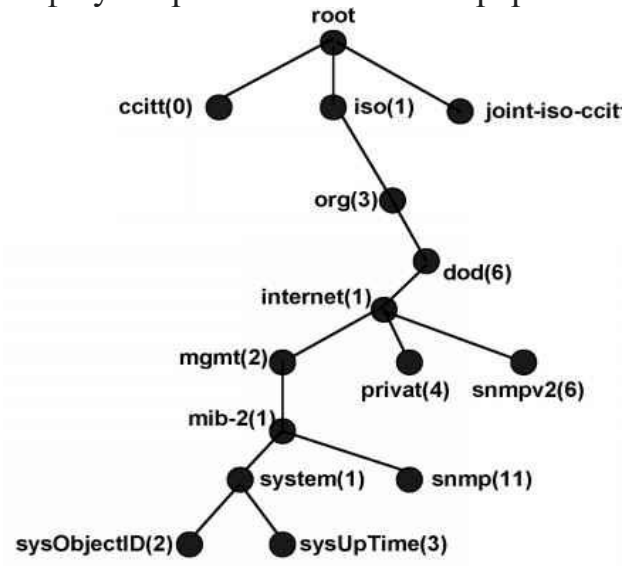


Рисунок 2.38 – Фрагмент ієрархії MIB

Існують три версії протоколу SNMP:

- SNMPv1 *(Version 1)* – перша версія протоколу, розроблена у 1980-х роках. Версія найпростіша в налаштуванні та підтримці. Для аутентифікації використовується лише community string (текстовий пароль, що передається у відкритому вигляді). Не підтримує роботу з великими обсягами даних (обмежені 32-бітні лічильники). Визначає чотири типи стандартних SNMP-операцій для керування об'єктами:

- Операція Get [одержати] застосовується щоб повернути (одержати) значення атрибутів керованого об’єкта із групи MIB.
- Операція Getnext [одержати наступний] існує, щоб повернути ім’я (і значення атрибутів) наступного один по одному керованого об’єкта в MIB.
- Операція Set [установити] застосовується, щоб установити на керованих об’єктах значення атрибутів (змінити зміст комірки таблиці MIB).
- Операція Trap [переривання] використовується мережними обладнаннями асинхронно; за допомогою переривання, зупинивши виконання інших програм керування, елементи мережі можуть самостійно, без спеціального запиту, повідомити менеджера мережі про виниклі відмови, перевантаження й т.п.

- SNMPv2c *(Version 2 Community)* – еволюція першої версії, що з'явилася у 1990-х роках. Найбільш поширена версія сьогодні. Підтримує 64-бітні лічильники для високошвидкісних інтерфейсів. Більшість систем Microsoft WinSNMP та мережевого обладнання підтримують v2c за замовчуванням. Для аутентифікації так само використовується лише community string (текстовий пароль, що передається у відкритому вигляді). Додано нові типи операцій, а саме:
  - Операція Getbulk [одержати перелік] використовується для добування великої кількості значень із таблиць, а не одиничних значень атрибутів.
  - Операція Inform [інформувати] дозволяє однієї NMS виконувати операцію Trap на інший NMS і, відповідно, одержувати відповідь на асинхронний запит.
  - Операція Report [рапорт] дозволяє агенту повідомити про стан керованого ресурсу; повідомлення видається без запиту.
- SNMPv3 *(Version 3)* – сучасна версія, головний акцент якої зроблено на безпеці. Введена модель безпеки на основі користувачів (USM) та контроль доступу (VACM). При аутентифікації відбувається перевірка справжності користувача за алгоритмами MD5 або SHA. Реалізований захист пакетів даних від перегляду сторонніми шляхом шифрування (DES, AES). На прийомі відбувається перевірка цілісності пакетів.

Більшість програм-менеджерів в SNMP забезпечують наступні функції керування:

– *Функції збору інформації про несправності* (alarm polling functions). SNMP-менеджери забезпечують можливість установити пороги чутливості (thresholds) на керованих об’єктах (наприклад, максимально припустиме число помилок), і вчасно видавати аварійне повідомлення, коли ці пороги перевищені. Реалізація даної функції дозволяє постійно контролювати технічну справність мережі і її окремих елементів.

– *Функції контролю тренда* (trend monitoring functions). Протягом певного часу SNMP-менеджери забезпечують можливість безперервного спостереження за деякими значеннями атрибутів керованих об’єктів; ці об’єкти

й значення атрибутів зафіксовані в MIB. Періодично проводиться зчитування значень атрибутів, що дозволяє оцінити робочі характеристики мережі в динаміці тобто побудувати тренд мережі за тією або іншою ознакою.

– *Функції переривання при прийманні* (trap reception functions). SNMP-менеджери забезпечують можливість приймання й фільтрації SNMP-переривань, які видаються мережними обладнаннями. SNMP-переривання є важливою частиною протоколу SNMP; переривання дозволяють мережним обладнанням самостійно, не чекаючи запиту, повідомляти про проблеми, відмови й т.п. Припустимі типи переривань звичайно реєструються адміністратором мережі. Переривання управляють повідомленнями про мережні події. Переривання видаються в асинхронному режимі.

У протоколі SNMP можна виділити наступні основні стандартизовані елементи:

1. *Стандартний формат повідомлення* (standard message format), який визначається форматом повідомлення UDP.

2. *Стандартний набір керованих об'єктів* (standard set of managed objects) являє собою набір стандартних об'єктів і значень (values) їх атрибутів в MIB. Ці значення можна одержати у відповідь на запити станції керування. Значення, одержуване у відповідь на запит (значення, що вертається) дозволяє зробити висновок про стан керованого об'єкта. Стандартний набір включає величини для контролю протоколів TCP, IP, UDP, мережних інтерфейсів і обладнання в цілому. Кожна величина асоційована з об'єктом, що мають унікальний об'єктний ідентифікатор.

3. *Стандартний спосіб додавання об'єктів* (standard way of adding objects). Наявність цього елемента – одна із причин того, чому протокол SNMP став широко відомим і придбав статус de-facto промислового стандарту керування. Цей метод дозволяє фірмам-виробникам розширювати стандартний набір керованих об'єктів за допомогою специфікації нових керованих об'єктів і додавання їх в MIB.

Усі вищезгадані типи операцій і запитів передаються у вигляді PDU певного формату, якими обмінюються обладнання, що підтримують протокол SNMP (рисунок 2.39 і таблиця 2.8).

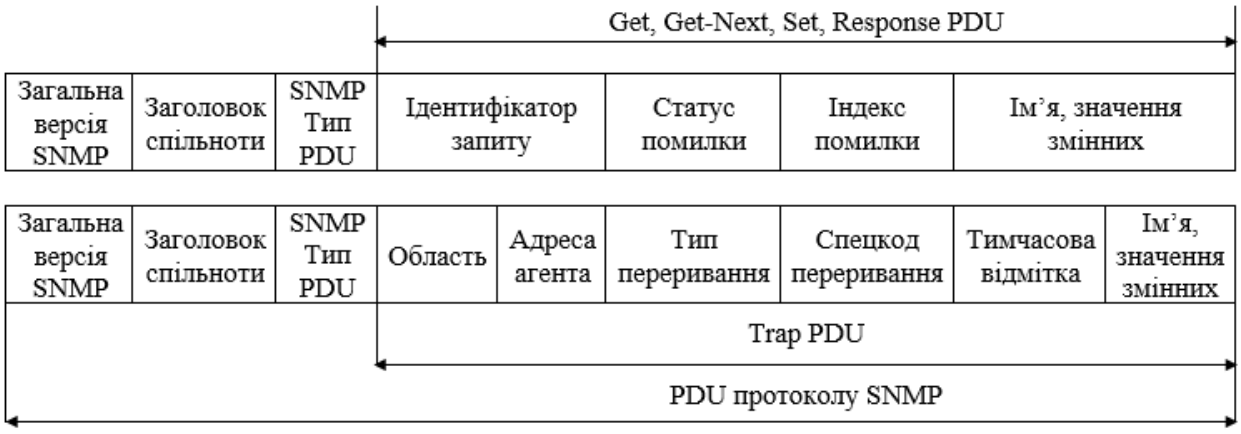


Рисунок 2.39 – Формати PDU в SNMP (vl і v2)

Таблиця 2.8 – Призначення полів PDU протоколу SNMP

Найменування поля	**Призначення**
Версія SNMP	Номер версії протоколу SNMP
Співтовариство	Пароль дозволу обміну між агентом і менеджером
Назва й тип PDU	Get (тип 0), Set (тип 2), Getnext (тип 1), Responce (тип 3)
Ідентифікатор запиту (Request-id).	Встановлює відповідність між командами й відповідями на команди
Статус помилки (Error-status)	Указує помилку і її тип
Індекс помилки (Error-index)	Встановлює зв’язок між помилкою й конкретною реалізацією керованого об’єкта
Ім’я, значення змінних (Variable bindings)	Дані SNMP PDU, які містять значення атрибутів, які записані за допомогою змінних і поточних значень змінних.
Назва й тип PDU	Getrequest, Setrequest, Trap, Informrequest, Response
Співтовариство, область (Enterprise)	Тип об’єкта, що генерує дане переривання
Адреса агента (Agent address)	Містить мережну адресу об’єкта, що генерує дане переривання
Загальний тип переривання (Generic trap type)	Містить загальні дані про тип переривання.
Спецкод переривання (Specific trap code)	Містить специфічний код переривання.
Часова оцінка (Time stamp)	Містить величину часу, що пройшов між останньою повторною ініціалізацією мережі й генерацією даного переривання.
Ім’я, значення змінних (Variable bindings)	Містить перелік значень атрибутів у вигляді змінних і їх значень із інформацією про переривання.

Використовуючи перераховані операції (команди) можна сформувати відповідні примітиви запитів для обміну між менеджером і агентом. Послідовність обміну перерахованими запитами (повідомленнями) представлена на рисунку 2.40.

Протокол SNMP використовує два типи транспортних портів. Порт 161 (UDP) застосовується агентом (комутатором, сервером, принтером) для прослуховування запитів від менеджера. Менеджер надсилає команди GET (читання даних) або SET (зміна налаштувань) саме на цей порт.

Порт 162 (UDP) використовується менеджером для отримання непередбачених сповіщень від пристроїв, відомих як Traps або Informs.

Для забезпечення використання протоколу SNMP шляхом подання команд з командного рядка (без програми менеджера) потребує встановлення набір інструментів, які входять до складу пакета Net-SNMP, наприклад доступний на сайті http://net-snmp.sourceforge.net.

За замовчуванням Net-SNMP встановлюється в папку *C:*/usr, а файли MIB зберігаються у каталозі /usr/local/ share/snmp/mibs. У процесі установки Net-SNMP поміщає в цей каталог кілька десятків файлів MIB, у тому числі UCD MIB (раніше Net-SNMP називався UCD-SNMP) і RFC 1313 MIB (MIB-II). Net-SNMP використовує файли MIB для перетворень між цифровими ідентифікаторами об'єктів і їх текстовими описаннями. Крім того, файли MIB надають інструментам доступ до інформації про кожний об'єкт (синтаксис, дозволений режим доступу, опис і т.д.).

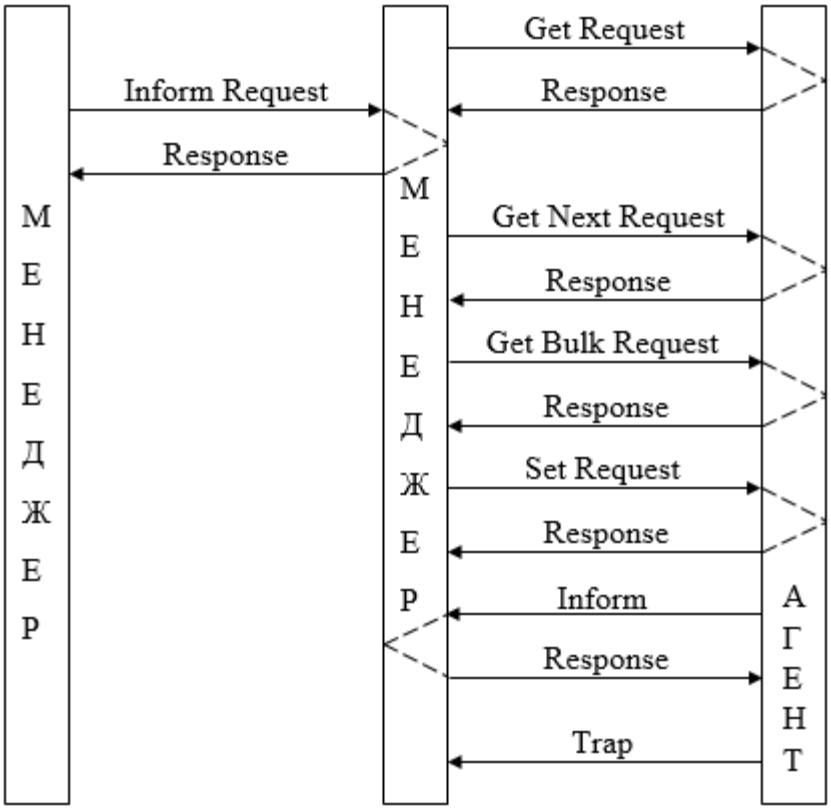


Рисунок 2.40 – Обмін запитами й відповідями (response) в SNMPv2

Команда snmpwalk виконує операцію getnext: ***snmpwalk -v2c -c public cisco.oreilly.com .1.3.6.1.4.1.9***. Команда snmpbulkwalk використовує послідовність команд getbulk для одержання великих частин MIB: ***snmpbulkwalk -v2c параметри ім'я_вузла ID об'єкту***. Команда snmpset використовується для зміни або установки значення об'єкта MIB. Команда виглядає таким чином: ***snmpset параметри ім'я_вузла ID об'єкту тип.значення***. Тип – це позначення з одного символу, що вказує тип даних для встановлюваного об'єкту (у таблиці 2.9 наведені припустимі типи). Для відправлення пастки використовують команду snmptrap. Її синтаксис наступний: ***snmptrap -v2c параметри ім'я_вузла час_роботи OID ловушки ID об'єкту***

тип значення. До складу пакета Net-SNMP входить зручний інструмент з назвою snmptranslate, який переводить цифровий ідентифікатор об'єкта в текстовий і навпаки. У загальному випадку він може використовуватися для пошуку інформації у файлах MIB. От його синтаксис: ***snmptranslate параметри ID об'єкту***.

Таблиця 2.9 – Типи об'єктів snmpset

Позначення	Тип
a	IP-адреса
b	Бітове поле
d	Десятковий рядок
D	Подвійне ціле
F	Із плаваючою крапкою
i	Ціле
I	64-бітне ціле зі знаком (Signed int64)
o	Ідентифікатор об'єкта
s	Рядок
t	Час (Time ticks)
u	Ціле без знаку
U	64-бітне ціле без знаку (Unsigned int64)
x	Шістнадцятковий рядок

У таблиці 2.10 наведені деякі з найбільш корисних параметрів, загальних для всіх команд Net-SNMP.

Для налаштування SNMP на пристроях Cisco (маршрутизаторах та комутаторах)потрібно виконати такі основні кроки налаштування.

Крок 1. Для базової роботи необхідно створити рядок спільноти (Community String) з правами «тільки для читання» (RO) або «читання та запис» (RW). Community String – це текстовий пароль для доступу до даних (тільки для SNMP v1/v2c).

router(config)# snmp-server community *private* RW
router(config)# snmp-server community *public* RO

Крок 2. Налаштування контактних даних, які допоможуть адміністратору ідентифікувати пристрій у системі моніторингу.

router(config)# snmp-server location Delta Building - 1st Floor
router(config)# snmp-server contact J Jones

Крок 3. Налаштування сповіщень (SNMP Traps), які пристрій надсилає на сервер самостійно при виникненні певної події.

router(config)# snmp-server enable traps
router(config)# snmp-server host 10.123.135.25 version 2c public

Для перевірки працездатності SNMP можна використати наступні команди:

show snmp – загальна статистика та стан налаштувань;
show snmp community – перегляд налаштованих рядків спільноти.

Таблиця 2.10 – Список основних параметрів командного рядка Net-SNMP

Параметр	**Опис**
-m	Указує, які модулі MIB команда повинна завантажити. Якщо ви прагнете, щоб команда розбирала файл MIB конкретного виробника, треба запустити команду з параметром -m ALL. Аргумент ALL указує команді прочитати всі файли MIB у каталозі. Установка для змінного середовища $MIBS значення ALL дозволяє досягти того ж результату. Якщо ви не прагнете, щоб команда читала всі файли MIB, то можете вказати після параметра -m розділений двокрапками список файлів MIB, які потрібно проаналізувати
-M	Дозволяє вказати список розділених двокрапками каталогів для пошуку файлів MIB. Цей параметр корисний, якщо ви не прагнете копіювати файли MIB у місце розташування MIB за замовчуванням. Установка змінної оболонки $MIBDIRS дозволяє досягти того ж результату
-IR	Виконує пошук ідентифікатора об'єкта з довільною вибіркою в базі даних MIB. За замовчуванням команди припускають, що ви вказуєте ідентифікатор об'єкта відносно *.iso.org. dod.internet.mgmt.mib-2*. На практиці цей параметр дозволяє уникнути написання довгих ідентифікаторів об'єктів, які не входять у субдерево *mib-2*. Наприклад, в MIB Cisco є група об'єктів за назвою *lcpu.* Якщо ви скористаєтеся параметром -IR, то зможете одержувати об'єкти цієї групи без уведення повного ідентифікатора. Досить наступної команди: snmpget -IR *ім'я_вузла community* lcpu.2
-On	Виводить ідентифікатори об'єктів у цифровому виді (наприклад, *.1.3.6.1.2.1.1.3.0)*
-Of	Виводить повні текстові ідентифікатори об'єктів
-Os	Виводить повний ідентифікатор об'єкта (тобто починаючи с.1)
-OS	Відображає тільки останню частину ідентифікатора об'єкта в символічному виді (наприклад, *sysuptime.0)*
-v	Указує, яку версію SNMP використовувати: -v1, -v2c і -v3 для SNMPvl, SNMPv2 і SNMPv3 відповідно
-h	Виводить довідкову інформацію
-c	Указує рядок community для SNMPvl або SNMPv2c

2.5.2 Протокол збору керуючої інформації Syslog

Syslog – це стандартний протокол і система збору текстових повідомлень про події, що відбуваються в операційній системі, програмах або мережевому обладнанні.

Фактично, це загальний журнал (log, лог), куди різні частини системи записують звіти про свою роботу: від інформації про успішний вхід користувача до критичних помилок обладнання. Ця інформація може використовуватися адміністратором для моніторингу й налагодження системи.

Кожне повідомлення містить три ключові частини:

– Facility (Джерело): Хто надіслав повідомлення (ядро, поштова служба, система безпеки, користувацька програма тощо).
– Severity (Рівень важливості): Наскільки критична подія. Всього існує 8 рівнів: від 0 (найбільш критична подія) до 7 (налагоджувальна інформація), які наведені в таблиці 2.11.
– Content (Текст): Сама суть події, час її виникнення та назва пристрою/процесу.

За замовчуванням формат повідомлень Syslog у програмному забезпеченні Cisco IOS виглядає таким чином:

seq no: timestamp: %facility-severity-MNEMONIC: description

Приклад вихідних даних про зміну стану каналу Etherchannel комутатора Cisco на активне буде виглядати наступним чином:

00:00:46: %LINK-3-UPDOWN: Interface Port-channel1, changed state to up

У цьому прикладі об’єктом є LINK, призначений рівень серйозності 3, у якості КОРОТКОГО КОДУ виступає UPDOWN.

Таблиця 2.11 – Рівні важливості повіжомлень Syslog

Назва рівня серйозності	**Рівень серйозності**	**Пояснення**
Надзвичайна ситуація	0 Рівень	Заборонено використовувати
Попередження	1 Рівень	Потребує прийняти негайні міри
Критичний	2 Рівень	Критичний стан
Помилка	3 Рівень	Стан помилки
Попередження	4 Рівень	Стан попередження
Повідомлення	5 Рівень	Нормальне, потребує контролю
Інформаційний	6 Рівень	Інформаційне повідомлення
Налагодження	7 Рівень	Повідомлення про налагодження

Адміністратор може здійснювати вибір типу інформації, збір якої буде здійснюватися.

Повідомлення Syslog можуть відправлятися по мережі на зовнішній сервер Syslog. Ці повідомлення можна прочитати без необхідності доступу до самого обладнання. Крім того, повідомлення Syslog можуть відправлятися у внутрішній буфер. Повідомлення, відправлені у внутрішній буфер, можна переглядати тільки через інтерфейс командного рядка обладнання. На рисунку 2.41 наведені всі можливі варіанти спрямування для повідомлень Syslog.

Адміністратор може вказати, які типи системних повідомлень (за рівнем важливості) будуть відправлятися в різні місця призначення. Наприклад, можна настроїти обладнання, щоб усі системні повідомлення відправлялися на зовнішній сервер Syslog. Однак повідомлення рівня debug будуть пересилатися у внутрішній буфер і будуть доступні тільки адміністраторові через інтерфейс командного рядка.

Можна видалено спостерігати за системними повідомленнями шляхом перегляду журналів на сервері Syslog або шляхом доступу до обладнання по протоколах Telnet, SSH або через порт консолі.

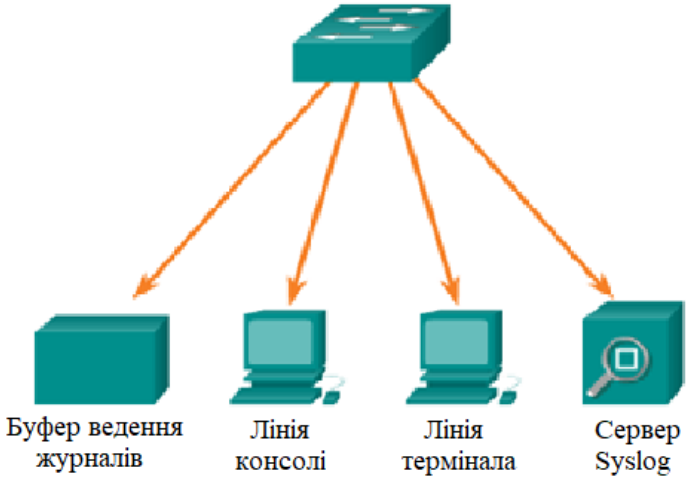


Рисунок 2.41 – Варіанти напрямів передачі повідомлень Syslog

Встановити варіанти спрямування повідомлень Syslog дозволяють наступні команди:

- **logging console** – для спрямування у консольну лінію (за замовчуванням);
- **logging vty** – для спрямування у лінію віддаленого доступу;
- **logging buffered** – для спрямування у буфер внутрішньої лінії пристрою;
- **logging *ip-address*** – для спрямування на Syslog-сервер;
- **logging monitor *L*** – встановлення мінімального рівня важливості повідомлень, які підлягають відправці (***L*** = 0…7).

Повідомлення журналу можуть супроводжуватися мітками часу. Також може призначатися адреса джерела повідомлень Syslog. Це підвищує ефективність налагодження й керування в режимі реального часу.

Якщо введена команда режиму глобальної конфігурації **service timestamps log uptime**, для регистрируємих подій відображається час, що пройшов з моменту останнього завантаження комутатора. У більш корисній версії цієї команди застосовується ключове слово **datetime** замість ключового

слова **uptime**; у цьому випадку для кожного зареєстрованого події будуть відображатися дата й час. При використанні ключового слова **datetime** необхідно настроїти годинник мережного обладнання.

Протокол Syslog використовує транспортні порти 514 (UDP частіше і TCP).

2.5.3 Протокол синхронізації часу NTP

Годинник мережевого устаткування можна настроїти одним із двох способів:

- вручну за допомогою команди clock set;
- Автоматично за допомогою протоколу NTP.

NTP працює за допомогою ієрархічної системи джерел часу для координування та налаштування налаштувань часу пристроїв у мережі. Він використовує концепції рівнів stratum, серверів та клієнтів для управління синхронізацією часу.

Рівні stratum представляють шари серверів часу в ієрархічній структурі NTP. Рівень stratum вказує на відстань від кінцевого еталонного годинника, при цьому stratum-0 є найбільш точним. Сервери stratum-1 безпосередньо підключені до еталонів stratum-0, таких як атомні годинники або приймачі глобальної системи позиціонування (GPS), що забезпечує найвищий рівень точності. Сервери stratum-2 синхронізуються з серверами stratum-1 і так далі.

Сервери NTP можуть бути класифіковані на різні типи залежно від їх рівнів stratum:

Сервери Stratum-1: Ці сервери мають безпосередній доступ до первинних джерел часу і вважаються найбільш точними еталонами часу.

Сервери Stratum-2: Ці сервери синхронізуються з серверами stratum-1. Вони надають інформацію про час нижчим рівням серверів і клієнтів. Сервери stratum-2 можуть синхронізуватися з декількома серверами stratum-1 для забезпечення резервування та надійності.

Сервери Stratum-3 і нижче: Ці сервери продовжують ієрархічну структуру, синхронізуючись з серверами вищих рівнів і надаючи інформацію про час клієнтам.

Процес синхронізації годинників у NTP включає обмін пакетами з часовими мітками між серверами і клієнтами. Процес синхронізації починається з того, що клієнт запитує інформацію про час у вибраного сервера. Сервер відповідає часовою міткою, що містить його локальний час. Клієнт порівнює цю часову мітку зі своїм локальним часом і розраховує зрушення між двома годинниками. За допомогою обміну інформацією з кількома серверами та врахування їх рівнів stratum та пов’язаних метрик, NTP може вибрати найбільш точні джерела часу та відповідно налаштовувати годинник клієнта. Всі повідомлення NTP використовують транспортний порт 123 (UDP).

Для підтримки синхронізації NTP постійно моніторить і налаштовує годинник кожного пристрою. Невеликі налаштування здійснюються з часом для

компенсації дрейфу годинників, тобто тенденції годинників дещо відхилятися від точного часу протягом тривалих періодів. Ці налаштування забезпечують, щоб пристрої залишалися в межах допустимого діапазону від еталонного джерела часу.

Щоб дозволити синхронізацію годинника мережевого устаткування із сервером часу NTP, використовується команда **ntp server *ip-address*** у режимі глобальної конфігурації. Приклад настроювання показаний на рисунку 2.42, тут маршрутизатор R1 настроєний як клієнт NTP, а маршрутизатор R2 виступає в якості довіреного сервера NTP. Мережне обладнання можна настроїти в якості сервера NTP (що дозволяє іншим обладнанням синхронізуватися з його часом) або в якості клієнта NTP.

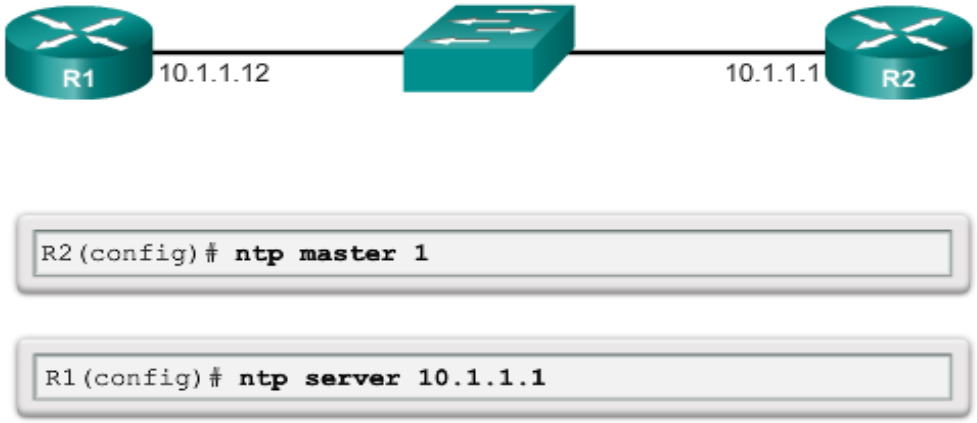


Рисунок 2.42 – Налаштування серверу та клієнту NTP

2.5.4 Контроль мережевого трафіку засобами NetFlow

NetFlow – це технологія Cisco IOS, що надає статистичні дані про пакети, які проходять через маршрутизатор або багаторівневий комутатор Cisco.

Протокол NetFlow надає дані, які дозволяють додати до моніторингу мережі й контролю безпеки, плануванню мережі й аналізу трафіку такі можливості, як виявлення вузьких місць мережі й облік IP для виставлення рахунків. Наприклад, на рисунку 2.43 PC1 підключений до PC2 за допомогою додатка, такого як HTTPS. Протокол NetFlow може контролювати це підключення рівня додатків, відслідковуючи кількість байт і пакетів для даного потоку. Потім він передає статистичні дані на зовнішній сервер, який називається збирачем даних (collector) NetFlow.

NetFlow обробляє з'єднання TCP/IP для ведення статистичного обліку, використовуючи поняття потоку. «Потік» (flow) – це односпрямована послідовність пакетів між певною системою-джерелом і певним призначенням. Для протоколу NetFlow, побудованого на основі TCP/IP, джерело й призначення визначаються IP-адресами мережного рівня, а також номерами порту джерела й портів призначення транспортного рівня.

