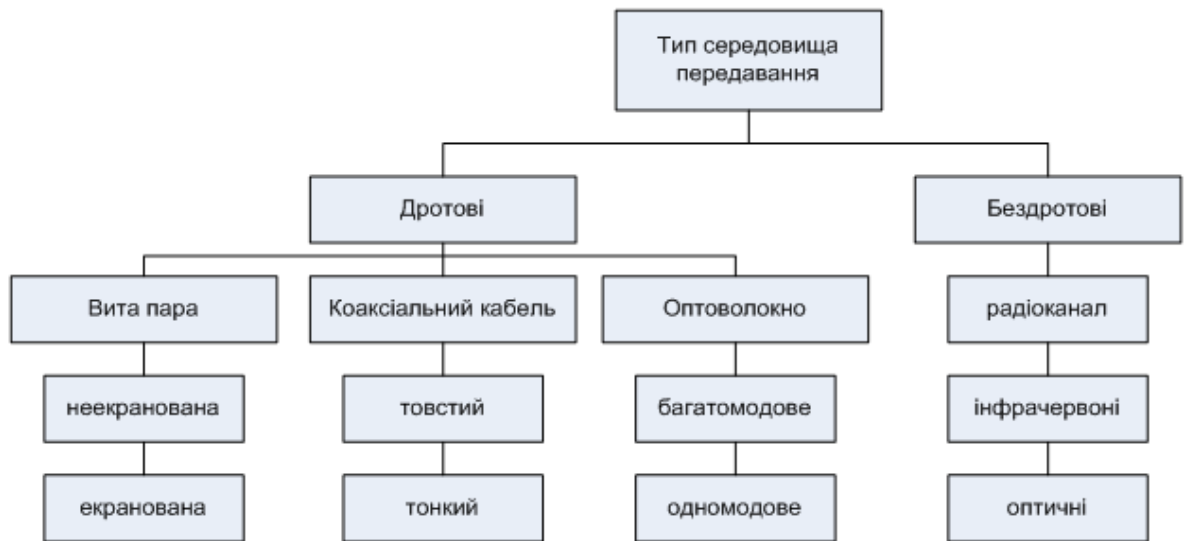


Лекція 2. Основи побудови комп'ютерних мереж (КМ)

1. Апаратні засоби побудови комп'ютерних мереж.
2. Модель взаємодії відкритих систем. Організація взаємодії вузлів мережі.



Схема, що представлена, показує яке середовище передавання використовується у мережах. За середовищем передавання комп'ютерні мережі розподіляються на 2 групи: дротові і бездротові. Розглянемо, які типи кабелю використовуються у дротових мережах.

Перші локальні комп'ютерні мережі в якості середовища передавання використовували коаксіальний кабель: з початку товстий, потім – тонкий. В першому випадку всі комп'ютери мережі підключались до кабелю, в другому – з'єднувались ним. Топологія такої мережі мала назву **загальна шина**. Головним недоліком такої мережі була низька надійність. Проблема з кабелем, в першому випадку, або зі з'єднанням, в другому, випадку, приводила до відмови всієї мережі.



Товстий коаксіальний кабель



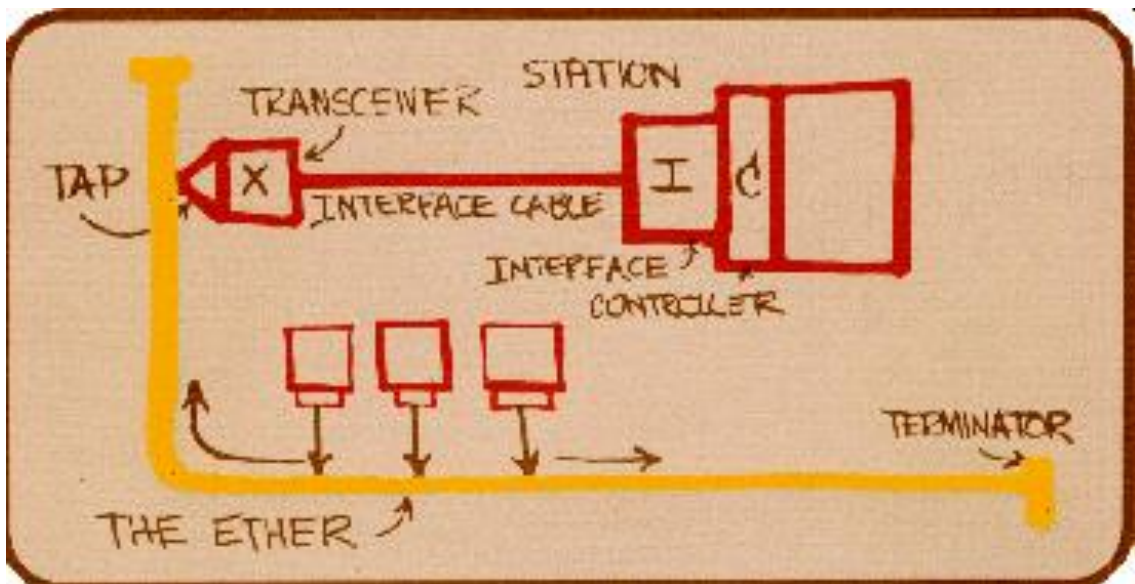
Тонкий коаксіальний кабель RG-58 з роз'ємом BNC



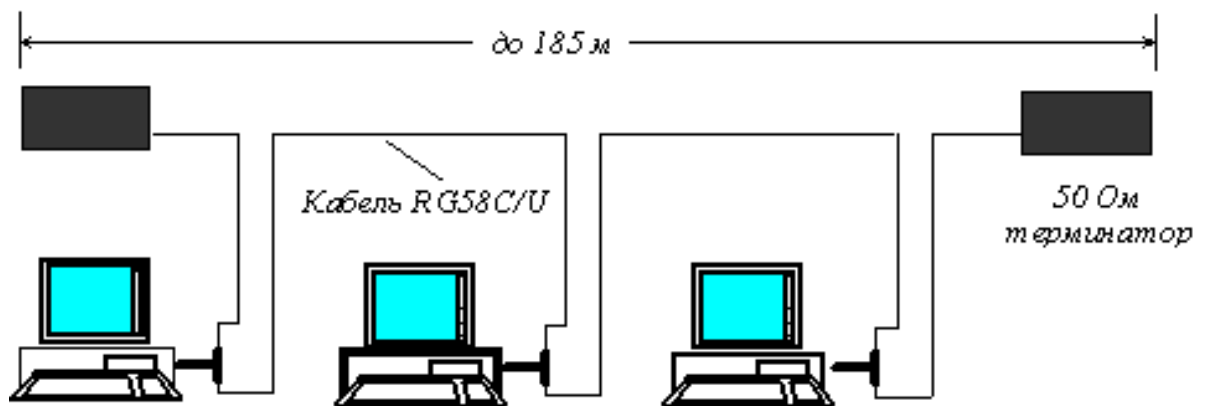
Термінатор

Т-конектор

У 1973 році англійським вченим Робертом Меткалфом були розроблені теоретичні основи першої технології побудови локальних комп'ютерних мереж на основі персональних комп'ютерів – **Ethernet**, але промисловим стандартом технологія Ethernet стала тільки в 1980 році. Максимальна швидкість передачі в останніх версіях мережі Ethernet була 10 Мбіт/с.



Малюнок мережі Ethernet, зроблений Меткалфом



Топологія «загальна шина»

Основним типом кабелю для дротових мереж зараз є неекранована вита пара 5 категорії (рисунок А). Кабель складається з 4 пар, кожна пара має свій колір. Всі дроти мають загальну ізоляцію. Скрутка робиться для зменшення впливу взаємо перешкод. На рисунку В показана вита пара, що має загальний екран, але такий дріт не вважається екранованим. Екранована вита має екран для кожної пари дротів і загальний екран (рисунок С). На рисунку D показана розетка, що використовується для підключення окремого комп'ютеру до мережі. На кожний кінець кабелю встановлюється роз'єм, який на рисунку Е показаний у захищеному варіанті з екраном.



