Introduction to communication networks

Lecture 1 - Basics

Jens Myrup Pedersen
Associate Professor
Aalborg University
jens@es.aau.dk

COLL

norden
NordForsk

Figures by Matija and Kardi (mst,kba@es.aau.dk), and to limited extend Andrew S. Tananbum, Computer Networks.

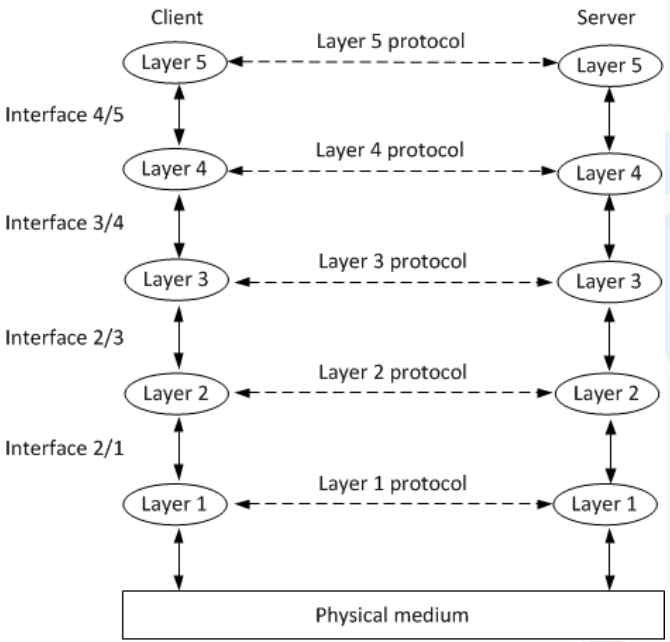
The purpose of this lecture is for you to

- Understand the principles of layered packet-based network architectures.
- Obtain an overview of the TCP/IP protocol stack and the OSI reference model, and to understand the differences between them.
- Get an understanding of connectionless and connection-oriented protocols.
- Get to know the units used for network speeds and bandwidth.

It is highly recommended for you to take notes during the lecture, especially focusing on these learning points.

First a little about procotol stacks:

- Example of a protocol stack.
- Each layer has its own functionality.
- Each layer offer ”services” to the layer above.
- Transparency: A layer don’t see what is underneath it, but virtually speaks to the corresponding layer at the other machine.



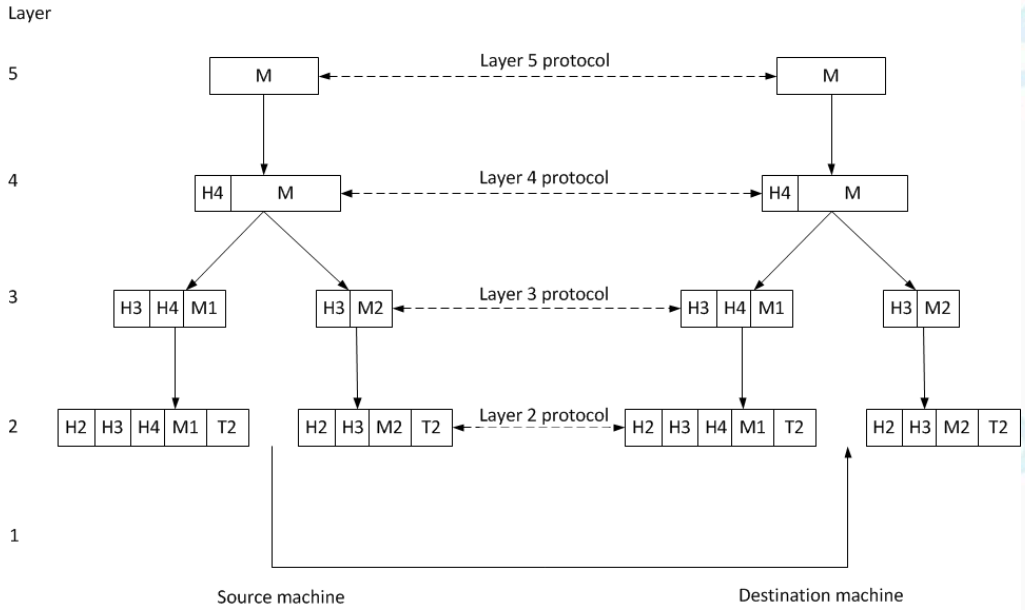
First a little about procotol stacks:

- Typical one protocol per layer
- Between each layer there is a well defined functionality
- The layered architectures allows for:
  - A modular design
  - Layers can be re-used for different protocols
  - Easier to change implementations (or add new protocols).
  - Easier to develop applications
  - Independency of the physical medium

So, how does it look from a message point of view?

