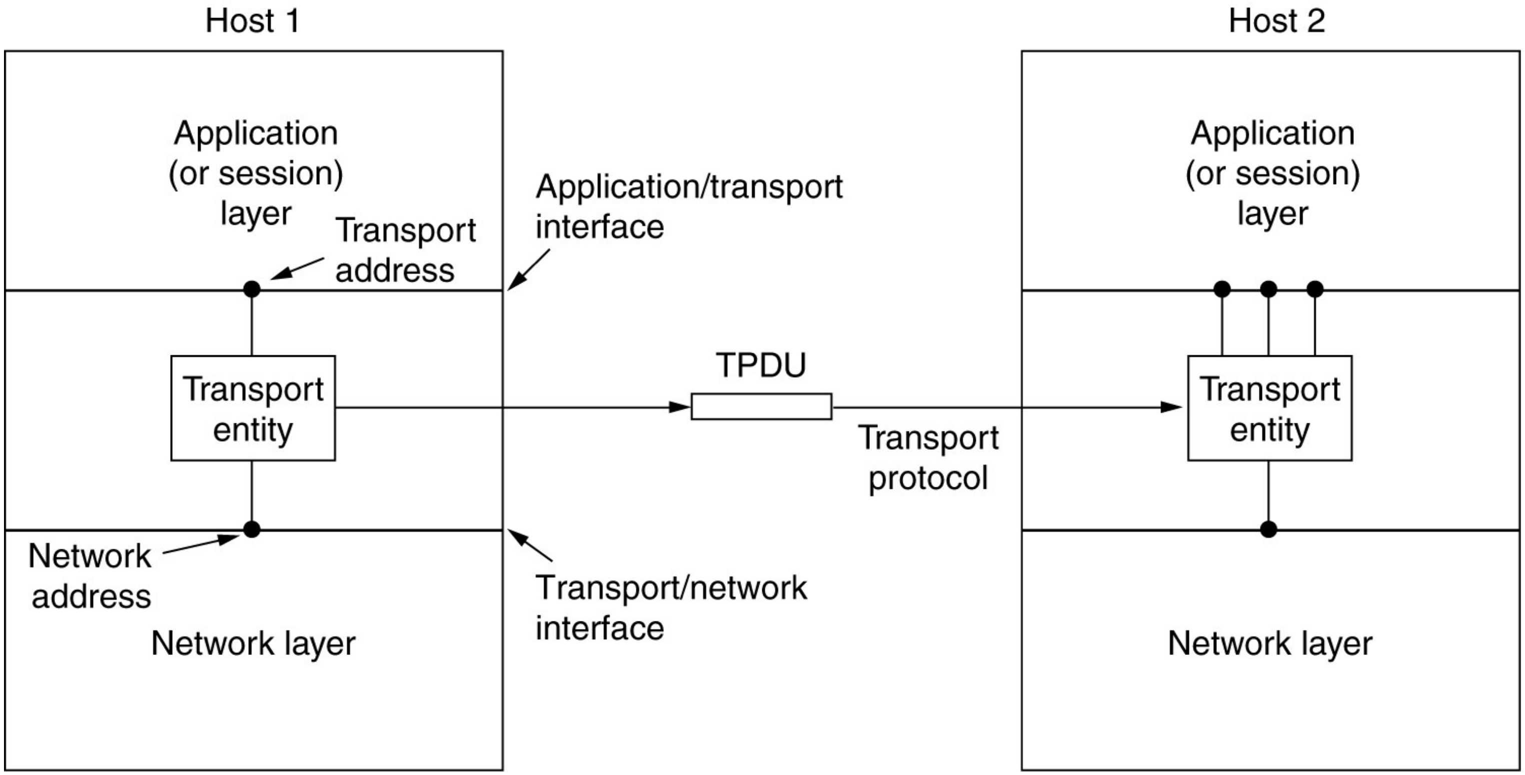
Chapter 6

The Transport Layer

The Transport Service

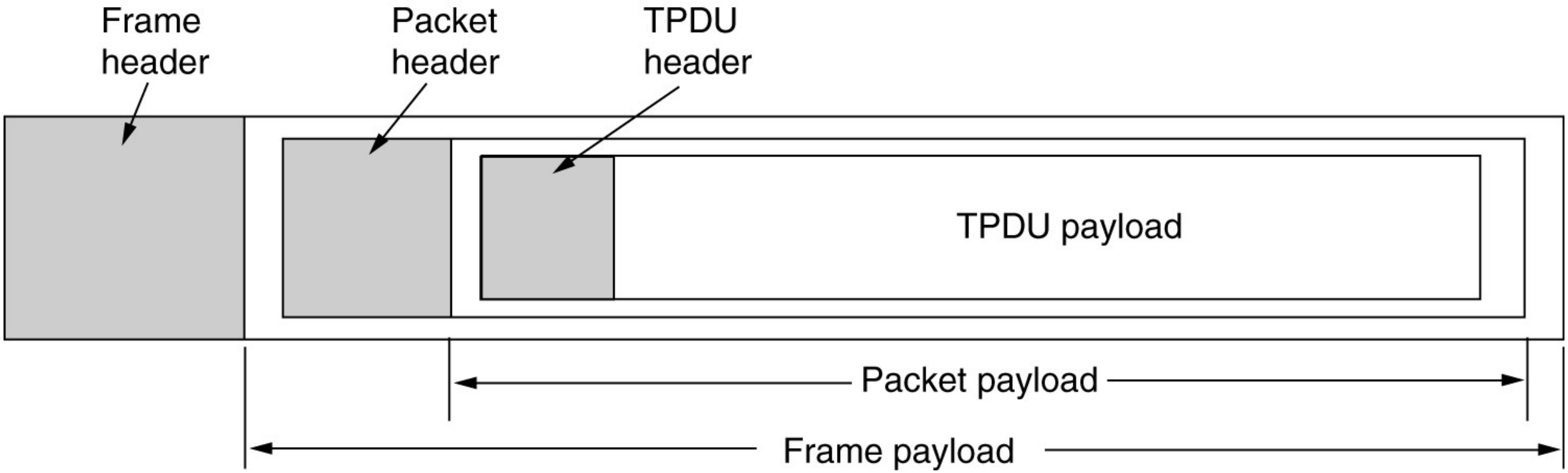
- Services Provided to the Upper Layers
- Transport Service Primitives
- Berkeley Sockets
- An Example of Socket Programming:
  - An Internet File Server
- User Datagram Protocol

Services Provided to the Upper Layers



The network, transport, and application layers.

Transport Service Primitives (2)



The nesting of TPDUs, packets, and frames.