А



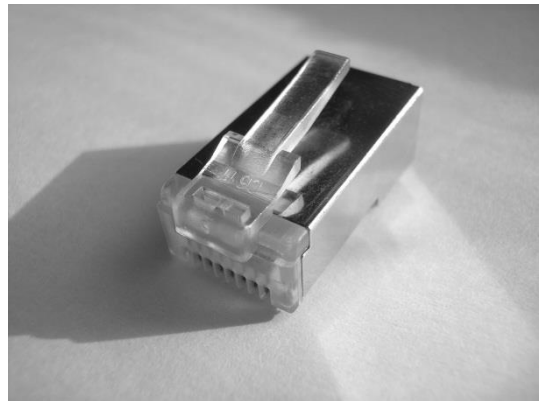
В



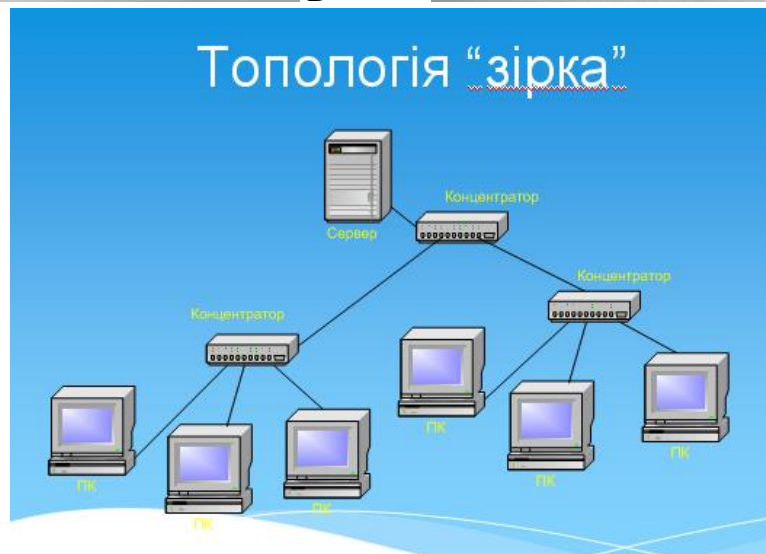
С



Д

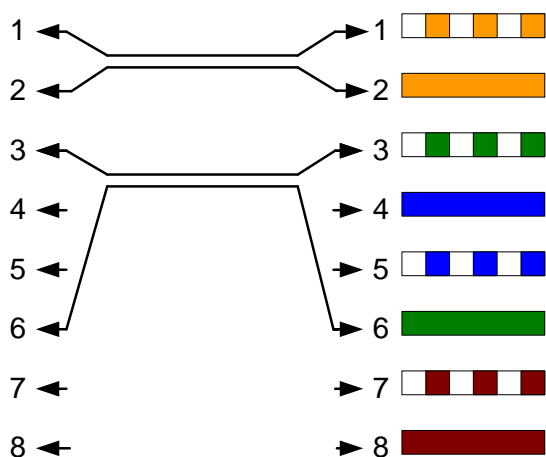
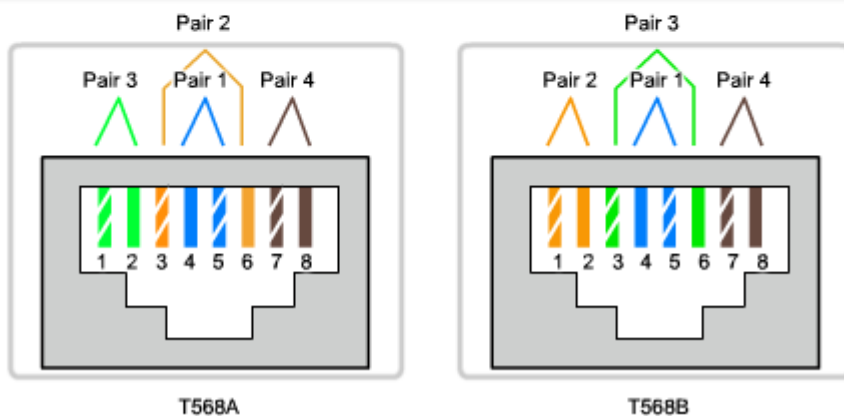


Е

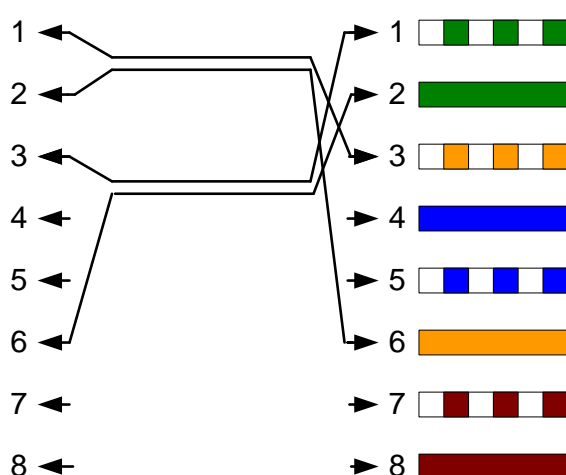


Straight-through, Crossover, and Rollover Cable Types

Cable Type	Standard	Application
Ethernet Straight-through	Both end T568A or both end T568B	Connecting a network host to a network device such as a switch or hub.
Ethernet Crossover	One end T568A, other end T568B	Connecting two network hosts. Connecting two network intermediary devices (switch to switch, or router to router).
Rollover	Cisco proprietary	Connect a workstation serial port to a router console port, using an adapter.

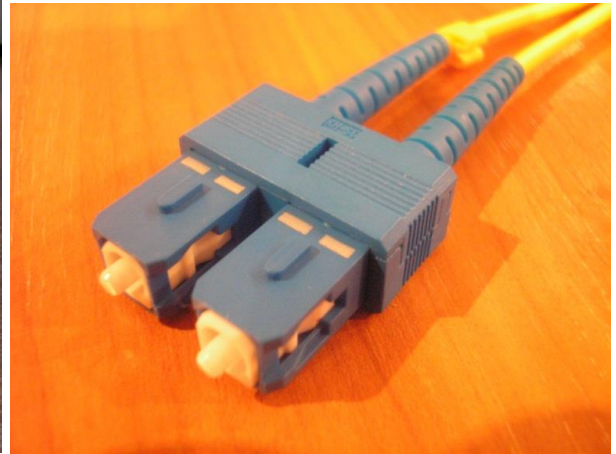


Прямий кабель

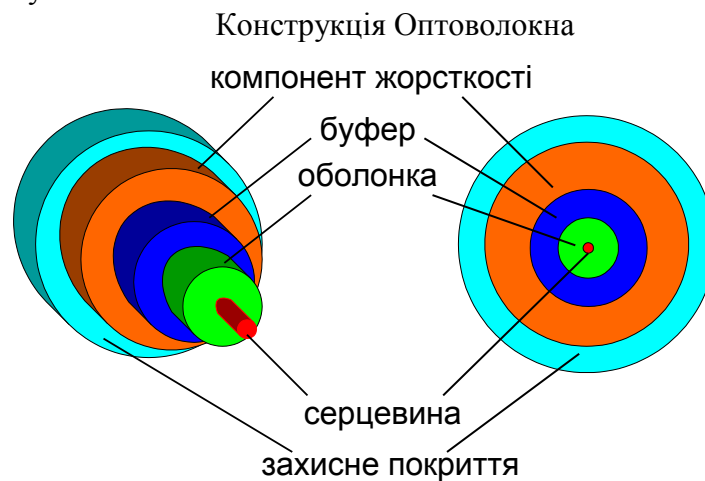


Перехресний кабель

Оптичне волокно



Даний тип кабелю має найвищу захищеність від перешкод і забезпечує передачу сигналу на найвищій швидкості і на велику відстань. Дуже широко використовується останнім часом. Існує одномодове і багатомодове оптичне волокно. Мода – це промінь, за допомогою якого здійснюється передача даних. Найбільш поширеним є використання багатомодового оптичного волокна. Недоліком оптичного волокна є висока вартість і складність монтажу.



Для з'єднання сегментів мережі, що використовують різні типи кабелю використовуються спеціальні пристрої – медіаконвертери. На рисунку показаний конвертер, що забезпечує підключення оптичного волокна і неекранованої виті пари.



Найбільший розвиток в теперішній час мають бездротові комп'ютерні мережі, що для передавання даних використовують радіочастотний діапазон. Для побудови бездротових мереж використовуються 2 пристрої: бездротовий мережевий адаптер і точка доступу (рисунк А). Бездротові адаптери виготовляються у різному виконанні: для встановлення в середину стаціонарного ПК (рисунк В), для встановлення у спеціальний порт портативного комп'ютеру (рисунк С), або у вигляді пристрою, що підключається до порту USB (рисунк D).

Телефонна проводка

Для скорочення витрат на розгортання кабельної проводки локальної комп'ютерної мережі було розроблено обладнання, яке здатне забезпечити передавання даних за допомогою телефонної проводки (телефонного кабелю) і навіть, електричного кабелю. В останній версії технології крім телефонного кабелю може використовуватися коаксіальний кабель, який зараз в будинках використовується для будови мережі кабельного телебачення. Швидкість передавання може бути достатньо високою – 320Мбіт/с. Максимальна кількість пристроїв, що підключаються одночасно – 64. Довжина телефонного кабелю, що використовується для підключення – 300 м, коаксіального – 500 м.

Мережний пристрій, що забезпечує створення мережі з декілька вузлів має назву концентратору. Концентратори мають від 8 до 24 портів. З переходом на концентратори змінилася топологія мережі – тепер замість «загальної шини» стала використовуватися топологія «ієрархічне дерево або зірка». Основна функція концентратора – це **повторення кадру** або на всіх портах, або тільки на деяких портах, відповідно до алгоритму, визначеного відповідним стандартом. Концентратор не в змозі визначити якому вузлу призначене те чи інше повідомлення.

Хаб (он же коммутатор) - устройство для объединения нескольких компов в сеть.

Действие его простое: получив пакет данных с одного порта, он рассылает его по всем остальным. Если несколько пакетов идут в один порт одновременно, то они сталкиваются. Отсюда и т.н. "коллизии", которые замедляют работу сети.

Свитч (он же концентратор) - тот же хаб, только с "мозгами". Он запоминает, какой адрес на каждом его порту, и когда приходит пакет данных (а в заголовке пакета есть информация, кому и от кого), он его отправляет на нужный порт.

Роутер (он же маршрутизатор) - устройство, объединяющее в себе свитч и прокси-сервер. Т.е., это устройство может работать как сервер, объединяя несколько компьютеров и предоставляя им выход в Инет через один внешний адрес. Иногда роутером может служить один из компьютеров сети. Но в случае, если покупка отдельного компа под роутер нецелесообразна, ставится более простой вариант в виде отдельного устройства. Конечно, возможностей у него меньше, но и цена тоже.

