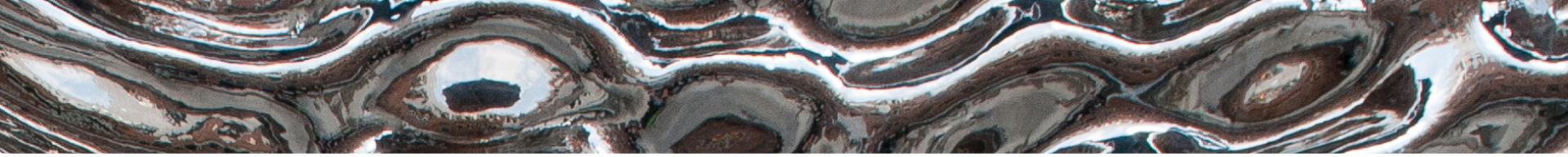


UiO : Institutt for informatikk
Det matematisk-naturvitenskapelige fakultet

IN1020 - Introduksjon til datateknologi

Forelesning – 23.11.2018

Oppsummering

Håkon Kvale Stensland & Andreas Petlund

simula

tfj

Nettverksdelen - Pensum

- Relevante kapitler fra boka (se pensumliste)
- Alt presentert på forelesningene
- Ukeoppgaver
- Obligatorisk oppgave 3 *(ikke pakketrace fra Wireshark)*

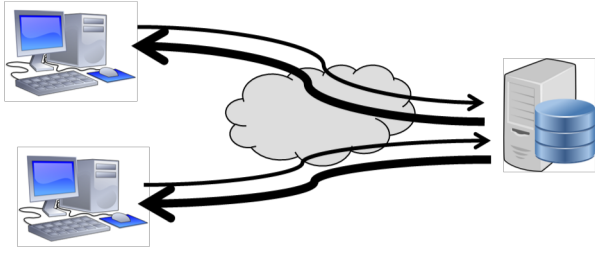
- **NB!** Tema som ikke nevnes i denne oppsummeringen er allikevel pensum!

Protokoller i nettverk

- En protokoll definerer strukturen på beskjeder sendt over et nettverk
- Hvorfor trenger vi protokoller?

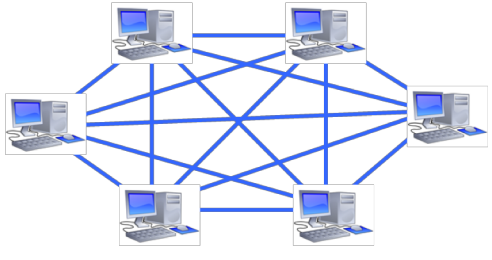
Aksessmodeller: Klient-tjener

- Klienter ber om en tjeneste (oppretter en forbindelse)
- Tjenere leverer tjenesten (svarer på forespørselen)



Aksessmodeller: Peer-to-Peer (P2P)

- Alle noder er likeverdige
- Alle noder kan nå hverandre
- Wierskapet er distribuert

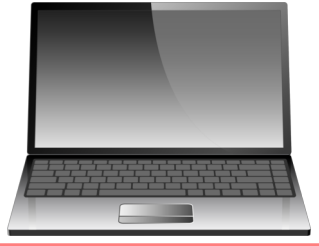


TCP/IP Referansemodellen

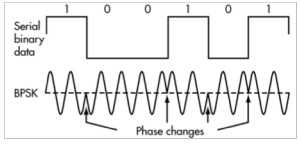
- Kjenne til egenskaper ved de forskjellige lagene.
  - Protokoller fra pensum som benyttes på forskjellige lag.
- Hvordan beveger data seg igjennom lagene over internett.
  - Hva legger de forskjellige lagene på av informasjon?

