System Architecture & Security

Applied Information Security

SSAS F2016

Søren Debois

Review

Last lecture

- Introduction to the course.
- What is IT Security?
  (Confidentiality, Integrity, Availability, Accountability)
- 12 Security Principles.
  Introduction to the command-line.

12 Principles (1-6)

- Simplicity
- Open Design
- Compartmentalisation,
- Minimum exposure
- Least Privilege
- Minimum trust & maximum trustworthiness,

12 Principles (7-12)

- Secure fail-safe defaults
- Complete mediation
- No single point of failure
- Traceability
- Generating secrets
- Usability

Trust

- “I write a web-app that I know will be accessible only from inside the ITU wired network. As such, I can't be bothered to encrypt sensitive information (passwords, grades) being sent to and from my app's server—only ITU staff has access to the wires anyway.”

- Poor choices (complete mediation, min exposure)

Secure, fail-safe defaults

- The computer games SimCity III and Diablo II both required an always-on internet connection to be playable, even in single-player mode, …
- Network outage is a failure.
- Not about Trust; the player is the adversary.
- Not about Generating Secrets: we're not generating any, here.

Traceability

- “Edward Snowden and Bradley Manning both accessed enormous amounts of information classified by the US Government. Their accesses were apparently neither logged nor constrained in ways beyond having access to a particular network.”

- Why would you say “Simplicity”?

Open Design

- About “security mechanisms”.
- Usually crypto algorithms.
- Don’t give away your source code and network schematics; that just violates “Minimum exposure”.

Meta

Office hours

- Very few people showed.

- Why?

30 people did not pass the Security Principle Quiz

Quiz meta

- Check your “course completion” on learnit. You can go to the exam iff its “completed”.

- (It’ll obviously be “not completed” when you haven’t done some quiz.)

Exam

- Written, on-premises, no restrictions.

- In particular, you can use the e-book if you like.

Lab sessions/Office hours

- Opportunity to work with other students.
- Mon 16-17 @4A14
  Tue 9-10 @2A12
  Wed 16-17 @3A12
  Thu 16-17 @2A12
  Fri 9-10 @2A12

Chapter 3:
Network Services

Plan

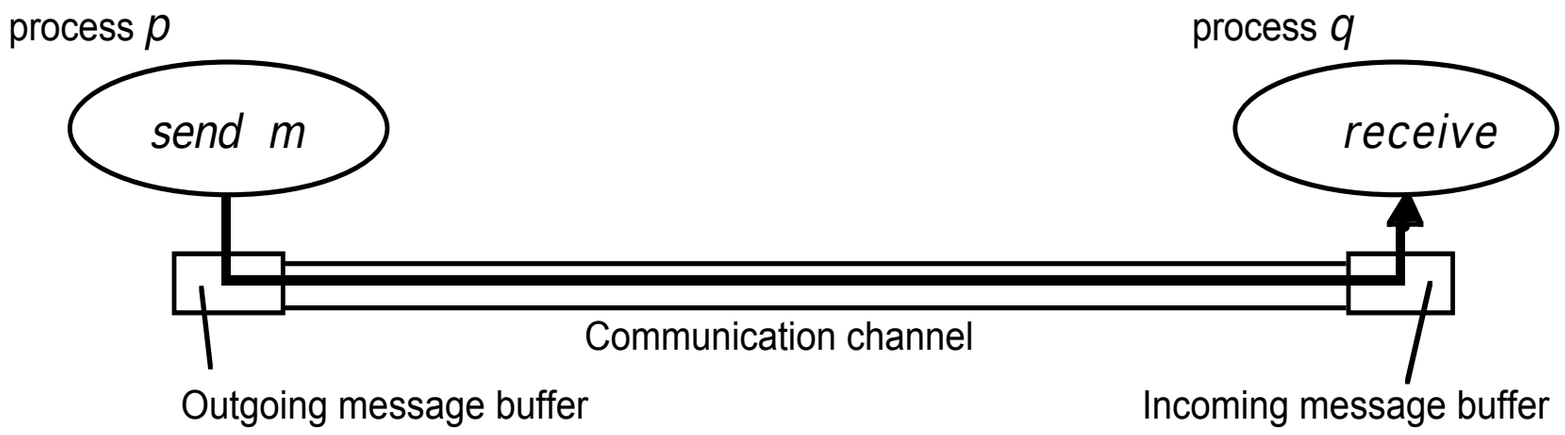
- Foundations of Networking
- IP
- UDP & TCP

Foundations of Networking

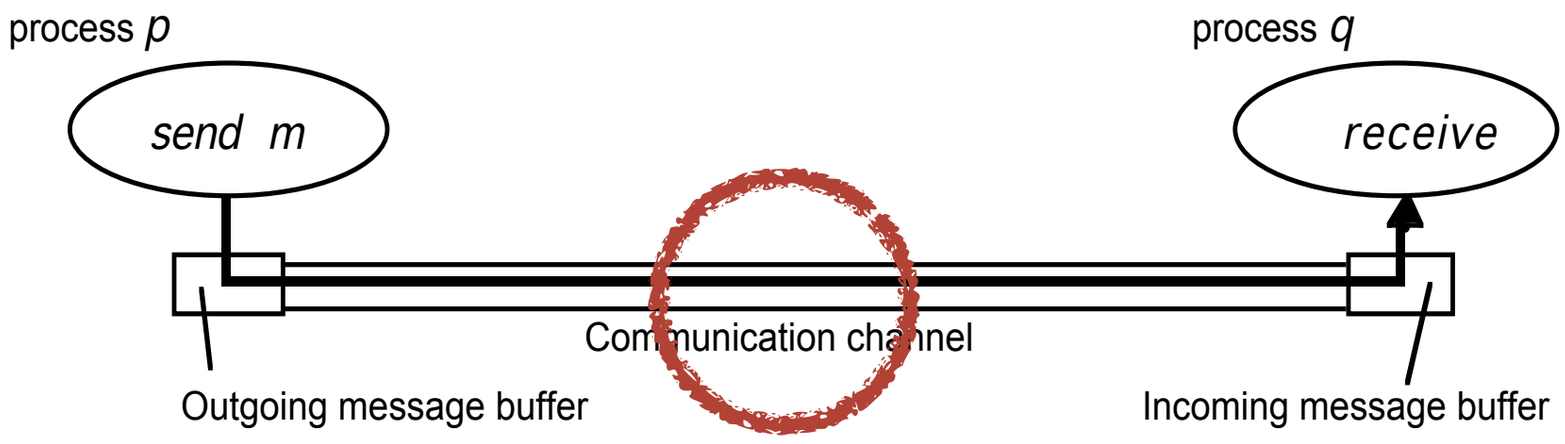
Terminology (i)

- *Processes* run on *hosts*.
- Processes send and receive *messages* on *communication channels.*
- Processes and communication channels are both subject to *failures.*
- Channel failures may yield *dropped, lost, re-ordered* or *duplicated* messages.

