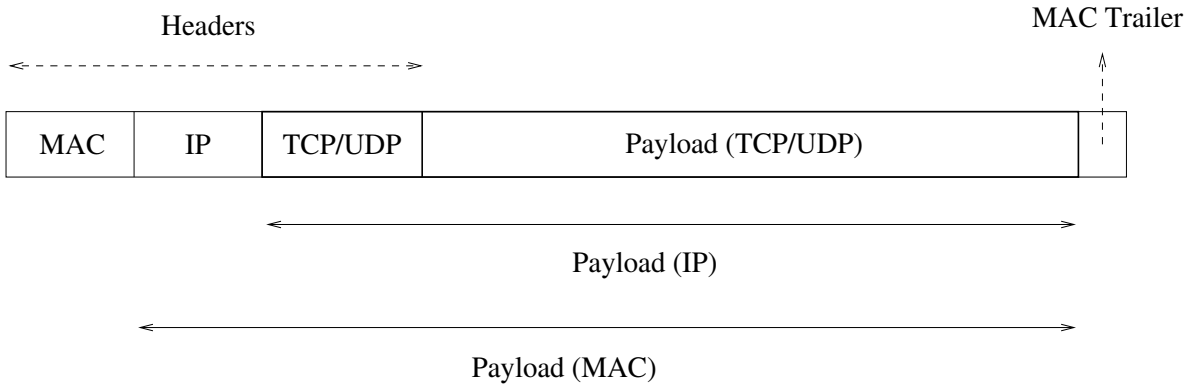
# Transport Layer Protocols

→ end-to-end

→ runs on top of network layer protocols

→ treat network layer & below as black box

Three-level encapsulation: e.g., TCP and UDP




Meaning of protocol "stack"

→ push/pop headers

Network layer (IP) assumptions:

- unreliable
- out-of-order delivery
- absence of QoS guarantees (delay, throughput, etc.)
- insecure (IPv4)

  → IPsec

Additional performance properties:

- works "ok"
- can break down under high load conditions

  → e.g., flash crowds, DDoS and worm attacks
- wide behavioral range

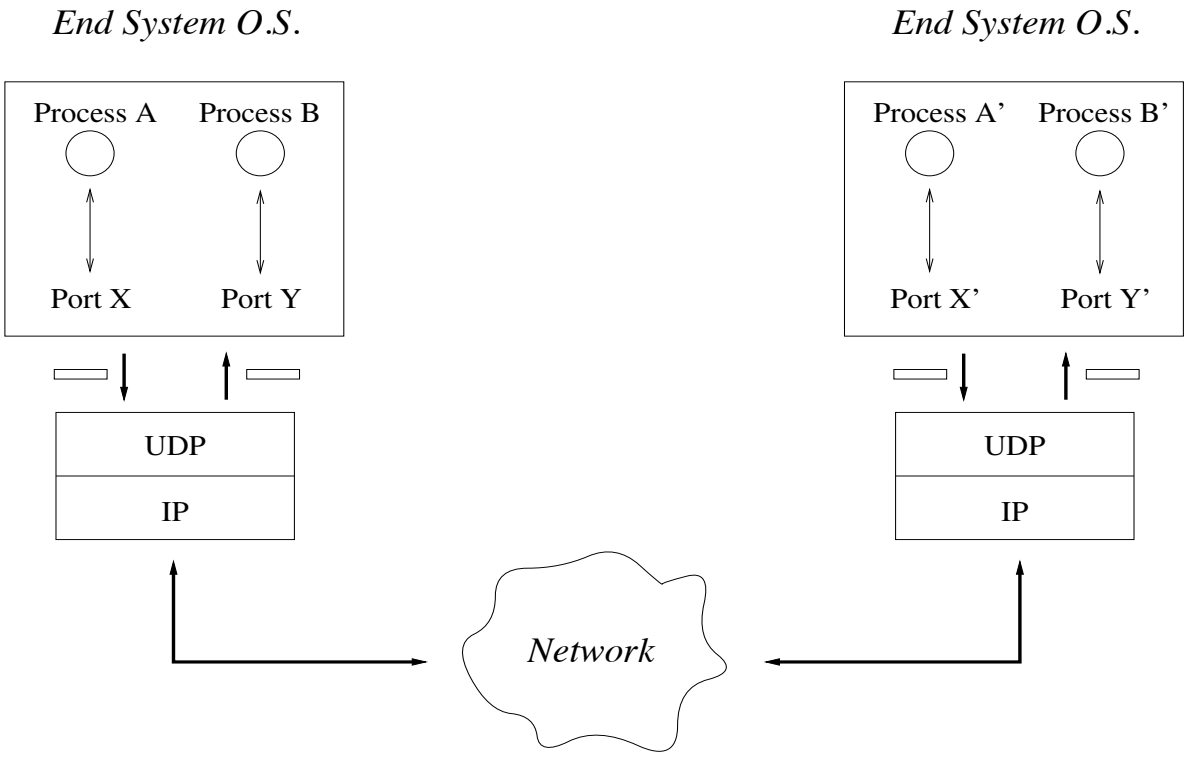
  → from good to bad

Goal of UDP (User Datagram Protocol):