Elements of Transport Protocols

- Addressing
- Connection Establishment
- Connection Release
- Flow Control and Buffering
- Multiplexing
- Crash Recovery

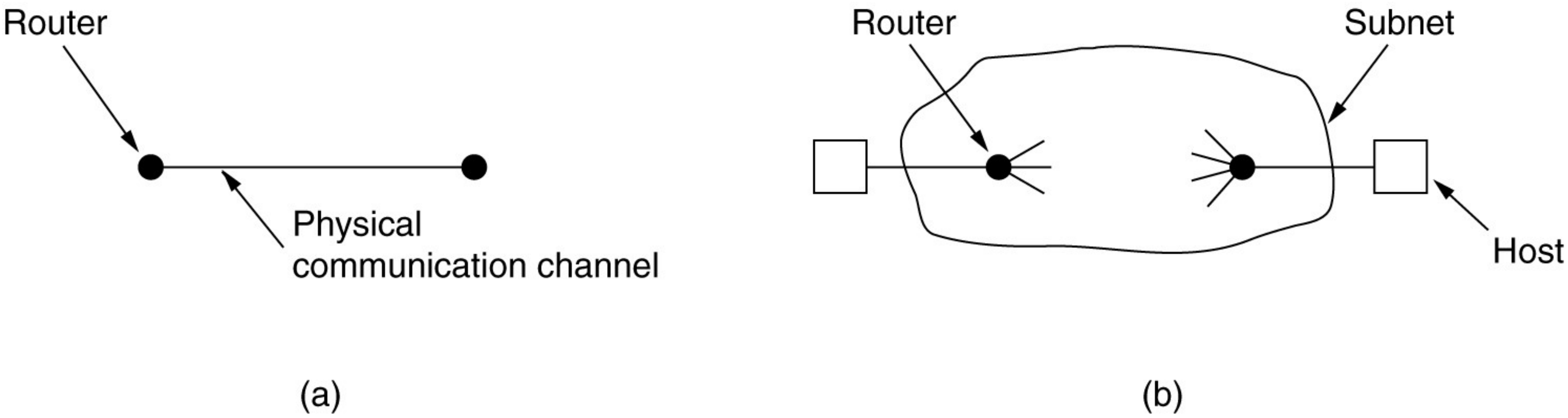
Transport Protocol

Transport Protocols resemble data link protocols:

Error control

Sequencing

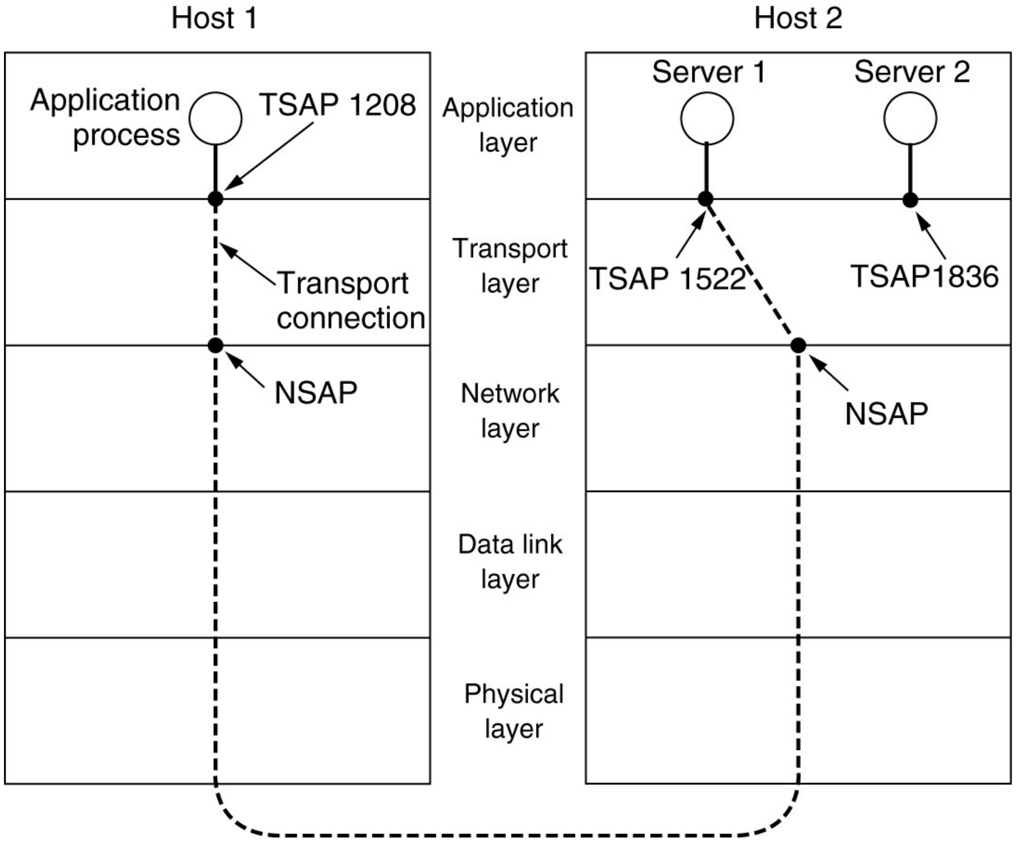
Flow Control



(a) Environment of the data link layer.

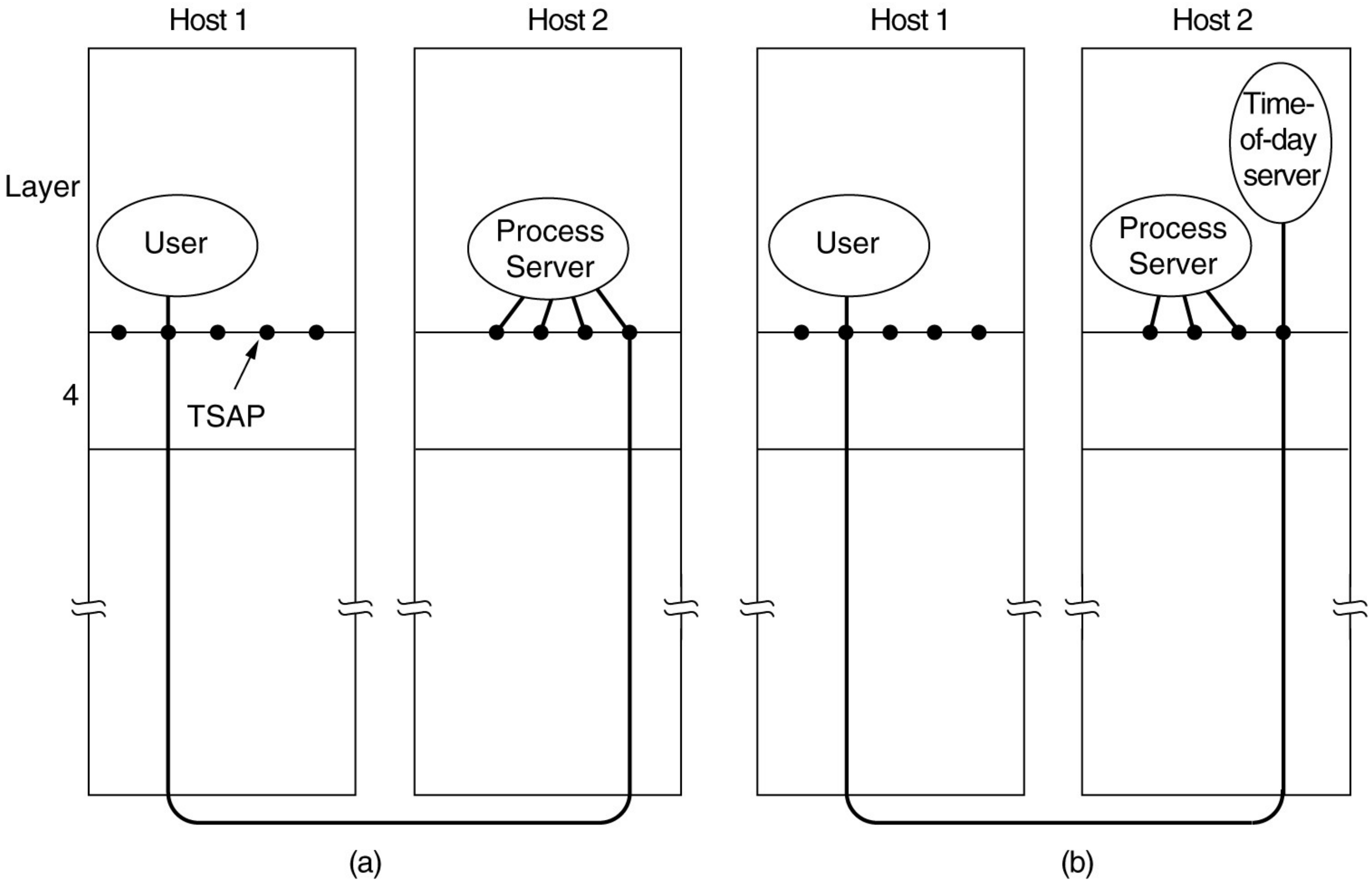
(b) Environment of the transport layer.