Basic model


process p
process q
send m
receive
Communication channel
Outgoing message buffer
Incoming message buffer

Basic model: Failures



Channel may exhibit omission & byzantine failures

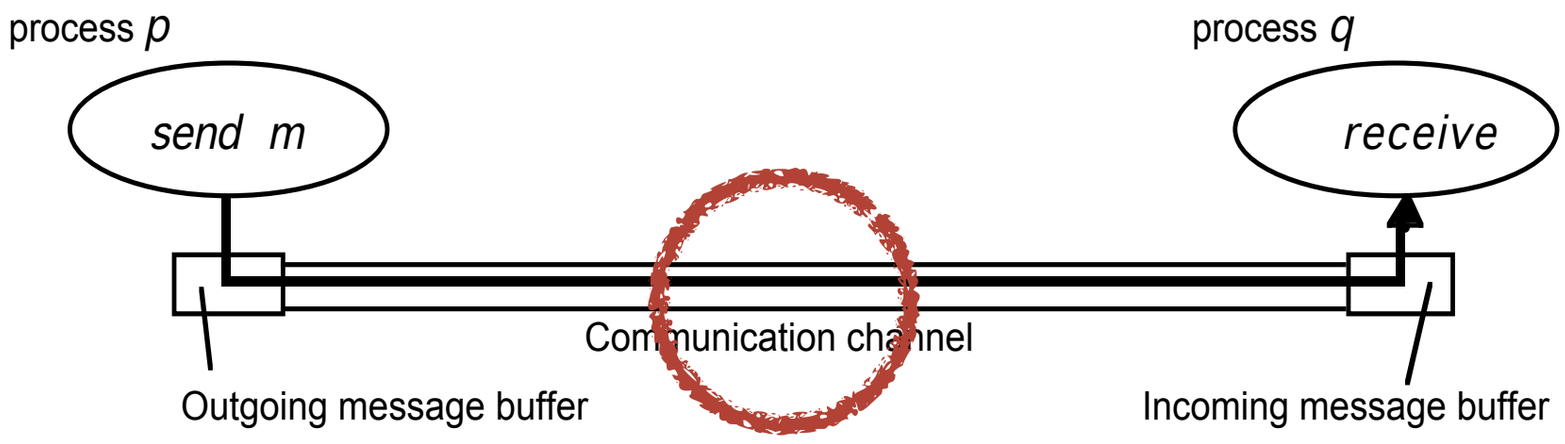
Channel failure handling

- Checksums
  (vs. message corruption)

- Acknowledgments, timeouts, re-transmissions
  (vs. lost messages)

- Sequence numbers
  (vs. duplicated/re-ordered messages)

Basic model: Security



Adversary may
intercept, copy, transform, insert, and delete
messages

Terminology (ii)

- Reliable vs unreliable channels/protocols
- Connection-oriented vs connection-less/protocols
- Datagram vs streaming

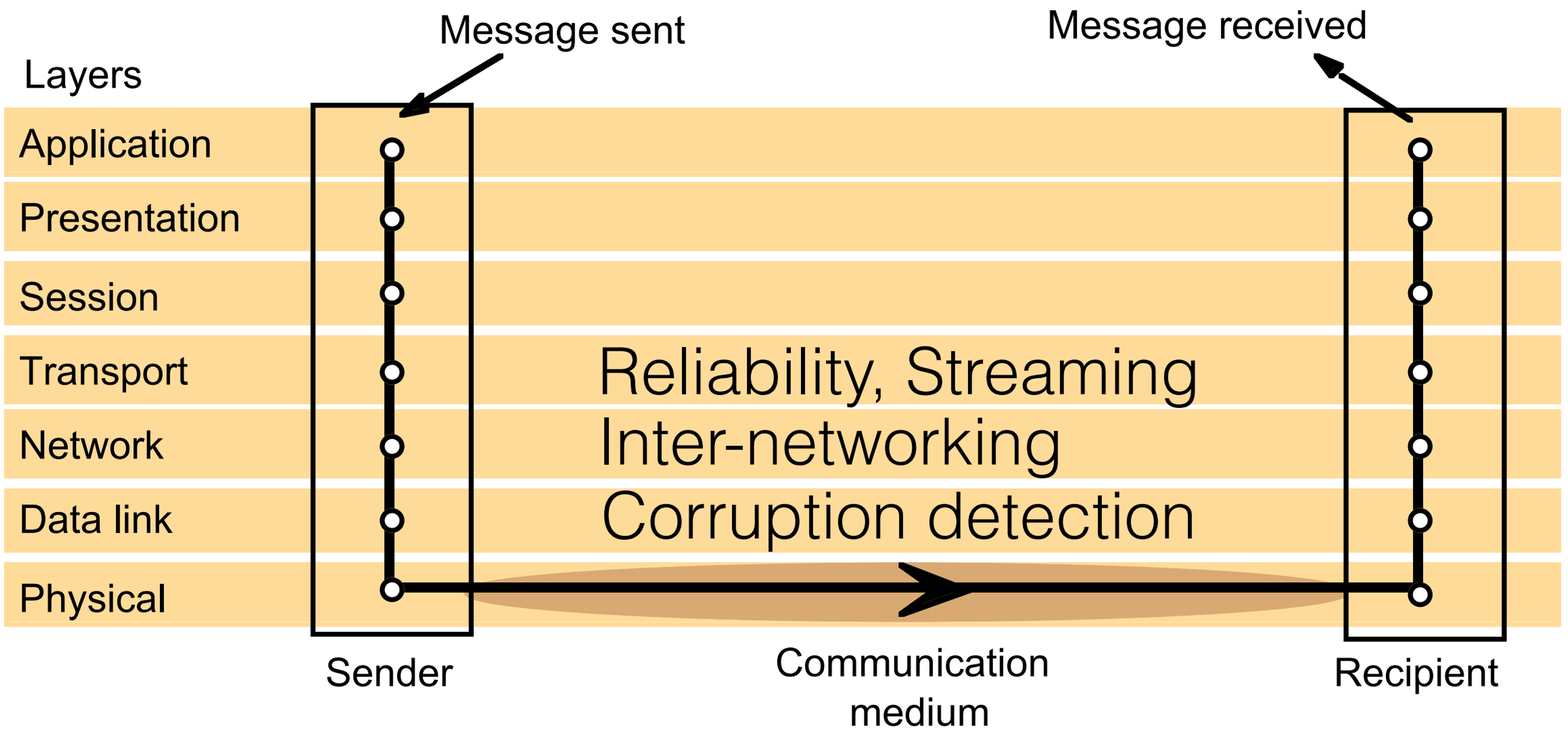
Terminology (iii)

- Processes adhere to *protocols*: specified sequences of message exchanges and data formats.

