



# UDP and TCP

---

รศ. ดร. อนันต์ พลเพิ่ม

Asso. Prof. Anan Phonphoem, Ph.D.

[anan.p@ku.ac.th](mailto:anan.p@ku.ac.th)

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

Computer Engineering Department

Kasetsart University, Bangkok, Thailand



# Outline

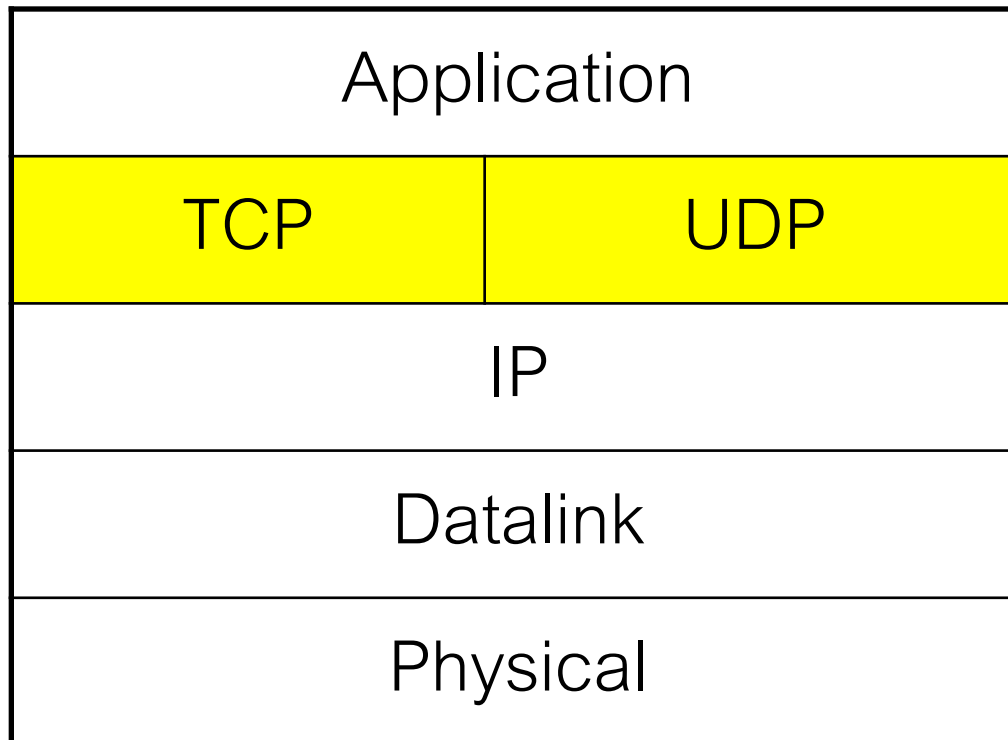
---

- **Transport Layer**
- User Datagram Protocol (UDP)
- Transmission Control Protocol (TCP)