Lag		Funksjon
5	**Applikasjon**	Applikasjonsrelaterte tjenester
4	**Transport**	Kobler sammen systemene ende-til-ende (TCP/UDP)
3	**Nettverk**	Rute data fra ende-til-ende systemer (IP)
2	**Link**	Pålitelig overføring mellom to noder
1	**Fysisk**	Sender bit ut på mediet (kablet eller trådløst)

Lagene i Internett (TCP/IP referansemodellen)

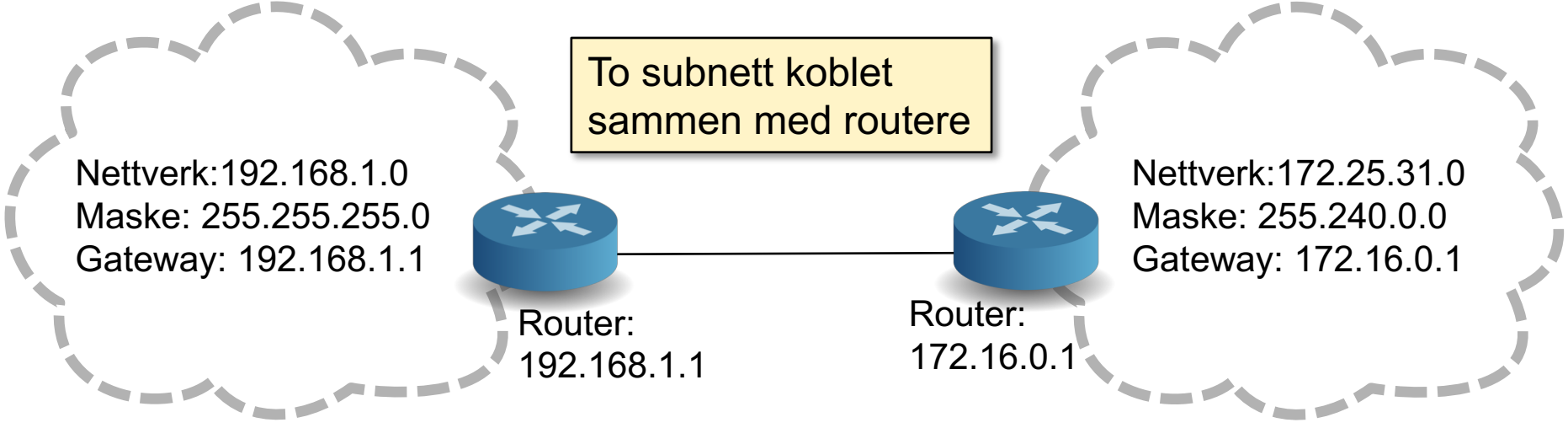


Applikasjonslaget	**http://www.uio.no**		Applikasjonslaget
Transportlaget	**192.168.1.5:80**		Transportlaget
IP/Internett	**192.168.1.5**	IP/Internett	IP/Internett
Linklaget	**A1:B2:C3:D4:E5:F6**	Linklaget	Linklaget
Fysiske lag		Fysiske lag	Fysiske lag

