Høgskolen i Østfold

Løsningsforslag

EKSAMEN

Emnekode: **ITF20205**	Emne: **Datakommunikasjon**
Dato: **04. Des 2015**	Eksamenstid: kl. **9:00** til kl. **13:00**

Hjelpemidler: • 4 sider (A4) (2 ark) med egne notater. • Kalkulator. • Gruppebesvarelse, som blir delt ut til de som har levert innen tidsfristen	Faglærer: Erling Strand

Eksamensoppgaven:
Oppgavesettet består av totalt 6 sider, som består av én forside, 3 sider med oppgaver, og 2 sider med vedlegg. Kontroller at oppgaven er komplett før du begynner å besvare spørsmålene.

Oppgavesettet består av 3 oppgaver. Alle spørsmålene teller likt. Alle svar må begrunnes.

Sensurdato: 4. Januar 2016
Karakterene er tilgjengelige for studenter på studentweb senest to virkedager etter oppgitt sensurfrist. Følg instruksjoner gitt på:
http://www.hiof.no/index.php?ID=7027

Alle svar må begrunnes

Oppgave 1

a) *Hva er hovedforskjellene mellom protokollene UDP og TCP? Nevn også noe om bruksområdene for de to protokollene. Altså hvilke typer applikasjoner hver og en av de passer best til, og hvorfor.*

UDP står for User Datagram Protocol og er en "connectionless" protokoll. Det vil si at forbindelsen ikke blir satt opp først. Det er data i første pakke. UDP er en usikker protokoll da den ikke gir noen respons tilbake om pakka har kommet fram, og veien som den tar kan være forskjellig for hver pakke som går. UDP inneholder lite "overhead", og det går da raskere å sende data med UDP enn TCP.

TCP står for Transmission Control Protocol og er en "connected oriented" protokoll. Det vil si at forbindelsen blir satt opp før data sendes. Alle pakkene som sendes i en melding følger samme veien. Pakkene kommer også fram i riktig rekkefølge og det er mulighet for retransmisjon ved feil. TCP-hodet inneholder nok info til å styre alt dette. Det blir da forholdsvis mye "overhead", som gjør at det går tregere med TCP enn UDP.

TCP er egnet der hvor det er viktig at alle pakker kommer riktig fram, selv om det ikke går så hurtig. F.eks. hvis en webside blir overført, eller i en forbindelse med banken. UDP er egnet der hvor det ikke er så kritisk hvis en pakke skulle bli borte, men det er viktig at forbindelsen er rask. F.eks. der hvor lyd og/eller film blir overført.

b) *I hodet på både en TCP og en UDP pakke er det to felt om portnummer; Source port # og Dest. Port #. Hva brukes de til?*

Portnummer er veien inn til en applikasjon på laget over. Hver applikasjon har sitt eget portnummer. Man kan sammenligne det med nummeret som står på døra inn til applikasjonen. Når en host (klient) f.eks. skal hente en webside fra en webserver, vil den sette Dest Port 80 på pakken som forespør om websiden. Da vil webserver applikasjonen ta i mot datapakka. Dest port # betyr altså at det er portnummeret på applikasjonen som pakken skal til. Klienten vil også si ifra hvilken port nummer den vil ha svaret på. Det blir Source Port nummer. Klienten velger blant de ledige portnumrene. Når webserveren sender websiden tilbake til klienten, vil den sette Dest Port nummer på den pakka, til det Source Port nummeret som klienten ville ha svaret på.

c) *Forklar litt om multipleksing på lag 4.*

Multipleksing betyr at flere blir mikset sammen til en. I demultipleksing blir disse flere skilt ut fra denne ene. Det finnes mange type multipleksing. På lag 4 brukes det i forbindelse med at data fra forskjellige applikasjoner blandes sammen til en datastrøm ned til IP laget, altså lag 3. Dette gjøres ved hjelp av portnummer. Hver applikasjon har sine portnummer. Dette portnummeret følger med dataene fra applikasjonen. Portnummer legges i lag 4 hodet på datapakka.

d) *I et LAN så er det en DHCP server. Hvorfor er den der, og hva er det den gjør? Hva måtte man ha gjort i en «host» for å komme på internet, hvis det ikke var en DHCP server tilgjengelig?*

DHCP står for Dynamic Host Configuration Protocol, og den brukes for å dele ut IP-adresse til en maskin som forespør om en IP-adresse.