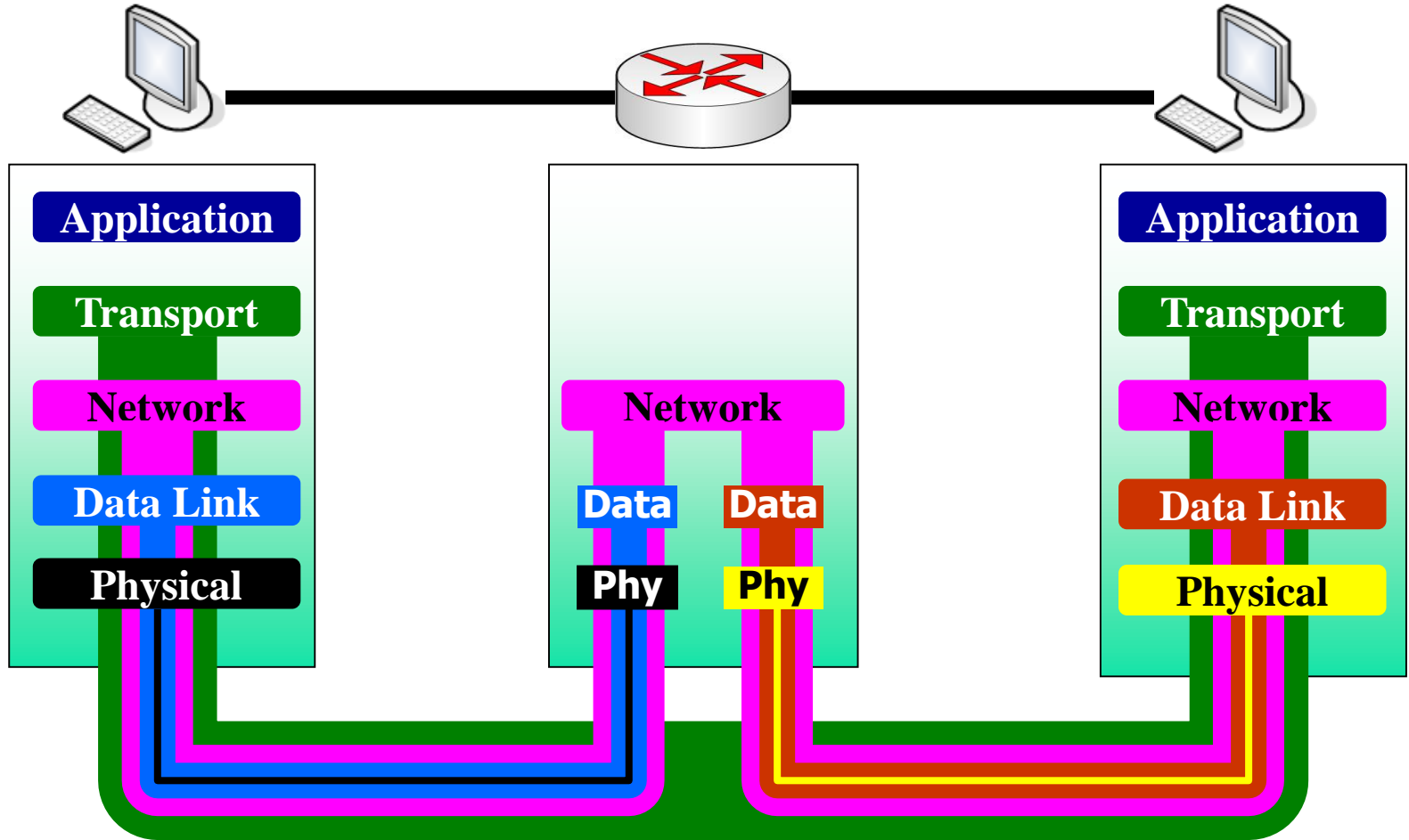


# TCP/IP protocol Suite

---



# TCP/IP protocol Suite





# Issues in Network Layer

---

- No/Little error detection
- No attempt to correct
  - Retransmission
- Point-to-point (connectionless)
- No handshaking
- No verification
- No flow control



# Transport Protocol

---

- Set of rules
  - for exchange control message / data
- End-to-End
- Application may by-pass Transport Protocol
  - build functions on top of IP
  - reduce overhead
  - application run on switch/router that has no Transport Layer



# Transport Protocols

---

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



# Ports and Addresses

---

- IP address is the end point of identifier
  - source / destination
- On same machine (same IP)
  - needs identifier for each application
  - multiple applications (e.g. 3 ftp sessions)
- “Port”
  - 16-bit number (65,356 ports) for each IP



# Ports and Addresses

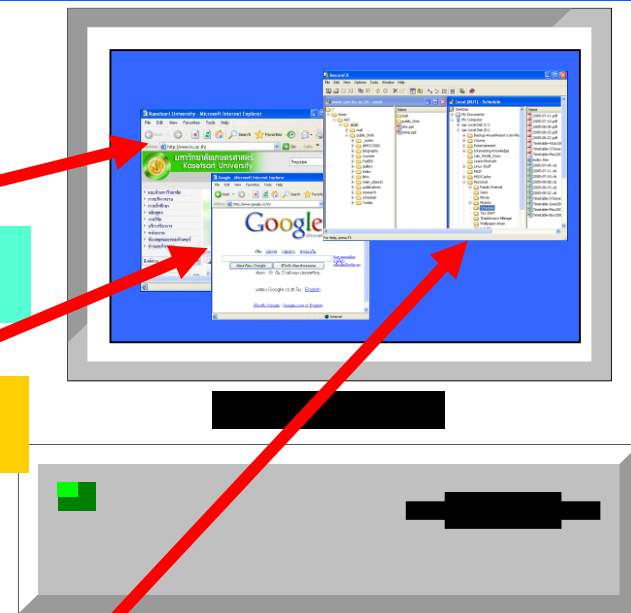
- Unique connection identifier
- [source IP] + [source port]
- [dest. IP] + [dest. port]

IP address: 158.108.1.2

(web) 158.108.1.2:**30000** -- 200.1.1.1:**80**

(web) 158.108.1.2:**35000** -- 200.1.1.1:**80**

(SSH) 158.108.1.2:**40000** -- 35.17.1.1:**22**





# Ports and Addresses

---

- Known destination IP
- Concern Issues
  - destination port ?
  - listen port ?
- IANA divides ports into three ranges
  - well-known ports
  - registered ports
  - dynamic / private ports



# Well-known ports

---

- 1 – 1023
- Used by server applications
- Use with restricted privileges (root)



# Well-known ports: Examples

Port Number	Description
80	WWW
25	SMTP
23	Telnet
22	SSH
21	ftp-control
20	ftp-data
161	SNMP



# Registered Ports

---

- 1024 – 49,151
- Assigned by IANA
- Used for server applications
  - Allow client to make contact



# Dynamic / Private ports

---

- 49,152 – 65,535
- Ephemeral ports (one day, temporary)
  - as-needed basis
  - freed up when done
- Assigned by local machine
- Never used for destination port at start
  - used by initiator as the return add.



# Connection-Oriented Transport

---

- Advantages
  - easy to communicate (if receiver is there!)
  - verify the real receiver
  - agree on some protocols
  - error correction
- Disadvantages
  - heavier-weight protocol
  - sophisticated implementation (**keep state**)
  - consume high bandwidth for management
- **TCP**



# Connectionless Transport

---

- Datagram approach
- Advantages
  - rapid data transfer
  - no connection setup
  - no need for connection maintenances
- Disadvantages
  - not reliable (or fail)
  - out of order packets
  - application more complex (takes precautions)
- **UDP**





# Outline

---

 Transport Layer

- **User Datagram Protocol (UDP)**
- Transmission Control Protocol (TCP)



# User Datagram Protocol (UDP)

---

- Characteristics
  - No error detection/recovery
  - No flow control
  - No checking for existing of destination
- Simple services
- Very useful protocol