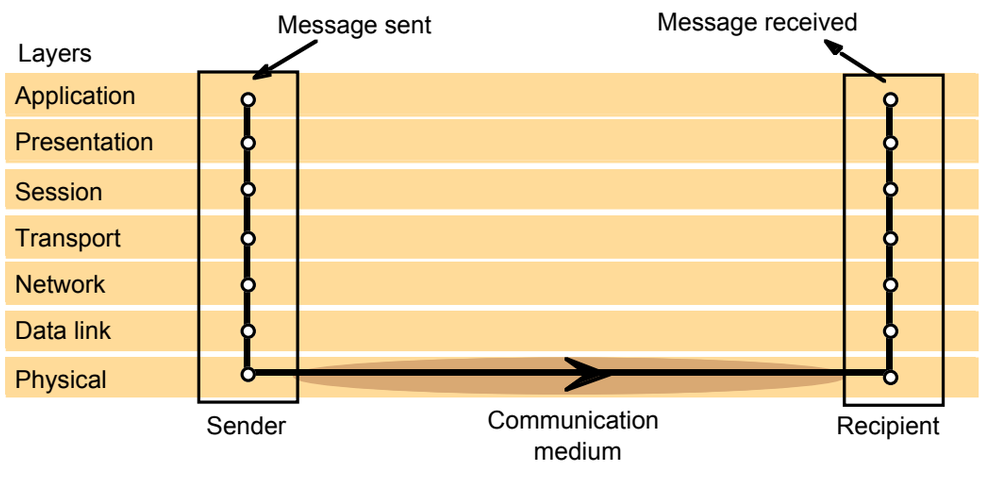
- E.g., TCP, UDP, SMTP, HTTP, …

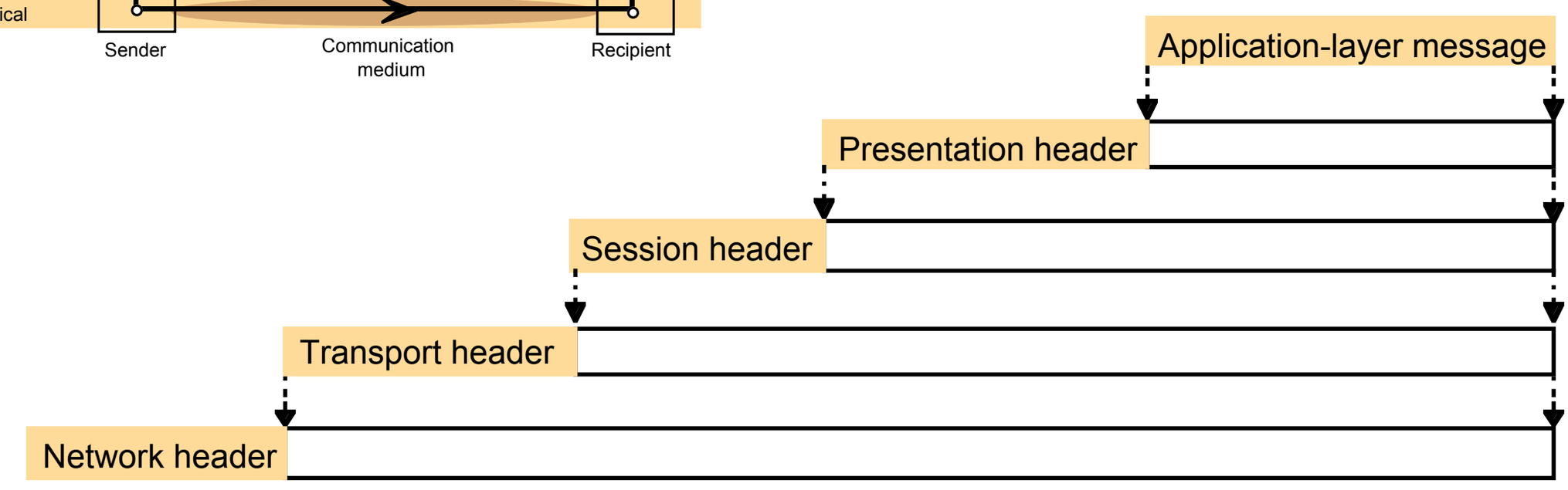
- Protocols are typically combined in *protocol stacks*, each layer protocol adding functionality.

