computer networks

- **IP Addressing schemes**
  - how do they work?
  - how do we assign them?
  - packets
  - how data is travelling between computers
- **DNS**
  - translate machine names to IP addresses and vice versa
- **TCP**
- **UDP**
- **Routing**

Client $\underset{\text{answer}}{\overset{\text{request}}{\rightleftarrows}}$ Server

SOCKET → allows communication between processes running on different computers.

every host has an unique 32 bit IP address.

is the IP enough to identify a process? NO. Many processes can run on the same machine.
we need additional information: PORT.

↓

unsigned short - 2 bytes $(0 - 65535 = 2^{16} - 1)$

HTTP: 80
Mail: 25

Lect 6

PROTOCOL = agreement about communication

- format of the messages (representation + length)
- meaning of the messages
- rules of exchange → sequencing
- procedures for handling errors

why do we need protocols?

- hardware is low level
- problems that can occur:
  - bits corrupted or destroyed
    ↓
    flipped (ex: 0110 → 1010)
  - entire packets lost
  - packets are duplicated
  - packets delivered out of order
  - flow control

PROTOCOL HIERARCHIES

Networks: organized as stacks of layers => • reduce complexity
• each layer offers services to higher layers.

≈ data abstraction

Network architecture = a set of layers & protocols.

OSI Reference Model:

	7 Application	A	All
	6 Presentation	P	people
	5 Session	S	seem
TCP, UDP →	4 Transport	T	to
ip protocol happens → how packets are routed	3 Network	N	need
	2 Data link	D	data
	1 Physical	P	processing.

TCP/IP Model: only 4 layers:

4 Application
3 Transport
2 Network
1 Physical + Data link.

Lect 7

The INTERNET PROTOCOL (IP)

ICPM = Internet Control Message Protocol

DHCP = Dynamic Host Configuration Protocol

NAT = Network Address Translation

classes of Networks:

		network	host
A	0	1 byte	3 bytes
B	10	2 bytes	2 bytes
C	110	3 bytes	1 byte
D	1110	multicast	
E	1111	experimental	

1) Classfull IP adressing

Addressing

A - 1 byte network => 8 bits => $2^8 = 256$ possible networks
- 3 bytes hosts => 24 bits => $\approx 2^{24}$ IPs (first ip = Network Address
last ip = Broadcast Address)

- [1.0.0.0 — 127.255.255.255]

B - 2 bytes network => 16 bits => 2^{16} possible networks = 65536 Net
- 2 bytes hosts => 16 bits => $\approx 2^{16}$ possible IPs (-2)

- [128.0.0.0 — 191.255.255.255]

C - 3 bytes network => $\approx 2^{24}$ networks.
- 1 byte hosts => $\approx 2^8 = 256$ => 254 hosts

- [192.0.0.0 — 223.255.255.255]

D - [224.0.0.0 - 239.255.255.255]

E - [240.0.0.0 - $\underbrace{255.255.255.255}_{\text{usually left out}}$]

usually left out => UNIVERSAL BROADCAST ADDRESS
$\ne$
Network broadcast

without Classful IP addressing => 2^{32} total networks for a routing table.

with Classful IP addressing => $2^8 + 2^{16} + 2^{24}$, which is actually less bc. prefixes:
$\underbrace{2^7 + 2^{14} + 2^{21}}$ <<< 2^{32}

the largest routing table would have this many networks

2) CIDR - Classless InterDomain Routing

LECT 9

address format: a.b.c.d /x ⟶ x = how many bits represents the network part of the address.

the beginning address in each block % by the number of addresses in the block.

network address AND
net
network address => it is valid

IP address AND
Network mask
Network address

LECT 10

private addresses => not routed in Internet
- 10.0.0.0 – 10.255.255.255
- 172.16.0.0 – 172.31.255.255
- 192.168.0.0 – 192.168.255.255

Provide the first valid IP and the broadcast for:

172.17.89.200 / 19

mask: /19 => 32 - 19 = 13 => 2^{13} hosts

172.17.89.200
255.255.224.0
172.17.64.0

89 = 0101 1001 AND
224 = 1110 0000
0100 0000

=> Network Address: 172.17.64.0

First valid IP: 172.17.64.1

Broadcast: To find it, we perform "OR" with "NOT" mask.

