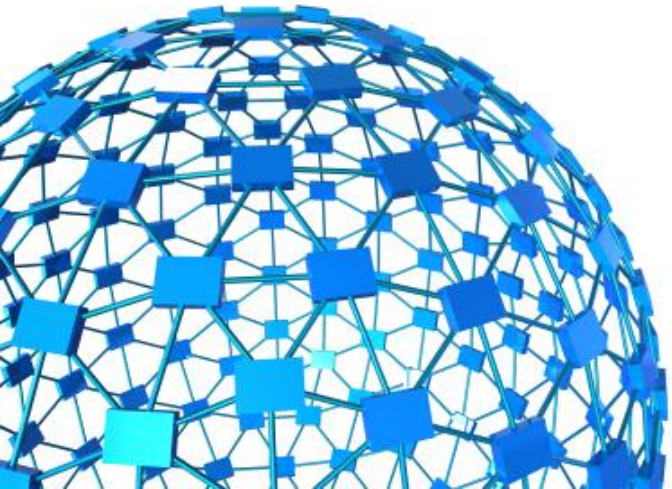
eleks®



# ТРАНСПОРТНИЙ РІВЕНЬ
# ПРОТОКОЛИ TCP/UDP

# ТРАНСПОРТНИЙ РІВЕНЬ ПРОТОКОЛИ TCP/UDP

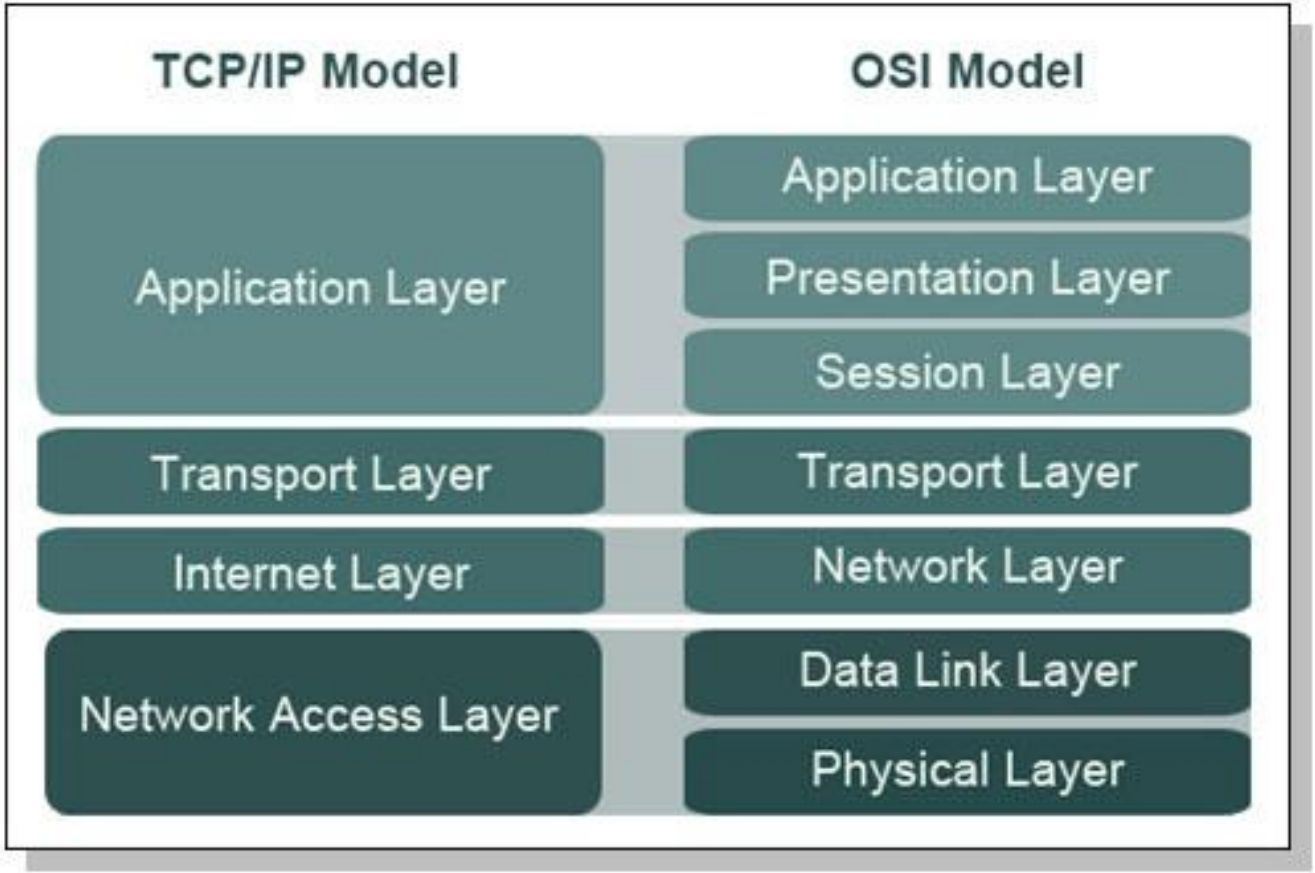
eleks®

Модель OSI
Протоколи TCP/UDP
Основні функції протоколу TCP
Порти
Заголовок сегмента TCP
Приклад заголовка TCP у Wireshark
Встановлення TCP з’єднання
Розрив TCP з’єднання
Керування втратами сегментів
Протокол UDP
Запитання

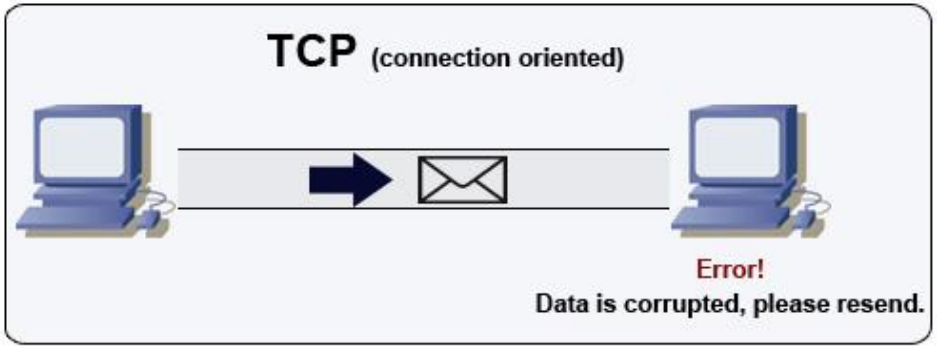
