



AARHUS UNIVERSITET

Software Engineering and Architecture

Networking 101



- Networking - not a curriculum issue in SWEA...

- But...
 - You see it everywhere

 - And you need some 'Network for dummies' for our Broker...

A Network

- ... in CS is basically two or more machines connected by electrical wires that allows to send signals between the machines...

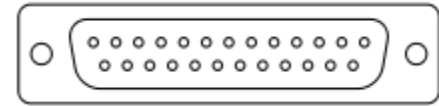




AARHUS UNIVERSITET

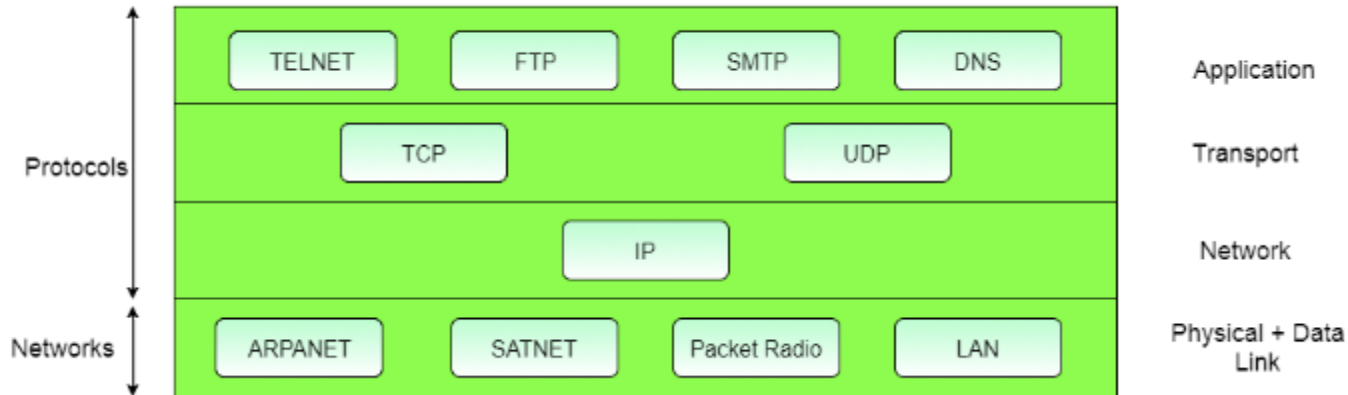
A Network

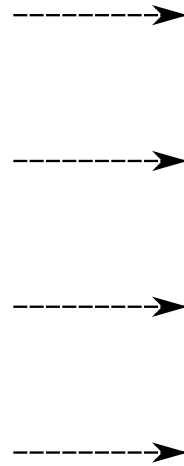
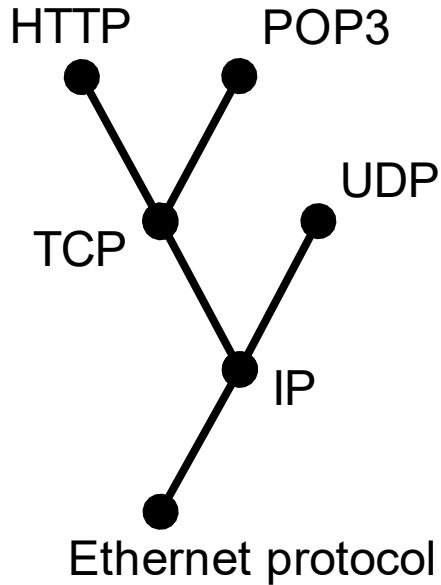
- My first exposure: RS232 on Z80 CPUs