Det finnes en DHCP server som dekker et eller flere subnett. Denne har et sett med IP-adresser som den kan dele ut fra, til maskiner som forespør om en IP-adresse.

Et eksempel på DHCP:

En maskin som blir slått på, og som er satt til å forespørre en DHCP-server om sin IP-adresse, sender en DHCP DISCOVER pakke til DHCP-serveren. Når denne kommer fram til DHCP-serveren, returnerer den IP-adresse i en DHCP-OFFER pakke. Hvis maskinen som mottar denne, godtar dette nummer, sender den en DHCPREQUEST pakke tilbake til DHCP serveren. DHCP-serveren sender tilbake en DHCPACK pakke. Samtidig settes en timer i gang. Enhver maskin må "oppdatere" sin IP-adresse jevnlig, ved å sende en melding til DHCP-serveren. Hvis DHCP-serveren ikke mottar en slik oppdatering innen timeren er gått ut, vil den frigjøre IP-adressen, som da kan deles ut til en annen maskin som forespør om en IP-adresse.

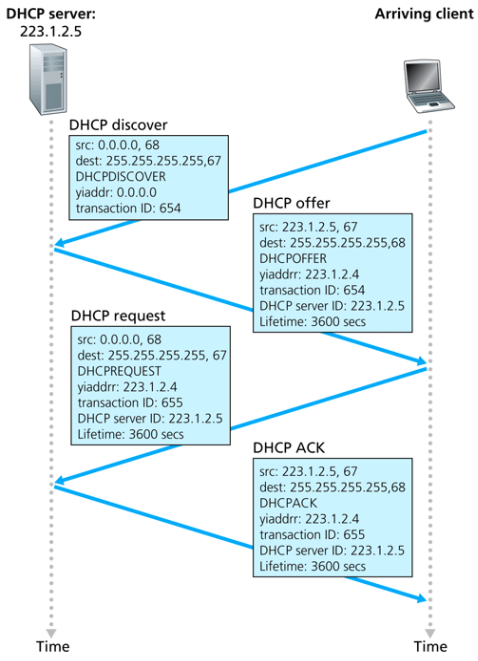


Figure 4.21 ♦ DHCP client-server interaction

Hvis det ikke hadde vært en DHCP server tilgjengelig måtte nettverksparametrene vært skrevet inn direkte, manuelt i host. Man måtte skrive inn IP adressen til host, og nettmasken på nettet. Dette måtte stemme med IP adressen til nettet som host'en er tilknyttet. Dessuten måtte IP adressen til gateway vært skrevet inn. Vanligvis bruker man også DNS, og da måtte IP adressen til en DNS server også vært skrevet inn.

e) Anta at du får følgende info etter en ping kommando:

Pinging www.princeton.edu [140.180.223.42] with 32 bytes of data:
Reply from 140.180.223.42: bytes=32 time=136ms TTL=244
Reply from 140.180.223.42: bytes=32 time=132ms TTL=244
Reply from 140.180.223.42: bytes=32 time=134ms TTL=244
Reply from 140.180.223.42: bytes=32 time=133ms TTL=244

Ping statistics for 140.180.223.42:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 132ms, Maximum = 136ms, Average = 133ms

Anta at du får følgende info etter en tracert commando:

Tracing route to www.princeton.edu [140.180.223.42]
over a maximum of 30 hops:

1 <1 ms <1 ms <1 ms c6500-h-1.hiof.no [158.39.165.3]
2 <1 ms <1 ms <1 ms halden-gw3.uninett.no [128.39.46.129]
3 2 ms 2 ms 2 ms ifi2-gw.uninett.no [128.39.254.241]
4 2 ms 2 ms 2 ms stolav-gw2.uninett.no [128.39.254.173]
5 10 ms 10 ms 10 ms dk-ore.nordu.net [109.105.102.66]
6 20 ms 20 ms 20 ms nl-sar.nordu.net [109.105.97.137]
7 104 ms 104 ms 104 ms us-man.nordu.net [109.105.97.139]
8 98 ms 98 ms 98 ms xe-2-3-0.118.rtr.newy32aoa.net.internet2.edu [109.105.98.10]
9 134 ms 135 ms 134 ms 216.27.100.5
10 130 ms 133 ms 132 ms remote1.princeton.magpi.net [216.27.98.114]
11 139 ms 142 ms 138 ms core-87-router.princeton.edu [128.112.12.130]
12 131 ms 134 ms 134 ms www-tmp.princeton.edu [140.180.223.42]

Trace complete.

1) *Hvor mange routere går datapakker til www.princeton.edu innom? – Nevn også hvor (alle steder) du finner ut av det, i svaret fra de to kommandoene ping og tracert (se over).*

Datapakkene går innom 12 routere på veien til www.princeton.edu. Det kan ses på svaret på tracert kommandoen. Her er alle routerne listet opp. Tracert får tak i denne info ved å sette verdien på TTL. TTL minker med en for hver router. Når TTL verdien har kommet til 0, vil ikke pakken sendes lenger, og routeren hvor TTL ble 0, sender info tilbake til senderen om at pakken er stoppet, og den sender også info om navn og adresse på routeren hvor det skjedde. Først setter tracert TTL =1. Da får den svar fra første routeren. Dertter sender den en ny pakke med TTL=2. Da får den svar fra den andre routeren osv.
Det er også mulig å se at det er 12 routere på TTL verdien på svaret fra en ping kommando. Hvis TTL verdien begynner med 0 og blir 255 i første routeren. Vi ser at den har verdien 244 i ping svaret. 255 i 1..router, 254 i 2..router, og 244 i 12. router.

2) *Anta at datahastigheten (den «fysiske») du har til www.princeton.edu er på 100 Mbit/s. (Altså $100 \cdot 10^6$ bit/s). Du skal bruke idle RQ overføring, med en pakkestørrelse på 4096 Byte. Hvor stor er effektiviteten på overføringen?*

I dataene ser vi at RTT = 133 ms.
Antall bit i pakken er L= 4096 [byte] · 8 [bit/byte] = 32768 [bit]
Bithastigheten R = 100 [Mbit/s] = $100 \cdot 10^6$ [bit/s]

Effektiviteten U blir da:

$$U = \frac{L/R}{RTT + L/R} = \frac{32768/100 \cdot 10^6}{133 \cdot 10^{-3} + 32768/100 \cdot 10^6} = \frac{32768}{13300 \cdot 10^3 + 32768} = \underline{\underline{2{,}46 \cdot 10^{-3}}}$$

3) *Hva blir den effektive datahastigheten på overføringen, - altså den hastigheten du som bruker opplever?*

Den effektive datahastigheten er lik (den fysiske) datahastigheten gager med effektiviteten:

$100 \cdot 10^6$ [bit/s] · $2{,}46 \cdot 10^{-3}$ = $246 \cdot 10^3$ [bit/s] = **246 Kbit/s**

f) Beskriv hvordan køkontroll virker i TCP.

Køkontroll er en metode for å styre kø. Kø kan oppstå i et nettverk ved at flere forbindelser går igjennom en nettverksenhet, f.eks en ruter. Summen av trafikken for hver forbindelse kan bli for stor for den ruteren, og da vil pakker bli tapt. TCP må ha mekanismer som vil kunne justere trafikken på sin forbindelse, slik at alle pakker slipper igjennom.

Nå har ikke nettverksenhetene i "vanlig" Internet, f.eks ruterne, mulighet til å gi beskjed om kø hos seg. Hadde den hatt den muligheten, kunne den ha gitt beskjed om køtilstanden tilbake til TCP forbindelsene. TCP må bruke andre midler for å finne ut av køtilstanden, og justere trafikken i henhold til den.