# МОДЕЛЬ OSI

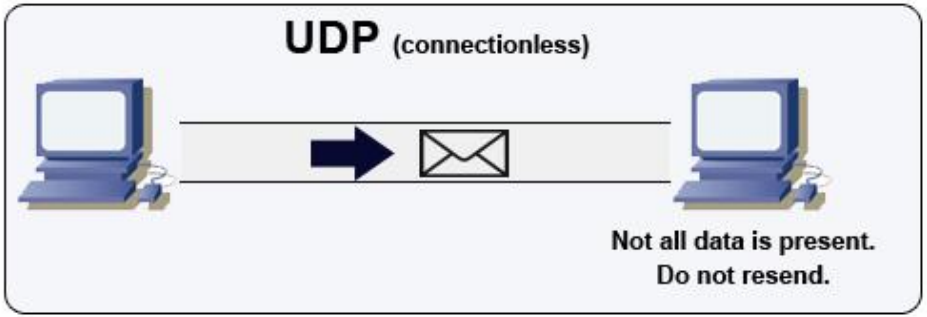
- ПРИКЛАДНИЙ (APPLICATION)
- ПРЕДСТАВНИЦЬКИЙ (PRESENTATION)
- СЕАНСОВИЙ (SESSION)
- ТРАНСПОРТНИЙ (TRANSPORT)
- МЕРЕЖЕВИЙ (NETWORK)
- КАНАЛЬНИЙ (DATA LINK)
- ФІЗИЧНИЙ (PHYSICAL)

# МОДЕЛЬ OSI


TCP/IP Model
OSI Model
Application Layer
Application Layer
Presentation Layer
Session Layer
Transport Layer
Transport Layer
Internet Layer
Network Layer
Network Access Layer
Data Link Layer
Physical Layer

