



Transport Over IP (II)

รศ.ดร. อนันต์ พลเพิ่ม

Asso. Prof. Anan Phonphoem, Ph.D.

anan.p@ku.ac.th

<http://www.cpe.ku.ac.th/~anan>

Computer Engineering Department

Kasetsart University, Bangkok, Thailand



Outline



Transport Layer



User Datagram Protocol (UDP)

- **Transmission Control Protocol (TCP)**
- Stream Control Transmission Protocol (SCTP)
- Real-Time Transport Protocol (RTP)

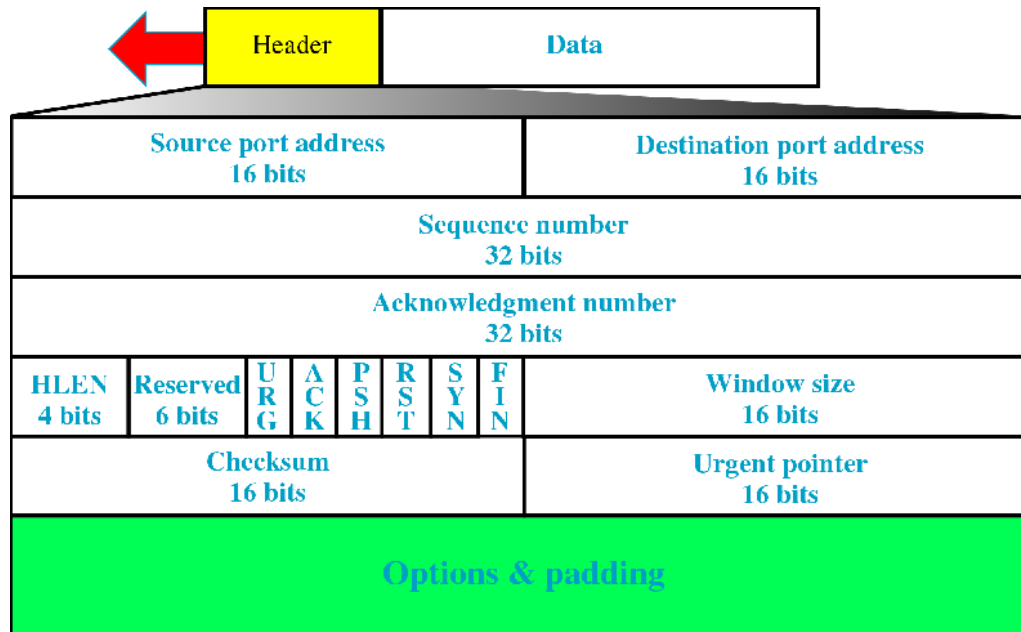


Transmission Control Protocol (TCP)

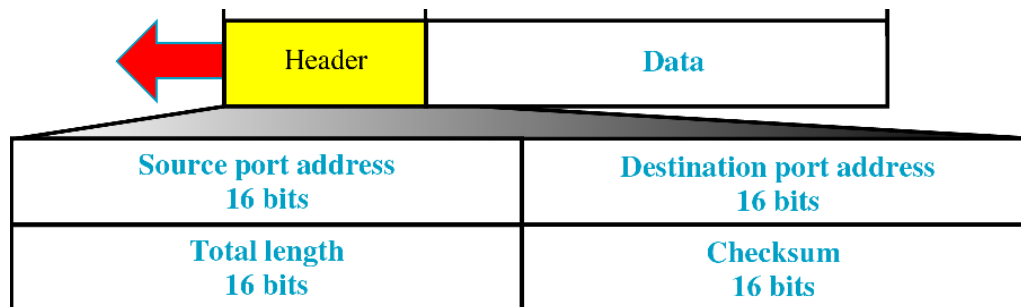
- Most popular/important in Internet
- Reliable transport protocol
- Connection-oriented
 - need establishment
 - [IP+source port] ↔ [IP+destination port]
 - guarantee delivery / error
- Checksum is mandatory

TCP Messages (Segment)

TCP Header



UDP Header

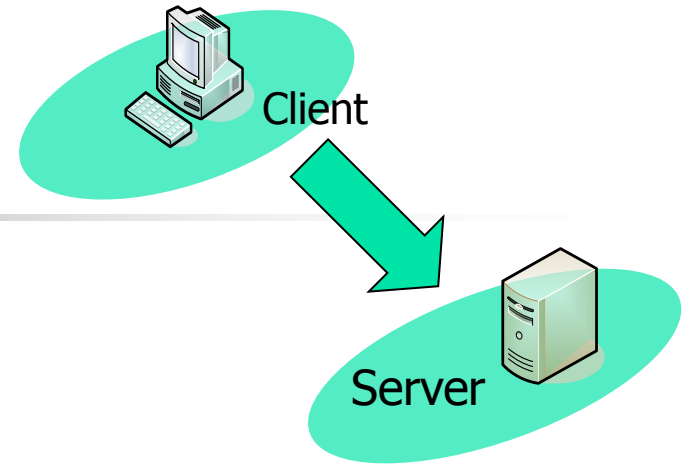




Connection Establishment

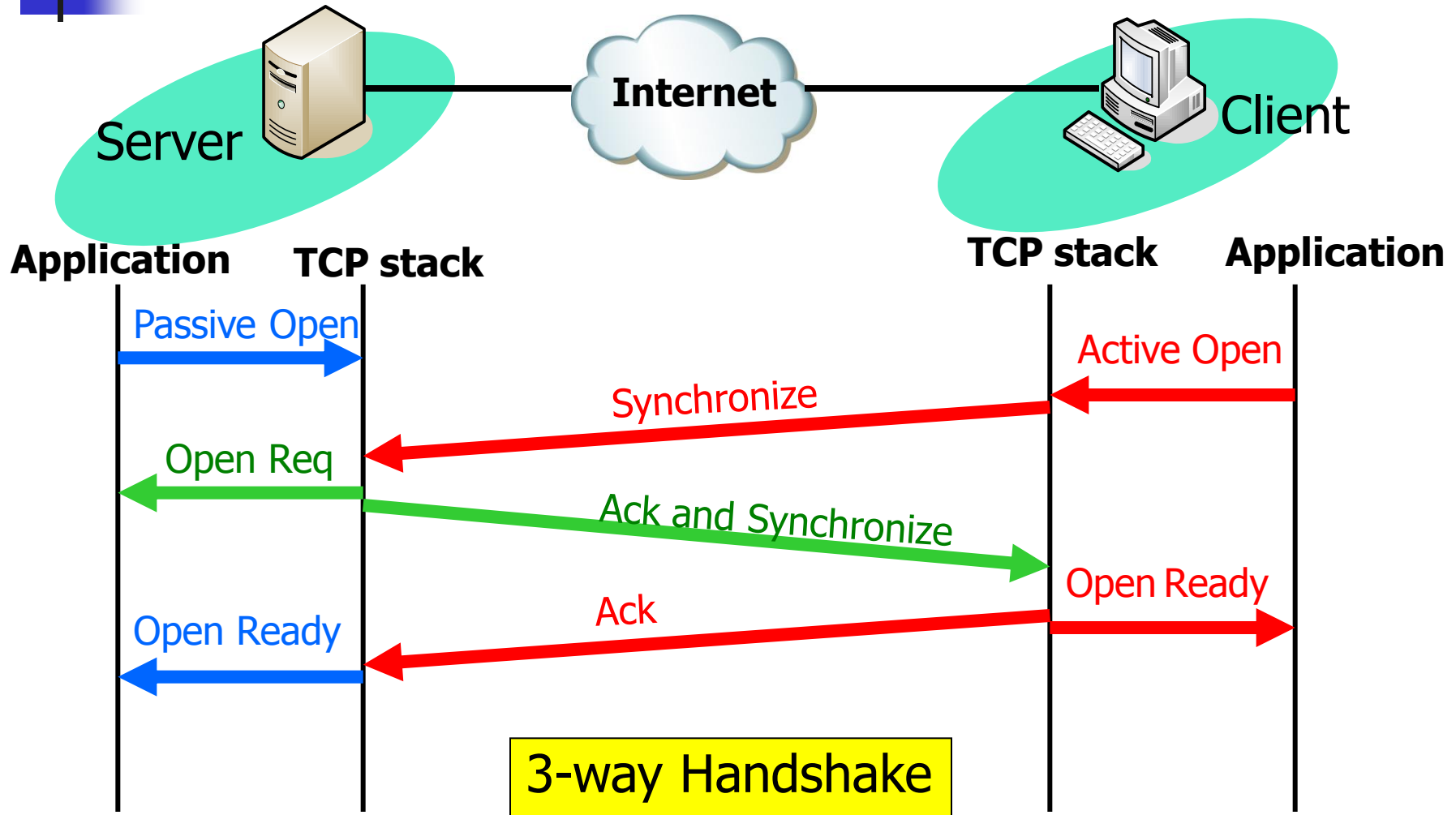
- Application requests connection through **socket API**
 - not part of TCP
 - method to access services provided by TCP
- To start connection establishment
 - receiver must be listening

Making connection



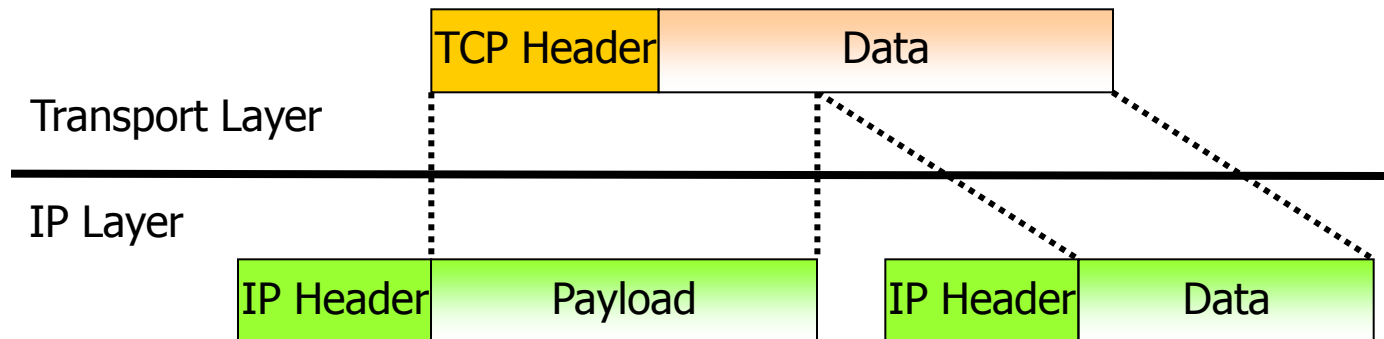
- Application on server
 - passive receive mode
 - listening by issue **Listen request** to socket **locally**
- Application on Client
 - request to socket API **locally** with
 - destination IP
 - destination port
 - (source port: if not specify, TCP assigns private port)

