Velkommen!

Tjenester i internett. Hvordan utnytte datasystemer

Gruppetime 11, 9. november 2021

(Hanne-)Kristin Schrøder Leiros

hkleiros@uio.no

Plan for dagen

- Repetisjon av pensum
  - Tjenester i internett
  - Hvordan utnytte datasystemer
- Ukesoppgaver
  - 8 stk
- Eksamensoppgaver om datasikkerhet

Tjenester i internett

Applikasjonslaget

Hva som ble gjennomgått

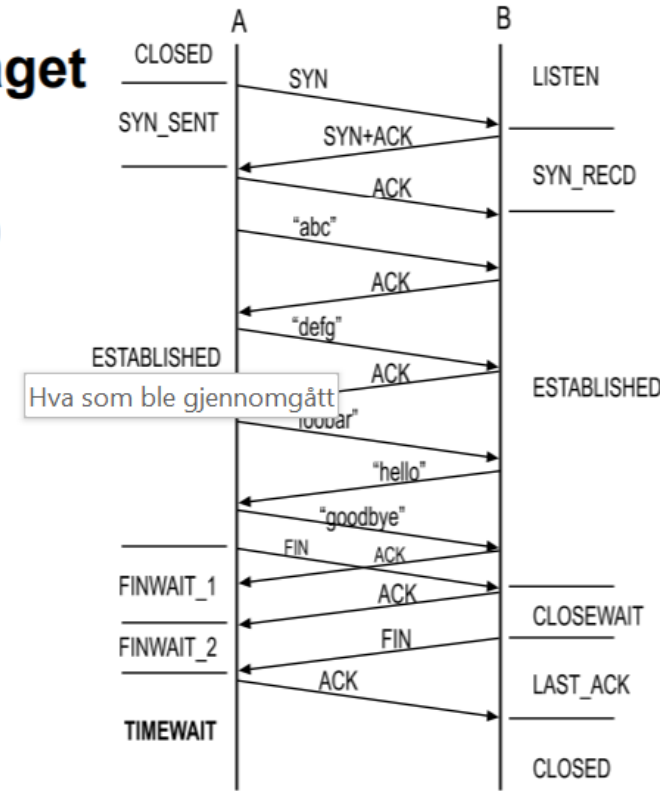
- **Repetisjon**
  - **Transportlaget**
- Forbindelsesstrategier
  - **Aksessmodeller**
    - **Klient-tjener**
      - Content Delivery Network
    - **Peer-to-Peer (P2P)**
- World Wide Web (www): HTTP-protokollen
- Strømming av video
- E-post
- Internet of things

Rep: Transportlaget

- TCP: HTTP, E-post, filoverføring, etc
- UDP: Strømming av video og lyd

Repetisjon: **Lag 4 - Transportlaget**

- **TCP:**
  - Oppsett av forbindelse (3-way handshake)
  - Pakkene leveres i riktig rekkefølge
  - *Pålitelighet* – Pakker sendes på nytt hvis kvitteringen (ACK) ikke kommer frem
  - Flytkontroll og metningskontroll
  - Feilsjekking av nyttelasten (sjekksum)
- **UDP:**
  - Tilkoblingsløs forbindelse
  - Ingen garantier, «Best-effort»
  - Feilsjekking av nyttelasten (sjekksum)



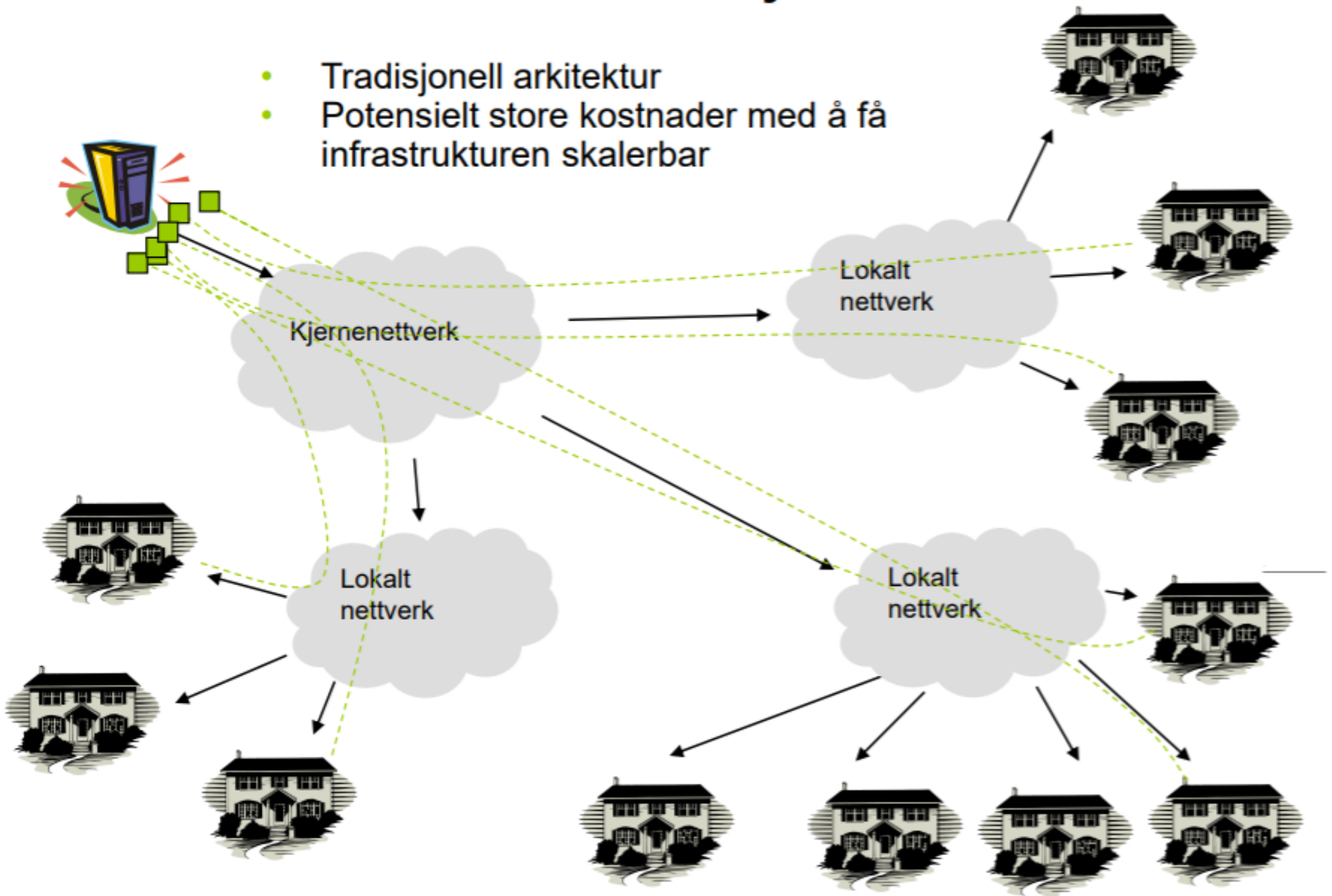
Hva som ble gjennomgått

Forbindelsesstrategier

- Pull
  - Klienten ber tjeneren om en tjeneste
  - Tradisjonelt den vanligste metoden
- Push
  - Tjeneren «dytter» en tjeneste (f.eks. en beskjed) til klienten
  - Krever at det er en forbindelse fra før, eller at klienten lytter
- Push
  - Variant av «push» der tjeneren dytter ut beskjeder til en gruppe av abonnenter