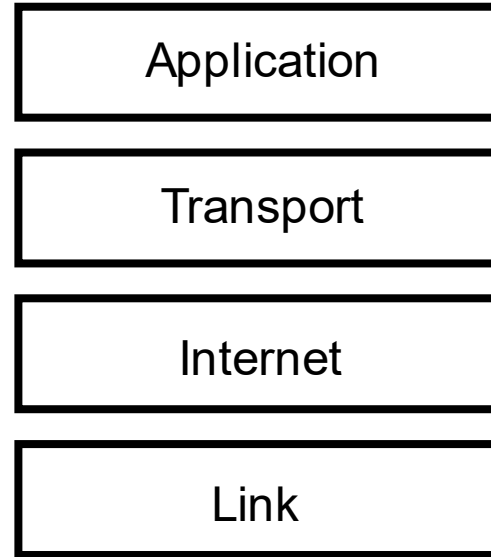
- Today's web: TCP/IP over Ethernet

- Transmission Control Protocol and Internet Protocol
 - By the US Department of Defence (DARPA)
- Key Idea
 - Segment transmission into Packets ("Datagrams")
 - Layered architecture, each with specific responsibilities (roles!)





TCP/IP - model



By Jsoon eu (talk) - I (Jsoon eu (talk)) created this work entirely by myself., CC BY-SA 3.0, <https://en.wikipedia.org/w/index.php?curid=29962617>

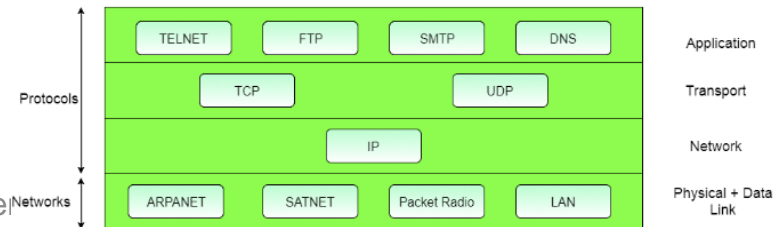
- Another but similar model

OSI Model			
	Layer	Protocol data unit (PDU)	Function ^[3]
Host layers	7. Application	Data	High-level APIs, including resource sharing, remote file access
	6. Presentation		Translation of data between a networking service and an application; including character encoding, data compression and encryption/decryption
	5. Session		Managing communication sessions, i.e. continuous exchange of information in the form of multiple back-and-forth transmissions between two nodes
	4. Transport	Segment, Datagram	Reliable transmission of data segments between points on a network, including segmentation, acknowledgement and multiplexing
Media layers	3. Network	Packet	Structuring and managing a multi-node network, including addressing, routing and traffic control
	2. Data link	Frame	Reliable transmission of data frames between two nodes connected by a physical layer
	1. Physical	Symbol	Transmission and reception of raw bit streams over a physical medium



TCP/IP Layers

- Transport layer
 - TCP Reliable, ordered, error-checked data delivery
 - Transmission Control Protocol
- Network / Internet Layer
 - IP Relaying datagrams across networks
 - Internet protocol
- Physical + Data Link Layer
 - 802.3 Ethernet Hardware and cables
 - 802.11 WiFi Cables gone





AARHUS UNIVERSITET

Internet Protocol

IP: Send datagram

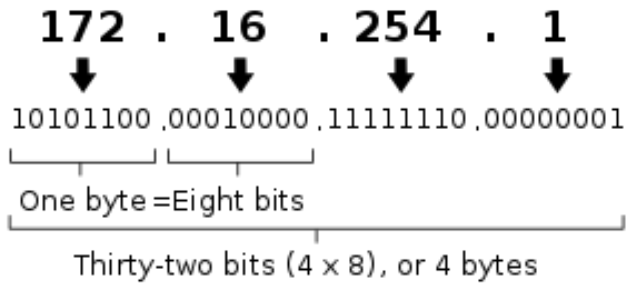
- Defines the terminology that we use and it pops up even at the software level
- *Every* computer on the network has an *address*
 - Type 'ifconfig'/'ipconfig' to find yours

```
d:\work\teaching\SWEA-E17\web>ipconfig
Windows IP Configuration

Ethernet adapter Ethernet:

    Connection-specific DNS Suffix  . : cs.au.dk
    Link-local IPv6 Address . . . . . : fe80::d3:4e4c:a5a8:553a%11
    IPv4 Address. . . . .             : 10.11.96.31
    Subnet Mask . . . . .             : 255.255.192.0
    Default Gateway . . . . .         : 10.11.64.2
```