TCP Connection Establishment

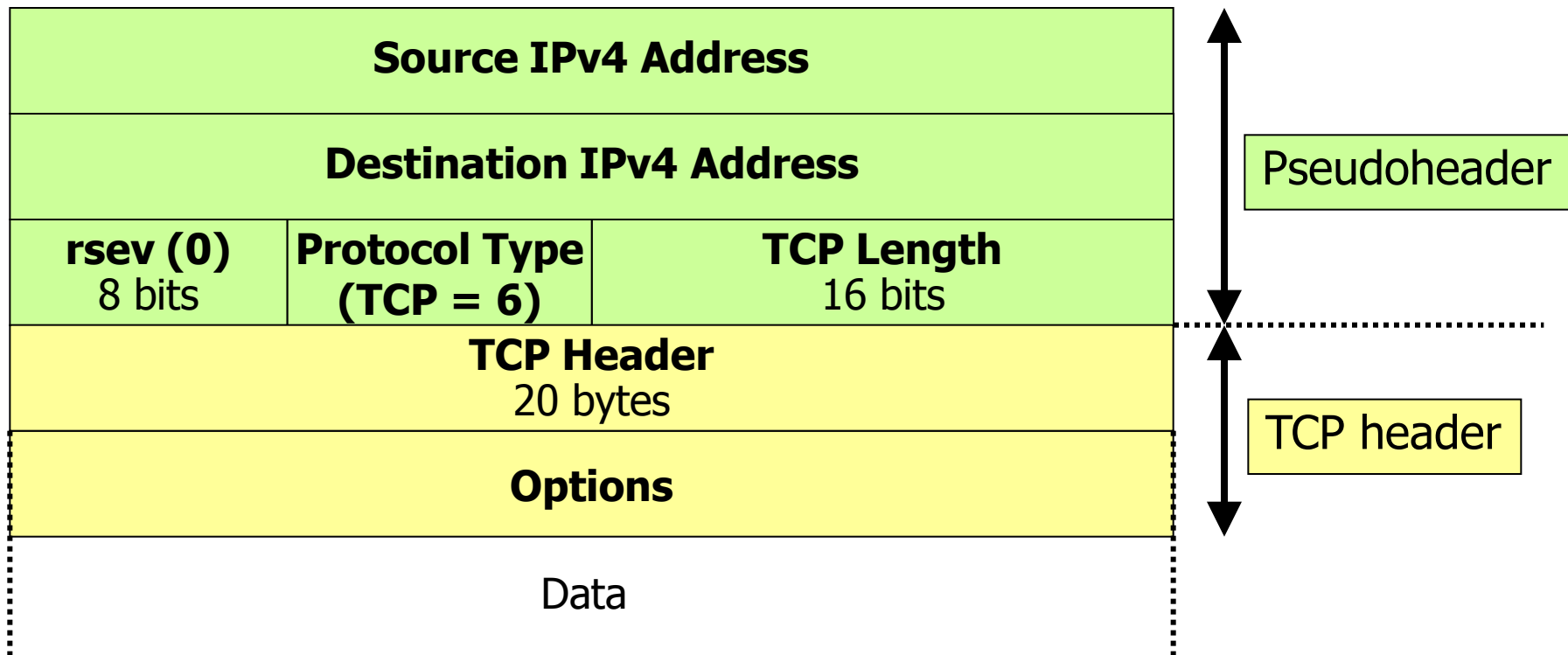


TCP Segment

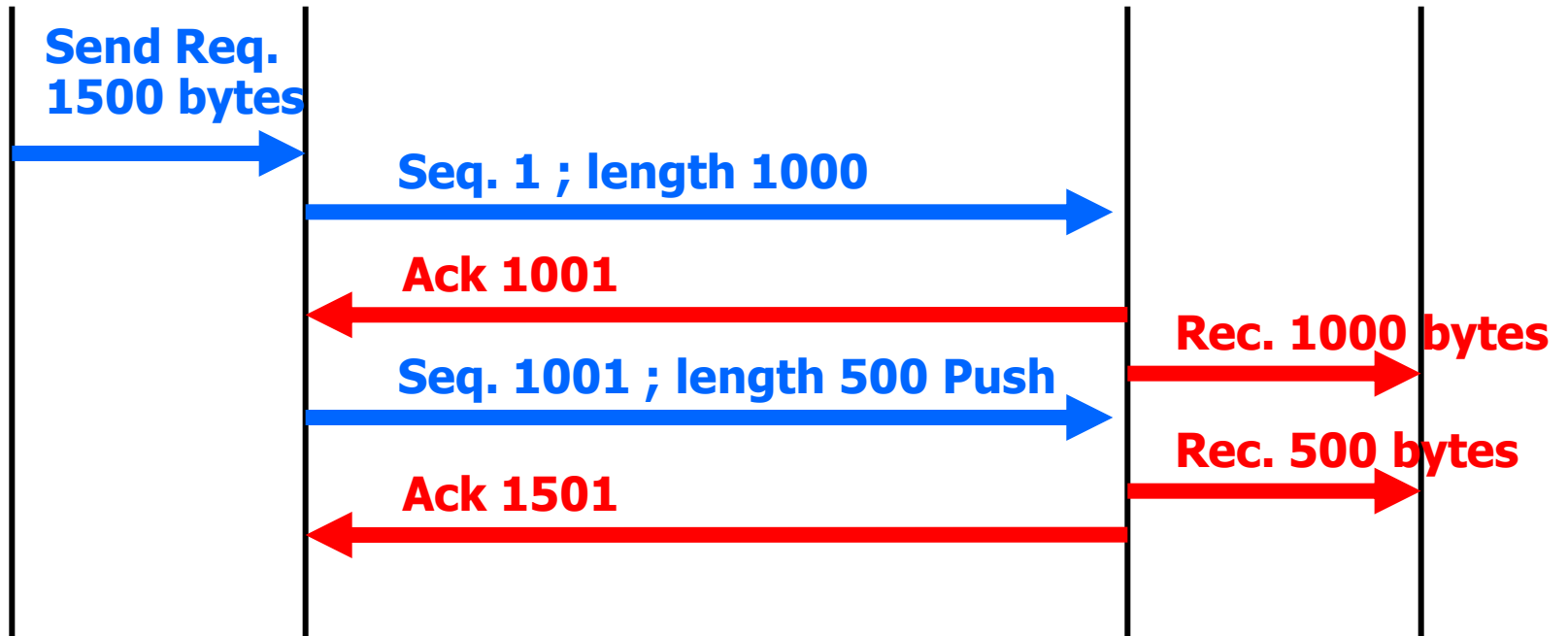
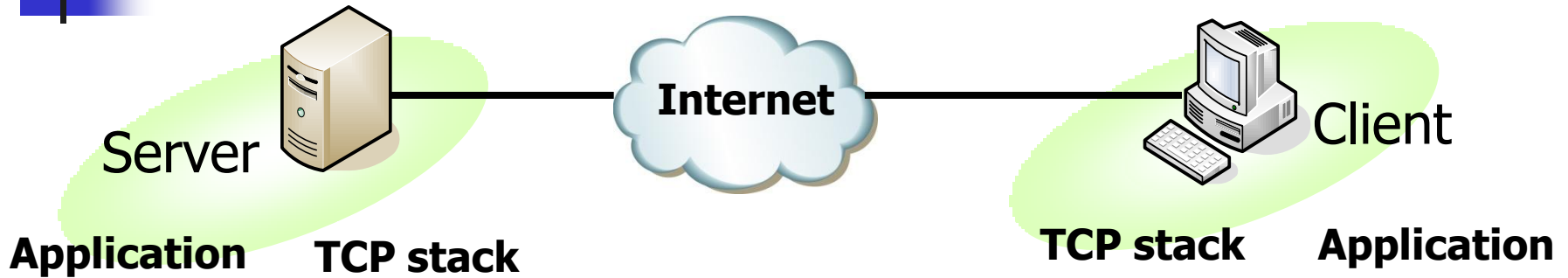
- Segment size
 - at source node = MTU local link
 - fit in IP packet
 - may be fragment along the way



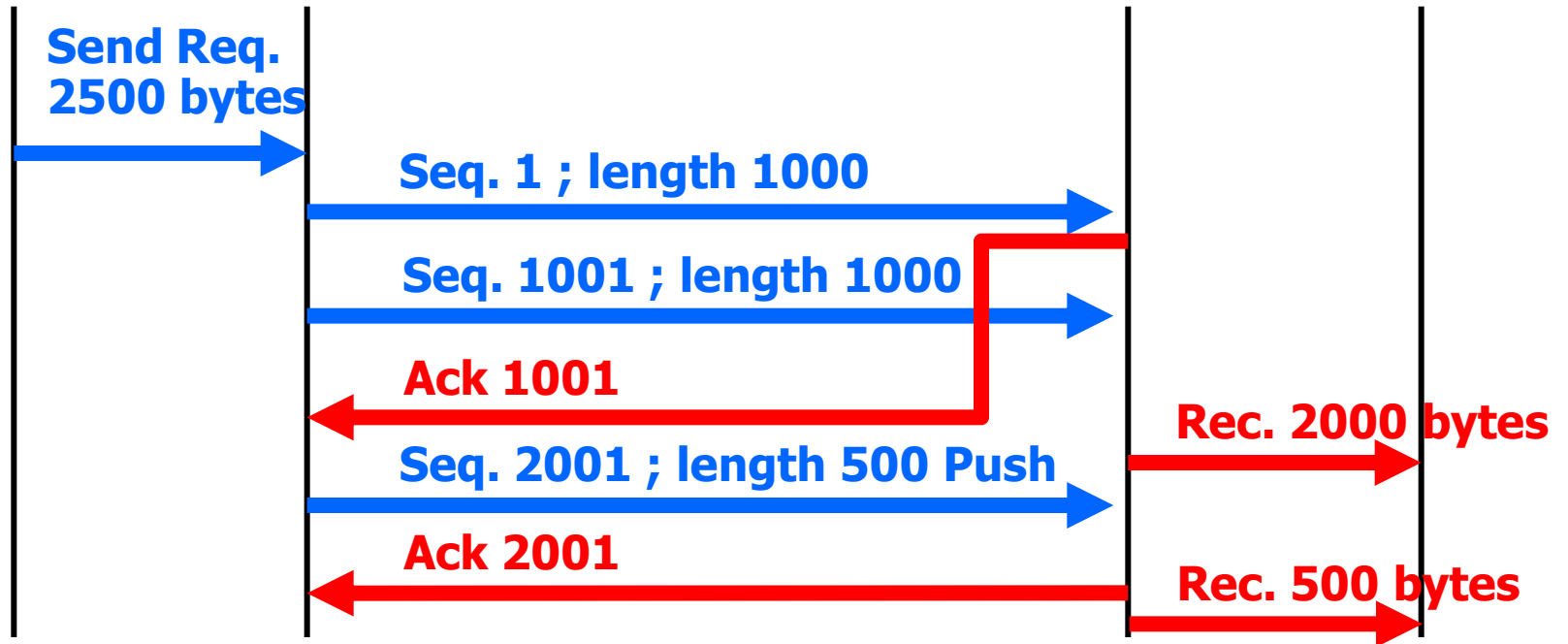
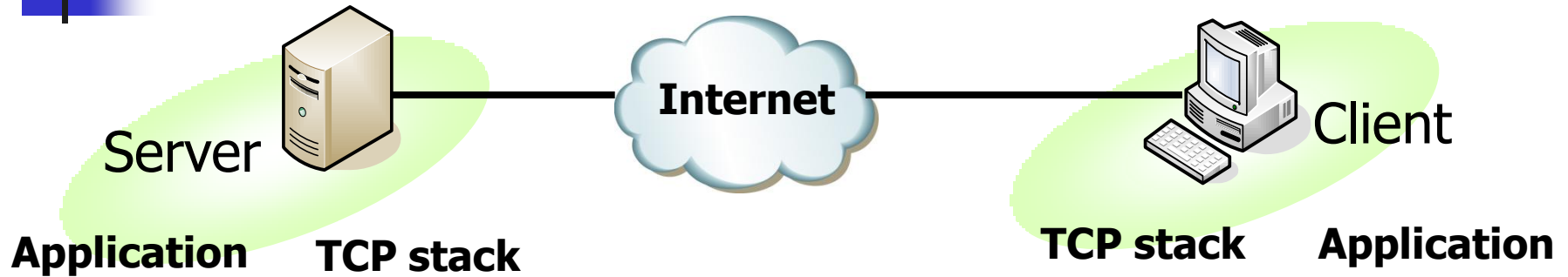
TCP Checksum



