



UiO **•** **Institutt for informatikk**

Det matematisk-naturvitenskapelige fakultet

**IN1020 - Introduksjon til datateknologi**

Forelesning – 02.11.2022

***Din hjemmeruter - Lagene spiller sammen***

*Håkon Kvale Stensland*



**simula**



## Plan for ”nettverksdelen” av IN1020

- *8. september - Introduksjon til operativsystemer*
- *20. oktober – Nettverk 101 – Introduksjon og historie*
- *26. oktober – Lagdeling og nettverksprotokoller*
- *27. oktober – Kryptering i datakommunikasjon og som sikkerhetstiltak*
- **2. november – Hvordan fungerer din trådløse ruter?**
- **3. november – Tjenester i Internett**

# Lagene spiller sammen

- *Repetisjon: Lagene i TCP/IP-modellen fra i forrige torsdag...*
- Dere har lært om lagene, men hvordan virker dette i praksis?
- Utgangspunkt i Ethernet / IP / TCP / Applikasjonslag
- Trafikkpolitiet i Internett (ISP og congestion control)
- Hvordan oversettes tekstlige maskinnavn navn til IP-adresser (DNS)

# Lagene i TCP/IP-modellen



**simula**



*Kan også inneholde en header,  
men det bestemmer applikasjonen*

## Lag 5 - Applikasjonslaget

- Lag med tjenester for applikasjoner:
- Eksempler:
  - Nettlesere (WWW)
  - E-post
  - Filoverføring
  - P2P
- *Mer om dette 3. november (Tjenester i Internett)!*



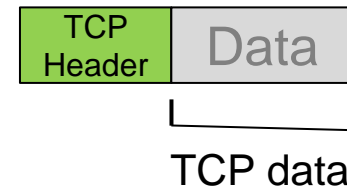
TCP data

# Lag 4 - Transportlaget

- TCP: HTTP, E-post, filoverføring, etc.
- UDP: Strømming av video og lyd
- Bruker «port» som en unik identifikator.
  - Representeres med et 16-bit heltall

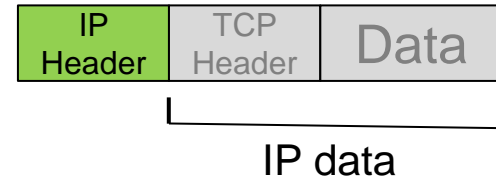
TCP Segment Header Format								
Bit #	0	7	8	15	16	23	24	31
0	Source Port				Destination Port			
32	Sequence Number							
64	Acknowledgment Number							
96	Data Offset	Res	Flags			Window Size		
128	Header and Data Checksum				Urgent Pointer			
160...	Options							

UDP Datagram Header Format								
Bit #	0	7	8	15	16	23	24	31
0	Source Port				Destination Port			
32	Length				Header and Data Checksum			



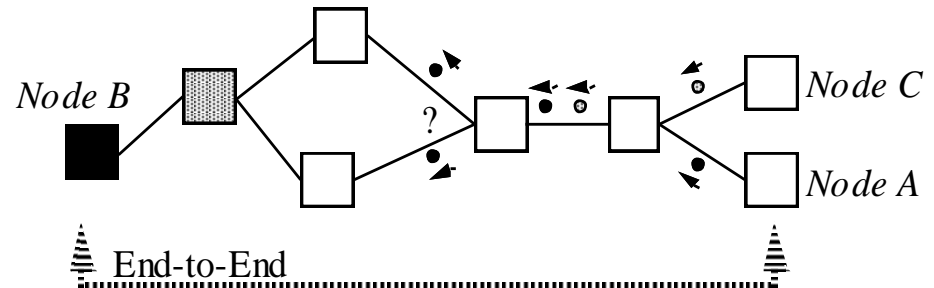
## Lag 4 - Transportlaget

- **TCP:**
  - Oppsett av forbindelse (3-way handshake)
  - Garanterer at pakkene leveres i riktig rekkefølge
  - *Pålitelighet* – Pakker sendes på nytt hvis kvitteringen (ACK) ikke kommer frem
  - Flytkontroll og meningskontroll
- **UDP:**
  - Tilkoblingsløs forbindelse
  - Ingen garantier
  - «Best-effort» levering av data

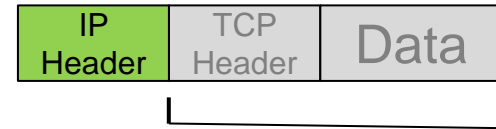


## Lag 3 - Nettverkslaget

- **Kobler sammen systemene ende-til-ende**
- **Ruting**
  - Statisk, definert under tilkobling eller dynamisk
  - Meningskontroll (for mange pakker på en sti)
  - Tjenestekvalitet (QoS)
- En «ruter» jobber på lag 3
- Eksempler:
  - IP (tilkoblingsløst)



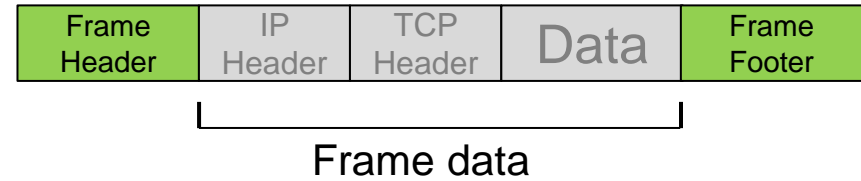




# Lag 3 - Nettverkslaget

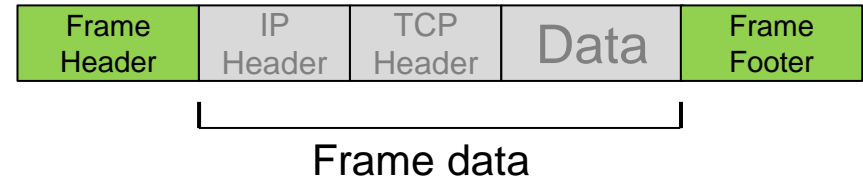
- Den mest brukte nettverkslagsprotokollen i dag er Internet Protocol (IP). Den mest brukte versjonen er IPv4.
- IPv4 bruker en 32-bit adresse, ( $4.3 \times 10^9$ )
- Den nye versjonen, IPv6 har 128-bit adresser ( $3.4 \times 10^{38}$ )
  - Representeres med fire 8-bit heltall: 192.168.1.101

Version		IHL		Type of Service		Total Length			
Identification				Flags		Fragment Offset			
Time to Live		Protocol		Header Checksum					
Source IP Address									
Destination IP Address									
IP Options (optional)						Padding			
Data									
More Data...?									



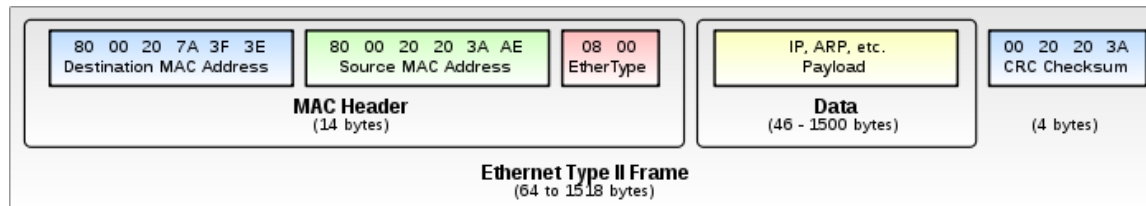
- **Pålitelig overføring mellom to enheter.**
  - Pakker som overføres i linklaget kalles «frames»
  - Feildeteksjon or retting innenfor en «frame»
- En «switch» vil kun jobbe på lag 2
- Lag 2 vil kunne ha enkel flytkontroll
  - Rask sender, treg mottaker
- Medium Access Control (MAC)





## Lag 2 - Linklaget

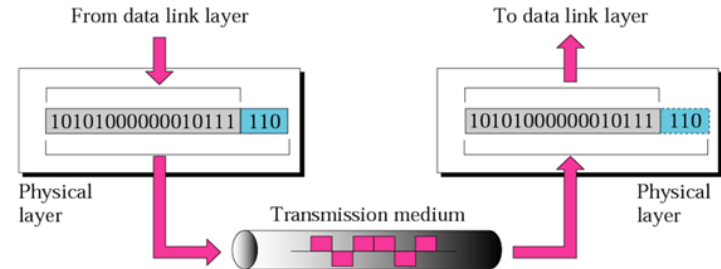
- Det vanligste linklagene er "Ethernet", og "WiFi". Disse er ganske like, men har noen forskjeller.
- Bruker en 6-byte adresse (48-bit) som ofte er lagret i nettverkskortet
  - MAC-adresse, brukes både på WiFi og Ethernet.
  - Hver *byte* representeres med en *heksadesimal* verdi: 07:01:02:01:2C:4B



# Lag 1 – Det fysiske laget

- **Signalrepresentasjonen av bits:**
- Sørger for at 1-bit også blir mottatt som 1-bit (og ikke et 0-bit)
- Mekanikk: Koblingstype, kabler/medium,..
- Elektronikk: Spenning, bit-lengde,..
- Formelle regler for kommunikasjon:
  - Enveis (unidirectional) – half-duplex
  - Toveis (bidirectional) – full-duplex
  - *Hva skal markere starten og slutten på overføringer*

– Eksempler: RS-232-C, 1000BASE-X



# Lagene i Internett (TCP/IP referansemodellen)



Applikasjonslag

<http://www.uio.no>

Transportlag

192.168.1.5:80

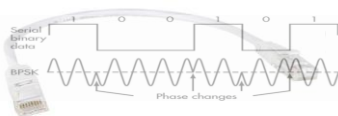
Nettverkslag

192.168.1.5

Linklag

A1:B2:C3:D4:E5:F6

Fysiske lag



Nettverkslag

Linklag

Fysiske lag



Applikasjonslag

Transportlag

Nettverkslag

Linklag

Fysiske lag



UiO • **Institutt for informatikk**

Det matematisk-naturvitenskapelige fakultet

Hva er en IP-adresse?



**simula**



## Hva er en IP-adresse

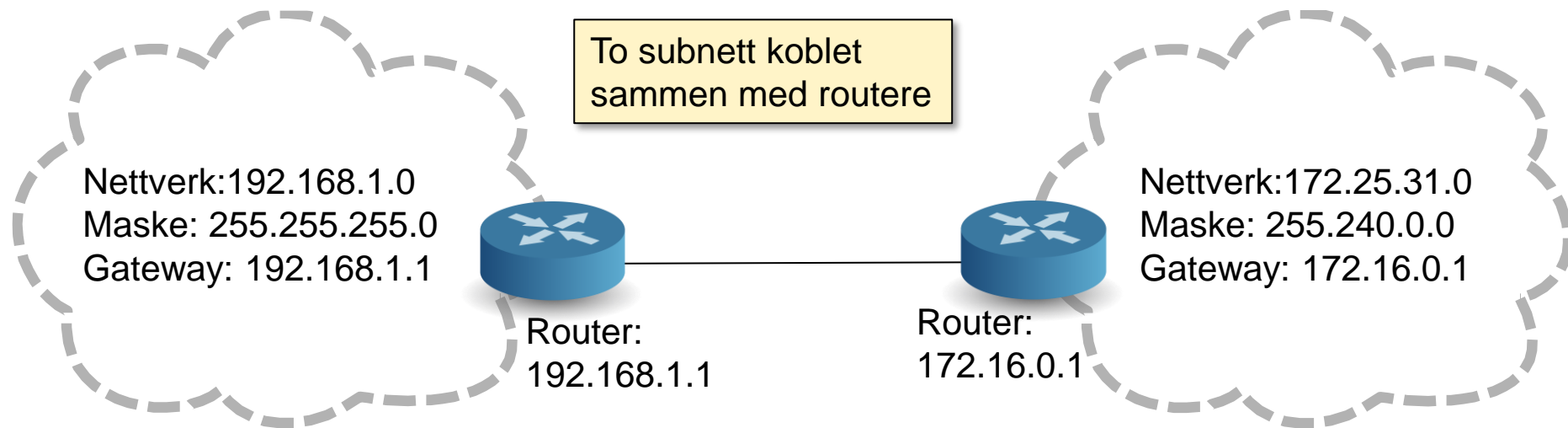
- Hvordan kan man sende til en annen maskin på tvers av mange små nettverk?
- Ved hjelp av adresser som er «unike» på hele Internett
- Postadresse:
  - Kristin Skar, Ole Johan Dahls hus, Gaustadalléen 23B, 0373 Oslo, Norge
- IP-adresse
  - Tilsvarende prinsipp, men skiller bare mellom adresser innenfor og utenfor det lokale nettverket.
  - Er adressen på mitt lokale nett?
    - Ja: Lever pakken rett til mottakeren
    - Nei: Send til router, som får ansvaret for å sende videre på Internett

# Lokalnettverk (LAN) og subnett

IP/Internett

Linklaget

- Internett er et sammenkobling av mindre, separate nettverk.
- Koblet sammen med switch og/eller HUB (lag 2\*)
  - Switch: filtrerer og videresender.
  - HUB: Videresender det som kommer inn på alle porter
- For å sende en pakke til en maskin utenfor ditt lokale nettverk må den sendes til en router som vet hvor den skal videresendes.