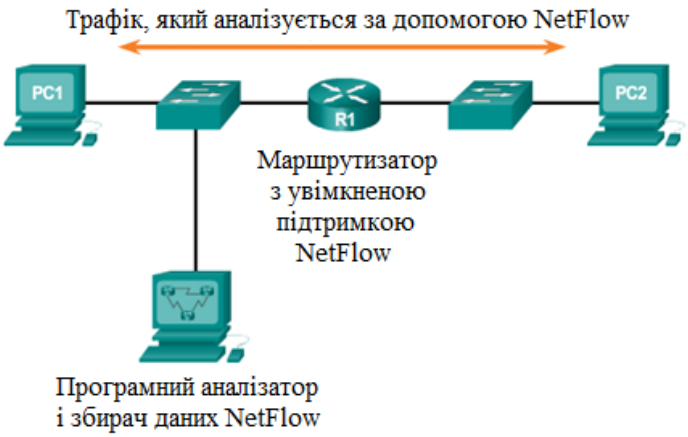


Рисунок 2.43 – Робота протоколу NetFlow

Існує кілька поколінь технології NetFlow, кожне з яких відрізняється вдосконаленнями в області визначення потоків трафіка. Спочатку протокол NetFlow розрізняв потоки за допомогою комбінації із семи характеристик. Якщо значення одного із цих полів відрізняється від значення іншого пакета, можна із упевненістю затверджувати, що ці пакети відносяться до різних потоків:

- IP-адреса джерела;
- IP-адреса призначення;
- номер порту джерела;
- номер порту призначення;
- тип протоколу рівня 3;
- маркування ToS (тип обслуговування);
- логічний вхідний інтерфейс.

Тип протоколу рівня 3 визначає тип заголовка, що іде за заголовком IP (звичайно TCP або UDP, але можливі інші варіанти включаючи ICMP). Байт ToS у заголовку IPv4 містить відомості про те, як обладнання повинні застосовувати правила якості обслуговування (QoS) до пакетів у даному потоці.

Порядок розгортання NetFlow на маршрутизаторі (рисунок 2.44) наступний:

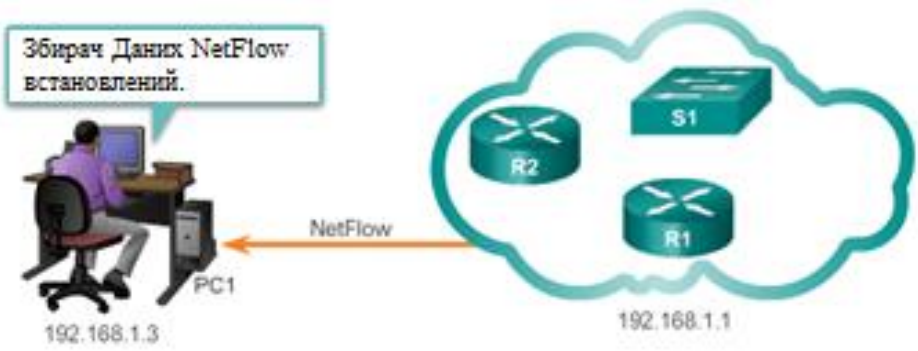
Крок 1. Настроювання збору даних NetFlow – NetFlow збирає дані із вхідних і вихідних пакетів.

Крок 2. Настроювання експорту даних NetFlow – необхідно вказати IP-адресу або ім’я вузла збирача даних NetFlow, а також порт UDP, що прослуховується збирачем NetFlow.

Крок 3. Перевірка працездатності й статистики NetFlow – після настроювання NetFlow експортовані дані можна проаналізувати на робочій станції, використовуючи такі додатки, як Solarwinds, NetFlow Traffic Analyzer, Plixer Scrutinizer або Cisco NetFlow Collector (NFC). У якості мінімального варіанта можна використовувати вихідні дані ряду команд show на самому маршрутизаторі.

На рисунку 2.44 маршрутизатору R1 призначена IP-адреса 192.168.1.1 на інтерфейсі g0/1. Збирачеві NetFlow призначена IP-адреса 192.168.1.3, і він налаштований для збору даних через порт UDP 2055. Здійснюється

спостереження за вхідним і вихідним трафіком через інтерфейс g0/1. Дані NetFlow відправляються у форматі версії 5.




```
R1(config)# interface GigabitEthernet 0/1
R1(config-if)# ip flow ingress
R1(config-if)# ip flow egress
R1(config-if)# exit
R1(config)# ip flow-export destination 192.168.1.3 2055
R1(config)# ip flow-export version 5
```

Рисунок 2.44 – Налаштування протоколу NetFlow

Перевірка NetFlow виконується шляхом перегляду інформації, що зберігається на збирачі даних NetFlow. Щоб переглянути зведену статистику NetFlow, а також довідатися, який протокол використовує максимальний об'єм трафіка й між якими вузлами цей трафік передається, можна і на маршрутизаторі ввести команду **show ip cache flow** у користувацькому або привілейованому режимі. Виконання цієї команда показана на рисунку 2.45.

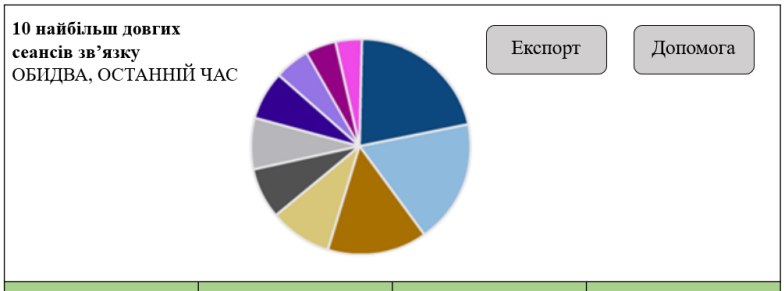
```
R1# show ip cache flow
IP packet size distribution (178617 total packets):
   1-32   64   96  128  160  192  224  256  288  320  352  384  416  448  480
   .002 .080 .008 .005 .001 .000 .001 .001 .000 .000 .000 .000 .000 .000 .000

    512  544  576 1024 1536 2048 2560 3072 3584 4096 4608
   .000 .000 .000 .000 .895 .000 .000 .000 .000 .000 .000

IP Flow Switching Cache, 278544 bytes
  5 active, 4091 inactive, 1573 added
  18467 ager polls, 0 flow alloc failures
  Active flows timeout in 1 minutes
  Inactive flows timeout in 15 seconds
IP Sub Flow Cache, 34056 bytes
  5 active, 1019 inactive, 1569 added, 1569 added to flow
  0 alloc failures, 0 force free
  1 chunk, 1 chunk added
  last clearing of statistics never
Protocol        Total    Flows   Packets Bytes  Packets Active(Sec) Idle(Sec)
--------        Flows     /Sec    /Flow  /Pkt     /Sec     /Flow     /Flow
TCP-Telnet          3       0.0        3    50      0.0       1.0      15.0
TCP-WWW           245       0.0        6    93      0.0       0.3       2.4
TCP-other         529       0.0       27    57      0.2       0.7       6.2
UDP-other         328       0.0        6   107      0.0       2.4      15.3
ICMP              711       0.0      226  1261      2.4       0.2      15.4
Total:           1816       0.0       98  1137      2.7       0.8      11.0
```

Рисунок 2.45 – Перевірка роботи протоколу NetFlow

Збирач даних NetFlow – це комп'ютер, на якому встановлене прикладне програмне забезпечення. Це програмне забезпечення спеціальне призначене для обробки «сирих» даних NetFlow. Збирач даних можна настроїти для одержання інформації NetFlow від багатьох мережних обладнань. Збирачі даних NetFlow поєднують і впорядковують дані NetFlow відповідно до вказівок мережних адміністраторів у рамках обмежень програмного забезпечення.

На збирачі NetFlow дані NetFlow записуються на диск через задані проміжки часу. Адміністратор може виконувати кілька схем або потоків збору даних одночасно. Наприклад, різні сегменти даних можуть зберігатися для підтримки планування замість обліку викликів і часу розмови абонента; збирач даних NetFlow може легко створити належні схеми агрегації.

На рисунку 2.46 представлений інтерфейс збирача даних NetFlow, що пасивно прослуховує експортовані дейтаграми NetFlow. Додаток збору даних NetFlow являє собою багатофункціональне, просте у використанні, масштабоване рішення для регулювання споживання експортованих даних NetFlow з різних обладнань. Передбачуване використання в організаціях варіюється, однак часто метою є підтримка важливих потоків, пов'язаних з додатками користувачів. Сюди відносяться облік, виставлення рахунків, планування мережі і її моніторинг.



ДІАЛОГ	ВХІДНІ БАЙТИ	ВИХІДНІ БАЙТИ	ВХІДНІ ПАКЕТИ
Між KJOHNSTONXP і rapidshare.com (198.18.131.7)	354,4 Кбайт	10,7 Мбайт	621
Між Mbishopwks і rapidshare.com (198.18.131.7)	243 Кбайт	9 Мбайт	445

Рисунок 2.46 – Приклад інтерфейсу збирача даних NetFlow

Збирач даних NetFlow надає візуалізацію й аналіз записаних і об'єднаних даних потоку в режимі реального часу. Можна вказати маршрутизатори й підтримувані комутатори, а також схему агрегації й тривалість зберігання даних до наступного періодичного аналізу. Можна виконувати впорядкування й візуалізацію даних зручним для користувачів способом: у вигляді стовпчастих діаграм, кругових діаграм або гістограм упорядкованих звітів. Дані після цього можна експортувати в електронні таблиці, наприклад Microsoft Excel, для більш детального аналізу, визначення тенденцій і звітності.

2.5.5 Режим забезпечення роботи засобів SPAN

SPAN (Switched Port Analyzer) – це технологія в мережевих комутаторах (світчах), яка дозволяє копіювати трафік з одного або кількох портів і перенаправляти цю копію на визначений порт для аналізу. Цю функцію також називають *дзеркалюванням трафіку* (рисунок 2.47).

В багатьох мережних комутаторах є можливість дзеркалювати трафік з одного порту на інший (локальне дзеркалювання), або, наприклад, із зазначеного VLAN на порт іншого VLAN (вилучене дзеркалювання), де перебуває підключений аналізатор трафіка – пристрій з прикладним програмним забезпеченням.

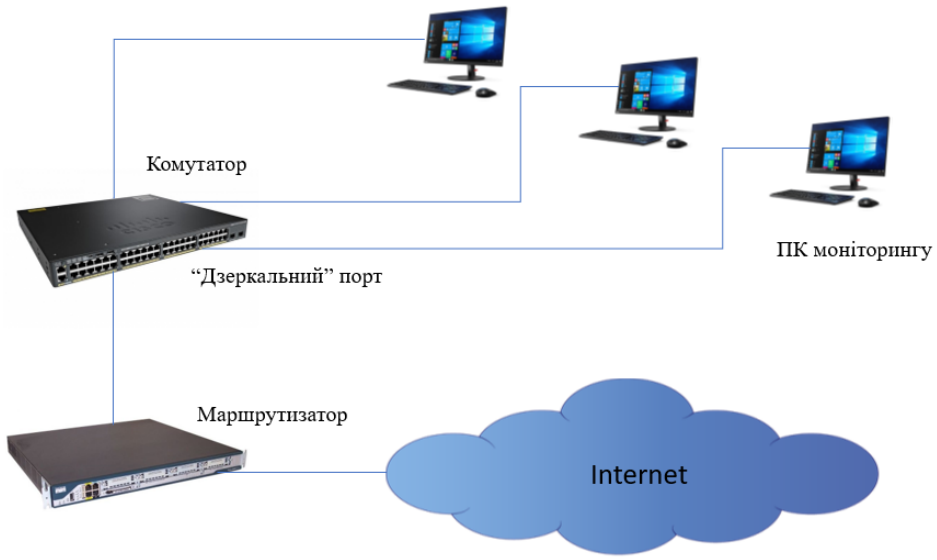


Рисунок 2.47 – Робота функції SPAN

Розрізняють дві технології SPAN це:

- SPAN (Switch Port Analyzer), який працює в межах одного комутатора;
- RSPAN (Remote Switch Port Analyzer), який може дзеркалювати і передавати трафік між комутаторами.

SPAN може використовуватися:

- для пошуку несправностей, шляхом аналізу трафіку за допомогою аналізатора;
- для перенаправлення трафіку для аналізу у системі виявлення вторгнень (IDS).
- для запису даних VoIP.

Для настроювання роботи SPAN, необхідно:

1. Створити список джерел – звідки брати трафік для аналізу або інших цілей. Тут можна вказувати порти (≥1), або VLAN (≥1).
2. Указати куди доставляти дані із джерел, описаних у першому пункті. Тобто куди буде дзеркалюватися трафік (наприклад, порт).

Джерелами трафіка можуть бути layer 2 порти: access port, trunk port, etherchannel; layer 3 (routed port) і так далі. Якщо джерело зазначене як VLAN, то буде деркалюватися трафік усіх портів, які включені в дану VLAN і в цей час

активні. Можна включати або видаляти з VLAN порти, і це відразу буде впливати на SPAN/RSPAN.

Що стосується RSPAN, то тут джерело описується аналогічно: або порт/порти, або VLAN/VLANи. А от Destination описується на основі спеціального RSPAN VLAN, а не окремого порту.

Приклад настроювання SPAN

Припустимо, потрібно одержувати трафік з порту f0/34 (source) на порт f0/33 (destination):

ASW(config)#monitor session 1 source interface f0/34
ASW(config)#monitor session 1 destination interface f0/33

Для того, щоб обмежити трафік, що тільки входить, потрібно зробити наступне:

ASW(config)#monitor session 1 source interface f0/34 rx

rx – параметр, що обмежує тільки вхідний трафік для дзеркалювання з даного інтерфейсу.

Якщо потрібний тільки вихідний трафік, потрібно зробити так:

ASW(config)#monitor session 1 source interface f0/34 tx

У прикладі, дзеркалюється трафік тільки з одного порту, але нічого не заважає зробити дзеркальовання із двох і більш портів, для цього просто додати команди з відповідними інтерфейсами:

ASW(config)#monitor session 1 source interface f0/34
ASW(config)#monitor session 1 source interface f0/35
ASW(config)#monitor session 1 destination interface f0/33

Тепер трафік із двох інтерфейсів f0/34 і f0/35 буде копіюватися на destination port. Команда ***show monitor session*** *[номер сесії]* дозволяє переглянути налаштування функцій SPAN.

Контрольні питання до розділу 2

1. Для безкласової IP адреси XXX.XXX.XXX.XXX/YY визначити її призначення (network, host, broadcast).
2. Розподілити мережу з IP адресою XXX.XXX.XXX.XXX на підмережі з маскою XXX.XXX.XXX.XXX/YY. Навести повну маску мережі, кількість підмереж, кількість хостів в кожній підмережі та для кожної підмережі першу IP-адресу підмережі, IP-адресу широкомовного повідомлення та одну IP-адресу будь-якого хоста.

3. Замість однієї мережі XXX.XXX.XXX.XXX класу C, забезпечити утворення Z безкласових мереж з можливістю розміщення в них RR, RR, R по T, R по N хостів відповідно. Навести номер і маску кожної мережі та максимальну кількість хостів в них.

4. Описати порядок роботи протоколу управляючих повідомлень ICMP (зокрема в утілитах Ping та TracerRoute)

5. Описати та охарактеризувати протокол визначення сусідніх пристроїв CDP

6. Описати та охарактеризувати протокол маршрутизації RIP.

7. Описати та охарактеризувати протокол маршрутизації OSPF.

8. Описати та охарактеризувати протокол маршрутизації BGP.

9. Описати та охарактеризувати протокол маршрутизації EIGRP.

10. Пояснити призначення та описати роботу протоколу віддаленого керування Telnet.

11. Пояснити призначення та описати роботу протоколу збору керуючої інформації Syslog.

12. Пояснити призначення та описати роботу протоколу контролю мережевого трафіку засобами NetFlow.

13. Пояснити призначення та описати роботу протоколу забезпечення роботи засобів SPAN.

14. Описати протокол динамічного настроювання хостів Dynamic Host Configuration Protocol (DHCP) - призначення та алгоритм роботи. Пояснити стек команд для конфігурування

15. Пояснити призначення та описати варіанти ACL списків.

16. Пояснити призначення, описати варіанти та роботу протоколів NAT (PAT).

17. Описати протокол управління мережами Simple Network Managment Protocol (SNMP) - призначення, алгоритм роботи, поняття MIB, стек команд для настроювання.

РОЗДІЛ 3 ПРОТОКОЛИ РЕАЛІЗАЦІЇ НАДІЙНИХ ТА БЕЗПЕЧНИХ МЕРЕЖ

3.1 Основи забезпечення безпеки інформації в інформаційно-комунікаційних мережах

3.1.1 Модель інформаційної безпеки

На рисунку 3.1 показаний взаємозв'язок між основними поняттями інформаційної безпеки.

Рисунок 3.1 – Поняття інформаційної безпеки

Активи – це щось цінне для організації, наприклад, дані і інша інтелектуальна власність, сервери, комп'ютери, смартфони, планшети та ін.

Загроза – потенційна небезпека для активу, наприклад, даних або самої мережі.

Вразливість – слабкість системи або її конструкції, яка може бути використана загрозою.

Поверхня атаки – загальна сума вразливостей в даній системі, доступна зловмисникові. Поверхня атаки описує вразливі точки, де зловмисник може потрапити в систему і отримати дані з системи. Наприклад, операційна система і веб-браузер можуть не мати необхідні виправлення з безпеки, тоді вони вразливі до атак.

Ризик – ймовірність того, що певна загроза використовує певну вразливість активу і приведе до небажаних наслідків.

Управління ризиками – це процес, котрий балансує операційні витрати на забезпечення захисних заходів з прибутком, досягнутим в результаті захисту активу.

Є чотири загальні способи управління ризиками:

прийняття ризику – це коли вартість варіантів управління ризиками перевищує вартість самого ризику і ризик приймається без дії;

запобігання ризику – це дії, які здатні усунути будь-який ризик. Як правило, це найдорожчий варіант управління ризиком;

обмеження ризику – це дії, які обмежують ризики для компанії в результаті визначених дій. Тобто частково виконується прийняття ризику, а частково запобігання ризику. Це найбільш поширена стратегія;

передача ризику – ризик переходить до третій стороні, наприклад страхової компанії.

Конфіденційність, цілісність і доступність, відомі як тріада CIA, є керівним принципом інформаційної безпеки для організації.

Конфіденційність (Confidentiality)

Синонімом терміну конфіденційність є приватність – обмеження доступу до інформації тільки колом авторизованих користувачів і гарантування відсутності доступу в інших осіб. Доступ до даних надається відповідно до політики безпеки або рівня секретності.

Цілісність (Integrity)

Цілісність – це точність, узгодженість і достовірність даних протягом усього їх життєвого циклу. Дані повинні залишатися незмінними в процесі передачі, а також недоступними для змін неавторизованими суб'єктами.

Доступність (Availability)

Доступність гарантує, що інформація доступна для авторизованих користувачів.

На рисунку 3.2 зображена модель інформаційної безпеки.

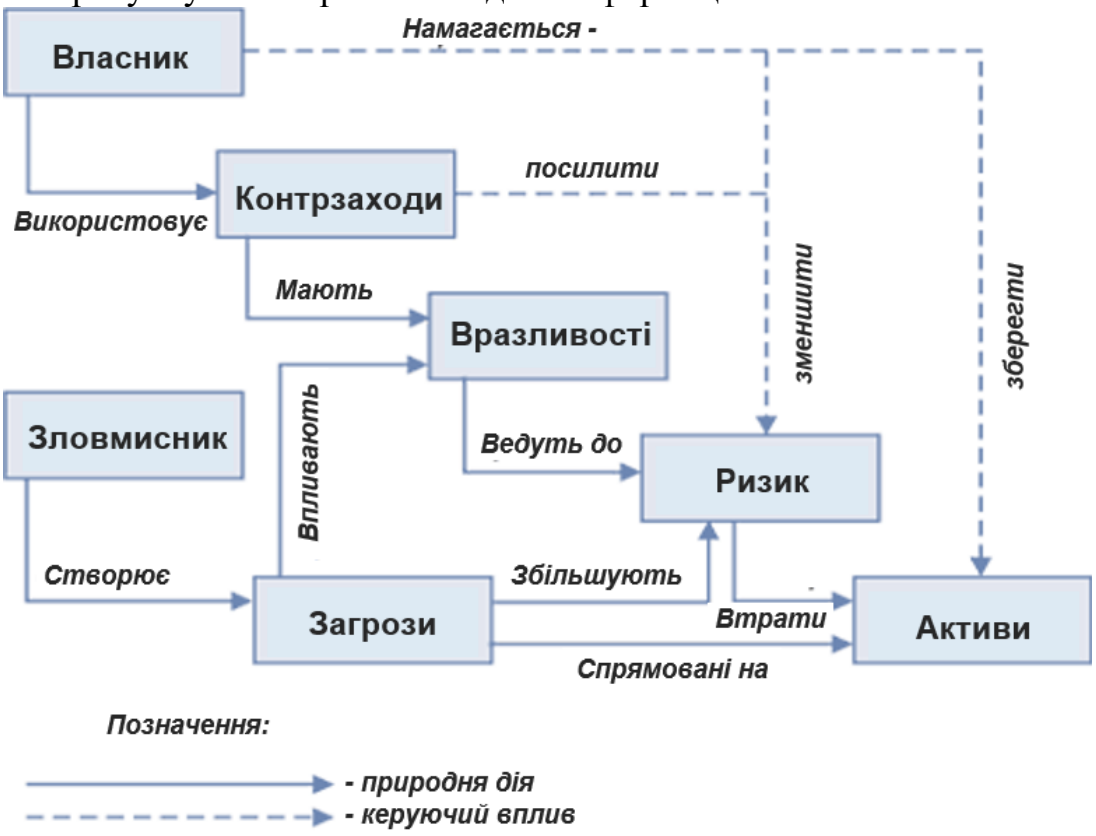


Рисунок 3.2 – Модель інформаційної безпеки

Зловмисники – це окремі особи або групи, які намагаються використовувати кібервразливості для особистої або фінансової вигоди. Виділяють три категорії зловмисників.

Любителі (Amateurs) – таких людей іноді називають скрипт-дітьми (Script Kiddies). Зазвичай це зловмисники, які практично не мають навичок і часто для запуску атак використовують існуючі інструменти або вказівки, знайдені в Інтернеті. Деякі з них роблять це просто з цікавості, а інші намагаються продемонструвати свою майстерність і заподіяти шкоду. Навіть якщо вони використовують прості інструменти, наслідки таких атак можуть бути руйнівними.

Хакери (Hackers) – це група зловмисників, яка зламує комп'ютери або мережі з метою отримання доступу. Залежно від наміру вторгнення ці зловмисники класифікуються як Білі, Сірі або Чорні капелюхи. *Білі капелюхи* втручаються в мережі або комп'ютерні системи, щоб виявити слабкі місця і поліпшити безпеку цих систем. Такі вторгнення виконуються за попереднім дозволом і все результати повідомляються власнику. *Чорні капелюхи* використовують будь-яку вразливість для отримання незаконної особистої, фінансової або політичної вигоди. *Сірі капелюхи* можуть виявити вразливість в системі і повідомити про неї власнику системи, якщо це збігається з їхніми планами. Деякі Сірі капелюхи публікують факти про уразливість в Інтернеті, щоб інші нападники могли нею скористатися.

Організовані хакери (Organized Hackers) – організації кіберзлочинців, хактівістів, терористів і хакерів, що фінансуються державою. *Кіберзлочинці* – це групи професійних злочинців, орієнтовані на контроль, владу і збагачення, вони високо кваліфіковані та організовані. *Хактивісти* роблять політичні заяви, щоб привернути увагу до важливих для них проблем. *Хакери, що фінансуються державою,* здійснюють розвідку або диверсії від імені свого уряду. Ці хакери зазвичай якісно підготовлені та добре фінансуються, а їх атаки зосереджені на потрібних для їх уряду конкретних цілях.

Кіберпростір став про ще одним важливим доменом війни, де країни можуть конфліктувати без зіткнення традиційних військ і техніки. Це дозволяє країнам з мінімальними військовими силами бути такими ж сильними, як і інші країни в кіберпросторі. ***Кібервійна*** – це Інтернет-конфлікт, який передбачає проникнення в комп'ютерні системи і мережі інших країн. Нападаючи мають ресурси і спеціальні знання, щоб проводити масштабні інтернет-атаки проти інших країн, завдаючи шкоди або порушуючи роботу сервісів, наприклад припинення роботи енергосистеми.

3.1.2 Поширені загрози безпеки мереж передачі даних

Деякі різновиди загроз для інформаційних ресурсів і систем наведені на рисунку 3.3.

Вірус – це тип шкідливих програм, який поширюється, вставляючи копію себе в іншу програму. Потім віруси поширюються з одного комп'ютера на

інший, інфікуючи всі комп'ютери мережі. Більшість вірусів потребують допомоги людини для поширення. Наприклад, через інфікований ним диск USB. Віруси можуть перебувати в стані спокою протягом тривалого періоду часу, а потім активуватися в певний час і дату.

Простий вірус може встановити себе в першому рядку коду в виконуваному файлі. Після активації вірус може перевірити диск на інші виконувані файли, щоб він міг заразити всі файли, які він ще не заразив. Віруси можуть бути нешкідливими, наприклад, що відображають зображення на екрані, або вони можуть бути руйнівними, наприклад, ті, які змінюють або видаляють файли на жорсткому диску. Віруси також можуть бути запрограмовані на мутацію, щоб уникнути їх виявлення.

Рисунок 3.3 – Поширені загрози для інформаційних ресурсів і систем

Більшість вірусів тепер поширюються за допомогою накопичувачів USB, компакт-дисків, DVD-дисків, мережевих ресурсів і електронної пошти. Зараз віруси електронної пошти є найбільш поширеним типом вірусу.

Комп'ютерні черв'яки – подібні вірусам, так як вони розмножуються і можуть викликати один і той самий збиток. Однак черви розмножуються використовуючи вразливості в мережах. Черви можуть уповільнювати роботу мереж, коли вони поширюються з системи в систему.

У той час як вірус вимагає запуску хост-програми, черв'яки можуть працювати самі. Крім початкової інфекції, вони більше не вимагають участі користувачів.

Троянські коні – це програмне забезпечення, яке зовні є законним, але воно містить шкідливий код, який використовує привілеї користувача, який виконує його. Часто трояни прикріплені до онлайн-ігор.

Користувачів зазвичай обманюють при завантаженні і виконанні програмного продукту та троянського коня на своїх системах. Під час гри, користувач не помітить проблем, а троянський кінь встановлюється в системі користувача у фоновому режимі. Зловмисний код троянського коня продовжує працювати навіть після закриття гри.

Це може привести до негайного ушкодження, забезпечити віддалений доступ до системи або доступ через задні двері (backdoor). Він також може

виконувати дії, зазначені дистанційно, наприклад «відправити мені файл паролів один раз в тиждень».

Всі три наведені типи шкідливого програмного забезпечення можуть реалізовувати одну з наступних функцій.

Ransomware – це шкідлива програма, яка забороняє доступ до інфікованої комп'ютерної системи або її даним. Потім кіберзлочинці вимагають виплати, щоб разблокувати комп'ютерну систему. Ransomware часто використовує алгоритм шифрування для шифрування системних файлів і даних.

Шпигунське програмне забезпечення (Spywire) – це шкідлива програма, яка використовується для збору інформації про користувача і передачі інформації іншому суб'єкту без згоди користувача.

Adware – це шкідлива програма, яка відображає спливаючі вікна для отримання доходу їх автору . Програма може аналізувати інтереси користувачів шляхом відстеження відвідуваних веб-сайтів. Потім вона може формувати спливаючі реклами, що стосуються цих сайтів.

Scareware – це шкідливе програмне забезпечення включає в себе програмне забезпечення, яке використовує соціальну інженерію для шокування або хвилювання, створюючи сприйняття загрози. Вона, як правило, спрямована на нічого не підозрює користувача і намагається переконати користувача інфікувати комп'ютер шляхом вжиття заходів щодо усунення фіктивної загрози.

Фішинг – ця шкідлива програма намагається переконати людей розкривати конфіденційну інформацію. Наприклад , отримання електронного листа від свого банку з проханням розкрити свої рахунки і PIN-коди.

Rootkits – це шкідлива програма, встановлена на скомпрометовану систему, що надає привілейований доступ для зловмисника.

Атаки класифікують в трьох основних категоріях:

- розвідувальні атаки;
- атаки доступу;
- атаки DoS.

Зловмисники використовують ***атаки розвідки*** (або reconnaissance) для несанкціонованого виявлення і відображення систем, служб або вразливостей. Тобто зловмисник може отримати профіль системи, включаючи тип і версію операційної системи. Якщо система не повністю виправлена, зловмисник шукатиме відомі уразливості для впливу . Розвідувальна атака має такі етапи:

інформаційний запит про ціль – зловмисник шукає початкову інформацію про мету. Використовуються доступні інструменти, включаючи пошук Google на сайті організації. Публічна інформація про цільову мережу доступна з реєстрів DNS, використовуючи утиліти dig, nslookup і whois.

ping перевірка – зловмисник ініціює цю перевірку мереж, виявлених попередніми запитами DNS для визначення мережевих адрес, які є активними. Це дозволяє створити логічну топологію цільової мережі.

сканування портів активних IP-адрес – зловмисник ініціює сканування портів на хостах, виявлених попередньою перевіркою, щоб визначити, які порти або сервіси відкриті. Інструменти сканування портів, такі як Nmap, SuperScan,

Angry IP Scanner і NetScanTools, ініціюють підключення до хостів та сканування відкритих портів.