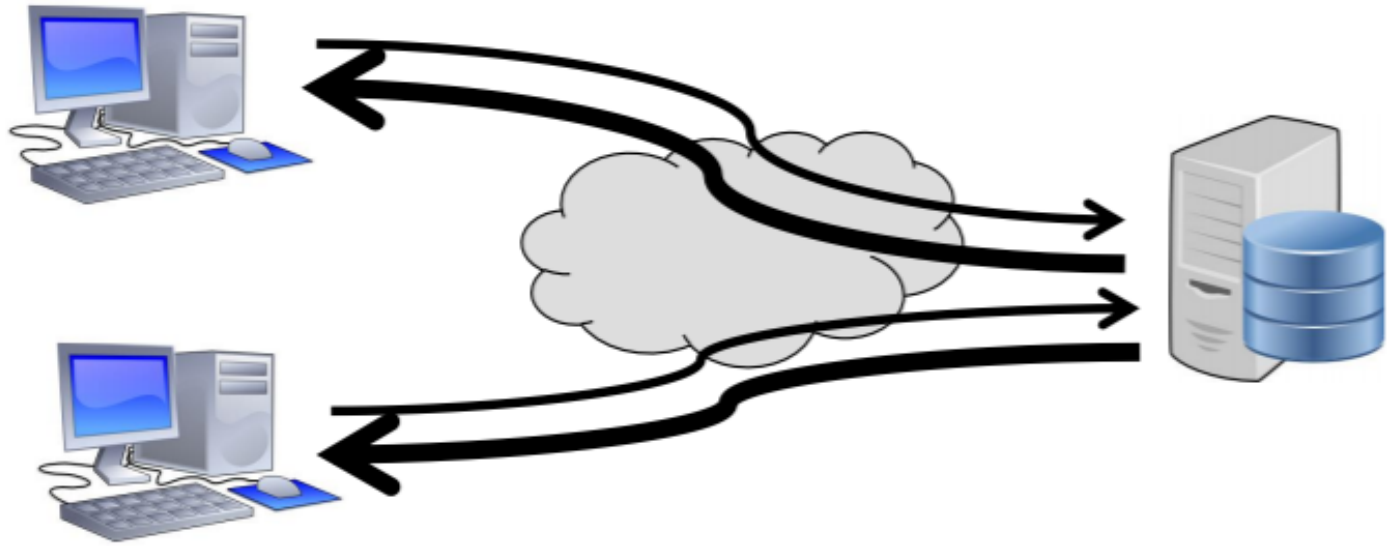
Klient-Tjener

- Tradisjonell arkitektur
- Potensielt store kostnader med å få infrastrukturen skalerbar



Klient-tjener

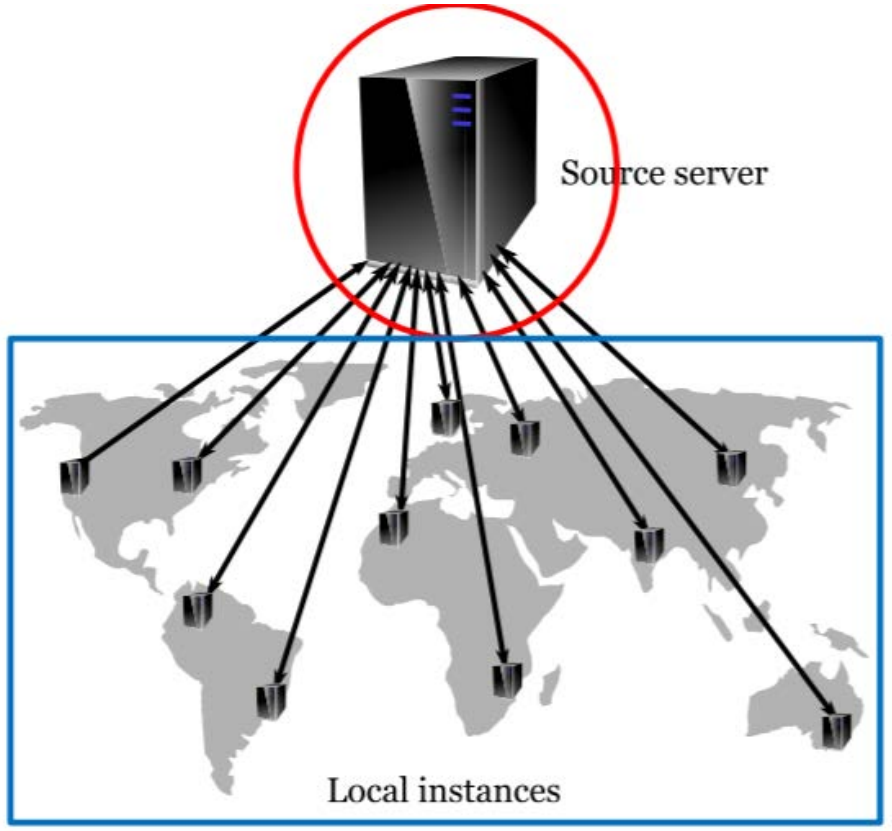
- Tradisjonell kommunikasjonsmodell
  - Klienter ber om en tjeneste
    - Oppretter en forbindelse
  - Tjenere leverer tjenesten
    - Svarer på forespørselen
- Eks: Webklient(nettleser)/Webtjener, Mailklient/tjener



Content Delivery Network – CDN

- Løse Klient/tjener- modellens skaleringsproblem med at tjenerne ligger langt unna
- Innhold som skal levers til klienter over hele verden legges til maskiner som ligger fysisk nærmere brukerne
- - Koster ekstra maskinvare og lagringsplass
- + Sparer kapasitet i backbone-nettet
- + Gir lavere forsinkelse på forespørsler
- + Hindrer overbelastning av tjeneren

