

## Архітектура протоколів TCP/IP. Огляд базових протоколів.

### Модель обміну даними OSI

ISO (International Standards Organisation) пропонує нам використовувати розроблену нею архітектурну модель, яка називається **Open System Interconnect (OSI) Reference Model** – опорна модель взаємодії відкритих систем - OSI модель.

7 рівнів

<b>Прикладний</b>	мережеві аплікації
<b>Рівень представлення</b>	стандартизує представлення даних в аплікаціях
<b>Сеансовий</b>	управляє сеансами взаємодії аплікацій
<b>Транспортний</b>	забезпечує виявлення та корекцію помилок передачі даних
<b>Мережевий</b>	управляє мережевими з'єднання на користь вищих рівнів
<b>Канальний</b>	забезпечує доставку даних по фізичному каналу
<b>Фізичний</b>	визначає фізичні характеристики мережевого каналу

Рівень не містить визначення конкретного протоколу. Функціональність рівня може бути зреалізована будь-яким числом протоколів.

Тобто кожен рівень містить довільне число протоколів, що реалізують його функціональність. Напр http, ftp - протоколи одного (прикладного) рівня.

Кожен протокол реалізує взаємодію тільки з протоколом рівного положення. **Протокол рівного положення** – це реалізація ТОГО Ж протоколу на еквівалентному рівні віддаленої системи.

Протокол одного рівня НЕ ТЕ САМЕ що протокол рівного положення.

Ще один важливий момент: передача даних МІЖ рівнями в межах однієї системи.

В передачі даних приймають участь всі рівні. СТЕК. Вищий рівень покладається в процесі передачі даних на нижчий і не знає нічого про решту.

Дані передаються **вниз** по стеку і в кінці кінців будуть передані по фізичній мережі протоколом найнижчого рівня. На приймаючій стороні дані передаються по стеку **вверх** до приймаючої аплікації: від нижчого протоколу до вищого.

Окремий рівень може не знати, як працюють сусідні. Необхідно знати тільки спосіб передачі і отримання даних (а це і описується протоколами).

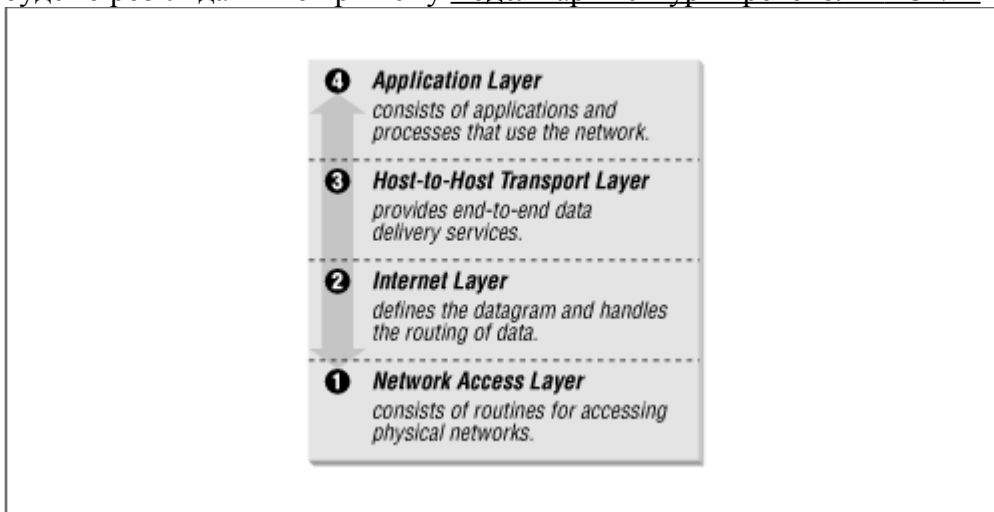
Ізоляція функцій мережевої взаємодії дозволяє мінімізувати вплив різноманітних техногічних змін на все сімейство протоколів цілому.

Тип мережевої карти не портебує зміни браузера і зміна клієнта е-пошти не потребує заміни мережевого кабелю.

## Архітектура протоколів TCP/IP

Немає загальноприйнятого стандарту, як описувати протоколи TCP/IP в термінах моделі OSI. 3-5 рівнів.

будемо розглядати 4-ох рівневу модель архітектури протоколів TCP/IP



<b>Прикладний рівень</b>	Складається з окремих аплікацій і процесів	
<b>Транспортний рівень</b>	Забезпечує функціональність доставки даних	TCP, UDP
<b>Рівень Інтернет</b>	Дає визначення дейтаграм (базових фрагментів передачі даних) і відповідає за маршрутизацію	IP, ICMP
<b>Рівень доступу до мережі</b>	Складається з ф-цій доступу до фізичних мереж.	