Protocol layers



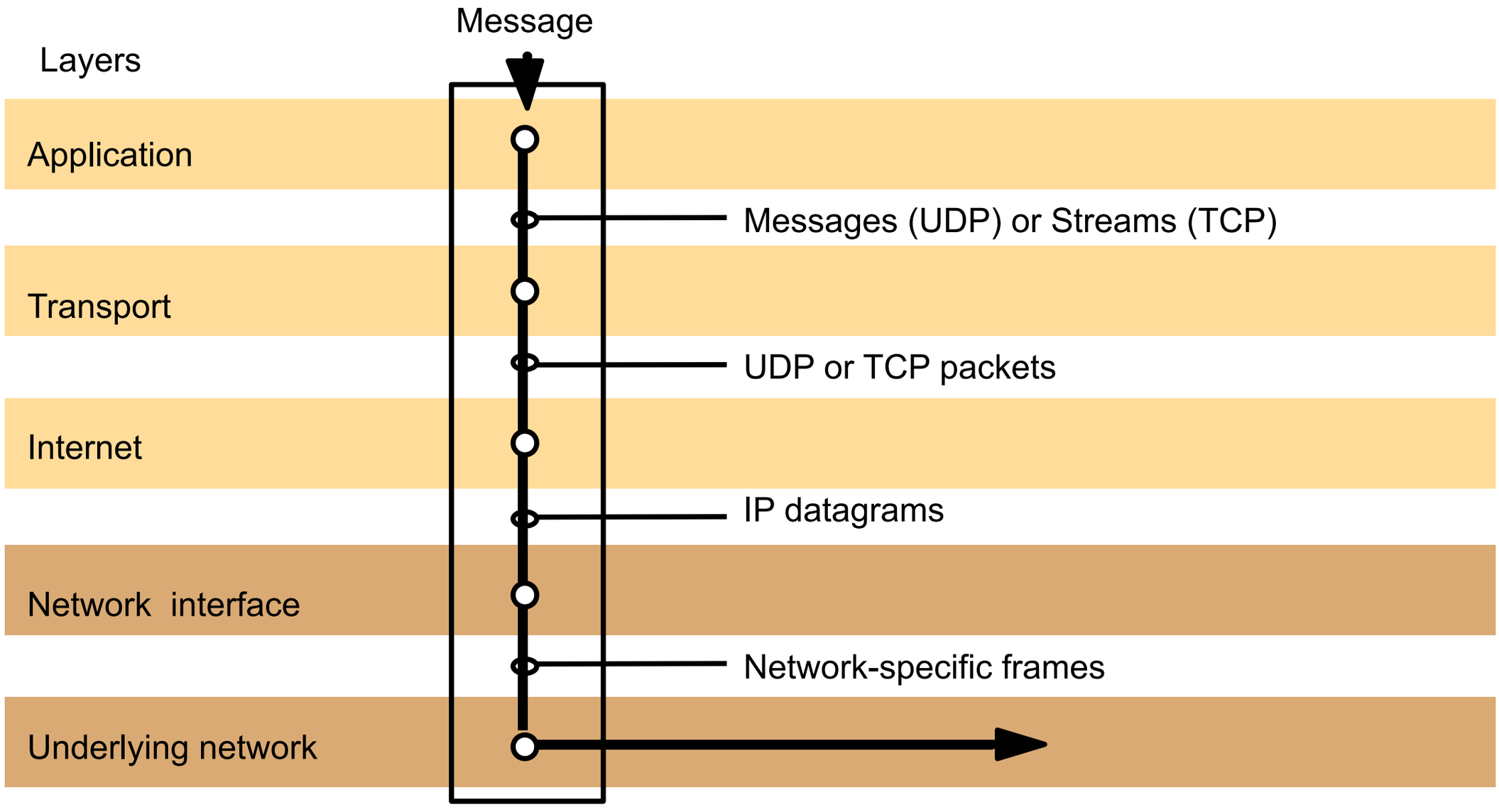
Protocol layers headers



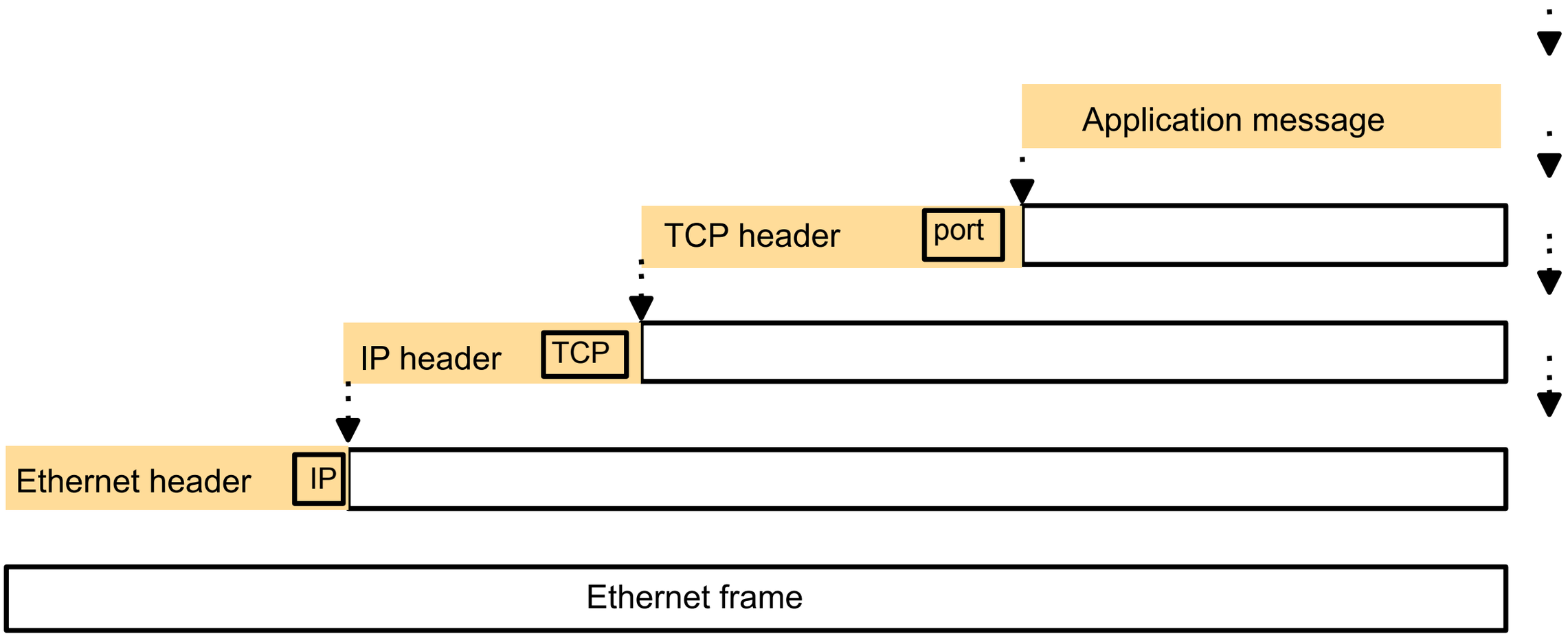


IP

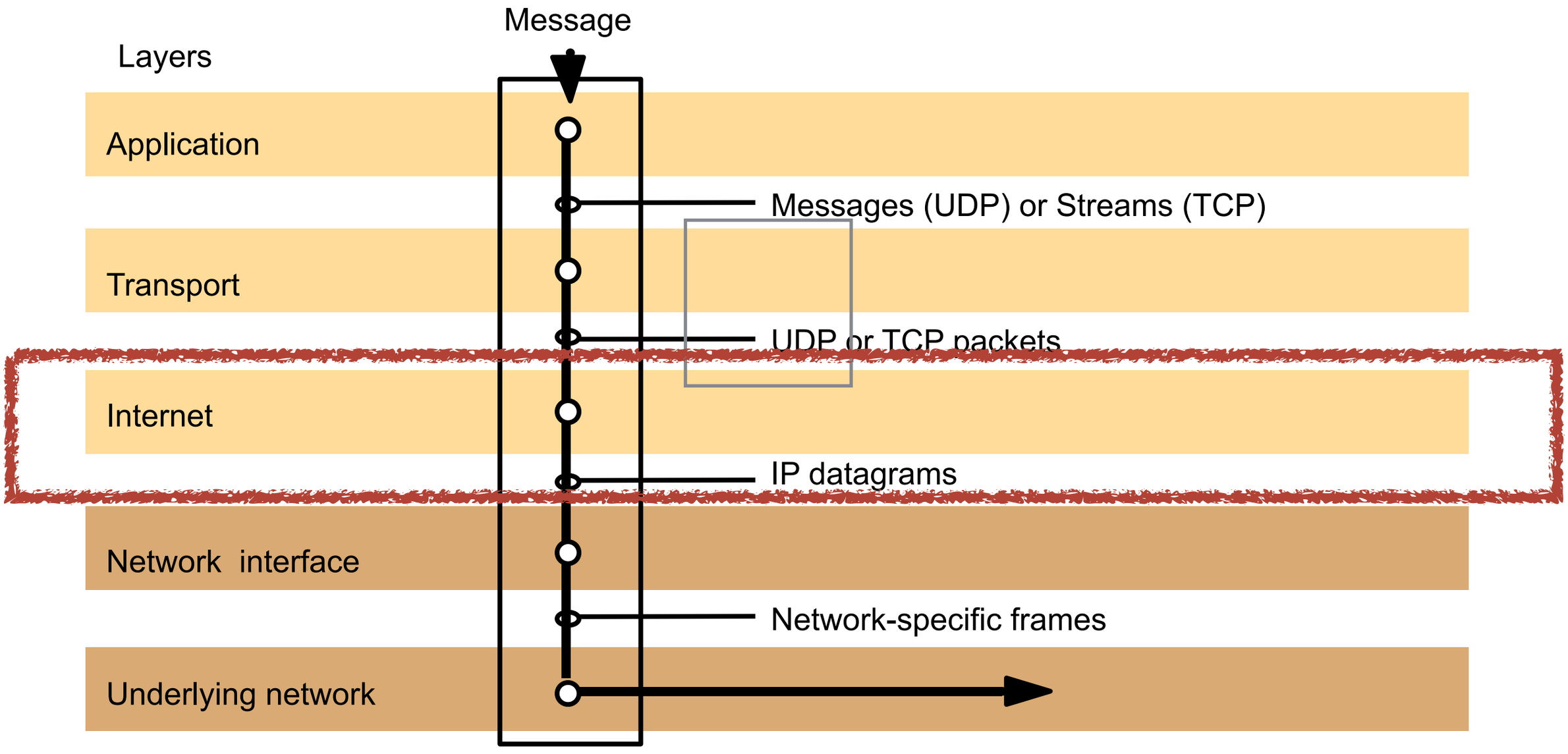
TCP/IP stack


Message
Layers
Application
Transport
Internet
Network interface
Underlying network
Messages (UDP) or Streams (TCP)
UDP or TCP packets
IP datagrams
Network-specific frames

TCP/IP stack headers



TCP/IP stack


Message
Layers
Application
Transport
Internet
Network interface
Underlying network
Messages (UDP) or Streams (TCP)
UDP or TCP packets
IP datagrams
Network-specific frames