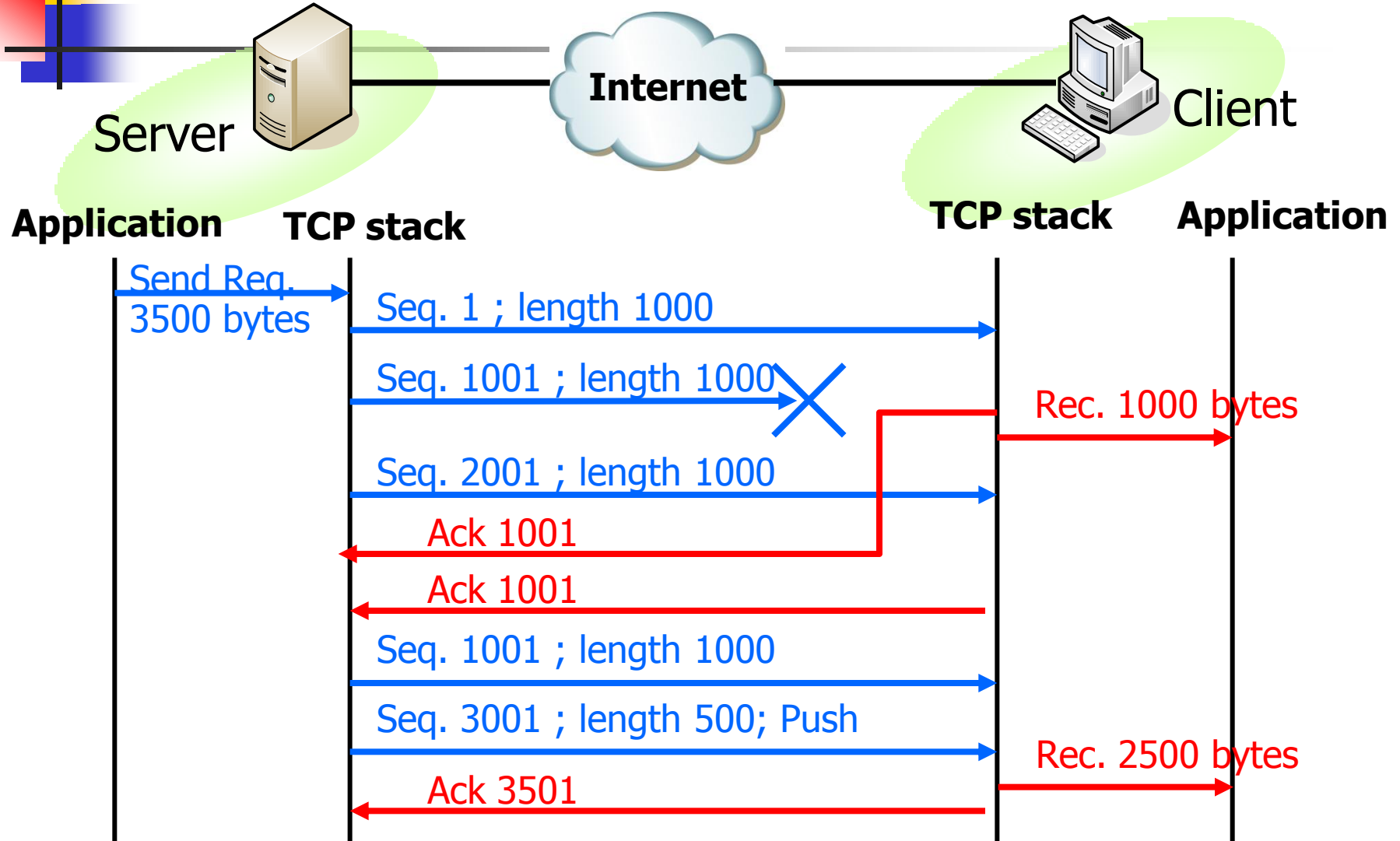
Data Transmission (I)



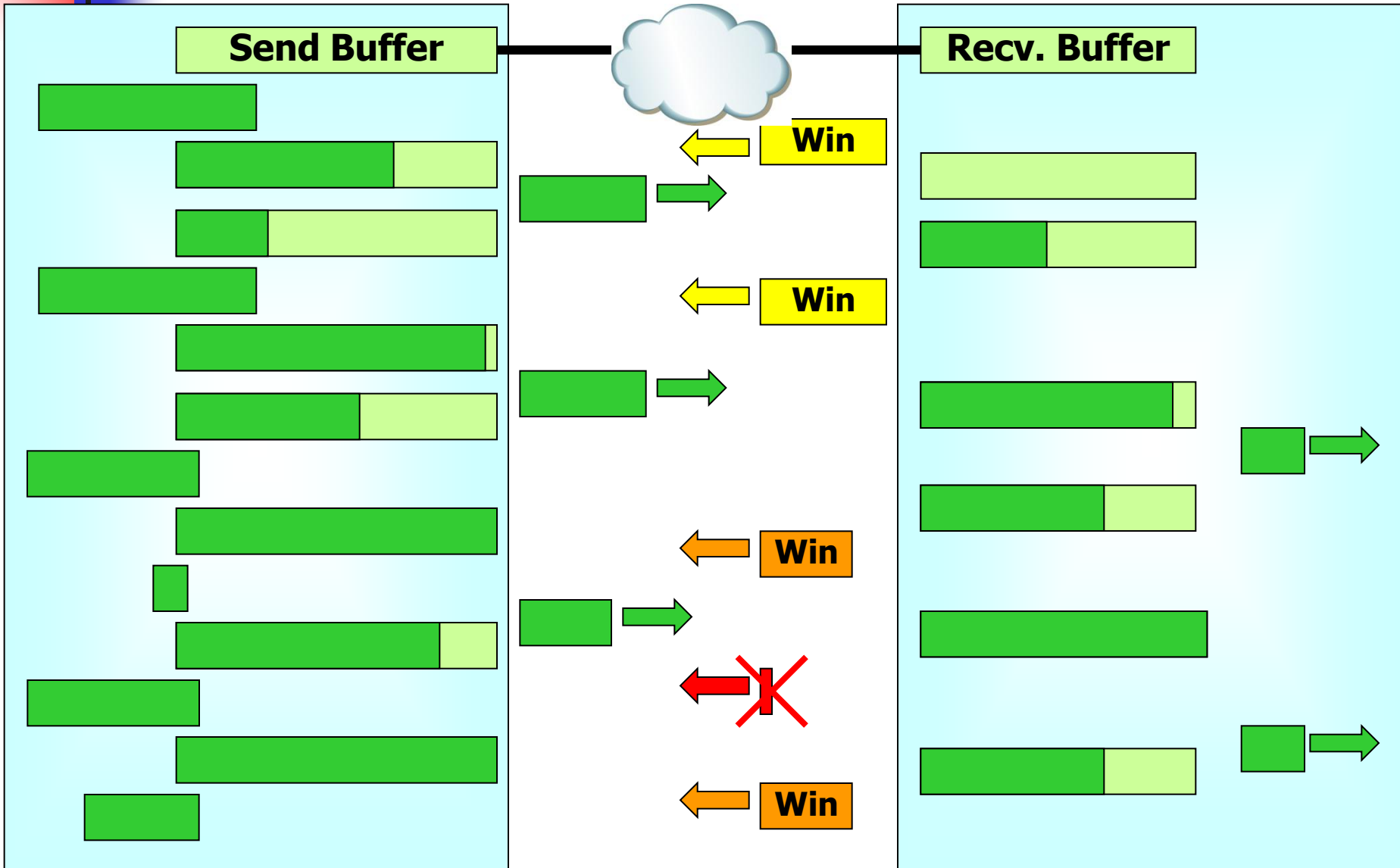
Data Transmission (II)



Data Transmission (III)

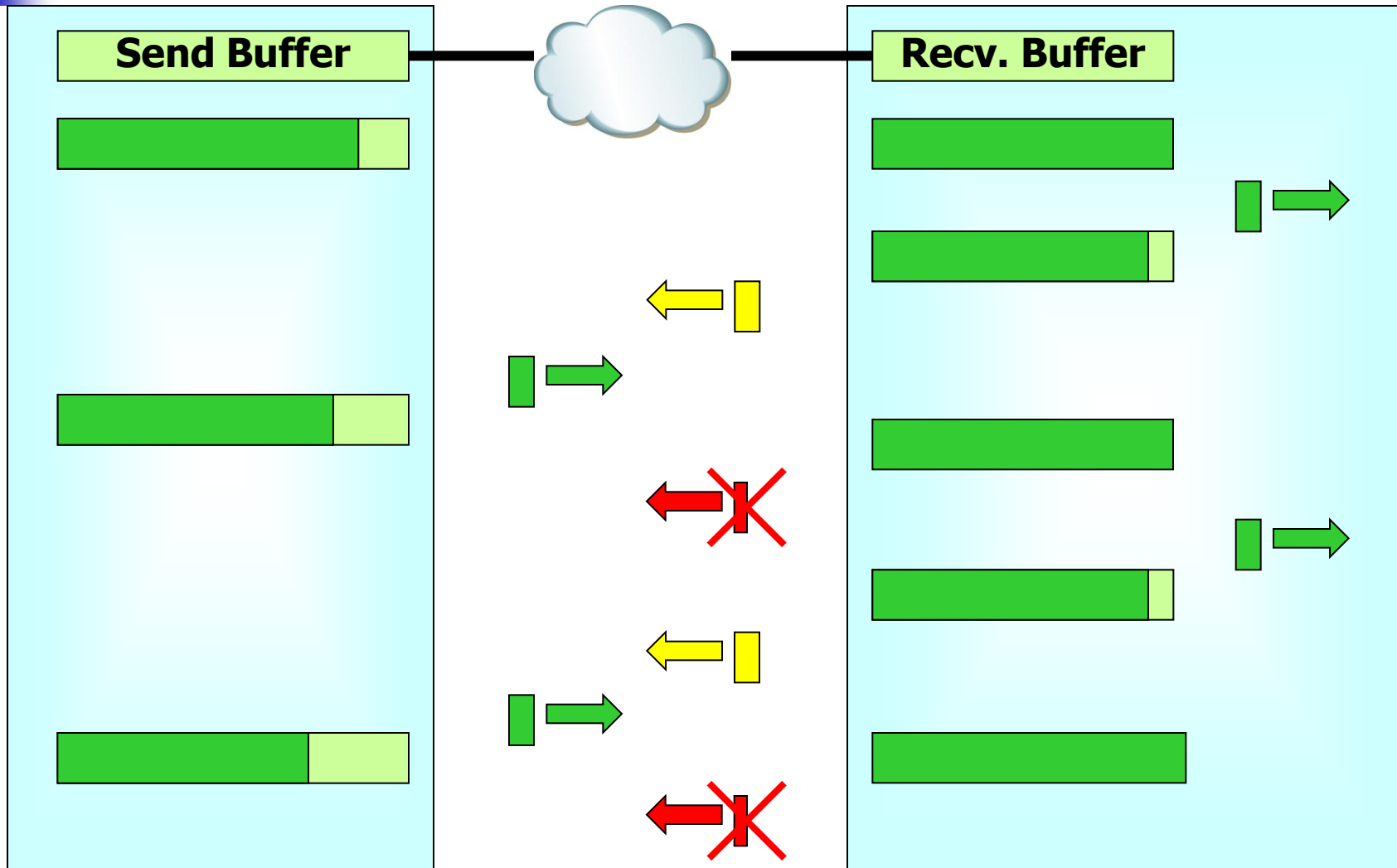


TCP Window Control



Poor TCP Window Control

"Silly Window Syndrome"