Addressing



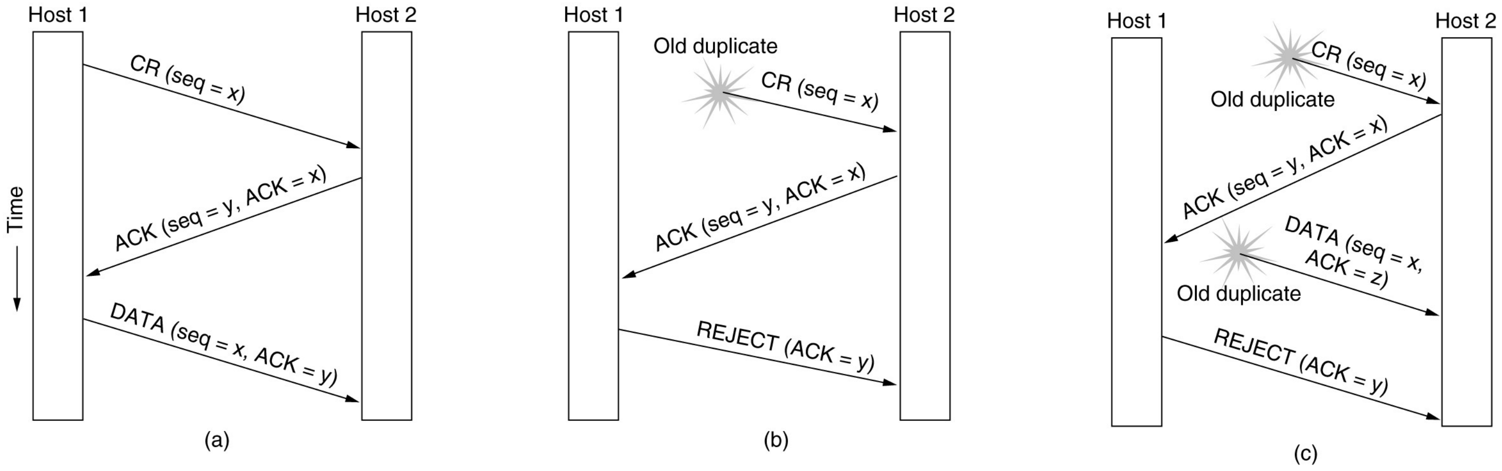
TSAPs, NSAPs and transport connections.

Connection Establishment



How a user process in host 1 establishes a connection with a time-of-day server in host 2 (xinetd).

Connection Establishment (3)



Three protocol scenarios for establishing a connection using a three-way handshake. CR denotes CONNECTION REQUEST.

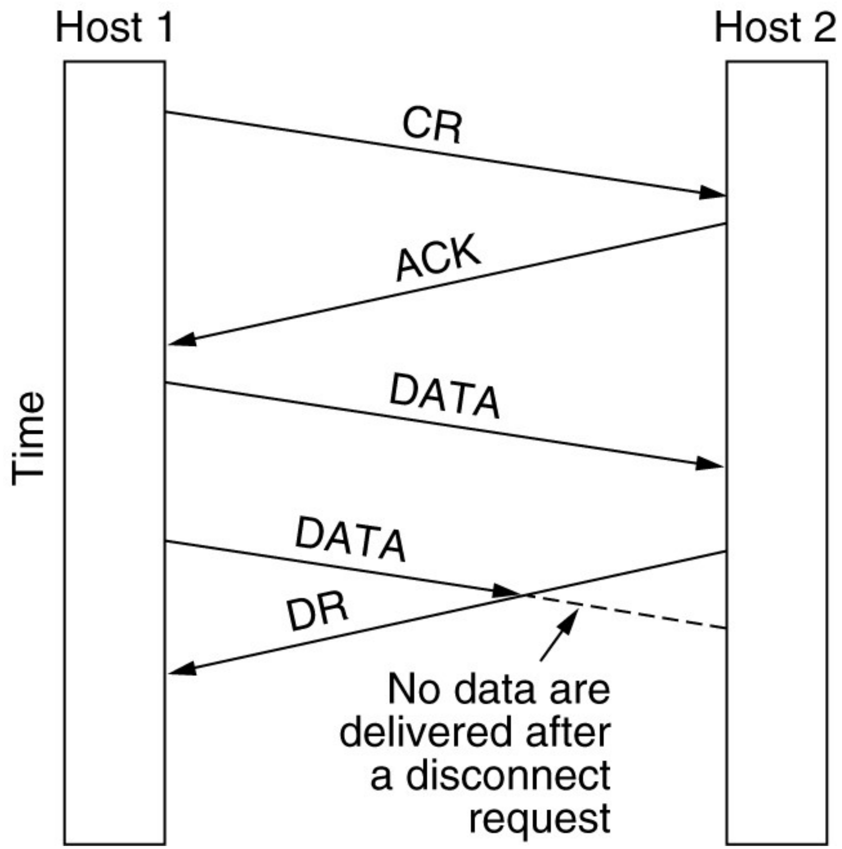
(a) Normal operation,

(b) Old Dup CONNECTION REQUEST appearing out of nowhere.

(c) Duplicate CONNECTION REQUEST and duplicate ACK.