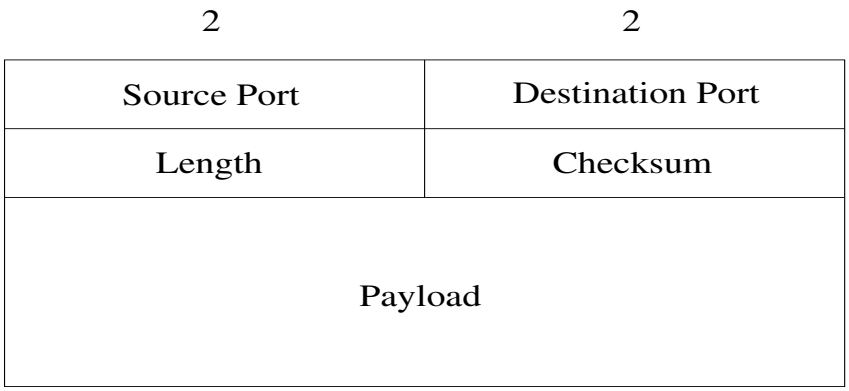
$\longrightarrow$ process identification

$\longrightarrow$ port number as demux key

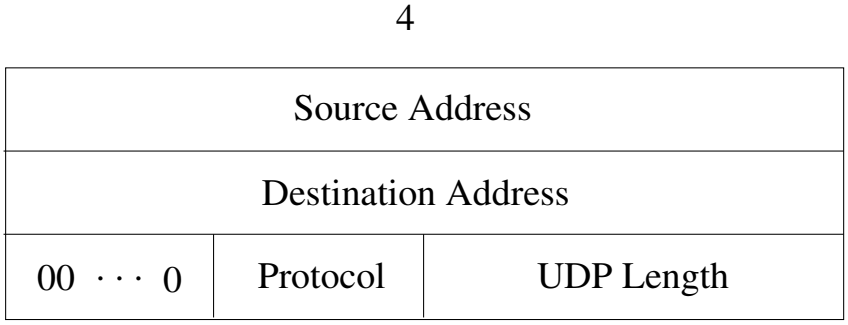
$\longrightarrow$ minimal support beyond IP



UDP packet format:




Checksum calculation: pseudo header




$\longrightarrow$ pseudo header, UDP header and payload

UDP usage:

- multimedia streamining

  → lean and nimble

  → at minimum requires process identification

  → reliability addressed above UDP if needed: FEC preferred over ARQ

  → congestion control addressed above UDP

  → e.g., QUIC for HTTP/3

- lightweight client/server applications

  → reduce persistent state which leads to overhead

Goals of TCP (Transmission Control Protocol):

- process identification
- reliable communication: ARQ
- speedy communication: congestion control
- segmentation and MTU
- connection establishment and tear-down