Така модель добре передає особливості рівневої ієрархії протоколів.

Як і в моделі OSI дані передаються вниз по стеку, переправляються по мережі, а тоді передаються по стеку вверх.

При проходженні даних крізь стек додаються **заголовки (інкапсуляція даних)**

Прикладний рівень	дані
Транспортний рівень	Заголовок, дані
Рівень Інтернет	Заголовок (Заголовок, дані)
Рівень доступу до мережі	Заголовок (Заголовок (Заголовок, дані))

(Заголовок, дані) – це дані для трансп. рівня

Заголовок – **header** – блок управляючої інформації.

Доповнення інформації на кожному рівні заголовками називається **інкапсуляцією**.

### **Рівень доступу до мережі**

реалізується протоколами найнижчого рівня, драйверами.

ARP – Address Resolution Protocol – відображення IP адрес в фізичні адреси, прийняті в конкретній мережі.

Саме завдяки цьому рівню маємо незалежність TCP/IP від апаратного забезпечення.

З появою нового мережевого обладнання розробляються нові протоколи цього рівня (низькорівневі протоколи).

### **Рівень Інтернет**

#### **Найважливіший протокол цього рівня – IP.**

IPv4 – сучасна версія

IPv5 – експериментальна версія (в мережах реального часу)

IPv6 – нова версія з багатьма нововведеннями (“довга” IP-адреса, покращена маршрутизація etc) несумісна з IPv4. Причина розробки – мала к-сть адрес в IPv4.

Незважаючи на те, що кореневі dns (зовсім віднедавна!) вже включили підтримку IPv6, ми будемо говорити про IPv4, оскільки жити він буде ще дуже і дуже довго.

#### **Ф-ціональність IP:**

- визначає дейтаграму (базову одиницю передачі інформації в мережі)
- визначає схему Інет адресації
- здійснює обмін даними між рівнями доступу до мережі та транспортним
- здійснює маршрутизацію (routing) дейтаграм
- відповідає за розбиття і зборку дейтаграм

Особливості:

IP працює **без створення логічних з'єднань**: передача даних **НЕ вимагає підтвердження зв'язку, не відбувається обмін управляючою інформацією**. Ця ф-ція покладена на прктоколи вищих рівнів.

IP працює **без контролю помилок передачі даних**. В цьому сенсі називається т.зв.

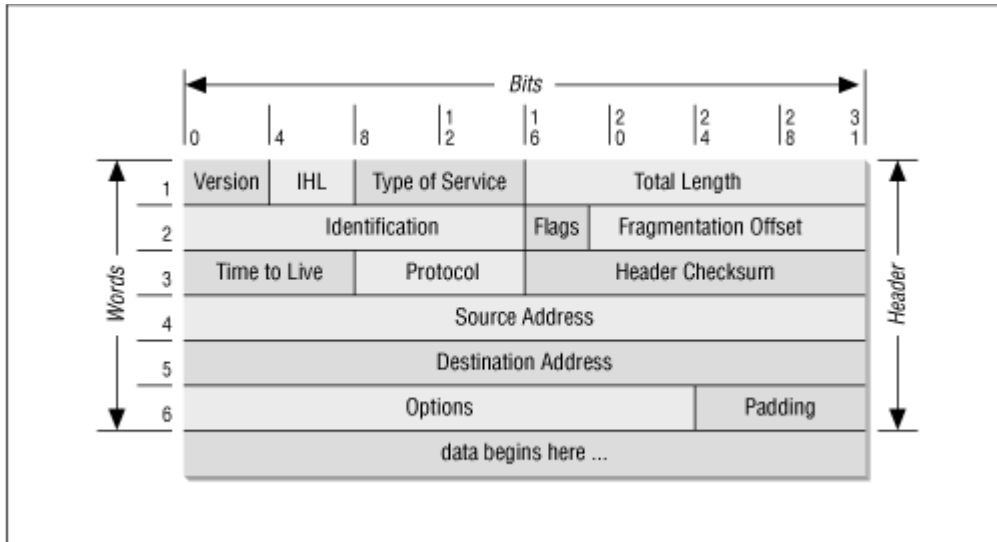
**“ненадійним”** протоколом. Однак на нього можна “покластися” в плані точної доставки даних в доступну мережу. Неможливо лише перевірити, чи дані були коректно отримані.

Дейтаграма (datagram).

Мережі на основі комутації пакетів (дейтаграм).

Пакет – аналогія з поштовим листом. Пакет – блок даних, що містить інформацію про його доставку.

Формат пакету:



IHL – Internet Header Length – довжина заголовку 5 або 6 32-бітних слів слів.  
5-те слово – адреса доставки.