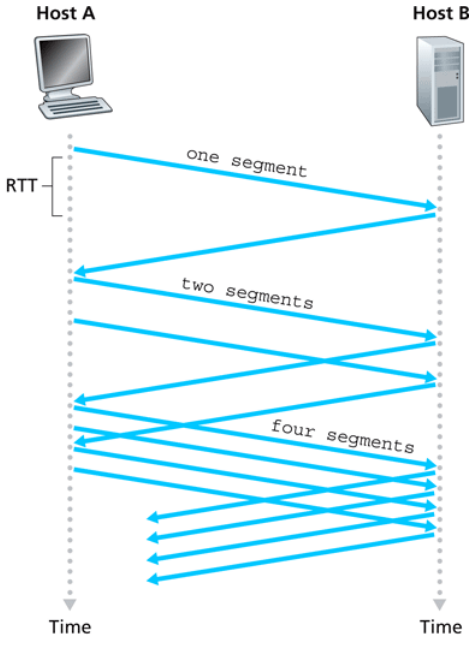


Figure 3.52 ♦ TCP slow start

TCP starter forsiktig ved å bruke "slow-start". Den begynner ved å sende kun en pakke. Neste pakke sendes først etter at ACK har kommet. Da vil den øke vindustørrelsen til 2. For hver ACK legger den på en pakke i vinduet. Det vil medføre at vindusstørrelsen dobles for hver gang (hver RTT). Slik fortsetter den inntil den oppdager at pakker blir borte (ved at timeout slår til). Da justerer den vindusstørrelsen tilbake til 1 pakke. (En pakke er MSS stor). Samtidig har den notert hva vindusstørrelsen var da timeout slo til. "Threshold" settes til halvparten av den verdien.

Nå vil den kjøre slow-start igjen, men den vil doble vindusstørrelsen kun inntil den har kommet til "Threshold". Det var jo den vindusstørrelsen hvor det gikk bra forrige gang. Ved neste dobling ble det jo pakketap.

Når vindusstørrelsen har nådd ”Threshold” verdien, går den inn i ”congestion avoidance” tilstand, hvor økning av vindusstørrelsen skjer mye mer forsiktig. I stedet for dobling, øker den med kun 1 for hver RTT. Slik øker den inntil det blir tap av pakker igjen.

Tap av pakker oppdages enten ved at timeout slår til, eller at det har kommet 3 duplikate ACK. Dvs at 2 like ACK har kommet 3 ganger. En duplikat ACK betyr at pakken kom riktig fram på andre forsøk. Køen er da ikke så stor. Hvis timeout slår til, betyr det at ingen pakker har kommet fram. Da er køen stor.

Hvis tap oppdages ved at 3 duplikate ACK har kommer, går den inn i ”Fast recovery” tilstand. Det vil si at vindusstørrelsen går tilbake til halvparten av hva den var da tap skjedde, og den går inn i ”congestion avoidance” tilstand. Dvs. pakkestørrelsen øker kun med 1 for hver RTT. Denne økningen fortsetter inntil det blir tap igjen.

Hvis tap oppdages ved at timeout slår til, går vindusstørrelsen tilbake til slow start tilstand. Vindusstørrelsen går da tilbake til 1.

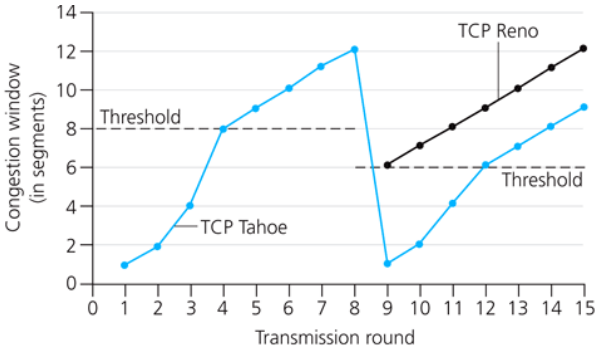


Figure 3.53 ♦ Evolution of TCP's congestion window (Tahoe and Reno)

For hver gang det blir duplikat ACK (to like ACK etter hverandre), øker duplikat ACK telleren.