запуск сканерів вразливості – зловмисник використовує інструмент сканування вразливостей, такий як Nipper, Secunia PSI, Core Impact, Nessus v6, SAINT або Open VAS для запиту ідентифікованих портів. Мета полягає у виявленні потенційних вразливостей на цільових хостах.

запуск інструментів впливу – зловмисник намагається використовувати виявлені вразливості в системі. Зловмисник використовує такі інструменти використання вразливостей, як Metasploit, Core Impact, Sqlmap, інструментарій соціального інженера і Netsparker.

Атаки доступу використовують відомі вразливості служб аутентифікації, FTP-сервісів і веб-служб для доступу до веб-облікових записів, конфіденційних баз даних та іншої конфіденційної інформації. Мета зловмисника може полягати в крадіжці інформації або дистанційному управлінні внутрішнім хостом. Можливі такі варіанти атак доступу:

атака паролем – зловмисники намагаються виявити критичні паролі системи, використовуючи різні методи, такі як фішингові атаки, атаки з перебору словників, атаки з використанням грубої сили, перехоплення в мережі або використанні методів соціальної інженерії. Грубі атаки паролів включають множинні спроби з використанням таких інструментів, як Ophcrack, L0phtCrack, THC Hydra, RainbowCrack і Medusa.

Pass-the-hash – зловмисник вже має доступ до машини користувача і використовує шкідливе програмне забезпечення для доступу до збережених хешей паролів. Потім зловмисник використовує хеші для аутентифікації на інших віддалених серверах і пристроях без використання грубої сили.

Використання довіри – зловмисники використовують довірений хост для доступу до мережевих ресурсів (рисунок 3.4). Наприклад, зовнішній хост, який звертається до внутрішньої мережі через VPN. Якщо цей хост атакується, зловмисник може використовувати довірений хост, щоб отримати доступ до внутрішньої мережі.

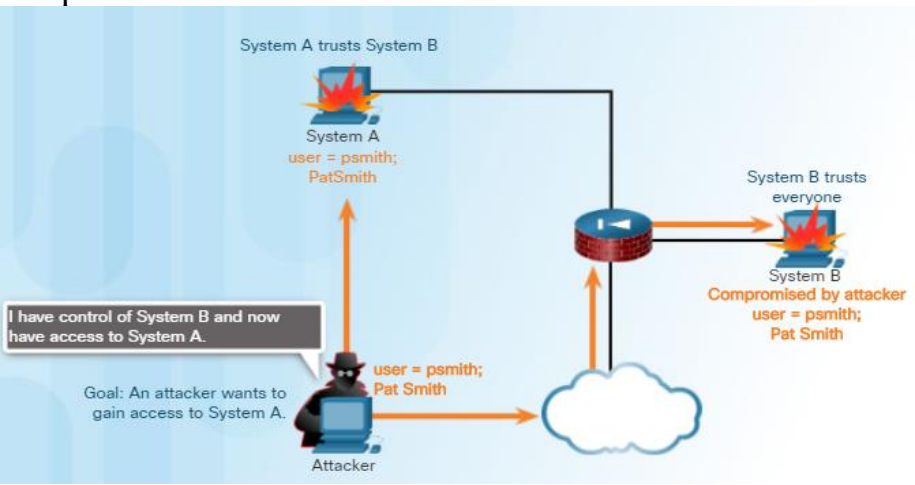


Рисунок 3.4 – Використання довіреного хоста для доступу до мережевих ресурсів

Перенаправлення портів – це коли зловмисник використовує скомпрометовану систему в якості бази для атак на інші цілі (рисунок 3.5).

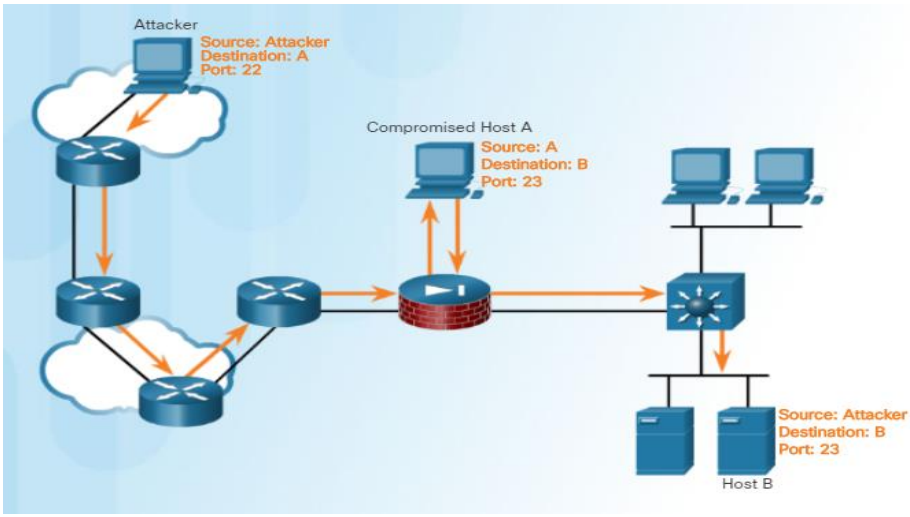


Рисунок 3.5 – Використання перенаправлення портів для атак на інші цілі

Спуфінг-атака на комутатор – це тип атаки мережі VLAN, при якому використовується неправильно налагоджений транковий порт. За замовчанням транкові порти мають доступ до усіх мереж VLAN і передають трафік для декількох VLAN через один і той же фізичний канал.

У простому випадку спуфінг-атаки комутатора зловмисник у своїх інтересах користується тим, що порт комутатора за умовчанням налаштований на динамічний автоматичний режим включення транкового режиму. Зловмисник конфігурує систему так, щоб вона поводилася як комутатор. Для такого спуфінга зловмисник повинен уміти імітувати повідомлення 802.1Q і DTP. Змусивши комутатор думати, що інший комутатор намагається створити транковий канал, зловмисник може отримати доступ до усіх мереж VLAN, дозволених на цьому транковому порту.

Атака з подвійним тегуванням – це атака з подвійною інкапсуляцією. Цей вид атаки заснований на використанні принципів роботи апаратного забезпечення на більшості комутаторів. Більшість комутаторів виконують лише один рівень деінкапсуляції 802.1Q, що дозволяє зловмисникові вставляти в кадр приховану мітку 802.1Q. Мітка дозволяє пересилати кадр в мережу VLAN, яка не вказана первинною міткою 802.1Q. Важлива властивість атаки з подвійною інкапсуляцією полягає в тому, що вона діє, навіть якщо транкові порти відключені.

Атака VLAN hopping з подвійним тегуванням відбувається в три кроки (рисунок 3.6):

1. Зловмисник відправляє на комутатор кадр з подвійним тегуванням 802.1Q. Зовнішній заголовок містить мітку мережі VLAN зловмисника, яка співпадає з native VLAN транкового порту. Передбачається, що комутатор обробляє отриманий від зловмисника кадр, ніби він знаходиться на транковому порту або порту з голосовою VLAN (комутатор не повинен отримувати

тегований кадр Ethernet на порту доступу). Як приклад уявіть, що мережею native VLAN є VLAN 10. Внутрішній тег – це VLAN, який є ціллю атаки. В даному випадку – VLAN 20.

2. Кадр прибуває на комутатор, який перевіряє перші 4 байти тега 802.1Q. Комутатор бачить, що кадр призначений для VLAN 10, яка є мережею native VLAN. Видаливши мітку VLAN 10, комутатор пересилає пакет з усіх портів мережі VLAN 10. На транковому порту віддаляється мітка мережі VLAN 10, але пакет не тегується наново, оскільки він є частиною native VLAN. В цей час мітка мережі VLAN 20 залишається незайманою і не перевіряється першим комутатором.

3. Другий комутатор перевіряє тільки внутрішній тег 802.1Q, відправлений зловмисником, і бачить, що кадр призначений для мережі VLAN 20, що є метою зловмисника. Другий комутатор відправляє кадр на порт, що атакується, або доповнює його лавинною розсилкою залежно від того, чи є в таблиці MAC-адрес запис для вузла, що атакується.

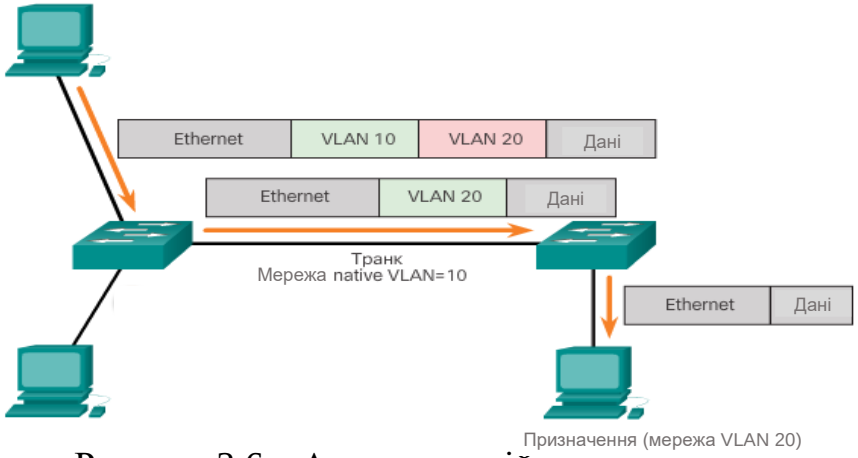


Рисунок 3.6 – Атака з подвійним тегуванням

Цей вид атаки є однонапрямленим і працює, тільки якщо зловмисник підключений до порту, що знаходиться в тій же VLAN, що і мережа native VLAN транкового порту. Запобігти такій атаці не так легко, як зупинити звичайні атаки VLAN hopping.

Атака «людина в середині» – зловмисник розташовується між двома законними об'єктами (рисунок 3.7), щоб читати, змінювати або перенаправляти дані, що передаються між ними.

Соціальна інженерія – це тип атаки доступу, коли зловмисник намагається маніпулювати людиною у виконанні дій або розголошення конфіденційної інформації, такої як паролі та імена користувачів. Як правило, це передбачає використання соціальних навичок для маніпулювання внутрішніми користувачами мережі для розкриття інформації, необхідної для доступу до мережі. Соціальні інженери часто покладаються на готовність людей бути корисними або їх слабкі місця. Наприклад, зловмисник може попросить в уповноваженого працівника допомогти в терміновій проблемі, що вимагає негайного доступу до мережі. Зловмисник може звернутися до самолюбства співробітника, застосувати владу або звернутися до жадібності працівника.

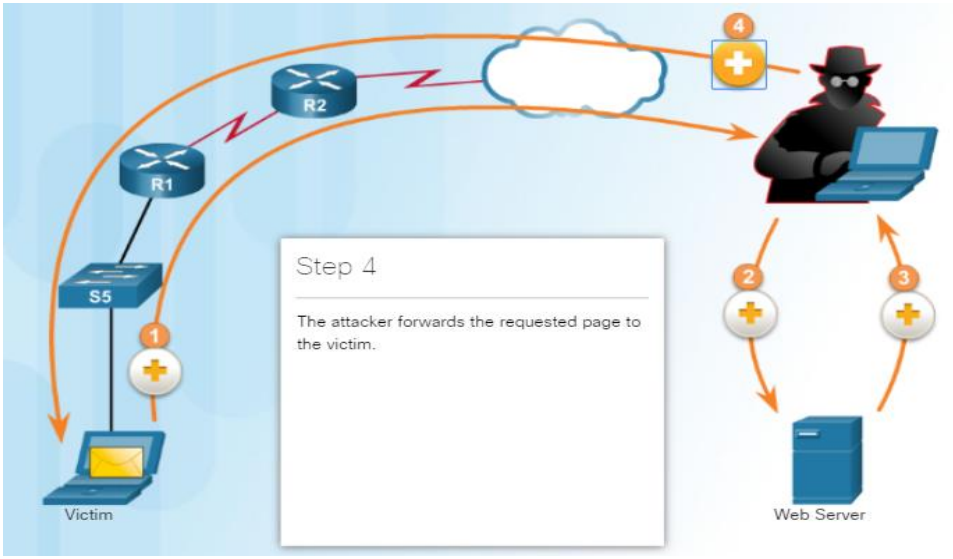


Рисунок 3.7 – Атака *«людина в середині»*

Лавинна атака таблиці MAC-адрес. Таблиця MAC-адрес комутатора містить MAC-адреси, які пов'язані з кожним фізичним портом і відповідною VLAN. Усі моделі комутаторів серії Catalyst використовують таблицю MAC-адрес для комутації 2-го рівня. З кадрів, що надходять на інтерфейси комутатора, MAC-адреси джерела реєструються в таблиці MAC-адрес. Коли комутатор 2-го рівня отримує кадр, він шукає в таблиці MAC-адрес MAC-адресу призначення. Якщо для цієї MAC-адреси існує запис, комутатор пересилає кадр тільки у відповідний інтерфейс. У випадку якщо MAC-адреси немає в таблиці MAC-адрес, комутатор розсилає кадр з кожного інтерфейсу, крім того, на якому цей кадр був отриманий.

Подібна поведінка комутатора відносно невідомих MAC-адрес може використовуватися для атаки на комутатор. Цей вид атаки називається переповнюванням таблиці MAC-адрес. Атаку переповнювання таблиці MAC-адрес іноді називають лавинною атакою.

На рисунку 3.8(а) вузол Host A відправляє трафік на вузол Host B. Комутатор отримує кадри і шукає MAC-адрес призначення в таблиці MAC-адрес. Якщо комутатор не може знайти MAC-адресу призначення в таблиці MAC-адрес, то він копіює кадр і розсилає його (широкомовна розсилка) з кожного інтерфейсу комутатора, крім того, на якому він був отриманий.

На рисунку 3.8(б) вузол Host B отримує кадр і відправляє відповідь на вузол Host A. Таким чином комутатор дізнається, що MAC-адрес вузла Host B знаходиться на інтерфейсі 2 і записує цю інформацію в таблицю MAC-адрес.

Вузол Host C також отримує кадр, спрямований від вузла Host A на вузол Host B, але, оскільки MAC-адресом призначення цього кадру являється вузол Host B, вузол Host C відкидає цей кадр.

При нормальному режимі, будь-який кадр, відправлений вузлом Host A (чи будь-яким іншим вузлом) на вузол Host B, пересилається на інтерфейс 2 комутатори, а не вирушає по широкомовній розсилці з усіх інтерфейсів.

Як показано на рисунку 3.8(в), зловмисник на вузлі Host C може відправляти на комутатор кадри з неіснуючими, випадково згенерованими MAC-адресами джерела і призначення. Комутатор оновлює таблицю MAC-адрес інформацією з фіктивних кадрів. Коли таблиця MAC-адрес наповнюється фіктивними MAC-адресами, комутатор входить в так званий режим з пропусканням трафіку. У цьому режимі комутатор відправляє усі кадри по широкомовній розсилці усім пристроям в мережі. В результаті зловмисник може бачити усі кадри, що розсилаються.

Деякі засоби мережевої атаки можуть генерувати до 155 000 записів в таблиці MAC-адрес комутатора в хвилину. Максимальний розмір таблиці MAC-адрес може розрізнятися залежно від комутатора (але як правило розрахований на 10 000 адрес).

Як показано на рисунку 3.8(г), до тих пір, поки таблиця MAC-адрес комутатора залишається заповненою, комутатор передає усі отримані кадри з кожного інтерфейсу по широкомовній розсилці. В даному прикладі кадри, відправлені з вузла Host A на вузол Host B, також розсилаються з інтерфейсу 3 комутатори по широкомовній розсилці і видні зловмисникові на вузлі Host C.

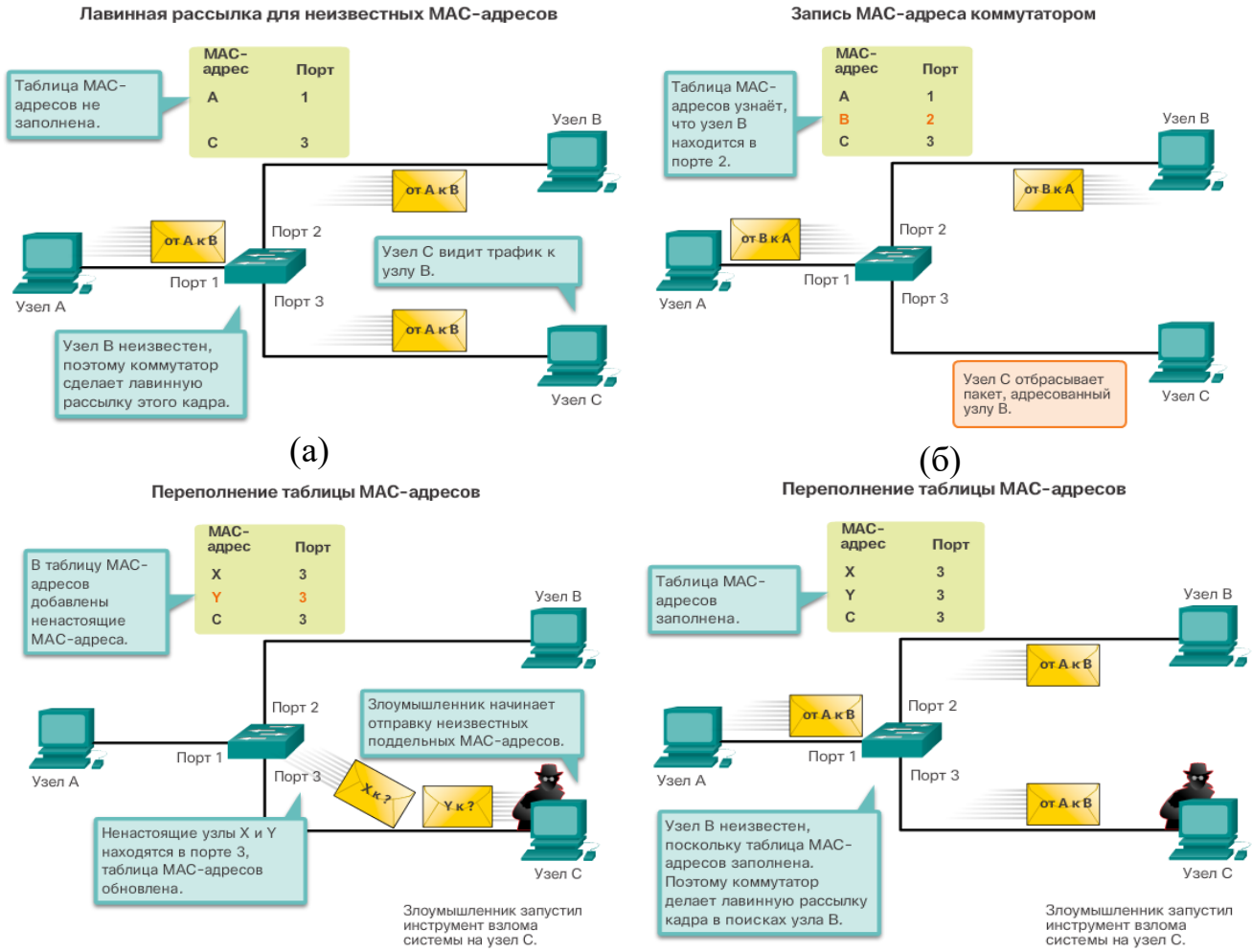


Рисунок 3.8 – Атака переповненням таблиці MAC-адрес

DHCP-атаки виконуються на комутовані мережі двома типами, які виконуються послідовно (як правило):

- атака виснаження ресурсів;
- DHCP-спуфінг.

При атаках виснаження ресурсів DHCP зловмисник створює лавинну розсилку з DHCP-запитів на DHCP-сервер, щоб використати усі доступні IP-адреси, які може призначити сервер DHCP. Після розподілу усіх IP-адрес, які можуть бути призначені DHCP-сервером, відбувається відмова в обслуговуванні (DoS-атака), оскільки нові клієнти не можуть отримати доступ до мережі. *DoS-атака* – це будь-яка атака, яка використовується для перевантаження конкретних пристроїв і мережевих сервісів несанкціонованим трафіком, внаслідок чого дозволений трафік не може отримати доступ до цих ресурсів.

При DHCP-спуфінге зловмисник настроює в мережі несправжній DHCP-сервер, щоб видавати для клієнтів DHCP-адреса. Мета цієї атаки – змусити клієнтів використати неправдиву службу доменних імен (DNS), а також використати вузол або обладнання зловмисника в якості шлюзу за умовчанням.

Перед DHCP-спуфінгом часто використовується атака виснаження ресурсів DHCP для відмови обслуговування санкціонованого DHCP-сервера, завдяки чому набагато простіше впровадити в мережу фіктивний DHCP-сервер.

Для зниження ризику атак DHCP слід використати функції відстежування DHCP і безпеці інтерфейсу на комутаторах Cisco Catalyst. Ці функції будуть розглянуті детально пізніше.

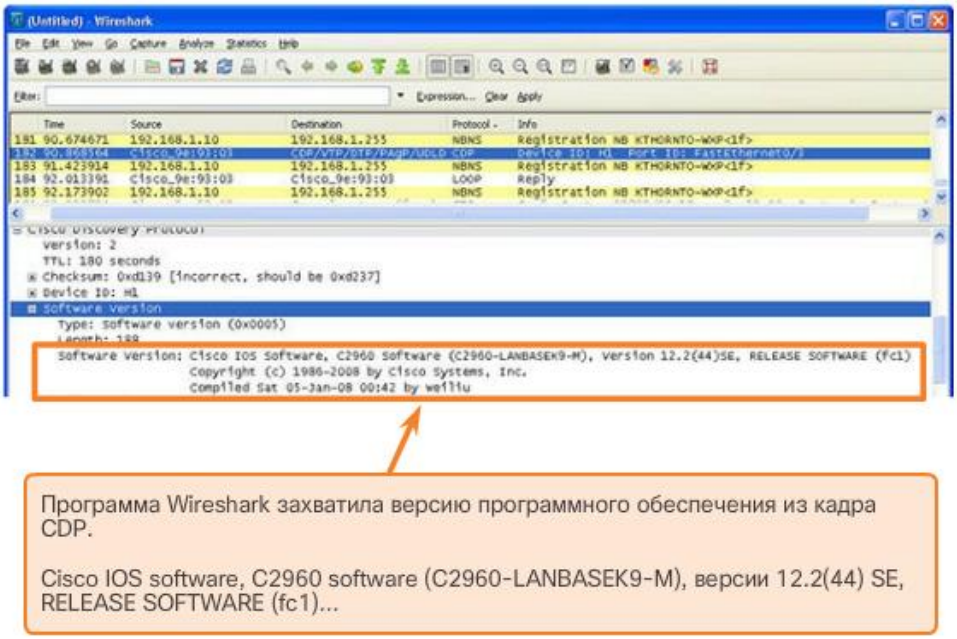
Використання вразливостей протоколу CDP

Протокол виявлення Cisco (CDP) – це запатентований протокол, який може бути налагоджений на будь-яких обладнаннях Cisco. CDP виявляє інші обладнання Cisco з прямим підключенням, завдяки чому пристрої можуть автоматично настроювати свої підключення. В деяких випадках це спрощує процедури налаштування і підключення.

За умовчанням усі інтерфейси на більшості маршрутизаторів і комутаторів Cisco налагоджені для використання CDP. Інформація CDP вирушає в періодичних незашифрованих широкомовних розсилках і оновлюється локально у базі даних CDP кожного пристрою. Оскільки CDP є протоколом 2-го рівня, повідомлення CDP не поширюються маршрутизаторами.

CDP містить наступну інформацію про пристрій: IP-адрес, версію IOS, а також відомості про платформу, можливості і native VLAN. Цією інформацією може скористатися зловмисник для пошуку способів атаки мережі, як правило, у формі атаки DoS.

На Рисунок 3.3 показана частина даних, захоплених програмою Wireshark, в яких відображений зміст пакету CDP. Версія ПО Cisco IOS, виявлена через CDP, дозволить зловмисникові визначити, чи є слабкі місця в системі безпеки, характерні саме для цієї версії IOS. Крім того, у зв’язку з тим, що CDP не аутентифікований, зловмисник може створювати фіктивні пакети CDP і відправляти їх на обладнання Cisco з прямим підключенням.



Программа Wireshark захватила версию программного обеспечения из кадра CDP.

Cisco IOS software, C2960 software (C2960-LANBASEK9-M), версии 12.2(44) SE, RELEASE SOFTWARE (fc1)...

Рисунок 3.3

Атаки Telnet

Telnet – це незахищений протокол, який може бути використаний для отримання зловмисником видаленого доступу до мережевого обладнання Cisco. Існують інструменти, що дозволяють зловмисникові почати атаку методом повного перебору проти каналів vty на комутаторі.

Атака методом повного перебору (метод грубої сили)

На першому етапі атаки методом повного перебору зловмисник використовує список поширених паролів і програму, розроблену для встановлення сеансу Telnet за допомогою усіх слів зі списку. Якщо пароль не виявлений на першому етапі, починається другий етап, в якому зловмисник використовує програму, що створює послідовні поєднання символів, намагаючись підібрати пароль. Маючи достатню кількість часу, таким методом зловмисник може зламати практично усі використовувані паролі.

DoS-атака протоколу Telnet

Telnet також може бути схильний DoS-атаці. При атаках типу «відмови в обслуговуванні» (DoS-атаках) зловмисник використовує уразливості серверного програмного забезпечення Telnet, запущеного на комутаторі, що робить сервіс Telnet недоступним. Цей тип атак блокує видалений доступ адміністраторові до функцій управління комутатором. Такі атаки можуть бути об’єднані з іншими прямими атаками на мережу як частину скоординованої спроби закрити мережевому адміністраторові доступ до базових пристроїв на час пролому в системі безпеки.

Атаки відмови в обслуговуванні (DoS) є поширеними мережевими атаками. Атака DoS призводить до певної перерви в обслуговуванні користувачів, пристроїв або додатків.

Існує два основних джерела атак DoS:

переважна кількість трафіку - це коли мережа, хост або програма не можуть обробляти величезну кількість даних, що призводить до падіння системи або її дуже повільної роботи.

зловмисно відформатовані пакети - це коли зловмисно відформатовані пакети пересилаються на хост або програму, і одержувач не може їх обробити .

DDoS атаки - це розподілена атака DoS (DDoS), вона відбувається з кількох координованих джерел. Атака DDoS може використовувати сотні або тисячі джерел, які в DDoS-атак на основі IoT.

У атаках DDoS беруть участь наступні компоненти :

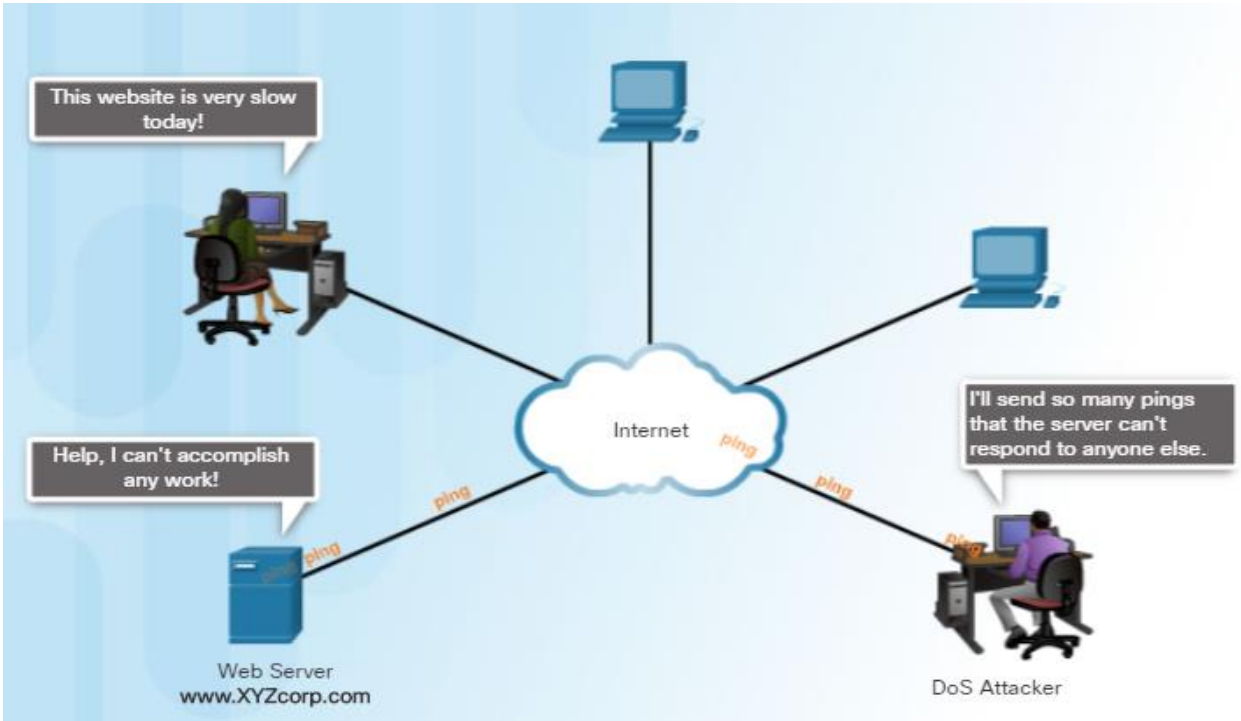
зомбі - група зкомпрометованих хостів (тобто агентів). Ці хости виконують шкідливий код, Котрий називають роботами (тобто ботами), зловмисне програмне забезпечення зомбі постійно намагається самостійно поширюватися як черв'як.