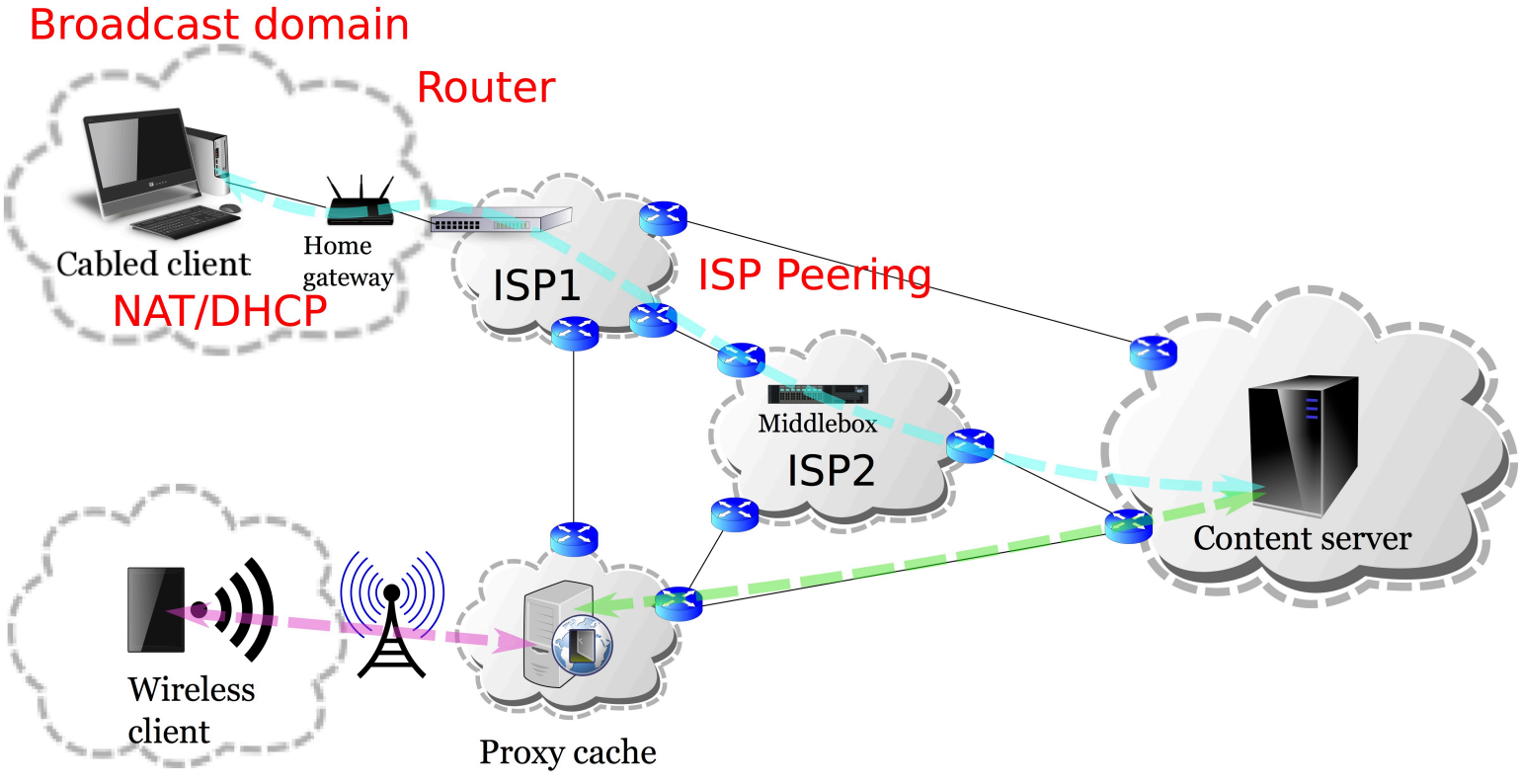


Lokalnettverk (LAN), subnett og broadcast

- Internett er en sammenkobling av mindre, separate nettverk.
- Koblet sammen med switcher og/eller HUBer.
  - Kunne regne seg frem til nettmaske og broadcast adresser.