- Samme tanke bak som minnehierarki i datamaskiner

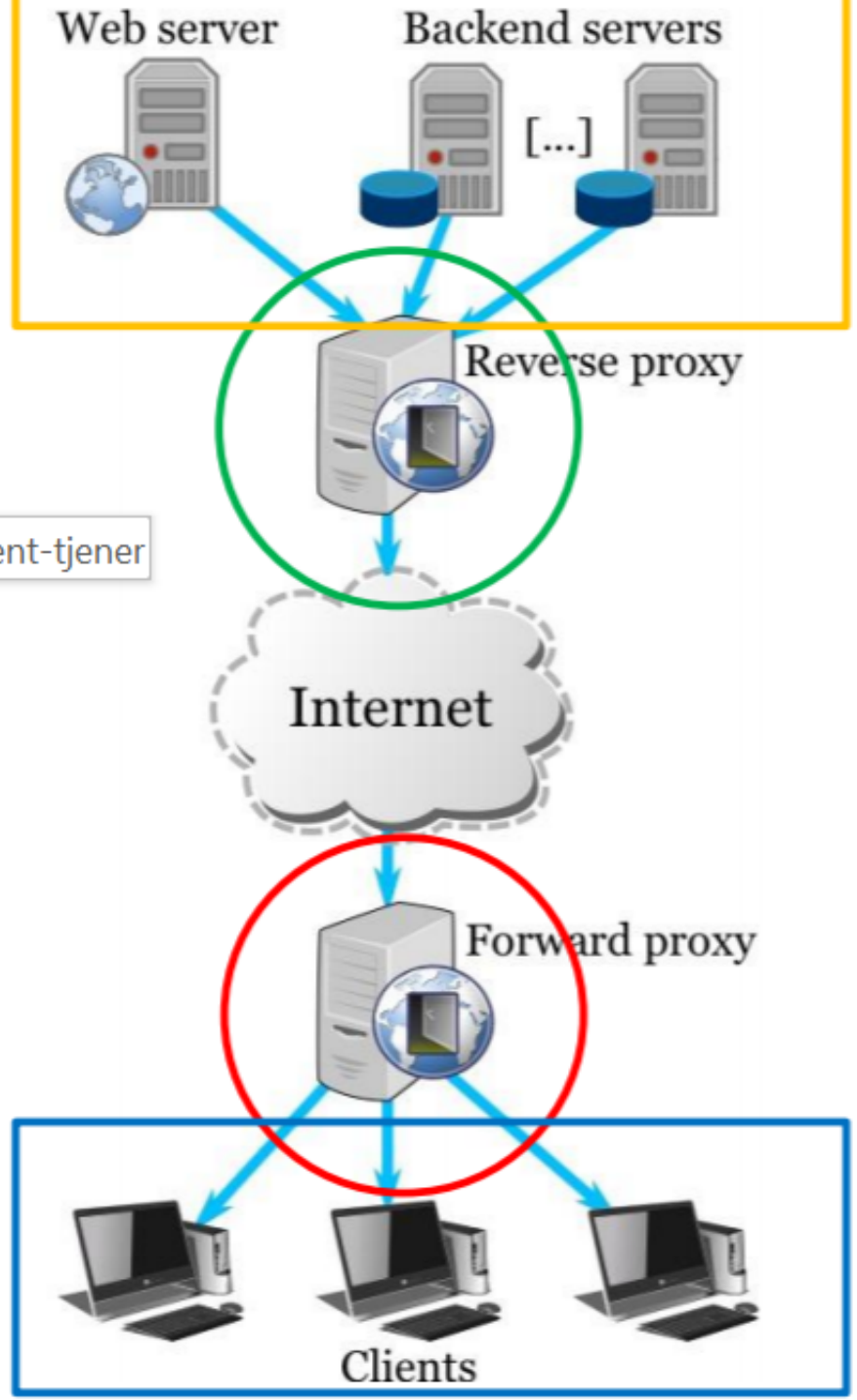


"B. Briscoe *et al*., "Reducing Internet Latency: A Survey of Techniques and Their Merits," in *IEEE Communications Surveys & Tutorials*

Proxy-cache

- En ”Forward Proxy” står nær klienten
  - og mellomlagrer data for klientene, slik at de ikke behøver å gå helt til kilden.
- En ”Reverse Proxy” står nær tjeneren
  - og mellomlagrer data fra én eller flere tjenere, slik at klienten slipper å gå helt til kilden(e).

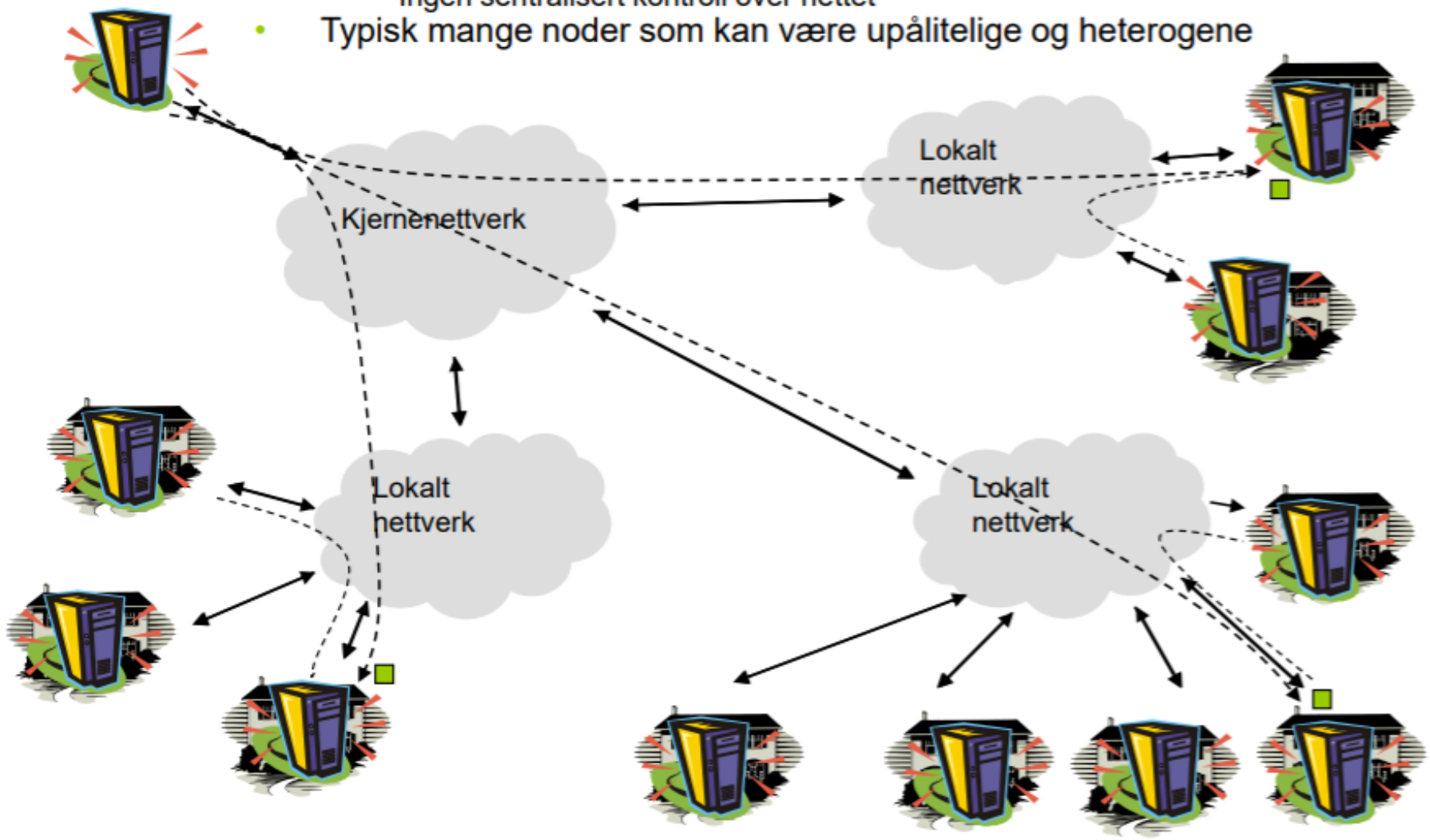
+ Lastbalansering
+ Sparer nettverkskapasitet
+ Lavere forsinkelse



B. Briscoe *et al.*, "Reducing Internet Latency: A Survey of Techniques and Their Merits," in *IEEE Communications Surveys & Tutorials*

Aksessmodeller: Peer-to-Peer (P2P)

- *Ikke en ny ide…- en distribuert systemarkitektur*
  - Ingen sentralisert kontroll over nettet
- Typisk mange noder som kan være upålitelige og heterogene