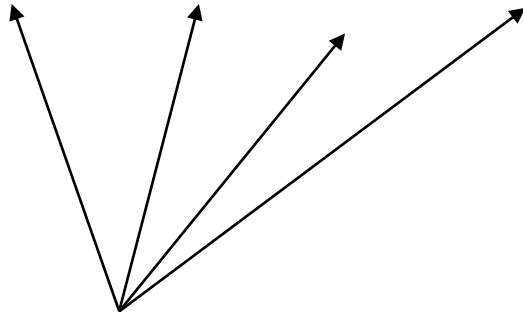


## IP-adresser (IPv4)

### IP-adresse

192.168.1.5

11000000.10101000.00000001.00000101



#### **Oktetter:**

Består av 8 bits hver. Maks verdi for hver oktett er 255

### Nettverksmaske

255.255.0.0

11111111.11111111.00000000.00000000

Masken angir hvilke bits som definerer dette subnettet.

Bits som er satt til 0 kan varieres for å angi IP-adresser i subnettet.  
(vertsaddressedel)

Bits som er satt til 1 angir delen av IP-adressen som definerer hvilket nettverk vertene tilhører.

## CIDR- og punktnotasjon av subnett

- Nettverksmasken består alltid av en sammenhengende serie "1" deretter en sammenhengende serie "0"
  - Eks: 255.255.255.0
  - 11111111.11111111.11111111.00000000
- Det er to vanlige måter å notere omfanget av et subnett:
  - Punktnotasjon:
    - For eksempel: 192.168.1.0
    - Må da oppgi nettverksmaske: 255.255.255.0
  - CIDR (Classless Inter-Domain Routing) notasjon:
    - 192.168.1.0/24
    - Vanlig punktnotasjon først.
    - Tallet etter skråstreken angir hvor mange bits nettverksmasken består av

Vertsdel

Nettverksdel

## Regne ut subnett fra en IP + nettverksmaske

En maskin i nettet har IP  $192.168.1.5 = 11000000.10101000.00000001.00000101$

Nettverksmasken er  $255.255.255.0 = 11111111.11111111.11111111.00000000$

For å finne subnettadressen til maskinen må du gjøre en bitvis **AND-operasjon** mellom IP-adressen og nettverksmasken.

$11000000.10101000.00000001.00000000 = \mathbf{192.168.1.0}$

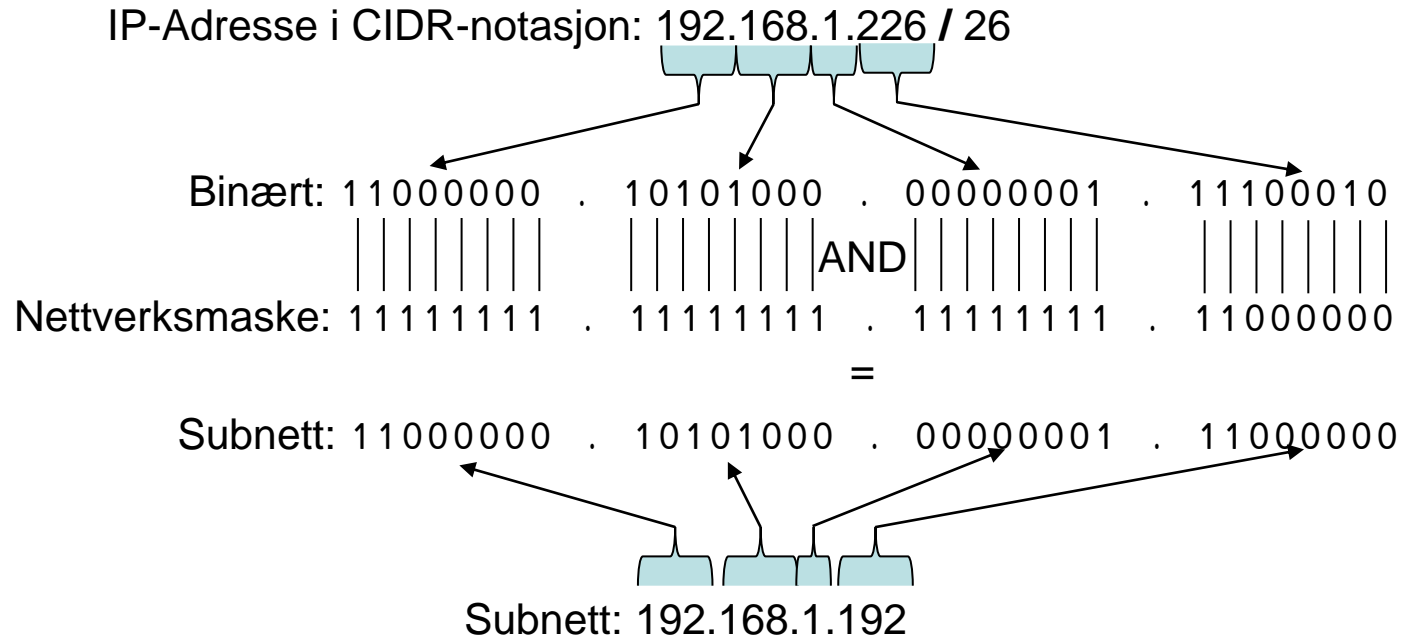
Dette er den første IP-adressen i subnettet og brukes til å identifisere subnettet.

## Kringkasting (send til alle)

- En melding som sendes ut på en spesiell adresse.
- Leveres til alle enheter som er koblet på samme LAN (nettverk):
  - Linklaget (MAC): FF:FF:FF:FF:FF:FF
  - IP/Internett: 255.255.255.255
- For en maskin på et subnett, finner du kringkastingsadressen ved å gjøre en bitvis **OR-operasjon** mellom maskinens IP-adresse og bit komplement (*bitvis invers*) av nettverksmasken.
  - Eks: IP-adresse 192.168.1.5 nettverksmaske: 255.255.255.0
  - $(192.168.1.5) \text{ OR } (0.0.0.255) = 192.168.1.255$

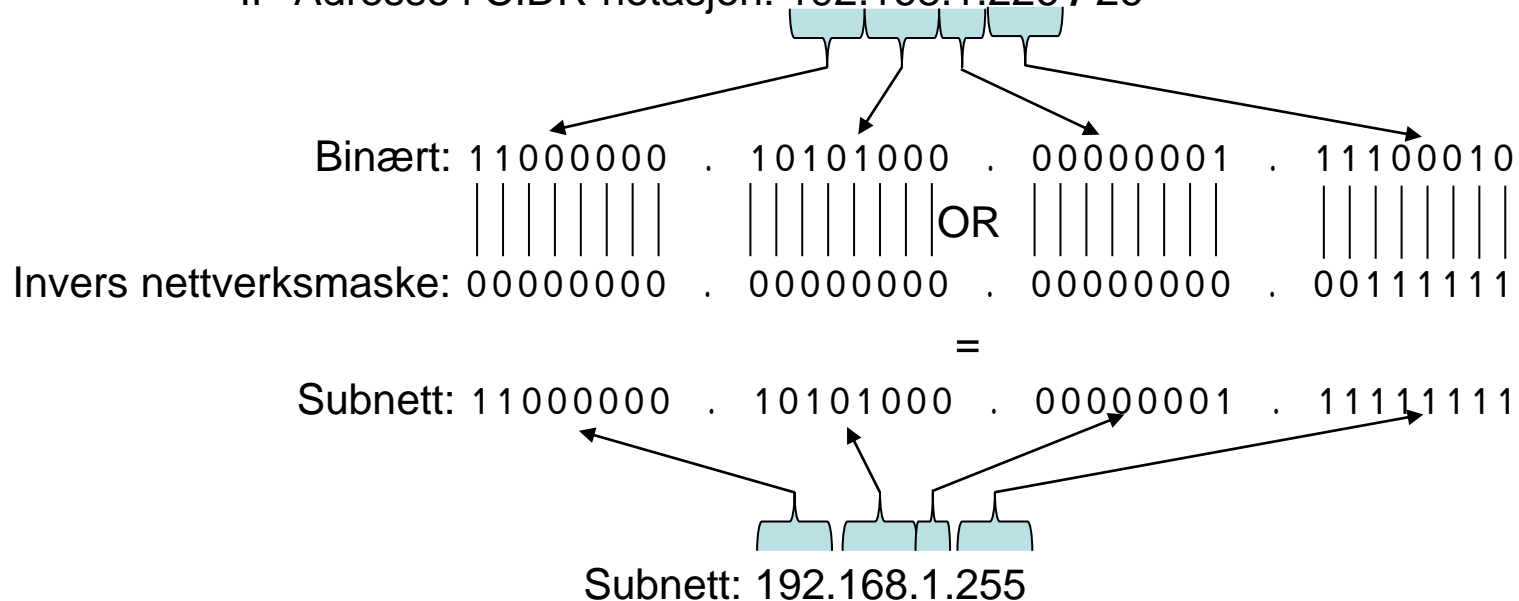


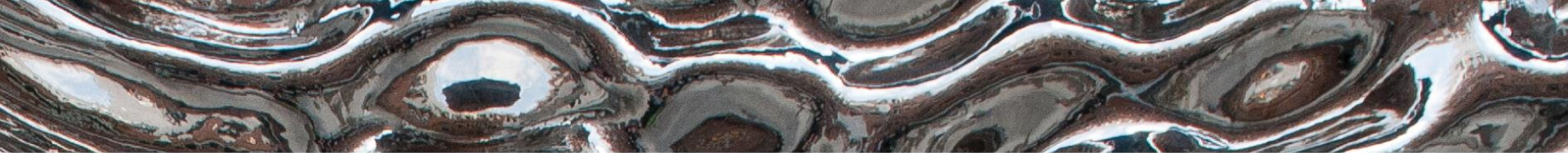
# Eksempel: subnettadresse fra IP / nettverksmaske