An IPv4 address (dotted-decimal notation)



- Some ranges are reserved
 - 10.*.*., 172.16.*.*., 192.168.*.* are private networks
 - 127.0.0.1 is *localhost* = myself



IP and Ports

AARHUS UNIVERSITET

- So given an IP (like 91.221.196.224) you uniquely identify a computer
- The OS of that computer expose 64K **ports**
 - Also predefined port numbers
 - 7: echo ('ping')
 - 20: ftp
 - 22: ssh
 - **80: HTTP**
- Thus
 - 91.221.196.224:80 is the HTTP port of a specific computer
 - As port 80 is active it is probably a web server

- On Linux, all ports below 1024 are reserved for 'root'
- Above that, it is 'free game' to assign/use a port, but you may interfere with other programs that have picked one...

4000	Yes	Yes	<i>Diablo II</i> game	Unofficial
5000–5500	No	Yes	<i>League of Legends</i> , a multiplayer online battle arena video game ^[188]	Unofficial



Datagram

- So, for node A and node B to communicate some data
 - Say, a request for a web page, and the server reply
- A creates a request
 - N datagrams (the data segmented into packet size)
 - Each datagram contains
 - Part i of the full data
 - Destination IP address Who is to receive
 - Source IP address Who should have the reply
- B creates a reply
 - Of course the same 😊



AARHUS UNIVERSITET

Domain Name System

IP addresses are a bit hard to remember, right?

- Who can remember 87.238.248.136 ???
- DNS (Domain Name System) are *Name Services*
 - Computers that translate names into IP addresses

SuperTool ^{Beta7}

baerbak.com DNS Lookup

a:baerbak.com Find Problems

Type	Domain Name	IP Address	TTL
A	baerbak.com	87.238.248.136 <small>DK</small>	6 hrs

	Test	Result
✓	DNS Record Published	DNS Record found

[dns check](#) [mx lookup](#) [whois lookup](#) [spf lookup](#) [dns propagation](#)

Reported by ns2.domainteam.dk on 11/2/2017 at 1:10:18 PM (UTC 0), [just for you](#) ([History](#))

Local names

- Any computer has its own name
 - Normally you give it a name when installing
- On Linux you may change it by editing a few files

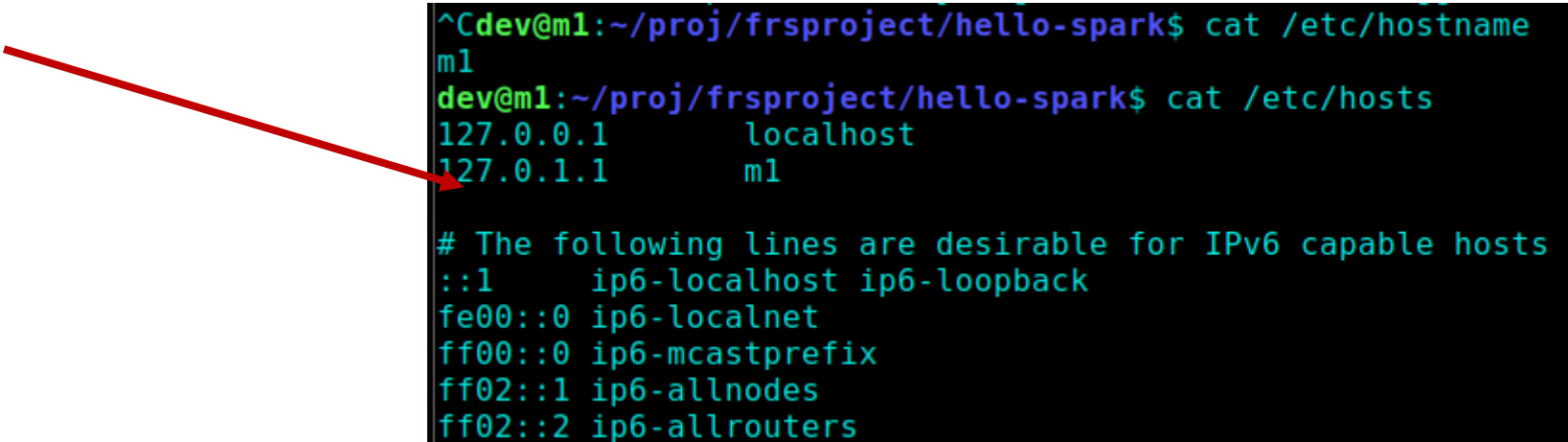
```
^Cdev@m1:~/proj/frsproject/hello-spark$ cat /etc/hostname
m1
dev@m1:~/proj/frsproject/hello-spark$ cat /etc/hosts
127.0.0.1    localhost
127.0.1.1    m1

# The following lines are desirable for IPv6 capable hosts
::1         ip6-localhost ip6-loopback
fe00::0     ip6-localnet
ff00::0     ip6-mcastprefix
ff02::1     ip6-allnodes
ff02::2     ip6-allrouters
```