На відміну від концентратора комутатор передає кадр даних тільки на той порт, на який потрібно. Тому основне застосування комутаторів полягає в сегментуванні мереж, чим досягається підвищення загальної пропускної здатності мережі. Якщо комп'ютер користувача підключений безпосередньо до порту комутатора, то йому виділяється канал, який комутується (мікросегментація). Комутатор дозволяє зняти обмеження на розмір мережі, що існують при наявності в ній тільки концентраторів. При цьому число сегментів у локальній мережі стає необмеженим. Важлива відмінність комутатора від концентратора полягає в здатності комутаторів поєднувати сегменти мережі, що працюють на різних швидкостях (10/100 Мбіт/с). Це можливо завдяки наявності в комутаторів буферів пам'яті.

Маршрутизатором називається багатопортовий/багатопротокольний пристрій, який призначений для пошуку найкращого шляху для доставки пакетів з однієї мережі в інші мережі. Визначення найкращого шляху може здійснюватися різним чином. Наприклад, оптимальний маршрут може визначатися кількістю маршрутизаторів між відправником і одержувачем пакета, або на основі доступної смуги пропускання і якості сервісу доставки.

2. Організація взаємодії вузлів мережі

Для того, щоб зрозуміти правила обміну інформацією між вузлами мережі (протоколи), згадаємо про правила обміну інформацією між людьми. У будь-якій бесіді між двома людьми є багато правил, або протоколів, яким зобов'язані слідувати обидва співбесідники. До категорії протоколів обміну інформацією між людьми відноситься:

- ідентифікація відправника і одержувача;
- вибраний засіб або канал зв'язку (особиста розмова, телефон, лист, фотографія);
- відповідний режим обміну даними (усна або письмова мова, ілюстрації, інтерактивний або односторонній зв'язок);
- спільна мова;
- граматична структура і структура речення;
- швидкість і час доставки.

Можна тільки уявити собі, якби правил (протоколів) обміну між людьми ні існувало: ми ніколи б не зрозуміли один одного.

При обміні даними комп'ютери як і люди, використовують правила, або протоколи.

Протоколи обміну інформацією між комп'ютерами (вузлами мережі) визначають те, як передається і доставляється повідомлення, а саме:

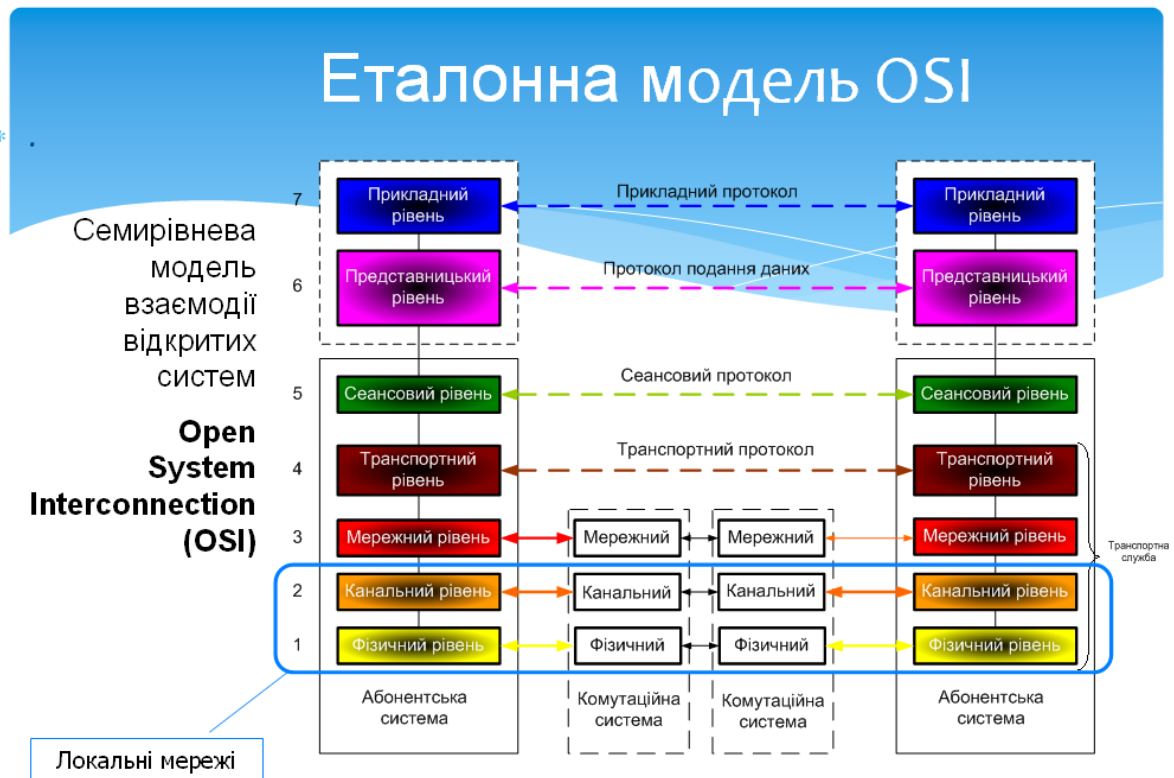
- формат повідомлення;
- розмір повідомлення;
- час доставки;
- спосіб підготовки;
- кодування;
- схему стандартного повідомлення.

Для кожного повідомлення використовується особливий формат, який називається кадром. Кадр діє приблизно так само, як і конверт листа: у ньому вказані адреси вузла-відправника і одержувача.

Формат і вміст кадру залежать від типу повідомлення і каналу передачі. Вузол-адресат не може успішно одержати і обробити повідомлення, що невірно відформатоване.

!!!!

Міжнародними стандартами визначена модель взаємодії відкритих систем, яка, в свою чергу, визначає 7 рівнів організації взаємодії між вузлами мережі. З точки зору даної моделі **протокол** – це формалізовані правила, за якими здійснюється обмін даними між мережевими компонентами вузлів мережі, що знаходяться на одному рівні моделі **OSI**. Існують 7 рівнів взаємодії, на кожному рівні може застосовуватися не один, а навіть декілька протоколів обміну. Особливе значення протоколи мають у локальних мережах. Всі комп'ютери локальної мережі повинні використовувати той самий протокол. Зараз найчастіше у локальних мережах застосовується протокол, який має назву Ethernet. В моделі OSI локальні мережі охоплюють два перших рівня: фізичний і канальний.



Основным, с точки зрения пользователя, является **прикладной уровень** - набор разнообразных протоколов, с помощью которых пользователи сети получают доступ к разделяемым ресурсам: файлам, принтерам, web-страницам и др. Также организует совместную работу протоколов.

Следующий (шестой) уровень называется **представительным** (уровень представления данных). Он определяет единый для всех систем синтаксис передаваемой информации. Необходимость данного уровня обусловлена различной формой представления информации в сети передачи данных и компьютерах. Этот уровень играет важную роль в обеспечении «открытости» систем, позволяя им общаться между собой независимо от их внутреннего языка. На данном уровне может выполняться шифрование и дешифрование данных, благодаря которому секретность обмена данными обеспечивается сразу для всех прикладных служб. Примером такого протокола для TCP/IP (протокол Интернет) является протокол Secure Socket Layer (SSL).

Пятый уровень называют **сеансовым**, так как основным его назначением является организация сеансов связи между прикладными процессами различных рабочих станций. На этом уровне создаются порты для приема и передачи сообщений и организуются соединения — логические каналы между процессами. Необходимость протоколов этого

уровня определяется относительной сложностью сети передачи данных и стремлением обеспечить достаточно высокую надежность передачи информации. На практике немногие приложения используют сеансовый уровень, и он редко реализуется в виде отдельных протоколов. Функции этого уровня часто объединяют с функциями прикладного уровня и реализуют в одном протоколе.

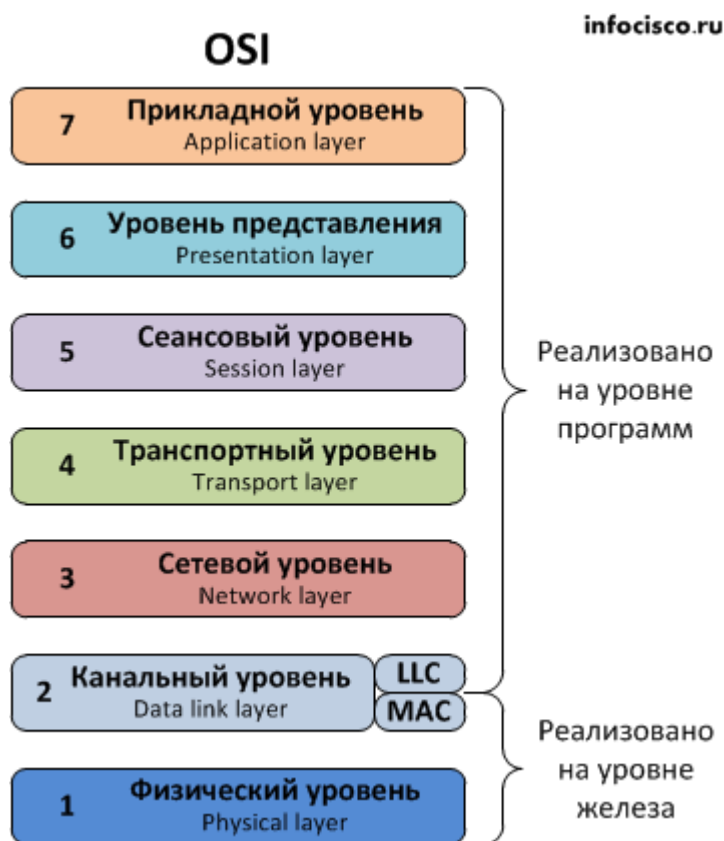
Четвертый, **транспортный** уровень (уровень сквозной передачи) служит для передачи данных между двумя взаимодействующими открытыми системами. На пути от отправителя к получателю пакеты могут быть искажены или утеряны. Транспортный уровень обеспечивает приложениям или прикладному и сеансовому уровням передачу данных с той степенью надежности, которая им требуется.