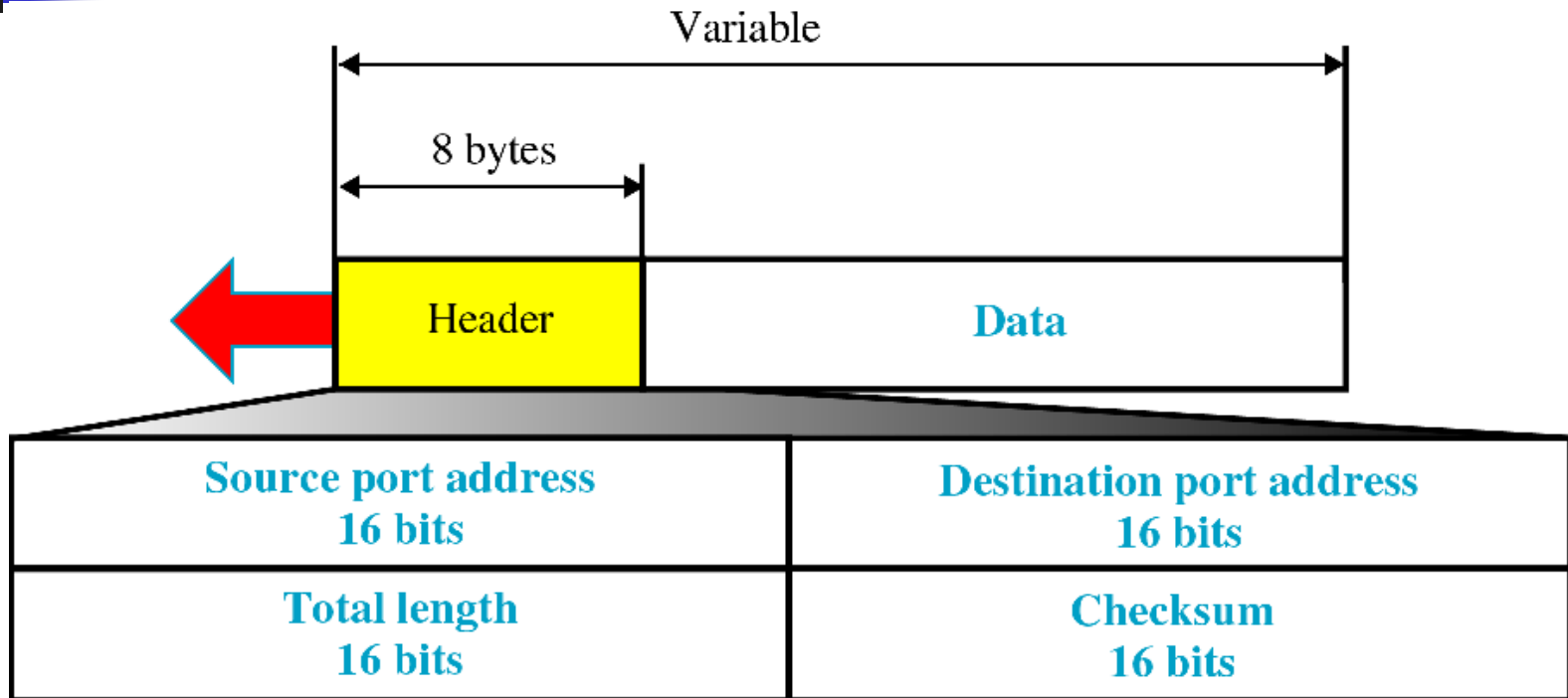


# UDP Basic Functions

---

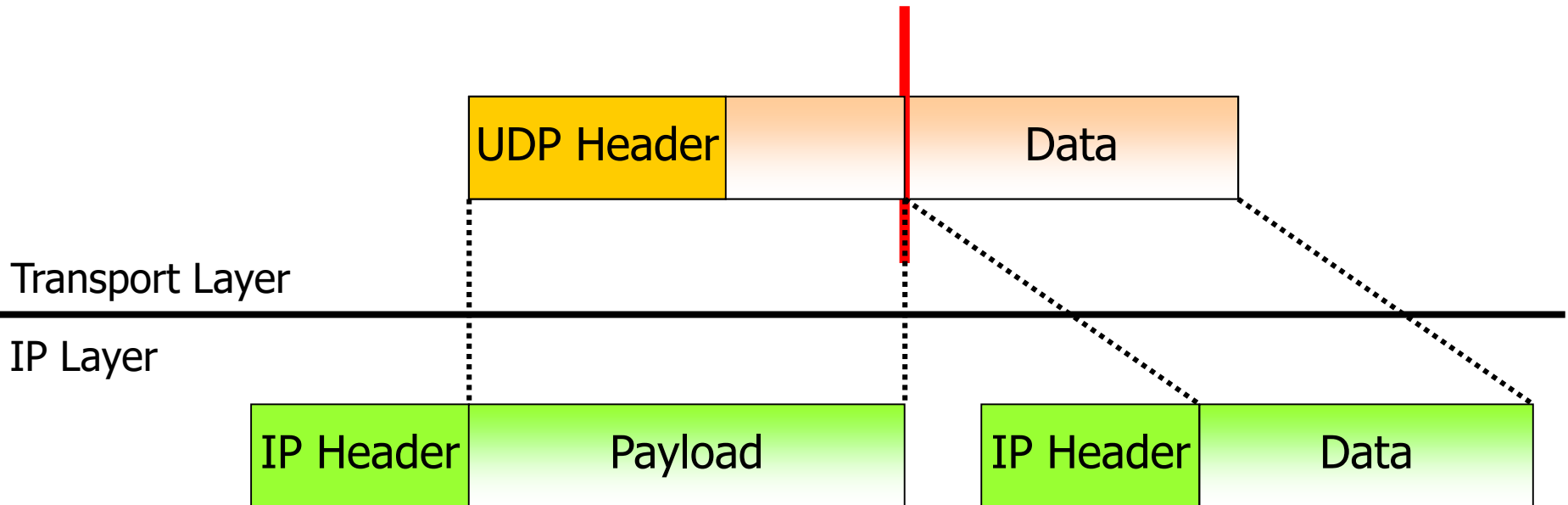
- provide basic functions above raw IP
  - Destination port – to specific application
  - Source port – for send back info
  - Data integrity verification – checksum
  - Data reassembly – segmented packets