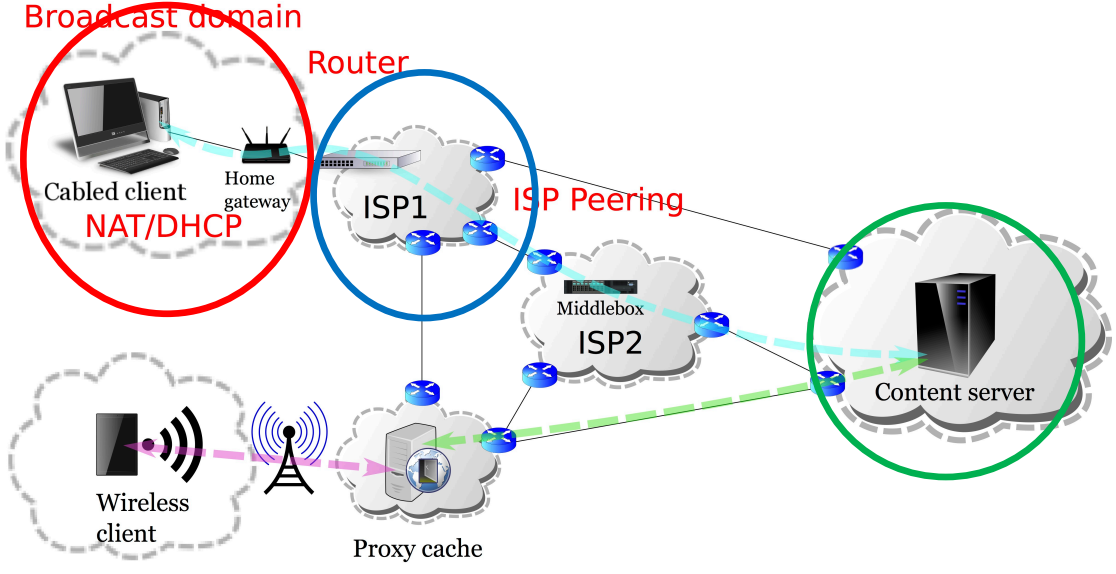


En pakkes vei gjennom nettet



Fysisk plassering av innholdet

Cachenivå	Fysisk beliggenhet	Est. RTT
Lokal Proxy	Organisasjon / LAN	< 10ms
Content Delivery Network	ISP	< 50ms
Original datakilde	Internett	~10ms - ~250ms



ARP – Koblingen mellom nettverk og IP

- Kjenne til hvordan ARP fungerer, og hvorfor vi trenger denne protokollen.
- 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).

Én IP-adresse – mange porter

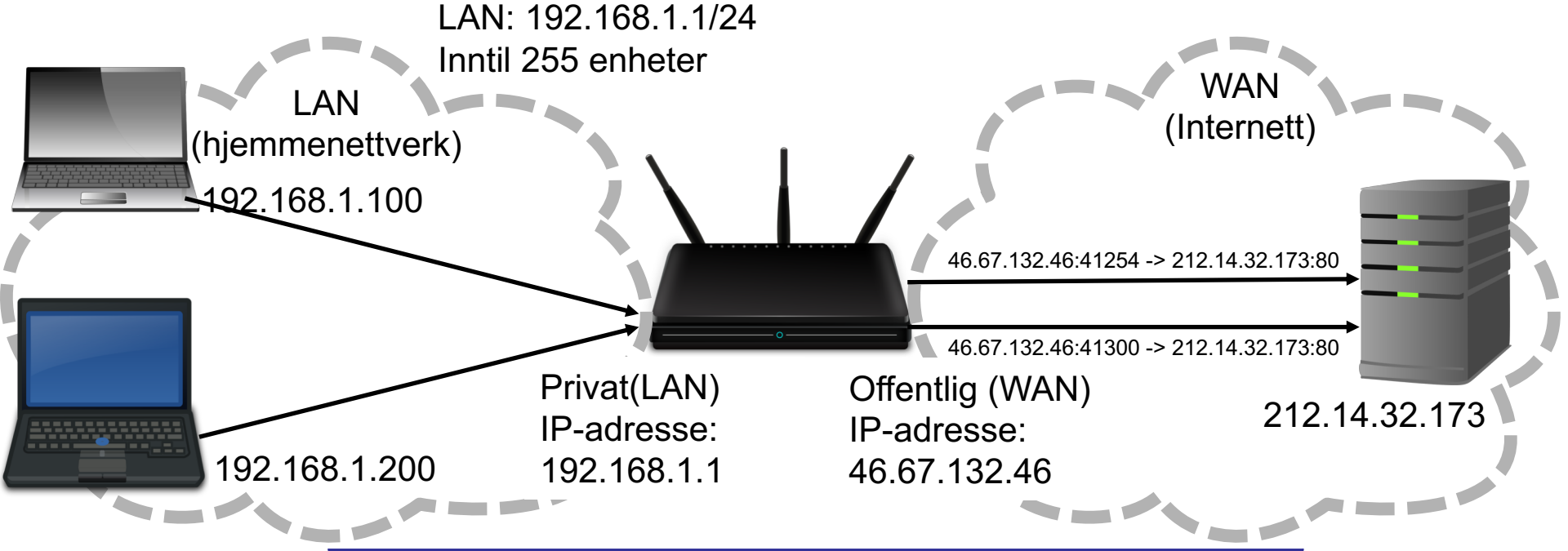
Hvordan kan en IP adresse brukes til mange tjenester?
Adressering i transportlaget.