Peer-to-Peer – P2P

- På flere måter likt CDN
  - Endesystemer kan være både klient og tjener
  - Distribuert lagring
  - Innholdet nærmener sluttbruker
  - Begrenset lagring per endesystem
- Ulikheter:
  - Ingen klare roller på endesystemene
  - Ikke noe hierarkisk forhold
  - Klienter vet nødvendigvis ikke hvor innholdet er
    - Trenger protokoller for å «oppdage innhold»
  - Endesystemer i et P2P nettverk kommer og går

P2P: styrker og svakheter

Styrker

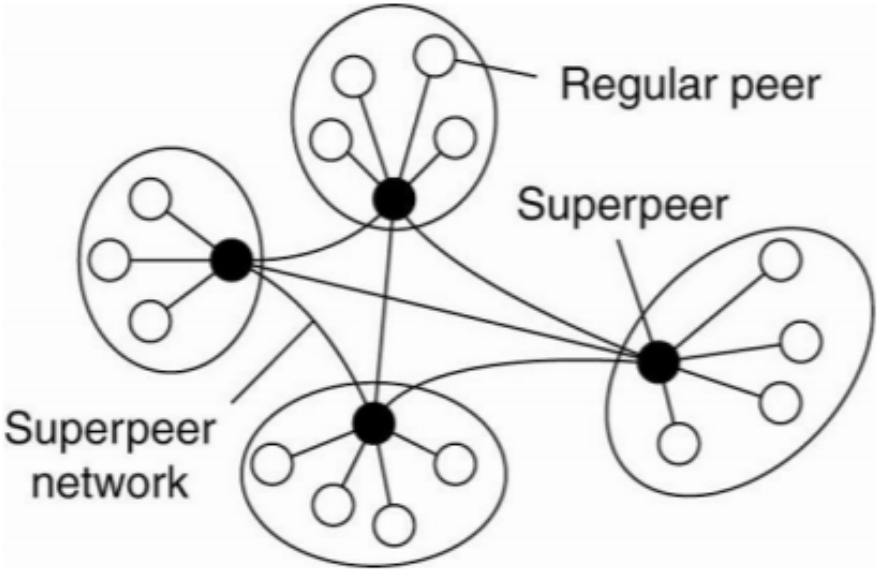
- Ideelt til å dele ressurser mellom enheter
- Ressurser som kan deles:
  - CPU
  - Lagringsplass
  - Nettverk
  - Kombinasjoner av disse
- Tilgangen til ressurser styres av algoritmer
- Robust mot endringer i nettverket

Svakheter

- Dårlig til applikasjoner som er avhengig av lav forsinkelse
- Om man er avhengig av en viss grad sentral kontroll
- Peers kommer og går
  - Trenger mye redundans og ressurser for å sikre kontinuitet
- Kan være vanskelig å effektivt indeksere og finne igjen elementer

Varianter av P2P-baserte topologier

- Ren P2P:
  - Alle peers likeverdige
- Hybrid P2P:
  - Noen sentrale noder som har mer funksjonalitet enn andre
    - sentrale tjenere eller
    - "superpeers"
  - Kan være valgt fordi:
    - De ikke står bak brannmur og/eller NAT
    - De har mye av en ressurs, f.eks. båndbredde, harddiskplass, CPU-regnekraft
  - Slike løsninger kan hjelpe til f.eks. å få ned forsinkelsen i en P2P-løsning





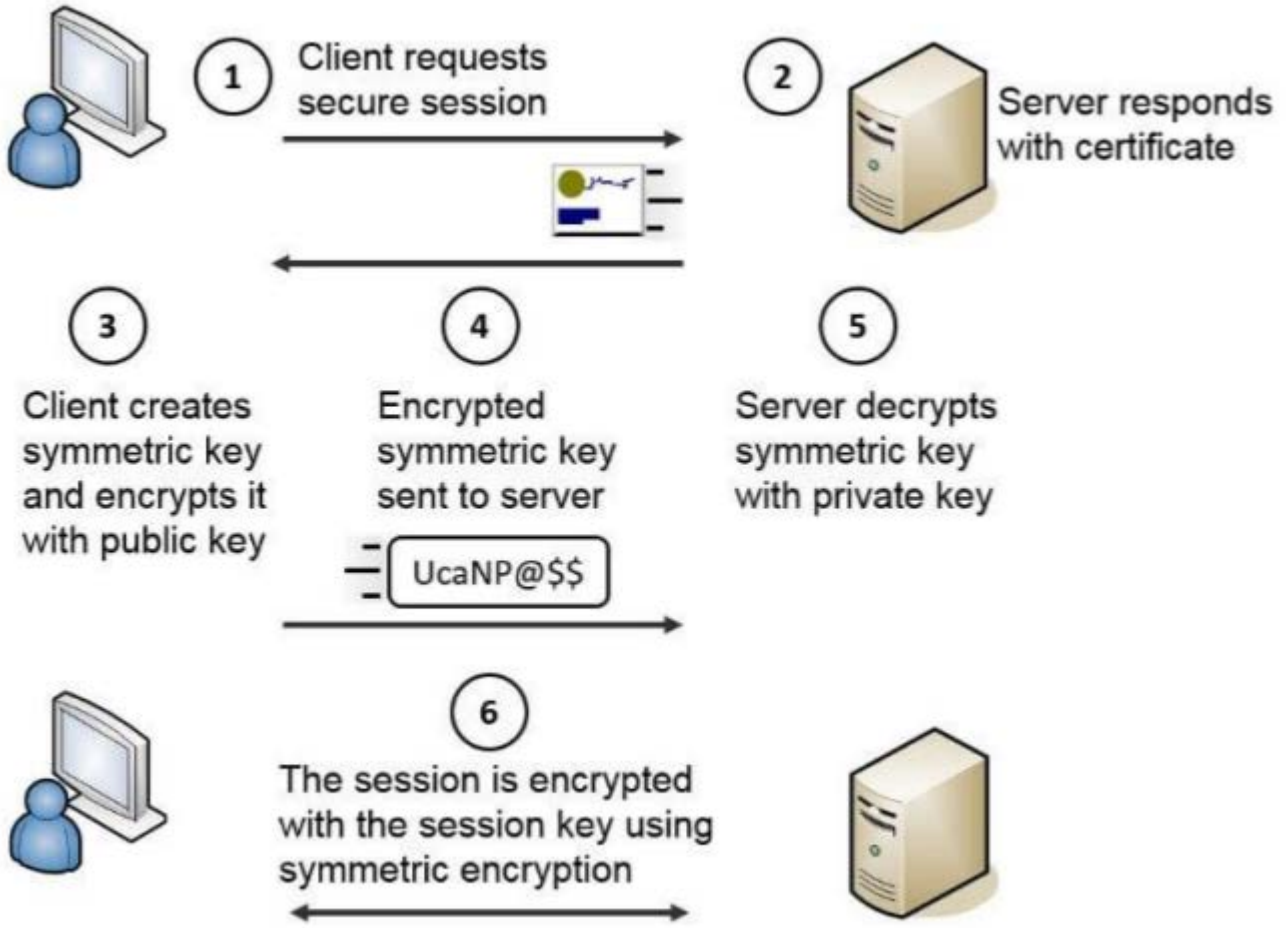
WWW: HTTP-protokollen

HTTP: hypertext transfer protocol

- Applikasjonsprotokollen for Web
- Klient/tjener-modellen – pull
  - Klient: nettleser som spør etter, får og viser webobjekter
  - Tjener: sender web-objekter som svar på forespørsler
- Bruker TCP som transportprotokoll
  - Klienten oppretter en TCP-forbindelse til tjeneren
  - Tjeneren godtar TCP-forbindelsen fra klienten
  - HTTP-meldinger utveksles mellom nettleseren og Webtjeneren
  - TCP lukkes
- HTTP er «stateless» /tilstandsløs
  - Tjeneren sparer ikke på tilstandsinformasjon om tidligere forespørsler