# ПРОТОКОЛИ TCP/UDP



# ОСНОВНІ ФУНКЦІЇ ПРОТОКОЛУ TCP

Основні функції, які виконуються всіма протоколами транспортного рівня:

- Сегментація і реасемблінг
- Мультиплексуваня з'єднань
- Протокол TCP додатково виконує наступні функції
- Попереднє встановлення з'єднання (Establishing a Sesssion)
- Надійна доставка даних (Reliable Delivery)
- Відновлення первинного порядку отриманих даних (Same Order Delivery)
- Керування передачею (Flow Control)

# ПОРТИ

- **Well Known Ports** (Добре відомі порти) - порти з номерами від 0 до 1023. Ці номери зарезервовані для служб і системних програм

- **Registered Ports** (Зареєстровані порти) - порти з номерами від 1024 до 49151. Ці номери портів призначені користувацьким процесам і програмам

- **Dynamic or Private Ports** (Динамічні або приватні порти) - порти з номерами від 49152 до 65535. Ці порти переважно динамічно вибираються клієнтськими програмами при ініціювані зєднання

# ЗАГОЛОВОК СЕГМЕНТА TCP

**TCP Header**

| *Offsets* Octet | | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Octet | Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| 0 | 0 | Source port | | | | | | | | | | | | | | | | Destination port | | | | | | | | | | | | | | |
| 4 | 32 | Sequence number | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 8 | 64 | Acknowledgment number (if ACK set) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 12 | 96 | Data offset | | | | Reserved 0 0 0 | | | | NS | CWR | ECE | URG | ACK | PSH | RST | SYN | FIN | Window Size | | | | | | | | | | | | | | |
| 16 | 128 | Checksum | | | | | | | | | | | | | | | | Urgent pointer (if URG set) | | | | | | | | | | | | | | |
| 20<br>... | 160<br>... | Options (if Data Offset > 5, padded at the end with "0" bytes if necessary)<br>... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

# ЗАГОЛОВОК СЕГМЕНТА TCP

Значення полів заголовка TCP, які містять прапорці, що використовуються в процесі встановлення з'єднання. TCP використовує 6 прапорців (в порядку їх появи)