- **Localhost** is 127.0.0.1 which is the IP address of the computer itself!

You Own DNS

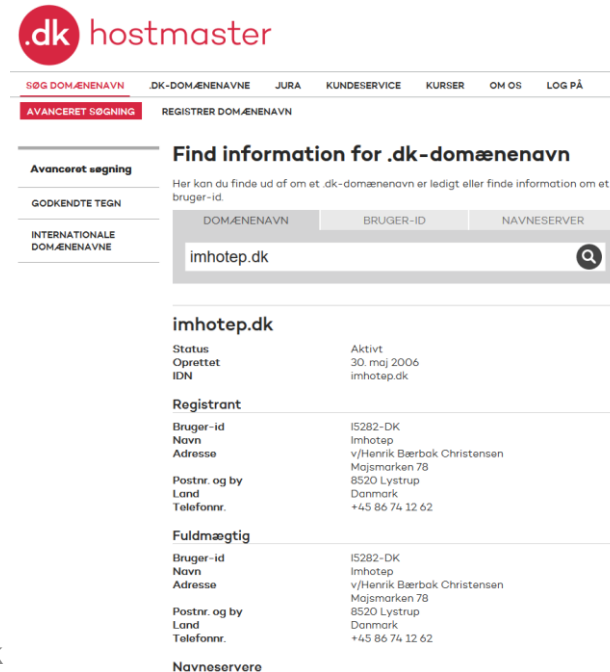
- You can actually maintain your own DNS by editing the *hosts* file on Linux



```
^Cdev@m1:~/proj/frsproject/hello-spark$ cat /etc/hostname
m1
dev@m1:~/proj/frsproject/hello-spark$ cat /etc/hosts
127.0.0.1    localhost
127.0.1.1    m1

# The following lines are desirable for IPv6 capable hosts
::1         ip6-localhost ip6-loopback
fe00::0    ip6-localnet
ff00::0    ip6-mcastprefix
ff02::1    ip6-allnodes
ff02::2    ip6-allrouters
```

- Only works on my machine ☹️
- So – how do I get a global domain name?
- For ‘.dk’ domains **DK-Hostmaster** keeps track of all Danish domains



.dk hostmaster

SØG DOMÆNENAVN | DK-DOMÆNENAVNE | JURA | KUNDESERVICE | KURSER | OM OS | LOG PÅ

AVANCERET SØGNING | REGISTRER DOMÆNENAVN

Find information for .dk-domænenavn

Her kan du finde ud af om et .dk-domænenavn er ledigt eller finde information om et bruger-id.

