NETTVERK

BEGREPER/FORKLARINGER

Båndbredde er hvor mye data du kan flytte på en gitt mengde tid

Hastighet er tiden det tar å gjennomføre et stadie av en oppgave

OPERATIVSYSTEMER

Et operativsystem er «dirigenten» i datamaskinen. Den sørger for at flere brukere kan dele en datamaskin og for at flere programmer kan kjøre samtidig. Den fungerer som et «mellomnivå» mellom maskinvare og programvare. De vanligste operativsystemene å bruke i dag er multibruker/multioppgaver

- Operativsystemer skjuler kompleksiteten i maskinvare
- API for å tillate programmer å bruke ressurser i maskinen
- Grunnleggende sikkerhetsmekanismer er bygd inn

Kjernen i operativsystemet styrer alt og kommuniserer med periferienhetene i datamaskinen gjennom drivere. Brukerprosessene må kommunisere via kjernen for å kunne benytte periferienhetene. Minnet i en datamaskin benyttes til mellomlagring.

Hovedkomponenter i et operativsystem

«Synlige» for bruker

- Brukergrensesnitt: Koblingen mellom bruker og datamaskin. Mekanismer for at bruker og programmer skal kommunisere med operativsystemet og bruke maskinen sine ressurser
- Filsystem: Mekanismer for at en bruker skal kunne lage, slette, modifisere og manipulere filer
- Enhetskontroll: Operativsystemet må kunne kontrollere enheter tilkoblet maskinen/gjøre at systemet skal kunne kontrollere enheter som er koblet til. Eks: USB, høyttaler, tastatur osv.

«(relativt) Transparent»

- Prosessorkontroll: Tilby mekanismer for at systemet effektivt og rettferdig skal kunne dele CPU for programmene som kjører
- Minnekontroll: Tilby mekanismer for at systemet skal kunne fordele minneressurser – allokerer plass til programmer
- Kommunikasjonstjenester: Tilby tjenester for at systemet skal kunne kommunisere med andre prosesser (på samme eller andre datamaskiner)

Systemkall: Grensesnittet mellom operativsystemet og programmer (bruker) er definert med systemkall. Systemkallet går inn i kjernen på operativsystemet

Brukernivå vs. Kjernenivå (beskyttelse)

Operativsystemet kjører i kjernemodus

- Real mode
- Tilgang til hele minne
- Alle instruksjoner kan bli kjørt
- Går forbi all sikkerhet
- Bytte til *Protected mode* eller *Long mode* snarest mulig

Vanlige programmer og flere API'er kjører i brukernivå

- *Protected mode*
- Bruke privilegerte nivåer (x86, «ring», ARM: «mode»)
- Ytre ringer:
    - Kan ikke kontakte maskinvare eller enheter direkte, kun gjennom et API
    - Tilgang kun til et begrenset minneområde
    - Begrenset tilgang til prosessoren

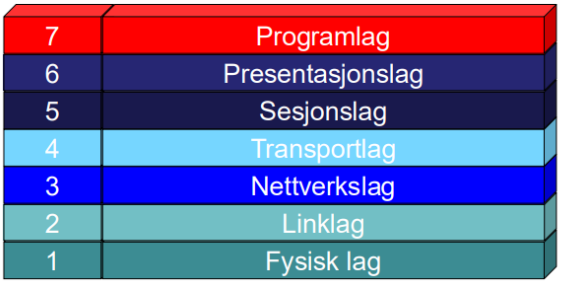
To hovedmodeller for hvordan et operativsystem er bygd opp

Monolittisk kjerne

- All beskyttet kode kjører i samme beskyttelsesområde
- Effektivt: alle komponentene kan kommunisere direkte (funksjonskall, skrive i minne)
- Feil kan få følger for andre komponenter enn der de oppstår
- Store og komplekse
- BSD, UNIX, Linux, Windows 10 (++), osv.

Mikro-kjerner

- Små moduler med minimal funksjonalitet (avbrudd, minne, prosessor, osv.)
- Så mange som mulig av tjenestene er/skriver i brukermodus
- Mye beskjeder frem og tilbake (ineffektivt)
- Vanskelig å utføre ting i riktig rekkefølge
- Vanskelig feilsøking pga. grensene
- Liten modulær, utvidbar, portabel, …
- MACH, Windows NT



OSI – Modellen / ISO Open Systems Interconnection

- Modell for lagdelte kommunikasjonssystemer
- Grunnleggende konsepter og terminologi
- Definerer syv lag med funksjonalitet
- Brukes ikke særlig i praksis i dag, da det er for tregt å implementere
- Data units har ulike navn ut i fra hvilket lag vi befinner oss i.
    - Transportlaget: packets
    - Nettverkslaget: datagram
    - Linklaget: Frame

TCP/IP – Modellen

TCP/IP – modellen / fem-lags referansemodellen er en forenkling av OSI. Man har gått ned til fem lag, hvor presentasjon, sesjon og applikasjonslaget er slått sammen til et lag (applikasjon). Noen anser også linklaget og det fysiske laget som slått sammen til ett lag (nettverksgrensesnitt).

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

Dataenheter i TCP/IP – modellen:
«Beskjeder» fra applikasjonslaget defineres som «data units». Disse har forskjellige navn avhengig av hvilket lag vi befinner oss i:

- Segments: transportlaget (kan inneholde fragmenter)
- Packets: nettverkslaget
- Frame: linklaget
- Bits: fysiske laget

Applikasjonslaget: der programmene i nettet kjører. Lag med tjenester for applikasjoner, som nettleser, e-post ol.