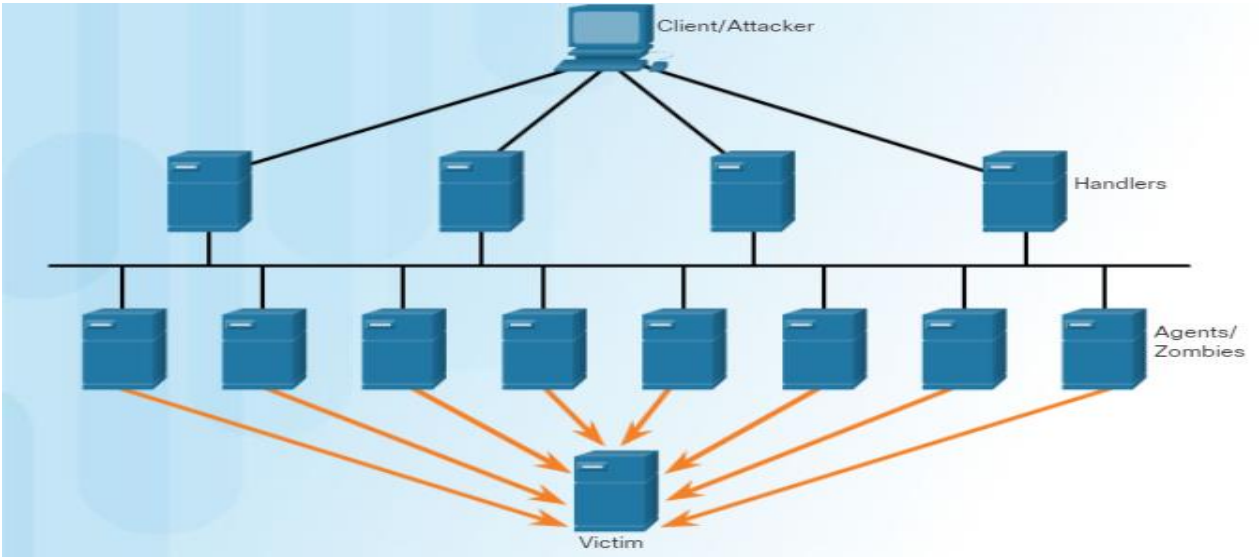
боти - це шкідливі програми, призначені для зараження хоста і спілкування з системою обробника, вони можуть реєструвати натискання клавіш, збирати аролі, зберігати і аналізувати пакети та ін.

ботнет - відноситься до групі зомбі, які були інфіковані з використанням шкідливих програм (тобто ботів) і керовані обробниками.

Обробники - головний командно-контрольний (CnC або C2) сервер управління групами зомбі. Засновник бот-мережі може використовувати Internet Relay Chat (IRC) або веб на сервері C2 для віддаленого управління зомбі.

Botmaster - це зловмисник, що керує ботнетом і обробниками.





3.1.3. Мережеві та протокольні засоби забезпечення конфіденційності передачі інформації (ACL, NAT)

Список контролю доступу (ACL) – це послідовний список правил дозволу або заборони, застосованих до інформаційних пакетів, відповідно до зазначених в них IP-адрес або ознак протоколів більш високого рівня. ACL-списки дозволяють ефективно контролювати трафік мережі на вході або виході маршрутизаторів. ACL-списки можна настроювати для усіх протоколів мережі, що маршрутизуються.

ACL-список реалізують послідовністю команд IOS, які визначають виходячи з інформації в заголовку пакету, чи пересилає маршрутизатор пакети чи скидає їх. ACL-списки є однією з найбільш використовуваних функцій операційної системи Cisco IOS.

На рисунку 3.4 наводиться приклад топології з використанням ACL-списків.

Залежно від конфігурації ACL-списки виконують наступні завдання:

– *Обмеження мережного трафіку* для підвищення продуктивності мережі. Наприклад, якщо корпоративна політика забороняє відеотрафік в мережі, необхідно настроїти і застосувати ACL-списки, блокуючі цей тип трафіку. Подібні заходи значно знижують навантаження на мережу і підвищують її продуктивність.

– *Управління потоком трафіку.* ACL-списки можуть обмежувати доставку оновлень маршрутизації. Такі налаштування мережі, дозволяють уникнути зайвого використання смуги пропускання.

– *Забезпечення базового рівня безпеки* відносно доступу до мережі. ACL-списки можуть відкрити доступ до частини мережі одному вузлу і закрити його

для інших вузлів. Наприклад, доступ до мережі відділу кадрів може бути обмежений і дозволений тільки авторизованим користувачам.

– *Фільтрування трафіку на основі його типу*. Наприклад, ACL-список може дозволяти трафік електронної пошти, але при цьому блокувати увесь трафік протоколу Telnet.

– Списки контролю доступу здійснюють сортування вузлів в цілях *дозволу або заборони доступу до мережних служб*. За допомогою ACL-списків можна дозволяти або забороняти доступ до певних типів файлів, наприклад FTP або HTTP.

Рисунок 3.4 – Топології з використанням ACL-списків

За замовчанням ACL-списки не конфігуровані на маршрутизаторі, тому маршрутизатор не фільтрує трафік. Трафік, що поступає на маршрутизатор, маршрутизується виключно на основі інформації таблиці маршрутизації. *Проте якщо ACL-список використовується на інтерфейсі, маршрутизатор виконує додаткове завдання, оцінюючи усі мережні пакети, що проходять через інтерфейс, з метою визначення дозволу пересилки пакету.*

NAT (Network Address Translation – «перетворення мережних адрес») забезпечує перетворення приватних адрес в публічні адреси. Це дозволяє пристрою з приватною IPv4-адресою діставати доступ до ресурсів поза своєю приватною мережею, включаючи ресурси в Інтернеті. У поєднанні з приватними IPv4-адресами NAT продемонстрував свою доцільність відносно економії публічних IPv4-адресов. Одна публічна IPv4-адреса може спільно використовуватися з сотнями, навіть тисячами пристроїв, для кожного з яких налагоджена унікальна приватна IPv4-адреса.

Термінологія NAT

У термінології NAT під «внутрішньою мережею» мається на увазі набір мереж, чиї адреси транслюватимуться. Термін «зовнішня мережа» відноситься до усіх інших мереж.

При використанні NAT IPv4-адреси представляють різні точки призначення залежно від того, де вони знаходяться: в приватній або в публічній мережі (Інтернет), а також від того, чи є трафік таким, що входить або вихідним.

У NAT передбачені 4 типи адрес:

– внутрішня локальна адреса;
– внутрішня глобальна адреса;
– зовнішня локальна адреса;
– зовнішня глобальна адреса.

При визначенні використовуваного типу адреси важливо пам’ятати, що термінологія NAT завжди застосовується з точки зору пристрою, адреса якого транслюватиметься:

– **Внутрішня (inside) адреса** – це адреса пристрою, яка перетворюється механізмом NAT.
– **Зовнішня (ouside) адреса** – це адреса обладнання призначення.

У рамках NAT по відношенню до адрес також використовується поняття локальності або глобальності:

– **Локальна адреса** – це будь-яка адреса, яка перебуває у внутрішній частині мережі.
– **Глобальна адреса** – це будь-яка адреса, яка перебуває в зовнішній частині мережі.

На рисунку 3.5 внутрішньою локальною адресою ПК 1 є 192.168.10.10. З точки зору ПК 1 веб-сервер використовує зовнішню адресу 209.165.201.1. Якщо пакети вирушають від ПК 1 на глобальну адресу веб-сервера, внутрішня локальна адреса ПК 1 перетворюється на 209.165.200.226 (внутрішня глобальна адреса). Адреса зовнішнього пристрою зазвичай не перетворюється, оскільки ця адреса зазвичай вже є публічною IPv4-адресою.

Отже, для ПК 1 використовуються різні локальна і глобальна адреси, а для веб-сервера в обох випадках використовується одна публічна IPv4-адреса. З точки зору веб-сервера трафік, вихідний від ПК 1, є таким, що надходить з адреси 209.165.200.226, внутрішньої глобальної адреси.

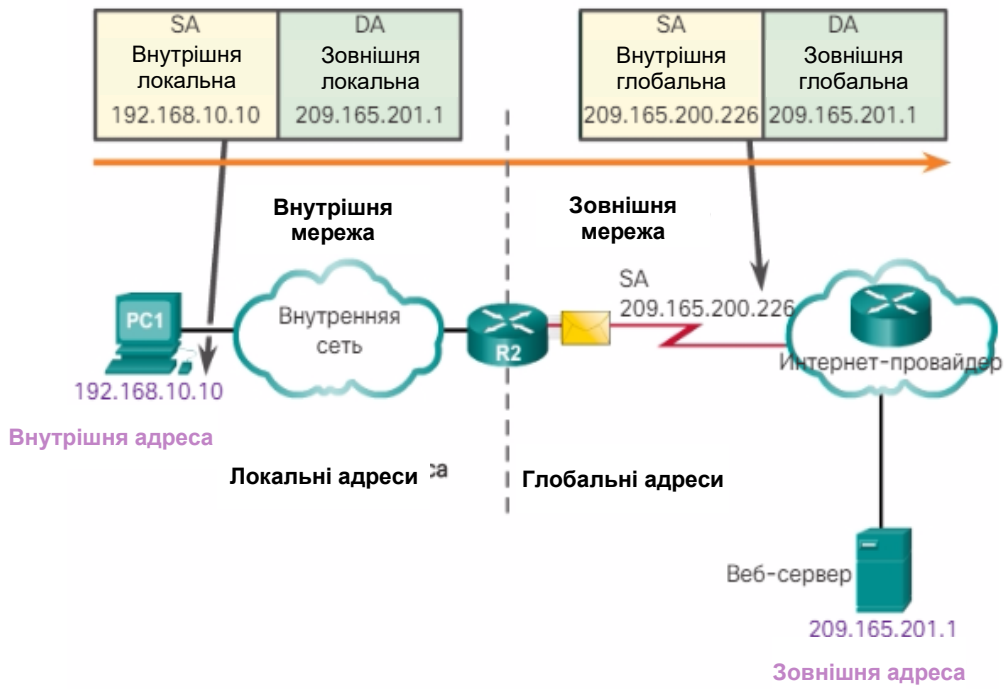


Рисунок 3.5 – Типи адрес NAT

Маршрутизатор NAT (R2 на рисунку 3.5) є точкою розмежування між внутрішньою і зовнішньою мережами, а також між локальними і глобальними адресами.

Існують три механізми перетворення:

– **Статичне перетворення (статичний NAT)** – це взаємно-однозначна відповідність між локальною і глобальною адресами.

– **Динамічне перетворення (динамічний NAT)** – це зіставлення адрес за схемою «багато до багатьох» між локальними і глобальними адресами.

Перетворення адреси і номера порту (PAT) – це зіставлення адрес за схемою «багато до одного» між локальними і глобальною адресами. Цей метод також називається перевантаженням (NAT з перевантаженням).

3.1.4. Протокол резервування першого переходу FHRP

Призначення, властивості протоколу FHRP

Існує проблема виходу з ладу маршрутизатора, що є шлюзом за замовчуванням. У разі збою маршрутизатора або інтерфейсу маршрутизатора (який служить шлюзом за замовчуванням) вузли, налаштовані за допомогою цього шлюзу, ізолюються від зовнішніх мереж. У комутованої мережі кожен клієнт отримує тільки один шлюз за замовчуванням. Неможливо використовувати другий шлюз, навіть якщо існує другий шлях для передачі пакетів з локального сегмента. Жоден з хостів не зможе відправляти повідомлення за межі локальної мережі. Буде потрібно якийсь час, щоб цей шлюз знову запрацював.

Потрібен механізм для забезпечення альтернативних шлюзів за замовчуванням в комутованих мережах, де два або більше маршрутизатора підключені до одних і тих же VLAN. Цей механізм забезпечується протоколами резервування першого переходу - First Hop Redundancy Protocols (FHRPs).

Топологія фізичної мережі на рисунку 3.6 показує два комутатора, маршрутизатори, ПК і сервер. Маршрутизатор R1 відповідає за маршрутизацію пакетів від ПК1. Якщо R1 стає недоступним, протоколи маршрутизації можуть динамічно сходитися. R2 тепер направляє пакети із зовнішніх мереж, які пройшли б через R1. Однак трафік з внутрішньої мережі, пов'язаної з R1, включаючи трафік з робочих станцій, серверів і принтерів, налаштованих з R1 в якості шлюзу, як і раніше відправляється на R1 і відкидається.

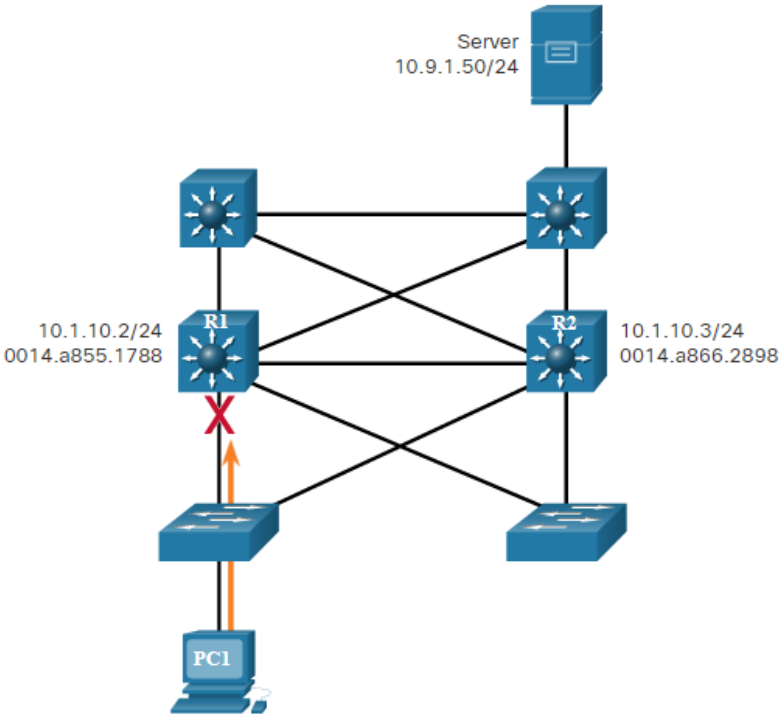


Рисунок 3.6 – Топологія фізичної мережі

Примітка. При розгляді резервування маршрутизатора не існує функціональної різниці між комутатором рівня 3 і маршрутизатором на рівні розподілу. На практиці комутатор рівня 3 зазвичай виступає в якості шлюзу для кожної VLAN в комутованій мережі. Це опис зосереджено на функціональності маршрутизації, незалежно від використовуваного фізичного пристрою.

Кінцеві пристрої зазвичай настроюються з одною IPv4-адресою для шлюзу. Ця електронна адреса не змінюється при зміні топології мережі. Якщо цей IPv4-адрес шлюзу за замовчуванням не може бути досягнутий, локальний пристрій не може відправляти пакети з сегмента локальної мережі і відключається від інших мереж. Навіть якщо існує резервний маршрутизатор, який може служити шлюзом за замовчуванням для цього сегмента, не існує динамічного методу, за допомогою якого ці пристрої можуть визначати адресу нового шлюзу.

Примітка. Пристрої IPv6 отримують адресу шлюзу динамічно з оголошення маршрутизатора ICMPv6. Однак при використанні протоколу FHRP пристрої IPv6 виграють завдяки більш швидкому переключенню на новий шлюз.

У протоколі FHRP як шлюз для робочих станцій в певному сегменті налаштований IPv4-адрес віртуального маршрутизатора. Коли кадри відправляються з хост-пристроїв на шлюз за замовчуванням, хости використовують ARP для визначення MAC-адреси, пов'язаного з IPv4-адресою шлюзу. Дозвіл ARP повертає MAC-адресу віртуального маршрутизатора. Кадри, відправлені на MAC-адресу віртуального маршрутизатора, можуть потім фізично оброблятися активним в даний момент маршрутизатором в групі віртуальних маршрутизаторів (рисунок 3.7).

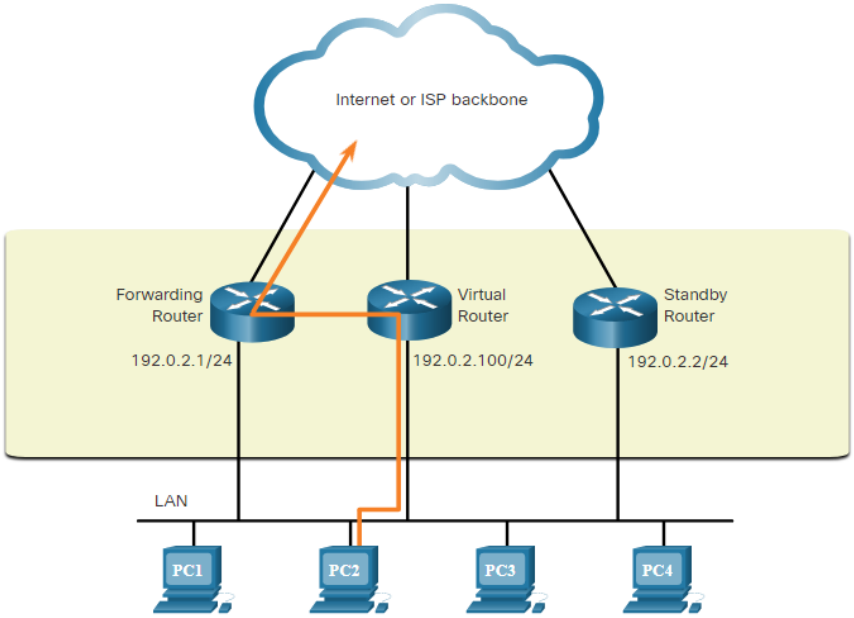


Рисунок 3.7 – Застосування віртуальних маршрутизаторів

Протокол використовується для ідентифікації двох або більше маршрутизаторів як пристроїв, що відповідають за обробку кадрів, що відправляються на MAC-адресу або IP-адреса одного віртуального маршрутизатора. Хост-пристрої відправляють трафік на адресу віртуального маршрутизатора. Фізичний маршрутизатор, який пересилає цей трафік, не важлива для хост-пристроїв.

Протокол резервування забезпечує механізм для визначення, який маршрутизатор повинен відігравати активну роль в пересиланні трафіку. Він також визначає, коли роль переадресації повинен прийняти резервний маршрутизатор. Перехід від одного маршрутизатора пересилання до іншого неважливий для кінцевих пристроїв. Така здатність мережі динамічно відновлюватися після збою пристрою, який виступає в якості шлюзу, називається резервуванням першого переходу.

Кроки для аварійного перемикання маршрутизатора

При збої активного маршрутизатора протокол резервування переводить резервний маршрутизатор в нову роль активного маршрутизатора, як показано на рисунку 3.8. Це кроки, які відбуваються при збої активного маршрутизатора:

1. Резервний маршрутизатор перестає бачити привітальні повідомлення від ретранслює маршрутизатора.

2. Резервний маршрутизатор приймає на себе роль перенаправляє маршрутизатора.

3. Оскільки новий маршрутизатор пересилання приймає як IPv4, так і MAC-адресу віртуального маршрутизатора, хост-пристрої не бачать збоїв в обслуговуванні.

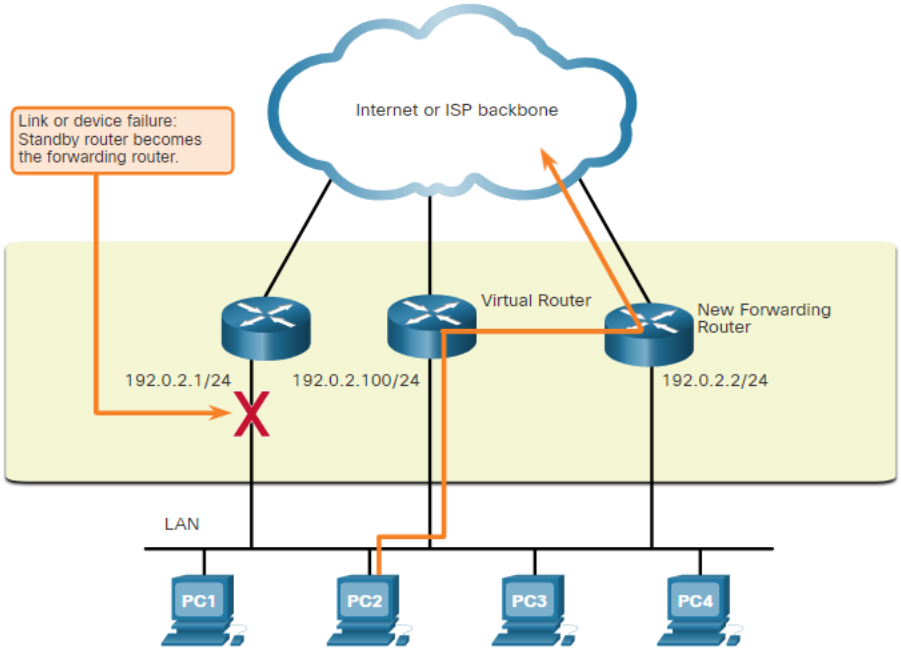


Рисунок 3.8 – Аварійне перемикання маршрутизатора

Різновиди протоколів FHRP

Далі перераховані всі варіанти, доступні для реалізації FHRP.

Протокол маршрутизатора з гарячим резервуванням (Hot Standby Router Protocol, HSRP)

HRSP є пропрієтарним FHRP, розробленим Cisco, який забезпечує прозоре перемикання при відмові пристрою IPv4 першого переходу. HSRP забезпечує високу доступність мережі, забезпечуючи резервування маршрутизації на першому переході для вузлів IPv4 в мережах, налаштованих з адресою шлюзу IPv4. HSRP використовується в групі маршрутизаторів для вибору активного пристрою і резервного пристрою. У групі інтерфейсів пристроїв активний

пристрій - це пристрій, який використовується для маршрутизації пакетів, резервне пристрій - це пристрій, який вступає в роботу при відмові активного пристрою або при виконанні встановлених умов. Функція резервного маршрутизатора HSRP полягає в тому, щоб відстежувати робочий стан групи HSRP і швидко брати на себе відповідальність за пересилку пакетів в разі збою активного маршрутизатора.

HSRP для IPv6

Це власний FHRP від Cisco, який забезпечує ті ж функції HSRP, але в середовищі IPv6. Група HSRP IPv6 має віртуальний MAC-адресу, отриманий з номера групи HSRP, і віртуальний локальний IPv6-адреса, отриманий з віртуального MAC-адреси HSRP. Періодичні оголошення маршрутизатора (RA) відправляються для віртуального локального адреси каналу IPv6 HSRP, коли група HSRP активна. Коли група стає неактивною, ці RA зупиняються після відправки остаточного RA.

Протокол резервування віртуальних маршрутизаторів версії 2 (Virtual Router Redundancy Protocol, VRRPv2)

Це загальнодоступний протокол, який динамічно розподіляє відповідальність за один або кілька віртуальних маршрутизаторів на маршрутизатори VRRP в локальній мережі IPv4. Це дозволяє декільком маршрутизаторів в каналі множинного доступу використовувати один і той же віртуальний адреса IPv4. Маршрутизатор VRRP налаштований для запуску протоколу VRRP в поєднанні з одним або декількома іншими маршрутизаторами, підключеними до локальної мережі. У конфігурації VRRP один маршрутизатор вибирається в якості майстра віртуального маршрутизатора, а інші маршрутизатори виступають в якості резервних копій на випадок збою майстри віртуального маршрутизатора.

VRRPv3

Дозволяє підтримувати адреси IPv4 і IPv6. VRRPv3 працює в мультивендорної середовищах і є більш масштабованим, ніж VRRPv2.

Протокол балансування навантаження шлюзу (Gateway Load Balancing Protocol GLBP)

Це власний протокол FHRP, розроблений компанією Cisco, який захищає трафік даних від несправного маршрутизатора або каналу, такого як HSRP і VRRP, а також дозволяє балансувати навантаження (також відому як розподілом навантаження) між групою резервних маршрутизаторів.

GLBP для IPv6

Це пропріетарний FHRP від Cisco, який забезпечує ті ж функціональні можливості GLBP, але в середовищі IPv6. GLBP для IPv6 забезпечує автоматичне резервне копіювання маршрутизатора для хостів IPv6, налаштованих з одним шлюзом за замовчуванням в локальній мережі. Кілька маршрутизаторів першого переходу в локальній мережі об'єднуються, щоб запропонувати один віртуальний маршрутизатор IPv6 першого переходу, одночасно розподіляючи навантаження пересилання пакетів IPv6.

Протокол виявлення маршрутизатора ICMP (Router Discovery Protocol, IRDP)

Зазначений в RFC 1256 протокол IRDP є застарілим рішенням FHRP. IRDP дозволяє хостам IPv4 знаходити маршрутизатори, які забезпечують підключення IPv4 до інших (нелокальним) IP-мереж.

Пріоритети, стан і робота маршрутизаторів з протоколом FHRP

HSRP протокол реалізований поверх стека протоколів TCP / IP, для доставки службової інформації використовується протокол UDP. Маршрутизатор або маршрутизовані комутатори, на яких налаштований і функціонує протокол HSRP, в рамках обміну службовою інформацією використовують так звані пакети вітання (hello packets). У свою чергу, дані пакети відправляються на IP-адреса групової розсилки 224.0.0.2 (HSRP Version 1) або на 224.0.0.102 (HSRP Version 2) по протоколу UDP на порт 1985.

HSRP забезпечує високу доступність мережі, забезпечуючи резервування маршрутизації в першому переході для IP-хостів у мережах, налаштованих з використанням IP-адреси шлюзу за замовчуванням. HSRP використовується в групі маршрутизаторів для вибору активного пристрою і резервного пристрою.

Роль активного і резервного маршрутизаторів визначається під час процесу вибору HSRP. За умовчанням як активного маршрутизатора обраний маршрутизатор з найбільшим за чисельністю адресою IPv4. Однак завжди краще контролювати, як ваша мережа буде працювати в нормальних умовах, ніж залишати це на волю випадку.

Пріоритет HSRP (HSRP Priority) може використовуватися для визначення активного маршрутизатора. Маршрутизатор з найвищим пріоритетом HSRP стане активним маршрутизатором. За замовчуванням пріоритет HSRP дорівнює 100. Якщо пріоритети рівні, як активного маршрутизатора вибирається маршрутизатор з найвищим цифровим адресою IPv4.

Щоб налаштувати маршрутизатор як активний маршрутизатор, використовують команду інтерфейсу standby priority. Діапазон пріоритету HSRP становить від 0 до 255.

HSRP заміщення

За замовчуванням після того, як маршрутизатор стає активним, він залишиться активним, навіть якщо інший маршрутизатор підключиться до мережі з більш високим пріоритетом HSRP.

Щоб новий процес вибору HSRP відбувався при підключенні маршрутизатора з більш високим пріоритетом, необхідно включити пріоритетне перемикання за допомогою команди інтерфейсу **standby preempt.**

Заміщення - це здатність маршрутизатора HSRP ініціювати процес переобрання. При включеному витіснення роль активного маршрутизатора буде виконувати маршрутизатор, підключений до мережі з більш високим пріоритетом HSRP.

Попередня установка дозволяє маршрутизатора ставати активним маршрутизатором, тільки якщо він має більш високий пріоритет. Маршрутизатор, включений для витіснення, з рівним пріоритетом, але з більш високим IPv4-адресою, що не буде витісняти активний маршрутизатор.

На рисунку 3.9 показана топологія в якій R1 був налаштований з пріоритетом HSRP 150, в той час як R2 має пріоритет HSRP за замовчуванням 100. Заміщення було включено на R1. R1 є активним маршрутизатором завдяки вищому пріоритету, а R2 є резервним маршрутизатором. Через збій харчування, що впливає тільки на R1, активний маршрутизатор більше не доступний, а резервний маршрутизатор R2 приймає на себе роль активного маршрутизатора. Після відновлення електропостачання R1 знову включається. Оскільки R1 має більш високий пріоритет і пріоритетне заміщення включено, це викличе новий процес виборів і R1 знову прийме роль активного маршрутизатора, а R2 повернеться до ролі резервного маршрутизатора.

Примітка. Якщо пріоритетне заміщення відключено, маршрутизатор, який завантажується першим, стане активним маршрутизатором, якщо під час процесу вибору іншихмаршрутизаторів в мережі не буде.

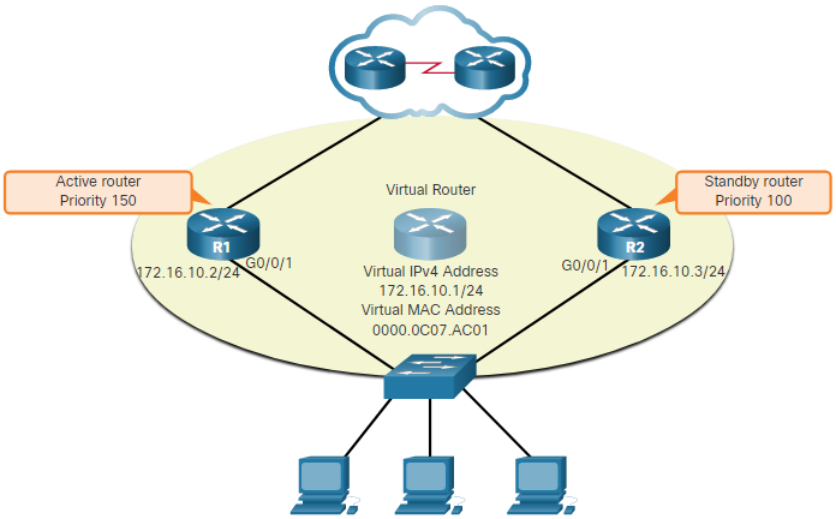


Рисунок 3.9 – Топологія мережі з заміщенням

Стани і таймери HSRP

Коли інтерфейс маршрутизатора налаштований з HSRP маршрутизатор відправляє і приймає пакети вітання HSRP, щоб почати процес визначення, яке стан він прийме в групі HSRP.