DOMÆNENAVN | BRUGER-ID | NAVNESERVER

imhotep.dk

imhotep.dk

Status	Aktivt
Oprettet	30. maj 2006
IDN	imhotep.dk

Registrant

Bruger-id	IS282-DK
Navn	Imhotep
Adresse	v/Henrik Bærbak Christensen Majsmarken 78
Postnr. og by	8520 Lystrup
Land	Danmark
Telefonnr.	+45 86 74 12 62

Fuldmægtig

Bruger-id	IS282-DK
Navn	Imhotep
Adresse	v/Henrik Bærbak Christensen Majsmarken 78
Postnr. og by	8520 Lystrup
Land	Danmark
Telefonnr.	+45 86 74 12 62

Navneservere



Name Servers

- But you only register the domain, you need a Name Server to handle the actual lookup
- *I log into my dk-hostmaster account and assign the name of my selected name service provider*

Navneservere

ns1.gratisdns.dk	GDNS1-DK
ns2.gratisdns.dk	GDNS1-DK
ns3.gratisdns.dk	GDNS1-DK
ns4.gratisdns.dk	GDNS1-DK
ns5.gratisdns.dk	GDNS1-DK

Create a New Name

- Scenario: *Mathilde wants her own MineCraft server*
- I do
 - Rent a virtual machine on DigitalOcean
 - So I get an IP address of that machine
- I log into my 'GratisDNS' account and create an A record

DNS for: imhotep.dk

A (Navn -> IP adresse)

A records peger på ipv4 records

Host	minecraft.imhotep.dk
IP (ipv4)	87.200.125.75
TTL	43200

Tilbage Tilføj/Ændre A record



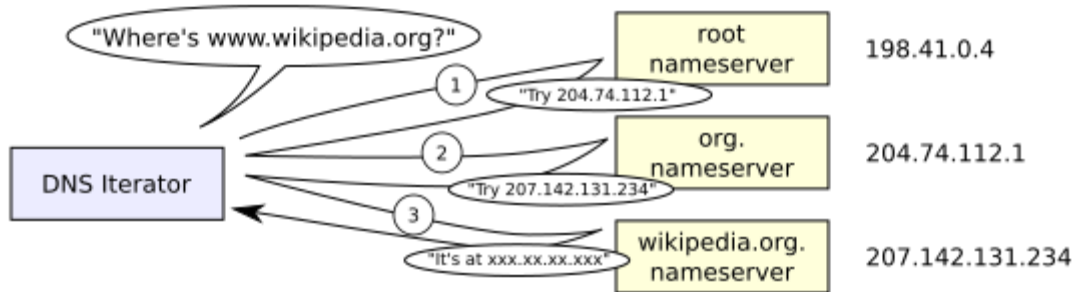


AARHUS UNIVERSITET

Resolving Names

- Any node on the IP network has a (local) Name Server registered, the one to contact *first*
 - Windows: 'nslookup'
 - Linux: 'nslookup' 😊
- Algorithm: "If I do not know, I know who knows"
 - Picks the name apart **right to left!**
 - dk *before* imhotep *before* www

```
d:\proj\SWEA-E18\web>nslookup
Default Server:  ge0.ns1.dk.ip.fullrate.dk
Address:  89.150.129.22
```



- Of course, contacting 3-8 servers just to resolve a DNS entry is *extremely expensive*
- **Caching** Tactic: ‘Maintain multiple copies of data’
 - Each DNS server caches the lookup
 - So my local DNS server knows the address immediately the next time I ask
 - Browsers maintain their own caches!
 - No need to talk to the DNS at all

Time To Live


- But but – what happens when IPs change then?
 - All the caches will send requests to the *old* node?
- The principle of **delegation** is used in DNS
 - I move my MineCraft server to another provider – and get a new IP address
 - The DNS system has to adapt: **TTL: TimeToLive**