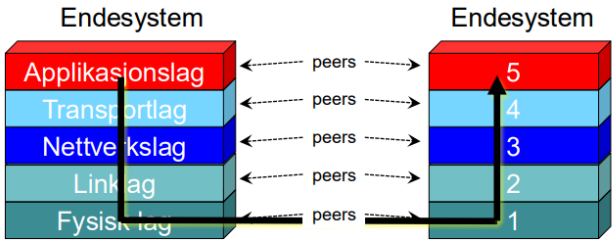
Transportlaget: kobler sammen ende-til-ende (TCP/UDP). Finner porten dataen skal sendes til (eks. nettleser). TCP (Transmission Control Protocol) har et oppsett av forbindelse med 3-veis håndtrykk. UDP (User Datagram Protocol) er en tilkoblingsløs protokoll og har ingen garantier.

Nettverkslaget: rute data fra ende-til-ende systemer (IP) – vanskeligst å gjøre endringer. Finner riktig mottaker. Ansvarlig for ruting – hvordan pakken skal flyte gjennom nettet. IP er den mest brukte nettverksprotokollen. En IP må være unik siden det er ende-til-ende. Dette gjør at vi begynner å få for få IP-adresser og vi går nå fra en 32-bit adresse (IPv4) til en 128-bit adresse (IPv6).

Linklaget: pålitelig overføring mellom to noder (MAC-adresse, Ethernet, LAN), at alt er sendt og i riktig rekkefølge. Passe på enkel flytkontroll og feildeteksjon. Hver endenote får en MAC-adresse på 6 byte som ofte er lagret i nettverkskortet. Dette gjør at ingen snakker i munnen på hverandre. Det vanligste linklaget er Ethernet og WiFi.

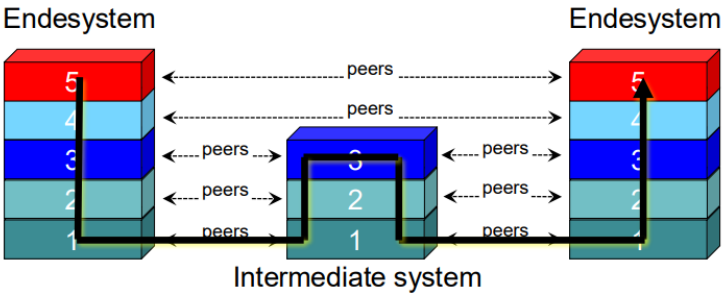
Det fysiske laget: sender bit ut på mediet (kablet eller trådløst). Består av signalrepresentasjon av bits. Sørger for formelle regler om kommunikasjon

Datastrøm mellom to systemer med direkte tilkobling



- Forespørselen går nedover lagene på "*avsender*"-siden og oppover hos mottaker.
- Det fysiske laget håndterer selve sendingen til neste (nabo) node.

Datastrøm mellom to systemer uten direkte tilkobling



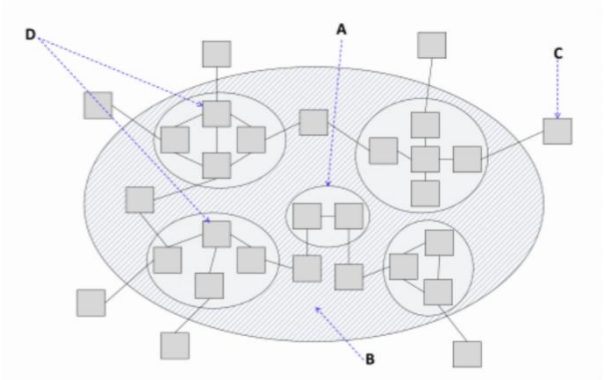
- Nettverksprotokollen (IP) tar hånd om rutingen til endelig destinasjon.
  - På hver ruter så går pakken opp til nettverkslaget for prosessering.
- Transport og applikasjonslaget kommuniserer ende-til-ende.

Datanettverk / lagdeling i internettarkitekturen

En **protokoll** definerer strukturen på beskjeder sendt over et nettverk. En protokoll må adressere mange kompleksiteter som; hvordan maskinvaren skal oppføre seg, hvordan beskjeden skal finne frem, om det er noen garantier for levering og hvordan håndtere kø, tap og andre problemer. En protokoll handler om kommunikasjon mellom samme lag. Protokollen definerer formatet på beskjeden og en header.

Nettverkskomponenter

- Subnet (A), en gruppe av enheter innenfor et nettverk (B)
- Endesystemet (C) finnes ytterst i nettverket (datamaskiner, mobiler, servere osv.)
- Intermediate systemer (D) finnes inne i et nettverk (ruter, switch, gateway osv.)

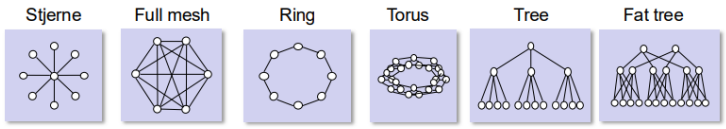


Nettverksstrukturer

- Punkt-til-punkt: forskjellige typer kabler eller radiolinker som kommuniserer fra punkt til punkt (en-til-en)
- Topologi: f.eks. ring eller stjerne

- Broadcast-systemer: nettverk som deler kommunikasjonsmedium. Én sender, alle lytter (én-til-mange), eks: trådløst internett. En melding som sendes ut på en spesiell adresse og leveres til alle enheter på samme LAN. Problem: hvordan takle kø, alle skal ikke få «høre» alle sitt

Topologieksempler:

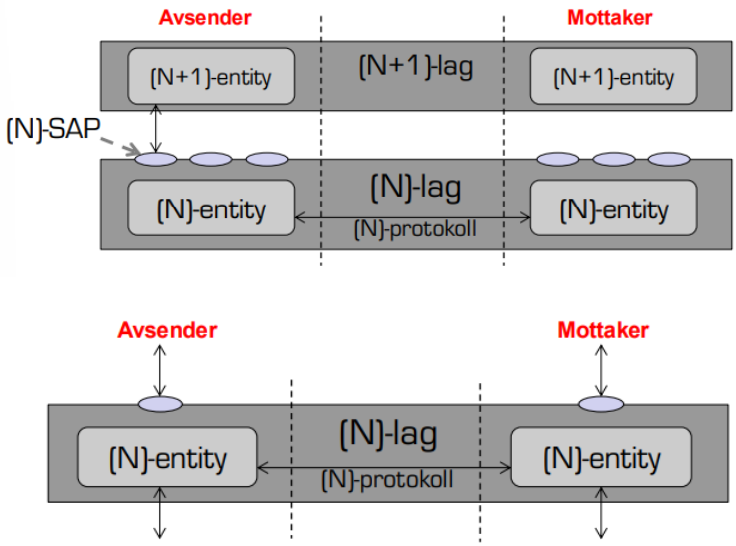


Protokoller og lag

Utfordring: komplisert å kommunisere med fremmede maskiner på nettet og interaksjon mellom forskjellige system og/eller nettverk. Forenkling: introdusere standardiserte abstraksjonsnivåer (lag).

Lag i nettverket (OSI)

- (N)-lag er et bestemt abstraksjonsnivå
- (N)-entity er oppgaven til et lag (typisk en prosess)
- (N)-service Access Point er tjenesteidentifikasjon. Beskriver hvordan lag N tilbyr en tjeneste til lag N+1
- (N)-protokoll er et sett med regler for hvordan data skal overføres mellom entity på samme nivå



Lagene spiller sammen / lokalnettverket (LAN) og subnet

Internett er en sammenkobling av mindre, separate nettverk. På lokale nettverk kan man koble sammen enheter ved hjelp av en switch eller en HUB, dette fungerer på nettverkslaget. For å sende en pakke til en maskin utenfor ditt lokale nettverk må den sendes til en router som vet hvor den skal videresendes.

Switch: filtrerer og videresender **HUB**: videresender det som kommer inn på alle porter

IP-adresser

En IP-adresse er en unik identifikator («postadresse» for en datamaskin). Ved hjelp av adresser som er unike på hele internett kan man sende til en annen maskin på tvers av mange små nettverk. Vi skiller også mellom adresser innenfor og utenfor det lokale nettverket.

IPv4 deler opp en IP-adresse i fire oktetter. For å definere et subnett må man ha en nettverksmaske. I nettverksmasken kan bits satt til 0 varieres for å angi IP-adresser i subnettet, mens bits satt til 1 angir delen av IP-adressen som definerer hvilket nettverk vertene tilhører.

Nettverksmasken består av en sammenhengende serie '1' deretter en sammenhengende serie '0'. Det er to måter for å notere omfanget av et subnett. Ved punktnotasjon 192.168.1.0, da må man oppgi nettverksmaske (255.255.255.0). CIDR-notasjon (Classless Inter-Domain Routing) er eksempelvis 192.168.1.0/24, har vanlig punktnotasjon først og tallet etter skråstreken angir hvor mange bits nettverksmasken består av (24 av 32).

En IP-adresse – mange porter, hvordan fungerer dette?

- Transportprotokollene UDP og TCP implementerer 'porter' som muliggjør totalt 65535 (per UDP/TCP) samtidige forbindelser på en IP-adresse. Informasjon som sendes over nettet kan aksepteres av mottaker-maskinen ved å bruke TCP- eller UDP-porter. Når et program på en datamaskin sender/mottar data over nettet, sendes denne dataen til en IP-adresse og en spesifikk port på den «eksterne maskinen», og tar imot dataen på en (vanligvis random) port på lokal/mottakermaskinen
- Ved bruk av TCP-protokollen tilkobles en TCP-port, og vice versa for UDP-protokollen
- Når en applikasjon kobler seg til en spesifikk port vil den ikke kunne brukes av andre applikasjoner. «First come, first served».

Kringkasting (send til alle)

En melding som sendes ut på en spesiell adresse, leveres til alle enheter som er koblet på samme LAN (nettverk). Egenskaper; kan bli problemer hvis to noder sender samtidig, feildeteksjon er viktig og når en node sender kan alle noder som er i rekkevidde lytte på kommunikasjonen.

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 settes typisk av til router om du skal koble LAN til internett
- Router fungerer ofte også som DHCP-tjener og DNS-cache i hjemmenettverk

DHCP

Utdeling av IP-adresser:

- Discovery (klient - tjener)
- Offer (tjener - klient)
- Request (klient - tjener)
- Acknowledge (tjener - klient)

- DHCP-tjener: sender til alle at en IP-adresse er tilgjengelig, samtidig som den sender en liste over nyttige adresser
- Request: klient broadcaster til alle at den har tatt den
- DHCP-tjener: Acknowledger at den adressen tilhører klienten

Prosedyre:

1. Det kommer en ny maskin på nettverket. Maskinen sier til alle (sender en kringkastingsadresse): finnes det en maskin (DHCP-server) med myndighet til å dele ut IP-adresser på dette subnettet?
2. DHCP-tjener (hjemmeruter) sier til alle: 192.168.1.5 er tilgjengelig. Her har du også en liste over andre viktige adresser som gateway og DNS.
3. Ny maskin: Tar i mot adressen.
4. DHCP-tjener: skriver opp at 192.168.1.5 er i bruk av den nye maskinen i en periode på 24 timer.