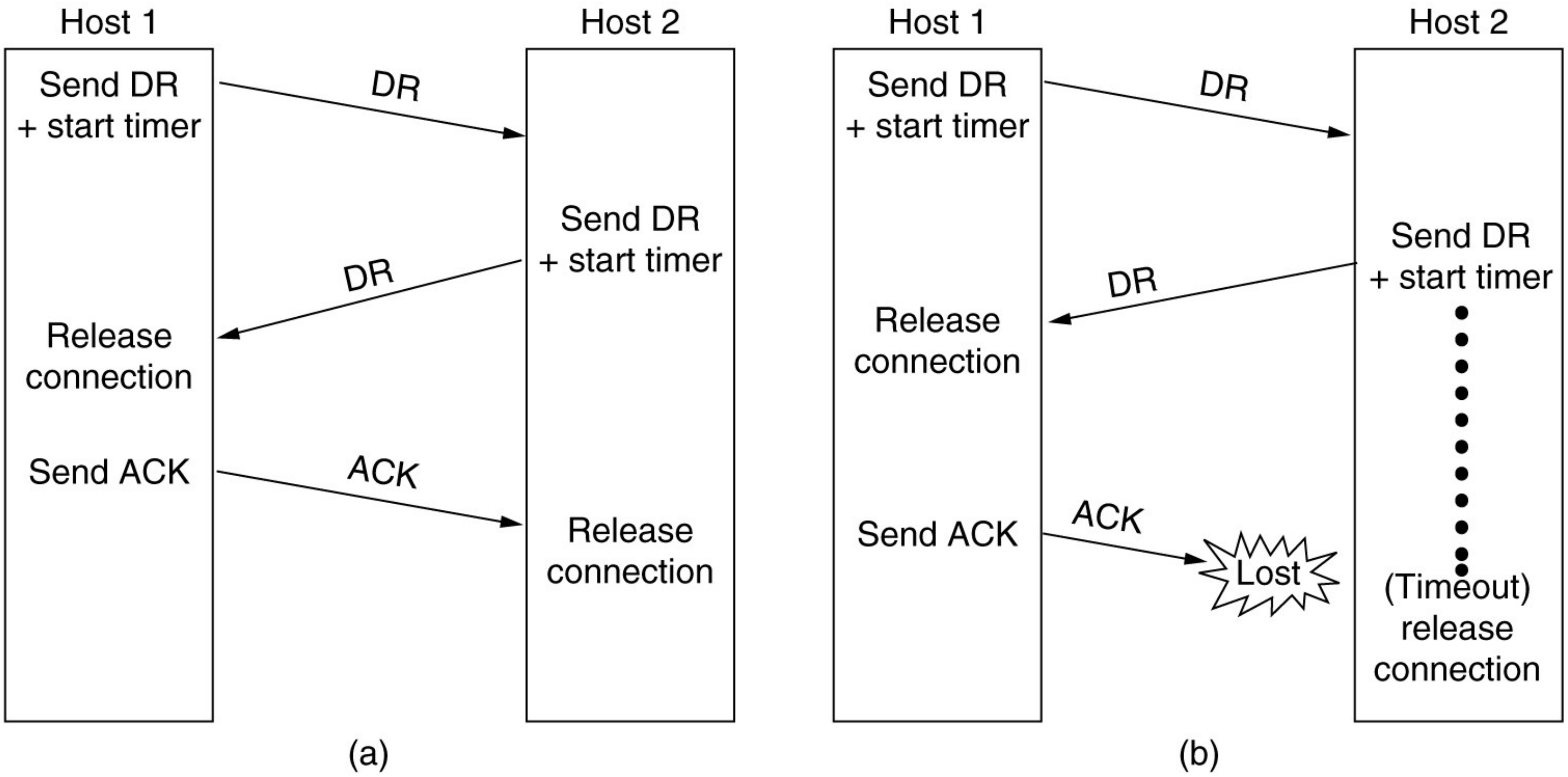
Connection Release

Asymmetric Release



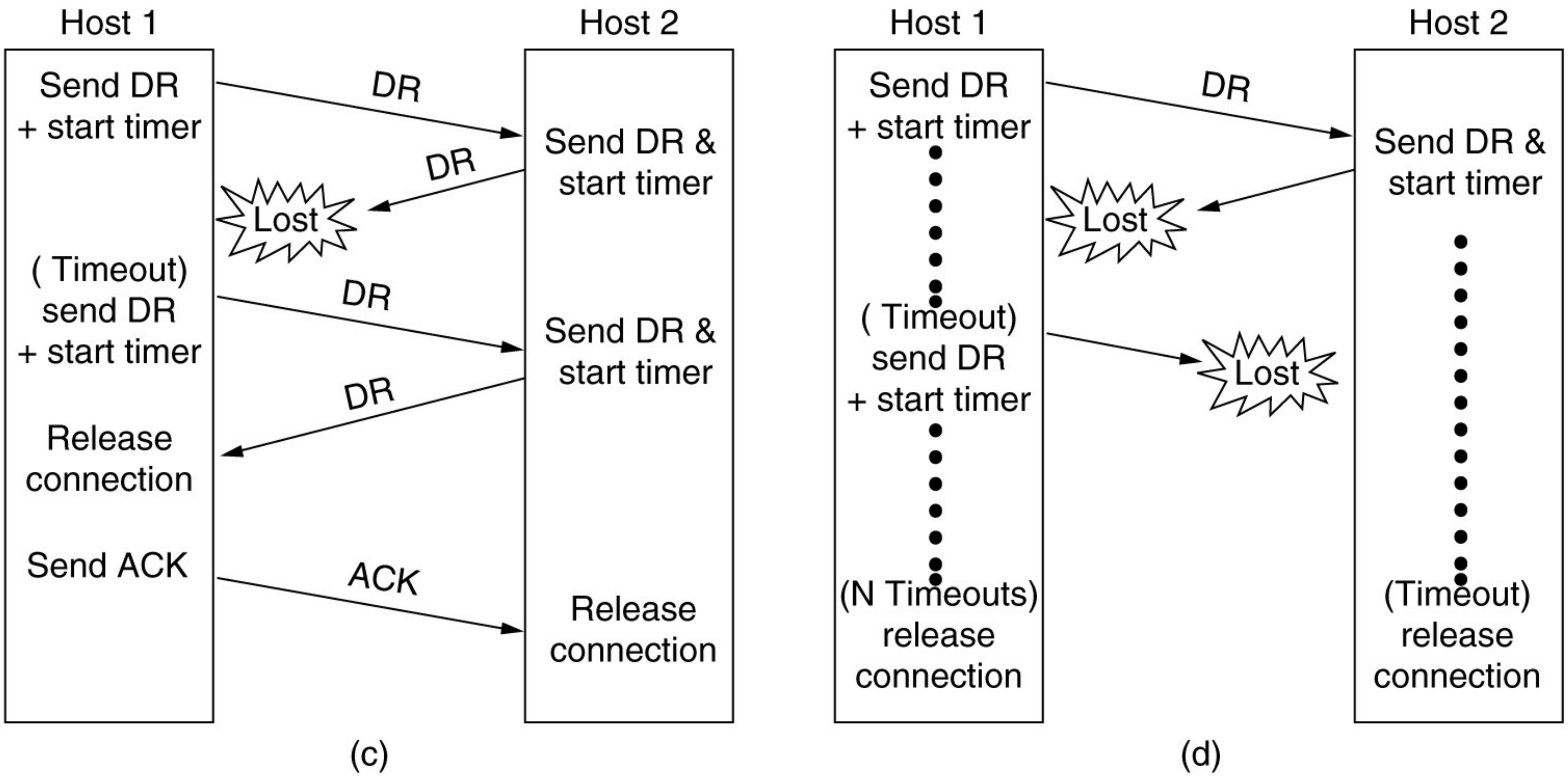
Abrupt disconnection with loss of data.