- A message is sent from one application to another. This is done as a packet: A sequence of bits.
- In the different layers, diffferent kinds of control information (headers and/or tails) are added. This control information can for example be addresses or different kinds of protocol information.
- There can also be limitations to packet sizes, so packets may be split up.
- This also means that in addition to the mssage being sent, each layer is adding additional information (or overhead): How much is added on each layer depends on the protocols being used.

Let us look at an example



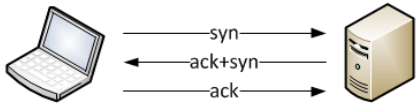
Connection-oriented or connection-less

- Important: connection-oriented and connection-less protocols.
- The simplest is a connection-less protocol. Here packets are simply sent out on the network, usually marked with an address, and we hope they arrive. However, there are no guarantees that all packets will arrive, and the sender has no way of telling whether this has happened.
- For a connection-oriented protocol, a connection is first set up, then packets are send, and eventually the connection is torn down. Acknowledgements can be used to assure the arrival of all packets, and the protocol can have mechanisms for re-transmissions and re-orderings of packets that don't arrive or arrive in the wrong order.

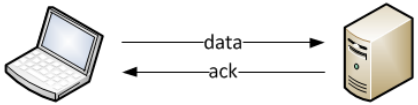
Connection-oriented or connection-less

Connection Oriented
(TCP)

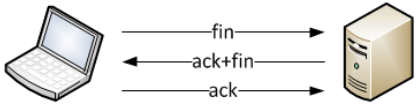
1. Establish a connection



2. Send data

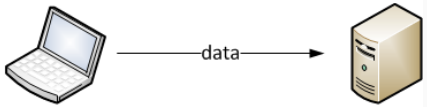


3. Close the connection



Connectionless
(UDP)

Send data and forget



Two important models

- The OSI reference model, developed by ISO
- The TCP/IP Protocol stack, used in the Internet

The OSI reference model

- Open Systems Interconnection, suggested by ISO
- Designed as it “should be done”.
- The reference-model is made prior to the standards for each protocol.
- 7 layers.

The OSI reference model

L7: Application layer
L6: Presentation layer
L5: Session layer
L4: Transport layer
L3: Network layer
L2: Data link layer
L1: Physical layer

In the following, let's have a look at the functionality of the different layers. We focus on the layers which are also used in the TCP/IP stack.

OSI: The physical layer

- Defines the physical transmission, e.g. voltage, frequency etc.
- For example: Electricity, laser, radio waves
- Set-up and tear-down of connections
- Offering services for the data link layer.

OSI: The data link layer

- The data link layer receives data from the network layer, which is then broken into smaller frames. The size of these frames depends on a number of issues, for example the error rate.
- At the physical layer, transmission errors can occur of many different reasons. It can be because of disturbance due to noise, weather conditions (especially for wireless), worn equipment and many other things. This error control is can be done in the data link layer.
- The data link layer takes care of physical addressing, error control, access control, and Medium Access Control (MAC).
- The data link layer can also contains mechanisms such as sliding windows, for controlling the sending speed, and thus avoid congestion.
- Examples: Ethernet, Token ring, PPP, HDLC, Frame Relay, ISDN, ATM.

OSI: The network layer

- Most important functionality is routing
- Quality of Service can also be handled here, for example by giving certain packets a higher priority. Elements of congestion control can also be done.
- Gateway-functionality between different networks is also typically handled at the network layer. Yet another example of the strength of layered protocols.
- Many networks use dynamic hop-by-hop routing, but source routing and static routing can also be used. Static routing is usually used only in very small or very reliable network (certain multi processor systems).
- Examples: IP, ICMP, IGMP, X.25, CLNP, ARP, RARP, BGP, OSPF, RIP, IPX, DDP

OSI: The transport layer

- The layer between applications and network layer. Offer end-to-end communication.
- Split and join packets before they are sent to and from the network layer.
- Decides which connection should be available for the upper layers:
  - Connection set-up and tear-down
  - Error correction can be done here
  - Acknowledgements and retransmissions
  - Handling out-of-order packet arrivals
- Some protocols such as TCP also have traffic/flow control
- Examples: TCP, UDP, RTP, SCTP, SPX, ATP