IPv4

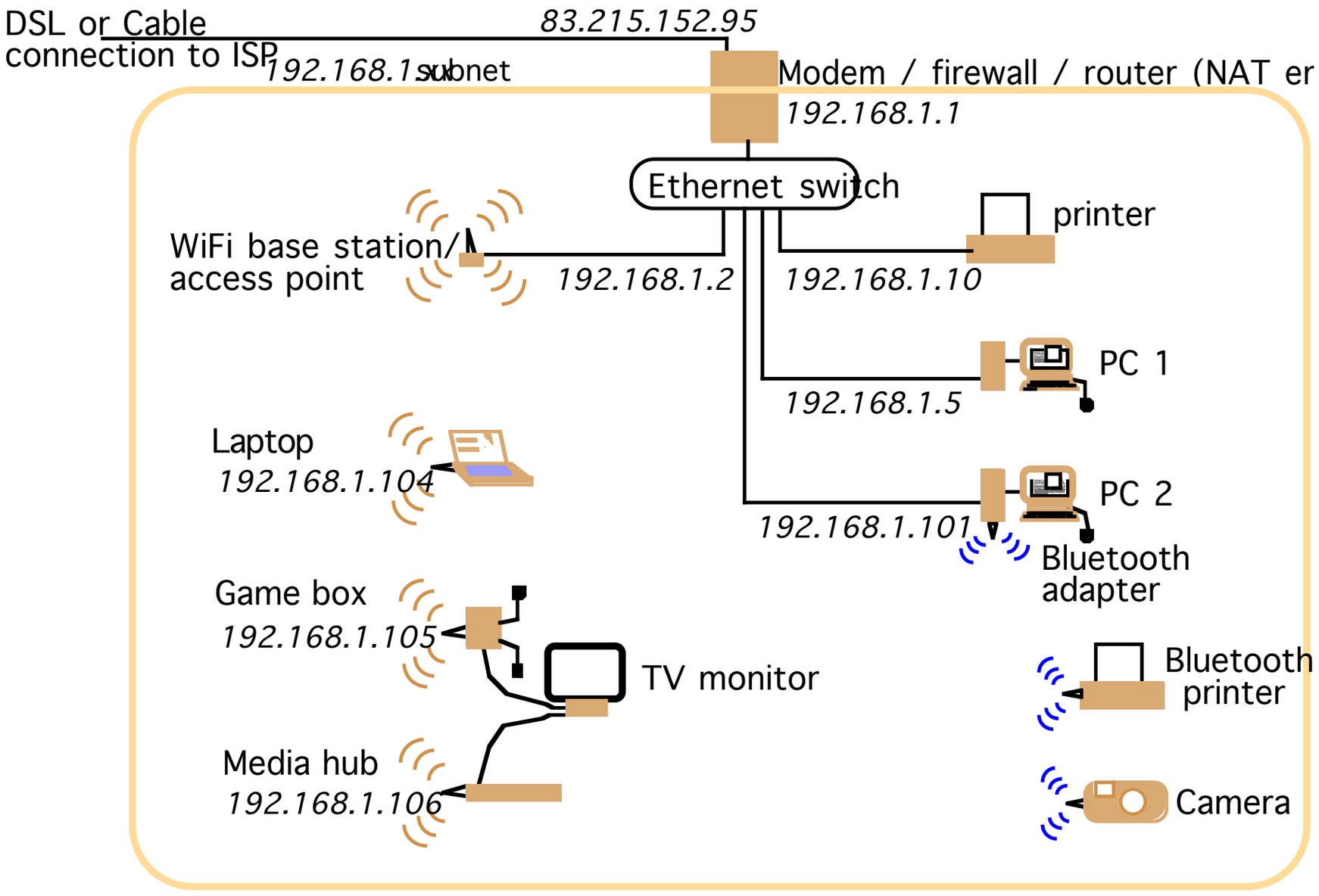
- Connection-less, unreliable, datagram protocol.
- Key functionality: Routing across distinct connected physical networks.
- IP-address: 4 byte logical address. E.g., 130.226.133.47
- CIDR: identifying IP ranges by IP-address + number of relevant bits in prefix: 130.226.132.0/23

IPv4 addresses

	octet 1	octet 2	octet 3		Range of addresses
	Network ID		Host ID		
Class A:	1 to 127	0 to 255	0 to 255	0 to 255	1.0.0.0 to 127.255.255.255
	Network ID		Host ID		
Class B:	128 to 191	0 to 255	0 to 255	0 to 255	128.0.0.0 to 191.255.255.255
	Network ID			Host ID	
Class C:	192 to 223	0 to 255	0 to 255	1 to 254	192.0.0.0 to 223.255.255.255
	Multicast address				
Class D (multicast):	224 to 239	0 to 255	0 to 255	1 to 254	224.0.0.0 to 239.255.255.255
Class E (reserved):	240 to 255	0 to 255	0 to 255	1 to 254	240.0.0.0 to 255.255.255.255

- 192.168.0.0/16, 10.0.0.0/8: private networks
- 127.0.0.0/8: loopback device
- 0.0.0.0/8: this host on this network
- 255.255.255.255/32: limited broadcast

Example network, NAT



MTU

- Maximum Transmission Unit
- Different physical links have different upper bounds on payload size
- IPv4 fragments packets on the fly

IPv4 Header

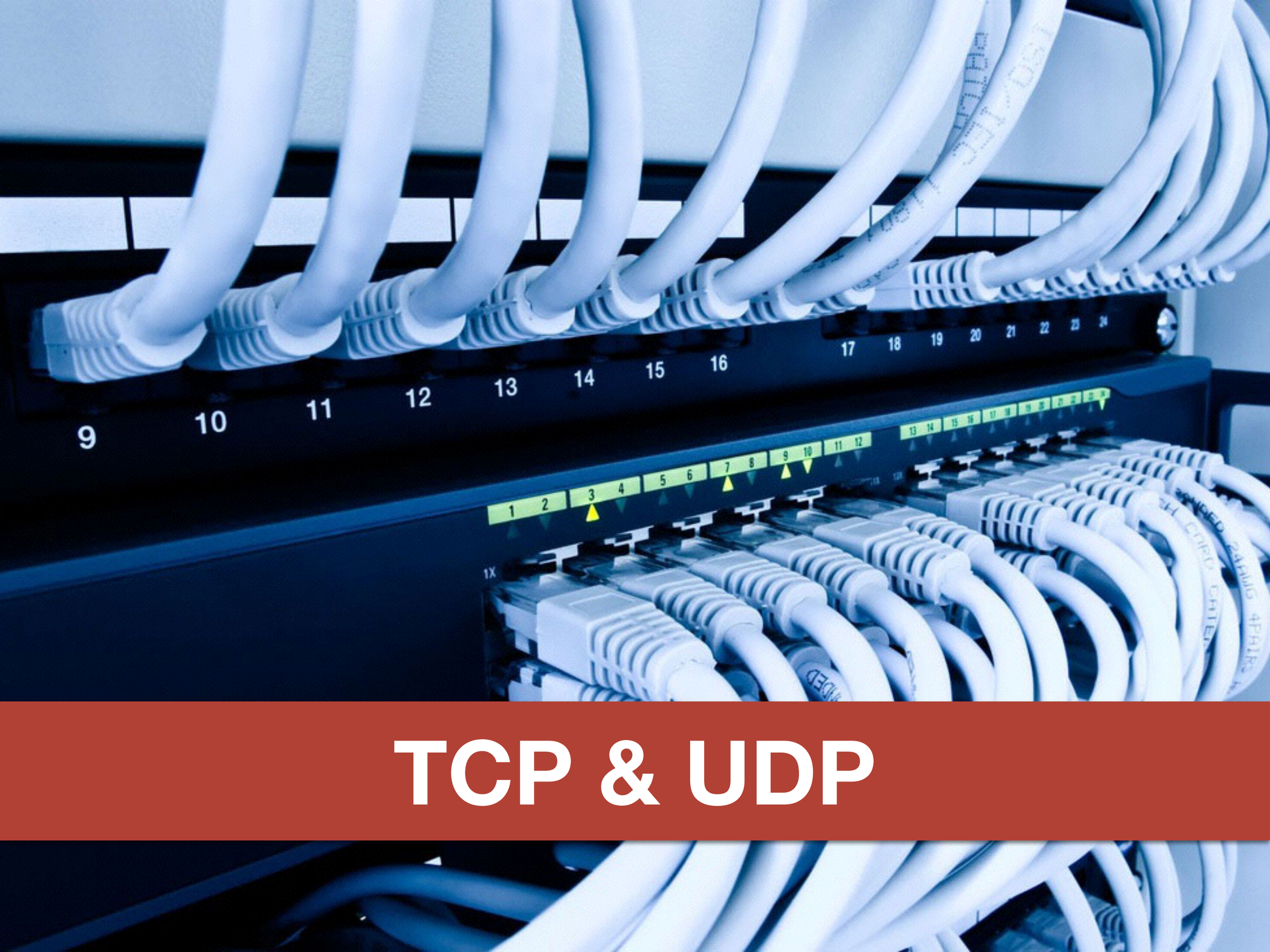
Offsets	Octet	0								1								2								3							
Octet	Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	0	Version				IHL				DSCP						ECN		Total Length															
4	32	Identification																Flags				Fragment Offset											
8	64	Time To Live								Protocol								Header Checksum															
12	96	Source IP Address																															
16	128	Destination IP Address																															
20	160	Options (if IHL > 5)																															