## Eksempel: kringkastingsadresse fra IP / nettverksmaske

IP-Adresse i CIDR-notasjon: 192.168.1.226 / 26





UiO • **Institutt for informatikk**

Det matematisk-naturvitenskapelige fakultet

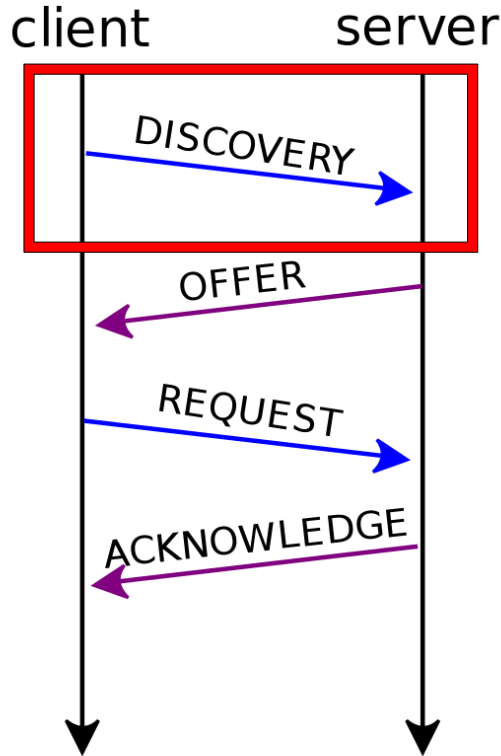
# DHCP – Utdeling av IP-adresser



**simula**

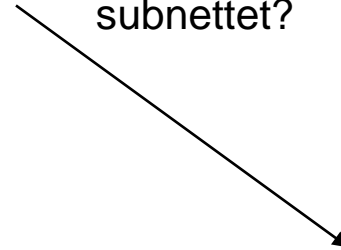


# DHCP - Automatisk utdeling av IP-adresser



Subnett: 192.168.1.0/24

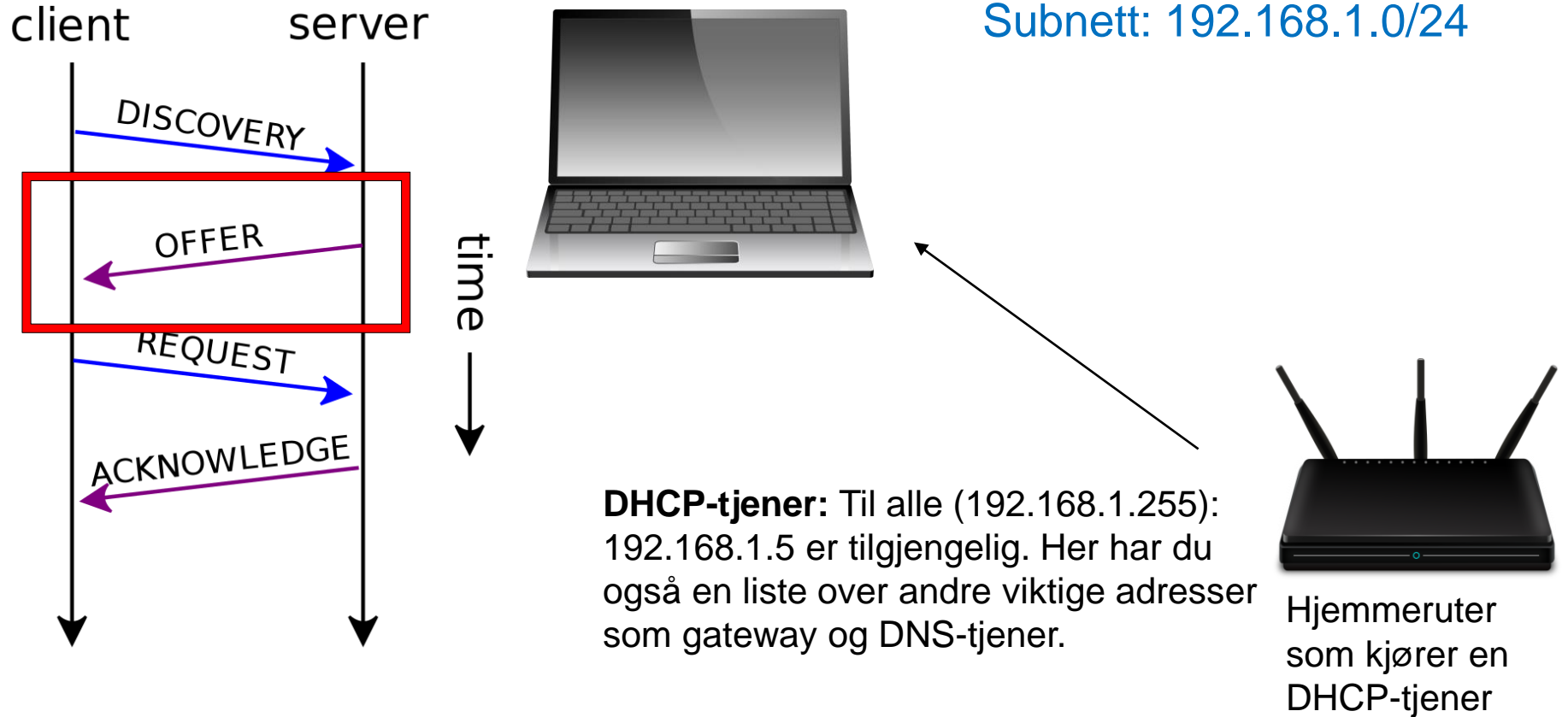
**Ny maskin:** Til alle (192.168.1.255):  
finnes det en maskin med myndighet  
til å dele ut IP-adresser på dette  
subnettet?



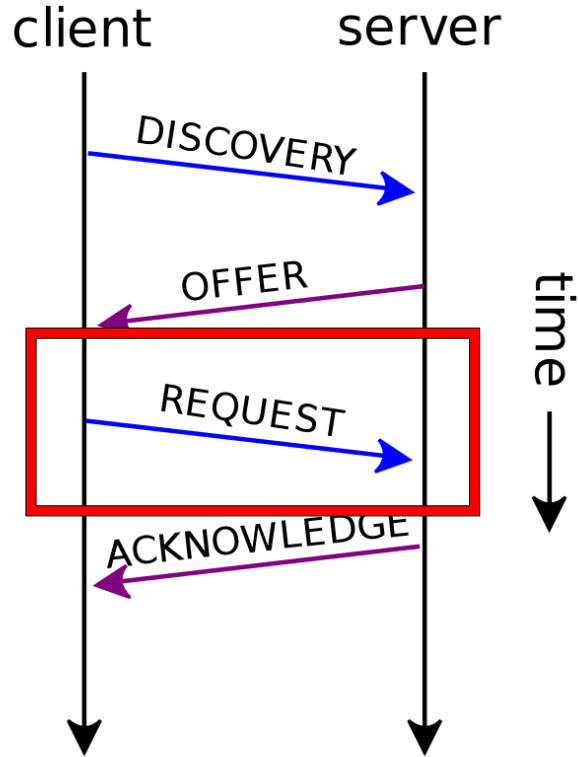
Hjemmeruter  
som kjører en  
DHCP-tjener



# DHCP - Automatisk utdeling av IP-adresser

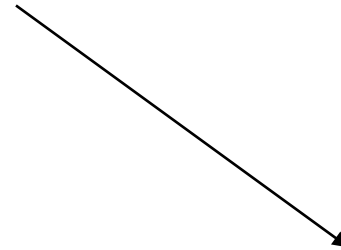


# DHCP - Automatisk utdeling av IP-adresser



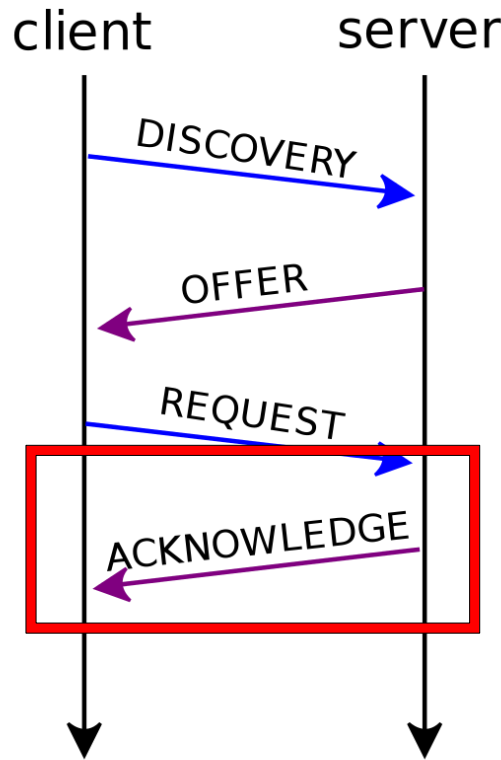
Subnett: 192.168.1.0/24

**Ny maskin:** Til alle (192.168.1.255):  
Ja takk!



Hjemmeruter  
som kjører en  
DHCP-tjener

# DHCP - Automatisk utdeling av IP-adresser



Subnett: 192.168.1.0/24

## DHCP-tjener:

Da har jeg skrevet opp at 192.168.1.5 er i bruk av deg for en periode på 24 timer. Ha en fin dag!



Hjemmeruter som kjører en DHCP-tjener



UiO • **Institutt for informatikk**

Det matematisk-naturvitenskapelige fakultet

# ARP – Koblingen mellom nettverk- og link-laget



**simula**



# ARP – Koblingen mellom nettverk og IP

- Nettverkskortene har en 6 byte lang media access control (MAC)-adresse som brukes til å identifisere maskinen innenfor et kringkastingsdomene (broadcast domain).
- For at IP skal fungere, må avsenderen vite hvilken MAC-adresse pakken skal sendes til.
- Address Resolution Protocol (ARP) kobler IP (Internett) og MAC (Linklaget).



# ARP-protokollen

Maskin 1 ønsker å kontakte 192.168.1.100

ARP-forespørsel  
Avsender: 11:22:33:44:55:66  
Mottager: FF:FF:FF:FF:FF:FF  
(kringkasting)  
"Hvem har 192.168.1.100?"



Router  
IP: 192.168.1.1  
MAC: F1:1B:14:12:A1:9F



Laptop 2  
IP: 192.168.1.100  
MAC: AA:BB:CC:DD:EE:FF



Maskin 1  
IP: 192.168.1.150  
MAC: 11:22:33:44:55:66



Laptop 1  
IP: 192.168.1.200  
MAC: A1:B2:C3:D4:E5:F6

# ARP-protokollen

Maskin 1 ønsker å kontakte 192.168.1.100



Router  
IP: 192.168.1.1  
MAC: F1:1B:14:12:A1:9F



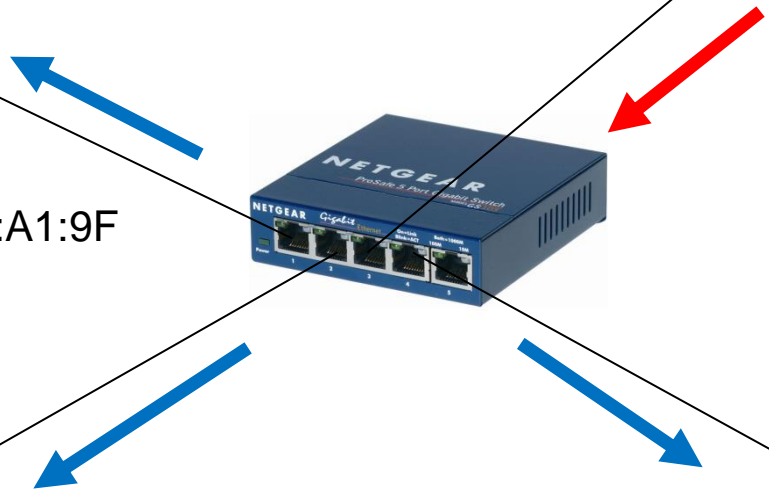
Laptop 2  
IP: 192.168.1.100  
MAC: AA:BB:CC:DD:EE:FF



Maskin 1  
IP: 192.168.1.150  
MAC: 11:22:33:44:55:66



Laptop 1  
IP: 192.168.1.200  
MAC: A1:B2:C3:D4:E5:F6



# ARP-protokollen

Maskin 1 ønsker å kontakte 192.168.1.100



Router

IP: 192.168.1.1

MAC: F1:1B:14:12:A1:9F



Laptop 2

IP: 192.168.1.100

MAC: AA:BB:CC:DD:EE:FF

ARP-svar

"Det er meg! Min MAC-adresse er  
AA:BB:CC:DD:EE:FF



Maskin 1

IP: 192.168.1.150

MAC: 11:22:33:44:55:66



Laptop 1

IP: 192.168.1.200

MAC: A1:B2:C3:D4:E5:F6



# ARP-protokollen

Maskin 1 ønsker å kontakte 192.168.1.100

IP/Internett

Maskinlaget



Router  
IP: 192.168.1.1  
MAC: F1:1B:14:12:A1:9F



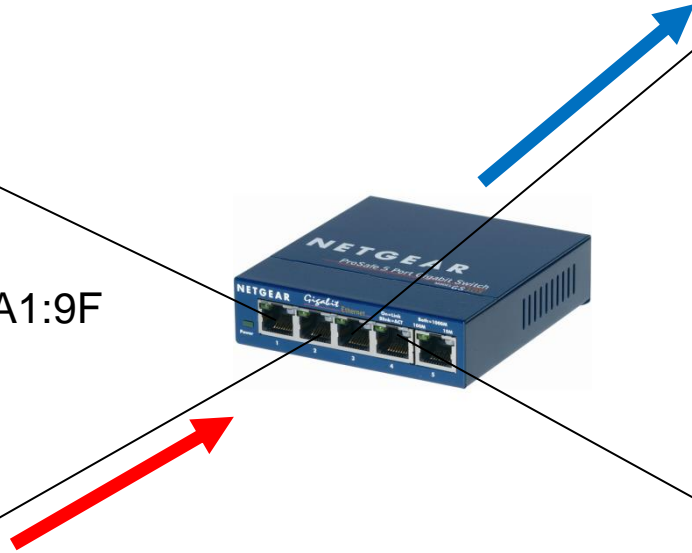
Laptop 2  
IP: 192.168.1.100  
MAC: AA:BB:CC:DD:EE:FF