Oppgave 2

a) *I IPv6 brukes ofte forkortede adresser. Skriv den hele og fulle adressen for den forkortede adressen: 2001:700:A00:24::2*

En IPv6 adresse består av 128/8 = 16 byte.

IPv6 adressen: 2001:700:A00:24::2 er skrevet på forkortet form. Det er 2 byte mellom hvert **:** .Vi ser at det er en plass hvor det er to kolon etter hverandre, slik :: . Her er det 0’ere som skal inn. Så mange som er nødvendig for å få en hel adresse. Innledende 0’ere er også blitt sløyfet.

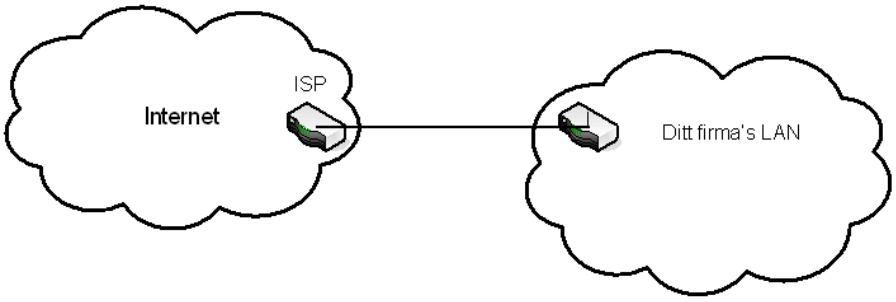
Den hele hele og fulle adressen blir da:

2001:0700:0A00:0024:0000:0000:0000:0002

b) *En host må ha en IP adresse for å kommunisere på internett. En host må også vite IP adressen til Gateway. Hvorfor må en host vite det?*

Hvis en host skal sende en IP pakke til en adresse som er utenfor lokalnettet (LAN) som hosten er på, må den sende pakken til Gateway'en på LAN. Gateway'en er forbindelsen ut til resten av internett.

Anta at du har startet et firma, og ønsker å ha et eget datanett til det firmaet. I dette datanettet skal alle host være direkte tilknyttet Internet, via en ruter (uten NAT). Av en internet-leverandør (ISP) får du nettadressen, med maske: 81.93.164.00/22. Det skal altså brukes IPv4.



c) *Hvor mange host kan du ha på dette nett?*

Maks antall host på et nett er gitt av antall bit i hostdelen av adressen.

/22 betyr at det er 22 bit i nettdelen av adressen. Da det er 32 bit i hele adressen blir antall bit i hostdelen av adressen: 32-22=10 bit. Antall host blir da:

2^{10}-2= **1022 host**

2 må trekkes fra fordi ingen host kan ha samme adresse som nettadressen, dvs der hvor alle bit i hostdelen er lik 0, og ingen host kan ha samme adresse som broadcastadressen, dvs der hvor alle bit i hostdelen er lik 1.

d) Hva blir broadcast-adressen på dette nett?

I broadcastadressen er alle bit i hostdelen av adressen lik 1.

Vi setter opp nettadressen på binær form, og gjør om alle hostbiter til 1. Med /22 må vi se på de to siste byte. Fet skrift angir bit i nettdelen av adressen.

164.00 -> **10100**100.00000000 Gjør om bitene i hostdelen til 1:

10100111.11111111 -> 81.93.167.255

Brodcastadressen er 81.93.167.255

Nå skal ditt firma utvide med to nye avdelinger. I begynnelsen ligger alle de tre avdelingene i samme hus. Du synes det er best å la disse avdelingene få hvert sitt datanett. I tillegg ønsker du å ha noen IP adresser ledige, samme antall som i et av de tre subnettene. Disse adressene skal brukes til utvidelser senere. Du skal derfor dele hele ditt datanett opp i fire like store subnett. «Subnet zero» og «all one subnett» skal være blant disse fire. Velg subnett zero som det nettet som er «ledig».