У таблиці 3.1 наведені стани HSRP.

Таблиця 3.1

Назва стану HSRP	**Опис стану HSRP**
Початкове **Initial**	Цей стан вводиться при зміні конфігурації або коли інтерфейс вперше стає доступним.
Вивчення **Learn**	Маршрутизатор не визначив віртуальний IP-адреса і ще не бачив вітальне повідомлення від активного маршрутизатора. У цьому стані маршрутизатор очікує відповіді від активного маршрутизатора.
Прослуховування **Listen**	Маршрутизатор знає віртуальний IP-адреса, але він не є ні активним, ні резервним маршрутизатором. Він слухає привітальні повідомлення від цих маршрутизаторів.
Оголошення **Speak**	Маршрутизатор періодично відправляє вітальні повідомлення і бере активну участь у виборі активного і / або резервного маршрутизатора.
Режим очікування **Standby**	Маршрутизатор є кандидатом на те, щоб стати наступним активним маршрутизатором, і періодично відправляє вітальні повідомлення.

Активні і резервні маршрутизатори HSRP відправляють привітальні пакети на груповий адресу групи HSRP за замовчуванням кожні 3 секунди (рисунок 3.10). Резервний маршрутизатор стане активним, якщо через 10 секунд він не отримає вітальне повідомлення від активного маршрутизатора. Можна зменшити ці настройки таймера, щоб прискорити аварійне перемикання або переривання. Проте, щоб уникнути збільшення завантаження ЦП і непотрібних змін стану очікування не варто встановлювати таймер вітання менше 1 секунди або таймер утримання менше 4 секунд.

0		7	15	23
Version	Type	Virtual Rtr ID	Priority	Count IP Addrs
Auth Type		Advet Int	Checksum	

IP Address (1)
…
IP Address (n)
Authentication Data (1)
Authentication Data (2)

Рисунок 3.10 – Структура пакета

Команди налаштування протоколу HSRP:

– standby **номер_групи** ip **ip-адреса** – вказує адресу ip-адреса віртуального маршрутизатора в зазначеній групі;

– standby version **версія_протокола** – вказує версію протоколу 1 або 2;

– standby **номер_групи** priority **значення_пріорітету** – вказує пріоритет маршрутизатора в зазначеній групі;

– standby **номер_групи** preempt – дозволяє перехоплення функцій якщо пріоритет вище ніж у активного маршрутизатора;

– standby **номер_групи** track **ім'я_інтерфейса_пріоритету** – знижує значення пріоритету маршрутизатора на зазначену величину пріоритету якщо інтерфейс недоступний. Значення відновиться якщо інтерфейс стане доступний;

– standby **номер_групи** timers **hellotime_сек holdtime_сек** – задає значення таймерів hellotime і holdtime;

– standby **номер_групи** authentication **пароль** – пароль для доступу до цієї групи.

Приклад налаштування (рисунок 3.11).

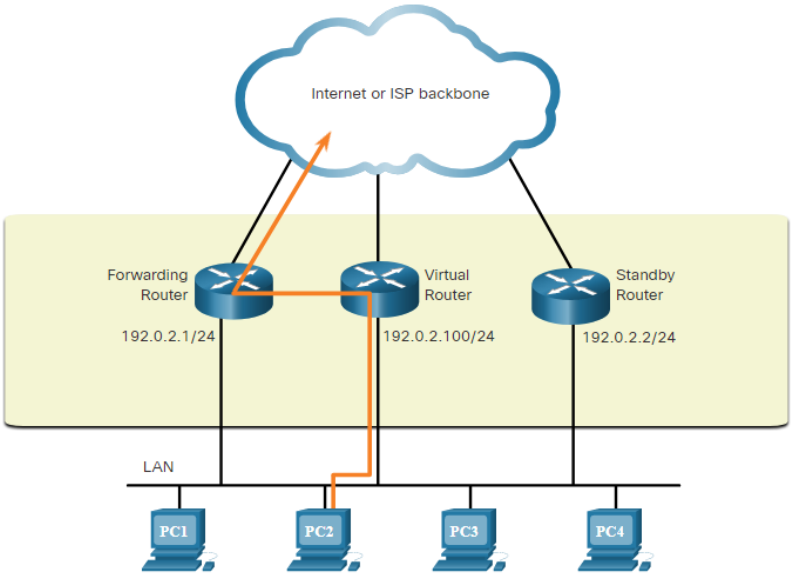


Рисунок 3.11 – Приклад налаштування

R1(config)# **interface g0/1**
R1(config-if)# **standby version 2**

R1(config-if)# **standby 1 ip 192.0.2.100**
R1(config-if)# **standby 1 priority 150**
R1(config-if)# **standby 1 preempt**

R3(config)# **interface g0/0**
R3(config-if)# **standby version 2**
R3(config-if)# **standby 1 ip 192.0.2.100**

R1# **show standby**
GigabitEthernet0/1 - Group 1 (version 2)
State is Active
4 state changes, last state change 00:00:30
Virtual IP address is 192.0.2.100
Active virtual MAC address is 0000.0C9F.F001
Local virtual MAC address is 0000.0C9F.F001 (v2 default)
Hello time 3 sec, hold time 10 sec
Next hello sent in 1.696 secs
Preemption enabled
Active router is local
Standby router is 192.0.2.2
Priority 150 (configured 150)
Group name is «hsrp-Gi0/1-1» (default)

R3# **show standby**
GigabitEthernet0/0 - Group 1 (version 2)
State is Standby
4 state changes, last state change 00:02:29
Virtual IP address is 192.0.2.100
Active virtual MAC address is 0000.0C9F.F001
Local virtual MAC address is 0000.0C9F.F001 (v2 default)
Hello time 3 sec, hold time 10 sec
Next hello sent in 0.720 secs
Preemption disabled
Active router is 192.0.2.1
MAC address is d48c.b5ce.a0c1
Standby router is local
Priority 100 (default 100)
Group name is «hsrp-Gi0/0-1» (default)

R1# **show standby brief**
P indicates configured to preempt.
|
Interface Grp Pri P State Active Standby Virtual IP
Gi0/1 1 150 P Active local 192.0.2.2 192.0.2.100

R3# **show standby brief**
P indicates configured to preempt.
|
Interface Grp Pri P State Active Standby Virtual IP
Gi0/0 1 100 Standby 192.0.2.1 local 192.0.2.100

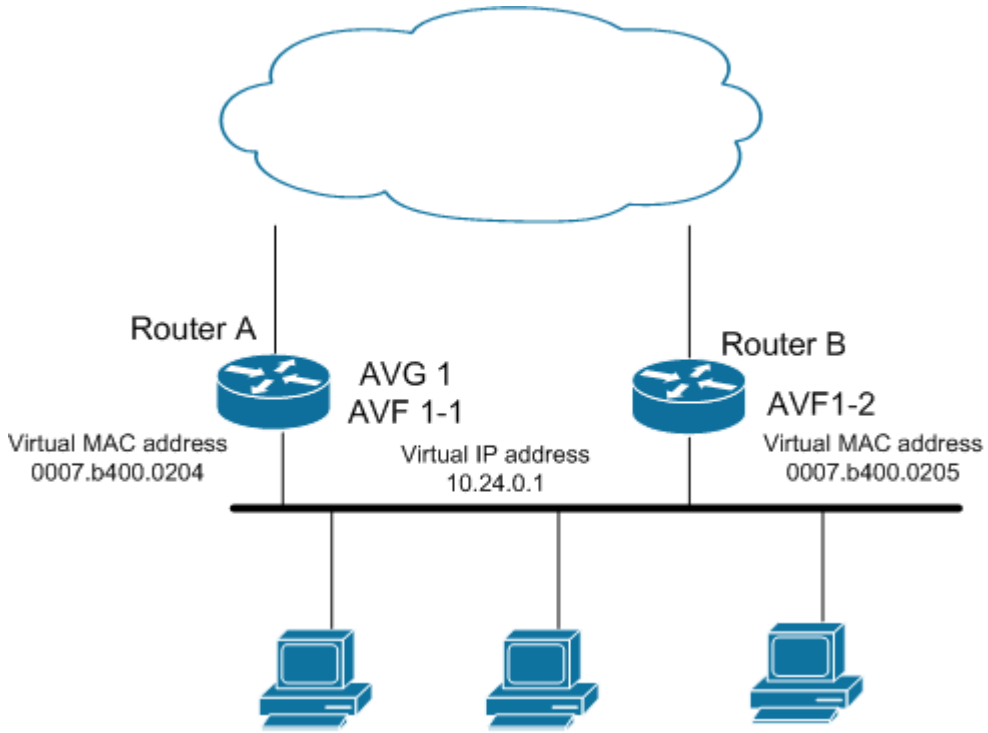
Балансування навантаження шлюза (GLBP)

GLBP працює аналогічно, але не ідентично іншим протоколам резервування шлюзу, такими як HSRP і VRRP.

Члени GLBP групи вибирають один шлюз який буде активним віртуальним шлюзом **active virtual gateway (AVG)** для цієї групи. Інші члени групи забезпечують резервування для AVG в разі якщо AVG стане недоступним. AVG призначає віртуальний MAC адреса для кожного члена GLBP групи. Кожен член групи бере участь в передачі пакетів, використовуючи віртуальний MAC адресу, виданий AVG. Цих членів групи називають **active virtual forwarders (AVFs)**. AVG відповідальний за видачу відповідей по протоколу **Address Resolution Protocol (ARP)** на запити до віртуального IP-адресою. Розподіл навантаження досягається тим що AVG відповідає на ARP запити використовуючи різні віртуальні MAC-адреси.

На рисунку 3.12 маршрутизатор A є AVG для GLBP групи, і відповідальний за віртуальний IP-адреса 10.24.0.1. Маршрутизатор A так само є AVF для віртуального MAC адреси 0007.b400.0204. Маршрутизатор B член тієї ж GLBP групи і призначений AVF для віртуального MAC адреси 0007.b400.0205. На клієнтах встановлюється шлюз з IP-адресою 10.24.0.1 і MAC адресою шлюзу 0007.b400.0204. У той час як на іншому клієнті MAC-адресу шлюзу за замовчуванням буде 0007.b400.0205 і таким чином маршрутизатор A буде розподіляти навантаження з маршрутизатором B.

GLBP підтримує до 4 маршрутизаторів в групі і до 1024 груп. Маршрутизатор відправляють один одному повідомлення hello кожні 3 секунди. Повідомлення відправляються на адресу 224.0.0.102, UDP порт 3222 (відправника і одержувача).



Рисунко 3.12 – Топологія мережі з балансуванням навантаження

GLBP Gateway Priority визначає роль, яку кожен маршрутизатор AVF грає в групі. Тобто за допомогою цієї властивості можна визначити послідовність вибору нового AVG, якщо старий AVG стане недоступним. Пріоритет можна визначити на кожному маршрутизаторі значенням від 1 до 255 командою: **glbp priority**. Маршрутизатор з великим пріоритетом стає AVG.

За замовчуванням схема вибору AVG тільки на основі пріоритету вимкнена. Запасний AVF стане AVG тільки якщо поточний AVG стане недоступним. Що б дозволити вибори AVG на основі пріоритету потрібно ввести команду: **glbp preempt**.

GLBP підтримує такі режими балансування навантаження:

– **None** – режим, при якому комутатор не забезпечує балансування навантаження. На всі запити клієнтів він відповідає своїм MAC-адресою. Другий комутатор починає роботу тільки після того як основний комутатор (AVG) вийде з ладу або стане недоступним.

– **Weighted load-balancing** – балансування навантаження проводиться відповідно до ваги кожного комутатора. Вага комутатора призначається адміністратором на кожному комутаторі окремо. Наприклад якщо в GLBP групі два комутатора, у AVG вага 70, а у AVF 140, то навантаження буде розподілятися 1: 2. Іншими словами з трьох отриманих запитів на MAC-адресу AVG один раз відповість своїм MAC-адресою і двічі MAC-адресою AVF комутатора.

– **Host-dependent load-balancing** – цей режим використовується в разі якщо є необхідність в реалізації трансляції адрес Network Address Translation (NAT), так як цей режим гарантує повернення клієнту того ж MAC-адреси AVF комутатора, який він використовував раніше і отже NAT сесія у клієнта не

переривається. Клієнти будуть отримувати ті ж MAC-адреси AVF до тих пір, поки кількість комутаторів в GLBP групі не зміниться.

– **Round-robin load-balancing** – режим використовується за умовчанням. В цьому режимі AVG видає MAC-адреси AVF поперемінно.

Налаштування GLBP на маршрутизаторах Cisco (рисунок 3.13):

– включення GLBP на інтерфейсі:

glbp <group> ip [ip-address [secondary]]

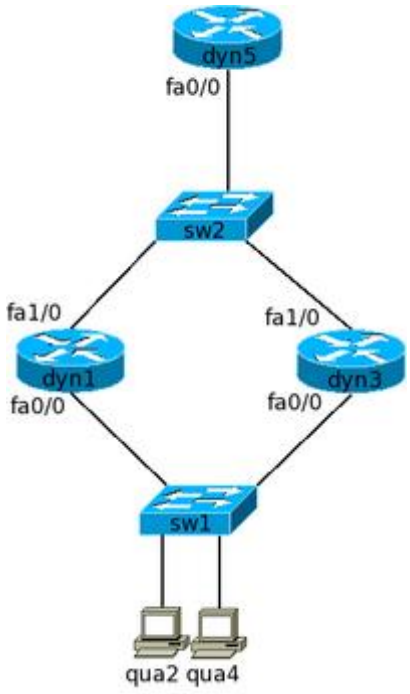


Рисунок 3.13 – Приклад налаштування

– включення режиму preempt GLBP:

У GLBP є два режими preempt:

– для **AVG** - за замовчуванням відключений,

– для **AVF** - за замовчуванням включений, з затримкою 30 секунд.

За замовчуванням режим preempt для **AVG** відключений. Тобто, backup virtual gateway може стати AVG тільки якщо поточний AVG вийде з ладу.

Включення режиму preempt для AVG:

dyn1(config-if)# glbp 1 preempt

Включення режиму preempt для AVF:

dyn1(config-if)# glbp 1 forwarder preempt [delay minimum <seconds>]

– налаштування методу балансування навантаження між маршрутизаторами (за замовчуванням round-robin):

dyn1(config-if)# glbp 1 load-balancing <host-dependent | round-robin | weighted>

– зміна методу балансування навантаження:

dyn1(config-if)# glbp 1 load-balancing host-dependent

– вимкнення балансування навантаження:

dyn1(config-if)# no glbp 1 load-balancing

– налаштування інтервалу між відправкою AVG повідомлень hello в групі GLBP:

dyn1(config-if)# glbp <group> timers [msec] <hellotime> [msec] <holdtime>
dyn1(config-if)# glbp <group> timers redirect <redirect> <timeout>

Параметри команди:

– redirect - налаштовує інтервал протягом якого AVG продовжує перенаправляти клієнтів AVF. За замовчуванням 600 секунд (10 хвилин);

– timeout - вказує інтервал в секундах до того як secondary virtual forwarder стане invalid. За замовчуванням 14,400 секунд (4 години).

Хоча діапазон значень параметра redirect дозволяє використовувати значення 0, фактично воно не повинно використовуватися. Це призведе до того, що таймер redirect не має терміну дії і, в разі виходу з ладу маршрутизатора, хости все одно будуть відправлятися на нього.

– Налаштування Object Tracking і вказівка об’єкта, який буде впливати на вагу GLBP, якщо інтерфейс вимикається, то вага зменшується на 11 (за замовчуванням на 10):

dyn1(config)# track 1 interface FastEthernet1/0 line-protocol
dyn1(config-if)# glbp 1 weighting track 1 decrement 11

– Налаштування порогових значень ваги, які регулюють чи маршрутизатор виконувати роль GLBP gateway:

dyn1(config-if)# glbp 1 weighting track 1 decrement 11

Для даного прикладу, початкове значення 200. Якщо інтерфейс fa1 / 0 вимикається, то значення стає 189. Так як воно менше, ніж значення **lower**, то маршрутизатор не може виконувати роль active forwarder. Коли інтерфейс

включиться, то значення стане знову 200. Так як це більше ніж значення **upper**, то маршрутизатор знову стає active forwarder.

За допомогою цих порогових значень можна прив'язувати певні значення ваг таким чином, щоб, наприклад, якщо два інтерфейси не доступні (з альтернативними шляхами до мережі), то тоді маршрутизатор не виконуватиме роль active forwarder.

– Налаштування *GLBP Client Cache:*

dyn1(config-if)# glbp 1 client-cache

За замовчуванням функція GLBP Client Cache відключена. Після включення функції AVG починає зберігати у себе інформацію про те які хости використовують який gateway.

– Перегляд інформації про GLBP Client Cache (зберігається тільки на AVG):

dyn1# show glbp detail

– Перегляд короткої інформації в групах:

dyn1# show glbp brief

– перегляд інформації про всі включені групи GLBP:

dyn1# show glbp

3.1.5. Протоколи комутованої мережі з агрегацією ліній

Принцип агрегації ліній EtherChannel

Якщо мережа включає в себе резервні комутатори та лінії, то для запобігання петель 2-го рівня налаштовується якась версія протоколу STP. Однак якщо виникає необхідність використовувати велику пропускну здатність в надлишкової мережі, то може допомогти протокол EtherChannel, який об'єднує кілька ліній між комутаторами в єдиний канал. Ці канали об'єднують паралельні лінії, забезпечуючи збільшену пропускну здатність, при цьому STP запобігатиме петлі. Є два протоколи EtherChannel - PAgP і LACP.

Однак кілька паралельних ліній, включених між комутаторами, будуть блокуватися протоколом Spanning Tree Protocol (STP), який за замовчуванням включений на пристроях рівня 2, таких як комутатори Cisco, для запобігання петель 2-го рівня (рисунок 3.14).

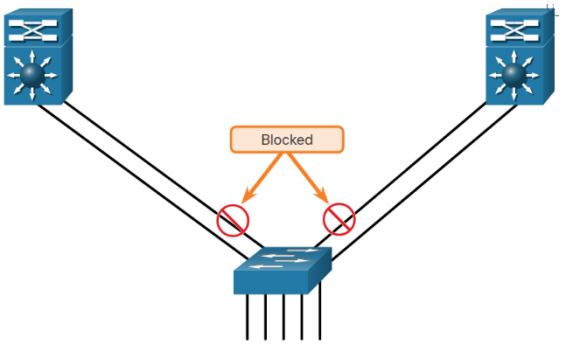


Рисунок 3.14 – Зображення прикладу агрегації ліній EtherChannel

Потрібно технологія агрегації ліній, яка дозволяє створювати паралельні лінії між пристроями, які не будуть блокуватися STP. Ця технологія відома як EtherChannel.

EtherChannel, **PortChannel** — технологія агрегації каналів, що була розроблена компанією Cisco Systems. Технологія дозволяє об'єднувати декілька фізичних каналів Ethernet в один логічний (віртуальний порт, який називається PortChannel) для збільшення пропускної здатності та підвищення надійності з'єднання. Можливо створити декілька окремих PortChannel на одному комутаторі.

Технологія EtherChannel надає ряд переваг:

1. Більшість завдань по налаштуванню можна виконувати на загальному інтерфейсі EtherChannel, а не на кожному окремому порту, забезпечуючи узгодженість конфігурації по всіх каналах.

2. EtherChannel спирається на існуючі порти комутатора. Немає необхідності докуповувати обладнання для більш швидкого з'єднання і забезпечення великої пропускної здатності (рисунок 3.18).

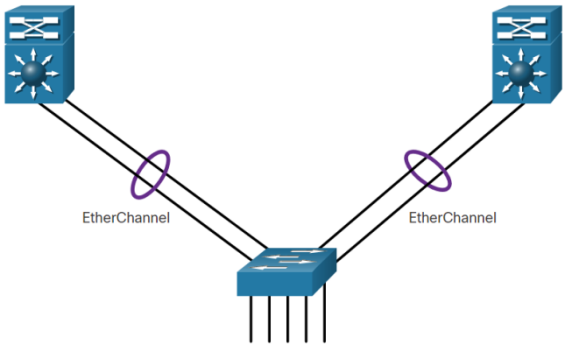


Рисунок 3.18 – Приклад формування EtherChannel

3. Балансування навантаження відбувається між лініями, які є частиною одного EtherChannel. Залежно від апаратної платформи може бути реалізований один або кілька методів розподілу навантаження. Ці методи включають розподіл навантаження по фізичних каналах, з урахуванням MAC-адреси джерела і призначення або з урахуванням IP-адреси джерела і призначення.

4. EtherChannel створює агрегацію, яка розглядається як одне логічне ланка. Коли між двома комутаторами існує кілька каналів EtherChannel, STP може заблокувати один з каналів, щоб запобігти петлі 2-го рівня. Коли STP блокує один з паралельних каналів, він блокує весь EtherChannel (всі порти, що належать цьому каналу EtherChannel). Якщо ж є тільки один канал EtherChannel, то все фізичні лінії в EtherChannel залишаються активними, тому що STP їх бачить як одну (логічну) лінію.

5. EtherChannel забезпечує резервування, тому що загальна лінія розглядається як одне логічне з'єднання, при цьому відключення однієї фізичної лінії в каналі не призводить до зміни топології. Тому перерахунок сполучного дерева не потрібно. Припускаючи, що принаймні одна фізична зв'язок присутній; EtherChannel залишається працездатним, навіть якщо його загальна пропускна здатність зменшується через втрату зв'язку в EtherChannel.

Обмеження реалізації EtherChannel:

1. В одному EtherChannel не можуть бути змішані різні типи інтерфейсів, наприклад, Fast Ethernet і Gigabit Ethernet не можна змішувати.

2. Кожен EtherChannel може складатися не більше ніж з восьми сумісних портів Ethernet. EtherChannel забезпечує полнодуплексную смугу пропускання до 800 Мбіт / с (Fast EtherChannel) або 8 Гбіт / с (Gigabit EtherChannel) між двома комутаторами або комутатором і хостом.

3. Комутатор Cisco Catalyst 2960 в даний час підтримує до шести каналів EtherChannel. Однак у міру розробки нових IOS і зміни платформ деякі карти і платформи можуть підтримувати збільшення кількості портів в каналі EtherChannel, а також підтримувати збільшення кількості каналів EtherChannel.

4. Конфігурація окремого порту в складі каналу EtherChannel повинна бути однаковою на обох пристроях. Якщо фізичні порти одного боку налаштовані як транки, фізичні порти іншого боку також повинні бути налаштовані як транки з тієї ж native VLAN. Крім того, всі порти в кожному каналі EtherChannel повинні бути налаштовані як комутовані порти.

5. Кожен EtherChannel має свій інтерфейс каналу логічного порту (рисунок 3.19) Конфігурація, що застосовується до інтерфейсу каналу порту, впливає на всі фізичні інтерфейси, які призначені цього інтерфейсу.

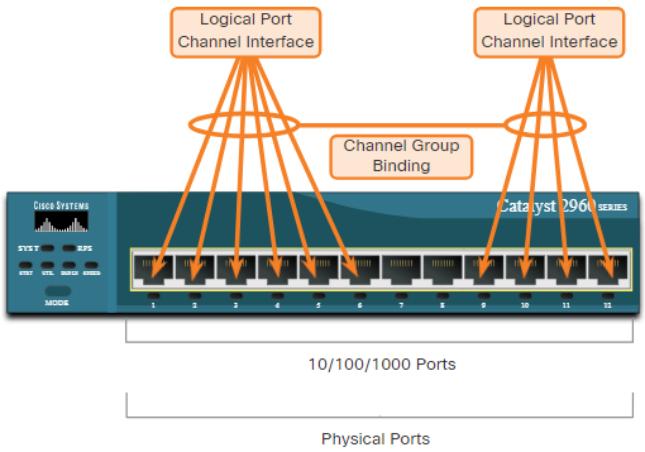


Рисунок 3.19 – Призначення інтерфейсів

Для того, щоб PortChannel міг існувати, необхідно, щоб всі вхідні в нього порти мали однакові параметри, а саме:

– Однакову швидкість (не можна створити portchannel, куди входили б, наприклад, FastEthernet0/1 і GigabitEthernet1/1 – або всі 10 Мбіт, або всі 100, або все – гігабіт і т.д.).

– Однакові налаштування дуплексу.

– Однакові налаштування VLAN-ів. У разі access портів – всі порти повинні бути в одному VLAN, в разі trunk портів – список дозволених VLAN (команда switchport trunk allowed vlan) так само повинен збігатися.

EtherChannel дає можливість об'єднувати від двох до восьми 100 Мбіт/с, 1 Гбіт/с або 10 Гбіт/с портів Ethernet (всі порти в каналі повинні мати однакову швидкість), який працює по витій парі або по оптоволокну, що дозволяє досягти результативної швидкості до 80 Гбіт/с. Додатково, від одного до восьми портів можуть бути неактивні і вмикатися в роботу при обриві з'єднання на одному з активних портів. За відсутності резервних портів, трафік автоматично розподіляється по з'єднанням що залишилися.

Канал може встановлюватися між маршрутизаторами, комутаторами і мережевими адаптерами на сервері (приклад на рисунок 3.20).

Усі мережеві адаптери, які є частиною каналу, отримують одну MAC-адресу, що робить канал прозорим для мережевих додатків. Балансування трафіку між портами виробляється на основі хеш-функції над MAC-адресою, IP-адресою або TCP і UDP портом джерела або одержувача. Таким чином, в деяких несприятливих випадках, весь трафік може передаватися по одному фізичному з'єднанню.

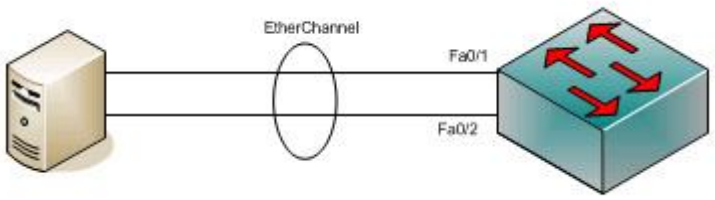


Рисунок 3.20 – EtherChannel між комутатором та сервером

При використанні протоколу STP разом з EtherChannel, усі з'єднання в каналі розглядаються як одне логічне і BPDU посилається тільки по одному з них. Спеціальний алгоритм дозволяє виявити невідповідності, коли один із комутаторів не налаштований для роботи з каналом.

При налаштуванні EtherChannel, порти на обох сторонах каналу додаються до нього вручну, або використовується протокол PAgP для автоматичної агрегації портів.

Коли два комутатора «домовляються» один з одним про використання між ними агрегованого каналу, застосовується один з двох протоколів: PAgP або LACP. PAgP - розроблявся спочатку cisco, а потім з'явився аналогічний відкритий стандарт LACP, який був оформлений у вигляді специфікації IEEE і

використовується як на Catalist, так і на комутаторах інших виробників. Іншими словами, в сучасних реаліях найкраще використовувати LACP, так як він сумісний з усіма, на відміну від PAgP. У функціональному ж плані протоколи аналогічні. PortChannel може бути налаштований в одному з трьох режимів: On, Active і Passive (в LACP) або On, Auto, Desirable (в PAgP відповідно). Для того, щоб між комутаторами піднявся і заробив portchannel, необхідно, щоб обидві сторони були налаштовані в режимі On, або одна була в режимі Active, а інша – Passive.

Протокол автоузгодження EtherChannel

Канали EtherChannel можуть бути сформовані шляхом узгодження з використанням одного з двох протоколів: протоколу агрегації портів (Port Aggregation Protocol, PAgP) або протоколу керування агрегацією каналів (Link Aggregation Control Protocol, LACP). Ці протоколи дозволяють портам зі схожими характеристиками формувати канал за допомогою динамічного узгодження з сусідніми комутаторами.

Примітка. Також можливо налаштувати статичний або безумовний EtherChannel без PAgP або LACP.

Робота PAgP

PAgP (вимовляється як «Pag - P») - це власний протокол Cisco, який допомагає в автоматичному створенні каналів EtherChannel. Коли канал EtherChannel налаштований з використанням PAgP, пакети PAgP відправляються між портами з підтримкою EtherChannel для узгодження формування каналу. Коли PAgP ідентифікує відповідні Ethernet-канали, він групує ці канали в EtherChannel. Потім EtherChannel додається в сполучна дерево як один порт.

При включенні PAgP також управляє EtherChannel. Пакети PAgP відправляються кожні 30 секунд. PAgP перевіряє узгодженість конфігурації і управляє додаванням каналів і збоями між двома комутаторами. Це гарантує, що при створенні EtherChannel всі порти мають однаковий тип конфігурації.