**URG** – наявність в сегменті термінових даних

**ACK** – сигналізує про необхідність підтвердження

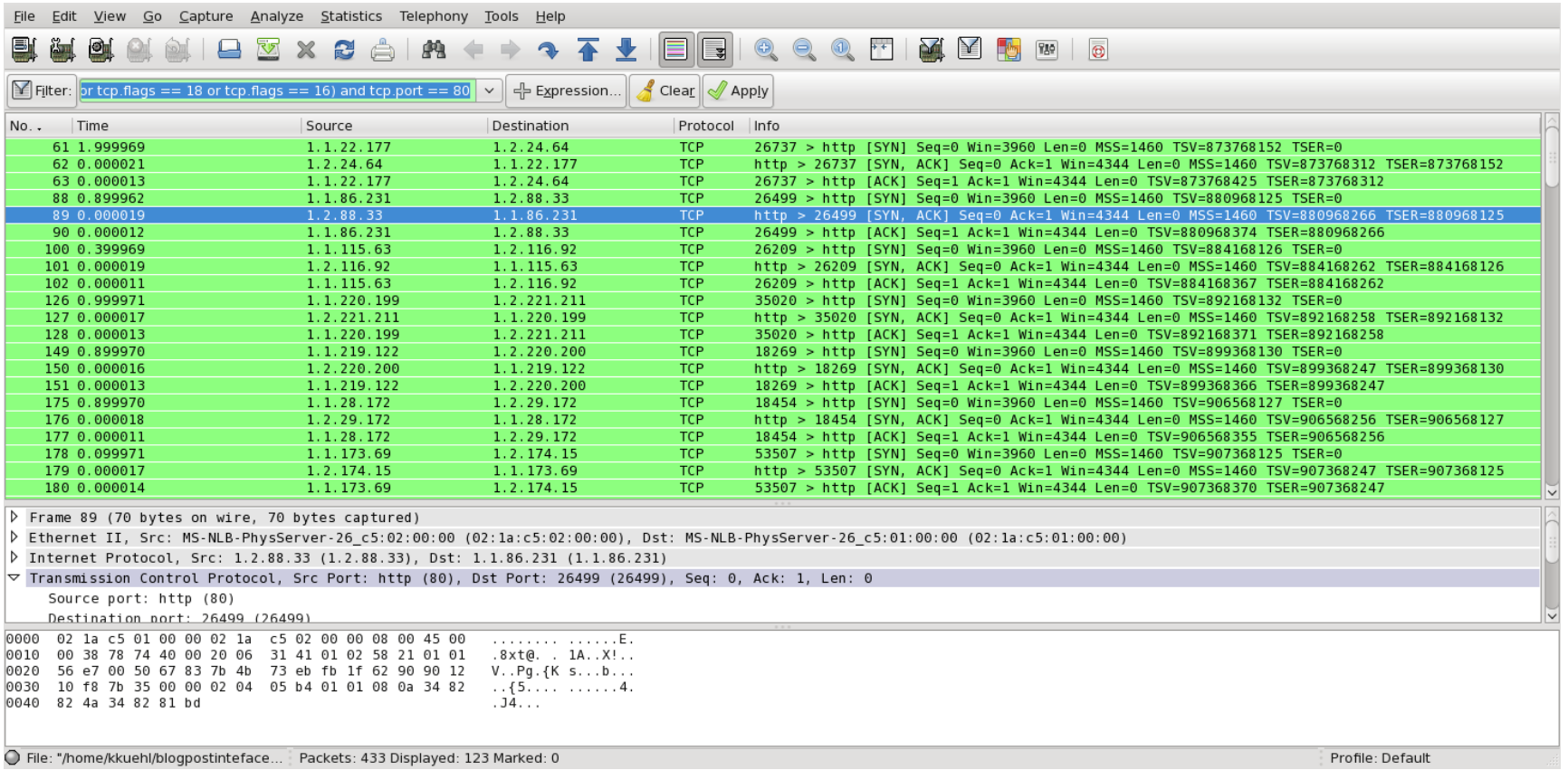
**PSH** – сигнал фінального сегмента в буфері відправника

**RST** – скидування з'єднання або його розрив

**SYN** – запит на встановлення з'єднання

**FIN** - запит на розрив з'єднання

# ПРИКЛАД ЗАГОЛОВКА TCP У WIRESHARK



# ВСТАНОВЛЕННЯ TCP З'ЄДНАННЯ

Процедура **Three-Way Handshake** виконує наступні дії:

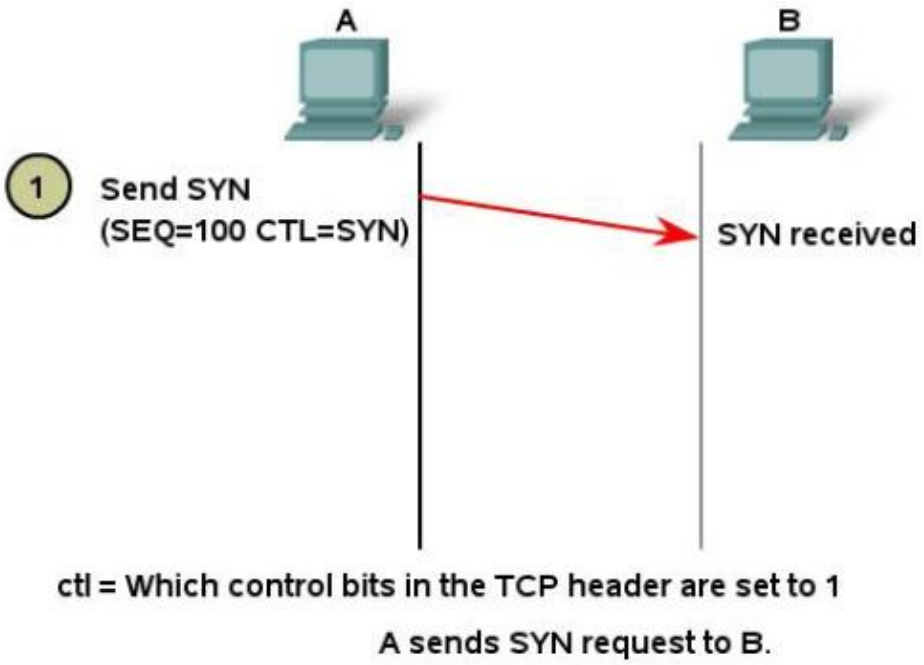
1. Встановлює присутність і працездатність отримувача,