Practice

- Applications: TCP, UDP, SNMP, ICMP…
- Nobody likes numbers.
- Domain Names, e.g., “www.itu.dk” are translated to IP addresses by the “Domain Name System”.

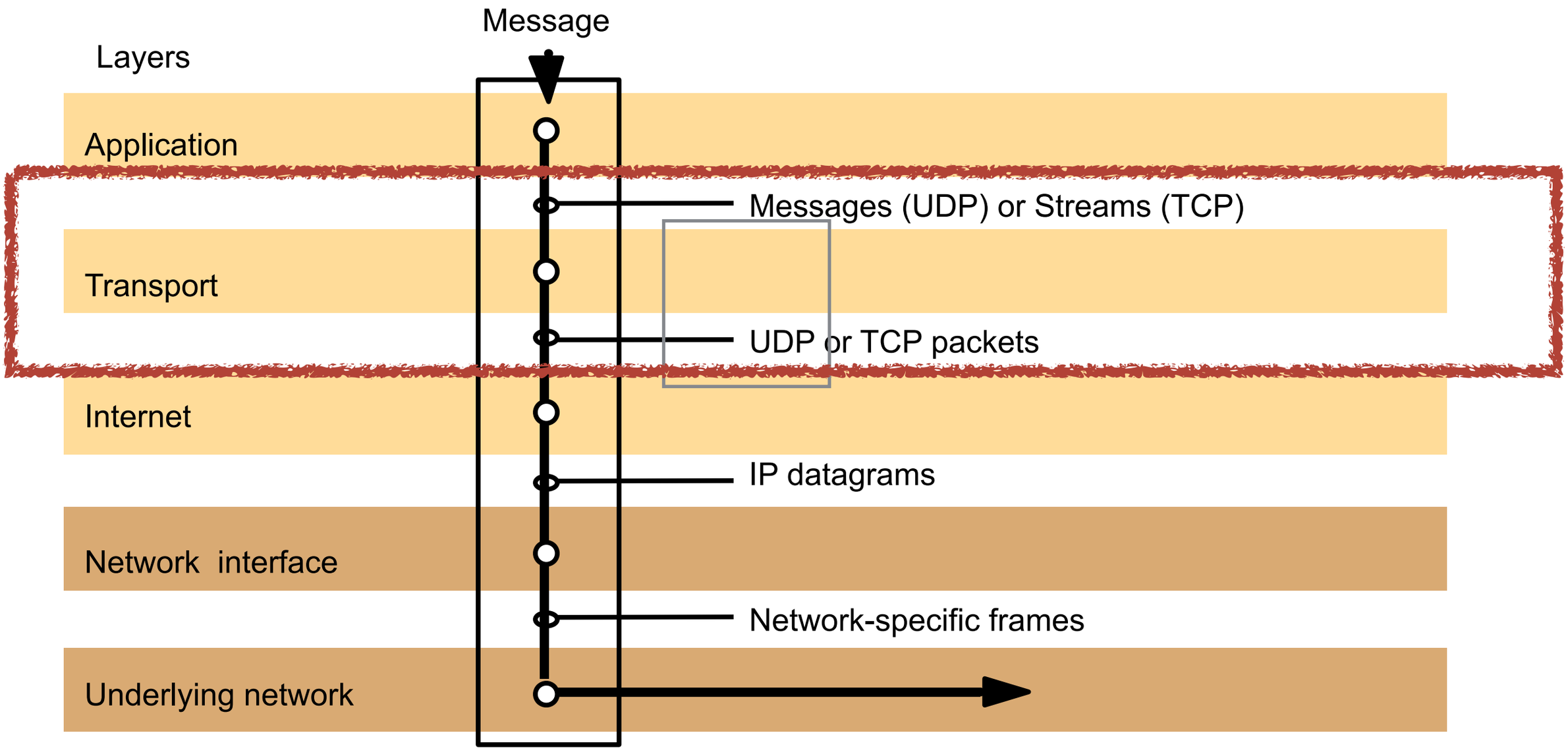
Practice

- Observe latency, connectivity using “ping”.
- Observe routes using “traceroute”.



TCP & UDP

TCP/IP stack

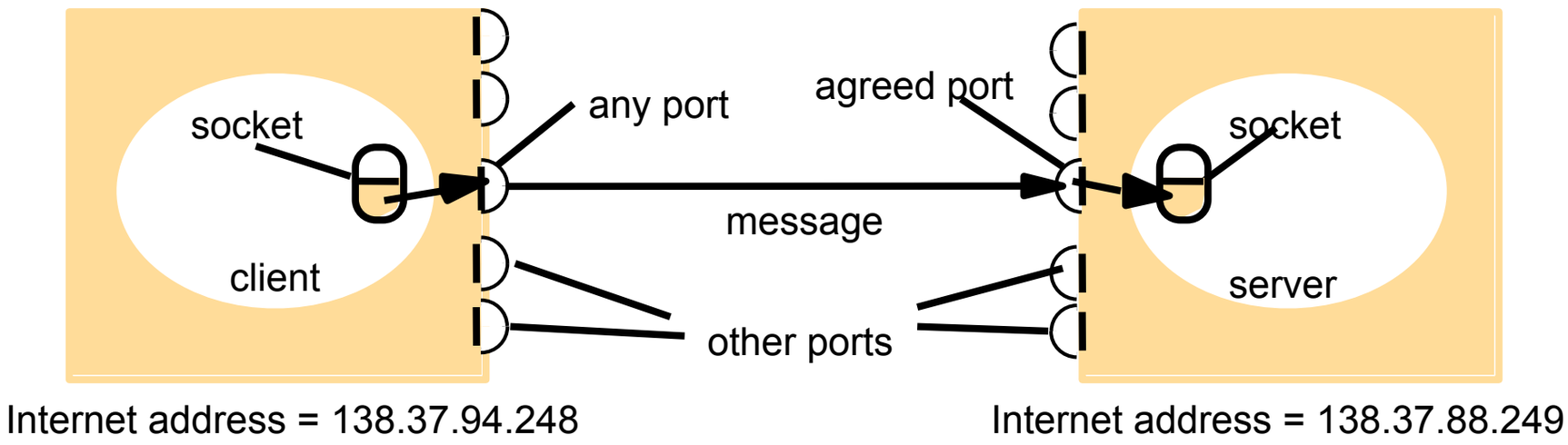

Message
Layers
Application
Transport
Internet
Network interface
Underlying network
Messages (UDP) or Streams (TCP)
UDP or TCP packets
IP datagrams
Network-specific frames

Ports in TCP/UDP

- Processes on a host are addressed by *ports.* A process is *bound* to a port.