- Den nye maskinen må altså spørre igjen innen 24 timer om å få beholde IP-adressen hvis ikke blir den sett på som ledig.

ARP – koblingen mellom nettverks-laget og link-laget (nettverk og IP)

- Nettverkskortene har en 6 byte lang 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. ARP kobler IP (nettverkslaget) og MAC (linklaget). ARP kan også brukes til å finne duplikate IP-adresser, dersom to maskiner svarer.
- ARP-forespørsel:
    - Sender til alle: Hvem har denne adressen?
    - Den som har den svarer
    - Den som spurte lagrer info i en tabell, dette kalles en ARP-tabell/ARP-cache
- Hvis kringkastingsdomenet er for stort når ARP sender ut en melding kan det stjele kapasitet som ellers burde vært brukt til å overføre data (DHCP har samme problem og avanserte switcher kan filtrere ARP og DHCP for å beskytte mot overbelastning fra kringkastet trafikk).
- ARP-tabellen lagrer informasjon om hvilke IP-adresser som assosieres med hvilke MAC-adresser. Når en pakke skal sendes til en IP-adresse sjekker systemet først om MAC-adressen finnes i tabellen; hvis ja er det ikke nødvendig og bruke ARP. Hvis nei sendes pakken til nettverket med ARP-protokollen som spør hvem som har den aktuelle

adressen - maskinen med denne IP-adressen svarer med en ARP-pakke som også inneholder MAC-adressen som kan ta imot pakker for den IP'en.

Kringkastingsdomener

Om det er for mange enheter så kan det bli problemer. ARP og DHCP-forespørsler går til alle, dette vil for et stort domene stjele kapasitet som burde brukes til å overføre data. Avanserte switcher kan fikse dette problemet. Burde unngå for store kringkastingsdomener, heller dele opp i subnett.

- Man kan skille mellom de ulike tjenestene man kan kjøre på en datamaskin ved bruk av porter. Notasjon IP : port

NAT

- NAT-routeren oversetter datatrafikk inn/ut av det private nettverket.
- Det finnes 232 IP-adresser, og vi begynner å få for få. For å unngå problemet og fremme sikkerhet så kan vi bruke adressene flere ganger, dette gjennom NAT.
- NAT utnytter at transportlaget tilbyr porter som en funksjon. Routeren gir en maskin på det lokale nettet en bestemt port, og lagrer dette i en tabell som oversetter den lokale adressen til sin eksterne adresse. Når maskinen vil snakke med en annen maskin på internettet så vil den andre maskinen svare med IP + porten som er gitt for maskinen og routeren kan finne ut hvilken maskin som den skal sende svaret til.
- Dette innebærer at en unik IP-adresse er nok for flere datamaskiner.
- Ulemper med NAT: Gir en ekstra kompleksitet, og nye forbindelser må komme fra innsiden av NAT-nettverket. Routeren må bevare tilstanden til forbindelsene og det gjør det vanskelig å koble til utenifra.

Ruting i internett

- Mål: videresende en datapakke slik at den til slutt når måladressen sin.
- ISP (Internet Service Provider) er internettleverandørene som gir deg tilgang til internett.
- ISP-peering er når ISPer inngår avtaler om å videresende hverandres trafikk. Dette fører til at økonomiske prinsipper påvirker trafikkflyt.
- BGP (Boarder gateway protocol) er protokollen som ISPene har for ruting mellom seg.
- Traceroute – viser hvor trafikken går
    - Kommandoen «traceroute» bruker en protokoll «Internet Control Message Protocol» (ICMP) til å spore hvor datapakkene er innom på veien

TCP

- Flytkontroll – ikke sende fortere enn mottageren kan ta imot
- Metningskontroll – Bruke ressursene på ditt lokale nett på en rettferdig måte ved å fordele likt på alle
    - Hvis en maskin merker at nettverket har alt for mye trafikk, så vil den trekke seg tilbake og bruke mindre trafikk. Det som muliggjør dette er at for hvert datasegment mottaker mottar så skal mottaker sende en bekreftelse på at det er mottatt. Så hvis pakke er gått tapt så vil hastigheten senkes og det vil da sendes pakkene som ble tapt på nytt.

    - 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 traffikk 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.
- TCP har også en leveringsrekkefølge, dette gjør at man kan forsikre seg om at pakkene du sender kommer i riktig rekkefølge til mottaker
- En mekanisme for å forsøke å dele kapasiteten over flaskehalsen likt
- Feilsjekking av nyttelasten (sjekksum)
- Pålitelig

Kryptering/sikkerhet i lagene

- Viktig å sikre informasjon som sendes i leddene på internett. Det er mulig å sikre informasjon i alle lagene.
    - VPN (IPSEC) kobler to subnett sammen så det fungerer som et LAN selv om de er fysisk adskilt (oppretter kryptert forbindelse og blir enige om en nøkkel som begge utveksler på en sikker måte)
    - TCPcrypt har som mål at alle TCP-forbindelser som settes opp skal være kryptert
    - SSL (Secure Sockets Layer) (https) er kryptering fra ende til ende
    - WPA (WiFi Protected Access) kryptering på trådløse nettverk, krypterer forbindelsen mellom deg og aksesspunkt/ruter

Hva er poenget med lagdelingen?