2. Перевіряє, чи на станції-отримувачі запущений процес, який обробляє запити до номера порта серверної програми, з якою встановлюється з'єднання

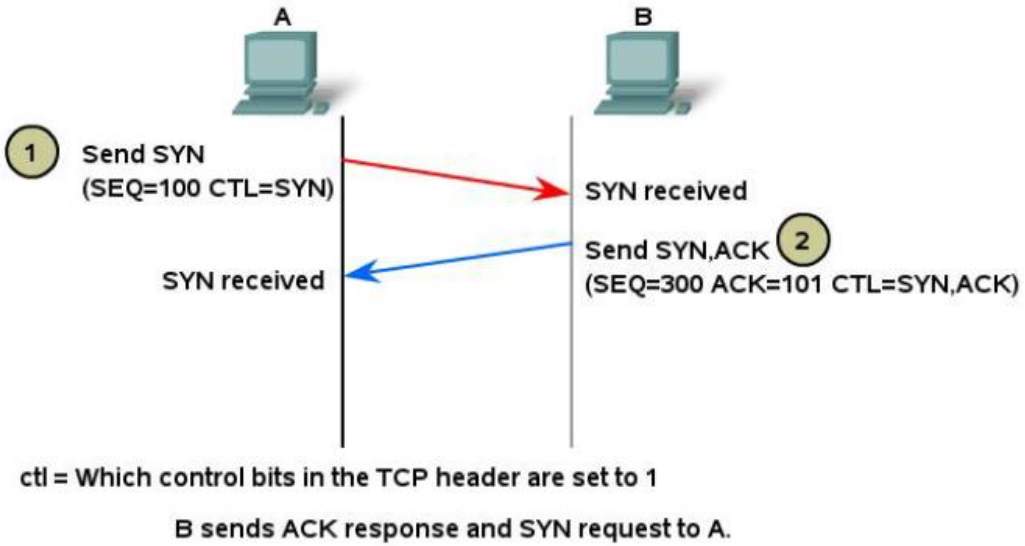
3. Інформує станцію призначення про наміри клієнта встановити з'єднання на цьому номері порта

# ВСТАНОВЛЕННЯ TCP З'ЄДНАННЯ

Клієнт відправляє сегмент, який містить деяке початкове значення (ISN – Initial Sequence Number) і встановлює спеціальний прапорець, який сигналізує про наміри встановити з'єднання