OSI: The session layer

- Handles sessions

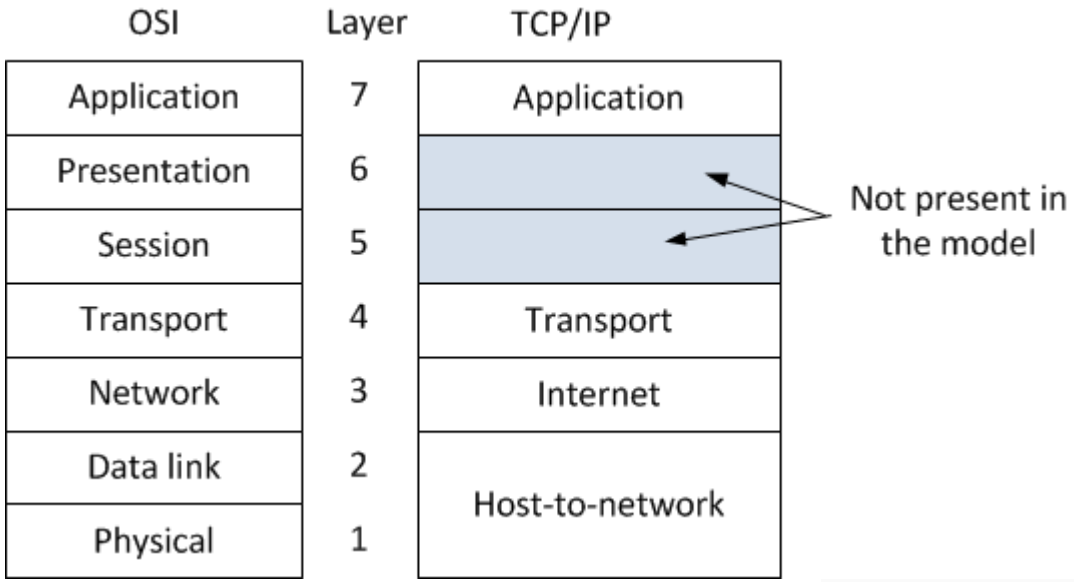
OSI: The presentation layer

- Handles data structures

OSI: The application layer

- The upper layer, which can be used directly by the user applications (such as web browsers, email clients, ssh clients etc.)
- Examples: HTTP, SMTP, SNMP, FTP, Telnet, Ssh and Scp, NFS, RTSP

OSI vs. TCP/IP


OSI
Layer
TCP/IP
Application
7
Application
Presentation
6
Session
5
Transport
4
Transport
Network
3
Internet
Data link
2
Host-to-network
Physical
1
Not present in the model

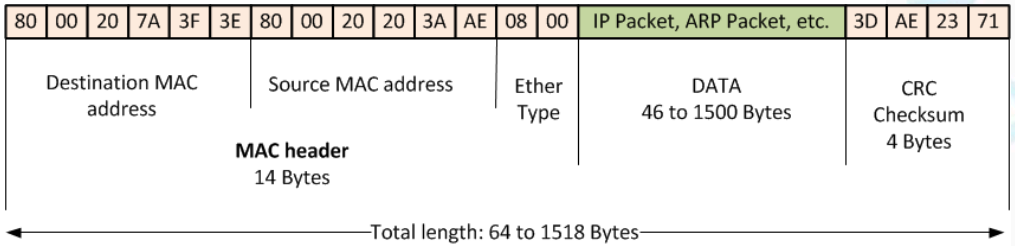
TCP/IP

- The protocols of layer 2,3 and 4 will be treated more in-depth in the coming lectures.

- In the following, a few differences to the OSI model will be highlighted, and some examples of protocol design given.