IP: 192.168.1.5

Port	Tjeneste
0	Reservert
1	tcpmux
…	
22	SSH
…	
25	SMTP
…	
1024-49151	Brukerporter
49152-65535	Dynamisk / privat

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

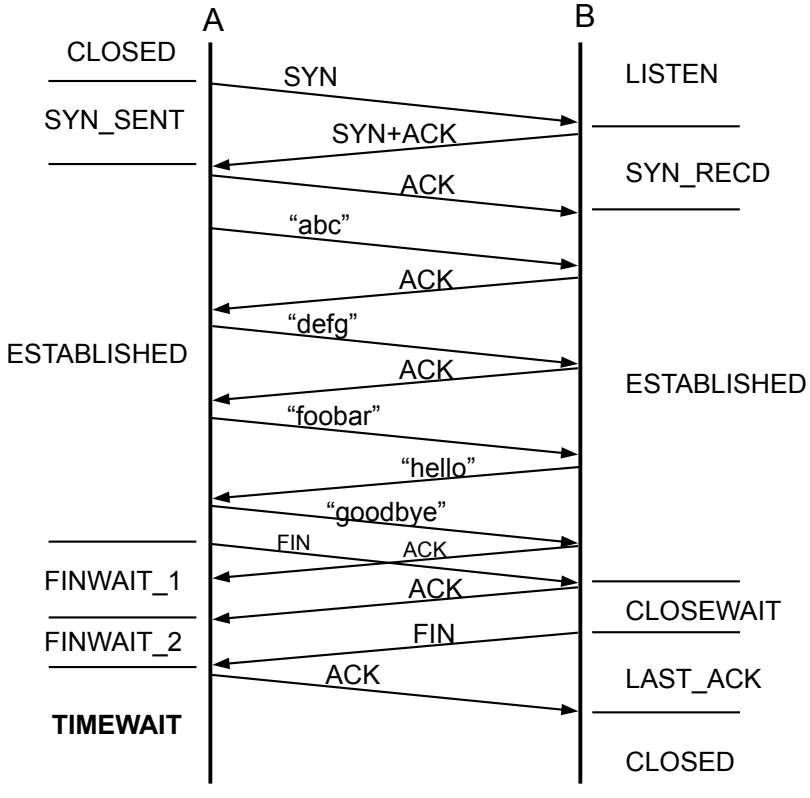
NAT – Network Address Translation



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

Transmission Control Protocol (TCP)

- Forbindelsesorientert
- Flytkontroll
- Metningskontroll
- Byte-strøm og levering i rekkefølge
- Pålitelighet
  - Implementert ved at bekreftelser på hver pakke sendes tilbake fra mottakeren
- Feilsjekking av nyttelasten (sjekksum)



Hvordan koble til en annen maskin?