Connection Release (2)



Four protocol scenarios for releasing a connection. (a) Normal case of a three-way handshake. (b) final ACK lost.

Connection Release (3)



(c) Response lost. (d) Response lost and subsequent DRs lost.

Flow Control and Buffering

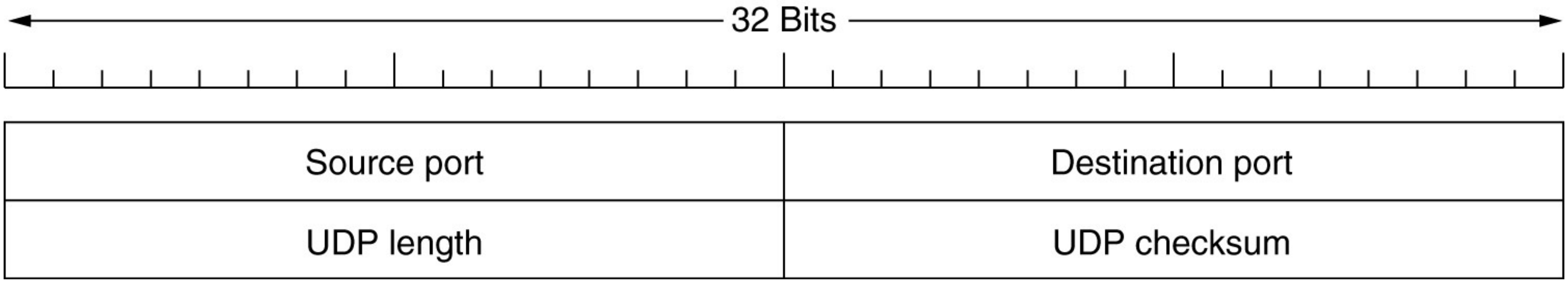
	A	Message	B	Comments
1	→	< request 8 buffers>	→	A wants 8 buffers
2	←	<ack = 15, buf = 4>	←	B grants messages 0-3 only
3	→	<seq = 0, data = m0>	→	A has 3 buffers left now
4	→	<seq = 1, data = m1>	→	A has 2 buffers left now
5	→	<seq = 2, data = m2>	• • •	Message lost but A thinks it has 1 left
6	←	<ack = 1, buf = 3>	←	B acknowledges 0 and 1, permits 2-4
7	→	<seq = 3, data = m3>	→	A has 1 buffer left
8	→	<seq = 4, data = m4>	→	A has 0 buffers left, and must stop
9	→	<seq = 2, data = m2>	→	A times out and retransmits
10	←	<ack = 4, buf = 0>	←	Everything acknowledged, but A still blocked
11	←	<ack = 4, buf = 1>	←	A may now send 5
12	←	<ack = 4, buf = 2>	←	B found a new buffer somewhere
13	→	<seq = 5, data = m5>	→	A has 1 buffer left
14	→	<seq = 6, data = m6>	→	A is now blocked again
15	←	<ack = 6, buf = 0>	←	A is still blocked
16	• • •	<ack = 6, buf = 4>	←	Potential deadlock

Dynamic buffer allocation. The arrows show the direction of transmission. An ellipsis (…) indicates a lost TPDU.

The Internet Transport Protocols: UDP

- Introduction to UDP
- Remote Procedure Call
- The Real-Time Transport Protocol

Introduction to UDP



32 Bits

Source port	Destination port
UDP length	UDP checksum

The UDP header.

The Internet Transport Protocols: TCP

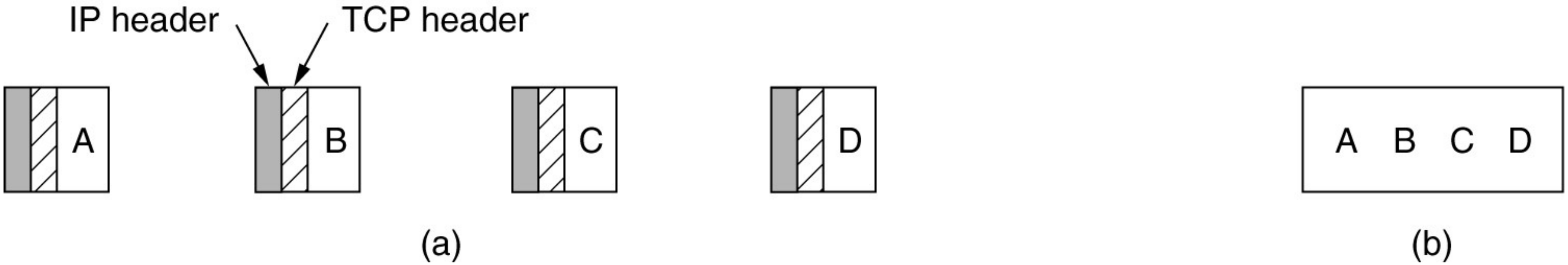
- Introduction to TCP
- The TCP Service Model
- The TCP Protocol
- The TCP Segment Header
- TCP Connection Establishment
- TCP Connection Release
- TCP Connection Management Modeling
- TCP Transmission Policy
- TCP Congestion Control
- TCP Timer Management
- Wireless TCP and UDP
- Transactional TCP

The TCP Service Model

Port	Protocol	Use
21	FTP	File transfer
23	Telnet	Remote login
25	SMTP	E-mail
69	TFTP	Trivial File Transfer Protocol
79	Finger	Lookup info about a user
80	HTTP	World Wide Web
110	POP-3	Remote e-mail access
119	NNTP	USENET news

Some assigned ports.