DNS for: imhotep.dk

A (Navn -> IP adresse)
A records peger på ipv4 records

Host	minecraft.imhotep.dk
IP (ipv4)	87.200.125.75
TTL	43200

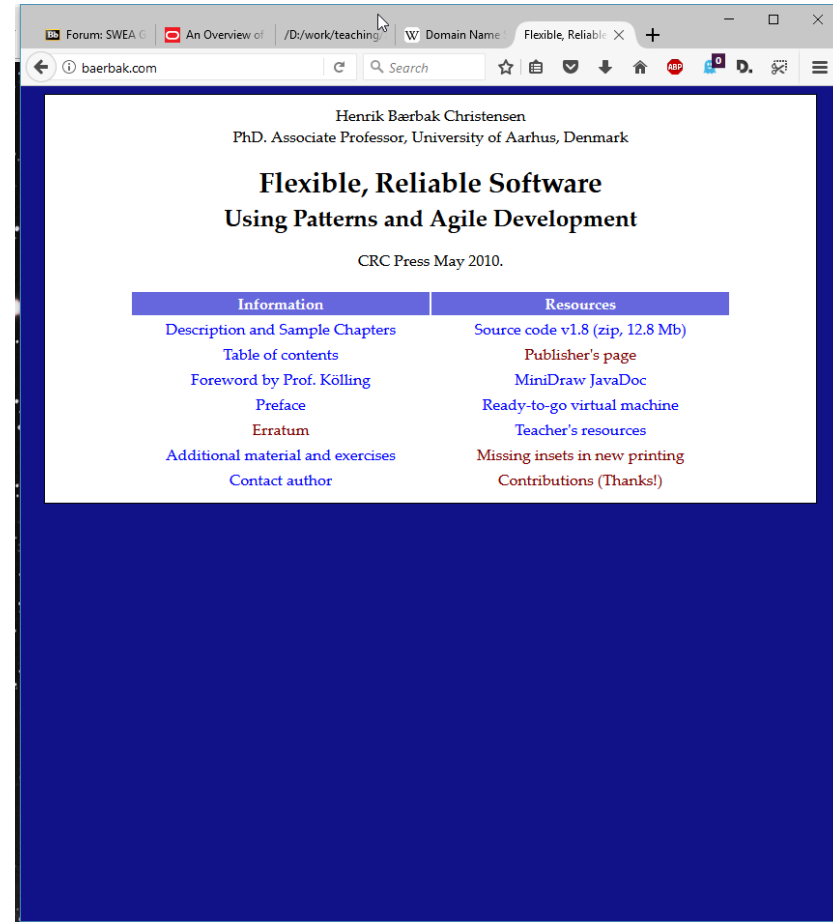
Tilbage Tilføj/Ændre A record



Takes up to 24 hours
world-wide

- baerbak.com will become
 - <http://www.baerbak.com>
- Firefox calls DNS server
 - Translate it into IP address
- Firefox will then send a http request to port 80 on that ip address

- ... which will return a HTML document





Summary

- To send a datagram, you have to know the address of the receiver
- Every node in an IP network has an **IP address**
 - IP address xxx.xxx.xxx.xxx (or IPv6)
- Nodes for a wider audience use **DNS** servers to assign a **hostname** to a specific IP address
 - www.dr.dk instead of xxx.xxx.xxx.xxx
- Every node has **65.536 ports**
 - Quite a few below 1024 are reserved



AARHUS UNIVERSITET

TCP

The last piece of the puzzle



Actually, rather hidden

- IP splits data into packets/datagrams and sends them
 - But they get lost!
 - They become garbled
 - They arrive out-of-order
- TCP introduce reliability
 - Get packet 1, 2, 3, 5, ~~7~~, 6...
 - Request packet 4 again, and 7 as it was garbled
 - Forward the full data by putting segments in correct order