# UDP Message Format



Total length = UDP header+ Payload  $\leq 2^{16}$  bytes

# UDP Message Fragmentation





# UDP Checksum

---

- Why needed it ?
  - Data link already provided CRC
  - No corrupted frame accepted
- Connectionless !
  - loss packets due to route
- Sender can choose not to use – set to 0
  - performance issue
- If error found
  - discard packet
  - responsibility of application



# Protocols that use UDP

---

- TFTP (trivial FTP)
- BOOTP (for diskless station)
- Network Time Protocol (NTP)
  - need basic delivery and checksum
- Network File System (NFS)
  - need low overhead
- Link Management Protocol (LMP)
  - original use raw IP (has its own error process)
  - move to UDP (preserve scarce IP protocol identifier)
- Label Distribution Protocol (LDP)
  - need to support multicast and broadcast
  - discovery mechanism



# Outline

---



Transport Layer



User Datagram Protocol (UDP)

- **Transmission Control Protocol (TCP)**



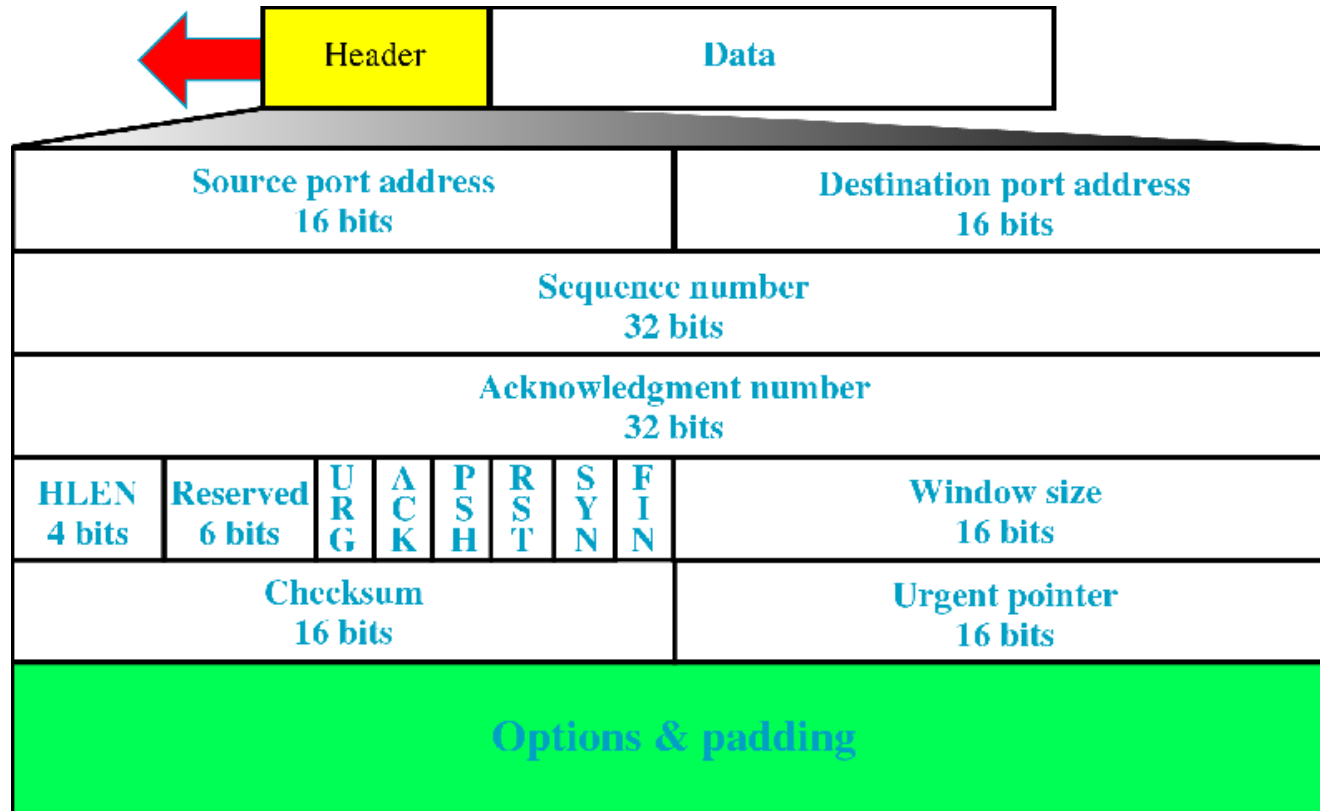


# Transmission Control Protocol (TCP)

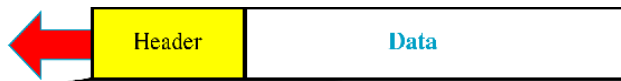
---

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

# TCP Messages (Segment)



# TCP V.S. UDP



Source port address 16 bits				Destination port address 16 bits				
Sequence number 32 bits								
Acknowledgment number 32 bits								
HLEN 4 bits	Reserved 6 bits	URG	ACK	PSH	RST	SYN	FIN	Window size 16 bits
Checksum 16 bits				Urgent pointer 16 bits				
Options & padding								

**TCP** Header  
20 Bytes<sup>+</sup>



Source port address 16 bits		Destination port address 16 bits	
Total length 16 bits		Checksum 16 bits	