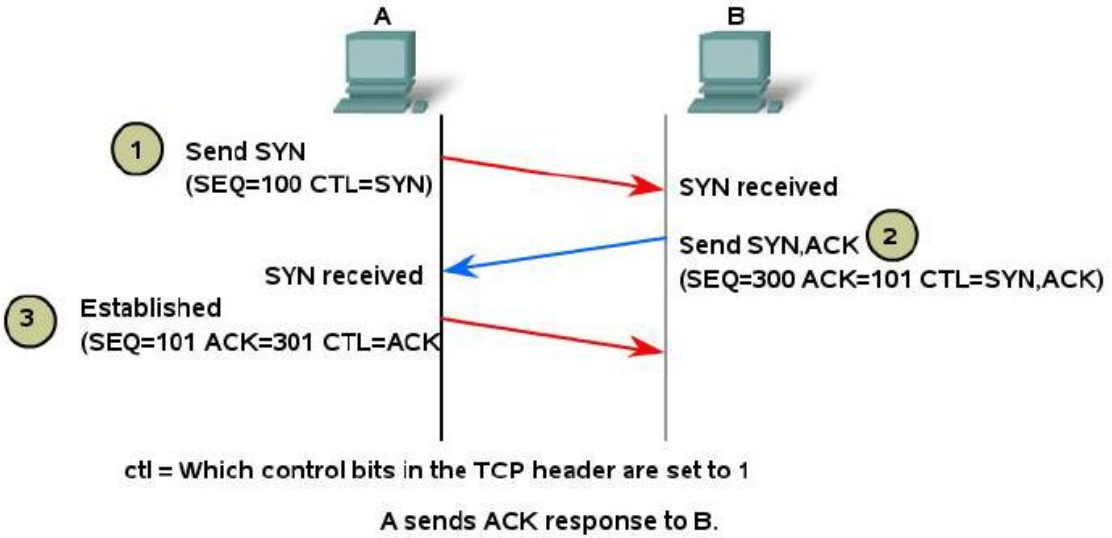
# ВСТАНОВЛЕННЯ TCP З'ЄДНАННЯ




Сервер надсилає у відповідь сегмент з підтвердженням ISN клієнта (ISN клієнта +1) і власним ISN. Обмін цими значеннями дозволяє організувати контроль над передачею даних

# ВСТАНОВЛЕННЯ TCP З'ЄДНАННЯ

Сегмент з підтвердженням клієнтом ISN сервера (ISN сервера + 1) завершує процес встановлення з'єднання

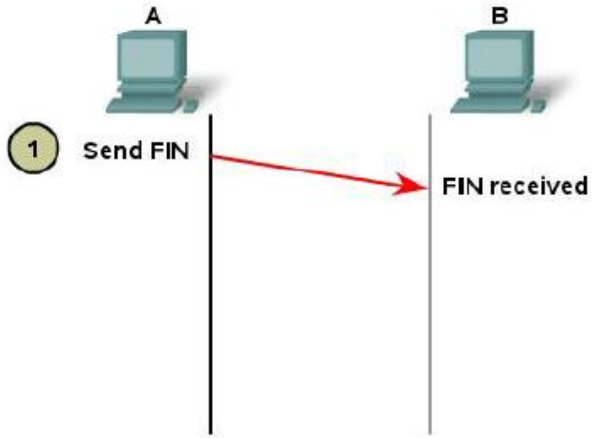


# РОЗРИВ TCP З'ЄДНАННЯ

Процес розриву з'єднання проходить в чотири етапи (хоча насправді кількість етапів розриву з'єднання залежить від розробника конкретного стеку TCP/IP, стандарт передбачає саме чотири етапи)

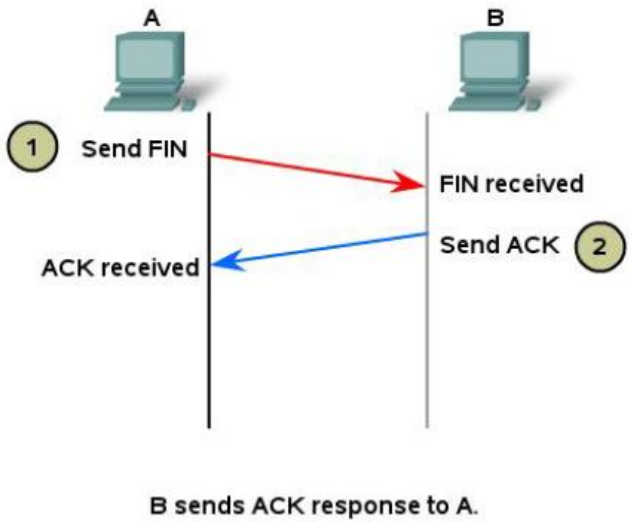
# РОЗРИВ TCP З'ЄДНАННЯ

Станція ініціює розрив з'єднання, відправляючи сегмент без корисних даних із встановленим бітом FIN




A sends FIN request to B.