Maskin 1  
IP: 192.168.1.150  
MAC: 11:22:33:44:55:66



Laptop 1  
IP: 192.168.1.200  
MAC: A1:B2:C3:D4:E5:F6



# ARP-protokollen

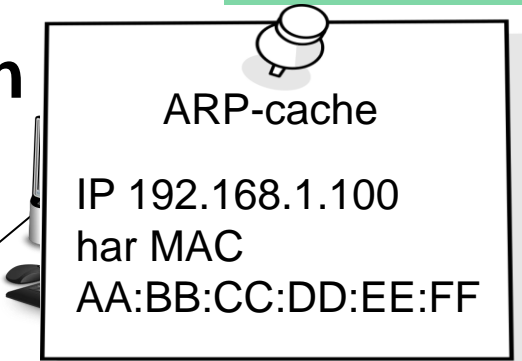
Maskin 1 ønsker å kontakte 192.168.1.100



Router  
IP: 192.168.1.1  
MAC: F1:1B:14:12:A1:9F



Laptop 2  
IP: 192.168.1.100  
MAC: AA:BB:CC:DD:EE:FF



Maskin 1  
IP: 192.168.1.150  
MAC: 11:22:33:44:55:66



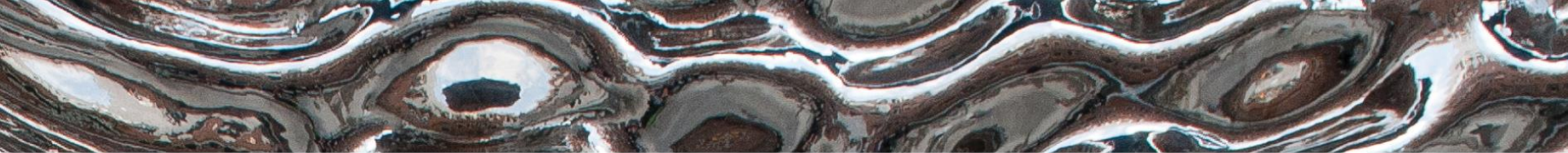
Laptop 1  
IP: 192.168.1.200  
MAC: A1:B2:C3:D4:E5:F6

# ARP-tabell på nidur (login.ifi.uio.no)

```
haakonks@nidur:~  
[haakonks@nidur ~]$ arp -ne  
Address                HWtype  HWaddress           Flags Mask           Iface  
129.240.65.2           ether   d0:94:66:47:c7:47   C                   ens192  
129.240.65.87          ether   78:2b:cb:4c:6e:cd   C                   ens192  
129.240.65.38          ether   00:50:56:a9:88:ec   C                   ens192  
129.240.65.12          ether   00:50:56:80:0a:83   C                   ens192  
129.240.65.65          ether   00:50:56:a9:ec:3c   C                   ens192  
129.240.65.62          ether   00:50:56:80:35:4b   C                   ens192  
129.240.65.59          ether   00:50:56:80:f2:fe   C                   ens192  
129.240.65.20          ether   00:50:56:a9:51:98   C                   ens192  
129.240.65.1           ether   00:00:0c:9f:f0:03   C                   ens192  
129.240.65.254         ether   00:de:fb:f0:61:c7   C                   ens192  
129.240.65.25          ether   00:25:64:ff:06:a8   C                   ens192  
129.240.65.61          ether   00:50:56:80:68:b4   C                   ens192  
129.240.65.83          ether   b8:ac:6f:d2:68:38   C                   ens192  
129.240.65.44          ether   00:50:56:80:50:2b   C                   ens192  
129.240.65.253         ether   00:de:fb:20:38:5f   C                   ens192  
129.240.65.63          (incomplete)                ens192  
129.240.65.102         (incomplete)                ens192  
129.240.65.60          ether   00:50:56:80:75:22   C                   ens192  
129.240.65.41          ether   bc:30:5b:3a:cb:40   C                   ens192  
[haakonks@nidur ~]$ █
```

- Om kringkastingsdomenet inneholder for mange enheter, kan det by på problemer:
  - ARP går til alle maskiner i kringkastingsdomenet. Om domenet er for stort kan dette stjele kapasitet som burde ha vært brukt til å overføre data.
  - DHCP-forespørsler går også til alle -> samme problem.
- Avanserte switcher kan filtrere ARP og DHCP for å beskytte mot overbelastning fra kringkastet trafikk.
- ARP kan også brukes til å finne duplikat IP-adresser:
  - "Hvem har IP 0.0.0.0" sendes til MAC  
FF:FF:FF:FF:FF:FF
  - Svar fra alle: IP kommer flere ganger = feil





UiO : **Institutt for informatikk**

Det matematisk-naturvitenskapelige fakultet

# Porter & NAT



**simula**



# Én IP-adresse – mange porter

Men du ønsker å kjøre mange tjenester på samme maskin.  
**Hvordan skiller man tjenestene fra hverandre?**



IP: 192.168.1.5

Transportprotokollene (UDP, TCP) implementerer ”porter” som muliggjør totalt 65535 samtidige forbindelser på én IP-adresse

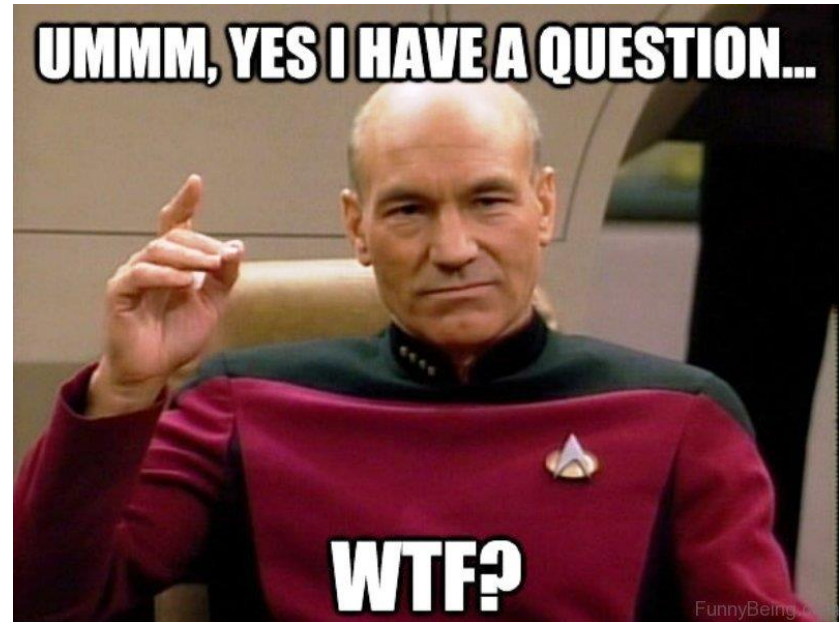
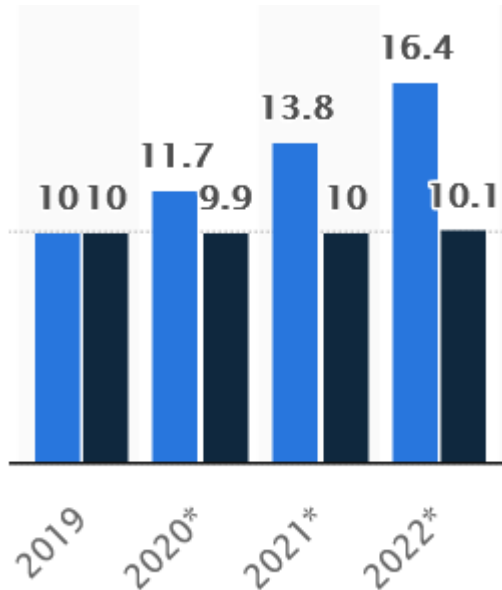
En vanlig måte å notere IP og port på er IP:port (eks.: 192.168.1.5:22 )

Port	Tjeneste
0	Reservert
1	tcpmux
...	
22	SSH
...	
80	HTTP
...	
1024-49151	Brukerporter
49152-65535	Dynamisk / privat

# Antall IP-adresser

**Maks antall mulige IP-adresser på 32 bit er  $2^{32}$ , eller 4.294.967.296**

...men anslag sier at det er over 23.000.000.000 enheter tilkoblet Internett i dag<sup>1</sup>...



NAT – Network Address Translation

1) Kilde: <https://www.statista.com/statistics/1101442/iot-number-of-connected-devices-worldwide/>



# Omgå problemet: Bruk adressene flere ganger

Nettverk: 192.168.1.0  
Maske: 255.255.255.0  
Gateway: 192.168.1.1

Internett

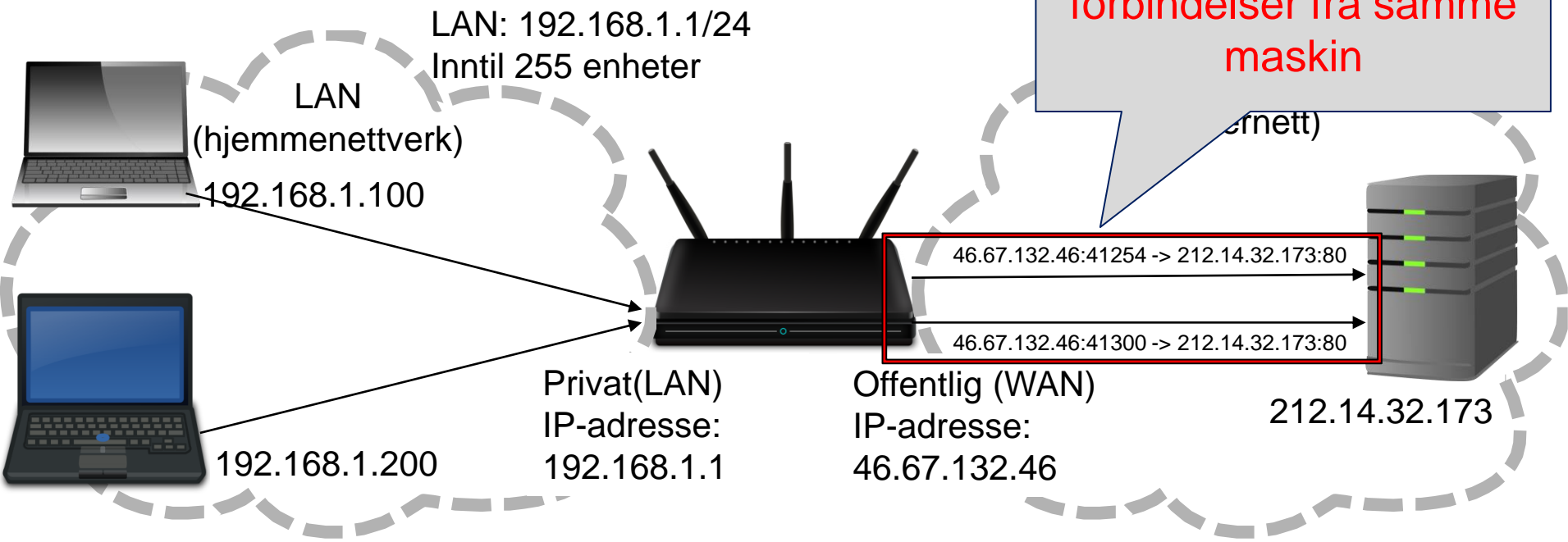
Men hvordan??