- Lagdeling = standardiserte abstraksjonsnivåer
- Lagdelingen av en datamaskin (fra transistorer til internett): er en forutsetning for å lage en datamaskin. Takket være lagdeling er det mulig å håndtere kompliserte konstruksjoner
- Lagdeling i internettarkitekturen: utfordringene utgjør at det kan være veldig komplisert å kommunisere med fremmede maskiner på nettet og interaksjon kan påvirkes av forskjellige typer system/nettverk (f.eks. PC/CPU vs. MobilTlf/CPU), man må sørge for at maskinene snakker samme språk. Løsningen på dette er standardiserte abstraksjonsnivåer/moduler/lag. Data som sendes over nettet går altså gjennom flere lag og oversettes til et språk som mottakermaskinen til slutt kan forstå. Lagene kan byttes ut/endres for å tilpasse ulike oppgaver og språk/kommunikasjonsmåter (hvilket er enda en motivasjon for å bruke flere lag).
- Lagdelingen gjør det f.eks. mulig for nettverkskomponenter (som switcher og routere) å implementere bare et delsett av lag, for å støtte bare det som trengs for å sende en pakke videre

DNS – hvordan fungerer det?

- DNS synker domenenavn med IP-adresser hvilket gir brukere domenenavn/adresser som er enkle å huske, mens maskinene på nettet bruker IP-adressene
- DNS identifiserer og lokaliserer datasystem og andre ressurser på nettet; når man skriver inn en webadresse/URL matcher DNS denne adresse med tilsvarende IPadresse og kobler en til den nettsiden.
- DNS er en database som forvarer alle domenenavn og tilsvarende IP-adresser for en spesifikk top-level domain (TLD), f eks .com, .net etc.
- DNS databaser er lagret på fysiske servere i en distribuert database, dvs. mange maskiner på en måte som gjør at man kan få tak i den informasjonen man trenger hvor enn man befinner i verden.
- Hierarkisk navnetilordning: .com > google.com > mail.google.com - Ulike maskiner har ansvar for forskjellige deler av navneoppslagene.
- DNS har en enkel klient/tjener arkitektur (den mest brukte tilkoblingsmåten på nettet), man setter opp en maskin/tjener som oppretter en tjeneste/lager en port f eks TCP. Alle maskiner vet da at hvis de kobler seg til akkurat den porten - kobles de til en DNS-tjeneste. UDP (vanligst frem til nå, brukes fortsatt på de fleste klientmaskiner) eller TCP port 53, fra nylig må tjener bruke TCP. Dette grunnes overgang til internett protokoll 6 hvilket medfører veldig lange IP-adresser, som ikke får plass i en DNSpakke. Informasjonen som da skal sendes med UDP, må deles opp i to pakker hvilket medfører større risiko for å miste informasjon på veien. Klienter som bruker TCP avvises ofte.
- Klassisk metode/rekursivt oppslag: forespørselen går igjennom mange maskiner før den returneres til maskinen som først spurte. Dataflyten konsentreres rundt de sentrale tjenerne der også mange tilstander lagres. En del ressurser blir dårlig brukt.
- Ny metode/iterert oppslag: man delegerer til den første gyldige DNS-tjeneren i stedet for å gå igjennom mange maskiner på veien. Tilstanden lagres bare av den lokale tjeneren inntil svaret er levert.

Navnehierarki og rottjenere

- Rottjenere (et set med maskiner) har lister på TLDs samt adresse til hvem det er som er ansvarlig for disse domenene. Siste leddet i en nettadresse. F eks .no, .com
- Disse TLD' forgrener seg nedover, f eks. .no > uio.no > ifi.uio.no > www.ifi.uio.no
- Rottjenerne administreres av et foretak, ICANN (frem til nylig eid/administrert i USA, er nå mer globalisert). I Norge er det UNINETT-NOR, disse godkjenner ansøkninger om bruk av nye domenenavn. Deretter er UIO administrator.
- Hver DNS-tjener har autoritet over en del av hierarkiet, de skal ha en liste over alle som sorteres under sitt sub-tre. Må kunne replikeres, derfor er det minst to tjenere på alle nivåer. Alle må kjenne til adressene til rottjenerne.
- Rottjenerne er ansvarlige for "Root Zone File"/liste over TLDer og hvem som kontrollerer dem. Det finnes 13 rottjenere hver av 6 er replikert globalt med en teknikk som heter "anycast".
- Rottjeneren kontaktes når man mislykkes med navneoppslag. I praksis mellomlagres denne informasjon i cache på de fleste system.

- DDoS-angrep: denial of service-angrep, man prøver å sende så mange forespørsler at serveren går ned (har vært nær, men ikke skjedd). - Rottjenerne ligger fremst i USA.

Tjenester i internett

Forbindelsesstrategier

- **Pull**: klienten ber tjeneren om en tjeneste, og er den vanligste metoden
- **Push**: tjeneren dytter en tjeneste til en klient, dette krever at det er en forbindelse fra før eller at klienten lytter (eks. push varsler)
- **Publish-subscribe**: variant av push der tjeneren dytter ut beskjeder til en gruppe av abonnenter

Aksessmodeller

Klient-tjener:

- Tradisjonell kommunikasjonsmodell/arkitektur: Klient ber om en tjeneste (oppretter en forbindelse), tjenere leverer tjenesten (svarer på forespørselen)
- Eks. Webklient (nettleser), mail
- Potensielt store kostnader med å få infrastrukturen skalerbar

Peer-to-peer:

- Første velkjente programmet: Napster
- Fildeling (brukt til f.eks. musikk og film. Eks. Popcorn time)
- Hvis en maskin har tilgang til en fil, blir dette delt i et felles nettverk
- Alle noder er likeverdige
    - En maskin kan være både klient og tjener samtidig
- Alle noder kan nå hverandre
- Eierskapet er distribuert
- Distribuert systemarkitektur
- Typisk mange noder som kan være upålitelige og heterogene
- Styrker:
    - Ideelt til å dele ressurser mellom enheter
    - Tilgangen/fordelingen av ressurser styres av algoritmer
    - Robust mot endringer i nettverket av peers
- Svakheter:
    - Egner seg dårlig til applikasjoner som er avhengig av lav forsinkelse.
        - Onlinespill med høyt tempo (som First Person Shooters, f.eks. ”Battlefield”)
    - Om man er avhengig av en viss grad av sentral kontroll.
        - P2P er vanskelig å temme når ”katta først er sluppet ut av sekken”
    - Peers kommer og går
        - Mye redundans og ressurser kreves for å sikre kontinuitet
    - Kan være vanskelig å effektivt indeksere og finne igjen elementer

Fysisk plassering av innholdet

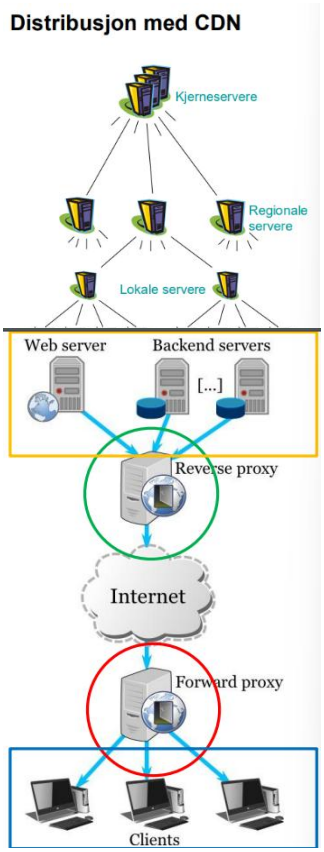
Man kan utvide cachehierarkiet for å gjelde også utenfor datamaskinen. Da kan man mellomlagre data på ulike avstander fra brukeren også på internett.

- Eksempelvis ha en lokal proxy på et lokalt internett, hvor den har en cache som kan levere data med kort responstid til brukerne av det lokale nettet.
- Internettleverandører har ofte ting lagret hos den for store bedrifter som trenger data i store deler av verden.
- Man kan også gå til den originale datakilden selv om dette tar mye lengre tid

CDN

Content Delivery Network er kopier av data som man ønsker å dele/levere til folk som er nærme der brukeren er. For eksempel hos internettleverandøren eller i sky-tjenester. Dette gir kortere RTT-tid og forhindrer også overbelastning på tjeneren.

Minus: koster ekstra maskinvare og lagringsplass.



Proxy-cache

Er en forward-proxy som star nære klientene og vil da levere til klientene. Mellomlagrer data for klientene, slik at de ikke behøver å gå helt til kilden.

En "Reverse proxy" står nær tjenerene og mellomlagrer data fra en eller flere tjenere slik at klienten slipper å gå helt til kilden(e). Fungerer som en front for klienten. Dette bruker mye på Wi-Fi på fly.

Head of line blocking

- Mottaker må vente på et forsinket segment før mer data kan leveres
- Tapte pakker må sendes på nytt
- Dette fører til en forsinkelse

Videostreaming – utfordringer

- Konstant strøm av data (til avspilling slutter)
- Nedlasting: hele innholdet lastes ned til en lokal maskin
- UDP eller TCP?
    - UDP kan fungere helt fint, fungerer raskt og merker det ikke om det mangler noen pakker
    - Mer TCP i det siste
- Nettverksutfordringer
    - Forsinkelser, tap, variasjoner i leveringstid («jitter»)
    - Jitter kompensasjon
    - Tapskompensasjon

HTTP-streaming

Dynamisk, adaptiv streaming over http (dash)

Bruker TCP som transportprotokoll. Dette gjøres ved å dele videoen opp i biter på cirka 2-10 sek. Hver av de bitene er en film som kan spilles av helt uavhengig. Det finnes også flere versjoner av hver bit, en med god kvalitet, en med middels kvalitet, og en med dårlig kvalitet. Måten man sender dataen på er ved at videospilleren fungerer som en nettleser. Den spør serveren om den kan levere et segment og serveren sender med lavest kvalitet og mottaker mottar. Dersom dette går fint, ingen pakker mistet ol. så sender serveren gradvis med bedre og bedre kvalitet, helt til optimal kvalitet er oppnådd.

HTTP-protokollen

- Er en applikasjonlagsprotokoll, som inneholder meldingskoder og metadata som trengs for å overføre websider oppå det som er av TCP-headere og IP-headere
- Har en klient-tjener modell
- Bruker TCP som transport
- HTTP har ingen tilstandsinformasjon, dette er en fordel for protokoller som trenger å være effektive

Persistente FORBINDELSER

Ikke persistent

- Tjeneren leser forespørselen, svarer, lukker TCP-forbindelsen
- Hver overføring lider av TCP sin gradvise økning i senderate (slow start)
- Mange nettlesere åpner flere parallelle forbindelser for å prøve å ta mer kapasitet og få ting til å fortere omgå et designproblem

Persistent

- Standard for HTTP/1.1
- Over samme TCP-forbindelse: tjeneren leser forespørselen, svarer, leser ny forespørsel
- Klienten sender forespørsler for alle de objektene den trenger så fort den mottar hoveddokumentet (HTML)
- Færre RRTer, mindre slow start
- Persistent med pipelining: spør etter mange objekter på én gang (enda færre RTTer)