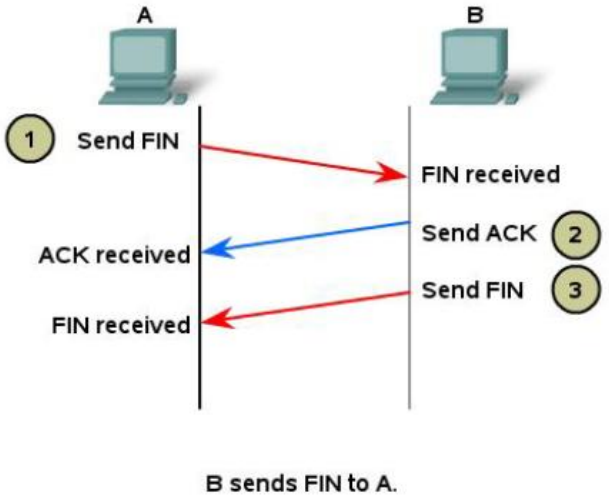
# РОЗРИВ TCP З'ЄДНАННЯ




Партнер підтверджує отримання цього сегмента прапорцем ACK

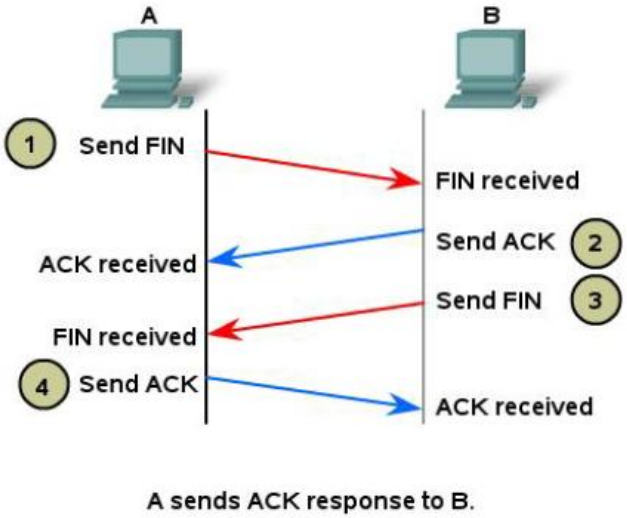
# РОЗРИВ TCP З'ЄДНАННЯ

Партнер відправляє власний сегмент з бітом FIN (в деяких релізаціях стеку протоколів TCP/IP цей крок може бути поєднаний із другим кроком)



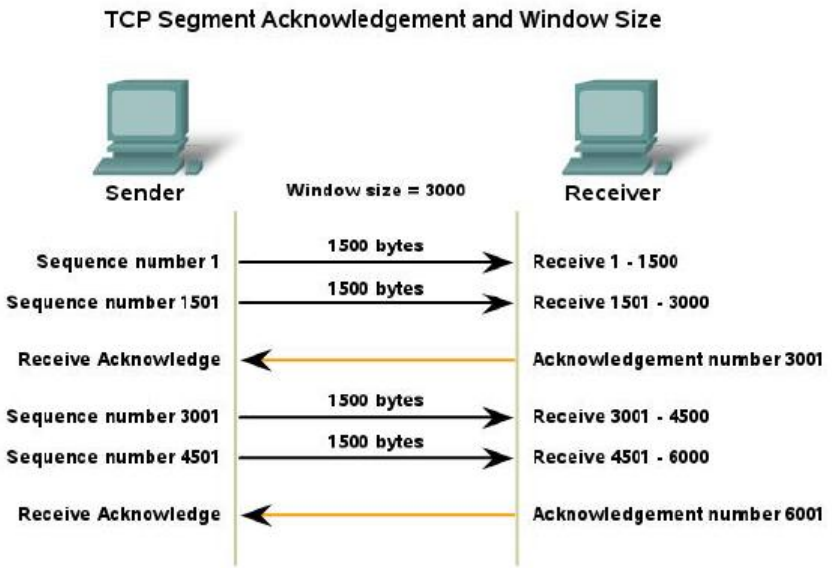

B sends FIN to A.