Nettverk: 192.168.1.0  
Maske: 255.255.255.0  
Gateway: 192.168.1.1

Nettverk: 192.168.1.0  
Maske: 255.255.255.0  
Gateway: 192.168.1.1



# NAT – Network Address Translation



Kilde IP	Mottaker	Oversatt adresse
192.168.1.100	212.14.32.173:80	46.67.132.46:41254
192.168.1.200	212.14.32.173:80	46.67.132.46:41300

# Private IP-adresser

RFC1918 name	IP address range	number of addresses	largest CIDR block (subnet mask)	host id size	mask bits
24-bit block	10.0.0.0 – 10.255.255.255	16,777,216	10.0.0.0/8 (255.0.0.0)	24 bits	8 bits
20-bit block	172.16.0.0 – 172.31.255.255	1,048,576	172.16.0.0/12 (255.240.0.0)	20 bits	12 bits
16-bit block	192.168.0.0 – 192.168.255.255	65,536	192.168.0.0/16 (255.255.0.0)	16 bits	16 bits

- Private IP-adresser er adresser som er reservert for bruk i lukkede nettverk og nettverk med NAT mot Internett
- Disse IPene skal ikke være direkte koblet mot Internett!
- En hjemmeruter er vanligvis satt opp til å gi deg et LAN med et subnett fra en av disse segmentene.

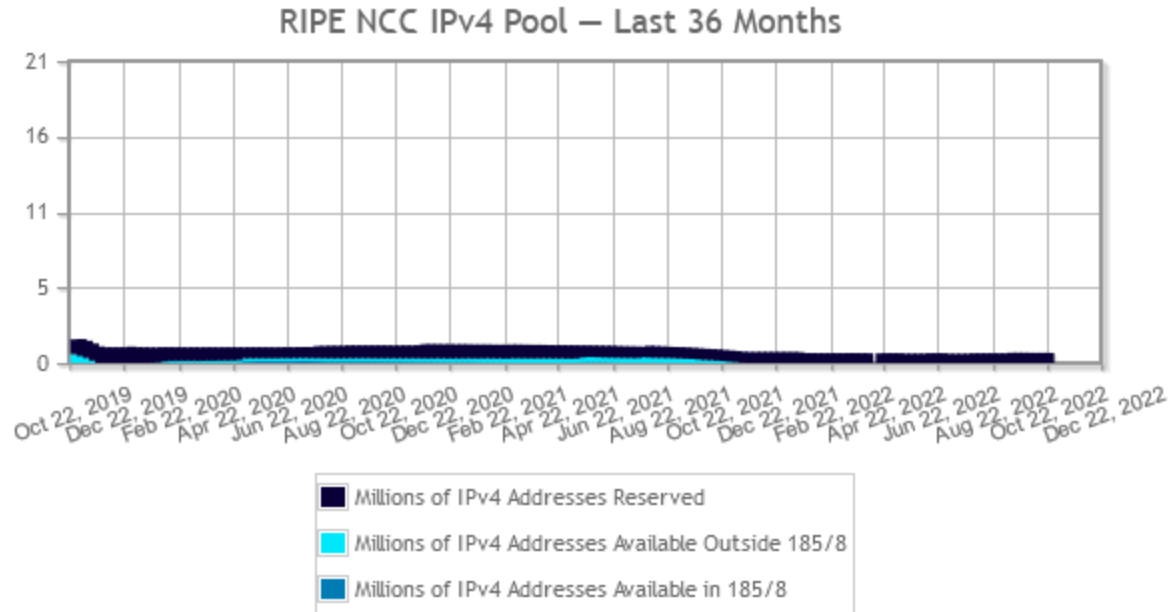
## Ulemper med NAT

- Ekstra kompleksitet i nettverket
- Routeren må bevare tilstanden til forbindelsene
- Nye forbindelser **må** initieres fra innsiden av NAT-nettverket
- Gjør det vanskelig å koble til utenifra
  - Universal Plug and Play (UPnP)
    - Enheter på LAN kan automatisk åpne opp for forbindelser utenifra
  - STUN / TURN ++
    - Bruke maskiner med gyldig Internett IP-adresse til å sette opp forbindelsen.
  - Videresending av porter
    - f.eks. 46.67.132.46:5000 -> 192.168.1.5:5000
  - demilitarized zone (DMZ): 46.67.132.46:\* -> 192.168.1.5:\*

## Anatomien til et LAN – et typisk oppsett

- LAN ID (CIDR): 192.168.0.0/24
- Kringkastingsadresse: 192.168.0.255
- Tilgjengelige adresser: 192.168.0.1-192.168-0.254
  - 254 adresser
  - ...men 1 av disse må settes av til router om du skal koble LAN til Internett
    - f.eks. 192.168.0.1
  - Router fungerer ofte også som DHCP-tjener og DNS-cache i hjemmenettverk

# IPv4 -> IPv6

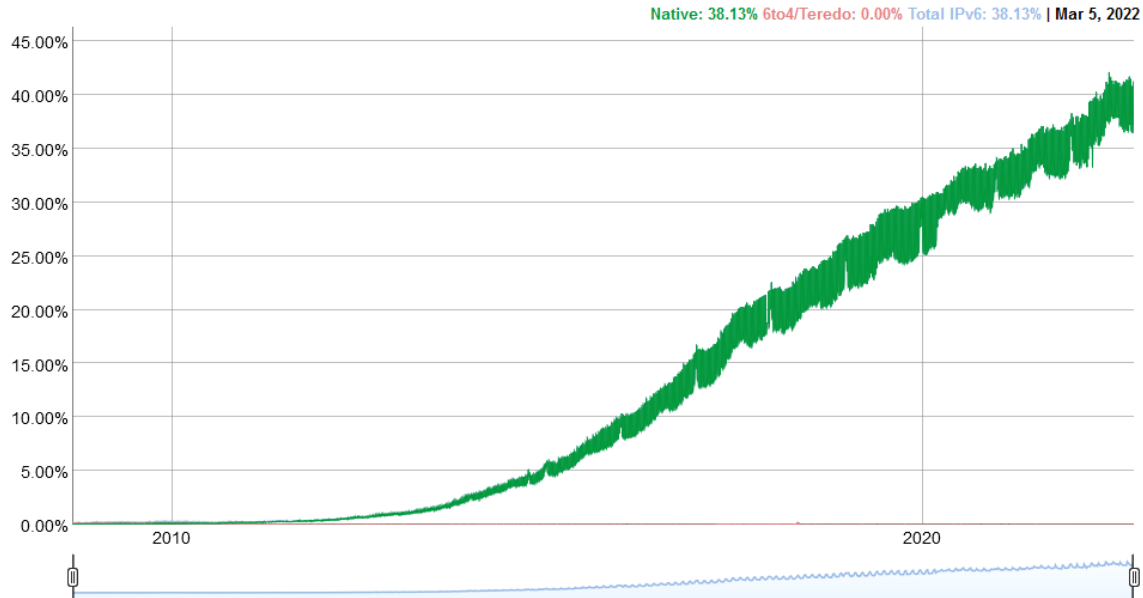


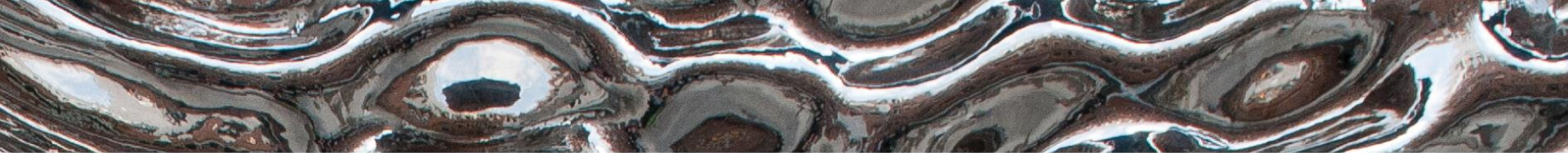
Selv med NAT i bruk på svært mange nettverk, er antall tilgjengelige IPv4-adresser blitt kritisk lavt.

- Lang prosess med å få i drift siden det må støttes i alle noder fra ende-til-ende
- 128-bits IP-adresse (mot 32-bit i IPv4)
- $2^{128}$  (eller  $3.4 \times 10^{38}$ ) mulige adresser

#### IPv6 Adoption

We are continuously measuring the availability of IPv6 connectivity among Google users. The graph shows the percentage of users that access Google over IPv6.





UiO : **Institutt for informatikk**

Det matematisk-naturvitenskapelige fakultet

# Ruteren sin oppgave i nettet



**simula**



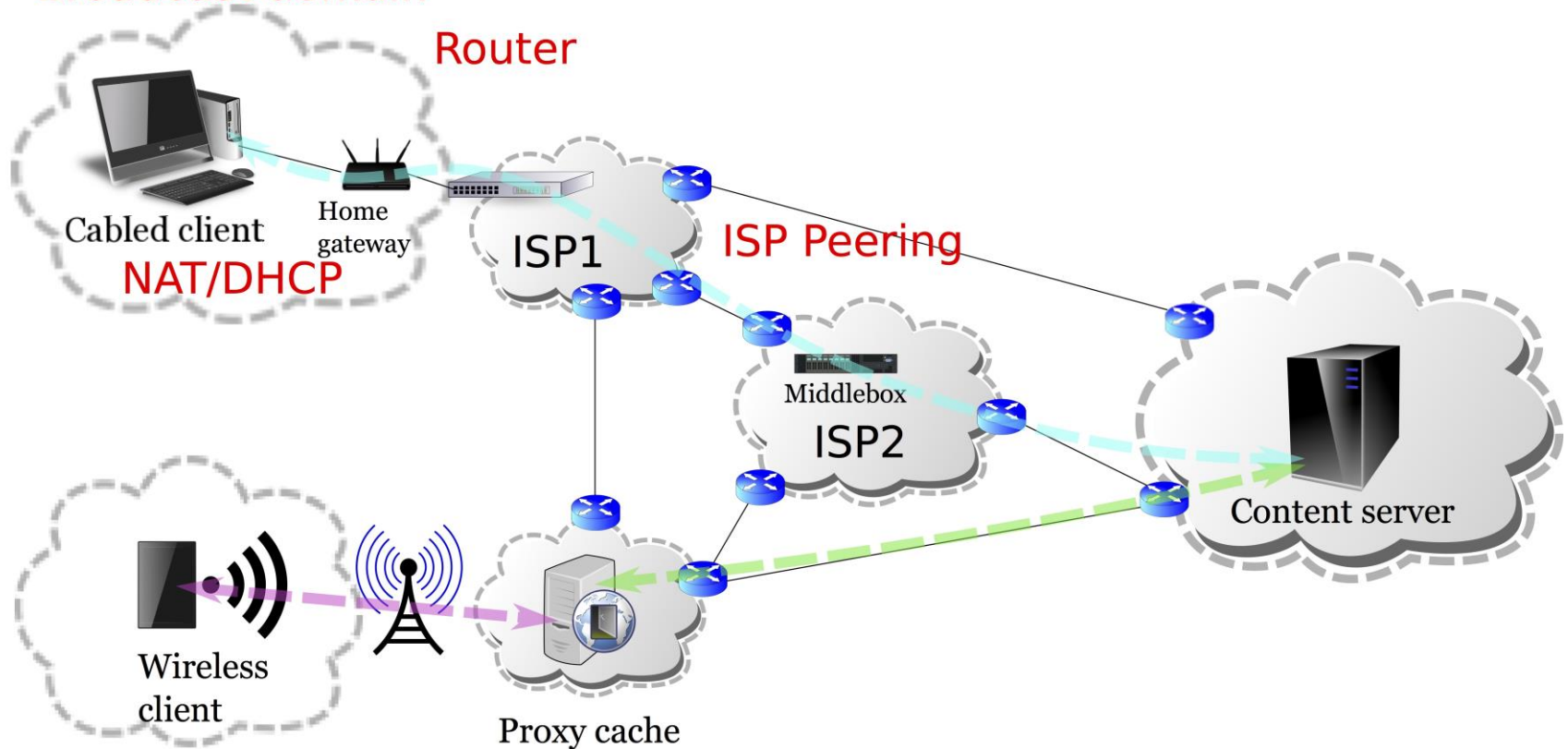
## Ruting i Internett (svært kort)

- Målet for ruting: å videresende en datapakke slik at den til slutt når måladressen sin.
- En *Internet Service Provider (ISP)* driver et nettverk og leverer datatransport til andre ISPer og sluttbrukere.
- *ISP Peering* er når ISPer inngår avtaler om å videresende hverandres trafikk. Økonomiske prinsipper er da med på å bestemme hvor trafikken flyter.
- *Border Gateway Protocol (BGP)* – Rutingprotokoll som brukes mellom ISPer. Store selskaper kan også bruke BGP.
- *OSPF* – eksempel på rutingalgoritme for mindre nettverk. Bygger et kart over mulige ruter til måladressene. Skalerer ikke for store nettverk.