- TCP/UDP protocols are used through a programming abstraction called a *socket*.

- Binding to ports 1-1023 require root privileges.



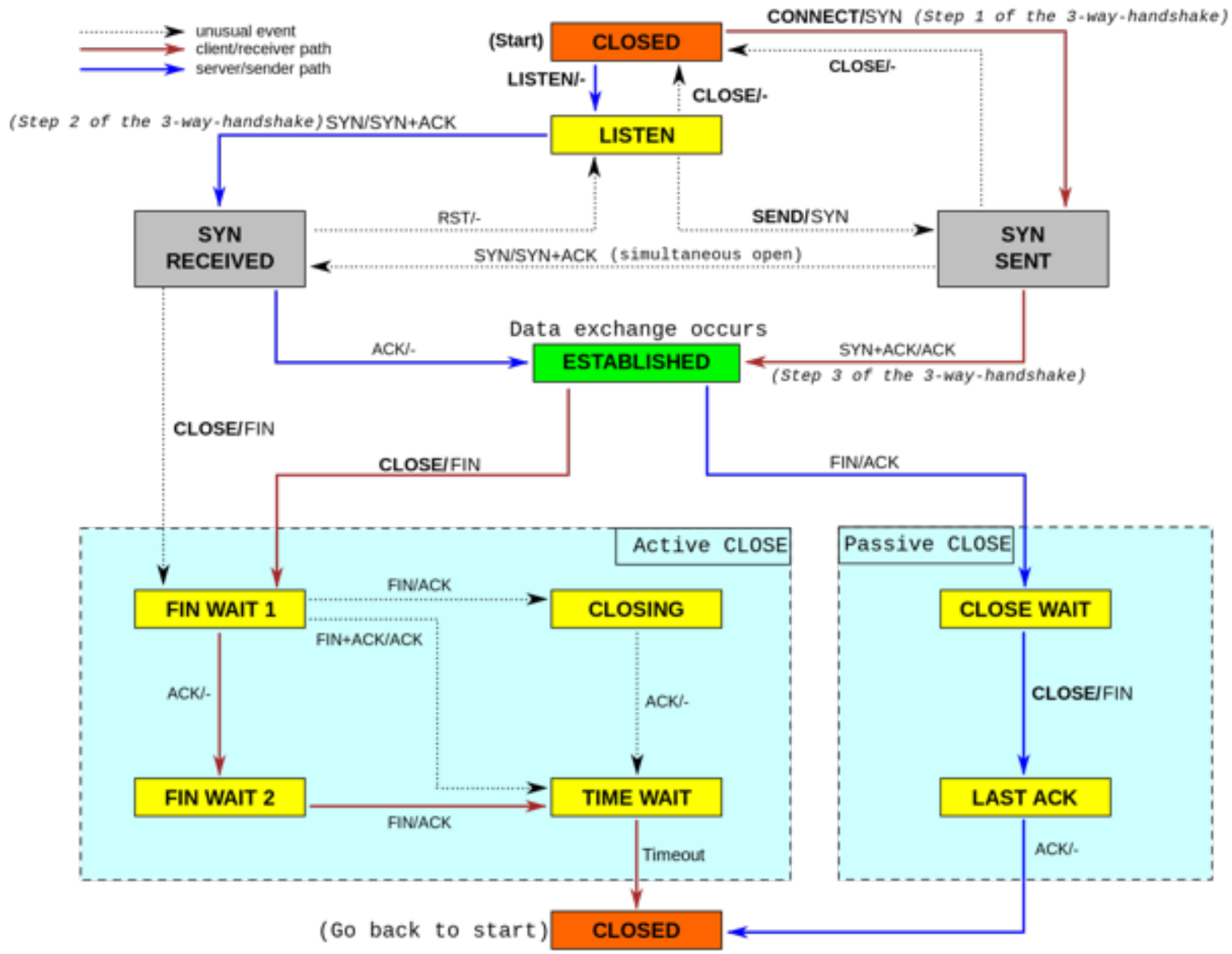
UDP

- Connection-less, unreliable, datagram-protocol.

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	Length																Checksum															

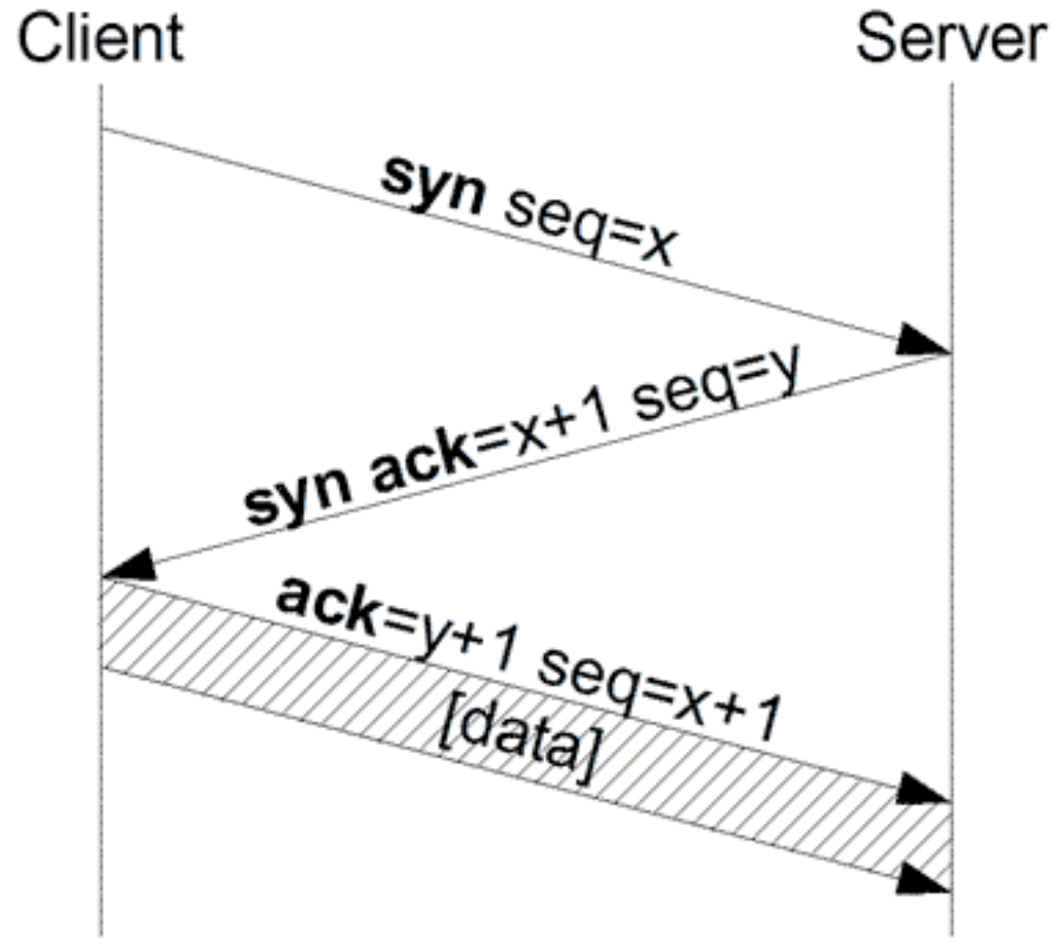
TCP

- Connection-oriented, reliable, streaming protocol.
- Specified as a fairly complex state machine:



Connection-oriented

- 3-way handshake
- A TCP connection is identified by all of (ip1, port1, ip2, port2).
- … so, a web-server at 130.226.133.47:80 can have multiple connections at that single port.