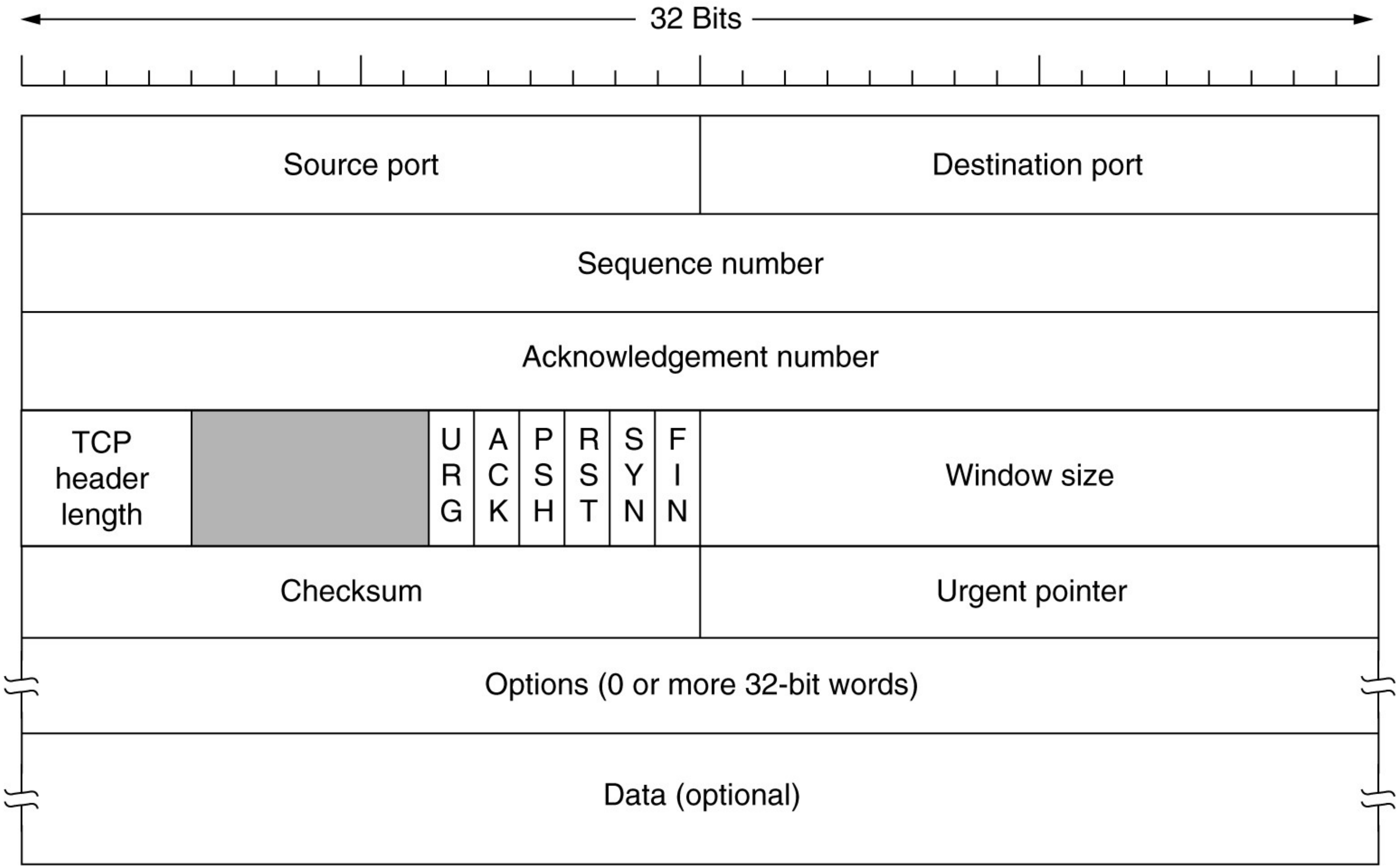
The TCP Service Model (2)



(a) Four 512-byte segments sent as separate IP datagrams.

(b) The 2048 bytes of data delivered to the application in a single READ CALL.

The TCP Segment Header



32 Bits

Source port

Destination port

Sequence number

Acknowledgement number

TCP header length

URG

ACK

PSH

RST

SYN

FIN

Window size

Checksum

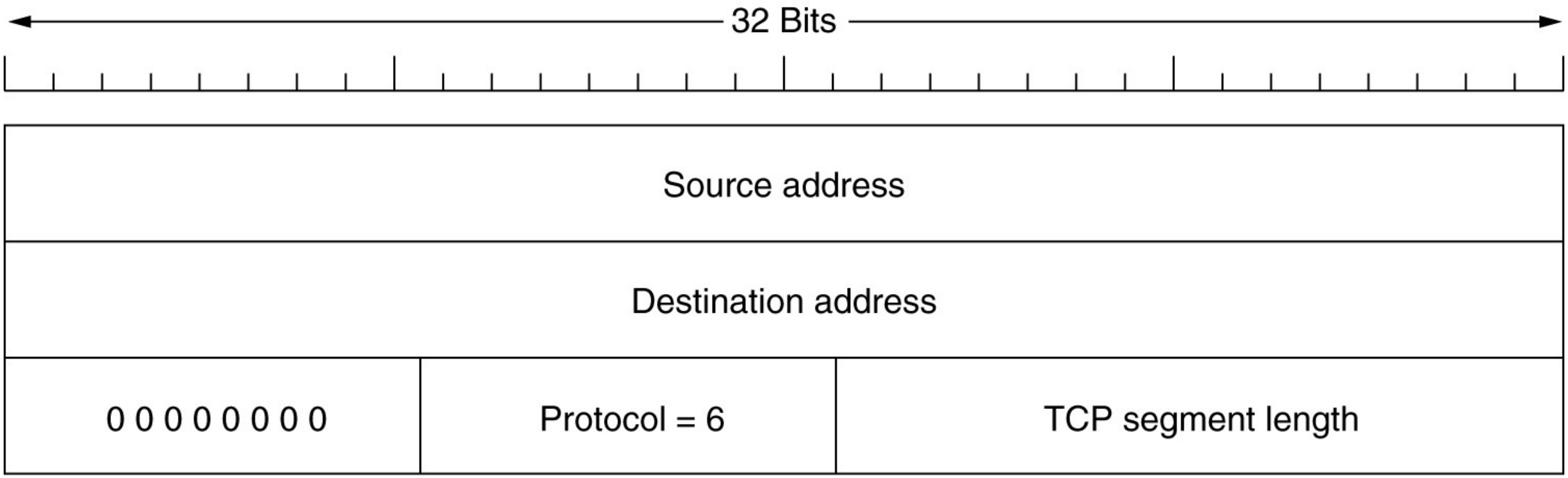
Urgent pointer

Options (0 or more 32-bit words)

Data (optional)

TCP Header.

The TCP Segment Header (2)