e) *Hva blir nettadressene til disse fire subnett, og hva blir nettmasken?*

For å lage 4 subnett må vi bruke 2 bit av hostdelen og gjøre om til nettdelen, fordi $2^2=4$

Vi setter opp de to siste byte på binær form. Fet skrift på bit indikerer nettdelen. Fet skrift og kursiv angir de hostbitene fra det originale nettet, som nå er brukt til subnett, og som da er blir nettbit

164.00 -> **10100*10***.00000000
165.00 -> **10100*01***.00000000
166.00 -> **10100*11***.00000000
167.00 -> **10100*11***.00000000
Vi ser at det nå er 8 bit i hostdelen. Da blir masken /24, eller 255.255.255.00.
Nettadressene til disse 4 nett, med maske, blir da

81.93.164.00 / 24
81.93.165.00 / 24
81.93.166.00 / 24
81.93.167.00 / 24

f) *Hva blir laveste og høyeste IP-adresse på en host på et av disse subnett? (Du velger selv hvilket subnett du ønsker å angi det på)*

Velger f.eks. nett 81.93.166.00 / 24
Laveste IP for en host er en over nettadressen: **81.93.166.01 / 24**
Høyeste IP for en host er en under broadcastadressen: **81.93.166.254 / 24**

g) *Nå skal et av disse subnett flyttes til en annen by. Du må da sette opp en punkt-til-punkt forbindelse til denne. Hvilket subnettnummer og maske, vil du gi ~~disse to~~ denne punkt-til-punkt forbindelsen~~e~~? Du skal bruke IP adresser fra de adressene som var ledig*

Bruker av subnett zero nettet, som er tenkt brukt til slikt. Dette punkt-til-punkt nettet har bare en host på hver ende, totalt to host. Trenger da bare 2 bit til hostdelen av adressen, da $2^2 – 2 = 2$. Dvs masken blir /30 (eller 255.255.255.252)

Velger adressene fra begynnelsen av adresseområdet. Ser på siste byte. Fet skrift angir bit som hører til nettdelen av adressene.:

0000000 -> **81.93.164.04 / 30**

h) *Nå skal bedriften lage fire nye kontorer, hvor det skal være plass til 14 host. Disse fire nye kontorene skal også ha hver sin punkt-til-punkt forbindelse. Hvilke nettadresser, med maske, vil du gi til disse nett? Du skal bruke av de IP-adressene som er ledige.*

For å få plass til 14 host på et nett må det være 4 bit i hostdelen av adressen, fordi 2^4-2=14
Antall bit i masken blir da 32-4= 28 Masken blir /28 (eller 255.255.255.240)

Fire nye punkt-til-punk forbindelser kan få adressene:
0000100 -> **81.93.164.08 / 30**
0000100 -> **81.93.164.12 / 30**
1111000 -> **81.93.164.244 / 30**
1111100 -> **81.93.164.248 / 30**

De fire nye kontorene kan få adressene
00010000 -> **81.93.164.16 / 28**
00100000 -> **81.93.164.32 / 28**
00110000 -> **81.93.164.48 / 28**
01000000 -> **81.93.164.64 / 28**

Oppgave 3

a) *Gi en beskrivelse av WiFi. Nevn hva standardene heter, hvilke frekvenser og bånd som brukes. Nevn også noe om aksessmetoden.*

WiFi er trådløst datanett. Det er IEEE som standardiserer de forskjellige WiFi.

Navn	Frekvens	Bånd
802.11 a	5 GHz	
802.11 b	2,4 GHz	13 stk 22 MHz
802.11 g	2,4 GHz	13 stk 22 MHz
802.11 n	2,4 og 5 GHz	
802.11 ac	5 GHz	

I WiFi brukes to forskjellige system når det gjelder aksess. Det er PCF (Point Coordination Function) og DCP (Distributed Coordination Function), I PCF er de ten basestasjon som spør hver enkel host om den har noe å sende. Ingen host kan aksessere mediet uten å ha blitt spurt.
I DCF kan hvilken som helst host i nettet ta initiativ til å sende. Da brukes CSMA/CA aksess.