HTTP med SSL – Secure Sockets Layer

- SSL sertifikater kan brukes til å lage en krypteringsnøkkel
- HTTPS
- Hengelåsen på venstre side av internettadressen indikerer at nettsiden har et SSL sertifikat og man kan trykke på den for å nærmere undersøke dette



Persistente og ikke persistente forbindelser

HTTP/1.0: ikke-persistent

- Tjeneren leser forespørselen, svare, lukker TCP-forbindelsen
- 2 RTTer for å hente objektet
  - TCP-forbindelse
  - Forespørsel og overføring
- Hver overføring lider av TCP sin slowstart
- Mange nettlesere åpner flere parallelle forbindelser

HTTP/1.1 og videre: Persistens

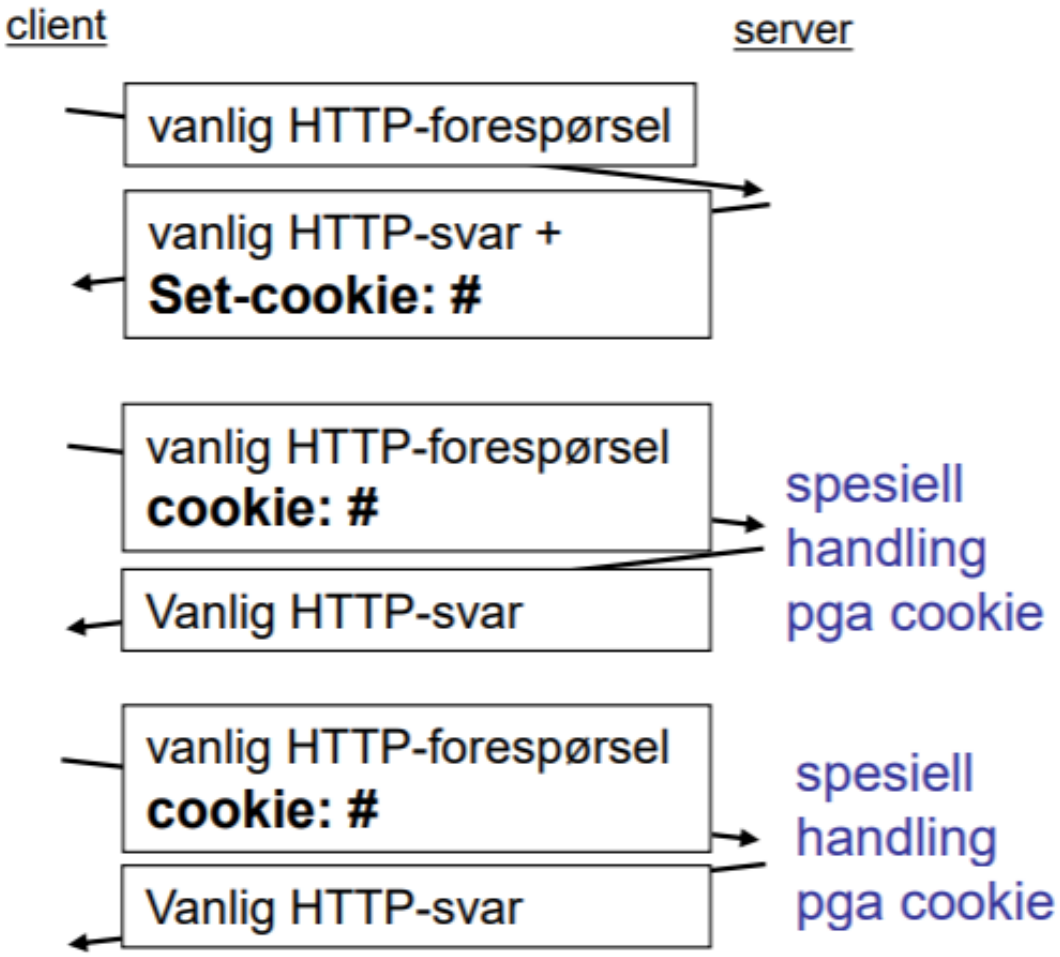
- Over samme TCP-forbindelse leser tjeneren forespørselen, svarer og leser ny forespørsel
- Klienten sender forespørsler for alle de objektene dne trenger som fort den mottar hoveddokumentet (HTML)
- Færre RTTer, mindre slowstart

Cookies: ta vare på "tilstand"

- Tjeneren lager et cookie # , tjeneren husker #, senere brukt til:
  - autentisering
  - Huske brukerpreferanser, tidligere valg
    - produkter brukeren har sett på o.l.
- tjeneren sender cookien til klienten i svarmeldingen

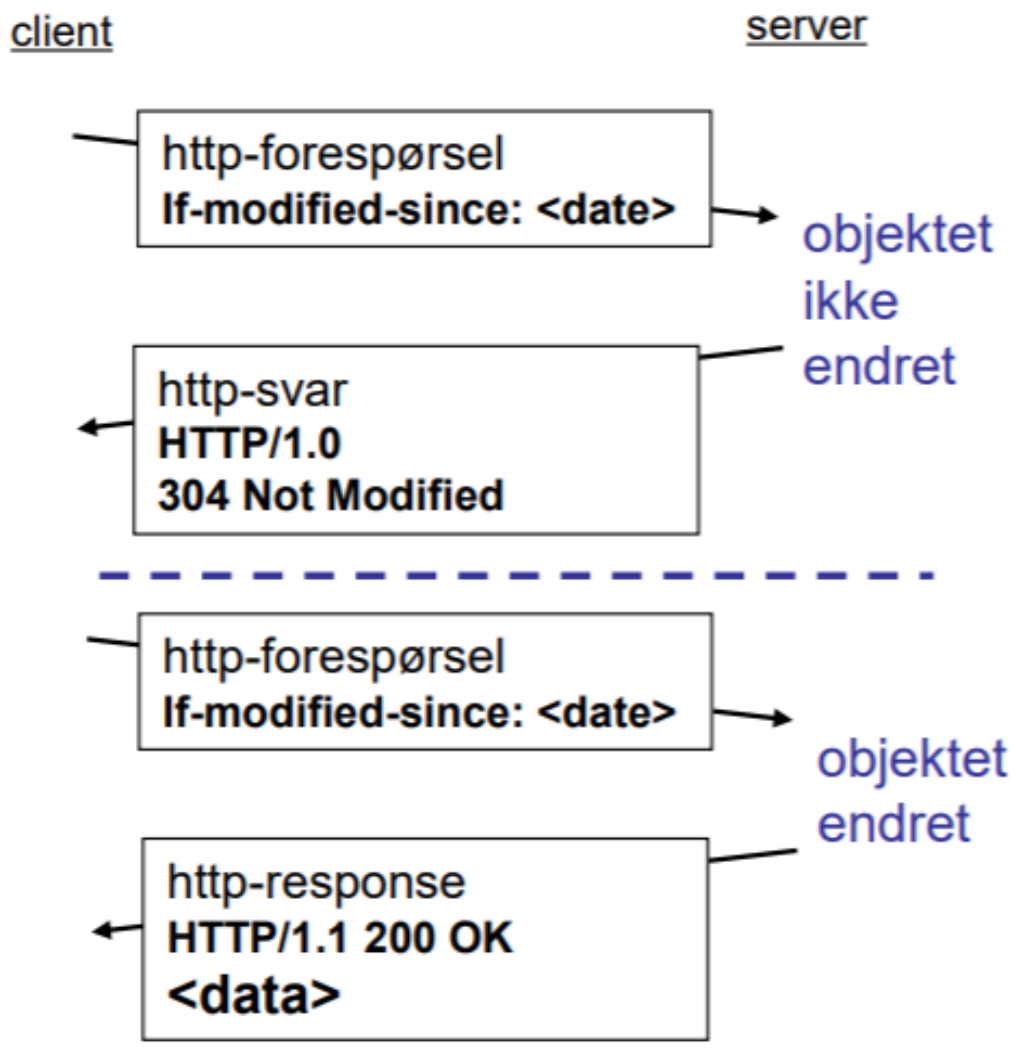
  Set-cookie: 1678453
- Klienten legger ved cookien med etterfølgende forespørsler