Примітка. У EtherChannel обов’язково, щоб всі порти мали однакову швидкість, налаштування дуплексу і інформацію про VLAN. Будь-яка модифікація порту після створення каналу також змінює всі інші порти каналу.

PAgP допомагає створити канал EtherChannel, визначаючи конфігурацію кожного боку і забезпечуючи сумісність каналів, щоб при необхідності можна було включити канал EtherChannel. Режими для PAgP наступні:

On – цей режим змушує передавати інтерфейс в канал без PAgP. Інтерфейси, налаштовані у включеному режимі, не обмінюються пакетами PAgP.

PAgP desirable – цей режим PAgP переводить інтерфейс в активний стан узгодження, в якому інтерфейс ініціює переговори з іншими інтерфейсами, відправляючи пакети PAgP.

PAgP auto – цей режим PAgP переводить інтерфейс в стан пасивного узгодження, в якому інтерфейс відповідає на пакети PAgP, які він отримує, але не ініціює узгодження PAgP.

Режими повинні бути сумісні з кожного боку. Якщо одна сторона налаштована на роботу в автоматичному режимі, вона перекладається в пасивний стан, чекаючи, поки інша сторона ініціює узгодження EtherChannel. Якщо для іншого боку також встановлено значення auto, узгодження ніколи не починається, і EtherChannel не створюється. Якщо все режими відключені за допомогою команди no або режим не налаштований, EtherChannel відключається.

Режим включення вручну поміщає інтерфейс в EtherChannel без будь-яких погоджень. Це працює, тільки якщо інша сторона також включена. Якщо інша сторона налаштована на узгодження параметрів через PAgP, EtherChannel не створюється, оскільки сторона, для якої задано режим включення, не виконує узгодження.

Приклад налаштувань режиму PAgP

Розглянемо два комутатора на рисунку 3.21. Чи встановлять S1 і S2 EtherChannel за допомогою PAgP, залежить від налаштувань режиму на кожній стороні каналу.

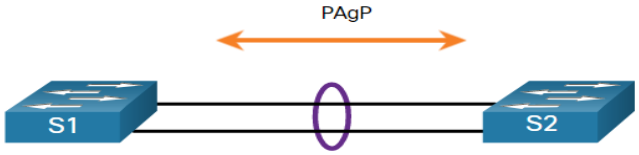


Рисунок 3.21 – Приклад налаштування

У таблиці 3.2 показана різна комбінація режимів PAgP на S1 і S2 і підсумковий результат встановлення каналу.

Таблиця 3.2

Режими PAgP

S1	S2	Channel Establishment
On	On	Yes
On	Desirable/Auto	No
Desirable	Desirable	Yes
Desirable	Auto	Yes
Auto	Desirable	Yes
Auto	Auto	No

Робота LACP

LACP є частиною специфікації IEEE (802.3ad), яка дозволяє об'єднувати декілька фізичних портів в єдиний логічний канал. LACP дозволяє комутатора автоматично погоджувати канал, відправляючи пакети LACP іншому комутатора. Розглянемо зміст відповідного кадру Ethernet від SW1. У Source-адреса він записує свій MAC-адресу, а в Destination-адреса (рисунок 3.22) мультікастового адреса 0180.C200.0002 (для протоколу PAgP використовується адреса 0100.0CCC.CCCC). Ця електронна адреса була зарезервована під протокол LACP. В інформаційному полі кадру передаються дані від LACP, це повідомлення використовується пристроями для багатьох цілей. Це синхронізація, збір, агрегація, перевірка активності і так далі, тобто у нього кілька функцій.

Перед початком взаємодії SW1 і SW2 вибирають собі віртуальний MAC-адресу. Зазвичай це найменший з наявних MAC-адрес на кожному з комутаторів (рисунок 3.23). І ось ці адреси вони будуть записувати в поля LACP (рисунок 3.24).

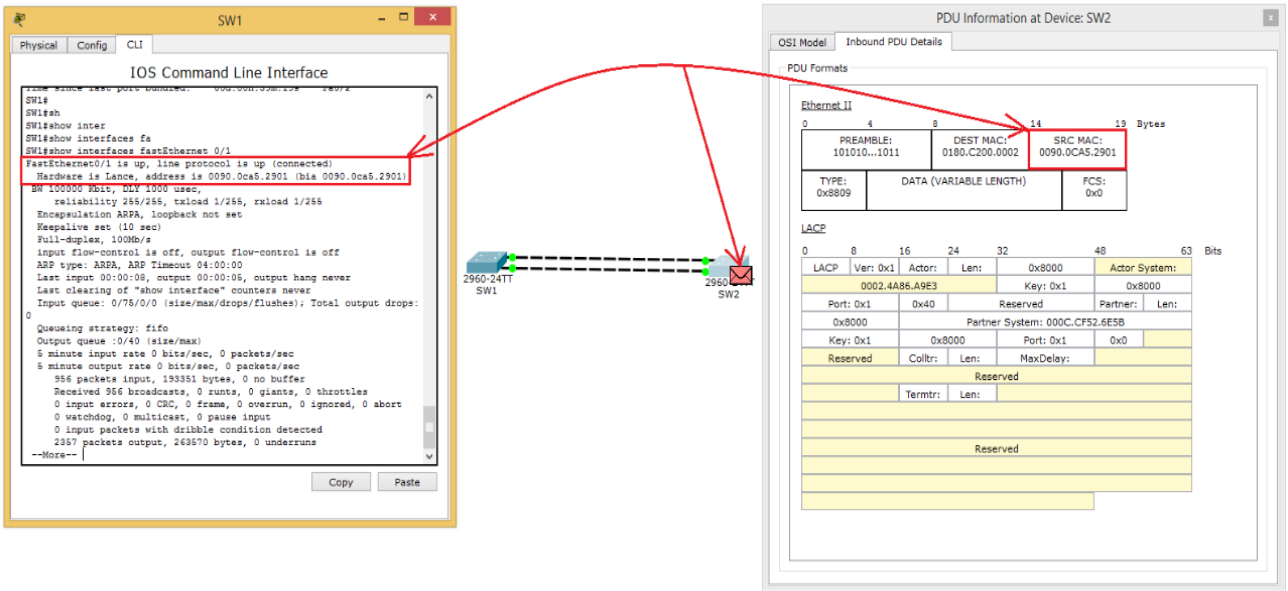


Рисунок 3.22 – Приклад налаштування

Рисунок 3.23 – Приклад налаштування

Він виконує функцію, аналогічну PAgP з Cisco EtherChannel. Оскільки LACP є стандартом IEEE, його можна використовувати для спрощення реалізації EtherChannels в мережах з обладнанням декількох виробників. На пристроях Cisco підтримуються обидва протоколи.

Примітка. LACP спочатку був визначений як IEEE 802.3ad. Однак LACP тепер визначено в більш новому стандарті IEEE 802.1AX для LAN і MAN.

LACP забезпечує ті ж переваги при веденні узгодження, що і PAgP. LACP допомагає створити канал EtherChannel, визначаючи конфігурацію кожного боку і перевіряючи їх сумісність, щоб при необхідності можна було включити канал EtherChannel.

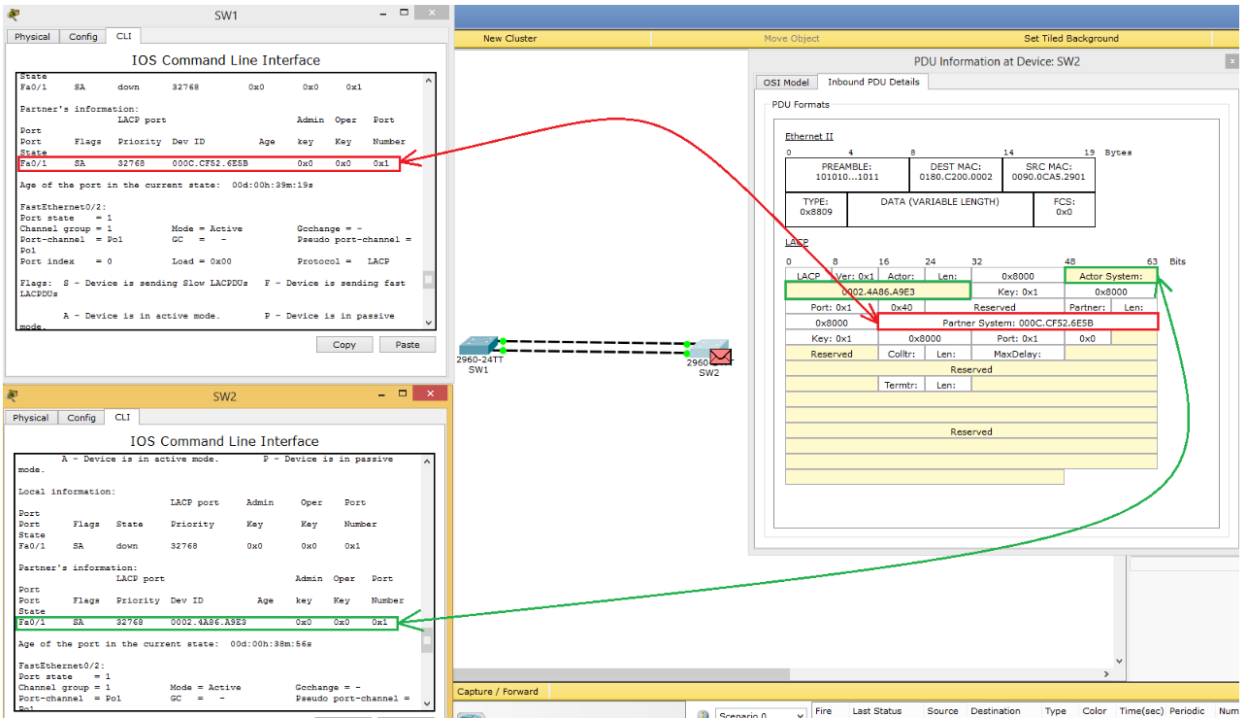


Рисунок 3.24 – Приклад налаштування

Режими для LACP наступні:

On – цей режим змушує передавати інтерфейс в канал без LACP. Інтерфейси, налаштовані у включеному режимі, не обмінюються пакетами LACP.

LACP active – цей режим LACP переводить порт в активний стан узгодження. У цьому стані порт ініціює переговори з іншими портами, відправляючи пакети LACP.

LACP passive – цей режим LACP переводить порт в стан пасивного узгодження. У цьому стані порт відповідає на пакети LACP, які він отримує, але не ініціює узгодження пакетів LACP.

Як і в разі PAgP, режими повинні бути сумісні з обох сторін, щоб канал EtherChannel міг сформуватися. Режим включення повторюється, оскільки він створює конфігурацію EtherChannel беззастережно, без динамічного узгодження PAgP або LACP.

LACP допускає вісім активних каналів, а також вісім резервних каналів. Резервна посилання стане активною в разі збою однієї з поточних активних посилань.

Приклад налаштувань режиму LACP

Розглянемо два комутатора на рисунку 3.25. Чи встановлять S1 і S2 EtherChannel з використанням LACP, залежить від налаштувань режиму на кожній стороні каналу.

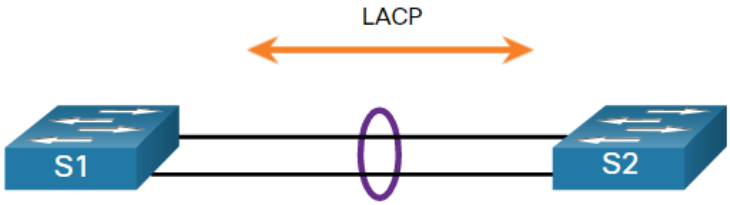


Рисунок 3.25 – Приклад налаштувань режиму LACP

У таблиці 3.3 показана різна комбінація режимів LACP на S1 і S2 і підсумковий результат встановлення каналу.

Таблиця 3.3

Режими LACP

S1	S2	Channel Establishment
On	On	Yes
On	Active/Passive	No
Active	Active	Yes
Active	Passive	Yes
Passive	Active	Yes
Passive	Passive	No

Порядок налаштування протоколів EtherChannel

Наступні рекомендації та обмеження корисні для настройки EtherChannel:

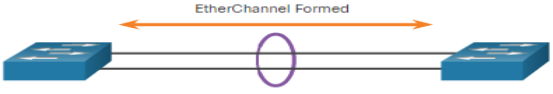
– Підтримка EtherChannel – все інтерфейси Ethernet повинні підтримувати EtherChannel без вимоги, щоб інтерфейси були фізично суміжними.

– Швидкість і дуплекс – налаштуйте все інтерфейси в EtherChannel для роботи з однаковою швидкістю і в одному дуплексному режимі.

– Відповідність VLAN – все інтерфейси в комплекті EtherChannel повинні бути призначені одній і тій же VLAN або налаштовані як транк.

– Діапазон VLAN – EtherChannel підтримує один і той же дозволений діапазон VLAN на всіх інтерфейсах транкінгового EtherChannel. Якщо допустимий діапазон VLAN не збігається, інтерфейси не формують EtherChannel, навіть якщо вони встановлені в auto або desirable режим.

На рисунку 3.26 показана конфігурація, яка дозволяє сформуватися EtherChannel між S1 і S2.



S1 Port Configurations	
Speed	1 Gbps
Duplex	Full
VLAN	10

S2 Port Configurations	
Speed	1 Gbps
Duplex	Full
VLAN	10

Рисунок 3.26 – Конфігурація, яка дозволяє сформувати EtherChannel між S1 і S2

EtherChannel формується, тільки коли настройки конфігурації збігаються на обох комутаторах.

Якщо ці параметри необхідно змінити, їх можна налаштувати в режимі настройки канального порту. Будь-яка конфігурація, яка застосовується до канального порту, також впливає на окремі інтерфейси. Однак конфігурації, які застосовуються до окремих інтерфейсів, не впливають на настройки канального порту. Тому внесення змін до конфігурації інтерфейсу, що є частиною каналу EtherChannel, може викликати проблеми сумісності інтерфейсу.

Канал порту може бути налаштований в режимі доступу, режим транка (найбільш поширеному) або на маршрутизації порту.

Приклад конфігурації LACP

EtherChannel відключений за замовчуванням і повинен бути налаштований. Топологія на рисунку 3.27 буде використана для демонстрації прикладу конфігурації EtherChannel з використанням LACP.

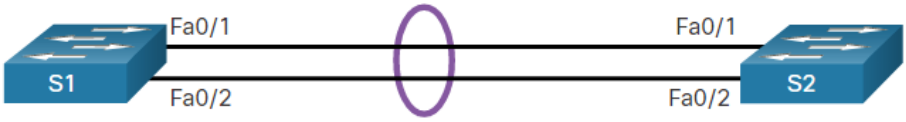


Рисунок 3.27 – Топологія для демонстрації прикладу конфігурації EtherChannel

Налаштування EtherChannel з LACP вимагає наступних трьох кроків (рисунок 3.28):

Крок 1. Вкажіть інтерфейси, які складають групу EtherChannel, за допомогою команди режиму глобальної конфігурації interface range interface. Ключове слово range дозволить вам вибрати кілька інтерфейсів і налаштувати їх всі разом.

Крок 2. Створіть інтерфейс каналу порту командою channel-group identifier mode active в режимі настройки діапазону інтерфейсів. Ідентифікатор вказує номер групи каналів. Ключові слова mode active активують це як конфігурацію LACP EtherChannel.

Крок 3. Щоб змінити налаштування ліній 2 рівня через інтерфейс канального порту, увійдіть в режим настройки інтерфейсу канального порту за допомогою команди interface port-channel + ідентифікатор інтерфейсу.

У прикладі на Рисунок 12 комутатор S1 налаштований з EtherChannel LACP. Канальний порт налаштований як магістральний інтерфейс з зазначеними дозволеними VLAN.

Перевірка EtherChannel

Завжди після налаштування пристрою в мережі необхідно перевірити його конфігурацію і, якщо є проблеми, їх потрібно буде локалізувати і усунути. Далі

розглянемо команди для перевірки, а також деякі поширені проблеми мережі з каналами EtherChannel і способи їх вирішення.

```
S1(config)# interface range FastEthernet 0/1 - 2
S1(config-if-range)# channel-group 1 mode active
Creating a port-channel interface Port-channel 1
S1(config-if-range)# exit
S1(config-if)# interface port-channel 1
S1(config-if)# switchport mode trunk
S1(config-if)# switchport trunk allowed vlan 1,2,20
```

Рисунок 3.28 – Налаштування EtherChannel з LACP

Команда **show interfaces port-channel** відображає загальний стан інтерфейсу канального порту (рисунок 3.29).

```
S1# show interfaces port-channel 1
Port-channel1 is up, line protocol is up (connected)
  Hardware is EtherChannel, address is c07b.bcc4.a981 (bia c07b.bcc4.a981)
  MTU 1500 bytes, BW 200000 Kbit/sec, DLY 100 usec,
     reliability 255/255, txload 1/255, rxload 1/255
(output omitted)
```

Рисунок 3.29 – Використання команди **show interfaces port-channel**

Якщо на одному пристрої налаштовано кілька інтерфейсів канальних портів, можна використовувати команду **show etherchannel summary** для відображення короткої інформації для кожного канального порту. На рисунку 3.30 показано, що в комутаторі налаштований один канал EtherChannel, номер групи 1, група використовує LACP. Канальна група складається з інтерфейсів FastEthernet0 / 1 і FastEthernet0 / 2. Група є каналом EtherChannel 2-го рівня, що відзначено буквами SU поруч з номером каналу порту.

```
S1# show etherchannel summary
Flags:  D - down        P - bundled in port-channel
        I - stand-alone s - suspended
        H - Hot-standby (LACP only)
        R - Layer3      S - Layer2
        U - in use      N - not in use, no aggregation
        f - failed to allocate aggregator
        M - not in use, minimum links not met
        m - not in use, port not aggregated due to minimum links not met
        u - unsuitable for bundling
        w - waiting to be aggregated
        d - default port
        A - formed by Auto LAG
Number of channel-groups in use: 1
Number of aggregators:           1
Group  Port-channel  Protocol    Ports
------+-------------+-----------+-----------------------------------------------
1      Po1(SU)         LACP      Fa0/1(P)    Fa0/2(P)
```

Рисунок 3.30 – Використання команди **show etherchannel summary**

Команда **show etherchannel port-channel** відображає інформацію про певний інтерфейсі канального порту (рисунок 3.31). У прикладі на Рисунок 15 інтерфейс Port Channel 1 складається з двох фізичних інтерфейсів, FastEthernet0 / 1 і FastEthernet0 / 2. Він використовує LACP в активному режимі. Він належним чином підключений до іншого комутатора з сумісною конфігурацією, тому показано, що канал порту використовується.

```
S1# show etherchannel port-channel
                Channel-group listing:
                ----------------------

Group: 1
----------
                Port-channels in the group:
                ---------------------------
Port-channel: Po1    (Primary Aggregator)
------------
Age of the Port-channel   = 0d:01h:02m:10s
Logical slot/port   = 2/1          Number of ports = 2
HotStandBy port = null
Port state          = Port-channel Ag-Inuse
Protocol            =   LACP
Port security       = Disabled
Load share deferral = Disabled
Ports in the Port-channel:
Index   Load   Port     EC state        No of bits
------+------+------+------------------+-----------
  0     00     Fa0/1    Active             0
  0     00     Fa0/2    Active             0
Time since last port bundled:    0d:00h:09m:30s    Fa0/2
```

Рисунок 3.31 – Використання команди **show etherchannel port-channel**

На будь-якому фізичному елементі інтерфейсу канальної групи EtherChannel команда **show interfaces etherchannel** може надати інформацію про роль даного інтерфейсу в EtherChannel. Рисунок 3.32 показує, що інтерфейс FastEthernet0 / 1 є частиною канальної групи EtherChannel 1. Протоколом для цього EtherChannel є LACP.

```
S1# show interfaces f0/1 etherchannel
Port state    = Up Mstr Assoc In-Bndl
Channel group = 1           Mode = Active          Gcchange = -
Port-channel  = Po1         GC   =   -             Pseudo port-channel = Po1
Port index    = 0           Load = 0x00            Protocol =   LACP
Flags:  S - Device is sending Slow LACPDUs   F - Device is sending fast LACPDUs.
        A - Device is in active mode.        P - Device is in passive mode.
Local information:
                            LACP port     Admin     Oper    Port
Port      Flags   State     Priority      Key       Number  State
Fa0/1     SA      bndl      32768         0x1       0x1     0x102       0x3D
Partner's information:
                  LACP port                       Admin  Oper   Port    Port
Port      Flags   Priority  Dev ID          Age   key    Key    Number  State
Fa0/1     SA      32768     c025.5cd7.ef00  12s   0x0    0x1    0x102   0x3Dof the port in the
current state: 0d:00h:11m:51sllowed vlan 1,2,20
```

Рисунок 3.32 – Використання команди **show interfaces etherchannel**

Поширені проблеми з конфігураціями EtherChannel

Всі інтерфейси в EtherChannel повинні мати однакову конфігурацію швидкості і дуплексного режиму, native і дозволені VLAN на з'єднувальних лініях і доступ до VLAN на портах доступу. Забезпечення цих змін значно зменшить можливі проблеми мережі, пов'язані з EtherChannel.

Поширені помилки EtherChannel можуть включать в себе наступні:

1. Призначені порти в EtherChannel не є частиною однієї VLAN або не налаштовані як транки.
2. Порти з різними native VLAN не можуть формувати EtherChannel.
3. Транкінг був налаштований на деяких портах, що складають EtherChannel, але не на всіх.
4. Якщо допустимий діапазон VLAN не збігається, порти не формують EtherChannel, навіть якщо PAgP встановлений в автоматичний або бажаний режим.
5. Параметри динамічного узгодження для PAgP і LACP не сумісні на різних кінцях EtherChannel.

Примітка: легко сплутати PAgP або LACP з DTP, оскільки всі вони є протоколами, що використовуються для автоматизації поведінки на магістральних каналах. PAgP і LACP використовуються для агрегації каналів (EtherChannel). DTP використовується для автоматизації створення магістральних посилань. Коли налаштована магістральна лінія EtherChannel, зазвичай спочатку налаштовується EtherChannel (PAgP або LACP), а потім DTP.

Крім об'єднання фізичних каналів в один логічний канал, Etherchannel забезпечує також Load-Balance – «балансування» (розподіл) завантаження, що надходить на передачу логічного каналу, між окремими фізичними каналами з його складу. Режим роботи Load-Balance перевіряється командою **show etherchannel load-balance** (рисунок 3.33).

```
SW1#show etherchannel load-balance
EtherChannel Load-Balancing Operational State (src-mac):
Non-IP: Source MAC address
  IPv4: Source MAC address
  IPv6: Source MAC address
```

Рисунок 3.33 – Перевірка режиму роботи Load-Balance за допомогою команди **show etherchannel load-balance**

В даному прикладі LACP виконує балансування виходячи із значення MAC-адреси джерела, тобто кадри від 1-ого MAC-адреси будуть передані через першу лінію, від 2-ого MAC-адреси через другу лінію, від 3-ого MAC-адреси знову через першу лінію і так буде чергуватися. Але такий підхід не завжди

вірний. Наприклад, є якась умовна мережу (рисунок 3.34), в ній SW1 підключені 2 комп'ютери, далі цей комутатор з'єднується з SW2 агрегованих каналом, а до SW2 поключается маршрутизатор. За замовчуванням Load-Balance налаштований на значення MAC-адреси джерела. В результаті з боку SW1 кадри з MAC-адресою 111 будуть передаватися по першій лінії, а з MAC-адресою 222 по другій лінії. З боку SW2 є тільки підключення одного маршрутизатора з MAC-адресою 333, і всі кадри від маршрутизатора будуть відправлятися на SW1 по першій лінії. Відповідно, друга лінія буде завжди простоювати. Тому логічніше тут налаштувати балансування не по Source MAC-адресу, а по Destination MAC-адресу. Тоді все, що відправляється 1-го комп'ютера, буде відправлятися по першій лінії, а другого - по другій лінії.

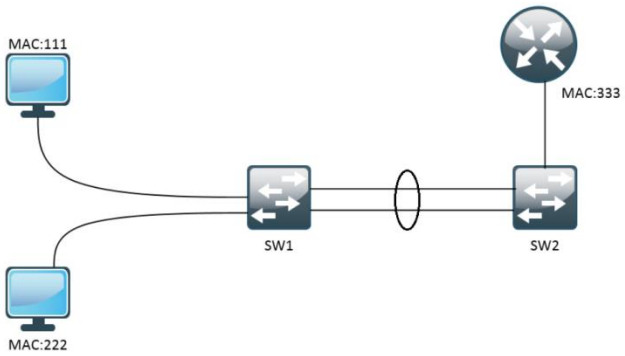


Рисунок 3.34 – Приклад мережі

Зміна режиму балансування виконується командою (рисунок 3.35) – **portchannel load-balance [argument].**

```
SW1(config)#port-channel load-balance ?
dst-ip Dst IP Addr
dst-mac Dst Mac Addr
src-dst-ip Src XOR Dst IP Addr
src-dst-mac Src XOR Dst Mac Addr
src-ip Src IP Addr
src-mac Src Mac Addr
```

Рисунок 3.35 – Зміна режиму балансування за допомогою команди **portchannel load-balance [argument]**

3.2. Використання списків управління доступом.

3.2.1. Варіанти списків управління доступом (ACL).

Список контролю доступу (ACL) – це послідовний список правил дозволу або заборони, застосованих до інформаційних пакетів, відповідно до зазначених в них IP-адрес або ознак протоколів більш високого рівня. ACL-списки дозволяють ефективно контролювати трафік мережі на вході або виході

маршрутизаторів. ACL-списки можна настроювати для усіх протоколів мережі, що маршрутизуються.

ACL-список реалізують послідовністю команд IOS, які визначають виходячи з інформації в заголовку пакету, чи пересилає маршрутизатор пакети чи скидає їх. ACL-списки є однією з найбільш використовуваних функцій операційної системи Cisco IOS.

На рисунку 3.36 наводиться приклад топології з використанням ACL-списків.

Залежно від конфігурації ACL-списки виконують наступні завдання:

– *Обмеження мережного трафіку* для підвищення продуктивності мережі. Наприклад, якщо корпоративна політика забороняє відеотрафік в мережі, необхідно настроїти і застосувати ACL-списки, блокуючі цей тип трафіку. Подібні заходи значно знижують навантаження на мережу і підвищують її продуктивність.

– *Управління потоком трафіку.* ACL-списки можуть обмежувати доставку оновлень маршрутизації. Такі налаштування мережі, дозволяють уникнути зайвого використання смуги пропускання.

– *Забезпечення базового рівня безпеки* відносно доступу до мережі. ACL-списки можуть відкрити доступ до частини мережі одному вузлу і закрити його для інших вузлів. Наприклад, доступ до мережі відділу кадрів може бути обмежений і дозволений тільки авторизованим користувачам.

– *Фільтрування трафіку на основі його типу*. Наприклад, ACL-список може дозволяти трафік електронної пошти, але при цьому блокувати увесь трафік протоколу Telnet.

– Списки контролю доступу здійснюють сортування вузлів в цілях *дозволу або заборони доступу до мережних служб*. За допомогою ACL-списків можна дозволяти або забороняти доступ до певних типів файлів, наприклад FTP або HTTP.

Рисунок 3.36 – Топології з використанням ACL-списків

За замовчанням ACL-списки не конфігуровані на маршрутизаторі, тому маршрутизатор не фільтрує трафік. Трафік, що поступає на маршрутизатор, маршрутизується виключно на основі інформації таблиці маршрутизації. *Проте якщо ACL-список використовується на інтерфейсі, маршрутизатор виконує додаткове завдання, оцінюючи усі мережні пакети, що проходять через інтерфейс, з метою визначення дозволу пересилки пакету.*

Типи ACL-списків Cisco для IPv4

Існує два типи ACL-списків Cisco для IPv4: ***стандартні і розширені*** ACL-списки.

Стандартні ACL-списки можна використати для дозволу або відхилення проходження трафіку тільки на основі IPv4-адрес джерела. *Адреса призначення пакету і порти, що беруть участь в передачі даних,* ***не оцінюються.*** У прикладі на рисунку 3.37 команда створення списку, що дозволяє увесь трафік від мережі 192.168.30.0/24. Стандартні ACL-списки створюються в режимі глобальної конфігурації.

```
access-list 10 permit 192.168.30.0 0.0.0.255
```

Рисунок 3.37 – Приклад стандартного ACL-списку

Розширені ACL-списки фільтрують IPv4-пакети, виходячи з декількох ознак:

– тип протоколу;
– IPv4-адреси джерела;
– IPv4-адреси призначення;
– TCP або UDP порти джерела;
– TCP або UDP порти призначення;
– додаткова інформація про тип протоколу.

На рисунку 3.38 розширений ACL-список 103 дозволяє трафіку з будь-якої адреси мережі 192.168.30.0/24 йти у будь-яку IPv4-мережу, якщо порт призначення – 80 (HTTP). Розширені ACL-списки створюються в режимі глобальної конфігурації.

```
access-list 103 permit tcp 192.168.30.0 0.0.0.255 any eq 80
```

Рисунок 3.38 – Приклад розширеного ACL-списку

Стандартні і розширені списки контролю доступу можна ***створювати з номером або іменем*** для ідентифікації ACL-списку і списку його правил.

Використання нумерованих ACL-списків – ефективний спосіб визначення типу ACL-списку в невеликих мережах, де в основному використовується трафік одного типу. Проте номер не містить інформації про призначення ACL-списку.