**UDP** Header  
8 Bytes

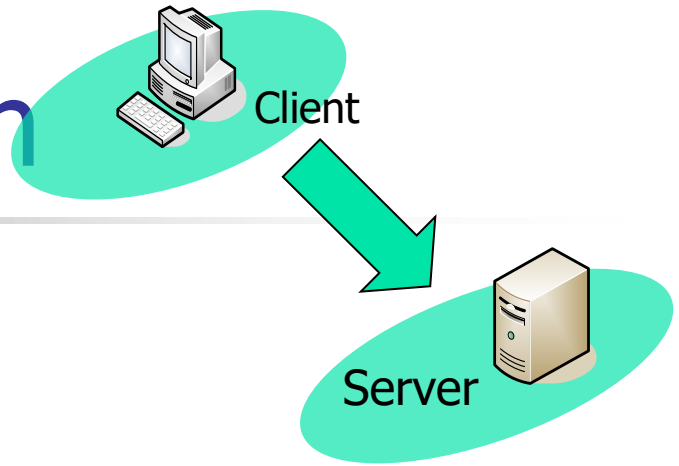


# Connection Establishment

---

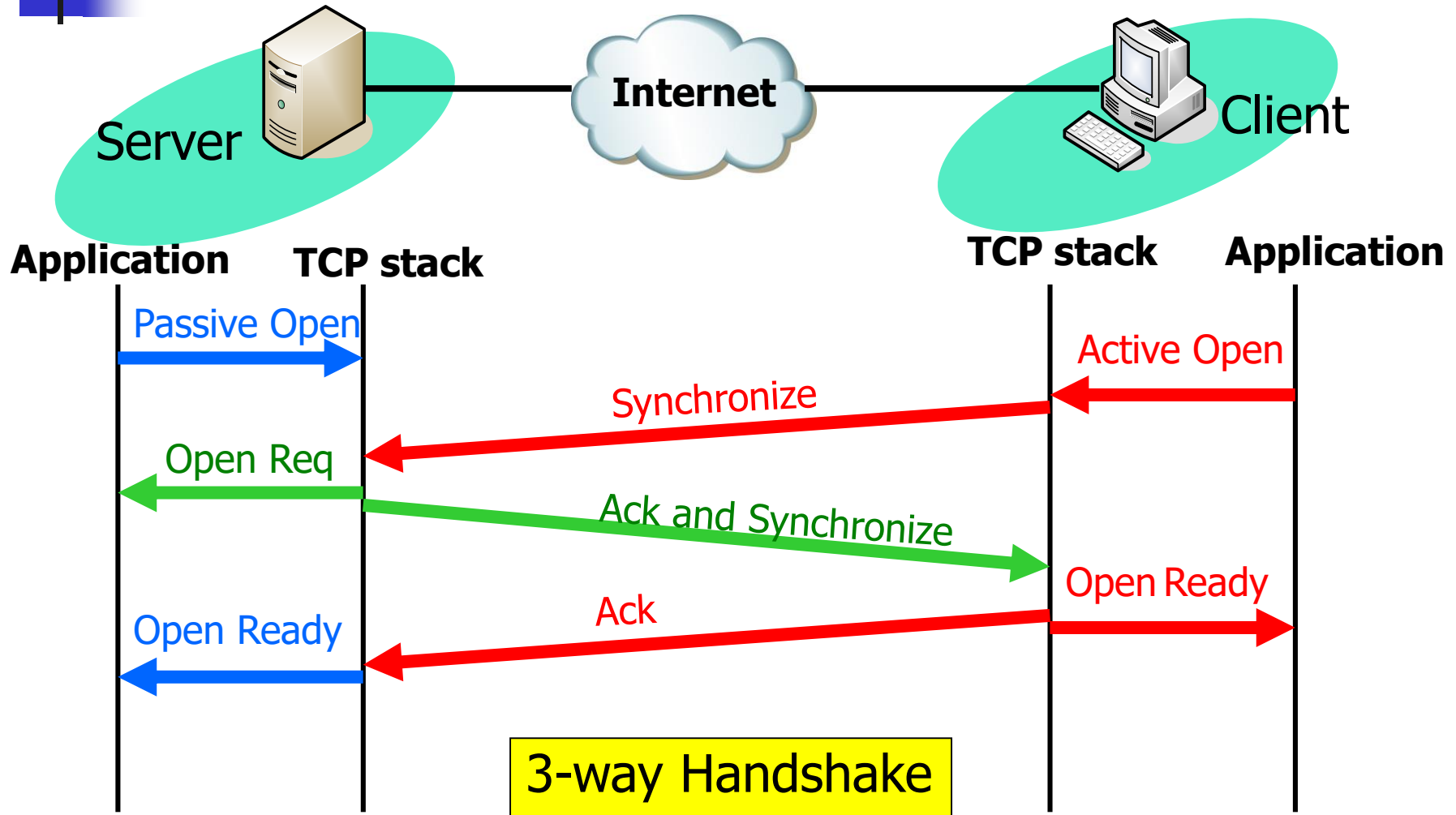
- Application requests connection through **socket API**
  - not part of TCP
  - method to access services provided by TCP
- To start connection establishment
  - receiver must be listening

# Making connection



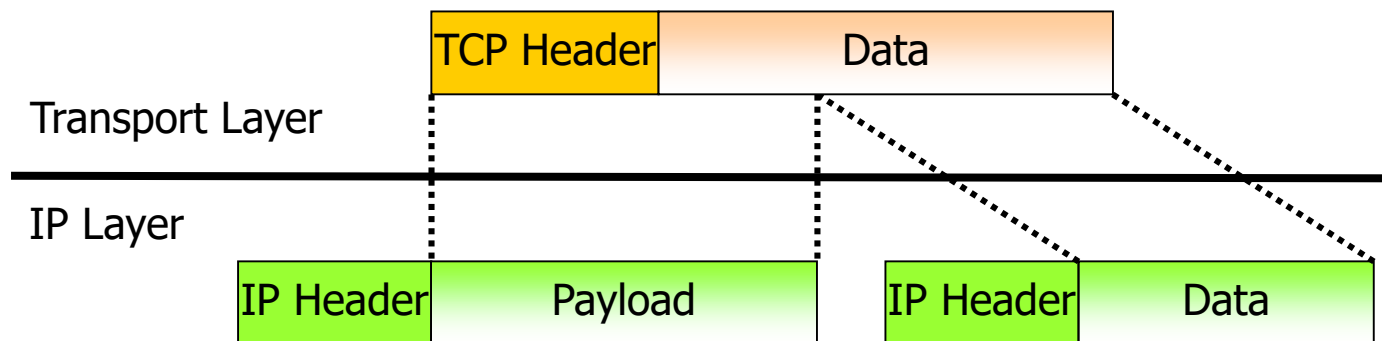
- Application on server
  - passive receive mode
  - listening by issue **Listen request** to socket **locally**
- Application on Client
  - request to socket API **locally** with
    - destination IP
    - destination port
    - (source port: if not specify, TCP assigns private port)