  cookie: 1678453



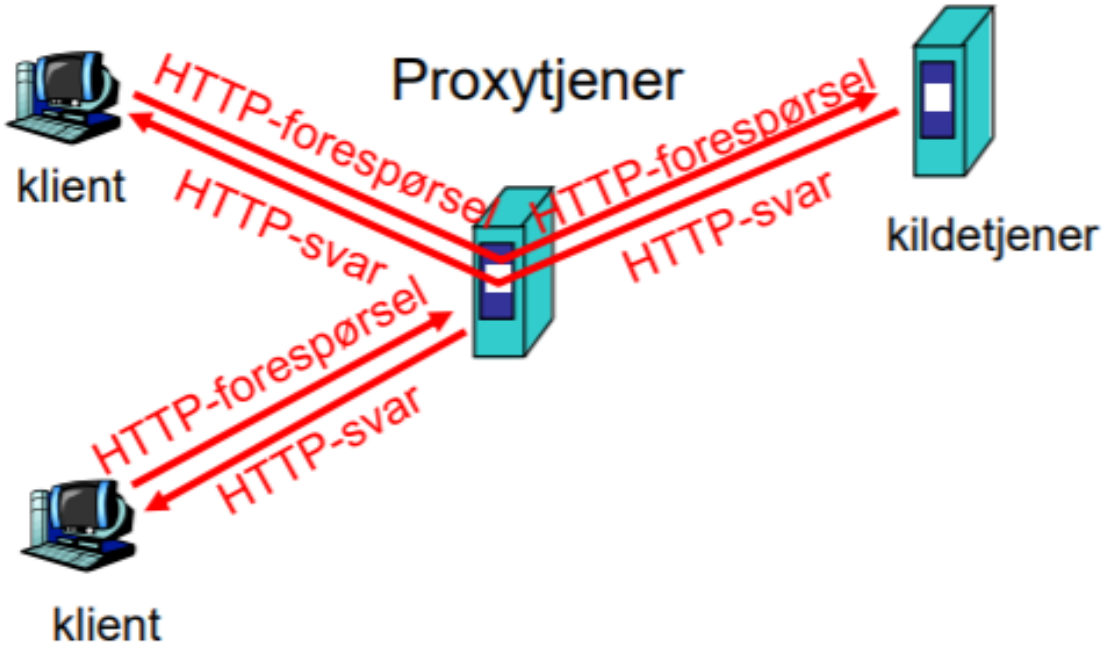
Betinget GET: klient-side caching

- Mål: ikke sende objektet om brukeren har en oppdatert versjon i lokal cache
- klient: oppgi tid/dato for cachet kopi i HTTP-forespørselen
  If-modified-since: <date>
- tjener: svaret inneholder ikke objektet dersom den cachede kopien er oppdatert:
  HTTP/1.0 304 Not Modified



HTTP proxytjener

- Ønsker å lever svar til klienten uten å gå helt til kildene
- Brukeren konfigurerer nettleseren
- Klienten sender alle HTTP-forespørslene til proxytjeneren
  - Om objektet er i web-cache leverer proxytjeneren objektet
  - Om objektet ikke er i web-cache sender proxytjeneren en forespørsel til kilden og lagrer svaret før den sender svaret til klienten

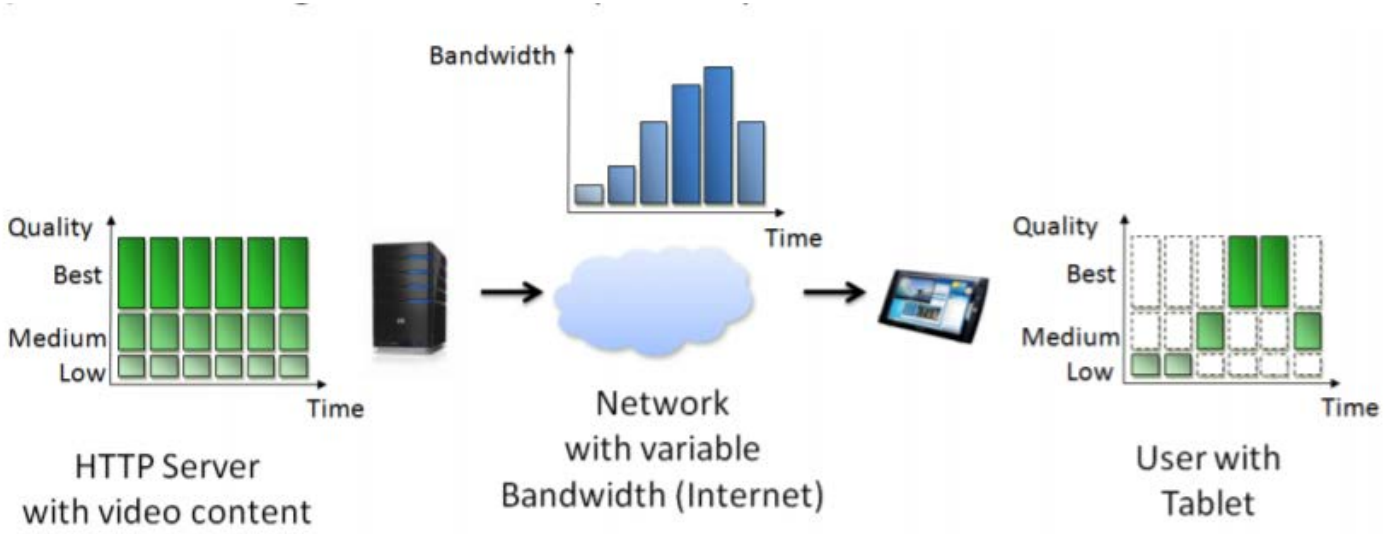


Videostreaming

- Største trafikkmengden som går over internett i dag
  - 60% i 2019
- En konstant strøm av data (til avspilling slutter
- Lettere å ta vare på opphavsrett
  - Lite lokal lagring
- Sparer nettverksressurser
  - Brukes bare for det av innholdet som blir sett
- Default å prøve med UDP også om det ikke går pga. f.eks. brannmurrelger så prøves TCP