# Ruting i Internett (svært kort)

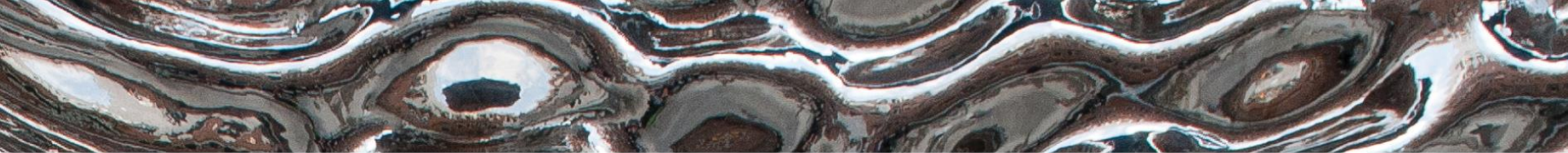
## Broadcast domain



# Traceroute – viser deg hvor trafikken går

- Kommandoen "traceroute" (tracpath på UiO Linux-maskiner) bruker en protokoll som heter "Internet Control Message Protocol"(ICMP) til å spore hvor datapakkene er innom på veien til målet.

```
26.10.2017-lagene spiller sammen — apetlund@nordur:~ — ssh login.ifi.uio.no — 8...
[apetlund@nordur ~]$ tracepath amazon.co.uk
1?: [LOCALHOST] pmtu 1500
1: ifi-gw21.uio.no 0.566ms
1: ifi-gw21.uio.no 0.503ms
2: uio-gw21.uio.no 0.692ms
3: uio-gw8.uio.no 0.367ms
4: oslo-gw1.uninett.no 0.481ms
5: se-tug.nordu.net 7.143ms
6: dk-uni.nordu.net 15.045ms asymm 7
7: dk-ore.nordu.net 15.104ms asymm 6
8: de-hmb.nordu.net 19.392ms asymm 7
9: de-ffm.nordu.net 27.891ms asymm 8
10: 52.95.216.56 24.836ms asymm 9
11: 54.239.107.36 34.809ms asymm 13
12: 54.239.107.21 27.731ms asymm 11
13: 54.239.5.27 46.082ms asymm 19
14: 176.32.106.248 47.040ms asymm 18
15: 52.93.36.165 46.711ms asymm 16
16: 54.239.44.162 65.882ms
17: 52.93.7.182 66.943ms asymm 18
18: 52.93.7.143 49.998ms asymm 16
19: 176.32.106.133 49.099ms asymm 17
^C
[apetlund@nordur ~]$
```



UiO : **Institutt for informatikk**

Det matematisk-naturvitenskapelige fakultet

# Transportlaget

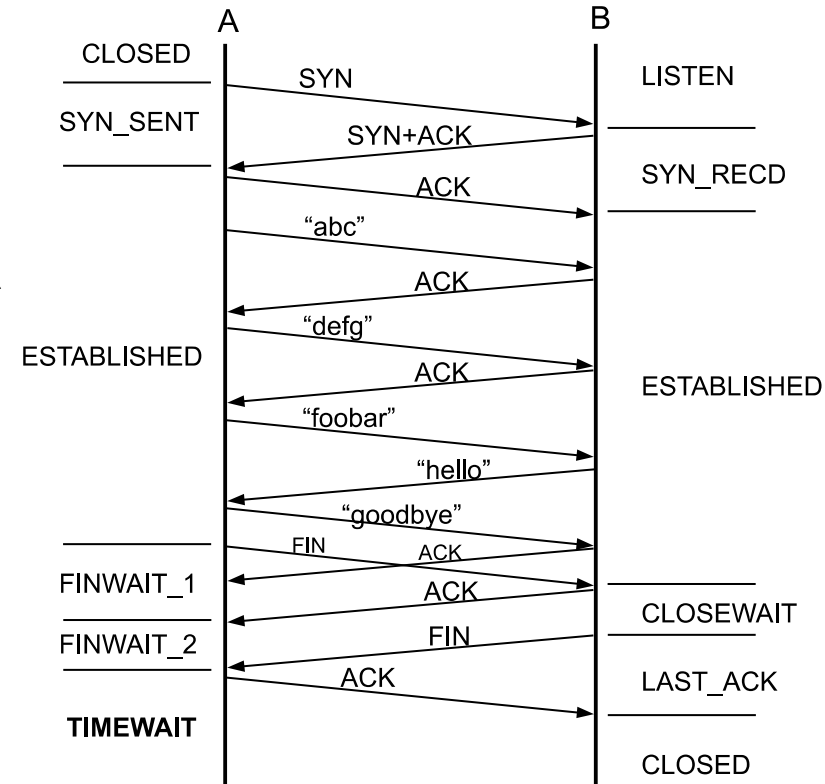


**simula**

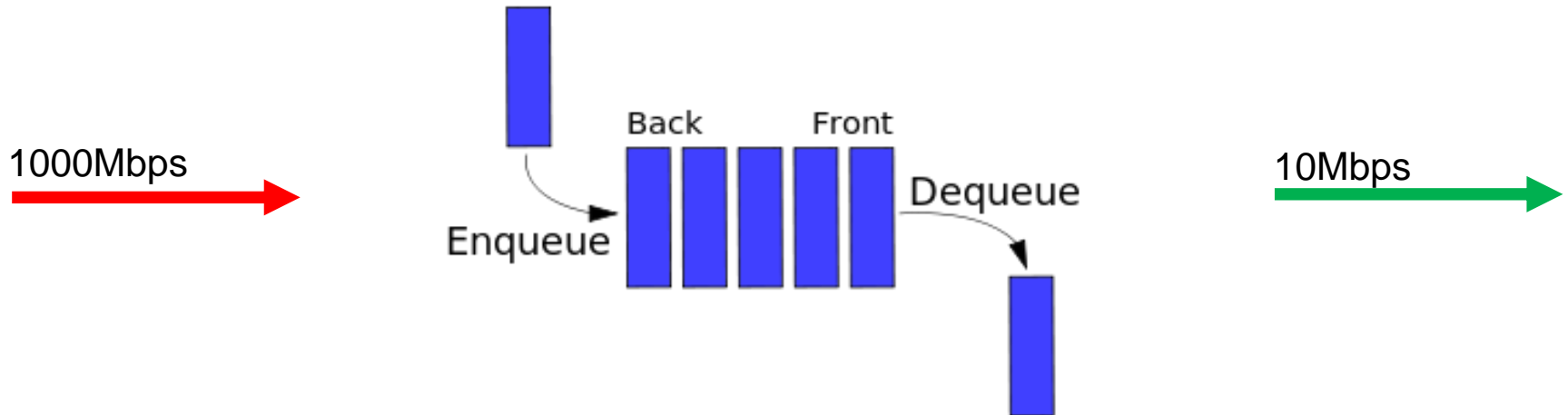


# Transmission Control Protocol (TCP)

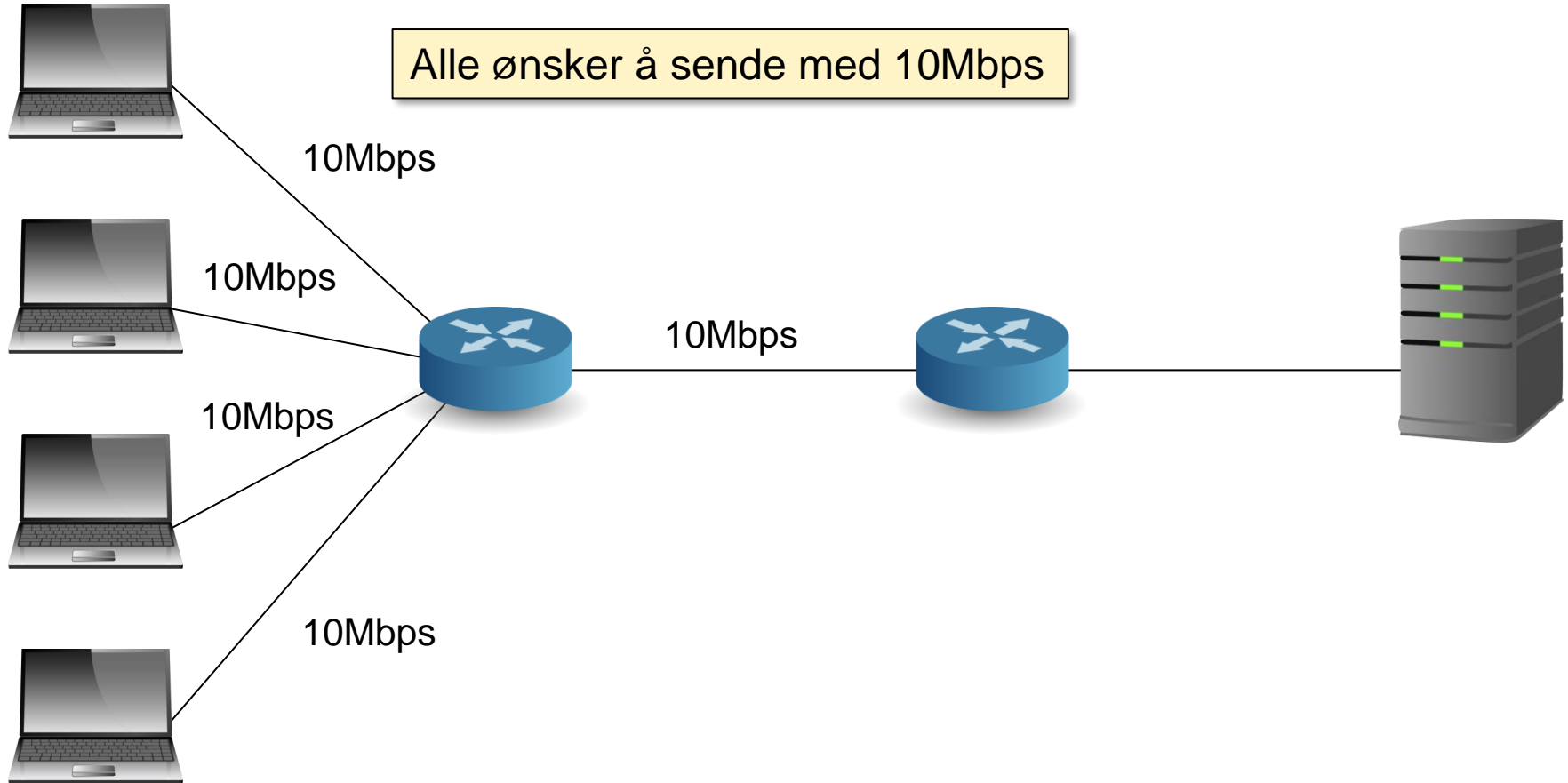
- Forbindelsesorientert
  - Settes opp ved et 3-veis-håndtrykk
    - SYN-SYN+ACK-ACK (se figur)
- Flytkontroll
  - Ikke sende fortere enn mottageren kan ta imot
- *Metningskontroll*
- Byte-strøm og levering i rekkefølge
- Pålitelighet
  - Implementert ved at bekreftelser på hver pakke sendes tilbake fra mottakeren
- Feilsjekking av nyttelasten (sjekksum)