Reliable/streaming

- Sequence numbers, timeouts, acknowledgments, re-transmissions.

- Sliding window protocol, piggy-backing

TCP

- ACK flag: acknowledgments.
- SYN flag: synchronise sequence numbers
- RST flag: I'm not talking to you.

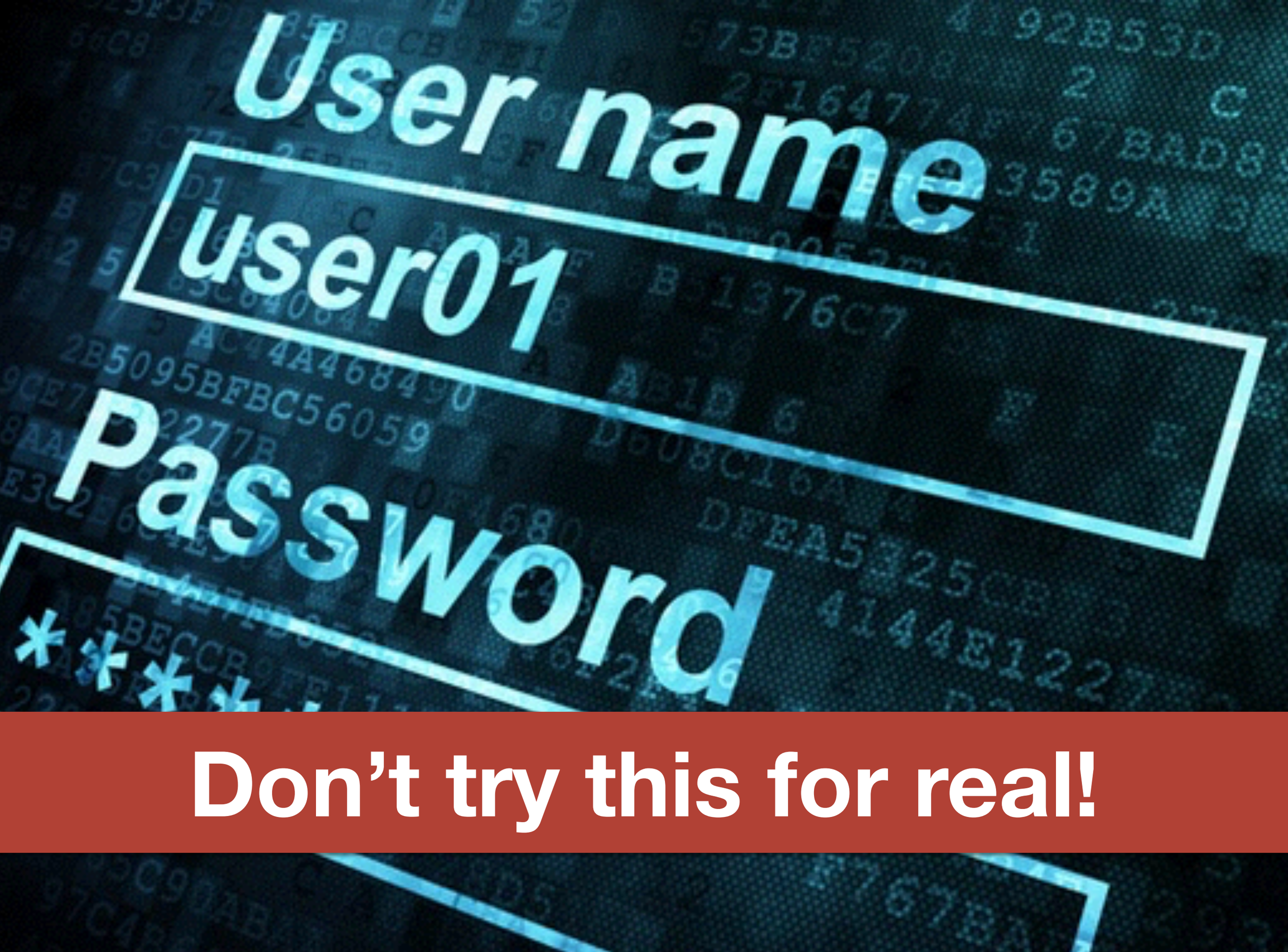
Offsets	Octet	0								1								2								3							
Octet	Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	0	Source port																Destination port															
4	32	Sequence number																															
8	64	Acknowledgment number (if ACK set)																															
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 ...	160 ...	Options (if *data offset* > 5. Padded at the end with "0" bytes if necessary.) ...																															

Practice

- Web-servers/HTTP uses TCP on port 80.
- E-mail/SMTP uses TCP on port 25.
- DNS lookups uses UDP on port 53.

Practice

- Observe traffic using “tcpdump”.
- Send/receive over UDP datagrams using “nc -u”.
- Connect/bind/communicate over TCP using “nc”.
- Portscanning, “nmap".

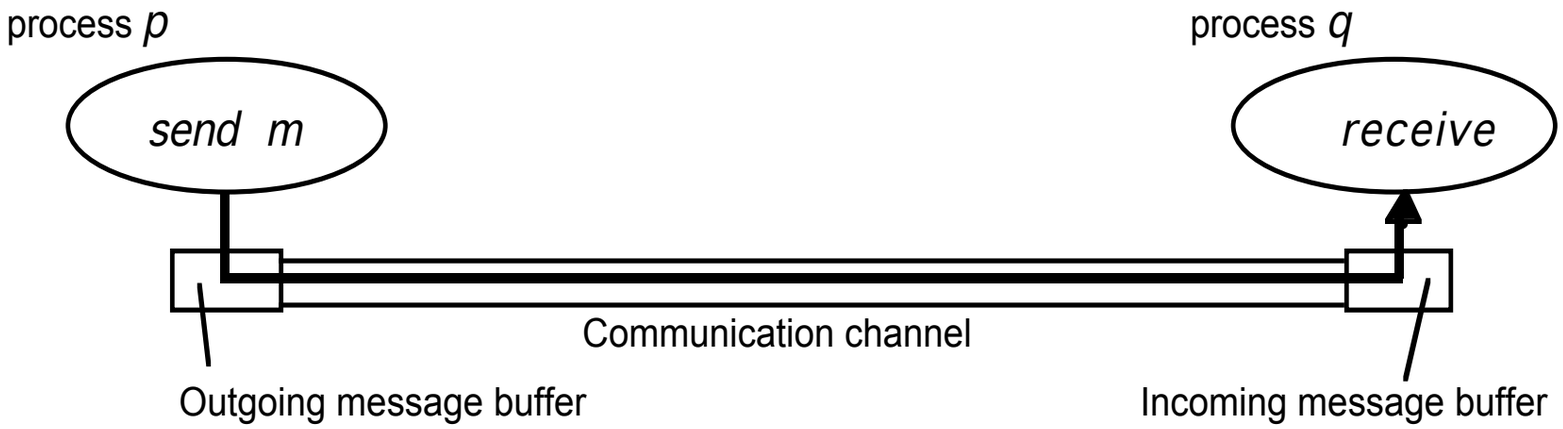


Don’t try this for real!

Summary

Foundations of Networking

- Terminology: Processes, channels, failures.



- Protocol layers, each protocol adding functionality

- Protocol layer headers.

TCP/IP

- IP: Routing, fragmentation.
  IP-address, CIDR, NAT.

- TCP/UDP: Ports, sockets.

- UDP: Datagrams. Adds checksum to IP.

- TCP: Reliability, congestion control, connections.
  3-way handshake, SYN.
  State machine.

Thank you!

- See learn-it for exercises etc.

- Questions?