# TCP Connection Establishment

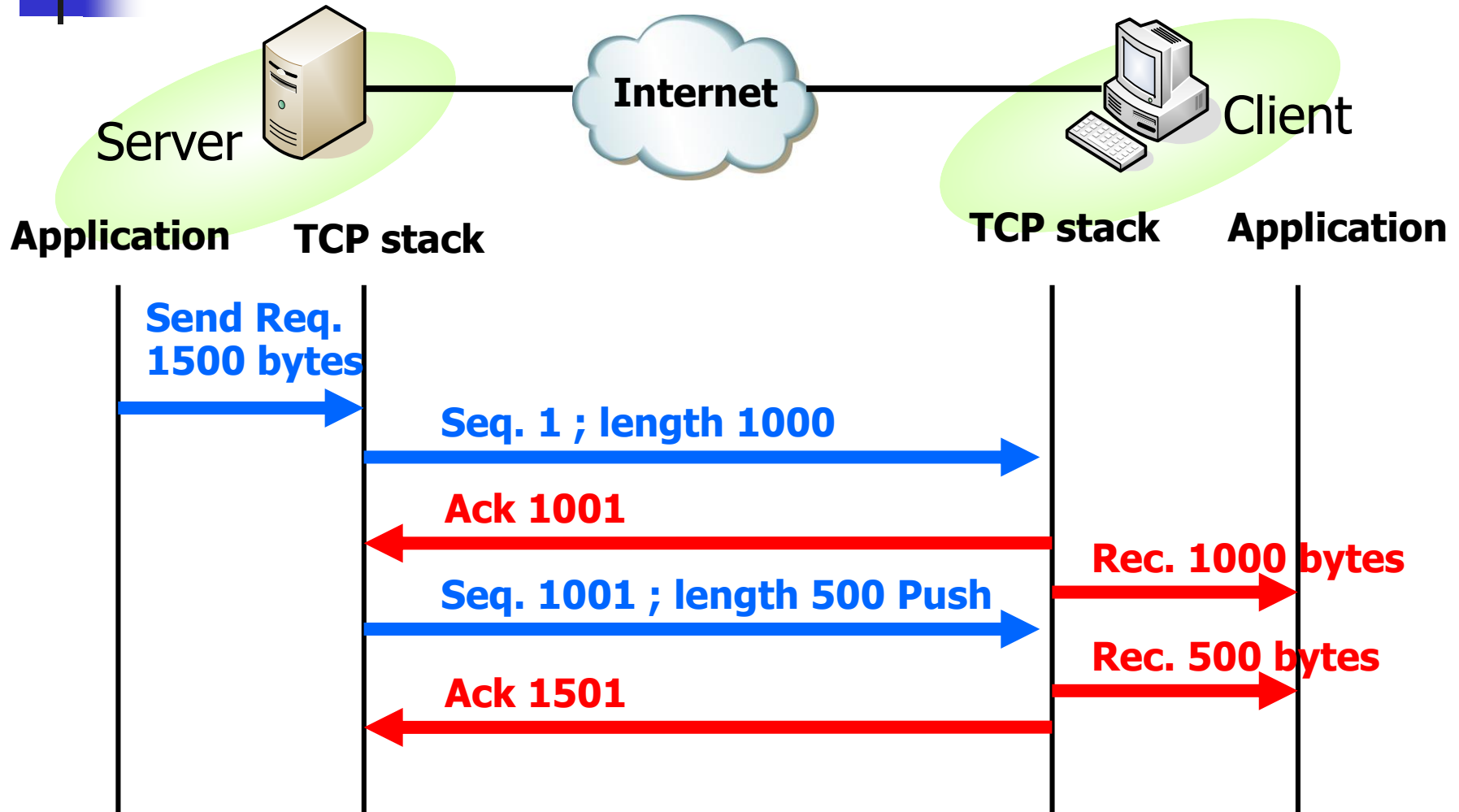


# TCP Segment

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

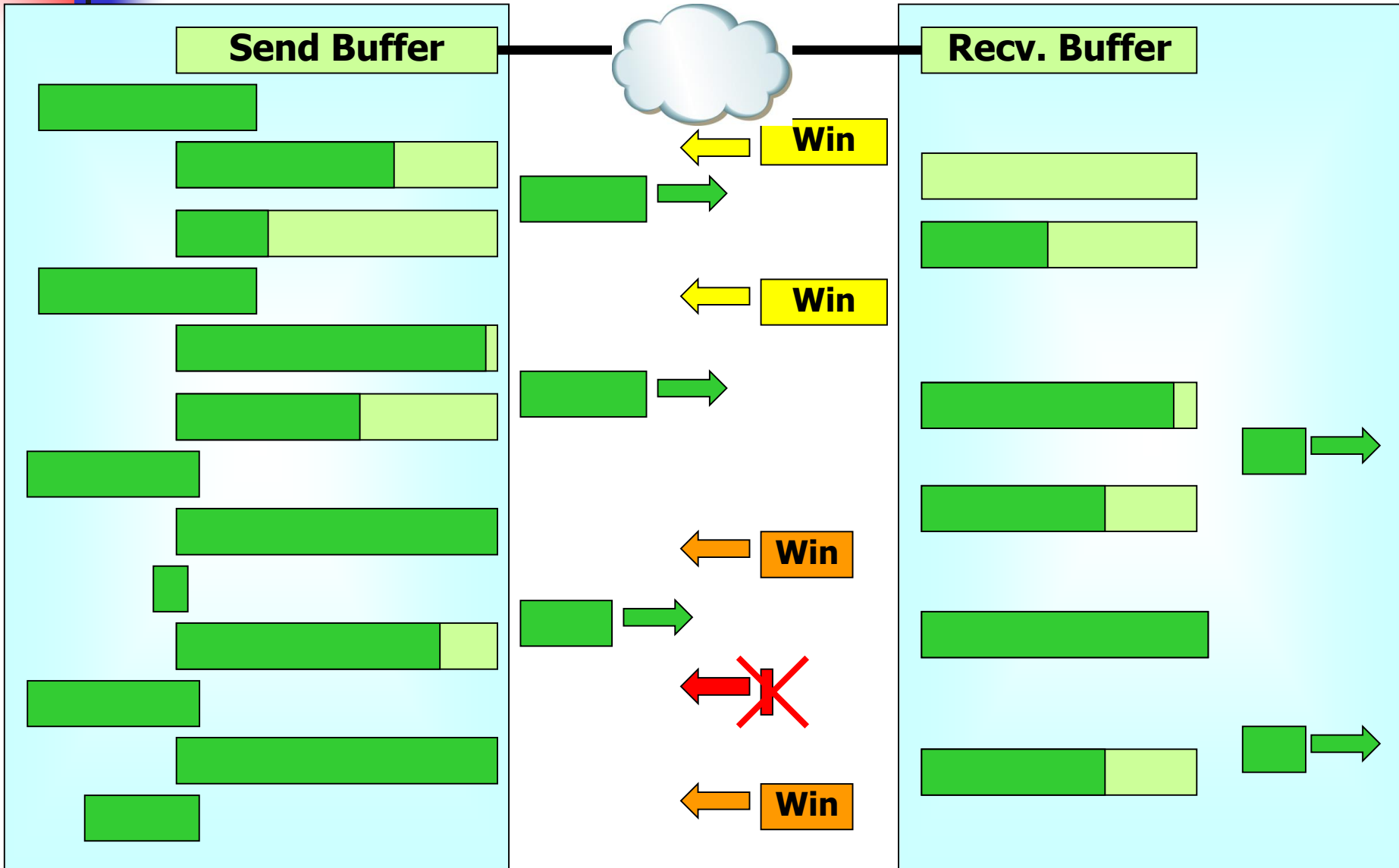


# Data Transmission (I)

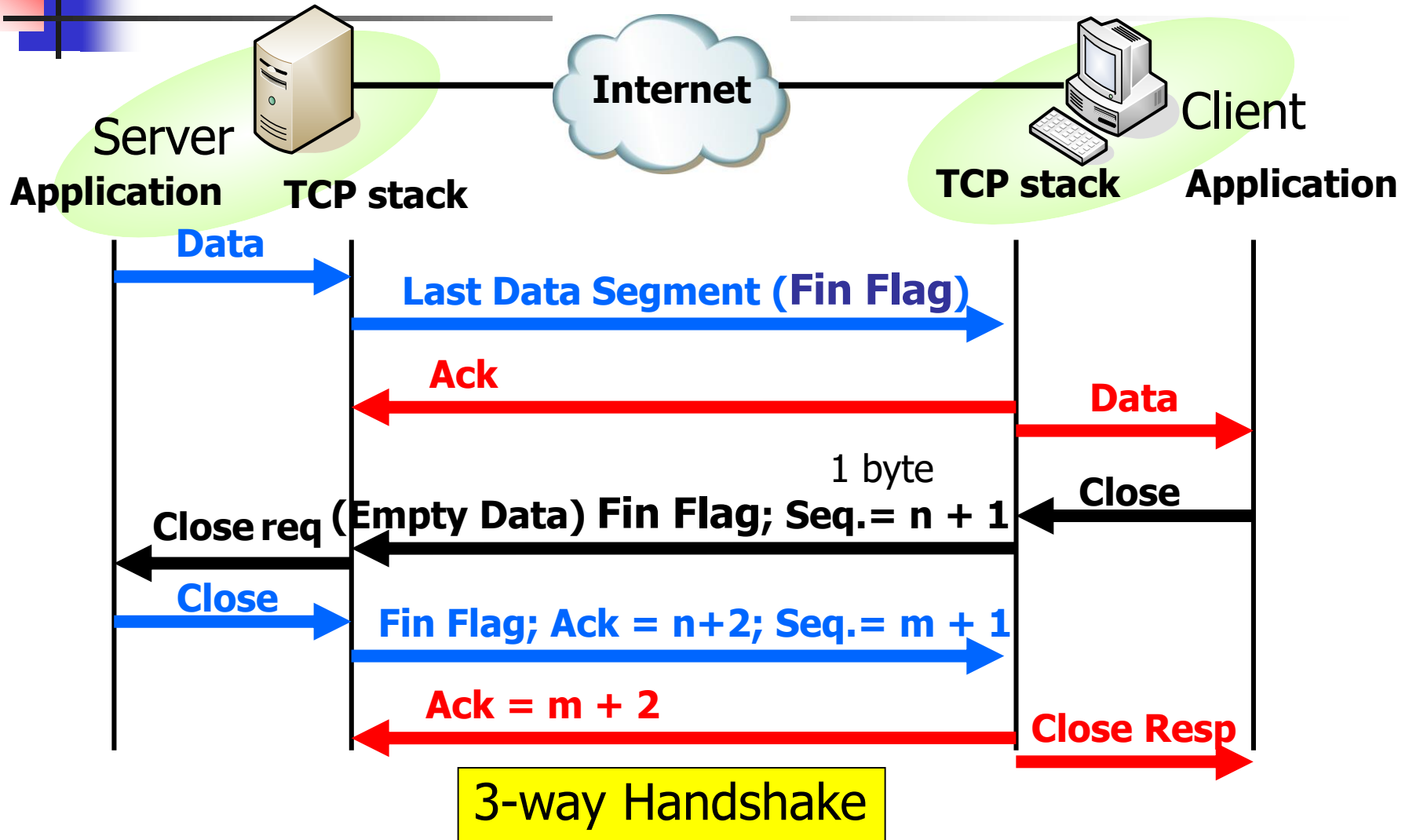




# TCP Window Control



# 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 HTTP
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)