$\rightarrow$ complex mix of functionalities

Provide stream interface to higher level protocols

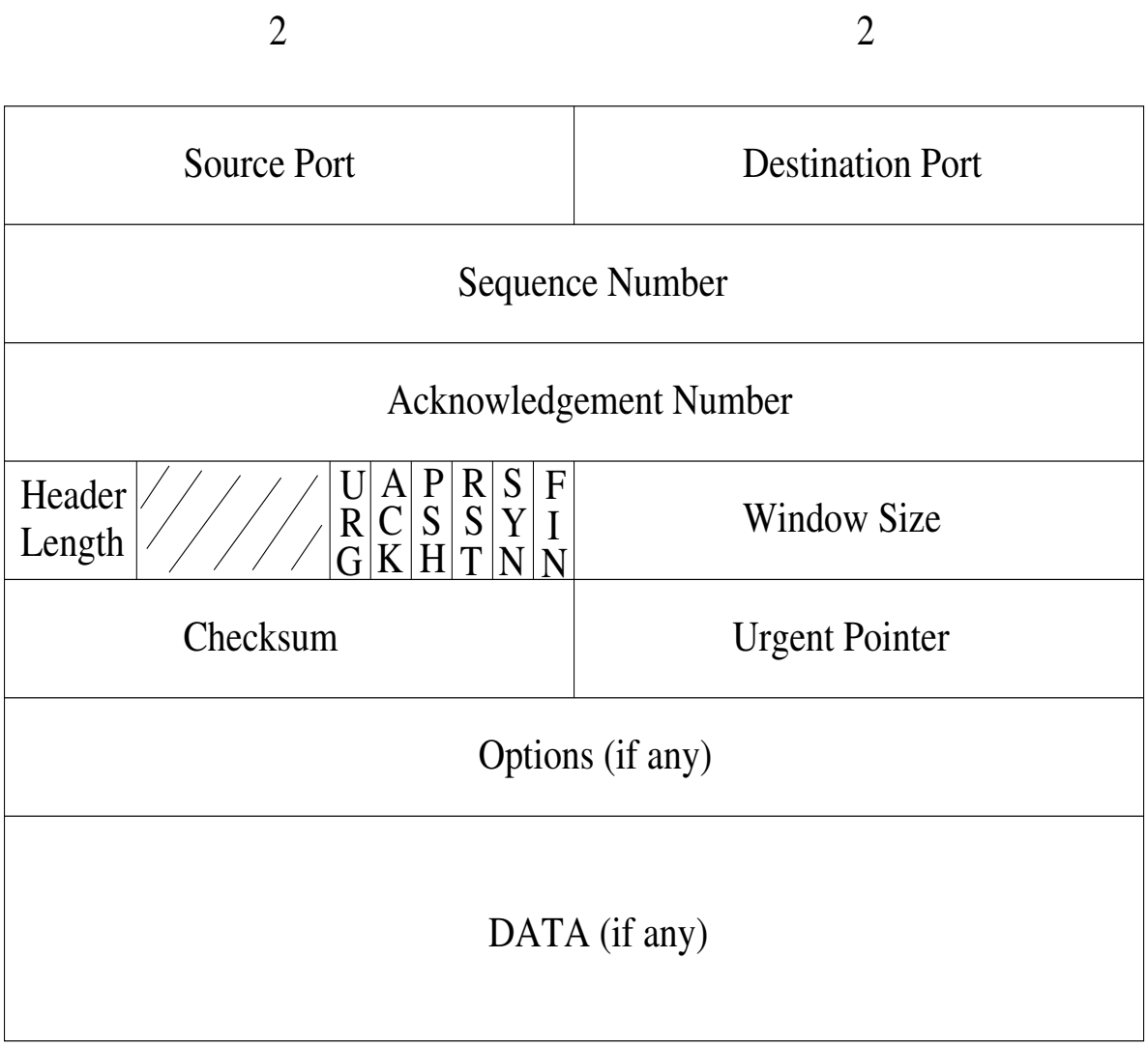
→ exported semantics: contiguous byte stream

→ e.g., accessed using read(), write() system calls

Segmentation:

- segment stream of bytes into blocks of fixed size
- segment size determined by TCP MTU (Maximum Transmission Unit)
- actual unit of transmission in ARQ
- efficiency and reduced fragmentation

# TCP packet format:



- Sequence Number: position of first byte of payload
- Acknowledgement: next byte of data expected (receiver)
- Header Length (4 bits): 4 B units
- URG: urgent pointer flag
- ACK: ACK packet flag
- PSH: override TCP buffering
- RST: reset connection
- SYN: establish connection
- FIN: close connection
- Window Size: receiver's advertised window size
- Checksum: prepend pseudo-header
- Urgent Pointer: byte offset in current payload where urgent data begins
- Options: MTU; take min of sender and receiver