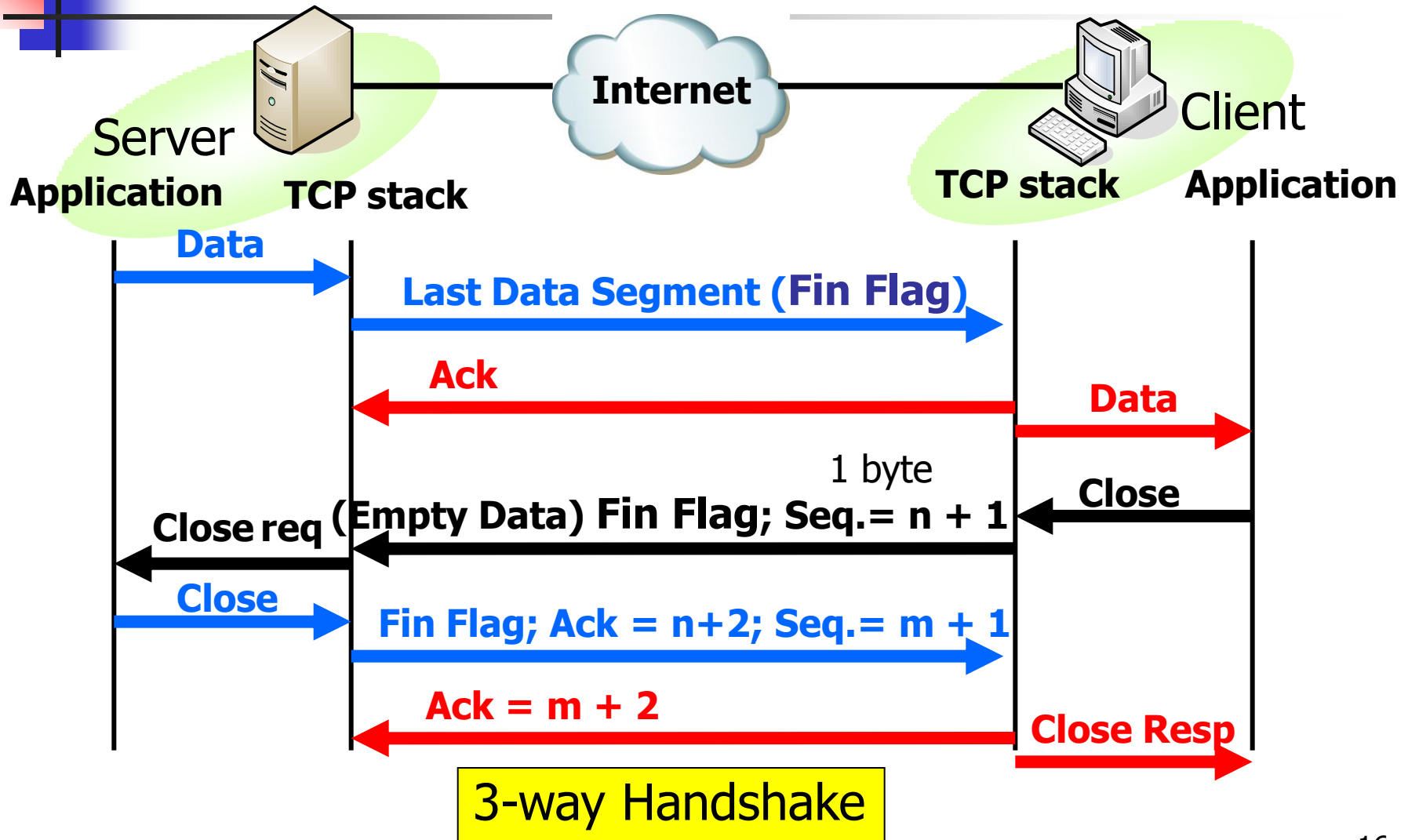


Urgent Data

- Set Urgent bit in the flag field
- Overtake any byte in queue (e.g. Esc)

| | | | | | | | | |
|----------------------------------|--------------------|----------------------------------|----------------------------------|-------------------------------------|----------------------------------|----------------------------------|----------------------------------|------------------------|
| Source port address 16 bits | | | | Destination port address 16 bits | | | | |
| Sequence number 32 bits | | | | | | | | |
| Acknowledgment number 32 bits | | | | | | | | |
| HLEN 4 bits | Reserved 6 bits | U R G | A C K | P S H | R S T | S Y N | F I N | Window size 16 bits |
| Checksum 16 bits | | | | Urgent pointer 16 bits | | | | |

Closing Connection (By either end)



Example: 3-way Handshake

| No. . | Time | Source | Destination | Protocol | Info |
|-------|-----------|---------------|---------------|----------|---|
| 69 | 12.203795 | 172.16.2.103 | 158.108.10.10 | TCP | nfa > http [SYN] Seq=0 win=16384 Len=0 MSS=1460 |
| 70 | 12.205400 | 158.108.10.10 | 172.16.2.103 | TCP | http > nfa [SYN, ACK] Seq=0 Ack=1 win=4128 Len= |
| 71 | 12.205594 | 172.16.2.103 | 158.108.10.10 | TCP | nfa > http [ACK] Seq=1 Ack=1 win=17520 Len=0 |
| 72 | 12.206442 | 172.16.2.103 | 158.108.10.10 | HTTP | GET /archive/flash:home/html/home_aux.shtml HTT |
| 73 | 12.292946 | 158.108.10.10 | 172.16.2.103 | TCP | [TCP segment of a reassembled PDU] |
| 78 | 12.417406 | 172.16.2.103 | 158.108.10.10 | TCP | nfa > http [ACK] Seq=679 Ack=257 win=17264 Len= |
| 79 | 12.436025 | 158.108.10.10 | 172.16.2.103 | TCP | [TCP segment of a reassembled PDU] |
| 80 | 12.439493 | 158.108.10.10 | 172.16.2.103 | HTTP | HTTP/1.1 200 OK (text/html) |
| 81 | 12.439824 | 172.16.2.103 | 158.108.10.10 | TCP | nfa > http [ACK] Seq=679 Ack=773 win=16749 Len= |
| 82 | 12.449322 | 172.16.2.103 | 158.108.10.10 | TCP | nfa > http [FIN, ACK] Seq=679 Ack=773 win=16749 |
| 83 | 12.450114 | 158.108.10.10 | 172.16.2.103 | TCP | http > nfa [ACK] seq=773 Ack=680 win=3450 Len=0 |

⊕ Frame 69 (62 bytes on wire, 62 bytes captured)

⊕ Ethernet II, Src: Dell_3a:93:38 (00:12:3f:3a:93:38), Dst: 08:1f:f3:a6:66:cc (08:1f:f3:a6:66:cc)

⊕ Internet Protocol, Src: 172.16.2.103 (172.16.2.103), Dst: 158.108.10.10 (158.108.10.10)

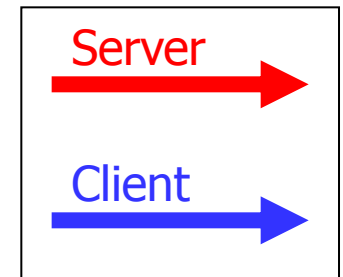
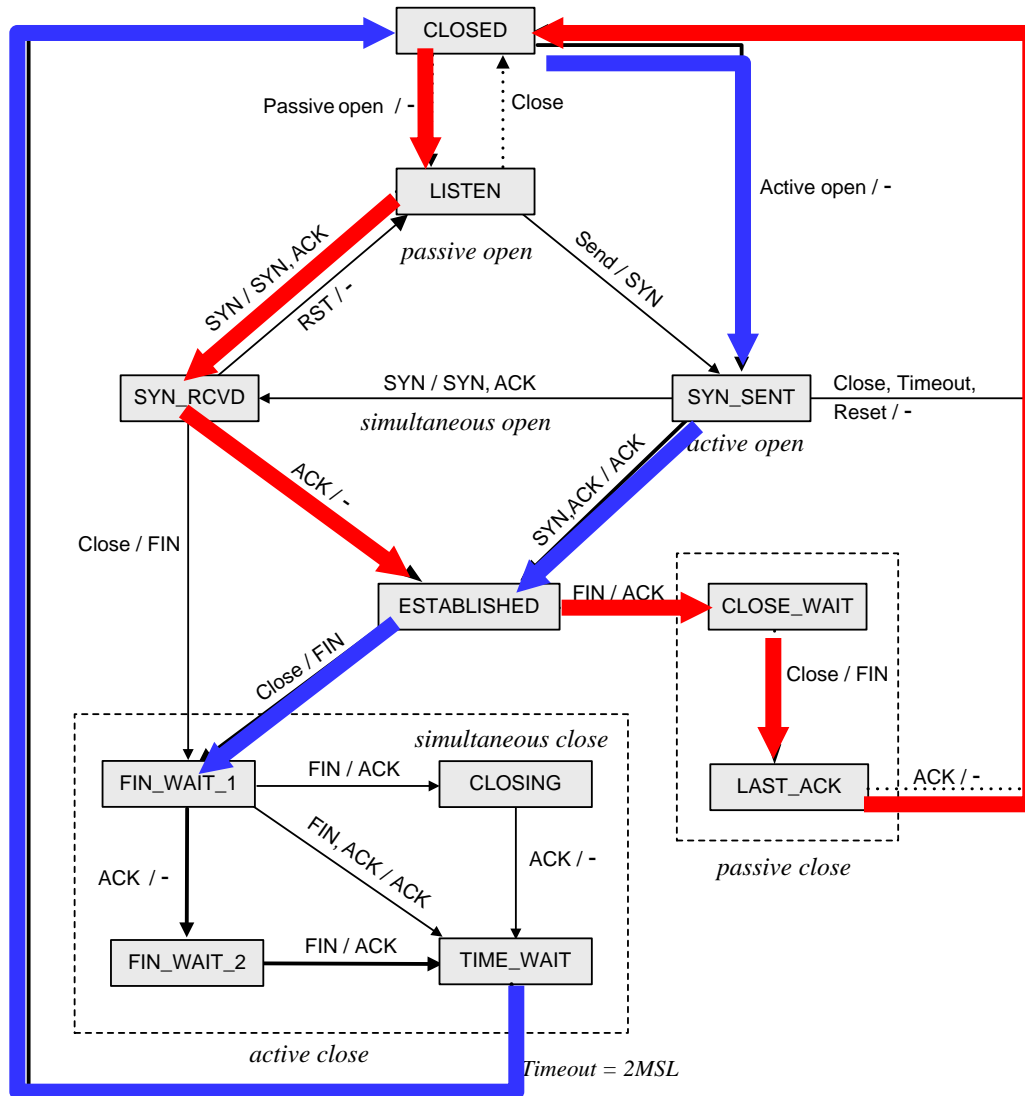
⊕ Transmission Control Protocol, Src Port: nfa (1155), Dst Port: http (80), Seq: 0, Len: 0

```

0000  08 1f f3 a6 66 cc 00 12  3f 3a 93 38 08 00 45 00  ....f... ?:.8..E.
0010  00 30 94 d9 40 00 80 06  0f 01 ac 10 02 67 9e 6c  .0..@... .....g.l
0020  0a 0a 04 83 00 50 ed aa  d6 15 00 00 00 00 70 02  .....P.. .....p.
0030  40 00 23 9e 00 00 02 04  05 b4 01 01 04 02        @.#.....

