

1)

En datapakke består av flere byte. Disse pakkene består derfor av en del informasjon som skal transporteres, og det er flere ting som må tas hensyn til. Arbeidsoppgavene deles derfor opp i lag. Denne lagdelingen kalles blant annet for Internett Protocol Stack og består av disse lagene:

-1: Physical layer

-3: Network layer

-4: Transport layer

-5: Application layer

2)

Lag 1: Protokollene i dette laget har som oppgave å sørge for at en bit blir overført uten feil mellom to forskjellige steder (punkter). Protokollene på dette laget innehar også beskrivelser av bits

Lag 2: Datalink-laget, forteller hva som må til (beskrives) for å kunne få overført datapakker mellom to punkter eller noder. Protokollene som opererer i dette laget, har derfor noen forskjellige arbeidsoppgaver. De inneholder for det første beskrivelser av hvordan den spesifikke delen av datapakken er bygd opp, slik at den kan overføres til neste punkt/node.

Protokollen innehar også informasjon om MAC-adressen til neste/punkt node slik at riktig punkt kan ta imot pakken. Feilkontroll og flytkontroll er også en viktig del, da det sikrer feilfri overføring mellom to noder. Til slutt finnes det også noe informasjon om hvilken protokoll på laget over som skal motta dataene.

Lag 3: Dette laget kalles nettverkslaget, og protokollene som arbeider der innehar informasjon om IP-adressen, og også informasjon som muliggjør like forhold under overføring. Dvs at eksempelvis to pakker bruker cirka like lang tid på å komme seg gjennom nettverket. Igjen inneholder også protokoller her info om hvilke protokoller på laget over som skal motta dataene.

Lag 4: Dette laget er transportlaget. Protokoller her, har som klar hovedoppgave å overføre datablokker mellom to endepunkter. Der endepunkter eksempelvis kan være et program på en PC, til et program på en annen PC. Ulikt fra de andre lagene, skal protokoller her også skjule funksjoner i underliggende nettverksavhengige lag fra lagene lokalisert over.

Lag 5: Applikasjonslaget. Protokoller her sørger for at data fra applikasjonene på hver side får, og sender de dataene som er nødvendige for at programmet skal fungere som normalt.

3)

Laginndeling ble gjeldende hovedsakelig fordi det er store mengder informasjon som skal med på en datapakke som skal sendes fra et sted på internett til et annet. Eksempel på slik informasjon som må tas hensyn til er:

- Hvilken datamaskin som skal ha datapakken, og hvilken applikasjon på den gjeldende maskinen som skal ha pakken.

- Feil i datapakken må også kunne behandles på en god måte, info om hvordan dette skal gjøres må ligge tilgjengelig

Det er også ytterligere info som skal med, og det problemet med hvor dette skal ligge i datapakken oppstår. Med andre ord, hvordan skal det være mulig å organisere all informasjonen. Det er her laginndelingen ble gjeldende. Lagdelingen er derfor opprettet/laget for å kunne organisere alle disse arbeidsoppgavene/informasjonen på en god måte

4)

Det finnes en rekke forskjeller mellom OSI-modellen og Internet protocol stack. En er antall definerte lag. OSI-modellen opererer med 7, mens TCP- modellen har 5.

En annen viktig forskjell er at OSI-modellen hovedsakelig definerer hvordan kommunikasjon over nettet skal foregå mellom systemer. Altså hvordan systemer som har en internettilkobling skal kommunisere med hverandre. TCP/IP er mer fokusert på hvordan den spesifikke datamaskinen skal kunne koble seg opp på internett.

Andre mindre forskjeller:

- OSI modellen sin "header" er 5 bytes, mens TCP/IP er 20.

- OSI-modellen sitt transport lag er bare tilkoblings- orientert, mens TCP/IP sitt transport lag opererer både med og uten tilkobling

5)

En protokoll er et sett med regler, eller fremgangsmåter for overføring av data mellom eksempelvis datamaskiner. En protokoll dekker ofte en eller flere av de etablerte lagene som er beskrevet ovenfor.

Det finnes en rekke ulik informasjon som er til stede i en protokoll. Den inneholder først og fremst hvilke typer datapakker som skal sendes. Videre hvordan disse ulike pakkene er bygd opp, og hvilken informasjon som er tilgjengelig i disse. En protokoll inneholder også informasjon om hvor i dataoverføringssekvensen overføringen befinner seg. Da om det er i starten, i dataoverføringen eller i avslutningen.

6)

En bit er den minste enheten for lagring av data. Et bit lagrer enten en 0, eller 1 som satt sammen representerer ulik informasjon. En byte er da en samling av bits, rettete sagt utgjør 8 bits, 1 byte. En datapakke er da en enda større lagringsenhet, og består av flere bytes.

7)

Hvordan en datapakke er bygd opp kommer noe an på hva slags type datapakke det er snakk om, og hvilke protokoller den opererer etter. Hovedsakelig er datapakker strukturert etter dette prinsippet: Header, Hoveddel (Data) og Footer.

Headeren inneholder informasjon om dataen som følger pakken. Dette kan typisk være hvor pakken kommer fra (adresse) og hvor den skal. Hvilken protokoll som blir benyttet samt lengden på pakken. Det er også vanlig at headeren gir noe informasjon hvilken pakke dette er, i en rekke av flere pakker. Altså hvilket nummer den er i rekken.

Hoveddelen er selve dataen. (kalles payload på engelsk). Noen nettverk opererer med forhåndsbestemte lengder på datapakkene og da er denne hoveddelen fylt med blank informasjon for å kunne møte kriteriet til nettverket. Ellers er den lengden nødvendig for å kunne holde alle dataene.

Footeren er slutten av pakken, og inneholder ofte en liten info som forteller mottakeren akkurat det. Den inneholder også en type feilsjekk. Altså er riktig data mottatt. Dette kan gjøres på flere måter, men en vanlig metode er CRC-metoden.

8)**320 Mbyte på en 100 Mbit/s linje**

Deler først 100 på 8, for å finne hastigheten i Mbyte/s

$$100/8 = 12,5$$

Deler deretter 320 på 12.5 for å finne antall sekunder

$$320/12.5 = 25.6$$

Det vil ta cirka 26 sekunder å overføre 320 MB på denne linjen.