На этом уровне определяется взаимодействие рабочих станций — источника и адресата данных, организуется и поддерживается логический канал (транспортное соединение) между абонентами.

Третий, **сетевой** уровень, предназначен для маршрутизации информации и управления сетью передачи данных. В отличие от предыдущих, этот уровень в большей степени ориентирован на сеть передачи данных. Здесь решаются вопросы управления сетью передачи данных, в том числе маршрутизация и управление информационными потоками.

Канальный уровень обеспечивает функциональные и процедурные средства для установления, поддержания и расторжения соединений на уровне каналов передачи данных. Процедуры канального уровня обеспечивают обнаружение и, возможно, исправление ошибок, возникающих на физическом уровне.

Физический уровень обеспечивает механические, электрические, функциональные и процедурные средства организации физических соединений при передаче бит данных между физическими объектами.



Как говорилось выше, сетевая модель – это модель взаимодействия сетевых протоколов (стандартов), вот на каждом уровне и присутствуют свои протоколы. Перечислять их скучный процесс (да и не к чему), поэтому лучше разберем все на примере, ведь усваиваемость материала на примерах гораздо выше ;)

Прикладной уровень

Прикладной уровень или уровень приложений (application layer) – это самый верхний уровень модели. Он осуществляет связь пользовательских приложений с сетью. Эти приложения нам всем знакомы: просмотр веб-страниц (HTTP), передача и приём почты (SMTP, POP3), приём и получение файлов (FTP, TFTP), удаленный доступ (Telnet) и т.д.

Представительский уровень

Представительский уровень или уровень представления данных (presentation layer) – он преобразует данные в соответствующий формат. На примере понять проще: те картинки (все изображения) которые вы видите на экране, передаются при пересылке файла в виде маленьких порций единиц и ноликов (битов). Так вот, когда Вы отправляете своему другу фотографию по электронной почте, протокол Прикладного уровня SMTP отправляет фотографию на нижний уровень, т.е. на уровень Представления. Где Ваша фотка преобразуется в удобный вид данных для более низких уровней, например в биты (единицы и нолики).

Именно таким же образом, когда Ваш друг начнет получать Ваше фото, ему оно будет поступать в виде все тех же единиц и нулей, и именно уровень Представления преобразует биты в полноценное фото, например JPEG.

Вот так и работает этот уровень с протоколами (стандартами) изображений (JPEG, GIF, PNG, TIFF), кодировок (ASCII, EBDIC), музыки и видео (MPEG) и т.д.

Сеансовый уровень

Сеансовый уровень или уровень сессий(session layer) – как видно из названия, он организует сеанс связи между компьютерами. Хорошим примером будут служить аудио и видеоконференции, на этом уровне устанавливается, каким кодеком будет кодироваться сигнал, причем этот кодек должен присутствовать на обеих машинах. Еще примером может служить протокол SMPP (Short message peer-to-peer protocol), с помощью него отправляются хорошо известные нам СМСки и USSD запросы. И последний пример: PAP (Password Authentication Protocol) – это старенький протокол для отправки имени пользователя и пароля на сервер без шифрования.

Транспортный уровень

Транспортный уровень (transport layer) – этот уровень обеспечивает надёжность передачи данных от отправителя к получателю. На самом деле всё очень просто, например вы общаетесь с помощью веб-камеры со своим другом или преподавателем. Нужна ли здесь надёжная доставка каждого бита переданного изображения? Конечно нет, если потеряется несколько битов из потокового видео Вы даже этого не заметите, даже картинка не изменится (м.б. изменится цвет одного пикселя из 900000 пикселей, который промелькнет со скоростью 24 кадра в секунду).

А теперь приведем такой пример: Вам друг пересылает (например, через почту) в архиве важную информацию или программу. Вы скачиваете себе на компьютер этот архив. Вот здесь надёжность нужна 100%, т.к. если пару бит при зачке архива потеряются – Вы не сможете затем его разархивировать, т.е. извлечь необходимые данные. Или представьте себе отправку пароля на сервер, и в пути один бит потерялся – пароль уже потеряет свой вид и значение изменится.

Таким образом, когда мы смотрим видеоролики в интернете, иногда мы видим некоторые артефакты, задержки, шумы и т.п. А когда мы читаем текст с веб-страницы – потеря (или сжатие) букв не допустима, и когда скачиваем программы – тоже все проходит без ошибок.

На этом уровне я выделю два протокола: UDP и TCP. UDP протокол (User Datagram Protocol) передает данные без установления соединения, не подтверждает доставку данных и не делает повторы. TCP протокол (Transmission Control Protocol), который перед передачей устанавливает соединение, подтверждает доставку данных, при необходимости делает повтор, гарантирует целостность и правильную последовательность загружаемых данных.

Следовательно, для музыки, видео, видеоконференций и звонков используем UDP (передаем данные без проверки и без задержек), а для текста, программ, паролей, архивов и т.п. – TCP (передача данных с подтверждением о получении, затрачивается больше времени).

Сетевой уровень

Сетевой уровень (network layer) – этот уровень определяет путь, по которому данные будут переданы. И, между прочим, это третий уровень Сетевой модели OSI, а ведь существуют такие устройства, которые как раз и называют устройствами третьего уровня – маршрутизаторы.

Все мы слышали об IP-адресе, вот это и осуществляет протокол IP (Internet Protocol). IP-адрес – это логический адрес в сети.

На этом уровне достаточно много протоколов и все эти протоколы мы разберем более подробно позже, в отдельных статьях и на примерах. Сейчас же только перечислю несколько популярных.

Как об IP-адресе все слышали и о команде ping – это работает протокол ICMP.

Те самые маршрутизаторы (с которыми мы и будем работать в дальнейшем) используют протоколы этого уровня для маршрутизации пакетов (RIP, EIGRP, OSPF).

Канальный уровень

Канальный уровень (data link layer) – он нам нужен для взаимодействия сетей на физическом уровне. Наверное, все слышали о MAC-адресе, вот он является физическим адресом. Устройства канального уровня – коммутаторы, концентраторы и т.п.

IEEE (Institute of Electrical and Electronics Engineers - Институт инженеров по электротехнике и электронике) определяет канальный уровень двумя подуровнями: LLC и MAC.

LLC – управление логическим каналом (Logical Link Control), создан для взаимодействия с верхним уровнем.

MAC – управление доступом к передающей среде (Media Access Control), создан для взаимодействия с нижним уровнем.

Объясню на примере: в Вашем компьютере (ноутбуке, коммуникаторе) имеется сетевая карта (или какой-то другой адаптер), так вот для взаимодействия с ней (с картой) существует драйвер. Драйвер – это некоторая **программа** - верхний подуровень канального уровня, через которую как раз и можно связаться с нижними уровнями, а точнее с микропроцессором (**железо**) – нижний подуровень канального уровня.

Типичных представителей на этом уровне много. PPP (Point-to-Point) – это протокол для связи двух компьютеров напрямую. FDDI (Fiber Distributed Data Interface) – стандарт передаёт данные на расстояние до 200 километров. CDP (Cisco Discovery Protocol) – это проприетарный (собственный) протокол принадлежащий компании Cisco Systems, с помощью него можно обнаружить соседние устройства и получить информацию об этих устройствах.

Физический уровень

Физический уровень (physical layer) – самый нижний уровень, непосредственно осуществляющий передачу потока данных. Протоколы нам всем хорошо известны: Bluetooth, IRDA (Инфракрасная связь), медные провода (витая пара, телефонная линия), Wi-Fi, и т.д.

Роботу мережі Інтернет забезпечує група ієрархічних, взаємопов'язаних протоколів, яку інакше називають **стеком** протоколів TCP/IP. Термін “стек” в даному випадку показує, що протоколи мають певну ієрархію і взаємодіють між собою, причому протокол нижнього рівня надає певні сервіси протоколу, що займає наступний, верхній рівень. Нагадаю, що протокол – це алгоритм, правила, за якими виконується взаємодія між вузлами мережі, на певному рівні моделі OSI. На рис. показано, як рівні моделі TCP/IP відповідають рівням еталонної моделі взаємодії відкритих систем.

На відміну від еталонної моделі OSI, модель TCP/IP має не 7 рівнів, а тільки 4. Верхні 3 рівня моделі OSI відповідають 1 рівню модель TCP/IP – рівню додатків. Транспортний рівень моделі TCP/IP і моделі OSI відповідають один одному, так само, як і наступний 3-й рівень, який в моделі OSI має назву мережевого, а у моделі TCP/IP – Інтернет. Останні, самі нижчі 2 рівня моделі OSI відповідають 1 рівню моделі TCP. Розглянемо функції кожного рівня моделі TCP/IP, а також протоколи, що використовуються на даному рівні та їх взаємодію з протоколами нижніх рівнів.

Клієнтське програмне забезпечення – веб-браузер и програмне забезпечення серверу - веб-сервер під час обміну інформацією використовують один з протоколів рівня додатків стеку TCP/IP, а саме – протокол HTTP (протокол передачі гіпертекстової інформації). На це вказує інформація, що введена у адресний рядок браузера – <http://vkontakte.ru>. Цей рядок має назву універсального покажчика ресурсу (URL) і, як можна бачити, містить назву саме протоколу HTTP. Зверніть увагу на те, що назва файлу у URL відсутня. В даному випадку сервер пересилає стандартну кореневу сторінку, що зазначена у його настройках.