IP займається доставкою дейтаграм, звідси його ф-ція

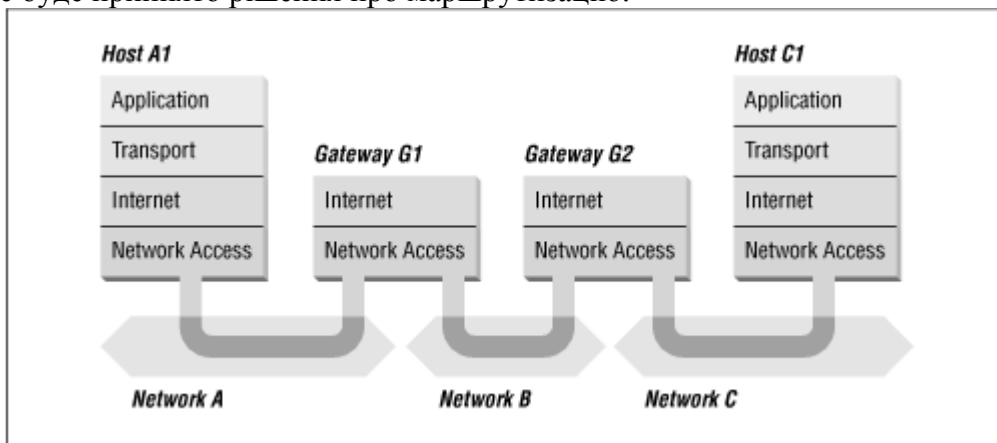
### **Маршрутизація дейтаграм**

в термінології TCP/IP розрізняють 2 типи мережних пристроїв: **шлюзи (gateways) і вузли (hosts)**. Ще, правда, розрізняють шлюзи і маршрутизатори (**routers**).

Шлюзи пересилають пакети між мережами.

Шлюз може входити в кілька мереж.

Шлюзи, на відміну від кінцевих вузлів вузлів, здійснюють обробку пакетів **ТІЛЬКИ ДО РІВНЯ Інет**: поки не буде прийнято рішення про маршрутизацію.



Розбиття дейтаграм.

MTU – Maximum Transmission Unit

Передача на транспортний рівень – на основі третього слова заголовку – (*Protocol*).

### **ICMP – протокол управляючих повідомлень – Internet Control Message Protocol**

служить для:

- управління потоком (якщо напр. швидкість поступлення дейтаграм зависока)
- виявлення недосяжних адресатів
- перенаправлення маршрутів

- перевірка стану віддалених вузлів (ping)