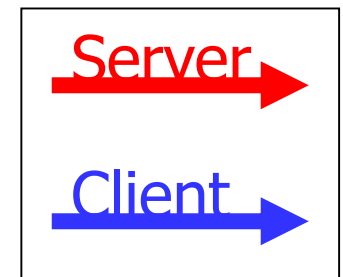
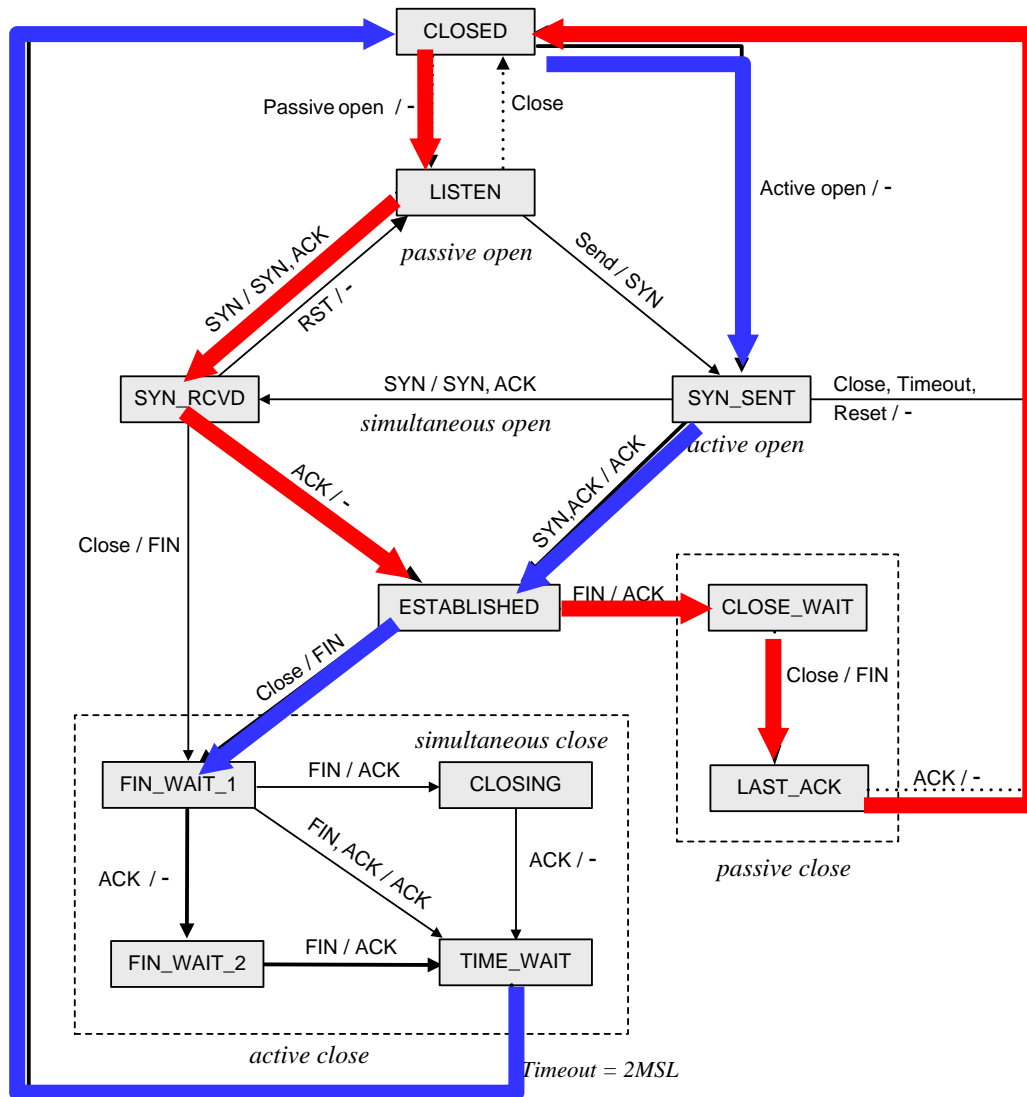
⊕ Ethernet II, Src: Dell\_3a:93:38 (00:12:3f:3a:93:38), Dst: 08:1f:f3:a6:66:cc (08:1f:f3:a6:66:cc)

⊕ 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





# 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)



# Choosing between TCP and UDP

---

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



# Summary

---

- Transport Layer
  - port & IP address
- User Datagram Protocol (UDP)
  - packet format (8-byte header)
  - functions
- Transmission Control Protocol (TCP)
  - packet format ( $\geq$  20-byte header)
  - connection establishment / tear down
  - window control