HTTP-streaming

- DASH: Dynamisk, adaptiv streaming over HTTP
- Veler videoen i segmenter: fullstendig uavhengige små filmer
  - Varighet 2-10 sekunder vanlig
- Velge antall kvalitetslag
- Velge tilpasningsstrategi
  - Klienten velger etter dens ressurser
  - Strategien hos avspilleren er det som utgjør forskjellen



E-post

- Mailklienter:
  - Message User Agent (MUA)
    - Epostleser
    - Skrive, redigerer, organisere og lese eposter

- Mailtjenere:
  - Mottak av melding / videresending av meldinger
    - Mail submission agent – MSA
    - Mail transfer agent - MTA
    - Mail delivery agent – MDA
    - Mail retrieval agent - MRA
  - Message Handeling Service - MHS

E-post tjenere

- Mailtjenere
  - Mailbox inneholder innkommende meldinger til brukeren
  - Meldingskø av utgående epostmeldinger (for sending)

- Simple Mail Transfer Protocol – SMTP
  - Mellom epostjenere for å sende epostmeldinger
  - Bruker TCP til pålitelig overføring av epost fra klient til tjener
  - Tre faser i overføringen
    - Håndtrykk
    - Overføring av beskjeder
    - Avslutning
  - Bruker persistente forbindelser
  - Push forbindelsesstrategi

Internet of Things – IoT

- Enheter til daglig bruk / husholdningen som leverer ekstra tjenester ved hjelp av Internett
- Kan være med å revolusjonere hverdagen
- MEN de skaper veldig mange sikkerhetsproblemer
- Ofte er ting koblet på internett bare fordi det høres kult ut men de mangler sikkerhetsoppdateringer og skaper sikkerhetshull i nettverket ditt

Hvordan utnytte datasystemer

Læringsmål sikkerhet uke 44

- Angrepsvektor og sikre *datasystemer*:
  - Hva skal til for at vi skal kunne stole på datasystemet?
  - Ingenting er sikrere enn det svakeste ledd.
- Kjenne *applikasjonssikkerhet*, og vanlige sårbarheter i applikasjoner.
- Kjenne til sårbarheter knyttet til *systemprogramvare*.
- Vite hva *skadevare* er, kjenne ulike typer skadevare og hva som kjennetegner dem.
- Mennesket som reell sårbarhet.

Begreper

- Digital sårbarhet
  - En feil i et IKT-system grunnet svakhet, feildesign eller feilimplementasjon
  - To typer:
    - kjente og aksepterte
    - ukjente eller feilvurderte
- Nulldagssårbarhet
  - En sårbarhet som er ukjent for eller uadressert av leverandør
- Angrepsvektor
  - Den konkrete metoden en angriper velger for å utføre et angrep på en datamaskin, et nettverkt eller et system
- Exploit
  - En konkret utnyttelse av en sårbarhet

Skadelig programvare - skadevare

- En samlebetegnelse på ulik programvare som har som oppgave å gjøre skade
- Utnytter sårbarheter i et datasystem
- Kan være et enkeltstående program, eller en del av et *vertsprogram*
- Kan ha den egenskapen at den kan formere seg og infiserer andre systemer

- Oppbyggng:
  1. Stridshode – hvordan skadevaren skal infisere et datasystem
  2. Spredning – hvordan skadevaren skal spre seg videre
  3. Nyttelast – oppgaven skadevaren er satt til å gjøre

Et utvalg av typer skadevare:

Gruppene er til dels overlappende, og ikke alltid presist definert.

 Virus	Kjennetegnes ved at det sprer seg hver gang en fil med virus blir åpnet av bruker eller systemet.
 Dataorm	Skadevare som på egenhånd er i stand til å spre seg og finne nye ofre.
 Trojaner	Skadevare som utgir seg for å være et nyttig program.
Bakdør	En ubeskyttet og ukjent åpning inn i et datasystem, laget av en angriper.
 Spionvare	Programvare for å spionere på en bruker i håp om å få tak i f.eks. informasjon, passord, kontoinformasjon.
 Tastelogger (Keylogger)	En enhet eller programvare som er koblet til tastatur eller maskin og som lagrer eller videresender alle tastetrykk. En variant av spionvare.
Rootkit	Skadevare som endrer eller erstatter funksjoner i OS eller applikasjoner, med mål å ta kontroll over en maskin eller et system.
Exploit	Et program, en sekvens med data, eller en oppskrift på hvordan en sårbarhet kan utnyttes.

Sikkerhet i OS

- Kjerne og driver i operativsystemet er dataprogrammer og kan derfor inneholde sårbarheter
- Kjernen i OS-et kjører i «supervisor mode» og har tilgang på alt, derfor er en sårbarhet i kjernen kritisk
- Ekstremt viktig å holde både operativsystem og drivere fri for sårbarheter
  - Trygg programvare
  - Holde programvaren oppdatert

Bufferoverskridelse

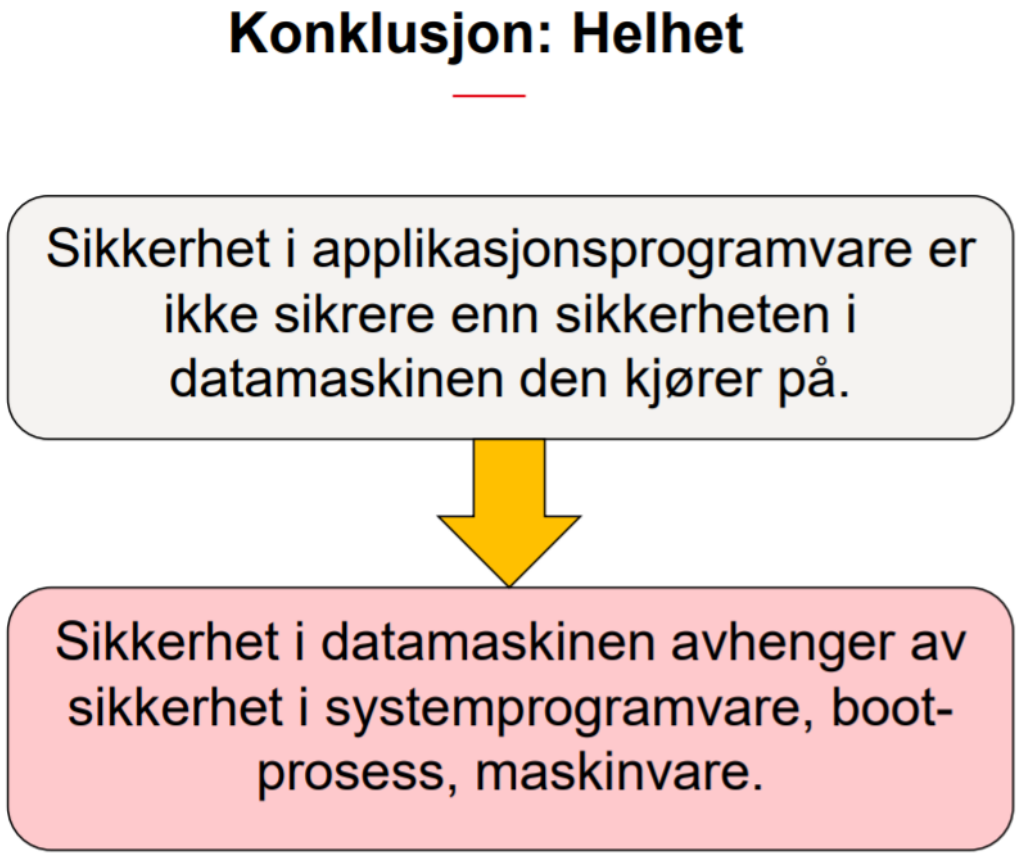
- Et buffer er en sammenhengende seksjon i minnet i datamaskinen avsatt til å inneholde data
- En feil i et dataprogram som gjør at når det kjøres kan det overskride bufferes grenser i minnet kalles **bufferoverskridelse**
- Dette kan krasje et program, føre til korrupte data eller i verste fall utnyttes av angripere
  - F.eks. ved å plassere skadevare på steder i minnet hvor det ligger kjørbar kode eller endre en funksjons returadresse til å isteden peke til adresse hvor det ligger skadevare
- Mottiltak:
  - Programmere trygt – desinfisere all input til programmer
  - Vanskeliggjøre utnyttelse av minnet ved f.eks. uforutsigbare adresserom og sjekksum for data i minnet