З цієї причини, для визначення списків контролю доступу Cisco використовується присвоєне ім'я списку. Іменовані списки створюються у режимі часткового конфігурування.

Існують такі правила привласнення імен і номерів ACL-списків:

a. нумерований список:
- номери від 1 до 99 та від 1300 до 1999 – стандартний ACL-список;
- номери від 100 до 199 та від 2000 до 2699 – розширений ACL-список;

b. іменований список:
- імена можуть містити літерно-цифрові символи;
- рекомендовано використовувати ВЕЛИКИ ЛІТЕРИ;
- в іменах заборонені пробіли та знаки пунктуації;
- записи ACL-списку можна додавати і видаляти.

3.2.2. Порядок налаштування та використання списків управління доступом.

Маршрутизатор працює як фільтр пакетів, коли перенаправляє або відкидає пакети на основі правил фільтрації. Фільтрація пакетів може здійснюватися на різних рівнях (3 та 4 – мережному та транспортному рівнях) моделі взаємодії відкритих систем (OSI) або на рівні міжмережного протоколу TCP/IP.

Для оцінки мережного трафіку, ACL-список перевіряє наступну інформацію із заголовка пакету рівня 3:

– IP-адреса джерела;
– IP-адреса призначення;
– тип повідомлення.

ACL-список також може перевіряти інформацію більш високого рівня із заголовка рівня 4, включаючи:

– порт джерела TCP/UDP;
– порт призначення TCP/UDP.

Списки контролю доступу забезпечують додатковий контроль над пакетами, які приймаються інтерфейсами, транзитними пакетами, які передаються через маршрутизатор, а також пакетами, які вирушають з інтерфейсів маршрутизатора. Списки контролю доступу не застосовуються до пакетів, створених маршрутизатором.

ACL-списки застосовують до трафіку, що входить або виходить:

– **ACL-списки по входу** – пакети, що входять, обробляються перед відправкою на вихідний інтерфейс. ACL-список по входу ефективний, оскільки він зберігає ресурси на пошук маршруту, якщо пакет скидається. Якщо пакет успішно проходить перевірку, він передається на обробку для подальшої маршрутизації. Вхідні ACL-списки є оптимальним рішенням для фільтрації пакетів, коли мережа, яка підключена до вхідного інтерфейсу, є єдиним джерелом пакетів, що вимагають аналізу.

– **ACL-списки по виходу** – пакети, що виходять, маршрутизуються на вихідний інтерфейс, а потім обробляються вихідним списком контролю доступу.

ACL-списки по Вихідні краще всього використовувати, коли однакові фільтри застосовуються до пакетів, що поступають з безлічі інтерфейсів, що входять, перед виходом на один вихідний інтерфейс.

Останній запис будь-якого ACL-списку – це завжди «неявна відмова» (цей рядок напряму не вказують). Це правило автоматично вставляється в кінець кожного ACL-списку, хоча і не є присутнім в ньому фізично. Непряма відмова блокує увесь трафік. Унаслідок цієї неявної заборони ACL-список, що не містить хоч би одного дозволяючого правила, блокує увесь трафік.

Логіка стандартного ACL-списку

На рисунку 3.39 наведений алгоритм обробки пакетів, що поступають на маршрутизатор через інтерфейс G0/0. Перевірки адрес їх джерела здійснюються на основі наступних записів у списку:

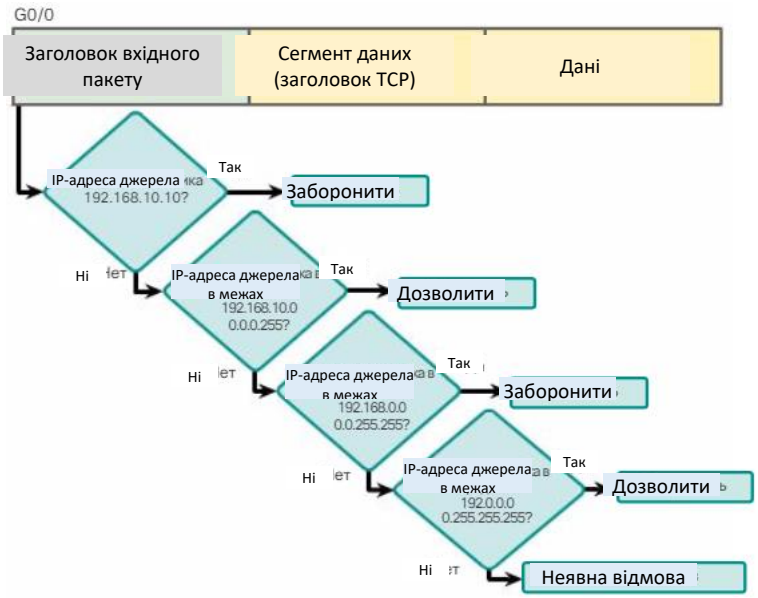


Рисунок 3.39 – Алгоритм обробки пакетів стандартним ACL-списком

access-list 2 deny 192.168.10.10
access-list 2 permit 192.168.10.0 0.0.0.255
access-list 2 deny 192.168.0.0 0.0.255.255
access-list 2 permit 192.0.0.0 0.255.255.255

Якщо пакети дозволені, вони прямують через маршрутизатор до вихідного інтерфейсу. Якщо пакети блокуються, вони відкидаються на вхідному інтерфейсі.

Основні правила застосування ACL-списків на маршрутизаторі:

Для кожного інтерфейсу може існувати декілька правил, необхідних для управління типами трафіку, яким дозволено входити або виходити через певний інтерфейс. Якщо маршрутизатор має два інтерфейси, зконфігурованих і для IPv4 і для IPv6. Якщо для обох протоколів потрібні ACL-списки на обох інтерфейсах і в обох напрямах, то потрібно буде створити 8 окремих ACL-списків. Кожен

інтерфейс матиме чотири ACL-списки: два списки для протоколу IPv4 і два – для протоколу IPv6. Для кожного протоколу потрібний один ACL-список для трафіку, що входить, і один – для вихідного трафіку.

Рекомендації по використанню ACL-списків:

– Доцільно використовувати ACL-списки в міжмережних екранах маршрутизаторів, розміщених між внутрішньою мережею і зовнішньою мережею, наприклад Інтернетом.

– Для управління трафіком, що входить або виходить, в певній частині внутрішньої мережі треба використовувати ACL-списки на маршрутизаторі, розташованому між двома частинами мережі.

– ACL-списки доцільні на пограничних маршрутизаторах, тобто маршрутизаторах, розташованих на межах мереж.

– ACL-списки потрібні для кожного протоколу мережі, налагодженого на інтерфейсі пограничного маршрутизатора.

Правильне розміщення ACL-списку може підвищити ефективність мережі. ACL-список можна розмістити для мінімізації надмірного трафіку (рисунок 3.40):

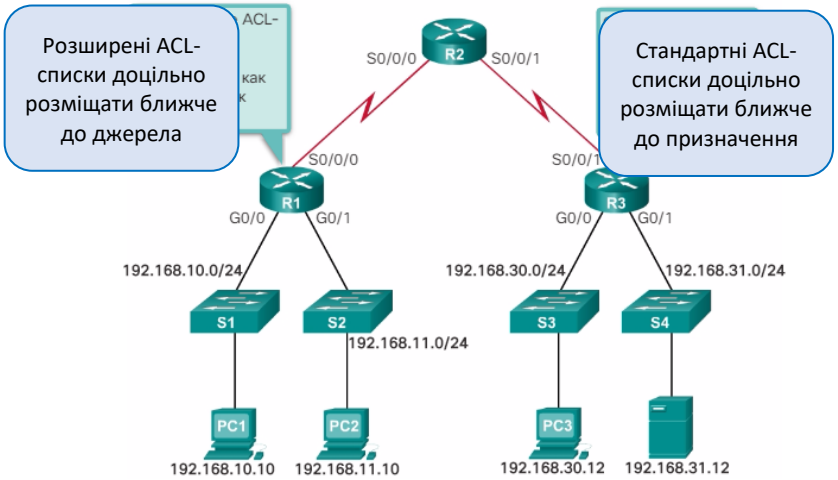


Рисунок 3.40 – Розміщення ACL-списку для мінімізації надмірного трафіку

– **Розширені ACL-списки** слід розміщувати максимально близько до джерела фільтрованого трафіку. Таким чином, небажаний трафік відхиляється близько до мережі-джерела, не перетинаючи інфраструктуру мережі.

– **Стандартні ACL-списки** – розміщують максимально близько до місця призначення. Розміщення стандартного ACL-списку у джерела трафіку дозволяє запобігти досягненню цим трафіком інших мереж через інтерфейс, на якому застосований ACL-список.

Коли трафік поступає на маршрутизатор, він порівнюється із записами в порядку, заданому в ACL-списку. Маршрутизатор продовжує обробку списку, поки не виявить збіг. Маршрутизатор обробляє пакет на основі першого знайденого збігу, інші записи маршрутизатором не враховуються.

Якщо до кінця списку збігу не знайдені, маршрутизатор відхиляє трафік.

Формат команд для реалізації та редагування ACL

Налаштування стандартних ACL-списків

Для використання стандартних нумерованих ACL-списків на маршрутизаторі Cisco необхідно спочатку створити стандартний ACL-список і потім активувати його на інтерфейсі.

Команда глобальної конфігурації **access-list** визначає стандартний ACL-список з номером в діапазоні від 1 до 99. У ОС Cisco IOS версії 12.0.1 цей діапазон розширений; для стандартних ACL-списків можуть використовуватися номери від 1300 до 1999. Це дозволяє створити до 798 можливих стандартних ACL-списків. Додаткові номери посилаються на розширений ACL-список по протоколу IP.

Нижче наводиться повний синтаксис команди стандартного ACL-списку:

Router(config)# **access-list** *access-list-number* {**deny** | **permit** | **remark**}*source* [*source-wildcard*][**log**]

Записи можуть дозволити (**permit**) або заборонити (**deny**) окремий вузол або діапазон адрес вузлів. Для створення в нумерованому ACL-списку 10 запису, що дозволяє певний вузол з IP-адресом 192.168.10.0, необхідно ввести наступну команду:

R1(config)#**access-list 10 permit host 192.168.10.10**

Для створення запису, який дозволить діапазон IPv4-адресів в нумерованому ACL-списку 10, що дозволяє усі IPv4-адреси в мережі 192.168.10.0/24, необхідно ввести наступну команду:

R1(config)# **access-list 10 permit 192.168.10.0 0.0.0.255**

Для видалення ACL-списку використовується команда глобальної конфігурації **no access-list**. Введення команди **show access-list** показує всі наявні ACL-списки.

Для документування і спрощення прочитання ACL-списків використовується ключове слово **remark**. Довжина коментаря обмежена 100 символами.

Ключові слова **host** й **any** в командах створення списків спрощують завдання, допомагаючи визначити найчастіше використовувану шаблонну маску. Ці ключові слова виключають необхідність введення шаблонних масок при визначенні конкретного вузла або цілої мережі. Ці ключові слова полегшують читання ACL-списку, надаючи візуальні підказки відносно критеріїв джерела або призначення.

Ключове слово **host** застосовується для маски 0.0.0.0. Ця маска вказує, що повинні співпадати усі біти IPv4-адреси, або співпадає тільки один вузол.

Ключове слово **any** застосовується для IP-адреси і маски 255.255.255.255. Ця маска вказує ігнорувати увесь IPv4-адрес або прийняти будь-яку адресу.

Наприклад, замість введення **192.168.10.10 0.0.0.0** можна ввести рядок **host 192.168.10.10**. Або замість інструкції **0.0.0.0 255.255.255.255** можна ввести окремо ключове слово **any.**

Із-за неявного правила «**deny any»** у кінці списку цей ACL-список блокує увесь інший трафік.

Створення стандартних іменованих ACL-списків

Привласнення імен ACL-спискам спрощує розуміння їх функцій. Наприклад, ACL-списку, налагодженому для заборони FTP, можна присвоїти ім'я «NO_FTP». При привласненні ACL-списку імені замість номера, режим конфігурації і синтаксис команд трохи міняються.

Крок 1. Для створенням іменованого ACL-списку потрібно виконати команду режиму глобальної конфігурації

Router(config)# **ip access-list** {**standard** | **extended**} *name*

Імена ACL-списків складаються з буквено-цифрових символів, вони чутливі до регістра і мають бути унікальними. Команда *ip access-list standard name* використовується для створення стандартного іменованого ACL-списку, тоді як команда *ip access-list extended name* застосовується для створення розширеного списку доступу. Після введення команди маршрутизатор входить в режим конфігурації іменованого ACL-списку.

Крок 2. У режимі конфігурації іменованих ACL-списків застосовуються команди *permit* або *deny*, щоб задати одне або більше за умови визначення відправки або відхилення пакету, у форматі наведеному нижче.

permit | deny *source* *source-wildcard*

Крок 3. ACL-список потрібно застосувати до інтерфейсу за допомогою команди **ip access-group** з визначенням, чи повинен ACL-список застосовуватися до пакетів, коли вони приходять на інтерфейс (*in*), або коли вони покидають його (*out*). Команда виконується в режимі часткового конфігурування відповідного інтерфейсу

ACL-статистика

Після застосування ACL-списку на інтерфейсі і завершення перевірки за допомогою команди **show access-lists** відображається статистика для кожного співпадаючого запису.

В процесі тестування ACL-списку, лічильники можна скинути, виконавши команду **clear access-list counters**. Цю команду можна застосовувати загалом або з вказівкою номера або імені конкретного ACL-списку.

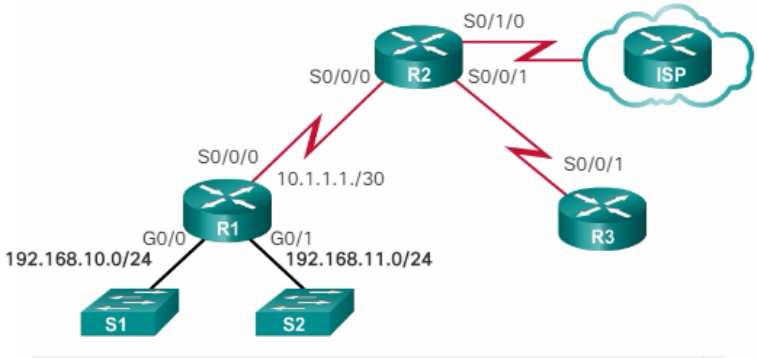
Налаштування розширених ACL-списків

Послідовність кроків налаштування розширених ACL-списків така ж, як для стандартних ACL-списків. Спочатку розширений ACL-список настроюється, а потім активується на інтерфейсі. При цьому слід враховувати, що синтаксис команди і параметри складніші для забезпечення підтримки додаткових функцій, що надаються розширеними ACL-списками, а саме:

permit | **deny** *tcp* ***source*** *source-wildcard* [operator [port]] ***destination*** *destination-wildcard* [operator [port]] [established]

На рисунку 6.6 наведений приклад розширеного списку контролю доступу. В даному прикладі мережний адміністратор настроїв ACL-списки для обмеження мережного доступу будь-якої зовнішньої мережі, щоб дозволити перегляд веб-сайтів тільки з LAN, підключеної до G0/0. ACL 103 дозволяє трафік, який надходить від будь-якої адреси мережі 192.168.10.0, перейти до будь-якого місця призначення з урахуванням обмеження, що трафік використовує тільки порти 80 (HTTP) і 443 (HTTPS).

Характер протоколу HTTP вимагає, щоб трафік повертався назад в мережу від веб-сайтів, до яких зверталися внутрішні клієнти. Мережний адміністратор хоче обмежити трафік, що повертається, до HTTP-обмінів від запитаних веб-сайтів, забороняючи увесь інший трафік. Це завдання виконує ACL 104, блокуючи увесь трафік, що входить, за винятком трафіку від раніше встановлених підключень. Запис *permit* в ACL 104 дозволяє трафік, що входить, параметром *established*. Параметр established дозволяє повернення в мережу 192.168.10.0/24 тільки того трафіку, який запит на який спочатку виходив з цієї мережі. Пакет задовольняє умовам, якщо зворотний сегмент протоколу TCP має біти ACK і RST, які вказують, що пакет належить існуючому підключенню. Без параметра *established* запису ACL-списку клієнт може послати трафік на веб-сервер, але не отримати зворотний трафік, що повертається від веб-сервера.




```
R1(config)# access-list 103 permit tcp 192.168.10.0 0.0.0.255 any eq 80
R1(config)# access-list 103 permit tcp 192.168.10.0 0.0.0.255 any eq 443
R1(config)# access-list 104 permit tcp any 192.168.10.0 0.0.0.255
          established
```

Рисунок 3.41 – Приклад розширеного списку контролю доступу

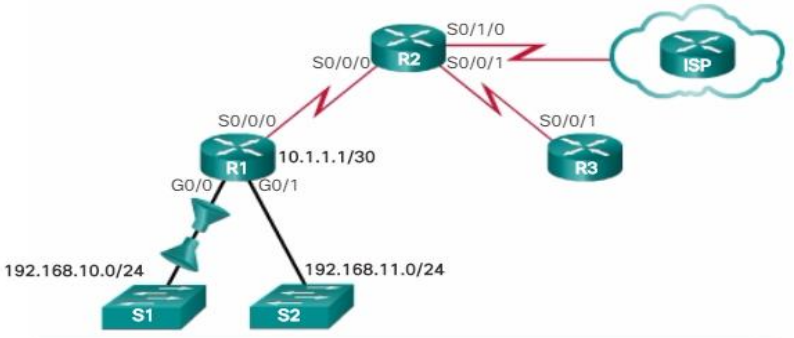
Застосування розширених ACL-списків на інтерфейсах

Навіть налагоджений ACL не фільтруватиме трафік до того, як буде застосований на інтерфейсі. Перед застосуванням ACL на інтерфейсі необхідно визначити, який вид трафіку фільтруватиметься: вхідний або вихідний. Коли користувач внутрішньої LAN дістає доступ до веб-сайту в Інтернеті, трафік йде в Інтернет. Коли внутрішній користувач отримує електронну пошту з Інтернету, трафік йде на локальний маршрутизатор. Проте у випадку із застосуванням списку контролю доступу на інтерфейсі, «in» і «out» набувають інших значень. З точки зору ACL-списку, «in» і «out» є посиланнями на інтерфейс маршрутизатора.

У топології, зображеної на рисунку 3.41, маршрутизатор R1 має три інтерфейси. Це послідовний інтерфейс S0/0/0 і два інтерфейси Gigabit Ethernet: G0/0 і G0/1. Враховуючи що розширений список контролю доступу зазвичай має бути застосований як можна ближче до джерела – у цій топології найближчий до джерела цільового трафіку інтерфейс G0/0.

Трафік веб-запитів від користувачів LAN 192.168.10.0/24 – трафік, що входить, на інтерфейс G0/0. Зворотний трафік від встановлених з’єднань до користувачів локальної мережі вважається вихідним з інтерфейсу G0/0. Приклад застосування ACL-списку на інтерфейсі G0/0 в обох напрямах зображений на рисунку 3.42. Вхідний ACL 103 перевіряє тип трафіку. Вихідний ACL 104 перевіряє трафік, що повертається від встановлених з’єднань. Це обмежує доступ в Інтернет для мережі 192.168.10.0, щоб дозволити тільки перегляд веб-сторінок.

Список доступу можна застосувати на інтерфейсі S0/0/0, але в цьому випадку при обробці списку контролю доступу маршрутизатор переглядатиме усі пакети, що приходять на нього, а не тільки трафік в мережу 192.168.11.0 і з неї. Це може стати причиною зайвого навантаження на маршрутизатор.




```
R1(config)# access-list 103 permit tcp 192.168.10.0 0.0.0.255 any eq 80
R1(config)# access-list 103 permit tcp 192.168.10.0 0.0.0.255 any eq 443
R1(config)# access-list 104 permit tcp any 192.168.10.0 0.0.0.255
established
R1(config)# interface g0/0
R1(config-if)# ip access-group 103 in
R1(config-if)# ip access-group 104 out
```

Рисунок 3.42 – Застосування ACL-списку на інтерфейсі

У прикладі на рисунку 3.43 заборонений трафік FTP з підмережі 192.168.11.0, який спрямовується в підмережу 192.168.10.0, але дозволений будь-який інший тип трафіку. При цьому використовується шаблонна маска і неявна

команди відмови *deny any*. Протокол FTP використовує порти TCP 20 і 21; таким чином, для заборони доступу FTP ACL-списку потрібно обидва ключові слова імені порту: *ftp* і *ftp-data* або *eq 20* і *eq 21*.

Якщо використовується номер порту замість імені порту, команди матимуть наступний вигляд:

access-list 114 permit tcp 192.168.20.0 0.0.0.255 any eq 20
access-list 114 permit tcp 192.168.20.0 0.0.0.255 any eq 21

Щоб виключити блокування усього трафіку командою *deny any*, представленою у вигляді непрямого запису у кінці кожного ACL-списку, необхідно додати команду **permit ip any any**. За відсутності, принаймні, однієї дозволяючої команди permit в ACL-списке увесь трафік на інтерфейсі, де застосований ACL, буде скинутий. Список контролю доступу повинен застосовуватися на інтерфейсі G0/1для того, щоб трафік, що входить з локальної мережі 192.168.11.0/24, фільтрувався при вході на інтерфейс маршрутизатора.

У прикладі на рисунку 3.44 заборонений трафік протоколу Telnet з будь-якого джерела в LAN 192.168.11.0/24, але дозволений увесь інший IP-трафік. Оскільки трафік, призначений для локальної мережі 192.168.11.0/24, є вихідним на інтерфейсі G0/1, ACL-список буде застосований на G0/1 з ключовим словом *out* і з використанням ключових слів *any* в командах дозволу. Запис, що містить дозвіл, додається, щоб уникнути непотрібного блокування трафіку.

У прикладах на Рисунок 3.43 і 3.44 у кінці ACL-списку наводиться команда permit ip any any. Для забезпечення більшої безпеки можна застосувати команду permit 192.168.11.0 0.0.0.255 any.

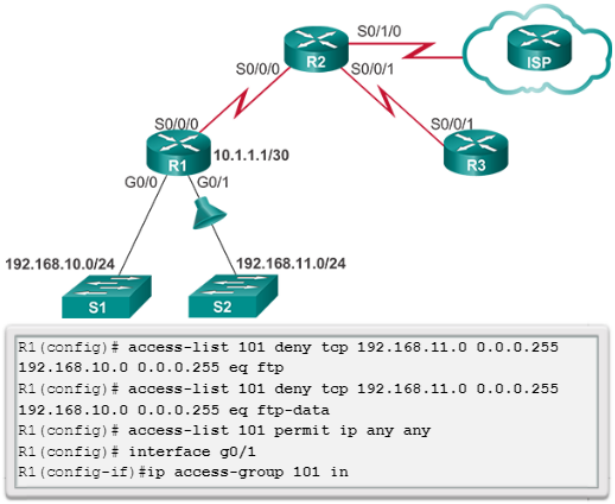


Рисунок 3.43 – ACL-список, заборони FTP

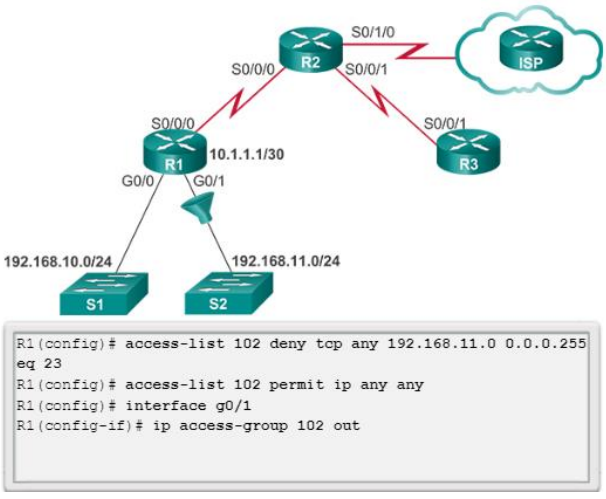


Рисунок 3.44 – ACL-список, заборони Telnet

Для редагування списку контролю доступу можна використовувати один з двох методів:

Метод 1. ***Текстовий редактор*** – при використанні цього методу ACL-список копіюється і вставляється в текстовий редактор, в якому виробляються

зміни. Поточний список доступу видаляється командою **no access-list**. Відредагований ACL-список потім вставляється назад в конфігурацію.

Метод 2. ***Порядкові номери*** – порядкові номери використовуються для видалення або вставки записи списку контролю доступу. Команда **ip access-list** {**standard** | **extended**} *name (number)* використовується для переходу в режим настройки списку контролю доступу. Якщо списку контролю доступу присвоєно номер, а не ім'я, то цей номер потрібно вказати в параметрі. Записи можна вставити або видалити, використовуючи номери рядків. Для видалення використовують команду **no** *номер рядка*. Для додавання використовується стандартна команда для правила, але з необхідним номером на початку (рисунок 3.45).

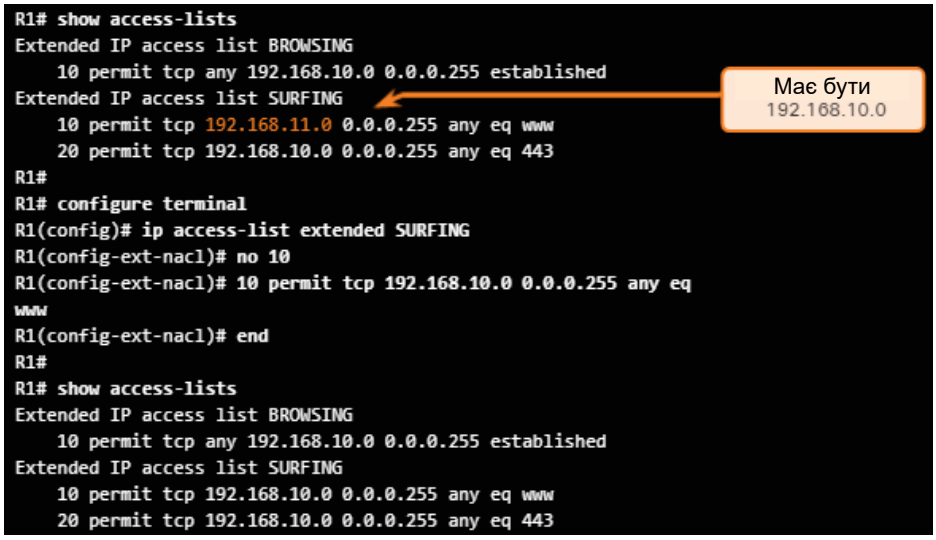


Рисунок 3.45 – Редагування ACL-списку

3.2.3. Протоколи зміни адрес NAT та PAT.

Перенаправлення портів – це перенаправлення трафіку, адресованого певному порту, від одного вузла мережі на інший вузол. Даний метод дозволяє зовнішнім користувачам зовні досягати порту для приватної IPv4-адреси (в локальній мережі), використовуючи маршрутизатор з підтримкою NAT.

Як правило, для роботи пірінгових програм обміну файлами і виконання таких операцій, як робота веб-сервера або вихідний FTP, потрібно, щоб порти маршрутизатора були перенаправлені або відкриті, як показано на рисунку 3.46. Оскільки NAT приховує внутрішні адреси, пірінгові програми працюють тільки зсередини – в цьому випадку NAT може зіставити вихідні запити і вхідні відповіді. Тож проблема полягає в тому, що NAT не дозволяє ініціювати запити зовні. Цю ситуацію можна вирішити за допомогою змін, внесених вручну. Можна налаштувати перенаправлення портів, щоб визначити конкретні порти, які можуть бути переадресовані на внутрішні вузли.

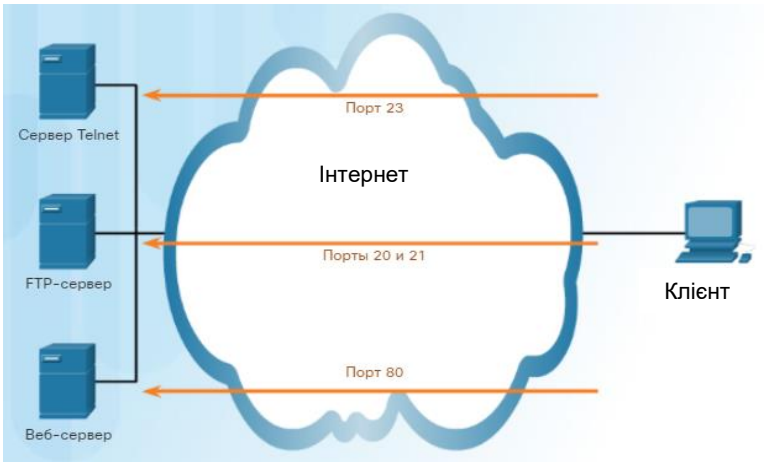


Рисунок 3.46 – Перенаправлення портів

На рисунку 3.47 зображений власник невеликого підприємства, що використовує сервер PoS (пункт продажу) для відстеження продажів і запасів на складі. Сервер доступний всередині складу, і оскільки йому призначена приватна адреса IPv4, публічний доступ до цього сервера з Інтернету неможливий. Включення на локальному маршрутизаторе перенаправлення портів надає власнику доступ до сервера пункту продажів з Інтернету. Перенаправлення портів на маршрутизаторі налаштовується за допомогою номера порту призначення і приватної IPv4-адреси сервера пункту продажів. Для доступу до сервера клієнтська програма повинна використовувати публічну IPv4-адресу маршрутизатора і порту призначення сервера.