- Løsning: bruk "fornuftige" navn
  - som f.eks.
    `www.google.com`

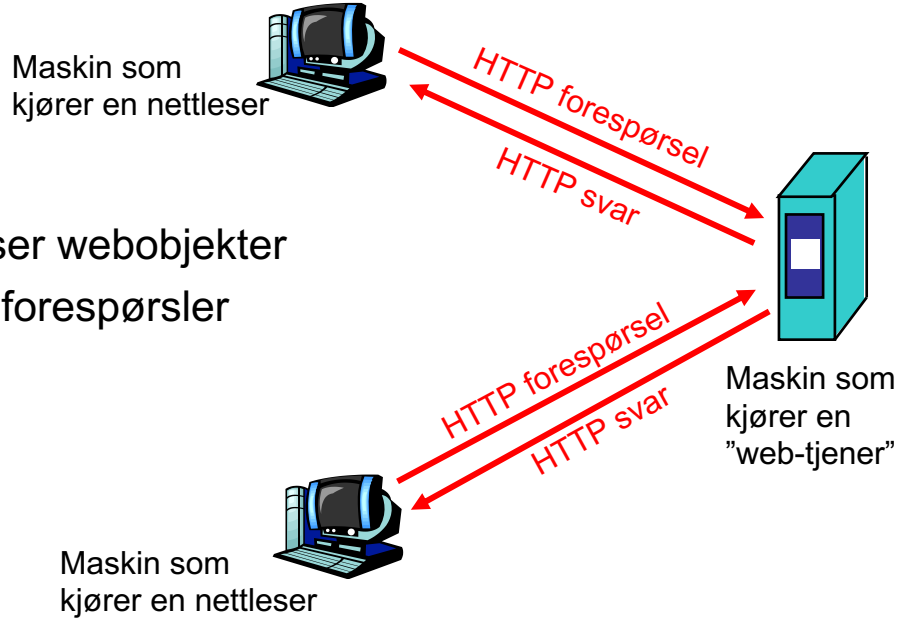
- **Domain Name System (DNS)**
  - Hierarkisk navnetilordning
    - f.eks.: .com → google.com → mail.google.com
  - Distribuert database
    - Top-level domains (TLDs)
    - Rottjenere
  - Enkel klient/tjener arkitektur

World Wide Web (www): HTTP-protokollen

HTTP: **H**yper**T**ext **T**ransfer **P**rotocol

- Applikasjonslagsprotokollen for Web
- Klient-/tjenermodell
  - *Klient:* nettleser som spør etter, får og viser webobjekter
  - *Tjener:* sender webobjekter som svar på forespørsler

- Tre hovedversjoner:
  - HTTP/1.0 (1990)
  - HTTP/1.1 (1999)
  - HTTP/2 (2015)



HTTP-protokollen

HTTP: bruker TCP som transport:

- Klienten oppretter en TCP-forbindelse (socket) til tjeneren, port 80
- Tjeneren godtar TCP-forbindelsen fra klienten
- HTTP-meldinger (protokollmeldinger på applikasjonslaget) utveksles mellom nettleseren (HTTP-klient) og Webtjeneren (HTTP-tjener)
- TCP-forbindelsen lukkes

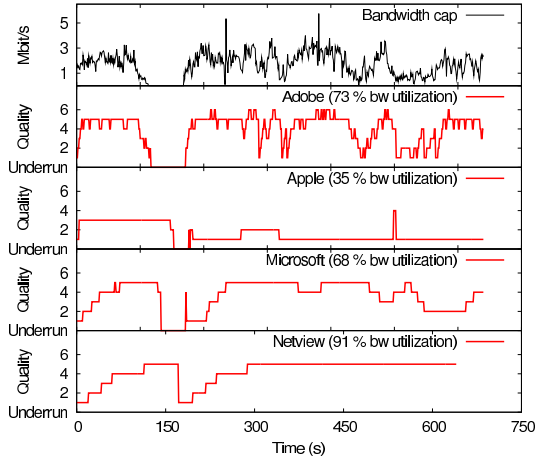
HTTP er “stateless”