Щоб отримати потрібну веб-сторінку клієнт посилає серверу за протоколом HTTP певну інформацію, що складається із заголовку і поля даних. Заголовок має інструкцію (команду) GET – отримати, а також містить назву потрібного файлу – `index.php`. Відповідь від сервера також здійснюється у вигляді HTTP-інформації, в заголовку якої завжди вказується спеціальний код. В нашому випадку, наприклад, цей код означає Ок, тобто сторінка, що запитується у сервера існує і може бути надана клієнту. Цей приклад описує важливу концепцію взаємодії вузлів мережі: коли певний рівень одного вузла взаємодіє з рівнозначним рівнем іншого вузла, обидва використовують протокол, заголовки якого містять інформацію про їх взаємодію. Такий процес має назву взаємодії рівнозначних рівнів.

Під час передачі інформація може бути втраченою або спотвореною внаслідок впливу перешкод, що виникають у каналах зв'язку. В цьому випадку у протоколу HTTP не має засобів, щоб усунути помилки або знову запитати передачу втрачених даних. Ці функції можна було додати до даного протоколу, але на рівні додатків діють і інші протоколи, тоді функції гарантованої передачі потрібно було б додати також і до них, що привело б до дублювання і ускладнення алгоритмів їх роботи. Тому функції гарантованої передачі інформації між вузлами відокремлені від протоколів рівня додатків і забезпечуються нижчим рівнем – транспортним і відповідним протоколом – TCP (Transmission Control Protocol). Таким чином, здійснюється взаємодія між протоколами не на одному рівні, а між суміжними протоколами. Тобто протокол нижчого рівня (в даному випадку - транспортний) надає певні послуги (забезпечує гарантовану передачу інформації) протоколам верхнього рівня, наприклад, HTTP. Розглянемо яким чином це відбувається.

Переваги протоколу TCP, його користь не будуть помітні поки дані, що передаються між вузлами мережі не будуть втрачені. В даному випадку протокол HTTP не буде ні чого роботи тому, цю задачу виконає протокол TCP. Таким чином, цей процес демонструє

взаємодію суміжних рівнів мережевої моделі, коли два сусідніх рівня стеку протоколів одного комп'ютеру взаємодіють один з одним для передачі інформації у мережі.

Инкапсуляция (лат. in capsula, в коробочке) — абстракция и сокрытие данных и действий от окружающей среды

Протокол TCP має механізм гарантованої доставки пакетів, але безпосередньо доставку пакетів він не виконує. Цю задачу вирішує протокол Інтернету, який надає такі послуги протоколу транспортного рівня. Робота протоколу Інтернету схожа на роботу поштового відділення. Тому для доставки повідомлень кожному вузлу (хосту) мережі присвоюється унікальний признак – адреса. На даному рівні моделі використовується протокол IP (Internet Protocol), тому адреса вузла має назву IP-адреси.



IP – адреса має дві частини:

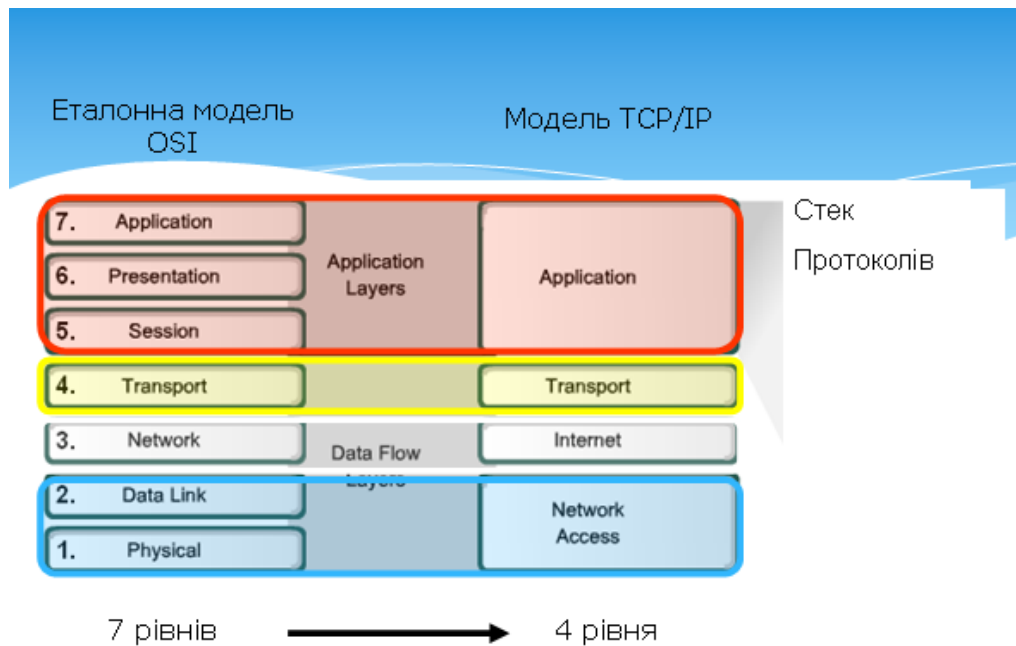
- ідентифікатор локальної мережі (мережева частина IP-адреси, IP-адреса мережі);
- ідентифікатор вузла у даній мережі.

Мережева частина IP-адреси визначається мережевою маскою або мережевим префіксом.

Приклад

IP-адреса: 193.162.230.115
маска: 255.255.255.0

IP-адреса мережі:
193.162.230.0



Роботу мережі Інтернет забезпечує група ієрархічних, взаємопов'язаних протоколів, яку інакше називають **стеком** протоколів TCP/IP. Термін “стек” в даному випадку показує, що протоколи мають певну ієрархію і взаємодіють між собою, причому протокол нижнього рівня надає певні сервіси протоколу, що займає наступний, верхній рівень. Нагадаю, що протокол – це алгоритм, правила, за якими виконується взаємодія між вузлами мережі, на певному рівні моделі OSI. На рис. показано, як рівні моделі TCP/IP відповідають рівням еталонної моделі взаємодії відкритих систем.

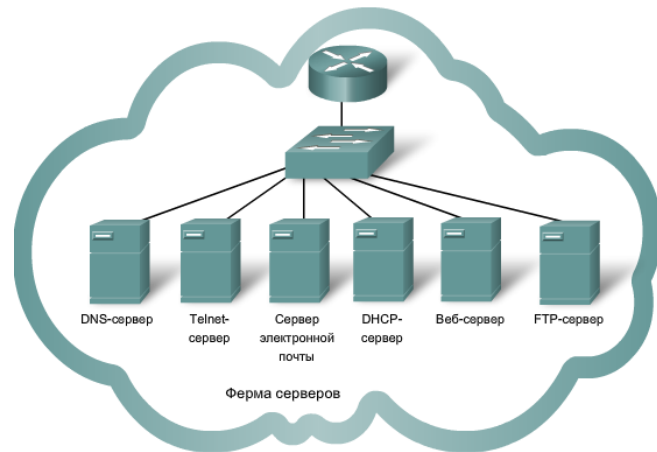
На відміну від еталонної моделі OSI, модель TCP/IP має не 7 рівнів, а тільки 4. Верхні 3 рівня моделі OSI відповідають 1 рівню моделі TCP/IP – рівню додатків. Транспортний рівень моделі TCP/IP і моделі OSI відповідають один одному, так само, як і наступний 3-й рівень, який в моделі OSI має назву мережевого, а у моделі TCP/IP – Інтернет. Останні, самі нижчі 2 рівня моделі OSI відповідають 1 рівню моделі TCP. Розглянемо функції кожного рівня моделі TCP/IP, а також протоколи, що використовуються на даному рівні та їх взаємодію з протоколами нижніх рівнів.

Application layer

В моделі TCP/IP рівень додатків надає певні служби програмному забезпеченню, що працює на комп'ютері.

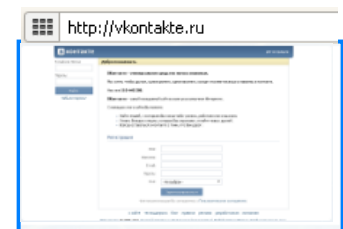
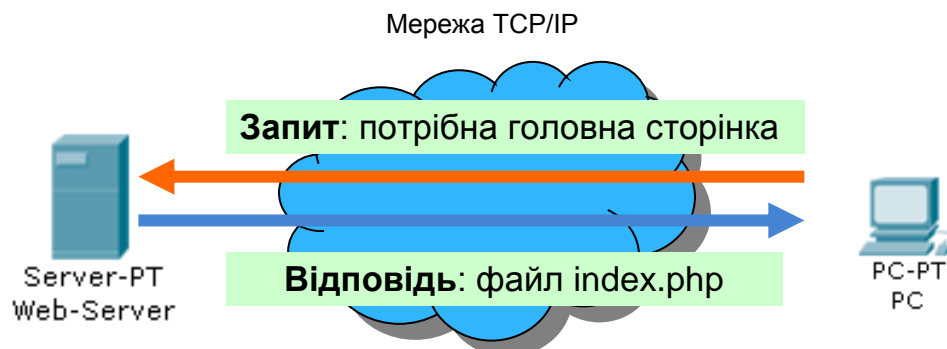
Даний рівень виконує функцію інтерфейсу між додатками, що використовуються користувачем і мережею.

Рівень не визначає вимоги до додатків користувача, проте стандартизує служби, якими можуть користуватися додатки.