COOKIES

- Tar vare på tilstand
- Lager spesialtilpasset innhold for klienten, basert på lagrede HTTP-forespørsler
- Vise spesialtilpasset reklame

E-post

- Simple Mail Transfer Protocol (SMPT)
- Bruker TCP (alt må komme frem)
- 3 faser:
    - Håndtrykk

    - Overføring av beskjeder
    - Avslutning
- Ascii 7-bit

KORREKTE PÅSTANDER FRA UNIVERSET

Transportlaget

Transportlaget er ansvarlig for å levere data mellom applikasjoner

ARP-tabellen lagrer oversettelser mellom IP-adresser og MAC-adresser

ARP oversetter IP-adresser til MAC-adresser

MAC-adresser er unike identifikatorer for nettverkskort

En MAC-adresse kan endres ved hjelp av en programvare

TCP/IP er en protokoll som brukes for å koble til og kommunisere over internet

Switcher opererer på linklaget og bruker MAC-adresser for å videresende data

Lagdeling

En hypervisor brukes til å kjøre og administrere virtuelle maskiner

BIOS står for Basic Output Input System

Et nettverksprotokollsett er en samling av flere protokoller som arbeider sammen

Et operativsystem sørger for at flere brukere kan bruke datamaskinen samtidig

Minnekontroll holder styr på hvilke deler av minnet som er I bruk og hvilke som er ledige

Router kobler forskjellige subnet sammen

VPN gir en sikker forbindelse over et offentlig nettverk

En hub sender data til alle enheter koblet til den, uavhengig av destinasjonen

PoE (Power over Ethernet) tillater overføring av elektrisk strøm over nettverkskabler

En driver er en programvare som lar operativsystemer kommunisere med maskinvareenheter

Protokoller trenger en header

Systemgjenoppretting I Windows kan brukes til å gjenoprette systemet til en tidligere tilstand

Mikrokjerner er modulære og utvidbare

Monolittiske kjerner er effective: alle komponenter kan kommunisere direkte

Monolittiske kjerner er store og komplekse

I monolittiske kjerner kjører all beskyttet kode I same beskyttelsesområde

FORKORTELSER

ARP: Address Resolution Protocol

CIDR: Classless Inter-Domain Routing

CDN: Content Delivery Network

DHCP: Dynamic Host Configuration Protocol

DNS: Domain Name System

FIFO: First in first out

HTTP: Hypertext Transfer Protocol

HTTPS: Hypertext Transfer Protocol Secure

ISO/OSI: Open System Interconnection

IP: Internet Protocol

ISP: Internet Service Provider

LAN: Local Area Network

MAC: Medium Access Control

MAN: Metropolitan Area Network

MSC: Messaging Service Center

NAT: Network Address Translation

NFC: Near Field Communication

PDU: Protocol Data Unit (beskjed)

P2P: Peer to peer

RTT: Round Trip Time

SAN: Storage Area Network

SAP: Service Access Point

SDU: Service Data Unit (nyttelasten i PDU)

SMTP: Simple Mail Transfer Protocol

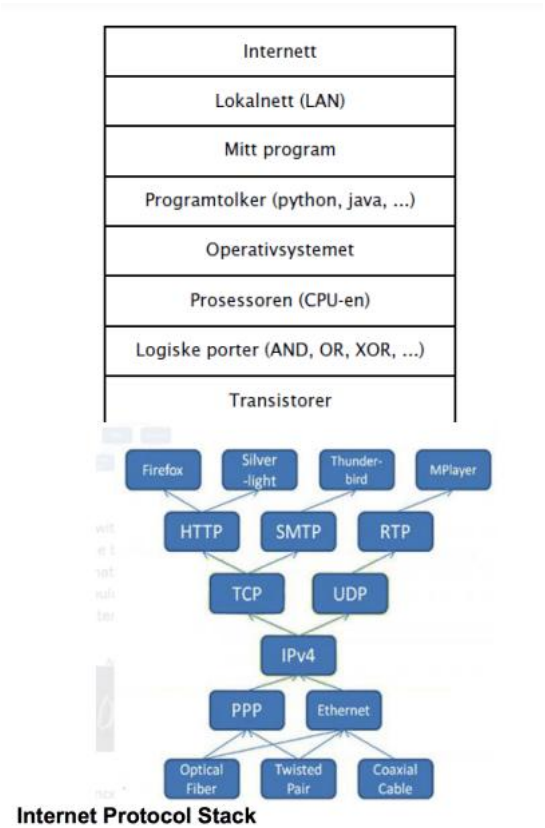
SSL: Secure Socket Layer

TCP: Transmission Control Protocol

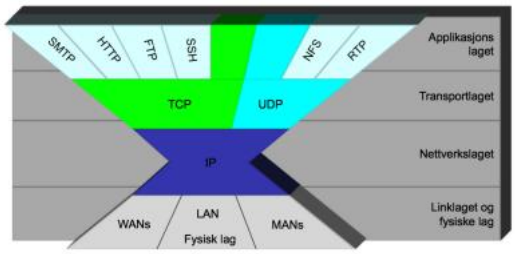
TCP/IP: nye ISO/OSI – modellen

TLD: Top Level Domain

UDP: User Datagram Protocol



Internet Protocol Stack



Kallenavn: "Timeglass-modellen"