- Tjeneren sparer ikke på tilstandsinformasjon om tidligere forespørsler

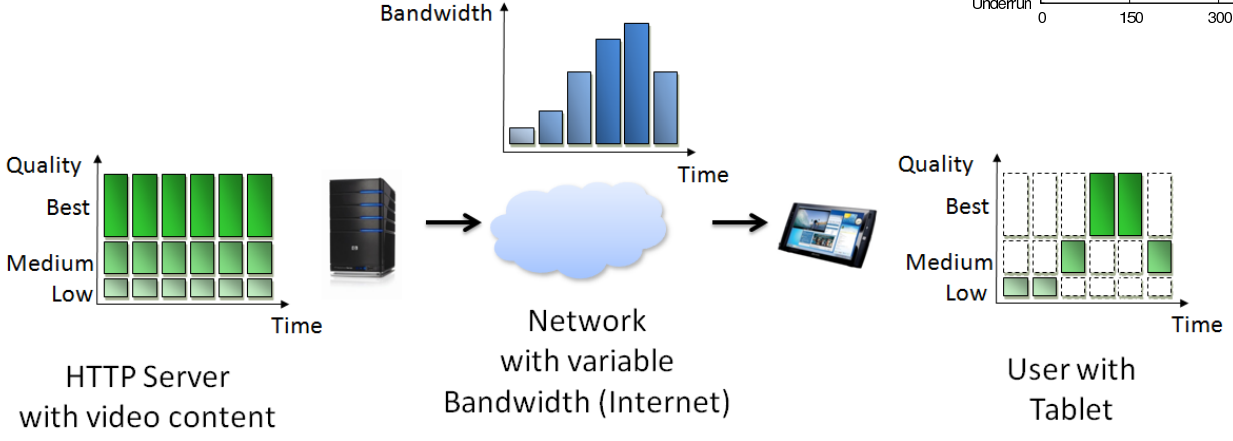
Protokoller som sparer på ”tilstand” er komplekse!

- Tilstanden må vedlikeholdes
- Om en tjener eller klient ”kræsjer”, kan tilstanden bli ulik mellom dem. Da må den gjennoprettes.

HTTP-streaming



Dynamisk, Adaptiv Streaming over HTTP (DASH)



- Video 50-70% av trafikken på Internett (telt i Bytes)
- En konstant strøm av data (til avspilling slutter)
- Dele videoen i segmenter: fullstendig uavhengige små filmer.

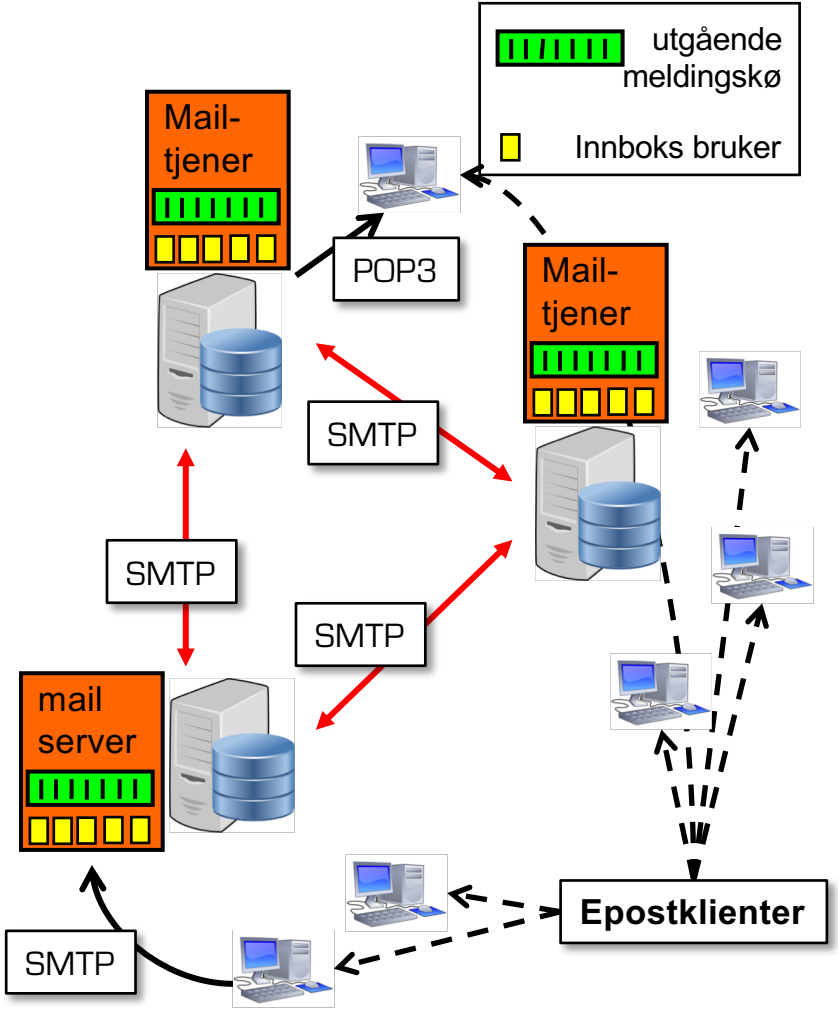
Tjenester: Epost

Mailtjenere:

- *mailbox* inneholder innkommende meldinger (hittil uleste) til brukeren
- *meldingskø* av utgående epostmeldinger (for sending)

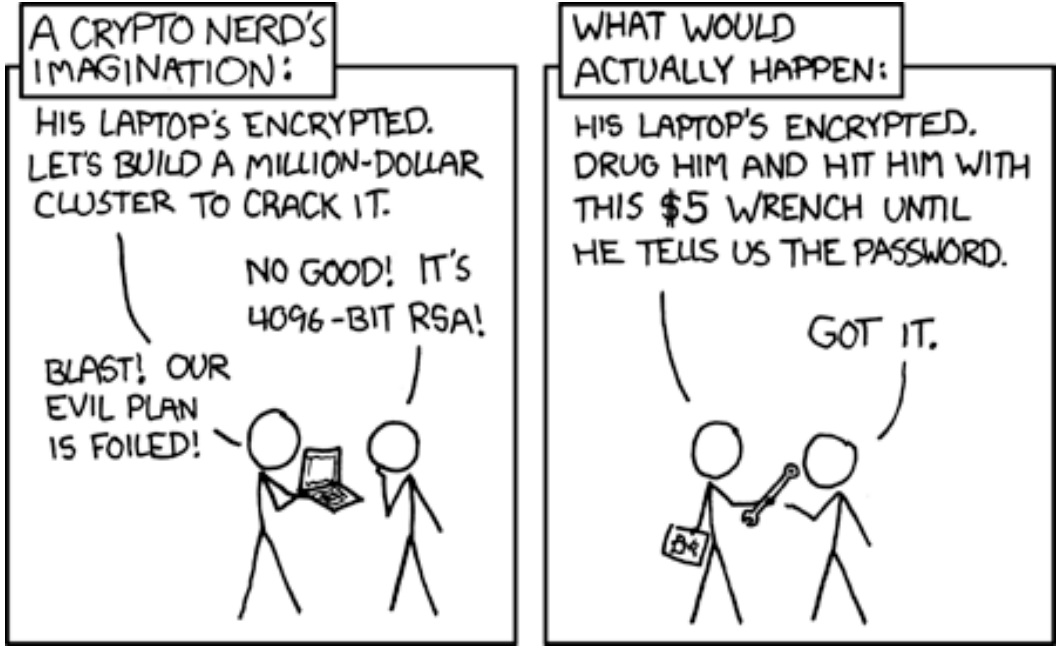
Simple Mail Transfer Protocol (SMTP):

- Mellom eposttjenere for å sende epostmeldinger
- klient: tjener som skal sende en epost
- tjener: den som mottar eposten

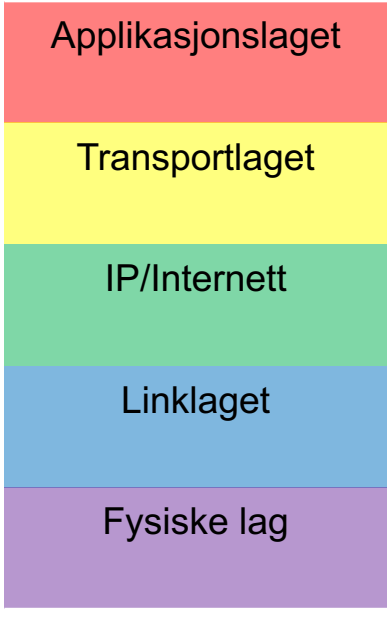


Kryptografi

- Hvorfor trenger vi kryptografi?
- Forskjellige teknikker for kryptering:
  - Hemmelig nøkkel (symmetrisk) kryptering.
  - Offentlig nøkkel (asymmetrisk) kryptering.
- Hash-algoritmer.



Kryptering / sikkerhet i nettverket



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.