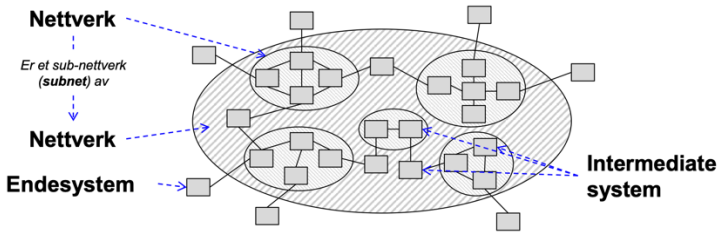
Notater fra IN1020 gruppetime uke 36

Nettverkskomponenter: Lett forklart er nettverk et system for **kommunikasjon mellom to eller flere enhete**r, og disse enhetene kan ha vidt forskjellige formål. Vi kan dele enhetene på nettverket i to kategorier: endesystem og intermediate system, der inndelingen skjer etter hvilken rolle de har i nettverket.

- **Endesystem**: Enheter som ligger helt ytterst i et nettverk, der kommunikasjonen over nettverket skjer primært mellom disse to enhetene (som vanligvis er styrt av mennesker eller prosesser på en datamaskin). Vi kan og dele inn endesystem inn i to kategorier: klienter og tjenere/servere. En server programvare/enhet som tilbyr en tjeneste til klientene – den SERVERER en tjeneste.
    - Eksempler: datamaskiner, mobil, printer, kjøleskap, tv – disse vil være klienter.
    - Eksempler: web server - dette omtales som server.

- **Intermediate system:** Enheter som ligger mellom endepunktene i et nettverk. Slike enheter brukes ikke direkte av brukere, men istedenfor mottar pakker med informasjon og sender disse videre i riktig «retning» på nettverket til det spesifiserte endesystemet.
    - Ressursen jeg brukte: https://www.cisco.com/c/en_in/solutions/small-business/resource-center/networking/network-switch-how.html



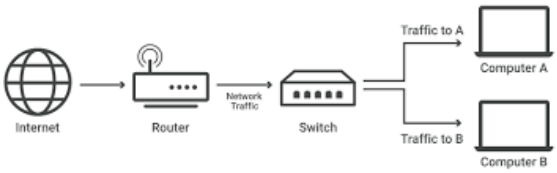
    - **Switcher**: En nettverkskomponent som kobler sammen flere enheter på et datanettverk. Switchen lar disse enhetene som er på det samme nettverket (subnettet) kommunisere med hverandre.



        - Dvs. at switcher kan koble sammen mobiler, dataamaskiner, printere, eller kjøleskap sammen og lar de kommunisere med hverandre.

    - **Routere**: Nettverkskomponent som lar ulike nettverk kommunisere sammen. Routeren videresender pakker/rette pakker mot mottakernoden på nettet, og er ansvarlig for å velge den beste ruten for datapakkene med informasjon skal ta gjennom nettverket slik at kommunikasjonen skal skje så effektivt som mulig.





- **Gateway**: En enhet som brukes for kommunikasjon mellom nettverk med **ulike protokoller**. Gatewayen vil på en måte «**oversette**» fra det «språket» det ene nettverket bruker til dent «språket» det andre nettverket bruker for at kommunikasjon mellom de skal være mulig. Gatewayen er «plassert» på kanten av nettverket som betyr at all kommunikasjon fra utsiden og inn til nettverket (og omvendt) vil skje gjennom gatewayen – derfor det heter en «gate».
  - Ressurs jeg brukte: https://digitalthinkerhelp.com/what-is-gateway-in-networking-types-examples-functions-uses-working/
    - Mange eksempler på gateways.
  - En gateway kan være mye, som for eksempel router eller brannmur etc.
  - Et gøyt eksempler er **«Internet-To-Orbit Gateway»** - «En internet-to-orbit-gateway (I2O) er en maskin som fungerer som en kontakt mellom datamaskiner eller enheter som er koblet til internett og pc-systemer i bane rundt Jorden, for eksempel satellitter eller bemannede romfartøy.»
    - Ressurs jeg brukte: https://no.wikipedia.org/wiki/Gateway#Internet-to-orbit-gateway

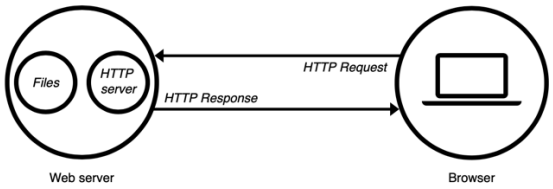
Nettverkstopologier: Handler om hvordan nettverket er satt opp eller konfigurert, og hvordan nodene i nettverket henger sammen. Hvordan nodene er arrangert påvirker ytelsen til nettverket (for eksempel med tanke på effektivitet). Vi kan kategorisere topologiene inn i to: punkt til punkt og broadcast.

- Om dere vil lese mer om de ulike topologiene samt. fordeler og ulemper:
  https://www.dnsstuff.com/what-is-network-topology
  https://techgenix.com/network-topology/
- **Punkt til punkt**: en til en – her skjer kommunikasjonen kun mellom 2 parter på en gang
  - Stjernetopologi, tretopologi, ring, mesh, hybrid
- **Broadcast**: en til alle – her skjer kommunikasjonen fra en part til alle i nettverket
  - Buss, ring

Aksessmodeller: Enkelt sagt så finnes det to måter en klient kan aksessere data på i nettverk.

- **Klient-tjener**: Klienten ber om en tjeneste og serveren leverer tjenesten.
  - At the most basic level, whenever a browser needs a file that is hosted on a web server, the browser requests the file via HTTP. When the request reaches the correct (hardware) web server, the (software) *HTTP server* accepts the request, finds the requested document, and sends it back to the browser, also through HTTP. (If the server doesn't find the requested document, it returns a 404 response instead.)

- Ressursen jeg brukte: https://developer.mozilla.org/en-US/docs/Learn/Common_questions/What_is_a_web_server



- **Peer-to-peer**: Alle nodene i nettverket er likestilte og sammenkobla slik at alle kan nå alle. Eierskapet her er **distribuert** slik at istedenfor å aksessere data fra en server, kan data aksesseres fra andre **likeverdige noder**. Man deler på ressursene og **ingen har en spesifisert rolle**. Endesystemene kan være både klienter og tjenere.
    - Veldig likt **CDN** (content delivery network). CDN er også brukt for å distribuere innhold på nettet, men det har en hierarkisk struktur. I denne «modellen» så er lite populær informasjon lagret «langt fra» sluttbrukerne, mens populær informasjon lagres nærmere sluttbrukerne siden den blir ofte aksessert.