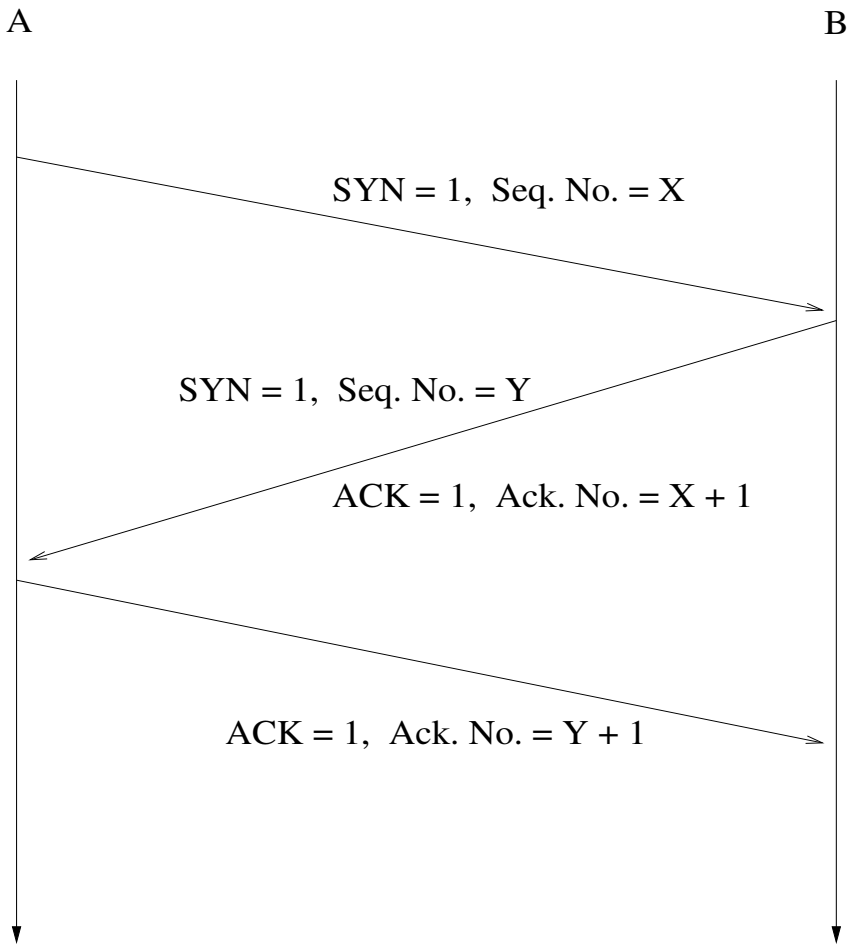
# Checksum calculation: pseudo header

4

| Source Address | | |
|---|---|---|
| Destination Address | | |
| 00 $\cdots$ 0 | Protocol | TCP Segment Length |

$\longrightarrow$ pseudo header, TCP header and payload

TCP connection establishment (3-way handshake):




- $X$, $Y$ are chosen randomly

  $\rightarrow$ mitigate sequence number prediction

- piggybacking

2-party consensus problem: are $A$ and $B$ in agreement about the state of affairs after 3-way handshake?

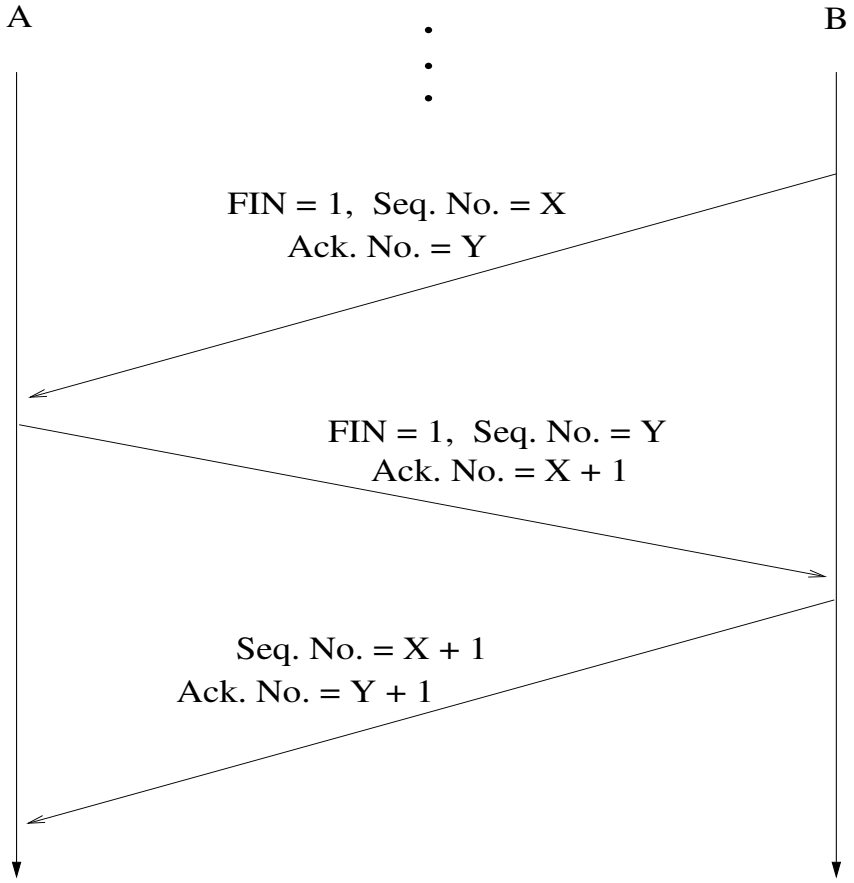
$\rightarrow$ in general: in networks with unbounded delay, there is no solution

$\rightarrow$ acknowledging the ACK ...

$\rightarrow$ lunch date problem

$\rightarrow$ also TCP session termination

TCP connection termination:




- full duplex
- half duplex