32 Bits

Source address

Destination address

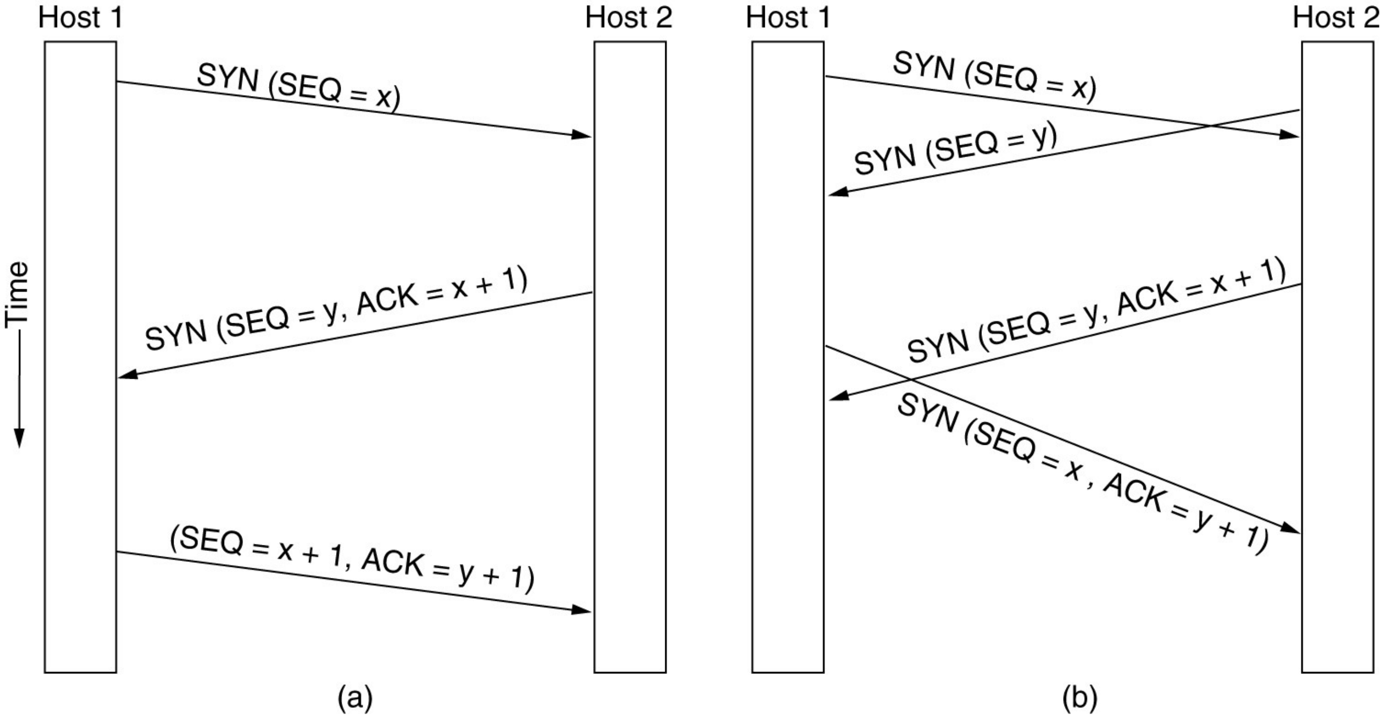
0 0 0 0 0 0 0 0

Protocol = 6

TCP segment length

The pseudoheader included in the TCP checksum.

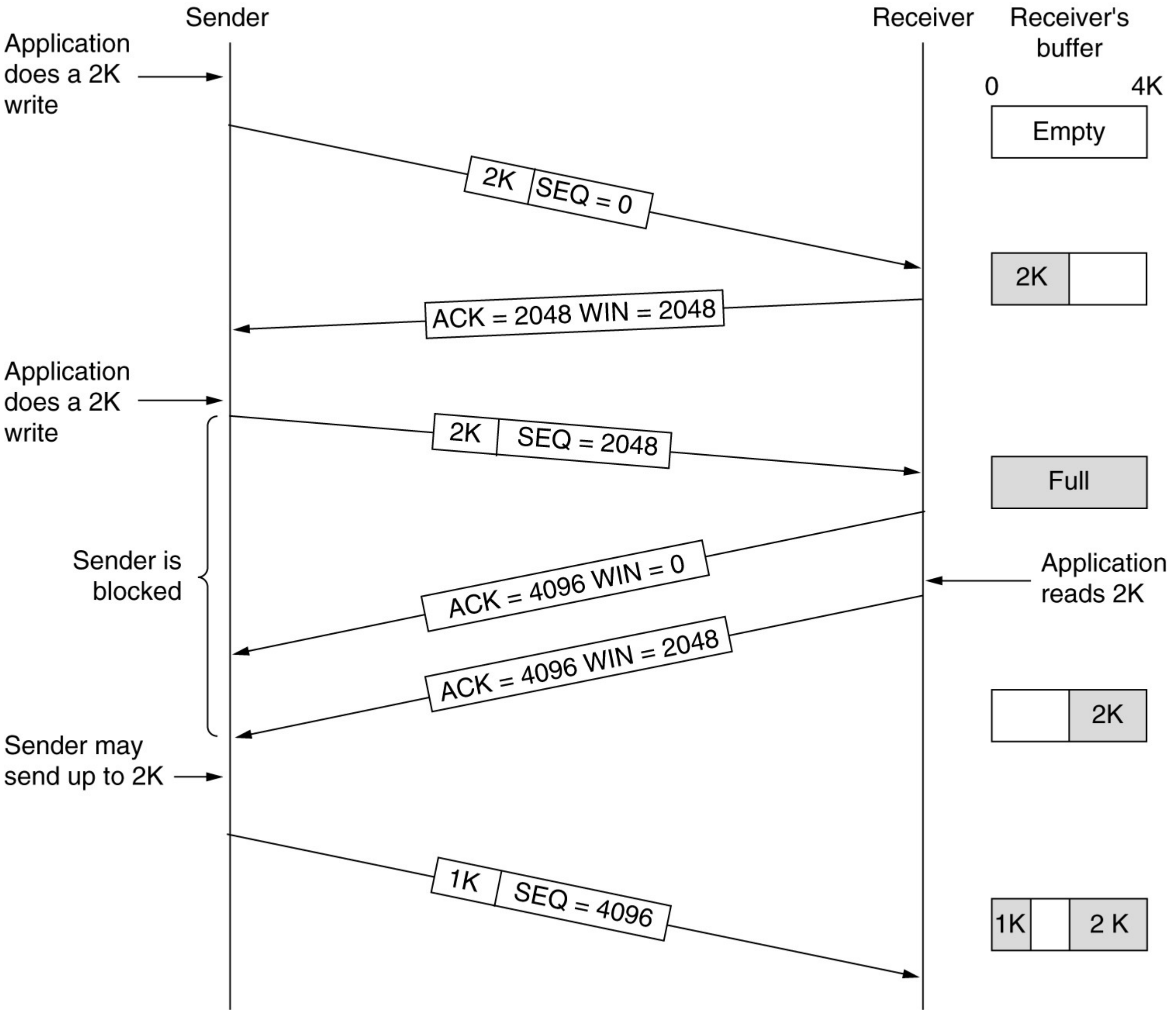
TCP Connection Establishment


Host 1
Host 2
SYN (SEQ = x)
SYN (SEQ = y, ACK = x + 1)
(SEQ = x + 1, ACK = y + 1)
Time
(a)
Host 1
Host 2
SYN (SEQ = x)
SYN (SEQ = y)
SYN (SEQ = y, ACK = x + 1)
SYN (SEQ = x , ACK = y + 1)
(b)

(a) TCP connection establishment in the normal case.