```

TCP State Machine





Errors in TCP

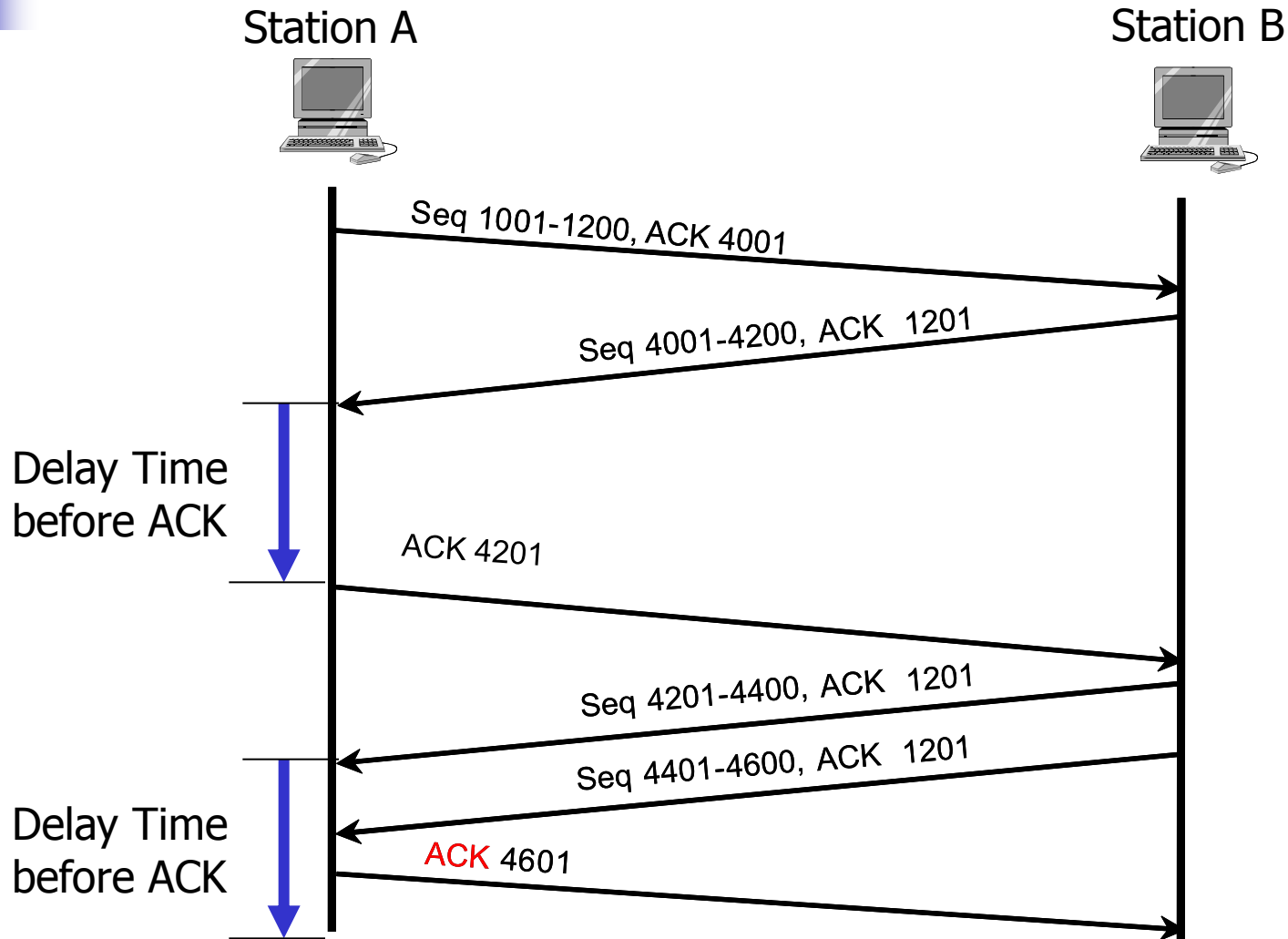
- Corrupted segment
- Wrong sequence segment
- Lost and redundant segment



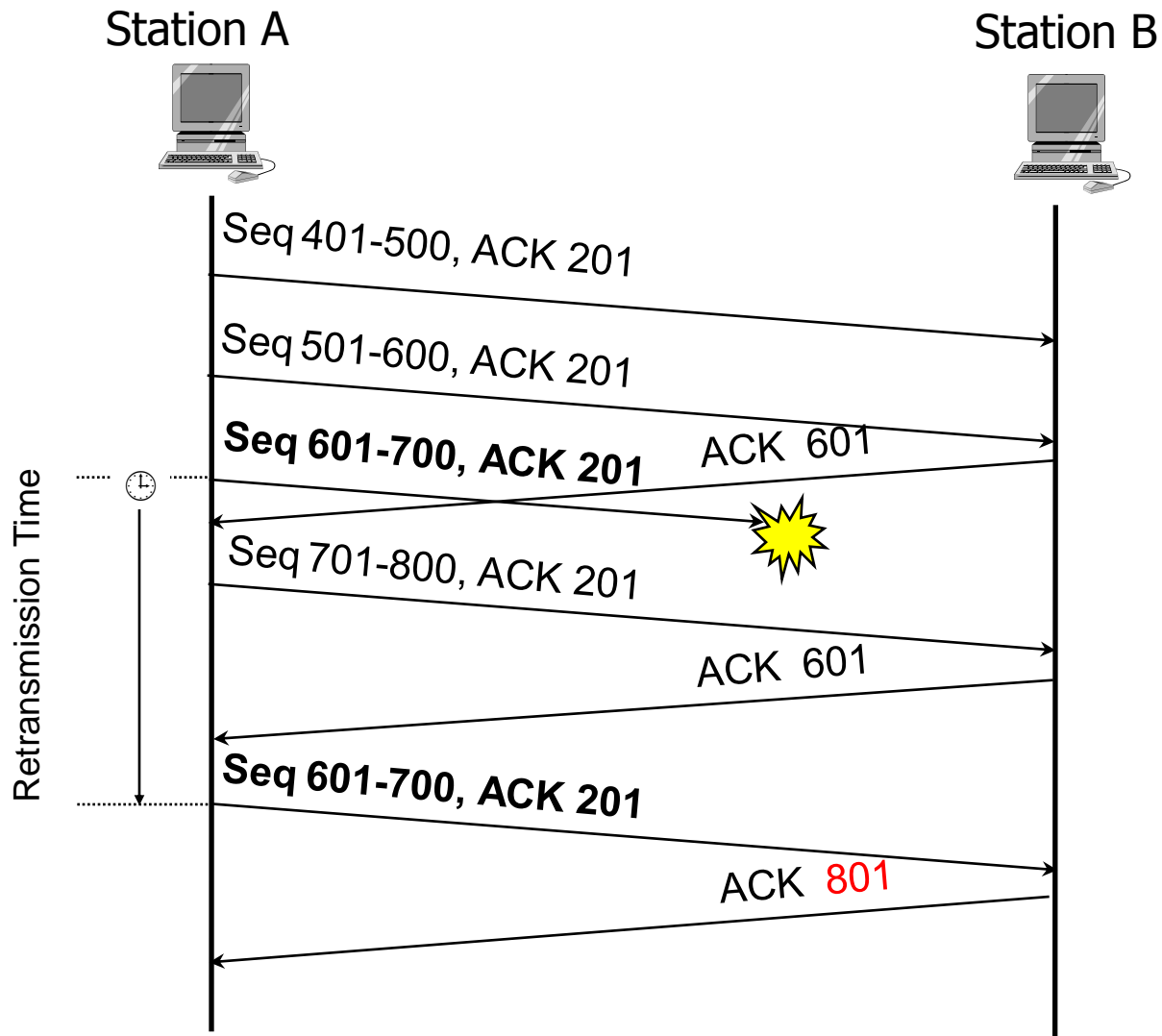
Error Control in TCP

- Checksum
- Acknowledgement
- Retransmission
 - Retransmission Time
 - 3-duplicate ACKs
 - Immediately ACK (Fast Retransmission)

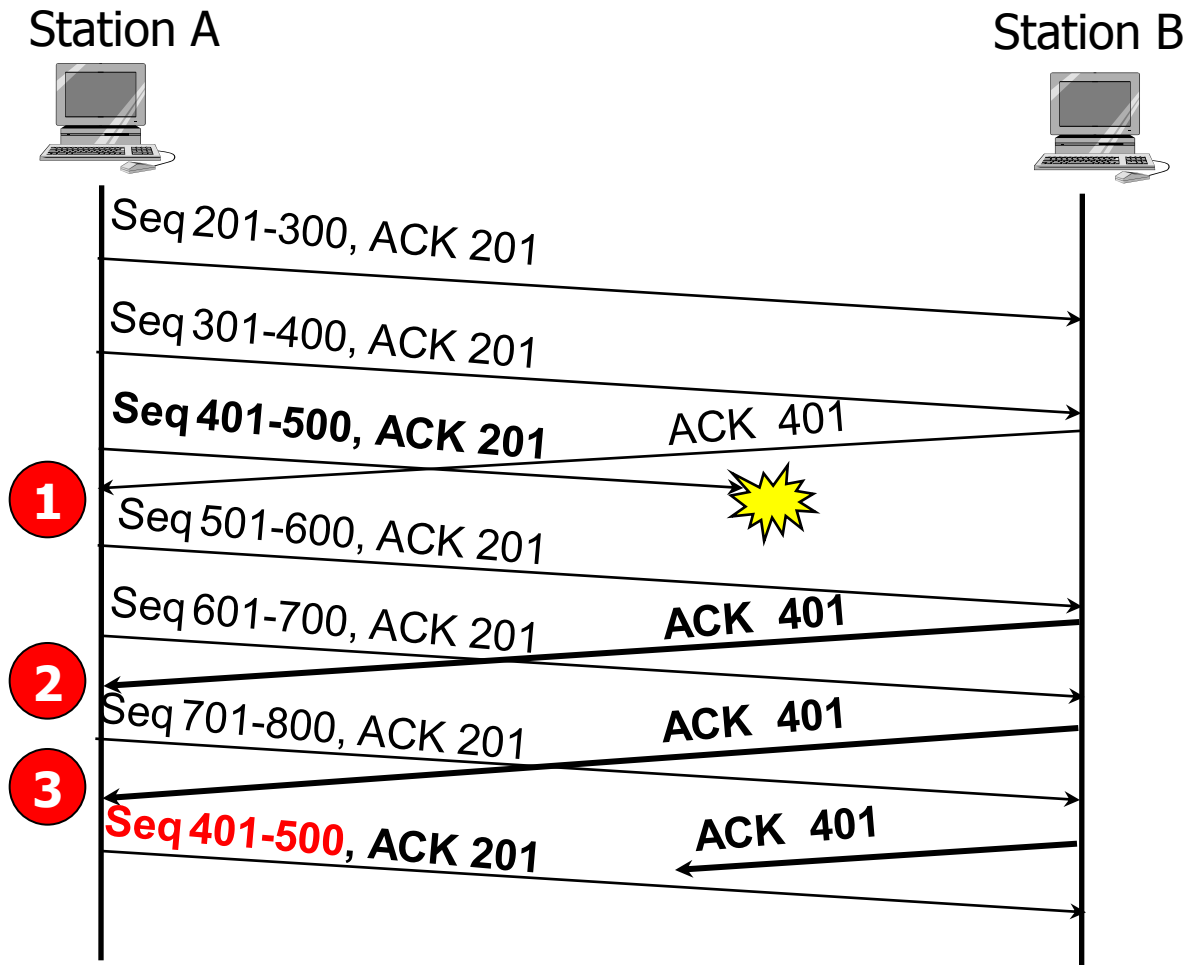
Acknowledgement



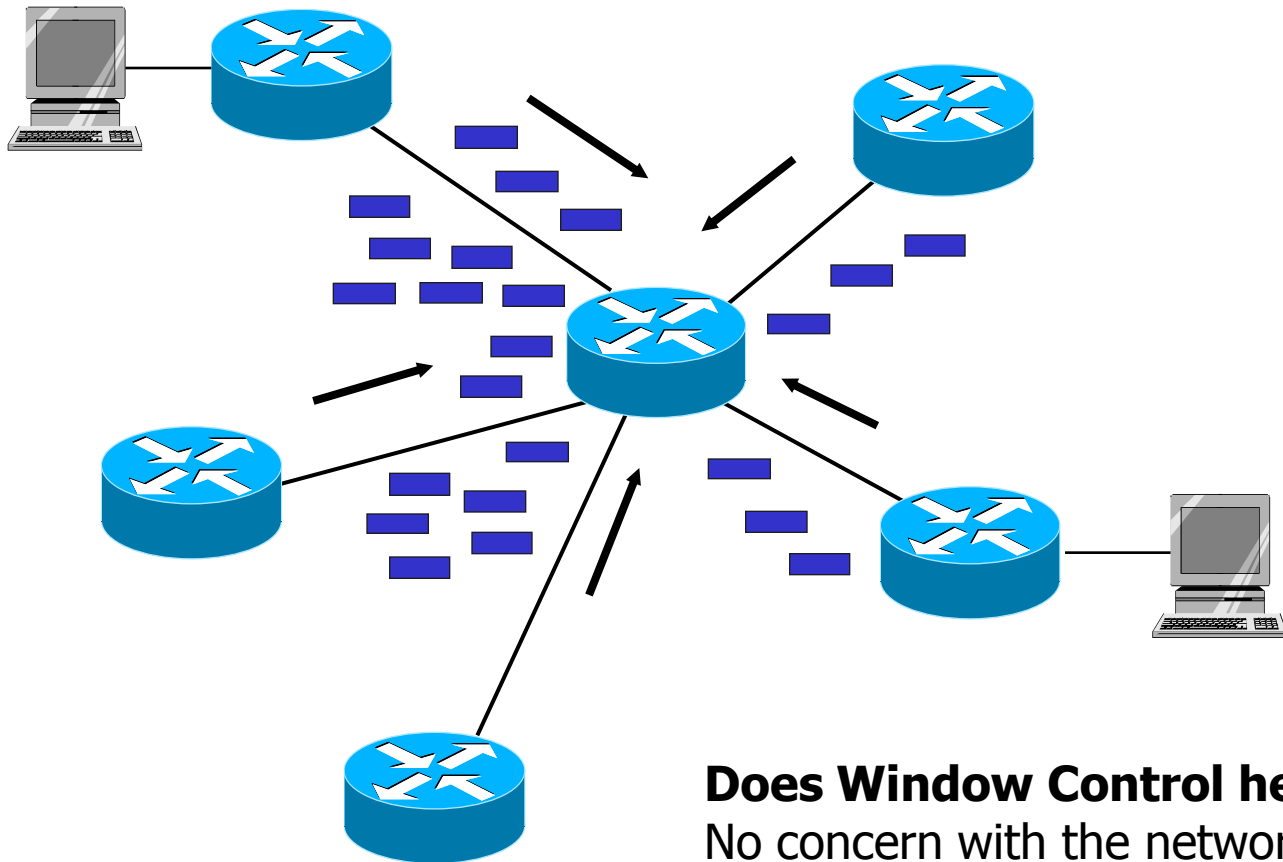
Lost Segment



Fast Retransmission



Congestion Control



Does Window Control help ?
No concern with the network status.
Every station keeps retransmitting.