# РОЗРИВ TCP З'ЄДНАННЯ

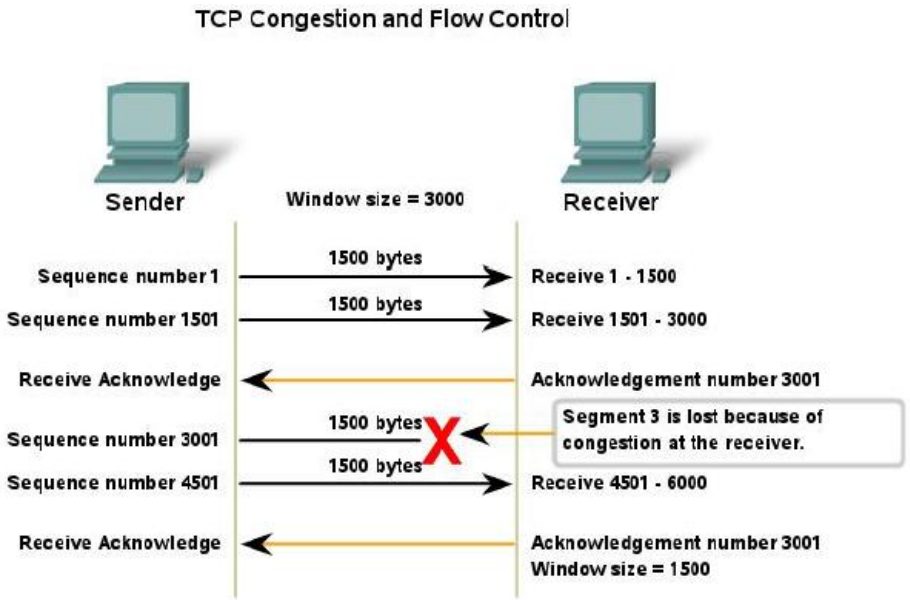



Станція ініціатор розриву підтверджує отримання запиту на розрив зєднання від партнера

# КЕРУВАННЯ ВТРАТАМИ СЕГМЕНТІВ




The window size determines the number of bytes sent before an acknowledgment is expected.
The acknowledgement number is the number of the next expected byte.

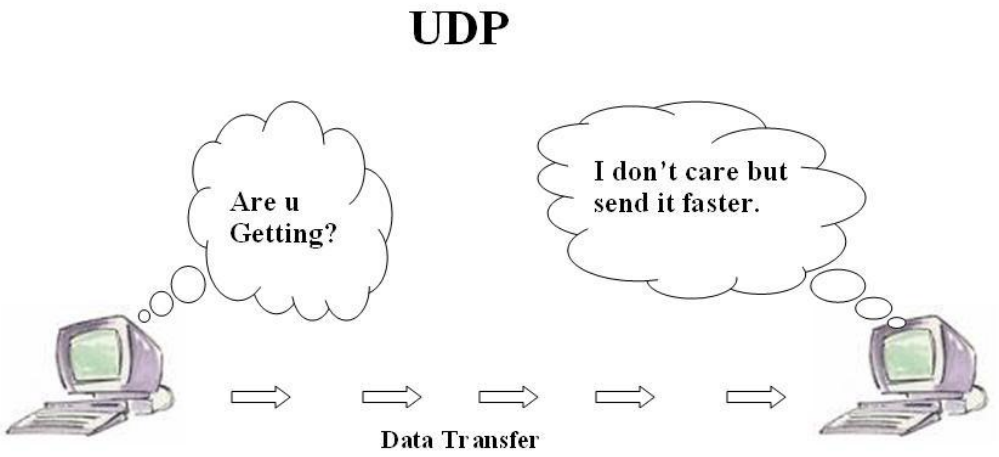



If segments are lost because of congestion, the Receiver will acknowledge the last received sequential segment and reply with a reduced window size.

# ПРОТОКОЛ UDP

Протокол UDP використовують такі сервіси та протоколи вищого рівня:

- TFTP (англ. Trivial File Transfer Protocol, найпростіший протокол передачі файлів),
- SNMP (англ. Simple Network Management Protocol, простий протокол управління мережею),
- DHCP (англ. Dynamic Host Configuration Protocol, протокол динамічної конфігурації вузла),
- DNS (англ. Domain Name System, служба доменних імен).



# ДЕ ПОЧИТАТИ ?

Джерела отримання додаткової інформації:

- http://en.wikipedia.org/wiki/Transmission_Control_Protocol
- http://www.ietf.org/rfc/rfc793.txt
- http://citforum.ru/internet/tifamily/tcpspec.shtml
- http://it.iut.ac.ir/sites/fsites/it/files/u4/uploads/Networking%20Class-Wireshar%20Labs-Solutions/05-Wireshark_TCP_Solution_July_22_2007.pdf
- http://packetlife.net/blog/2010/jun/7/understanding-tcp-sequence-acknowledgment-numbers/

# ЗАПИТАННЯ ?