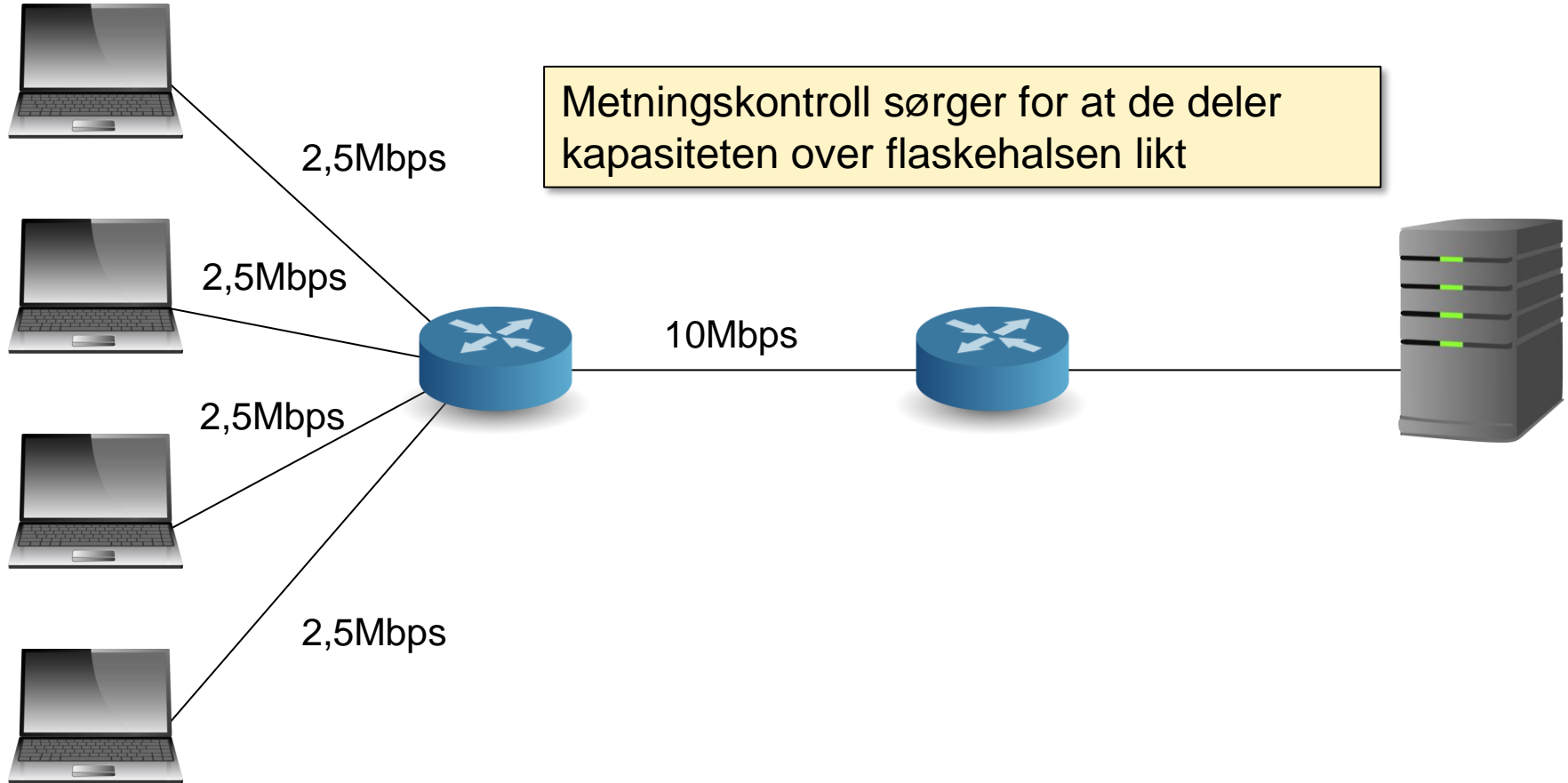
- En ruter er i utgangspunktet bare en FIFO (first-in-first-out) kø.
- Om det blir for mye trafikk over en gitt kø, vil det føre til stopp i trafikken. Dette kalles "congestion". Om det er så mye trafikk at det blir full stopp for alle, kalles det "congestion collapse".
- For å unngå dette bygde man inn mekanismer i TCP for å tilpasse senderaten til nettverksforholdene.



# fortsatt metningskontroll...

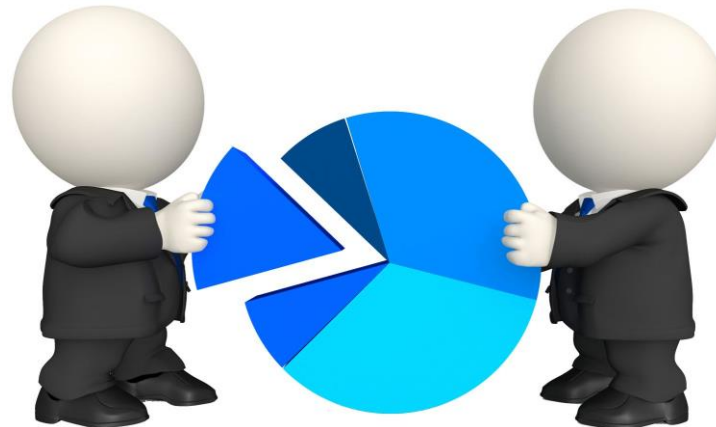


# ... ennå ikke ferdig med metningskontroll



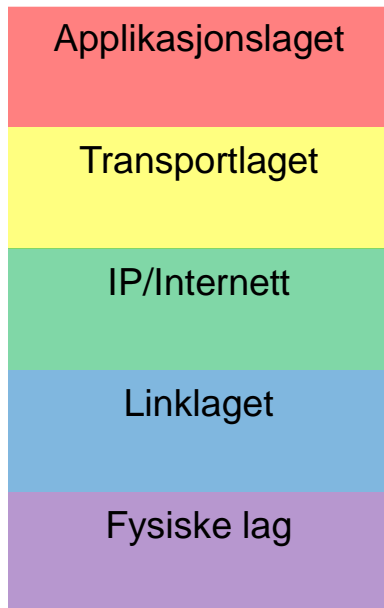
## ... stadig metningskontroll!

- Ressursene deles teoretisk sett likt mellom **forbindelser**
- For å kapre mer av kapasiteten: åpne flere forbindelser i parallell.
- Ingen god metode for å fordele ressurser i Internett rettferdig pr. maskin eller pr. bruker i dag.





# Kryptering / sikkerhet i lagene



*Secure Sockets Layer* – kryptering for ende-til-ende Applikasjoner – f.eks. nettbank eller butikker.

F.eks. *tcpcrypt*– har som mål at alle TCP-forbindelser som settes opp skal være kryptert

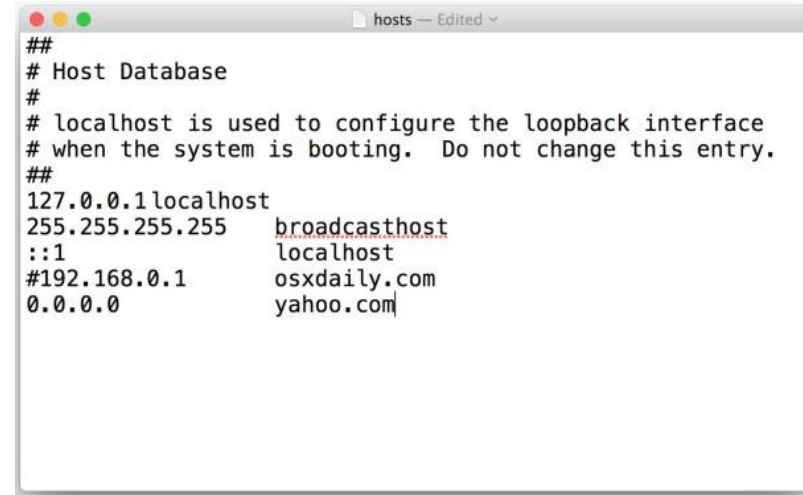
*VPN (IPSEC etc.)* – kobler to subnett sammen så det fungerer som ett LAN selv om de er fysisk adskilt

Kryptering på flere lag gjør det vanskeligere for uvedkomne å lytte til kommunikasjonen.

Adressen (avsender / mottakeren) er vanskelig å kryptere, da routere må vite hvor pakken skal leveres.

## Med IP-adresse?

- `telnet 127.0.0.1 23`  
snakker med min egen maskin
  - Brukes ofte da det er en veldefinert adresse
- `wget http://173.194.39.31:80/`  
snakker med en av Google sine maskiner mulig å huske, men ikke praktisk.
- `ssh 9.228.93.3`  
forsøke å kontakte en maskin du visste hadde denne adressen i 2001 umulig å huske om du ikke bruker den daglig
- det er mulig å lage alias for en IP-adresse i filen `/etc/hosts` (Unix variants)
- Opprinnelig administrert av Stanford Research Institute. Endringer ble distribuert via epost 😊



```
##
# Host Database
#
# localhost is used to configure the loopback interface
# when the system is booting. Do not change this entry.
##
127.0.0.1localhost
255.255.255.255 broadcasthost
::1 localhost
#192.168.0.1 osxdaily.com
0.0.0.0 yahoo.com
```

# Hvordan koble til en annen maskin?

Løsning: bruk “fornuftige” navn

- som f.eks.

```
ssh login.ifi.uio.no
```

```
wget www.google.com
```

- Ikke bare lettere å huske
- Har også en hierarkisk struktur (gjenspeiler organisasjonen)

Møt **Domain Name System (DNS)**



# DNS - overblikk

## Domain Name System

### Hierarkisk navnetilordning

I motsetning til den originale flate strukturen i /etc/hosts  
f.eks.: .com → google.com → mail.google.com