VPN: Virtual Private Network

WAN: Wide Area Network

SSH: Secure Shell

FTP: File Transfer Protocol

UTREGNINGER

En maskin i nettet har IP 192.168.1.5 = 11000000.10101000.00000001.00000101

Nettverksmasken er en 32-bit netmask/subnet mask som maskerer og separerer en IP-adresse inn i network-adressen og host-adressen (hver IP-adresse består av disse to komponentene). Subnettmasken lages ved å sette alle network bits til '1' og alle host bits til '0'

Nettverksmasken er 255.255.255.0 = 11111111.11111111.11111111.00000000

REGNE UT SUBNETTET FRA EN IP + NETTVERKSMASKE

For å finne subnettadressen til maskinen må man gjøre en bitvis AND-operasjon mellom IP-adressen og nettverksmasken:

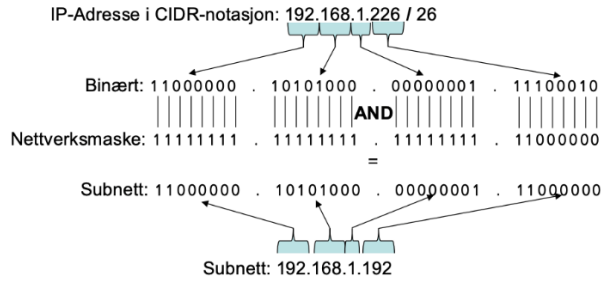
11000000.10101000.00000001.00000101

* 11111111.11111111.11111111.00000000

= 11000000.10101000.00000001.00000000 = 192.168.1.0

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

Subnettet til maskinen i CIDR-notasjon vil bli 192.168.1.0/24



REGNE UT BROADCAST ADRESSE FRA EN IP + NETTVERKSMASKE

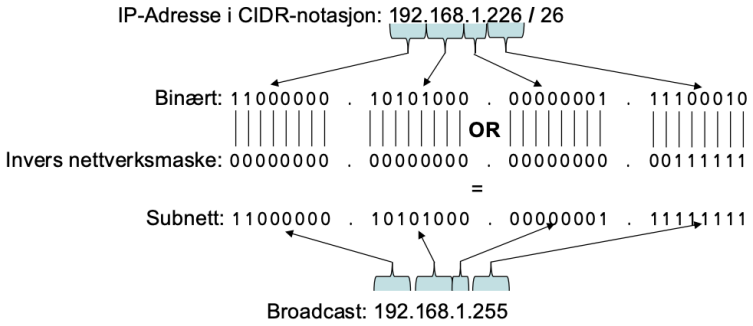
For å finne broadcastadressen må man gjøre en bitvis OR-operasjon mellom maskinens IP-adresse og bit kompliment (bitvis invers) av nettmasken

11000000.10101000.00000001.00000101

\+ 00000000.00000000.00000000.11111111

= 11000000.10101000.00000001.11111111 = 192.168.5.255

Broadcastadressen/kringkastingadressen til IP-adressen i CIDR-notasjon vil bli 192.168.5.255/24



PRØVEEKSAMEN

Operativsystemer

Hvilke påstander er sanne:

- Operativsystemet sørger for at flere brukere kan dele en datamaskin
- Det er operativsystemet sin oppgave å fordele ressurser på programmene som skal kjøre.
- Mobiltelefoner kjører kun ett program om gangen, og trenger derfor ikke et operativsystem.
- Operativsystemet sørger for at flere programmer kan kjøre samtidig.
- Et operativsystem tilbyr APIer for at programmer skal kunne bruke tjenester som lagring og nettverk.
- Operativsystemet skjuler kompliserte maskinvare detaljer for å gjøre det enklere å utvikle programvare.
- Vanlige programmer kjørere i kjernemodus fordi dette er mer effektivt.
- I operativsystemlaget er det modulen Enhetskontroll som har alt ansvaret for sikkerhet.

IPv6

Hva er hovedmotivasjonen til å bytte ut IPv4 med IPv6?

- Det blir flere tilgjengelige porter per IP-adresse
- Øke antallet globalt adresserbare IP-adresser
- Rutere slipper å koble sammen IP-adresser og MAC-adresser
- Gjøre det vanskelig å utføre "man-in-the-middle" angrep.

Overføringshastighet

Laste ned fil på 800 megabyte. Hastigheten er 50 megabit per sek ned og 25 megabit per sek opp. Hva er den teoretisk korteste overføringstiden?

- 128 sek
- 60 sek
- 32 sek
- 256 sek
- 10 sek

Punktnotasjon til CIDR / subnett-adresse / broadcast-adresse

IP-adresse: 213.58.192.58

Nettverksmaske: 255.255.255.224

Hva er IP-adressen i CIDR? 213.58.192.58/27

Hva er adressen til maskinen sitt subnett? 213.58.192.32

Hva er broadcast-adressen i dette subnettet? 213.58.192.63

Antall IP-adresser

Nettverksmaske: 255.255.255.224

Hvor mange gyldige IP-adresser kan tildeles verter i nettverket?

Transportprotokoller

Tjenestene FTP, IMAP og http bruker protokollen TCP. DHCP bruker UDP. DNS brukes begge.

Content Delivery Network

Hvilke av alternativene er positive effekter av CDN:

- Reduserer forsinkelsen for tilgang på innhold ved å flytte det nærmere brukeren fysisk.
- Sparer ressurser i kjernenettverket.
- Muliggjør bruk av kryptering under overføringen av data.
- Hindrer overbelastning av tjeneren ved å spre lasten over mange maskiner.
- Gjør at nettverket ikke kan kontolleres av en statsmakt eller et selskap.
- Flere maskiner fører til økt strømforbruk.
- Sikrer mot datainnbrudd.