Congestion Window

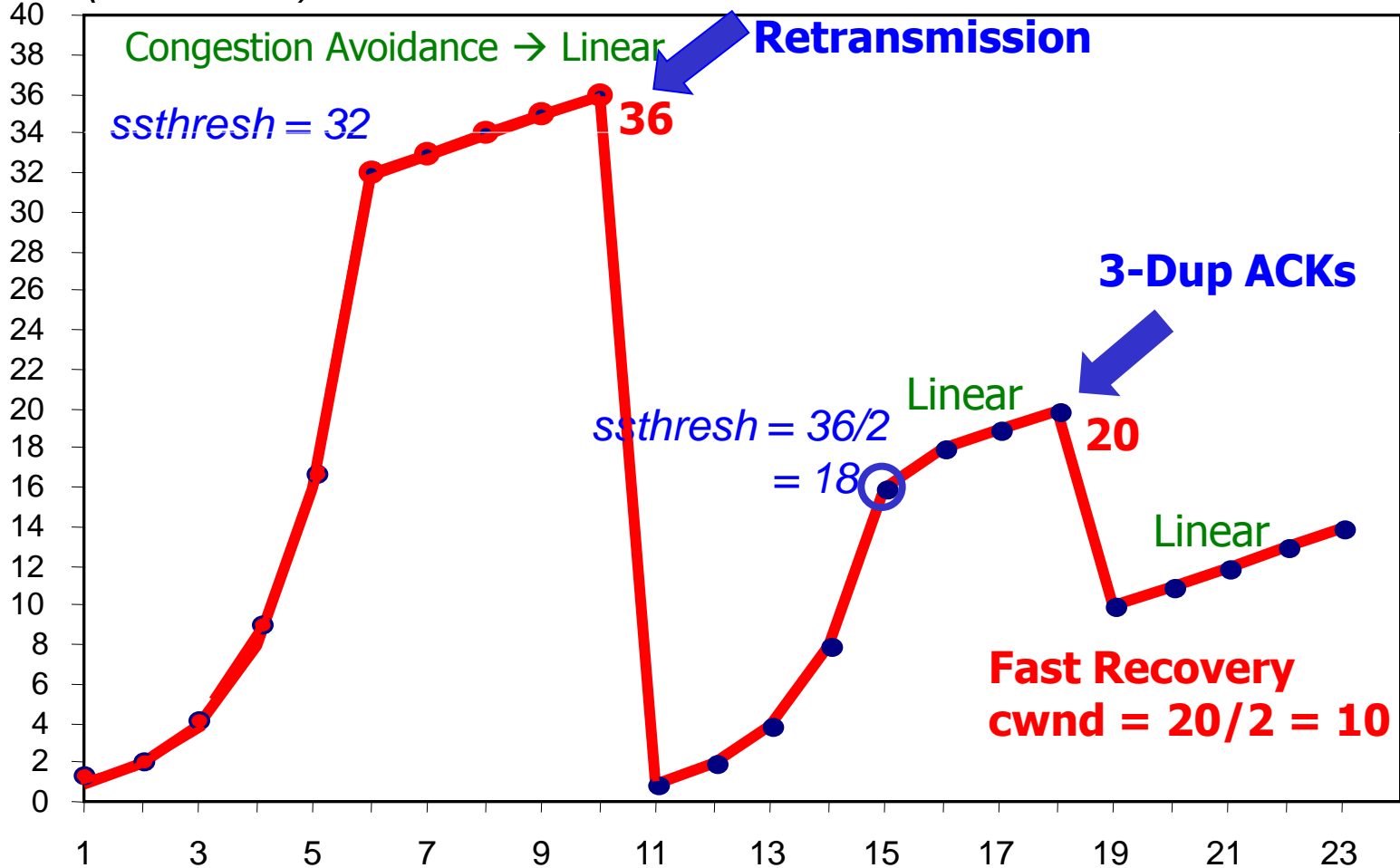
- cwnd
 - Concern about network status
- Different from “receiver window size”
 - Concern about Buffer @ receiver

Amount data send = min (cwnd, window size)

Slow Start

cwnd starts with value = 1
ssthresh: slow start threshold

cwnd (Max = 64)

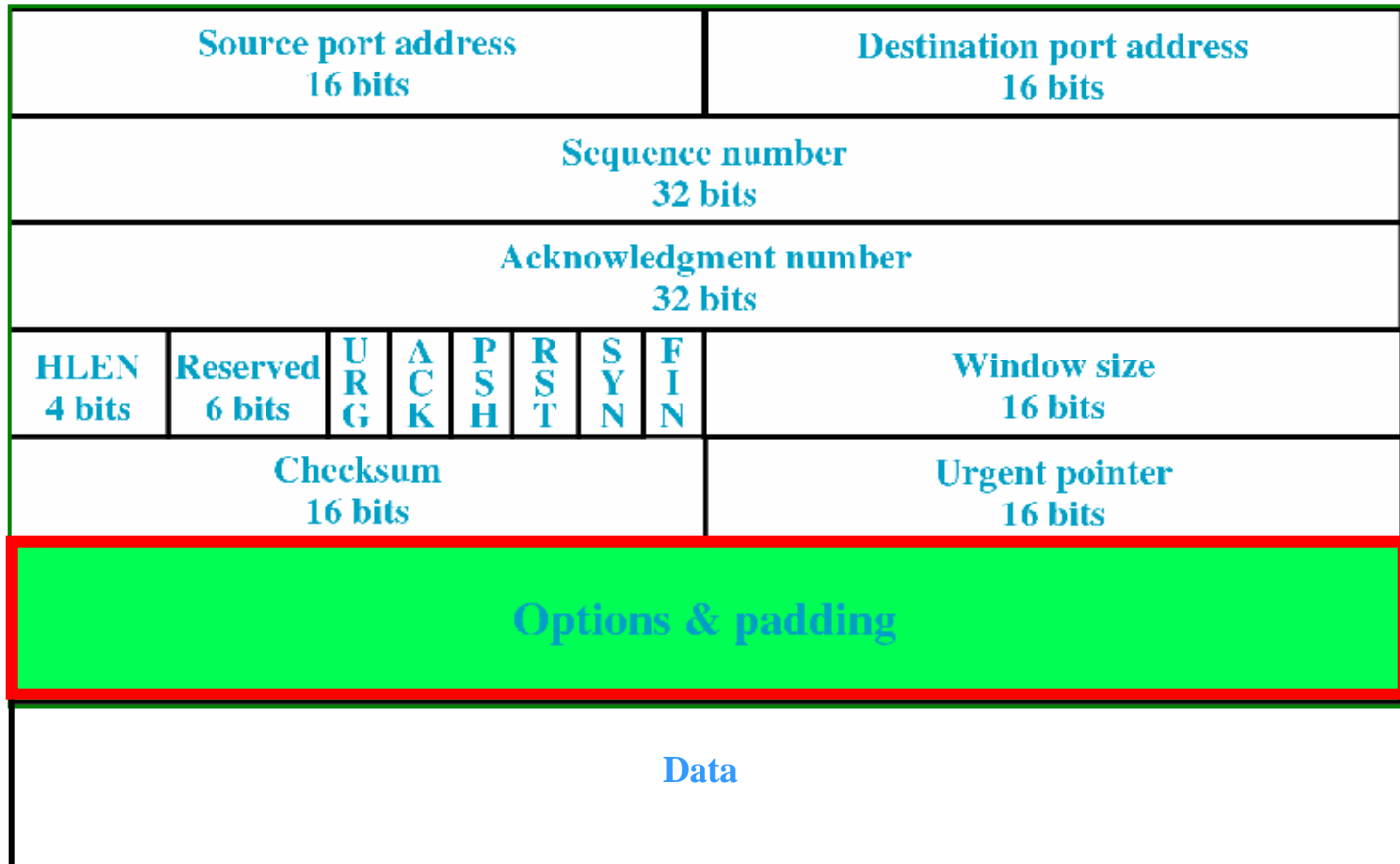




Understanding TCP Characteristics

- Keep-alive mechanism
- Sequence number
 - wrapping boundary
- **Slow start**
- No congestion detection in TCP
 - use **ICMP source quench** message
 - destination cannot receive too fast info., issue ICMP source quench to slow down sender

TCP Options



TCP Options

- Encoded as type-length-variable (TLV) sequences

| TCP Header | | | |
|--|--|--|-----------------------------------|
| 1st Option Type 8 bits | 1st Option Len 8 bits | 1st Option Data 16 bits | |
| 1st Option Data (Con't) | 2nd Option Type | 2nd Option Len | 2nd Option Data |
| 2nd Option Data | | | Padding |
| Data | | | |



TCP Options

To advertise the value of max. support TCP segment size

| | | |
|----------------------------|------------------------------|--|
| Type =2 (1 byte) | Length =4 (1 byte) | Max Segment Size in byte (2 bytes) |
|----------------------------|------------------------------|--|

To announce the window scaling factor

| | | |
|----------------|------------------|------------------------------|
| Type =3 | Length =3 | Window Scaling Factor |
|----------------|------------------|------------------------------|

To utilize time stamp

| | | |
|------------------------------------|-------------------|-------------------------|
| Type =8 | Length =10 | Timestamp |
| Timestamp (Con't) | | Echoed Timestamp |
| Echoed Timestamp (Con't) | | |



Choosing between TCP and UDP

- TCP
 - reliable transport services
- UDP
 - only delivery data to specific port



Protocols that use TCP

- File Transfer Protocol (FTP)
- Hypertext Transfer Protocol (HTTP)
- Simple Mail Transfer Protocol (SMTP)
- Post Office Protocol (POP3)
- Telnet
- Border Gateway Protocol (BGP-4)
- Label Distribution Protocol (LDP)



TCP well-known ports

| <i>Port</i> | <i>Protocol</i> | <i>Description</i> |
|-------------|-----------------|---|
| 7 | Echo | Echoes a received datagram back to the sender |
| 9 | Discard | Discards any datagram that is received |
| 11 | Users | Active users |
| 13 | Daytime | Returns the date and the time |
| 17 | Quote | Returns a quote of the day |
| 19 | Chargen | Returns a string of characters |
| 20 and 21 | FTP | File Transfer Protocol (Data and Control) |
| 23 | TELNET | Terminal Network |
| 25 | SMTP | Simple Mail Transfer Protocol |
| 53 | DNS | Domain Name Server |
| 67 | BOOTP | Bootstrap Protocol |
| 79 | Finger | Finger |
| 80 | HTTP | Hypertext Transfer Protocol |