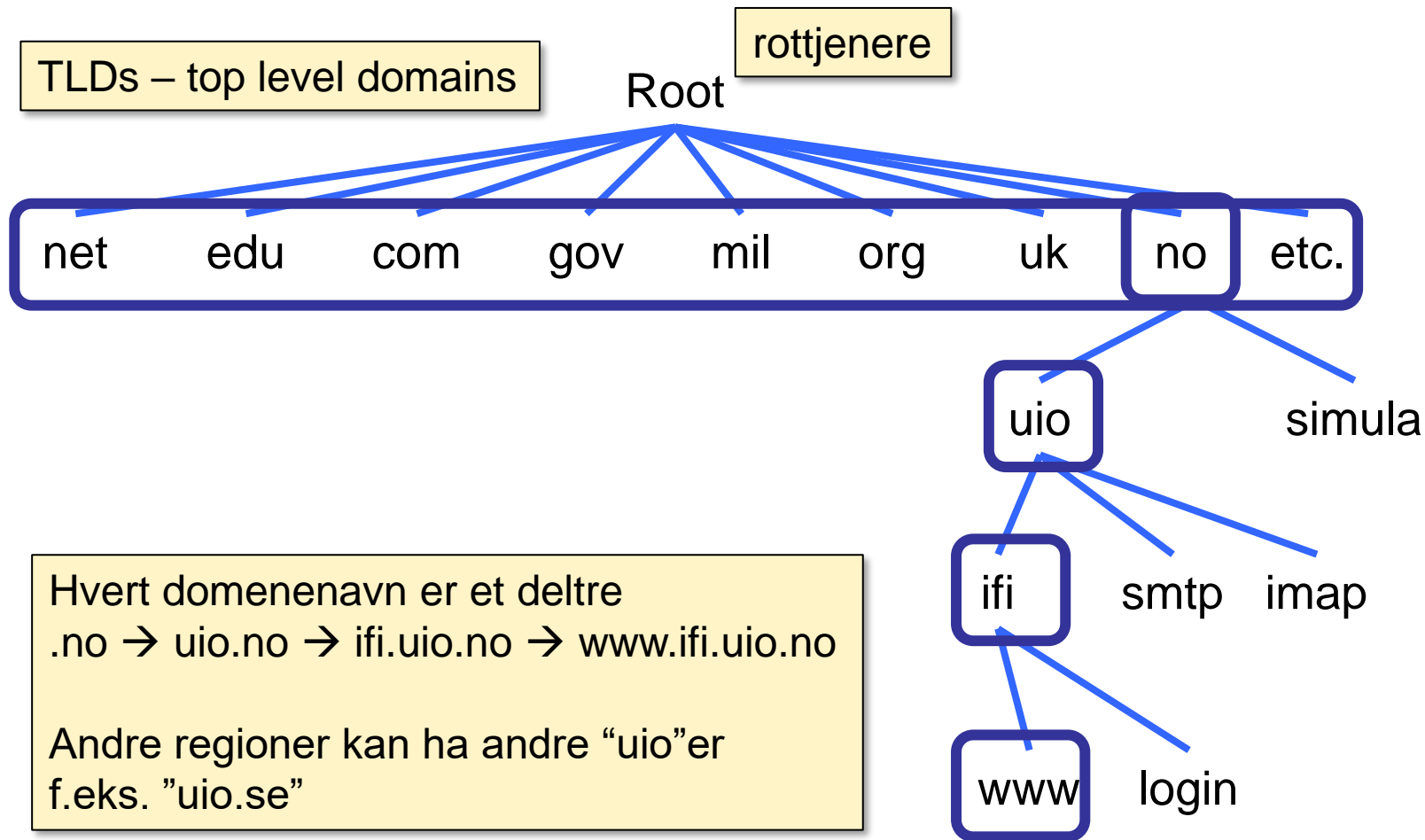
### Distribuert database

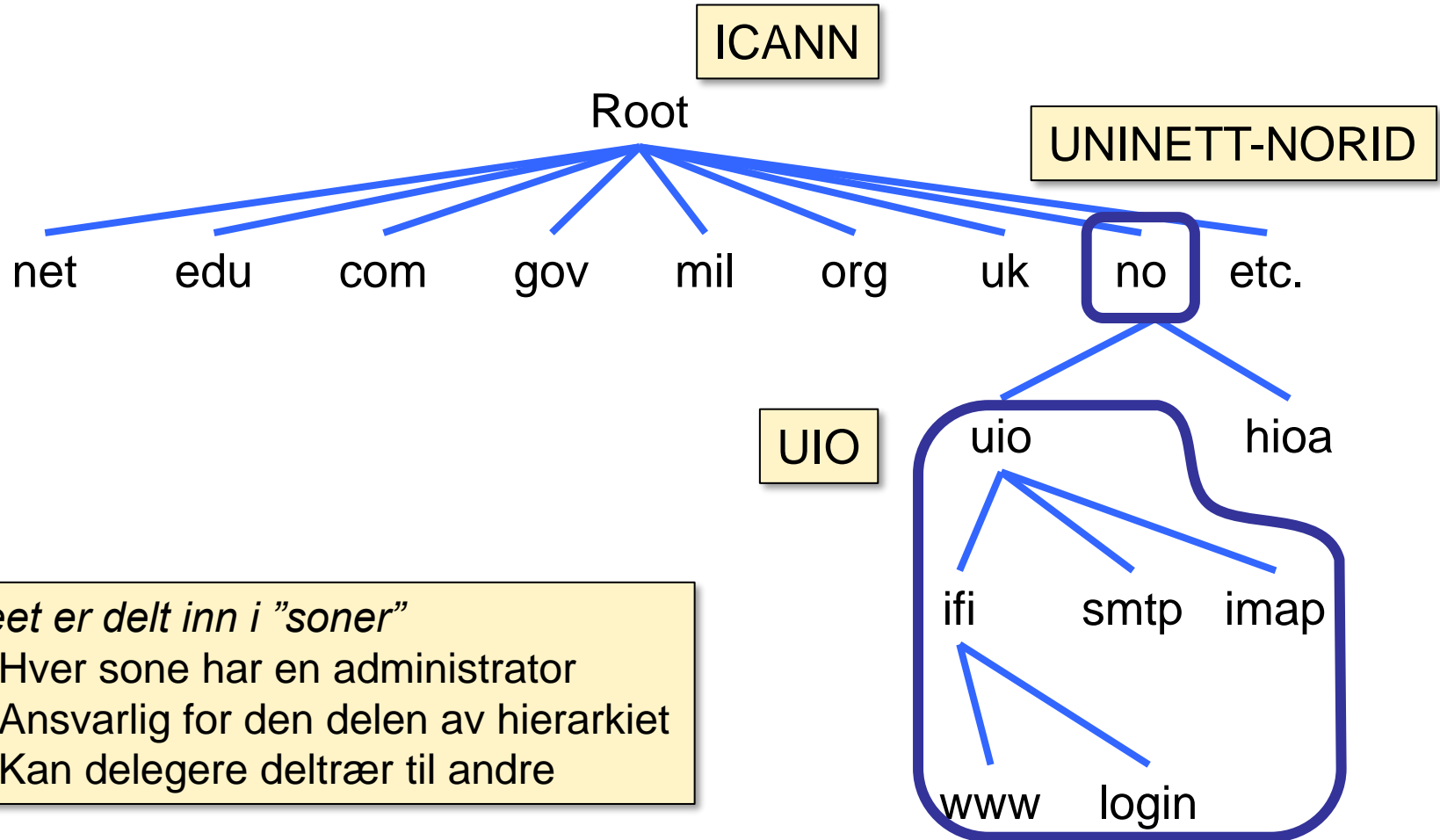
### Enkel klient/tjener arkitektur

- UDP eller TCP port 53
- tjenere må bruke TCP seg i mellom (fra nylig)
- klienter som bruker TCP avvises ofte
  - reduserer lasten på DNS-tjeneren



# Navnehierarki





*Treet er delt inn i "soner"*

- Hver sone har en administrator
- Ansvarlig for den delen av hierarkiet
- Kan delegere deltrær til andre

## Funksjonene til hver DNS-tjener

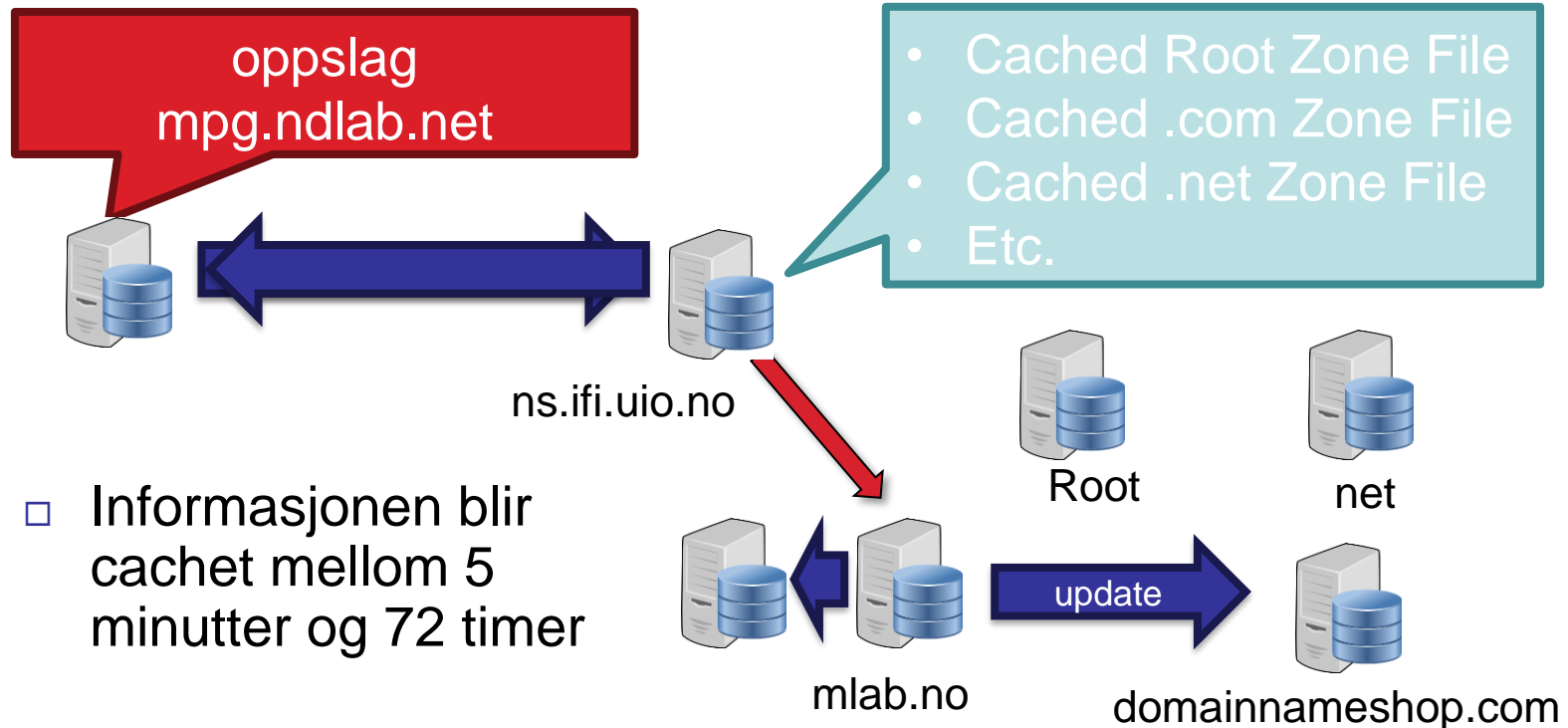
- Autoritet over en del av hierarkiet
  - Ikke behov for å lagre alle DNS-navn
- Lagre alle oppføringene for maskiner/domener i sin sone
  - **Må** replikeres for å sikre opetid (minst 2 tjenere)
- Må kjenne adressene til rottjenerne
  - Slå opp forespørsler på navn som den ikke lagrer selv



Rottjenerne kjenner til alle TLDene (Top Level Domains)

# Caching vs. oppdaterte data

- Caching reduserer forsinkelsen for et DNS-oppslag
- Caching reduserer lasten på DNS-tjenerne
- Caching forsinket oppdateringer





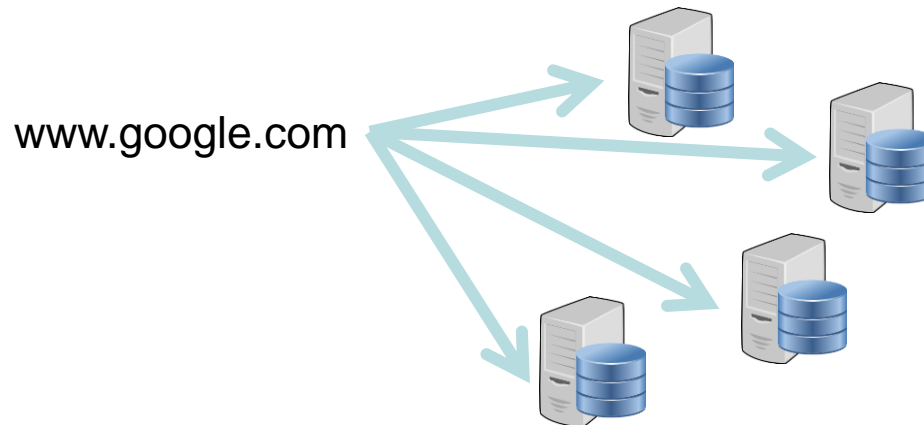
# Alias og lastbalansering

Én maskin kan ha mange alias



Ett domene kan kobles til mange maskiner

Eksempler:  
*k.root-server.net*  
og  
*login.ifi.uio.no*



# Nyttige nettverksverktøy

Verktøy	
ping	Måler forsinkelsen mellom din maskin og en annen på internett
tracert	Gir informasjon om hvert hopp en datapakke er innom på vei til målet.
tracert	Samme som tracert, men trenger ikke superusertilgang
netstat	Gir informasjon om nettverksforbindelser på din maskin
dig	Gjøre DNS-oppslag og se hva som returneres.
whois	Slå opp hvem som eier/disponerer en IP-adresse
tcpdump / Wireshark	Logge pakker som går inn og ut av maskinen, samt utføre analyse

## Ekstramateriale:

- **Wikipedia (gode artikler på disse temaene):**
  - Internet Protocol Suite – med link til de andre komponentene:  
[https://en.wikipedia.org/wiki/Internet\\_protocol\\_suite](https://en.wikipedia.org/wiki/Internet_protocol_suite)
- **Bøker og artikler:**
  - Tanenbaum, Andrew S. [Computer Networks](#). Prentice Hall PTR
  - James F. Kurose, Keith W. Ross. [Computer Networking: A Top-Down Approach](#)