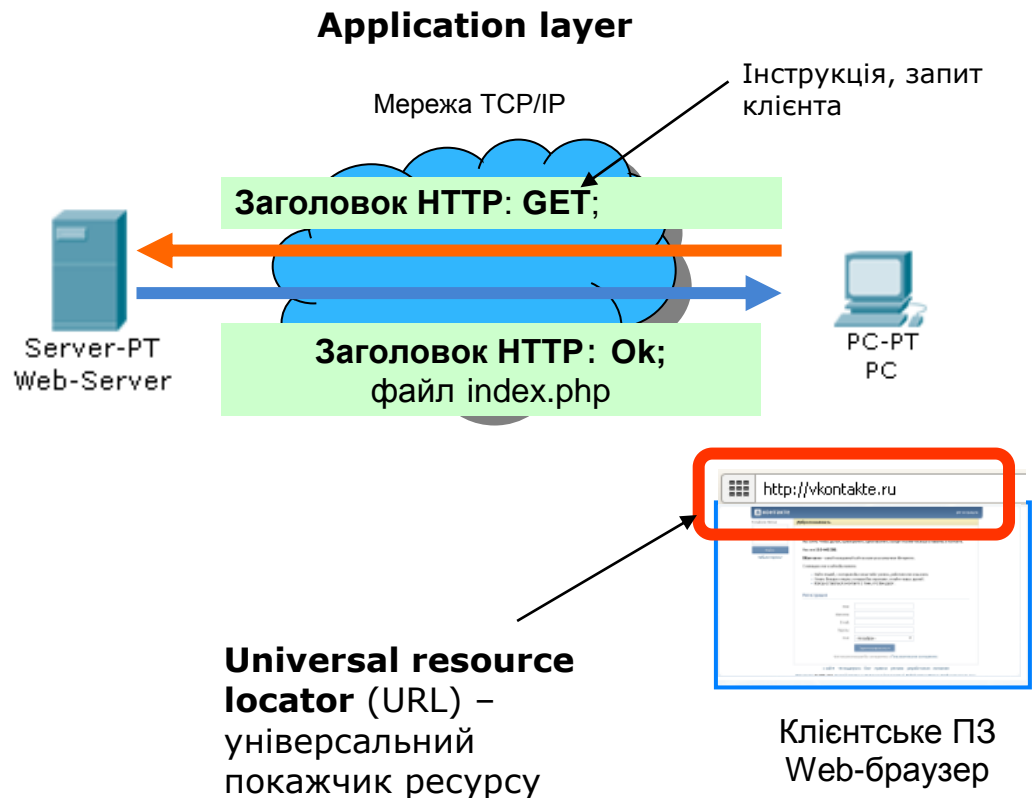
Служби рівня додатків

Взаємодія вузлів мережі



Клієнтське ПЗ
Web-браузер

Взаємодія рівнозначних рівнів

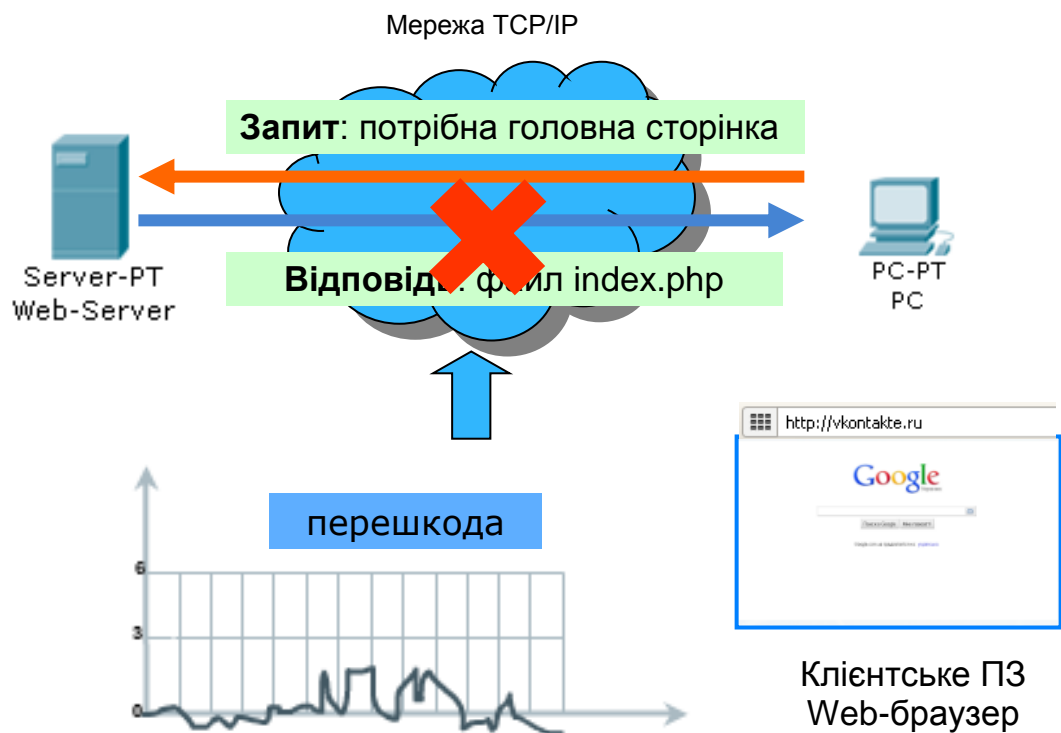


Клієнтське програмне забезпечення – веб-браузер и програмне забезпечення серверу – веб-сервер під час обміну інформацією використовують один з протоколів рівня додатків стеку TCP/IP, а саме – протокол HTTP (протокол передачі гіпертекстової інформації). На це вказує інформація, що введена у адресний рядок браузера – <http://vkontakte.ru>. Цей рядок має назву універсального показника ресурсу (URL) і, як можна бачити, містить назву саме протоколу HTTP. Зверніть увагу на те, що назва файлу у URL відсутня. В даному випадку сервер пересилає стандартну кореневу сторінку, що зазначена у його настройках.

Щоб отримати потрібну веб-сторінку клієнт посилає серверу за протоколом HTTP певну інформацію, що складається із заголовку і поля даних. Заголовок має інструкцію (команду) GET – отримати, а також містить назву потрібного файлу – `index.php`.

Відповідь від сервера також здійснюється у вигляді HTTP-інформації, в заголовку якої завжди вказується спеціальний код. В нашому випадку, наприклад, цей код означає Ok, тобто сторінка, що запитується у сервера існує і може бути надана клієнту.

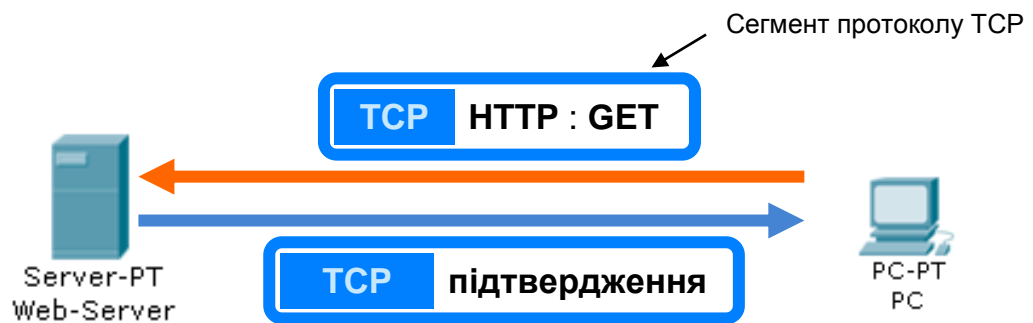
Цей приклад описує важливу концепцію взаємодії вузлів мережі: коли певний рівень одного вузла взаємодіє з рівнозначним рівнем іншого вузла, обидва використовують протокол, заголовки якого містять інформацію про їх взаємодію. Такий процес має назву взаємодії рівнозначних рівнів.



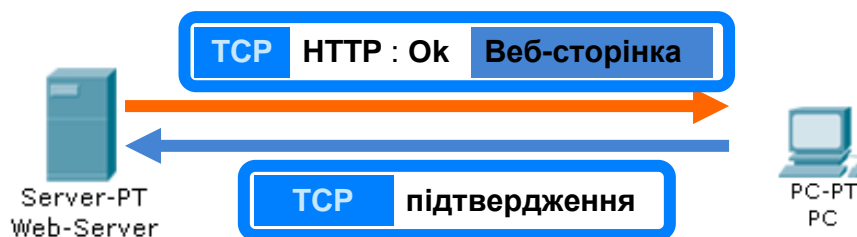
Під час передачі інформація може бути втраченою або спотвореною внаслідок впливу перешкод, що виникають у каналах зв'язку. В цьому випадку у протоколу HTTP не має засобів, щоб усунути помилки або знову запитати передачу втрачених даних. Ці функції можна було додати до даного протоколу, але на рівні додатків діють і інші протоколи, тоді функції гарантованої передачі потрібно було б додати також і до них, що привело б до дублювання і ускладнення алгоритмів їх роботи. Тому функції гарантованої передачі інформації між вузлами відокремлені від протоколів рівня додатків і забезпечуються нижчим рівнем – транспортним і відповідним протоколом – TCP (Transmission Control Protocol). Таким чином, здійснюється взаємодія між протоколами не на одному рівні, а між суміжними протоколами. Тобто протокол нижчого рівня (в даному випадку - транспортний) надає певні послуги (забезпечує гарантовану передачу інформації) протоколам верхнього рівня, наприклад, HTTP. Розглянемо яким чином це відбувається.