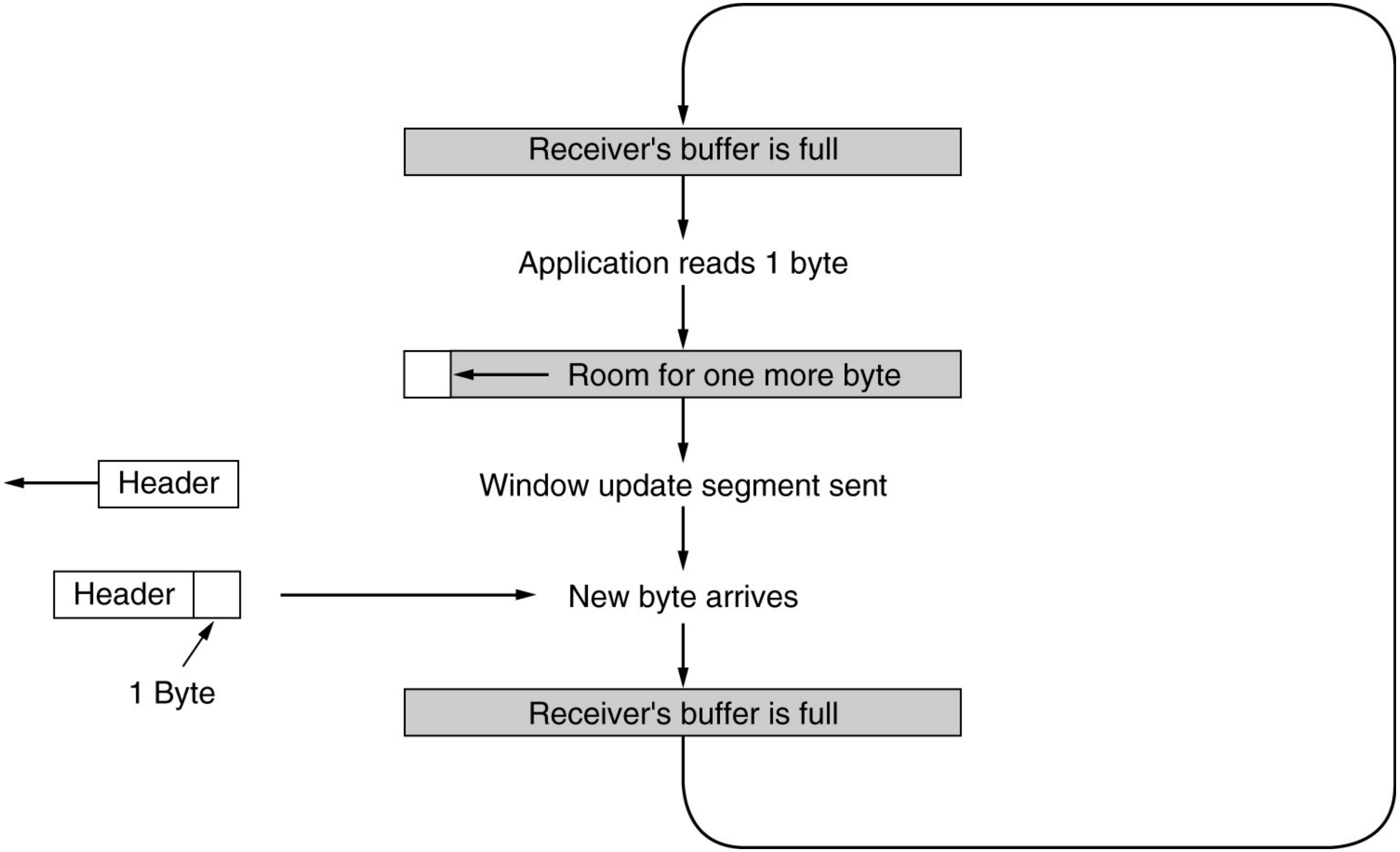
(b) Call collision.

TCP Transmission Policy



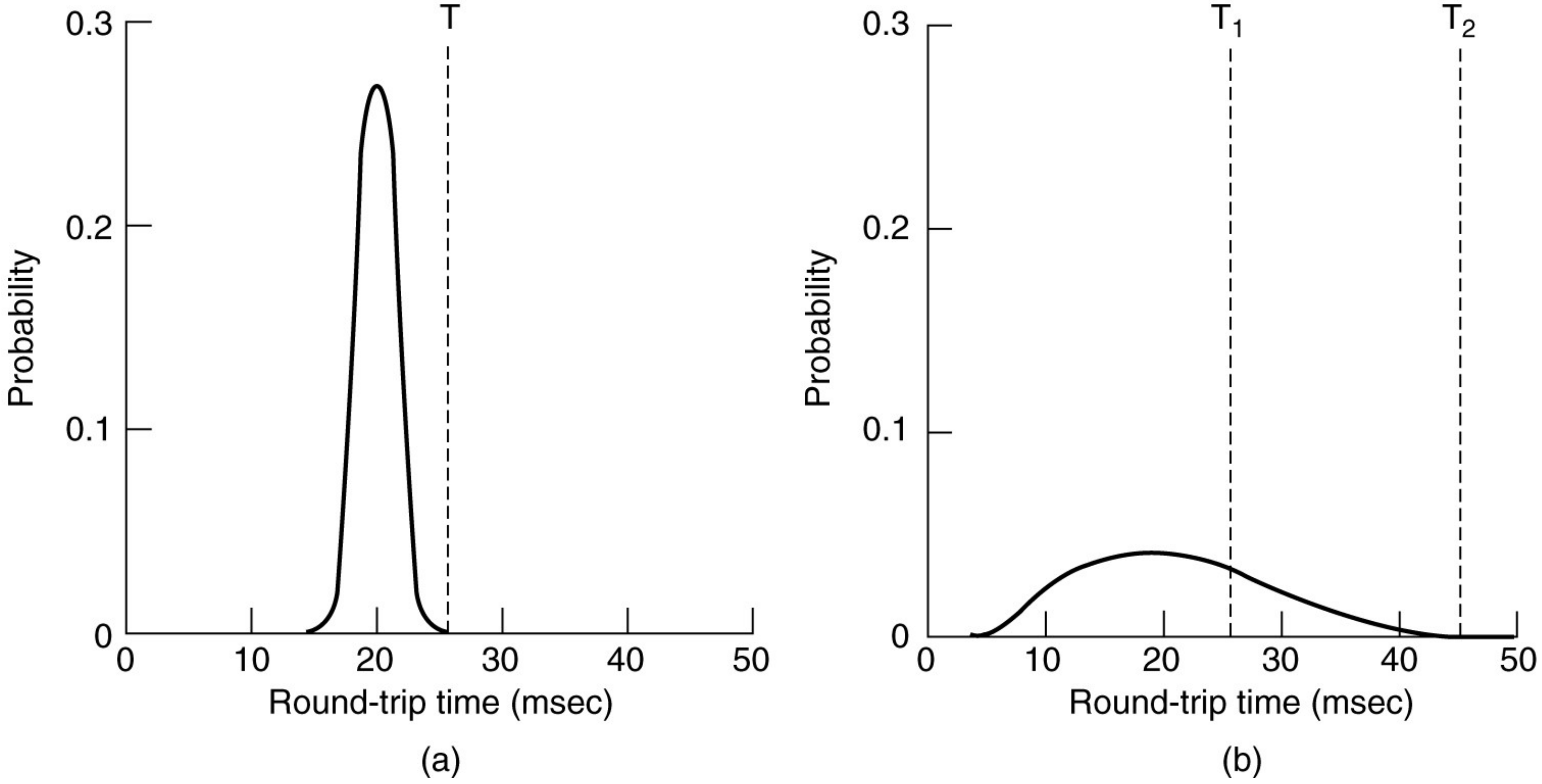
Window management in TCP.

TCP Transmission Policy (2)



Silly window syndrome.

TCP Timer Management



(a) Probability density of ACK arrival times in the data Link layer.

(b) Probability density of ACK arrival times for TCP.