Рисунок 3.47 – Використання перенаправлення портів

Якщо використовуваний номер порту відрізняється від стандартного порту призначення, який формується конкретним додатком, то його можна додати до URL-адреси (IPv4-адреси), відокремивши двокрапкою (:). Наприклад, якщо веб-сервер прослуховує порт 8080, користувач повинен ввести **http://www.example.com:8080 (або http: // 209.16.17.2:15555)** .

На рисунку 3.48 показано вікно настройки перенаправлення на один порт для маршрутизатора бездротового зв’язку Linksys. За замовчуванням перенаправлення портів на маршрутизаторі не включене.

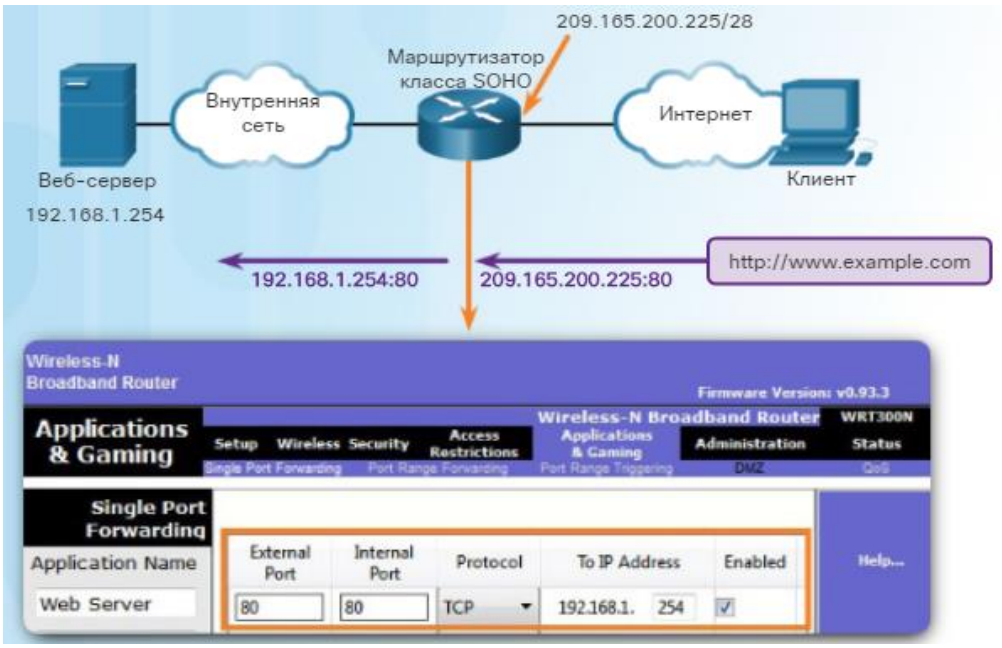


Рисунок 3.48 – Вікно налаштування перенаправлення на один порт для маршрутизатора бездротового зв’язку Linksys

Перенаправлення портів для додатків можна включити, вказавши внутрішню локальну адресу, на яку повинні перенаправлятися запити. На рисунку 3.48 запити до служби HTTP, що надходять на маршрутизатор бездротового зв’язку, перенаправляються на веб-сервер з внутрішньою локальною адресою 192.168.1.254. Якщо зовнішній WAN IPv4-адрес маршрутизатора бездротового зв’язку – 209.165.200.225, то зовнішній користувач може ввести **http://www.example.com**, і цей маршрутизатор перенаправить запит HTTP внутрішньому веб-серверу за IPv4-адресою 192.168.1.254, використовуючи номер порту за замовчуванням – 80.

Можна також вказати номер порту, що відрізняється від номера порту за замовчуванням, відповідного 80. Однак зовнішній користувач повинен знати конкретний використовуваний номер порту. Щоб визначити інший порт, необхідно змінити значення зовнішнього порту (External Port) у вікні перенаправлення на один порт (Single Port Forwarding).

Реалізація перенаправлення портів за допомогою команд IOS аналогічна застосуванню команд настройки статичного NAT. Перенаправлення портів фактично є статичним перетворенням NAT із зазначеним номером порту TCP або UDP.

У таблиці 3.4 розкриті складові команди статичного NAT, що забезпечує настройку перенаправлення портів за допомогою IOS:

Router(config)# ***ip nat inside source static*** {***tcp***|***udp*** *local-ip local-port global-ip global- port*} [**extendable**]

Таблиця 3.4 – Налаштування перенаправлення портів в Cisco IOS

Параметр	Опис
tcp або ***udp***	Вказує тип порту, що буде використаний TCP або UDP
local-ip	IPv4-адреса, призначена вузлу внутрішньої мережі, як правило з діапазону приватних адрес (RFC 1918)
local-port	Локальний порт TCP / UDP в діапазоні від 1 до 65535 до якого сервер очікує підключення
global-ip	Унікальна глобальна IPv4-адреса, яку зовнішні клієнти будуть використовувати для підключення до серверу
global- port	Глобальний порт TCP / UDP в діапазоні від 1 до 65535, який зовнішні клієнти будуть використовувати для підключення до серверу
extendable	Функція, що дозволяє налаштувати певні неоднозначні статичні перетворення, які дозволять за необхідності розширити перетворення до кількох портів

На рисунку 3.49 наведено приклад налаштування перенаправлення портів за допомогою команд IOS на маршрутизаторі R2. 192.168.10.254 – це внутрішня локальна IPv4-адреса веб-сервера, що прослуховує порт 80. Користувачі отримають доступ до цього внутрішнього веб-сервера за допомогою глобальної IPv4-адреси 209.165.200.225, яка є глобальною унікальною загальнодоступною IPv4-адресою. В даному випадку це адреса інтерфейсу Serial 0/1/0 маршрутизатора R2. В якості глобального порту налаштований порт 8080. Він буде портом призначення, що використовується разом з глобальною IPv4-адресою 209.165.200.225 для доступу до внутрішнього веб-сервера. У налаштуванні такого NAT використані наступні параметри команди:

– *local-ip* = 192.168.10.254
– *local-port* = 80
– *global-ip* = 209.165.200.225
– *global-port* = 8080

Якщо не використовується стандартний номер порту, що притаманний додатку (наприклад, для web-браузера це порти 80 або 8080), то клієнт має вказати номер порту в додатку.

Як і для інших типів NAT, для перенаправлення портів необхідно налаштувати як внутрішній, так і зовнішній інтерфейси NAT.

Аналогічно статичному NAT, для перевірки перенаправлення портів можна використовувати команду **show ip nat translations** , як показано на рисунку 3.50.

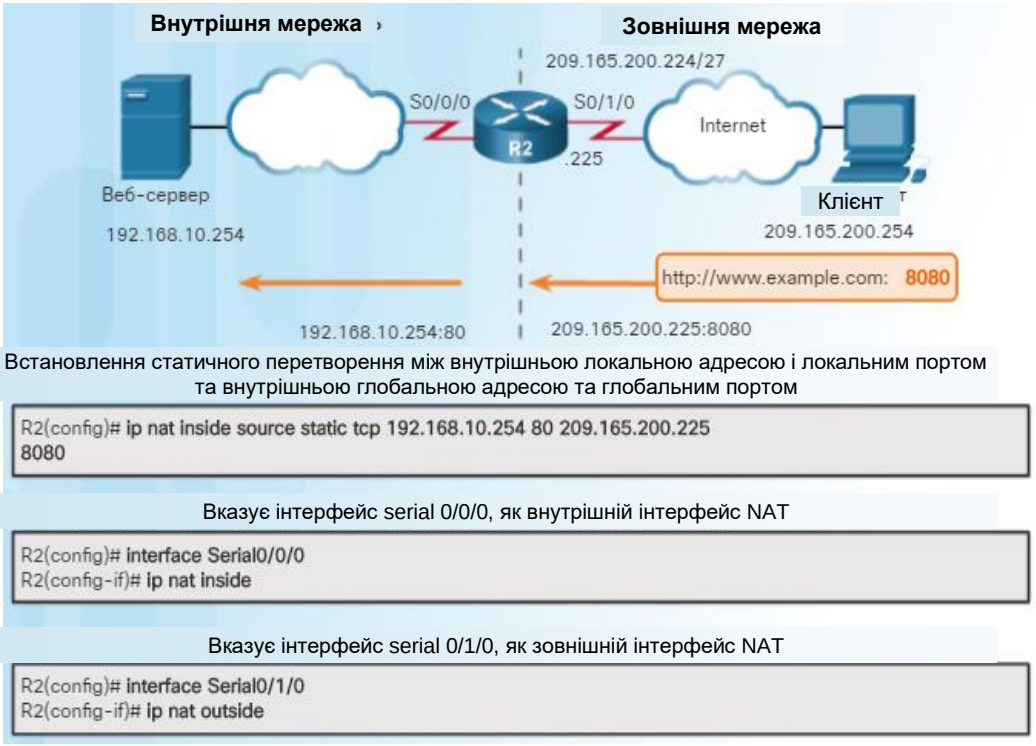


Рисунок 3.49 – Приклад налаштування перенаправлення портів за допомогою команд IOS

У розглянутому прикладі, коли маршрутизатор отримує пакет з внутрішньою глобальною IPv4-адресою 209.165.200.225 і TCP-портом призначення 8080, він виконує пошук в таблиці NAT, використовуючи в якості ключа IPv4-адрес призначення і порт призначення. Потім маршрутизатор перетворює адресу у внутрішню локальну адресу вузла 192.168.10.254 і порт призначення 80. Потім R2 пересилає пакет веб-сервера. Для зворотних пакетів, що йдуть від веб-сервера до клієнта, цей процес інвертується.

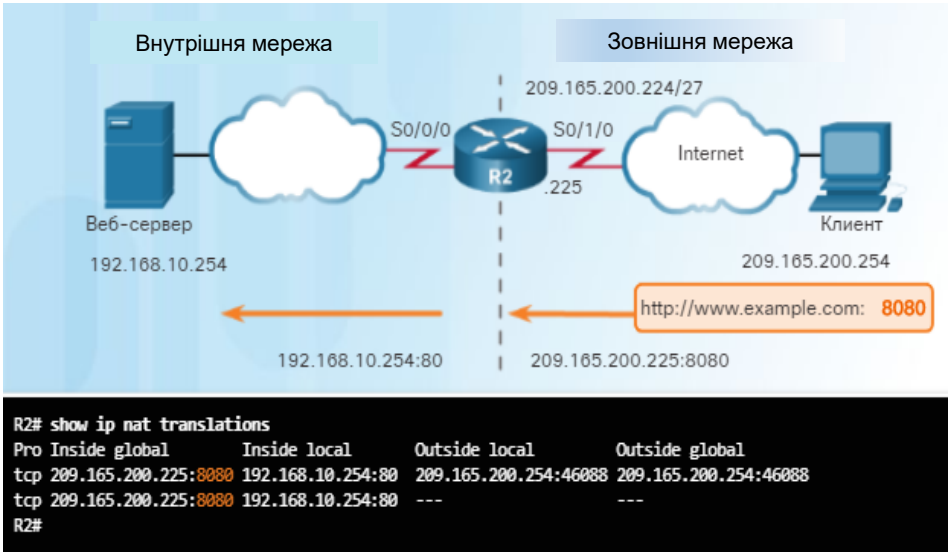


Рисунок 3.50 – Використання команди **show ip nat translations** для перевірки перенаправлення портів

3.3. Моделювання протоколів надійної роботи мереж.

3.3.1. Моделювання та дослідження роботи протоколів HSRP.

HSRP протокол реалізований поверх стека протоколів TCP/IP, для доставки службової інформації використовується протокол UDP. Маршрутизатори або комутатори, що маршрутизують, на яких налаштований і функціонує протокол HSRP, в рамках обміну службовою інформацією використовують так звані пакети вітання (hello packets). У свою чергу, дані пакети відправляються на IP-адресу групового розсилки 224.0.0.2 (HSRP Version 1) або на 224.0.0.102 (HSRP Version 2) протоколом UDP на порт 1985.

HSRP забезпечує високу доступність мережі, забезпечуючи резервування маршрутизації у першому переході для IP-хостів у мережах, налаштованих за допомогою IP-адреси шлюзу за замовчуванням. HSRP використовується у групі маршрутизаторів для вибору активного пристрою та резервного пристрою.

Роль активного та резервного маршрутизаторів визначається під час процесу вибору HSRP. За промовчанням активним маршрутизатором вибрано маршрутизатор з найбільшою за чисельністю адресою IPv4. Однак завжди краще контролювати, як ваша мережа працюватиме в нормальних умовах, ніж залишати це на випадок.

Пріоритет HSRP (HSRP Priority) можна використовувати для визначення активного маршрутизатора. Маршрутизатор із найвищим пріоритетом HSRP стане активним маршрутизатором. За промовчанням пріоритет HSRP дорівнює **100**. Якщо пріоритети рівні, в якості активного маршрутизатора вибирається маршрутизатор із найвищою цифровою адресою IPv4.

Щоб налаштувати маршрутизатор як активний маршрутизатор, використовують команду інтерфейсу **standby priority**. Діапазон пріоритету HSRP становить від 0 до 255.

HSRP заміщення

За промовчанням активізується маршрутизатор, він залишиться активним, навіть якщо інший маршрутизатор підключиться до мережі з більш високим пріоритетом HSRP.

Щоб новий процес вибору HSRP відбувався при підключенні маршрутизатора з більш високим пріоритетом, необхідно увімкнути пріоритетне перемикання за допомогою команди інтерфейсу **standby preempt.** *Заміщення* – це здатність маршрутизатора HSRP ініціювати процес переобрання. При увімкненому витісненні роль активного маршрутизатора виконуватиме маршрутизатор, підключений до мережі з більш високим пріоритетом HSRP.

Перед установка дозволяє маршрутизатору ставати активним маршрутизатором, тільки якщо він має вищий пріоритет. Маршрутизатор, включений для витіснення, з рівним пріоритетом, але з вищою IPv4-адресою, не витіснятиме активний маршрутизатор.

На рисунку 3.51 показана топологія, в якій R1 був налаштований з пріоритетом HSRP 150, в той час як R2 має пріоритет HSRP за замовчуванням 100. Заміщення було включено на R1. R1 є активним маршрутизатором завдяки

вищому пріоритету, а R2 є резервним маршрутизатором. Через збій живлення, що впливає тільки на R1, активний маршрутизатор більше не доступний, а резервний маршрутизатор R2 бере на себе роль активного маршрутизатора. Після відновлення живлення R1 знову вмикається. Оскільки R1 має більш високий пріоритет і пріоритетне заміщення включено, це викликає новий процес виборів і R1 знову прийме роль активного маршрутизатора, а R2 повернеться до ролі резервного маршрутизатора.

Примітка. Якщо пріоритетне заміщення вимкнено, маршрутизатор, який завантажується першим, стане активним маршрутизатором, якщо під час вибору інших маршрутизаторів у мережі не буде.

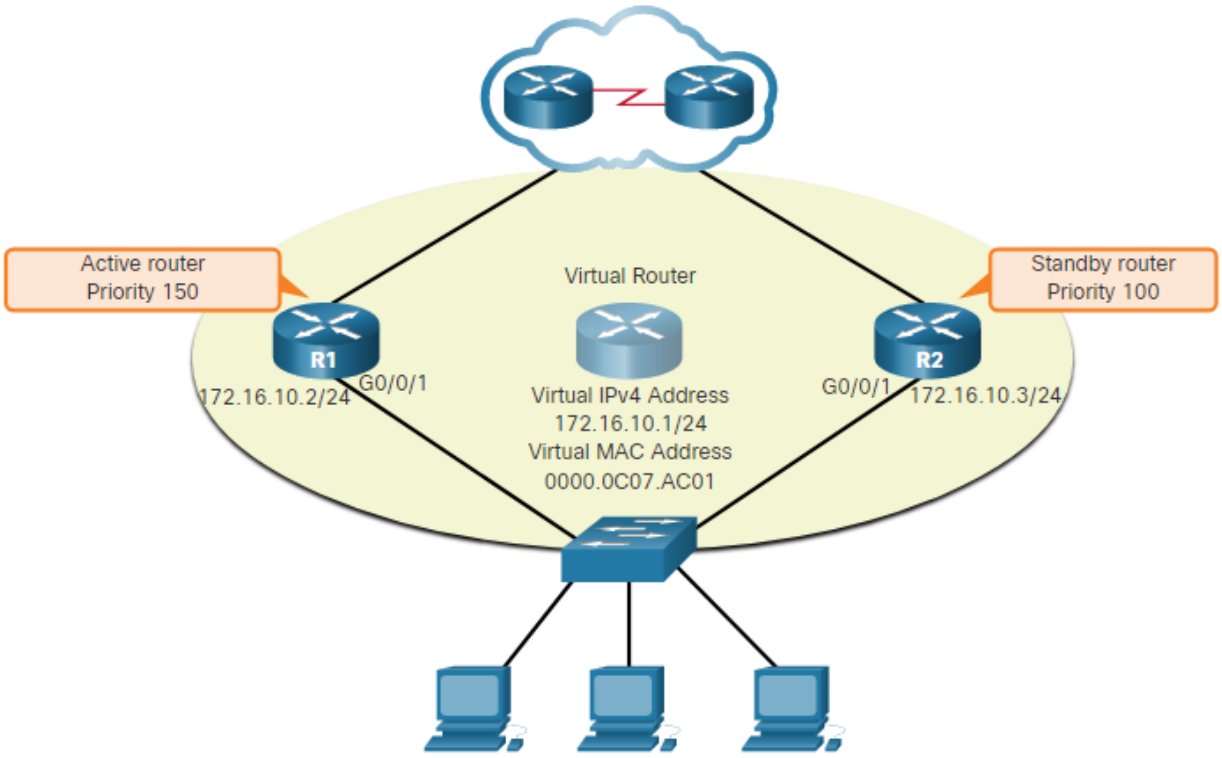


Рисунок 3.51 – Робота протоколу HSRP

Стан та таймери HSRP

Коли інтерфейс маршрутизатора налаштований з HSRP, маршрутизатор відправляє та приймає пакети привітання HSRP, щоб розпочати процес визначення, який стан він прийме в групі HSRP. У таблиці 3.5 наведено перелік та опис станів HSRP.

Активні та резервні маршрутизатори HSRP надсилають вітальні пакети на групову адресу групи HSRP за промовчанням кожні 3 секунди (рисунок 3.52). Резервний маршрутизатор стане активним, якщо за 10 секунд він не отримає вітальне повідомлення від активного маршрутизатора. Можна зменшити налаштування таймера, щоб прискорити аварійне перемикання або переривання. Проте, щоб уникнути збільшення завантаження ЦП та непотрібних змін стану очікування не варто встановлювати таймер вітання менше 1 секунди або таймер утримання менше 4 секунд.

Таблиця 3.5 – Перелік та опис станів HSRP

Назва стану HSRP	**Опис стану HSRP**
Початкове **Initial**	Цей стан вводиться за зміни конфігурації або коли інтерфейс вперше стає доступним.
Вивчення **Learn**	Маршрутизатор не визначив віртуальну IP-адресу і ще не бачив вітального повідомлення від активного маршрутизатора. У цьому стані маршрутизатор очікує на відповідь від активного маршрутизатора.
Прослуховування **Listen**	Маршрутизатор знає віртуальну IP-адресу, але вона не є ні активним, ні резервним маршрутизатором. Він слухає вітання від цих маршрутизаторів.
Оголошення **Speak**	Маршрутизатор періодично надсилає вітальні повідомлення та бере активну участь у виборі активного та/або резервного маршрутизатора.
Режим очікування **Standby**	Маршрутизатор є кандидатом на те, щоб стати наступним активним маршрутизатором і періодично надсилає вітальні повідомлення.

Структура пакету

0		7	15	23
Version	Type	Virtual Rtr ID	Priority	Count IP Addrs
Auth Type		Advet Int	Checksum	
IP Address (1)				
…				
IP Address (n)				
Authentication Data (1)				
Authentication Data (2)				

Рисунок 3.52 – Структура вітального пакету HSRP

Команди налаштування протоколу HSRP:

- standby *номер_групи* IP *IP-адреса* – вказує адресу IP-адресу віртуального маршрутизатора у зазначеній групі;
- standby version *версія_протоколу* – вказує версію протоколу 1 або 2;
- standby *номер_групи* priority *значення_пріоритету* – вказує на пріоритет маршрутизатора у зазначеній групі;
- standby *номер_групи* preempt – дозволяє примусове перехоплення функцій, якщо пріоритет вище, ніж у активного маршрутизатора;

- standby *номер_групи* track *ім'я_інтерфейсу пріоритет* – знижує значення пріоритету маршрутизатора на вказану величину пріоритету, якщо інтерфейс недоступний. Значення відновиться, якщо інтерфейс стане доступним;
- standby *номер_групи* timers *hellotime_сек holdtime_сек* – задає значення таймерів hellotime та holdtime;
- standby *номер групи* authentication *пароль* – пароль для доступу до групи.

На рисунку 3.53 наведена топологія на базі якої розглядається приклад налаштування протоколу HSRP.

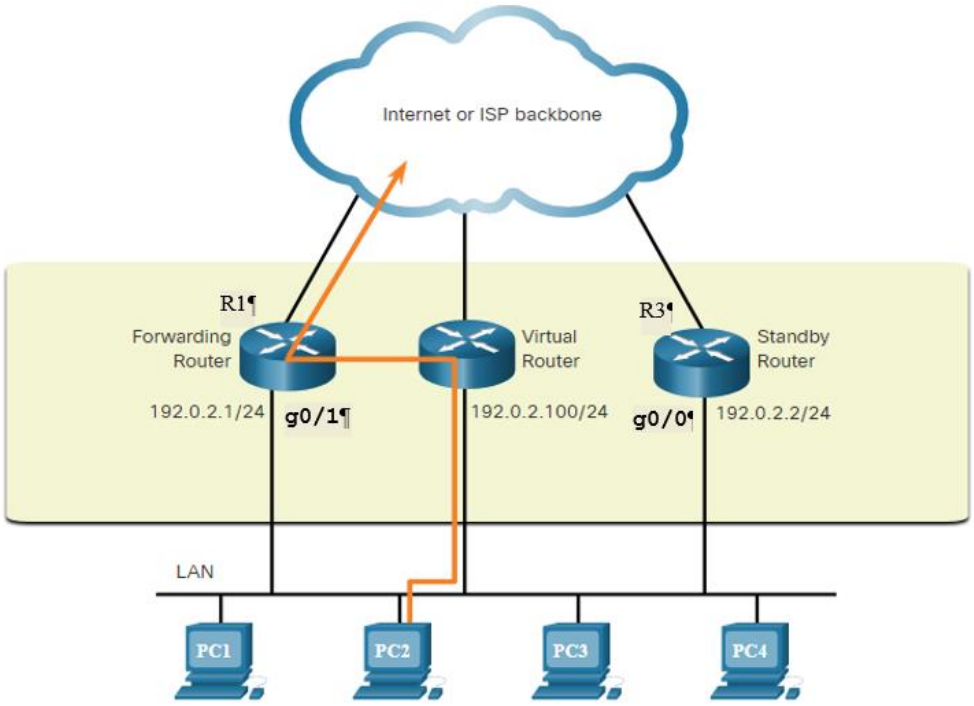


Рисунок 3.53 – Топологія мережі для налаштування протоколу HSRP

R1(config)# **interface g0/1**
R1(config-if)# **standby version 2**
R1(config-if)# **standby 1 ip 192.0.2.100**
R1(config-if)# **standby 1 priority 150**
R1(config-if)# **standby 1 preempt**

R3(config)# **interface g0/0**
R3(config-if)# **standby version 2**
R3(config-if)# **standby 1 ip 192.0.2.100**

R1# **show standby**
GigabitEthernet0/1 - Group 1 (version 2)
State is Active
4 state changes, last state change 00:00:30
Virtual IP address is 192.0.2.100
Active virtual MAC address is 0000.0C9F.F001

Local virtual MAC address is 0000.0C9F.F001 (v2 default)
Hello time 3 sec, hold time 10 sec
Next hello sent in 1.696 secs
Preemption enabled
Active router is local
Standby router is 192.0.2.2
Priority 150 (configured 150)
Group name is «hsrp-Gi0/1-1» (default)

R3# **show standby**
GigabitEthernet0/0 - Group 1 (version 2)
State is Standby
4 state changes, last state change 00:02:29
Virtual IP address is 192.0.2.100
Active virtual MAC address is 0000.0C9F.F001
Local virtual MAC address is 0000.0C9F.F001 (v2 default)
Hello time 3 sec, hold time 10 sec
Next hello sent in 0.720 secs
Preemption disabled
Active router is 192.0.2.1
MAC address is d48c.b5ce.a0c1
Standby router is local
Priority 100 (default 100)
Group name is «hsrp-Gi0/0-1» (default)

R1# **show standby brief**
P indicates configured to preempt.
|
Interface Grp Pri P State Active Standby Virtual IP
Gi0/1 1 150 P Active local 192.0.2.2 192.0.2.100

R3# **show standby brief**
P indicates configured to preempt.
|
Interface Grp Pri P State Active Standby Virtual IP
Gi0/0 1100 Standby 192.0.2.1 local 192.0.2.100

Контрольні питання до розділу 3

1. В середовищі програми Cisco Packet Tracer на підготовленому фрагменті мережі налаштувати маршрутизацію OSPF.
2. В середовищі програми Cisco Packet Tracer на підготовленому фрагменті мережі налаштувати маршрутизацію RIP.
3. В середовищі програми Cisco Packet Tracer на підготовленому фрагменті мережі налаштувати маршрутизацію BGP.
4. В середовищі програми Cisco Packet Tracer на підготовленому фрагменті мережі налаштувати маршрутизацію EIGRP.
5. В середовищі програми Cisco Packet Tracer на підготовленому фрагменті мережі налаштувати протокол збору керуючої інформації Syslog.
6. В середовищі програми Cisco Packet Tracer на підготовленому фрагменті мережі налаштувати протокол контролю мережевого трафіку засобами NetFlow.
7. В середовищі програми Cisco Packet Tracer на підготовленому фрагменті мережі налаштувати протокол забезпечення роботи засобів SPAN.
8. В середовищі програми Cisco Packet Tracer на підготовленому фрагменті мережі зробити попередні налаштування для IP-телефонії (voice VLAN, TRUNKport, interface LOOPBACK, DHCP server).
9. В середовищі програми Cisco Packet Tracer на підготовленому фрагменті мережі налаштувати IP-телефонію в одній зоні.
10. В середовищі програми Cisco Packet Tracer на підготовленому фрагменті мережі налаштувати міжзонову IP-телефонію.
11. В середовищі програми Cisco Packet Tracer на підготовленому фрагменті мережі налаштувати стандартний ACL список.
12. В середовищі програми Cisco Packet Tracer на підготовленому фрагменті мережі налаштувати розширений ACL список дозволу.
13. В середовищі програми Cisco Packet Tracer на підготовленому фрагменті мережі налаштувати розширений ACL список заборони.
14. В середовищі програми Cisco Packet Tracer на підготовленому фрагменті мережі налаштувати статичний протокол NAT.
15. В середовищі програми Cisco Packet Tracer на підготовленому фрагменті мережі налаштувати динамічний протокол NAT.
16. В середовищі програми Cisco Packet Tracer на підготовленому фрагменті мережі налаштувати протокол PAT для кількох зовнішніх локальних адрес.
17. В середовищі програми Cisco Packet Tracer на підготовленому фрагменті мережі налаштувати протокол PAT для однієї зовнішньої локальної адреси.

СПИСОК РЕКОМЕНДОВАНОЇ ЛІТЕРАТУРИ

1. Голь В.Д., Ірха М.С. Телекомунікаційні та інформаційні мережі: навчальний посібник. Київ : ІСЗЗІ КПІ ім. Ігоря Сікорського, 2021. 250 с.
2. В.Д. Голь, А.В. Толстова. Телекомунікаційні та інформаційні мережі / Керівництво з лабораторних занять та курсового проектування. – К.: ІСЗЗІ КПІ ім. Ігоря Сікорського, 2019. – 45 с.
3. А.Г. Микитишин, М.М. Митник, П.Д. Стухляк, В.В. Пасічник Комп'ютерні мережі [навчальний посібник] – Львів, «Магнолія 2006», 2019. – 256 с.
4. Комп'ютерні мережі. Навчальний посібник для виконання лабораторних робіт / Б.Ю. Жураковський, І.О. Зенів – Київ : КПІ ім. Ігоря Сікорського, 2020. –213 с.
5. Голь В.Д., Ірха М.С. Системи передачі даних. Київ: ІСЗЗІ КПІ ім. Ігоря Сікорського, 2021. 126 с.
6. Наталенко П.П. Телекомунікаційні та інформаційні мережі. – К.: ВІТІ, 2011. – 384 с.
7. Програма мережевої академії CISCO CCNA 1 и 2. – Львів, «Магнолія 2006», 2006. – 1168 с.
8. Програма мережевої академії CISCO CCNA 3 и 4. – Львів, «Магнолія 2006», 2008. – 876 с.
9. https://static-course-assets.s3.amazonaws.com/CN503/ru/ index.html#8.0.
10. https://drive.google.com/drive/folders/16n_rarnF47TIqrO12jvex2T3wU4a-uyv?usp=drive_ link.

ПРИМІТКИ