Outline

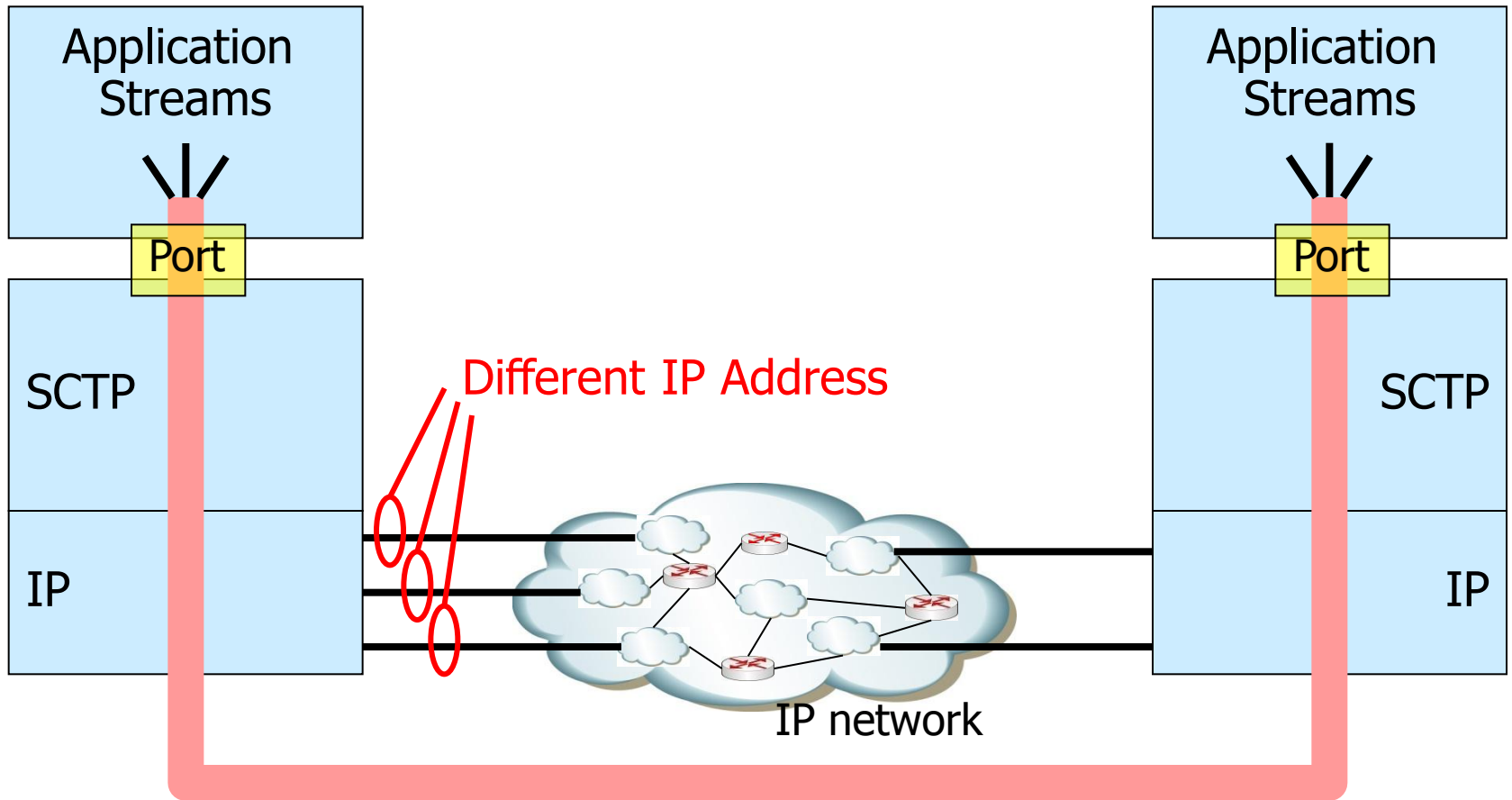
- Transport Layer
- User Datagram Protocol (UDP)
- Transmission Control Protocol (TCP)
 - Stream Control Transmission Protocol (SCTP)
 - Real-Time Transport Protocol (RTP)

Stream Control Transmission Protocol (SCTP)



- RFC 2960
- Transport **Packet Switched Telephone Network (PSTN)** connections over IP networks
- Reliable Connection-Oriented
- Support large blocks data transfer
- Includes sender pacing and congestion avoidance
- Establish and maintain **multiple streams** between same pair of end points

Multiple Streams

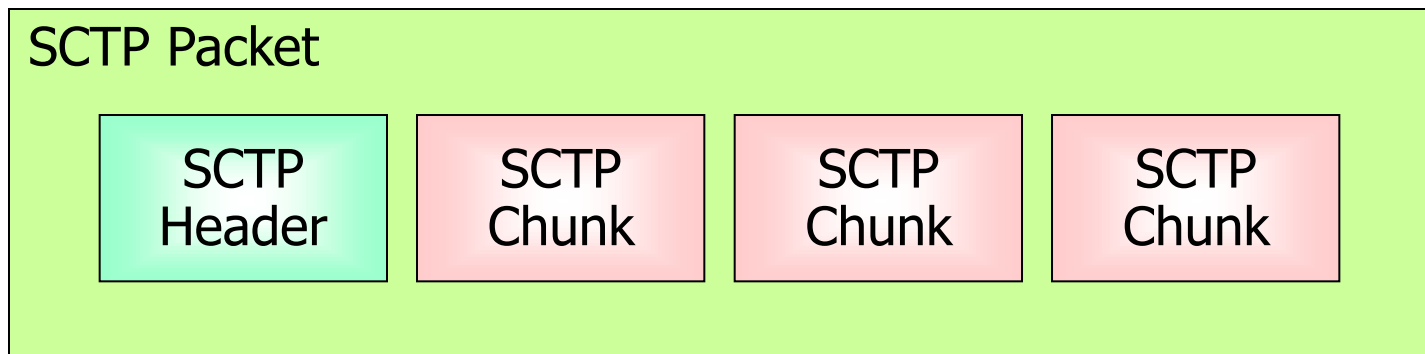




Performance Enhancements

- Bundle multiple SCTP messages into a single SCTP packet
 - reduces network overhead
 - reduce processing overhead
- Improve network-level fault tolerance
 - multihoming (multiple IP add.)
 - distinct routes
- Additional security for flooding /masquerate

SCTP Message Formats



SCTP Header: association and verification details

SCTP Chunk: control message / data for a stream



SCTP Message Formats

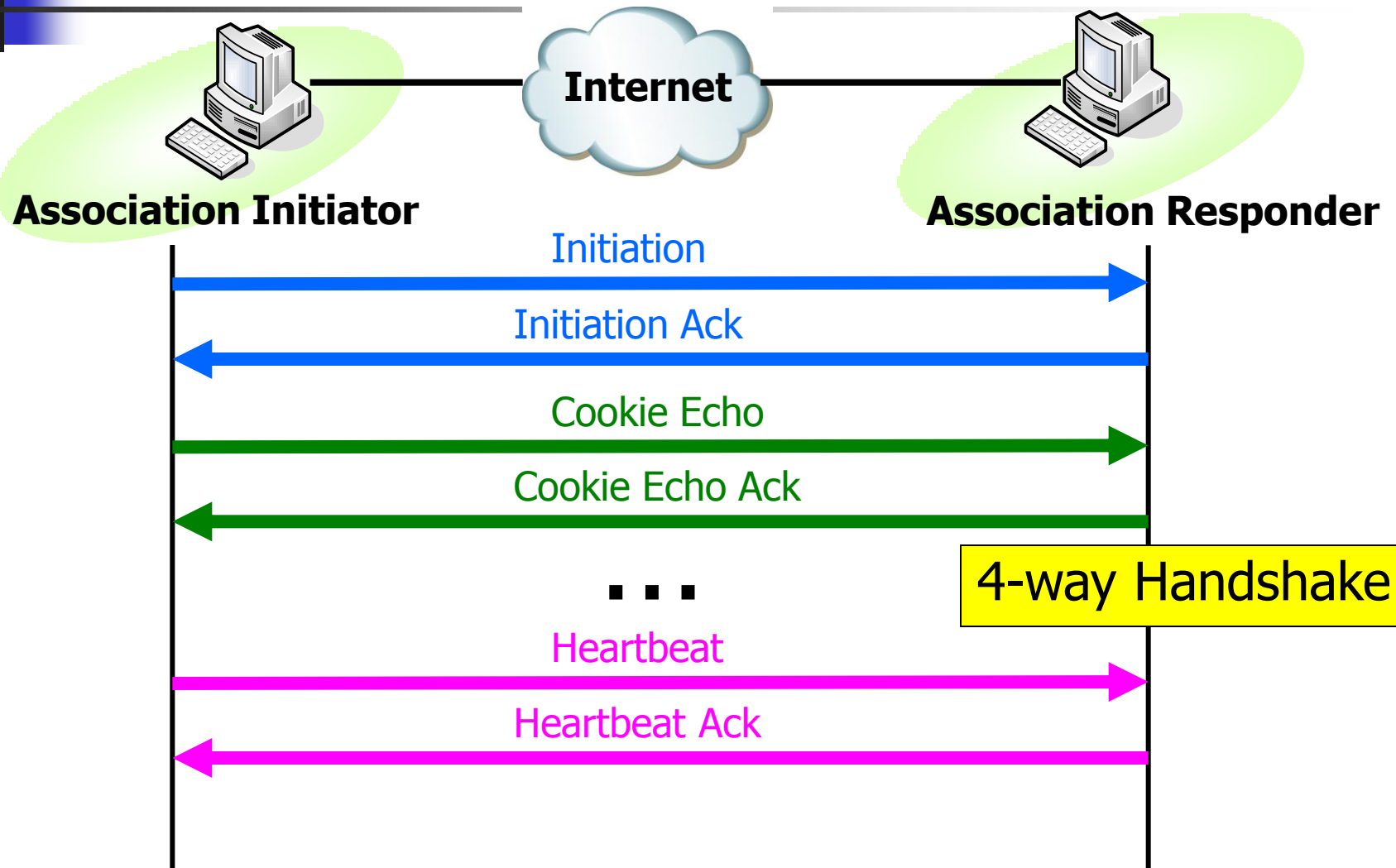
SCTP Header

| | |
|---|--------------------------------------|
| Source Port (16 bits) | Destination Port (16 bits) |
| Verification Tag | |
| Checksum (not using pseudo header) | |

SCTP Chunk

| | | |
|-------------------------------|-------------------------------|----------------------------------|
| Chunk Type (8 bits) | Chunk Flag (8 bits) | Chunk Length (16 bits) |
| Chunk Data | | |