AARHUS UNIVERSITET

Network Address Translation

Weird Behaviour Warning



Segmenting Networks

- Organizations, projects, homes create their own LANs.
 - Security, convenience, performance
- Example:
 - At home, I have a router that assigns each connected node an IP in the 192.168.x.x space
 - But at any time there are thousands of machines with IP 192.168.1.38
 - How does 'www.imhotep.dk' know which computer to return the HTML document to, then???

```
Ethernet adapter Ethernet:  
  
Connection-specific DNS Suffix . . :  
Link-local IPv6 Address . . . . . : fe80::48e:8e59:9c  
IPv4 Address. . . . . : 192.168.1.38  
Subnet Mask . . . . . : 255.255.255.0  
Default Gateway . . . . . : 192.168.1.1
```

All IP packets have a source IP address and a destination IP address. Typically packets passing from the private network to the public network will have their source address modified, while packets passing from the public network back to the private network will have their destination address modified. To avoid ambiguity in how replies are translated, further modifications to the packets are required. The vast bulk of Internet traffic uses [Transmission Control Protocol \(TCP\)](#) or [User Datagram Protocol \(UDP\)](#). For these protocols the [port numbers](#) are changed so that the combination of IP address and port information on the returned packet can be unambiguously mapped to the corresponding private network destination. [RFC 2663](#) uses the term *network address and port translation (NAPT)* for this type of NAT. Other names include *port address translation (PAT)*, *IP masquerading*, *NAT overload* and *many-to-one NAT*. This is the most common type of NAT and has become synonymous with the term "NAT" in common usage.

- So NAT in my router simply change IP:port of the datagrams so the web server returns to the router instead; once it has been received, the router forwards to the local node



- NAT makes networking behave ‘weird’:
 - *I can see you, but you cannot see me!*
 - My home computer can see the full internet, but no computer on the internet can see mine!
 - They can only see my ISP’s computer, which is the only one that can see my router, which is the only one who can see my computer!
- VMWare Player does NAT between your host machine and the course VM you are running

Implications

- VMWare Player does NAT between your host machine and the course VM you are running
 - It installs an additional network on the host

```
Ethernet adapter VMware Network Adapter VMnet8:  
  
Connection-specific DNS Suffix . . :  
Link-local IPv6 Address . . . . . : fe80::170:c808:a810:92a%11  
IPv4 Address. . . . . : 192.168.85.1  
Subnet Mask . . . . . : 255.255.255.0  
Default Gateway . . . . . :
```

- Therefore your host has multiple IP addresses, on multiple networks
 - Meaning host and VM can communicate on the 192.168.85.* network. ***Remember to use that for local testing!***



AARHUS UNIVERSITET

Handy commands



Some Nice Network Commands

AARHUS UNIVERSITET

- Debug 101
 - *Can my computer see the other computer???*
 - ‘ping www.imhotep.dk’
 - ‘ping 192.168.1.37’
- What is my IP?
 - Windows: ipconfig / linux: ifconfig

```
Ethernet adapter Ethernet:
```

```
Connection-specific DNS Suffix . . :  
Link-local IPv6 Address . . . . . : fe80::48e:8e59:9c35:9806%18  
IPv4 Address. . . . . : 192.168.1.38  
Subnet Mask . . . . . : 255.255.255.0  
Default Gateway . . . . . : 192.168.1.1
```

```
ens33 Link encap:Ethernet HWaddr 00:0c:29:58:f5:c2  
inet addr:192.168.85.128 Bcast:192.168.85.255 Mask:255.255.255.0  
inet6 addr: fe80::20c:29ff:fe58:f5c2/64 Scope:Link  
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1  
RX packets:83079 errors:0 dropped:0 overruns:0 frame:0  
TX packets:36489 errors:0 dropped:0 overruns:0 carrier:0  
collisions:0 txqueuelen:1000  
RX bytes:94975896 (94.9 MB) TX bytes:4403632 (4.4 MB)
```



- The Distributed course will go into the more details
 - I think 😊
- Lot of concepts but not core curriculum in SWEA
- However, you will bump into some of these issues in the mandatory project on distribution...