## Транспортний рівень

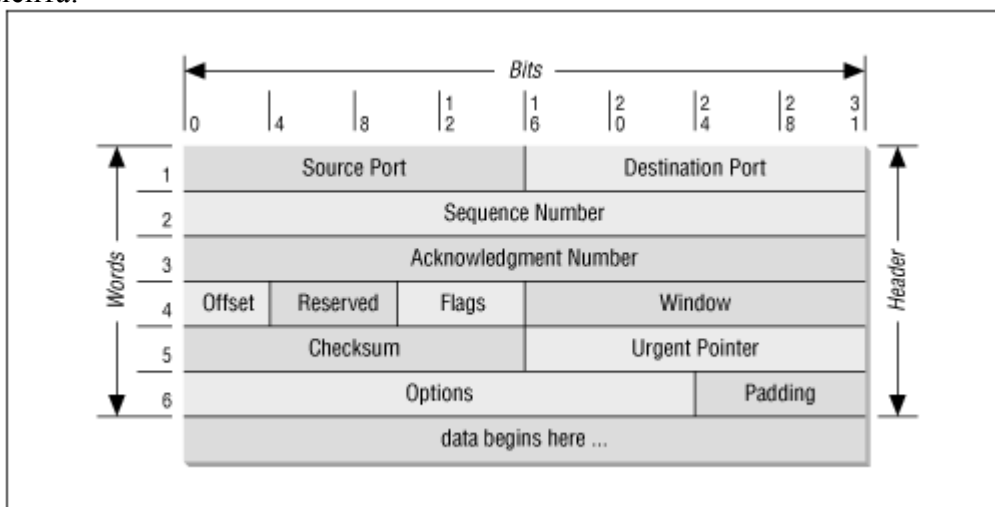
UDP та TCP

### TCP

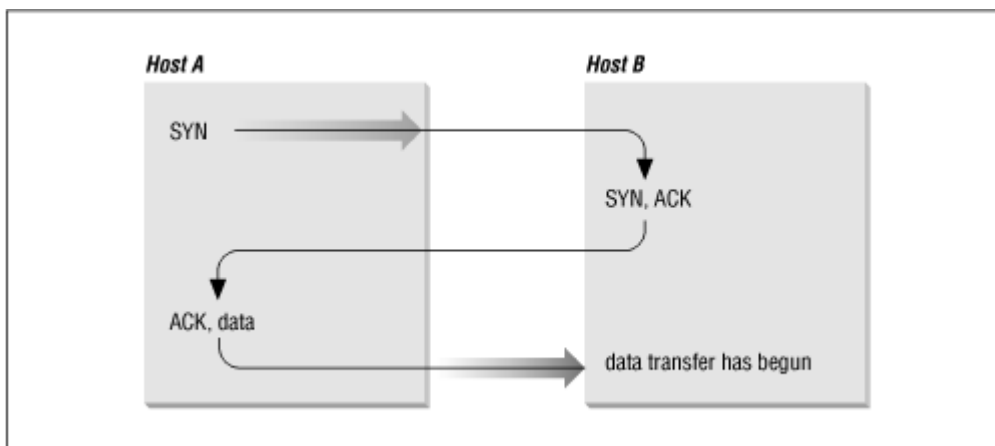
- надійний потоковий протокол, що вимагає створення логічних з'єднань  
 Реалізує механізм *підтвердження прийому з повторною передачею* – дані пересилаються, поки від адресата не буде отримано повідомлення про успішне отримання.

Одиницею обміну для TCP є **сегмент** (так як дейтаграма для IP)

Формат сегмента:



В TCP застосовується т.зв. *механізм потрійного рукоштовування (Three-way handshake)*



Здійснюється обмін трьома сегментами

**SYN** – Synchronize Sequence Number – біт синхронізації порядкових номерів

**ACK** – Acknowledgment – підтвердження (номер, який стане початковим для сегментів даних) після завершення передачі даних теж відбувається обмін у вигляді потрійного рукоштовування, посилається **FIN** (No more data from sender) біт.

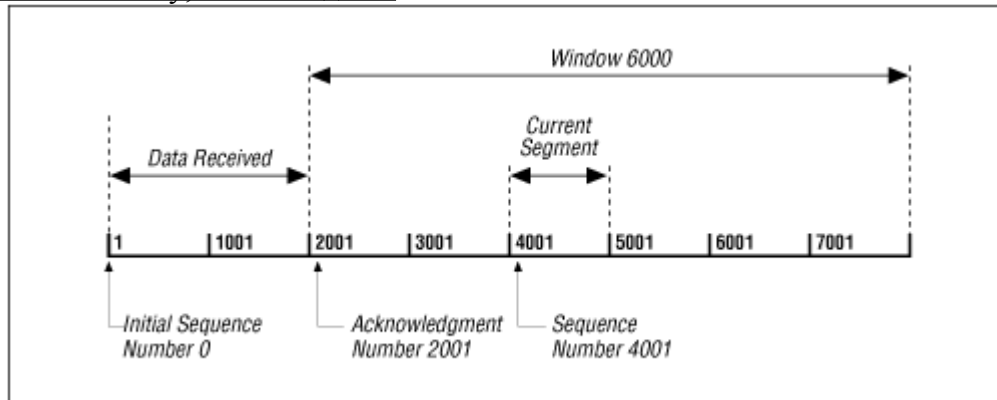
В TCP вважається, що дані є неперервним потоком, тому TCP здійснює контроль за **порядком слідування** сегментів.

Для цього служать поля заголовка 2-ге (Порядковий номер - Sequence Number) і 3-ге (Acknowledgment Number – номер підтвердження).

**Window – розмір вікна.** Вказує скільки даних адресат може прийняти.

Навіть, якщо для певних сегментів ще не отримано підтвердження про доставку, джерело може продовжувати відправку даних, поки не досягне розміну вікна.

Якщо дані не отримано, іде повторна їх передача, але ТІЛЬКИ починаючи з останнього підтверженого сегменту, а не ВСІ дані.



Підтвердження посилається адресатом не для кожного пакету.

**Source and Destination Ports** – порти аплікацій вищого рівня. TCP відповідає таким чином за передачу даних вищому рівню.

### **UDP**

протокол користувацьких дейтаграм. User Datagram Protocol

“ненадійний” протокол передачі (доставки) дейтаграм. БЕЗ встановлення логічного з'єднання.

Використовується в основному там, де передати дані ще раз вигідніше, ніж витратити ресурси на створення логічного з'єднання.

Працює за принципом: **запит – відповідь**, якщо відповіді немає протягом певного часу, аплікація просто повторює запит.

DNS як приклад.

### **Прикладний рівень**

Вершина архітектури TCP/IP.

Маса різних протоколів, служб. Постійно розробляються нові.

Telnet – Network Terminal Protocol

SSH – Secure Shell

SMTP – Simple Mail Transfer Protocol

HTTP – Hypertext Transfer Protocol

FTP – File Transfer Protocol

DNS – Domain Name System