Association and Management

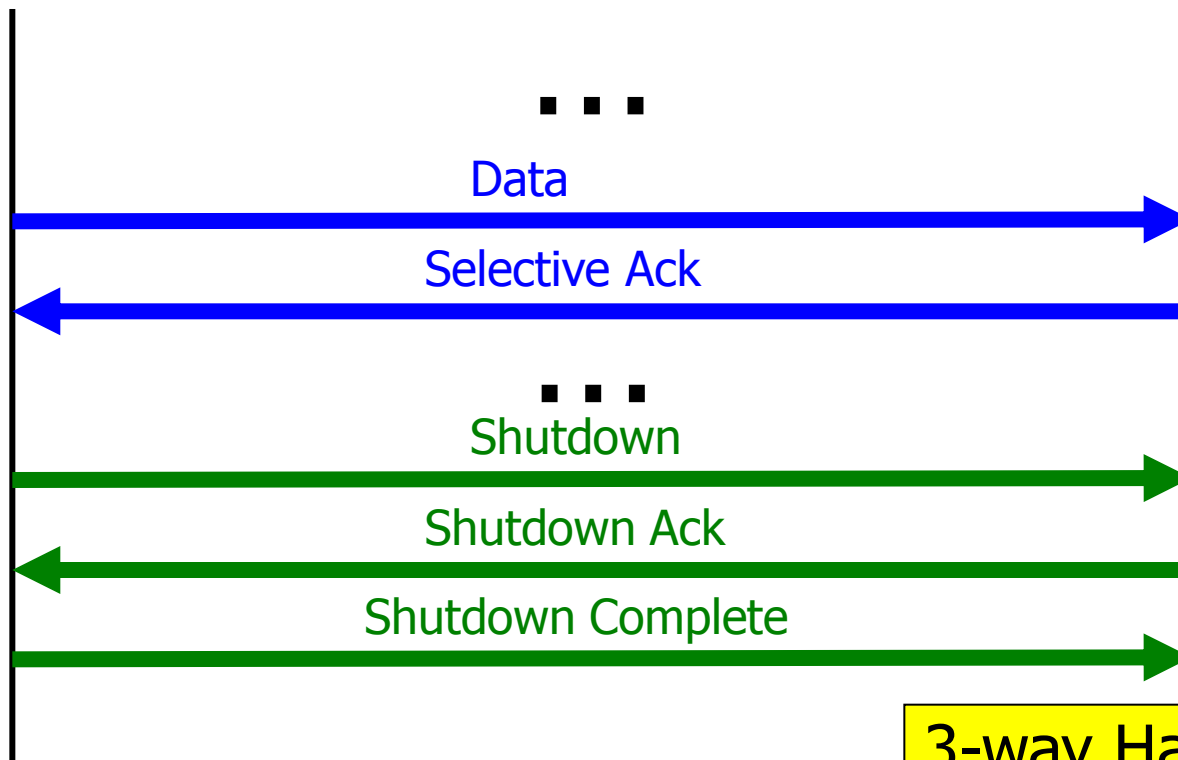
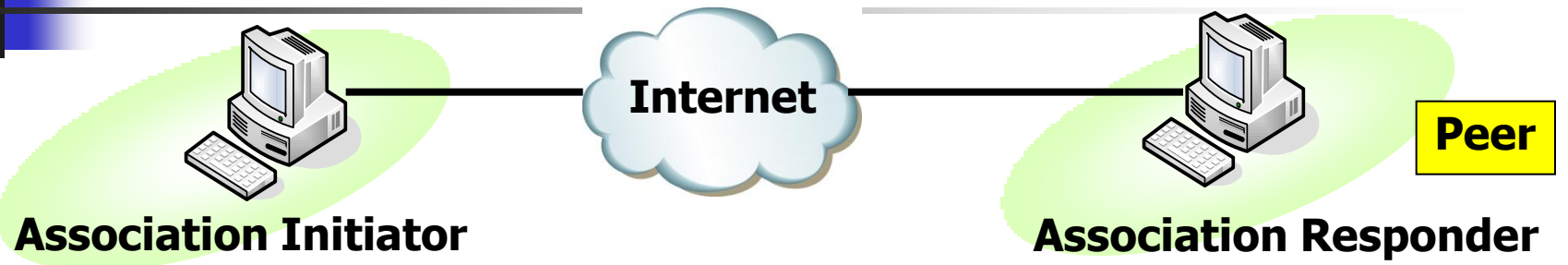




State Cookie

- Used for authentication
 - Message Authentication Code (MAC)
- Implement timer for controlling association

Association and Management



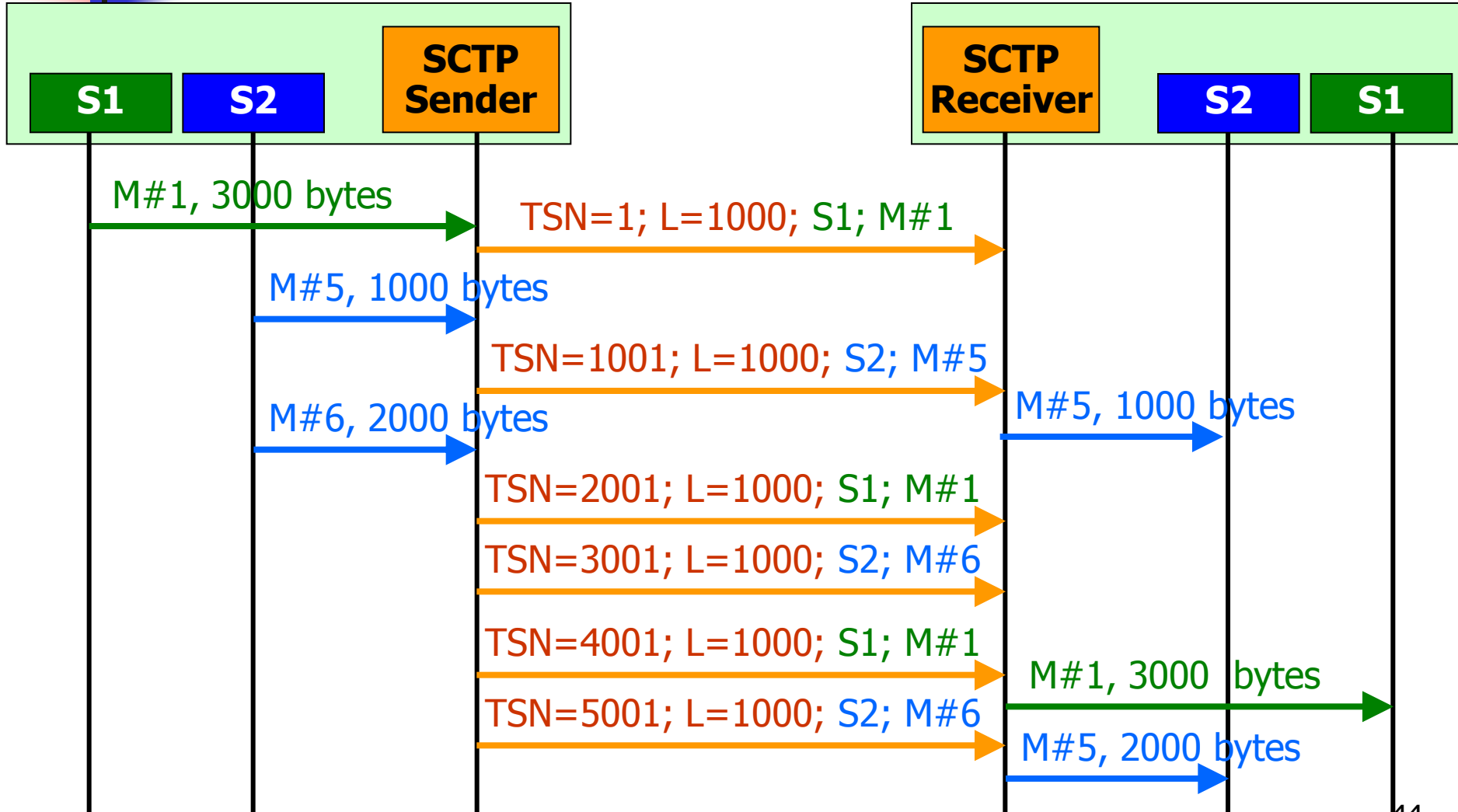
3-way Handshake



Data Transfer

- Same as TCP management
- Each chunk has Transmission Sequence Number (TSN)
 - identify 1st byte
- Most important feature – multiplex many streams onto same association
 - Stream Sequence Number

Multiplexing Data Streams





Note on SCTP

- Not commonly use
 - Lack of availability
 - both ends must implement SCTP
- Use in private networks
- Protocols that use SCTP
 - SIP
 - MTP2



Outline

- Transport Layer
- User Datagram Protocol (UDP)
- Transmission Control Protocol (TCP)
- Stream Control Transmission Protocol (SCTP)
- **Real-Time Transport Protocol (RTP)**

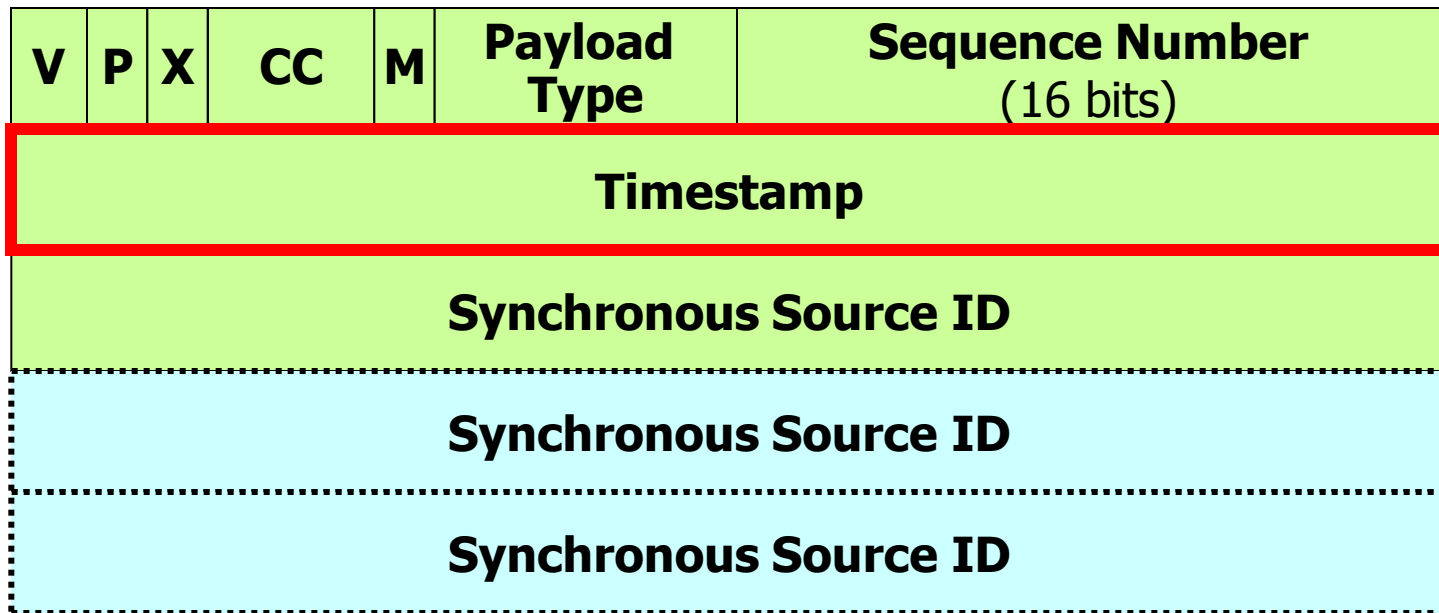


Real-Time Transport Protocol (RTP)

- Monitor and maintain QoS of real-time applications
- Lightweight
- Run over another transport protocol
 - Top-up protocol
 - top of UDP
- RTP needs management protocol
 - Real-Time Transport Control Protocol (RTCP)

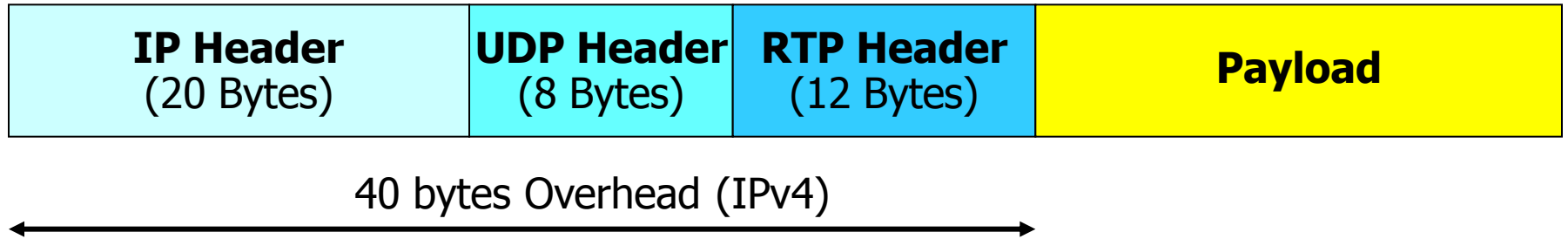
RTP Header

Version 2



At least 12 Bytes

RTP over UDP



For Audio: 16 bytes



71.4 % overhead



Summary

- Transport Layer
- User Datagram Protocol (UDP)
- Transmission Control Protocol (TCP)
- Stream Control Transmission Protocol (SCTP)
- Real-Time Transport Protocol (RTP)