Взаємодія суміжних рівнів

Транспортний рівень



Протокол TCP для виявлення помилок використовує механізм, який базується на підтвердженні, що прийнята інформація не має помилок



TCP - сегмент

Інформація протоколу HTTP



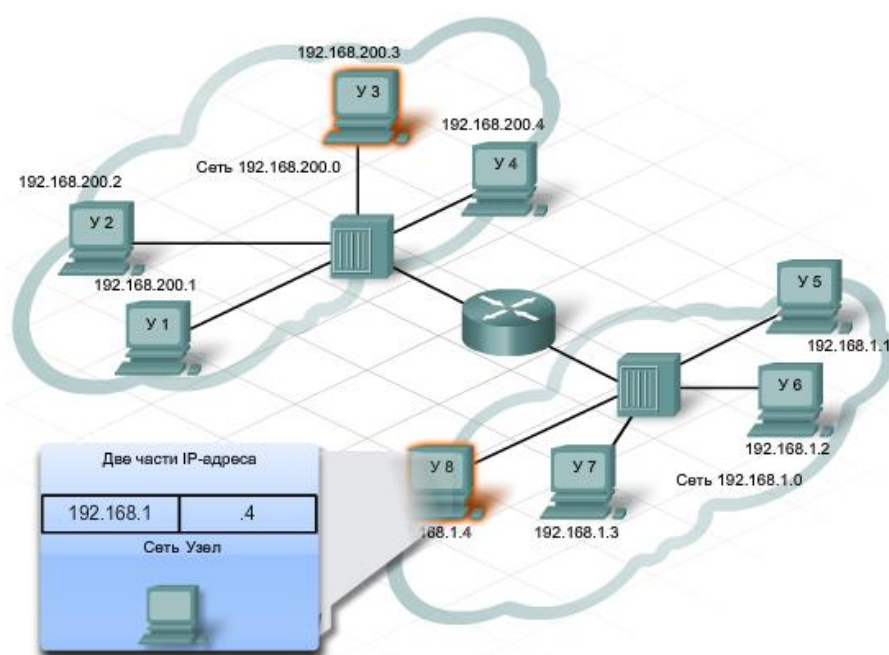
заголовок Інкапсульовані дані протоколу HTTP

Переваги протоколу TCP, його користь не будуть помітні поки дані, що передаються між вузлами мережі не будуть втрачені. В даному випадку протокол HTTP не буде ні чого робити тому, цю задачу виконає протокол TCP. Таким чином, цей процес демонструє взаємодію суміжних рівнів мережевої моделі, коли два сусідніх рівня стеку протоколів одного комп'ютеру взаємодіють один з одним для передачі інформації у мережі.

Инкапсуляція (лат. *in capsula*, в коробочке) — абстракція и сокрытие данных и действий от окружающей среды

Рівень Інтернету

Протокол IP: Internet Protocol



IP-адреса схожа на адресу місця проживання людини. Вона називається логічною адресою, оскільки присвоюється логічно, в залежності від місцезнаходження вузла. IP-адресу, або мережеву адресу, присвоює вузлу адміністратор.

Вузли, що знаходяться у одній мережі мають однакову мережеву частину

Протокол TCP має механізм гарантованої доставки пакетів, але безпосередньо доставку пакетів він не виконує. Цю задачу вирішує протокол Інтернету, який надає такі послуги протоколу транспортного рівня. Робота протоколу Інтернету схожа на роботу поштового відділення. Тому для доставки повідомлень кожному вузлу (хосту) мережі присвоюється унікальний признак – адреса. На даному рівні моделі використовується протокол IP (Internet Protocol), тому адреса вузла має назву IP-адреси.

IP-адреса (адреса протоколу IP)

IP-адреса: **193.162.230.115**
w.x.y.z

0..255 0..255 0..255 0..255

IP – адреса має дві частини:

- ідентифікатор локальної мережі (мережева частина IP-адреси, IP-адреса мережі);
- ідентифікатор вузла у даній мережі.

Мережева частина IP-адреси визначається мережевою маскою або мережевим префіксом.

Приклад

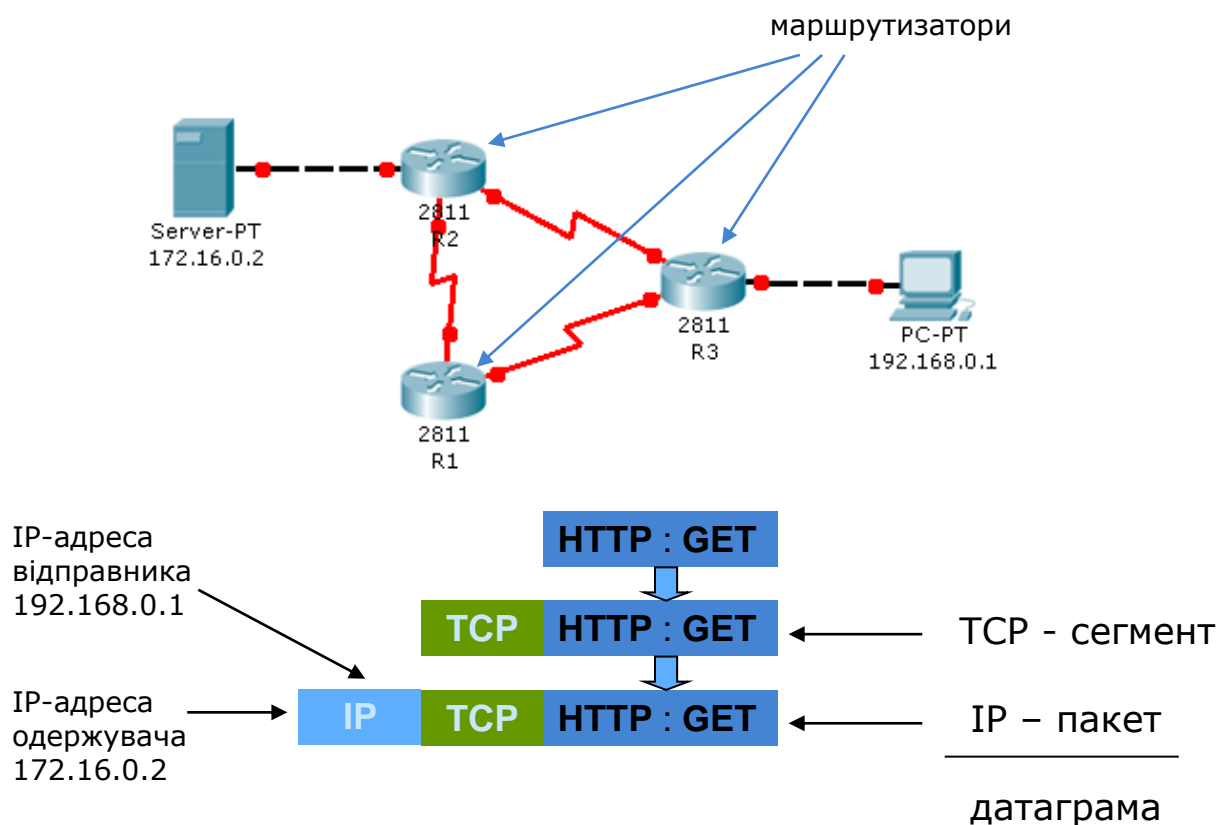
IP-адреса: 193.162.230.115

маска: 255.255.255.0

IP-адреса мережі:

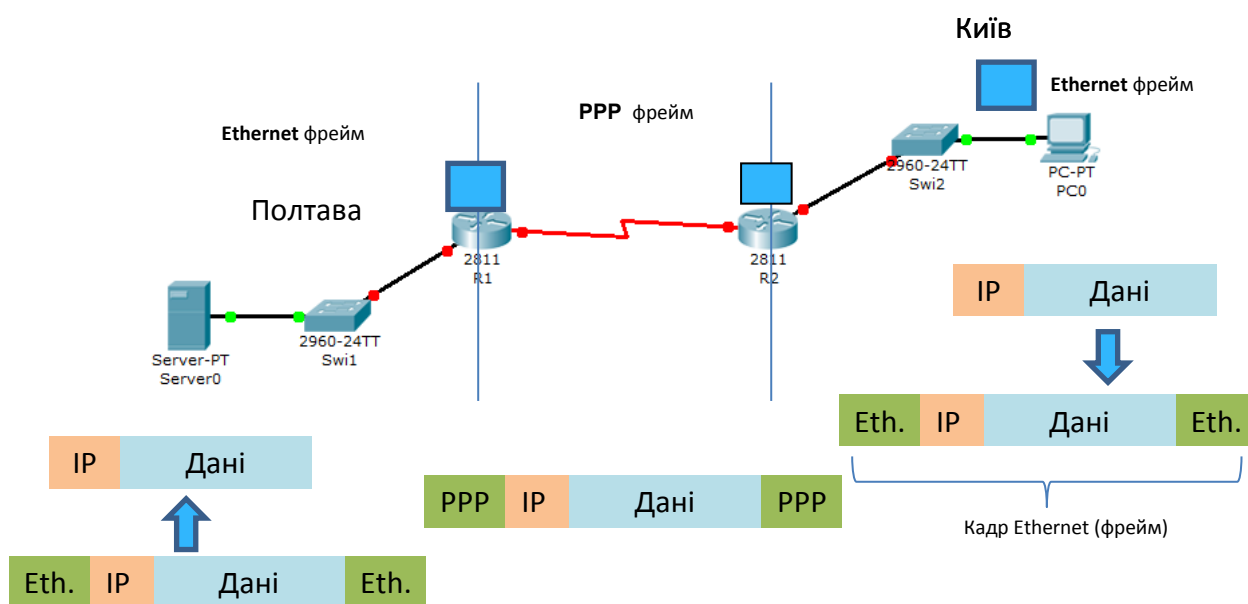
193.162.230.0

Рівень Інтернету



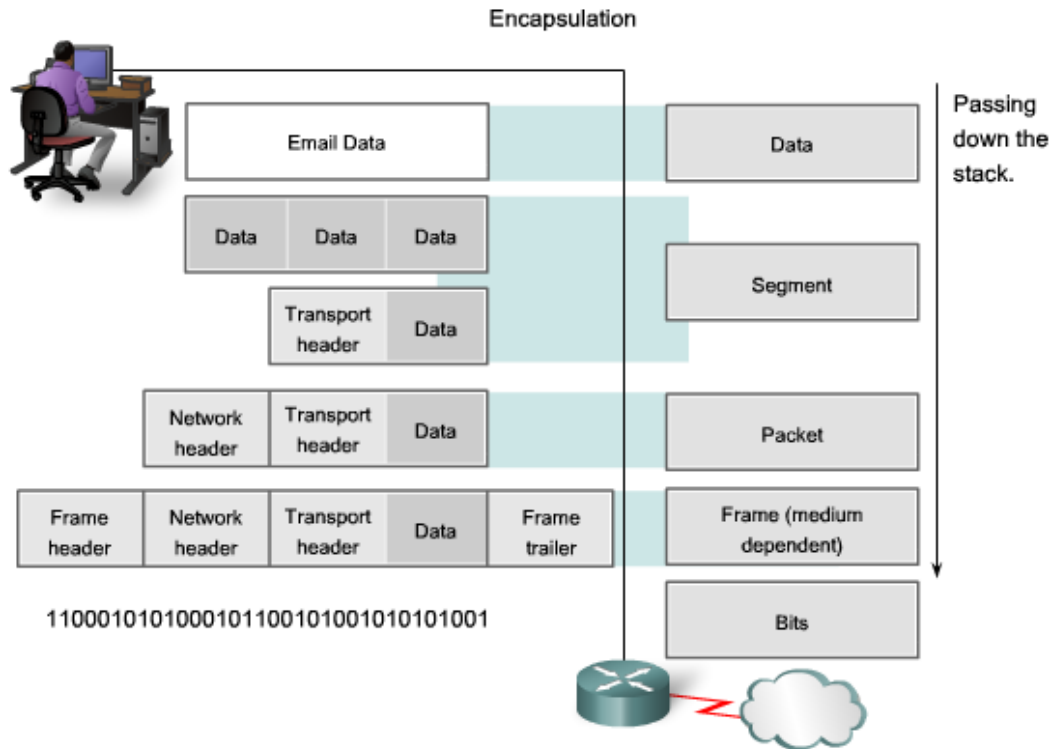
Як можна бачити з рисунку, на рівні Інтернету шляхом додавання до TCP-сегменту заголовку протоколу IP формується IP-пакет. Розмір пакету складає від 64 до 1500 байт. Заголовок IP-пакету, який також називають датаграмою, містить адресну і керуючу інформацію, наприклад, IP-адресу відправника і IP-адресу одержувача. Просування пакетів по мережі здійснюється пристроями, що мають назву маршрутизаторів. Пакети переходять від маршрутизатора до маршрутизатора, до місця призначення. При отриманні пакету кожний маршрутизатор вибирає найкращий маршрут його просування до мережі призначення. Процес вибору маршруту називається *маршрутизацією*.

Рівень доступу до мережі



Рівень доступу до мережі стандартизує апаратне забезпечення і протоколи, що використовуються для передачі даних по різних фізичним мережам. Термін «доступ до мережі» означає, що саме цей рівень визначає, як саме вузол підключається до фізичного середовища передавання. Наприклад, технологія локальних мереж і протокол Ethernet є добрим прикладом рівня доступу до мережі. Разом з тим, рівень доступу визначає стандарти і протоколи, що використовуються для побудови розподілених мереж (Wide Area Network - WAN). Рівень доступу до мережі також надає певні послуги протоколам верхнього рівня, наприклад, протоколу IP. Протокол IP розпізнає тільки загальну структуру мережі, а саме: як маршрутизатори з'єднані між собою, які вузли і до яких сегментів мережі підключені, яка використовується схема адресування. Інформація про те, які технології використовуються для передавання даних на нижчих не включена до протоколу IP, тому що можуть використовуватися самі різні технології і середовища передавання: дротові, бездротові, мідні та оптичні, високошвидкісні та низькошвидкісні. До рівня доступу відносяться безліч мережевих протоколів: Ethernet, Point-To-Point-PPP, Frame Relay, ATM тощо.

Інкапсуляція



Таким чином, для опису даних різних рівнів в моделі TCP/IP використовуються такі поняття, як *сегмент*, *пакет*, *фрейм*.

В моделі OSI використовується загальний термін – **протокольний блок даних** (Protocol Data Unit, PDU).

PDU – це інкапсульовані дані, до яких доданий заголовок (header) і кінцевик (trailer) протоколу відповідного рівня .

Ping і Tracert

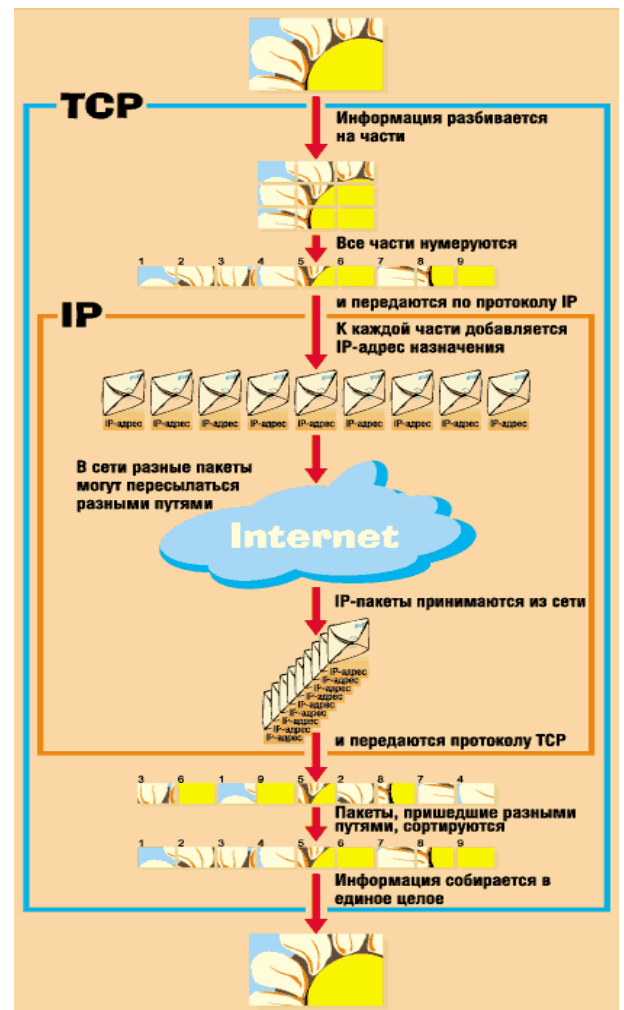
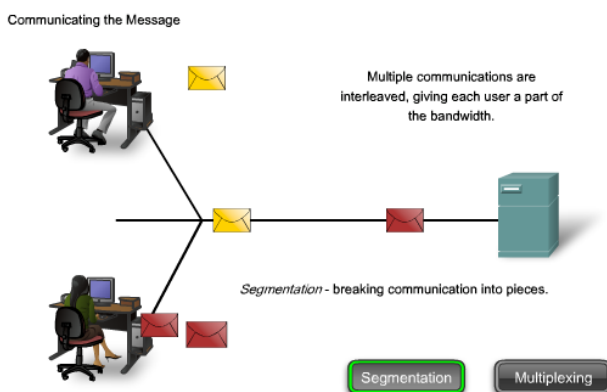
Перевірити підключення вузла до мережі дозволяє утиліта Ping.

Ping перевіряє наскрізне підключення між відправником і одержувачем, а також показує час проходження тестових пакетів.

Tracert дозволяє виявити проблемну ділянку між відправником і одержувачем, у випадку коли перевірка утилітою Ping показала негативний результат. Tracert відстежує шлях від відправника до одержувача. Кожний маршрутизатор вважається ділянкою шляху.

PathPing

Організація взаємодії протоколів TCP і IP



Фізичний рівень

- * Виконує функції перетворення даних, що передаються, в електричні (або світлові) сигнали, які поширюються по кабелю, а також зворотне перетворення.
- * На даному рівні визначаються характеристики фізичного середовища і параметри сигналів

Канальний рівень

- * Даний рівень включає функції управління доступом до мережі, формування кадрів даних, а також виявлення помилок передачі.

Мережний рівень

- * Мережний рівень включає функції комутації і маршрутизації пакетів з інформацією в мережах, що складаються з декількох простих мереж, а також буферизацію даних і регулювання потоків даних.

Транспортний рівень

- * Виконує функції нумерації переданих пакетів і контроль порядку їх проходження, а також узгодження різноманітних мереж між собою, наприклад, двох локальних мереж або локальної мережі і глобальної мережі

Сеансовий рівень

- * Виконує функції перетворення імен абонентів у мережні адреси, керування доступом до мережі на основі заданих прав доступу, а також взаємодії абонентів, що беруть участь у сеансі зв'язку

Представницький рівень

- * Даний рівень включає в себе функції трансляції форматів і синтаксису прикладних програм у форму, зручну для мережі, а також шифрування даних і їх стискування (у разі потреби).

Прикладний рівень

- * Рівень додатків – це функції підтримки прикладного програмного забезпечення кінцевого користувача.