TSM – trusted platform modul

- Genererer kryptografiske nøkler på en sikker måte
- En unik RSA-nøkkel er «brent inn» på den enkelte brikke
- Tilbyr integritetssjekk og autentisering av systemprogramvaren som startes
- ISO-standard for en dedikert maskinvare-komponent for kryptering
  - Å overlate håndtering av sikkerhet til fysisk maskinvare giren mer robust beskyttelse enn en programvareløsning
- Skal løse problemet om programvaren bootloader fra BIOS laster inn er identisk med den vi antar at starter og kjører

Applikasjonssikkerhet

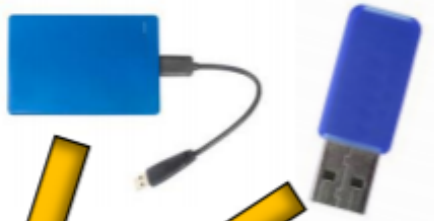
- Applikasjoner er sårbare til angrep og må sikres
  - I designfasen
  - I programmeringsfasen
  - I produksjonsfasen
- OWASP top 10 har en oversikt over de mest kritiske truslene og feilene



Angrepsvektorer: Hvordan kompromitteres datasystemer?

Aksessere ondsinnede eller infiserte nettsteder som inneholder ondsinnede script, eller laste ned og installere skadevare fra nettsteder.







Plugge inn eksterne enheter som er infisert med skadevare.



Infisering via *leveransekjeden*.





Kjøre vedlegg som kommer med e-post, som inneholder utnyttelser og skadevare.





Direkte angrep fra internet, som f.eks. utnytter sårbarheter i operativsystem eller applikasjoner som web-tjenere eller SQL-databaser.

Sikkerhetskultur

«Sikkerhetskultur er summen av de ansattes kunnskap, motivasjon, holdninger og adferd som kommer til uttrykk gjennom virksomhetens totale sikkerhetsadferd»

NSM

Menneskelige sårbarheter

- Manglende forståelse eller sikkerhetsbevissthet hos brukeren
- Sosial manipulasjon
  - Svindel
  - Utpressing
  - Utnytter menneskelig svakhet og tillitt
  - Eldste trikset i boka, mest brukt og enklest
  - Berører alt fra privatpersoner til store virksomheter
  - Både målrettet og tilfeldig

Typer sosial manipulasjon

- Spoofing
  - Forfalske avstender i e-post, SMS, IP-adresse eller telefonnummer
- Phishing
  - Lure offeret i håp om å tilegne seg informasjon
  - Falsk epost eller nettside
- Typer phishing:
  - **Masse-phishing**: Bredt utsendte e-post eller SMS – ofte mindre troverdig
  - **Spear-phishing**: målrettet angrep, samler opplysninger i forkant for å gjøre henvendelsen mer troverdig
  - **Whaling, CEO-phishing**: går etter «de store fiskene», målrettet angrep mot det med makt og myndighet

Mottiltak: Bevissthet og opplæring

- Opplæring er et svært viktig sikkerhetstiltak
- Huskeliste for brukere:
  - ✓ Stopp. Tenk. Klikk. Angripere vil at du skal handle først, tenke etterpå.
  - ✓ Virker noe for godt til å være sant så er det mest sannsynlig det…
  - ✓ Sjekk avsender på all e-post
  - ✓ Vær svært varsom med å følge lenker i e-post
  - ✓ Hold alltid operativsystem, drivere, programvare oppdatert med siste versjon.
  - ✓ Tillitt: Ikke stol på noen/noe før du vet du kan stole på dem/det



Eksamensoppgave 1.5 H2019

Digital signatur

Et konsulentfirma har kunder over hele landet. De ønsker å erstatte den tungvinte prosessen med å signere kontrakter med med penn på papir med en digital variant av signatur som kan garantere at det er kunden og ikke en svindler som signerer avtalen. De har hørt om en metode som kalles *digital signatur*, som kanskje kan gi dem nødvendig bekreftelse på identiteten til den som signerer en kontrakt digitalt.

1.6 Behandling av personopplysninger

Institutt for informatikk ønsker å teste en nytt datasystem for gjennomføring av eksamen, som bl.a gjør det mulig for studentene å programmere Python under selve eksamenen.

Datasystemet instituttet vil benytte er en ren sky-tjeneste fra en ekstern leverandør, tilgjengelig gjennom en web-applikasjon. Både lagring av data og kjøring av applikasjonen skjer på leverandørens datautstyr som står fysisk plassert i et EU-land, mens selve eksamen gjennomføres/avlegges via en nettleser på UiOs datamaskiner i UiOs lokaler.

Den enkelte student får utdelt et unikt kandidatnummer og et unikt passord, som de bruker når de skal logge inn i systemet for å avlegge eksamen. En levert besvarelse knyttes til en student gjennom studentens kandidatnummer.

Vurder følgende utsagn, og velg de du mener er korrekte med utgangspunkt i opplysningene gitt over.

1.7 Trusselmodellering - tilgjengelighet

Etter en vurdering av personopplysningsvern, skal Institutt for informatikk nå ta i bruk følgende datasystem for gjennomføring av eksamen (samme som i foregående oppgave):

Datasystemet er en ren sky-tjeneste fra en ekstern leverandør, tilgjengelig gjennom en web-applikasjon. Både lagring av data og kjøring av applikasjonen skjer på leverandørens datautstyr som står fysisk plassert i et EU-land, mens selve eksamen gjennomføres/avlegges via en nettleser på UiOs datamaskiner i UiOs lokaler.

Den enkelte student får utdelt et unikt kandidatnummer og et unikt passord, som de bruker når de skal logge inn i systemet for å avlegge eksamen. En levert besvarelse knyttes til en student gjennom studentens kandidatnummer.

Tilgjengeligheten til systemet er selvfølgelig svært viktig, da studentene må kunne logge inn og bruke systemet når de møter opp for å avlegge eksamen. Din oppgave er derfor å gjøre en vurdering av hva som kan være en trussel mot *tilgjengelighet*, gitt opplysningene over.