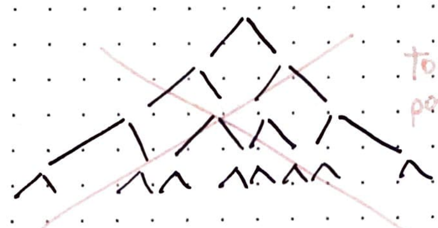
172.17.64.0 OR
0.0.31.255
172.17.95.255

0100 0000 OR
0001 1111
$\underbrace{0101 1111}_{95}$

lect 11

- subnet 203.10.93.0 / 24 => 30 subnets, is 203.10.93.30 a valid host ip after subnetting?

256 => 30



too many possibilities

203.10.93.30 – could this be a subnet?

0001.1110 => no mask we could add for this to be a network.

– could this be a broadcast?

all broadcasts should finish before a power of 2 (and therefore be odd). which means broadcast + 1 = power of 2

=> 30 + 1 = 31 ≠ power of 2 => it is not a broadcast
it is not odd

=> it will always be a host!

- 137.25.29.0 / 255.255.254.0 (8 . 8 . 7 . 0)   max # of subnets? max #. hosts / subnet?

max # of subnets => 137.25.29.0 / 23 => 9 bits hosts => $2^9 = 512$
smallest possible subnetwork: 4 hosts
} 512 : 4 = 128 subnets

max # hosts / subnet => do not subnet at all => 512 - 2 = 510 hosts

- subnet 172.26.0.0 / 16 in 1390 subnets of 30 hosts each. what is the net mask?

mask /16 => 2^{16} ips   (1390 ↓ not a power of 2)

a) 30 hosts / each => $2^{16} : 2^5 = 2^{11}$ = 2048 (> 1390) with 30 hosts / each

netmask: /27   255.255.255.224

b) subnet into exactly: 1390 => not equally sized => multiple netmasks

split to 1024 → still need 366 more

we split 366 out of those => 1024 – 366 = 658 : ~~/22~~ 26

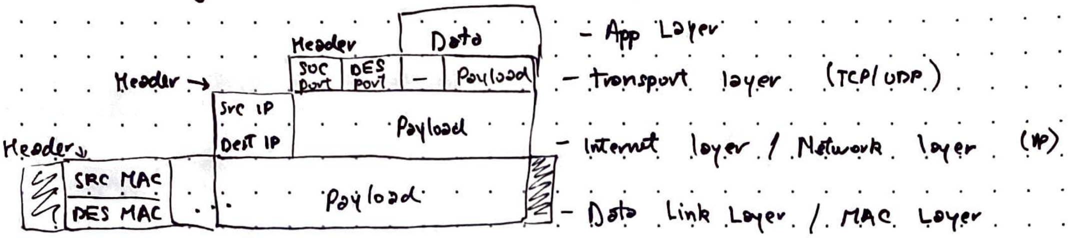
366 x 2 = 732 : ~~/23~~ 27

~~$2^{16} : 2^x$~~ = 2^{10} => 10 divisions (splits)
= 1024

=> x = 6, we split ~~6~~ 10 times
=> the mask increases by 6 => 16 + 6 = ~~22~~ 26

BRUNNEN

IP Datagram



ARP = Address Resolution Protocol

lect 12

NAT = Network Address Translation

ICMP = Internet Control Messaging Protocol
↳ signaling protocol

- ping

UDP = User Datagram Protocol
- functionality ≈ IP
  transport

- data also protected by checksum!

• unreliable
• not ordered
• lightweight
• datagrams
• no congestion control

TCP = Transmission Control Protocol

• ordered
• retransmission of lost packets
• error-free data transfer
• flow control
• congestion control