TCP/IP – Host-to-Network

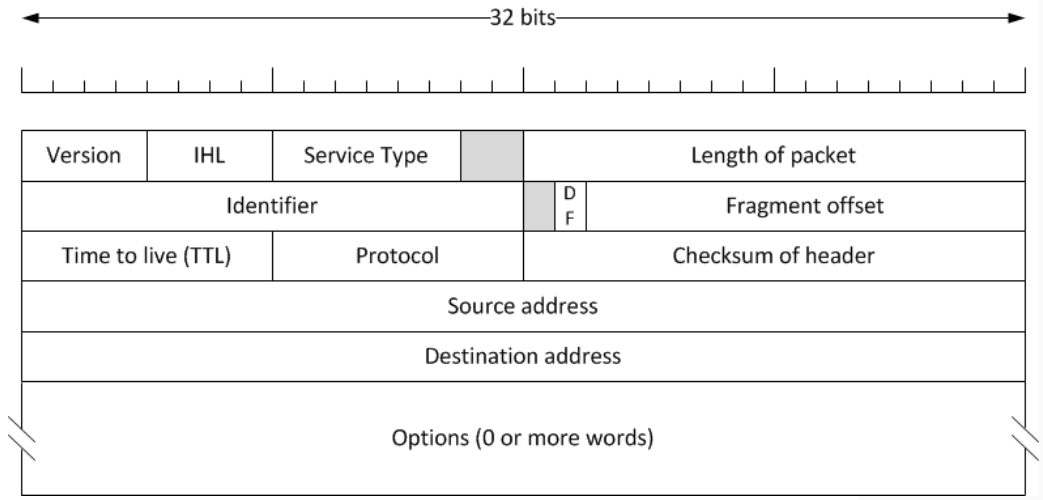
- Corresponds to layer 1+2 from the OSI model
- Among the data link layers, Ethernet is probably the most known. But it is also possible to run IP over ATM.
- Example: Ethernet.



Ethernet Frame format, type II

TCP/IP – Internet layer

- Corresponds to the network layer of the OSI model
- IPv4 is probably the best known.



TCP/IP – Transport layer

- Corresponds to the transport layer of the OSI model
- Most used/known protocols:
  - TCP: Reliable, connection oriented
  - UDP: Unreliable, connectionless
  - These protocols will be treated in more details in coming lectures.

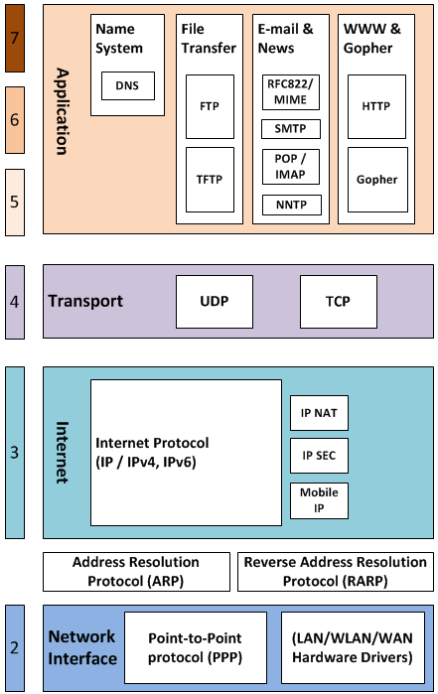
TCP/IP – Application layer

- Corresponds to session, presentation and application layer in one.
- HTTP, FTP, mail, news etc.
- Example: HTTP

```
GET / HTTP/1.1
Host: www.skat.dk
Connection: keep-alive
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.22 (KHTML, like Gecko)
Chrome/25.0.1364.160 Safari/537.22
Accept-Encoding: gzip,deflate,sdch
Accept-Language: en-US,en;q=0.8
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.3
Cookie: ASP.NET_SessionId=nwzotfy4pmpazvy4a01rkz55;
```

```
HTTP/1.1 200 OK
Cache-Control: private
Content-Length: 30266
Content-Type: text/html; charset=iso-8859-1
X-Powered-By: ASP.NET
Date: Sun, 10 Mar 2013 22:02:55 GMT
```

TCP/IP – an overview of protocols



To sum of this lecture – just a few facts

- One bit is a single ”0” or ”1”
- 1 byte is 8 bit
- 1Gbit/s = 1000Mbit/s = 1.000.000 Kbit/s = 1.000.000.000 bit/s

- If written as e.g. 1MB/s, this usually means one megabyte per second, but should be avoided to avoid confusion..

- So: ”Bandwidth” or ”throughput” is often measured as bit/s (or Gbit/s, Mbit/s, Kbit/s).

- When sending data from ”A” to ”B”, the time it takes to transmit a message depends on the bandwidth as well as the latency.

End of lecture

Please don't hesitate to send questions or comments!

Jens Myrup Pedersen
Associate Professor
Aalborg University
jens@es.aau.dk