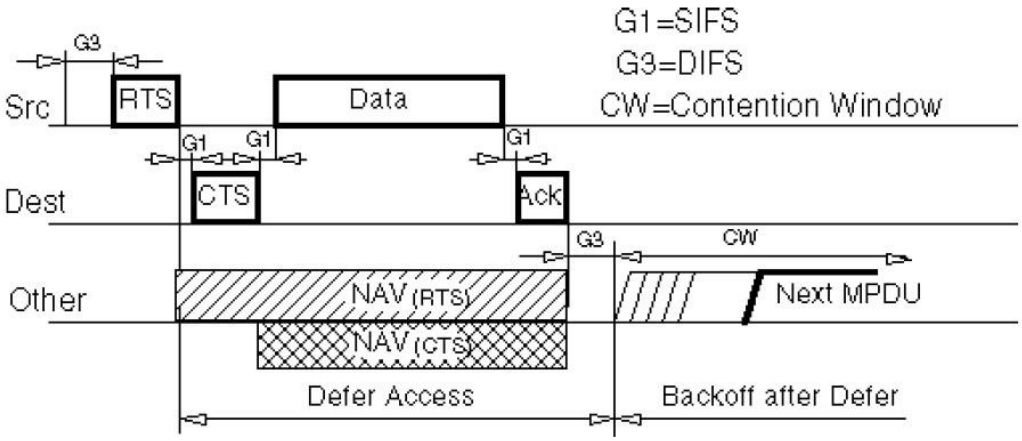
> **C**arrier **S**ense **M**ultiple **A**ccess med **C**ollision **A**voidance virker slik: En stasjon som ønsker å sende, må først lytte, for å høre om det er noen aktivitet på kanalen. Hvis det er aktivitet, må den vente. Så fort aktiviteten er over, må den vente ytterligere

DIFS tid før den starter sending. Hvis noen andre da har begynt å sende, må den fortsette å vente.

Når pakken er sent, venter den på ACK pakke fra mottager. Hvis den ikke kommer innen en viss tid, sender den pakken på nytt. Dette kan gjentas et visst antall ganger.

CSMA/CA kan også bruke RTS og CTS. Stasjonen som skal sende, sender da først en RTS pakke, hvor det er info om hvor lang tid sendingen vil pågå. Det er for at andre stasjoner ikke skal sende i denne perioden. Den mottagende stasjon svarer på RTS med en CTS pakke, med samme info om tid på sendingen. De stasjonene som kun hører mottager, vil dermed også ikke sende i den perioden. Når sende stasjonen har mottatt CTS, sende den datapakka

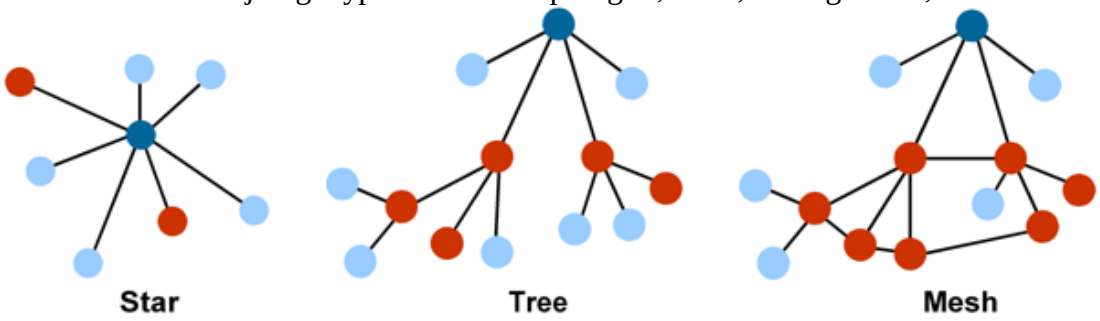
Nå skal DCF og PCF kunne virke samtidig i et nett. Det løses ved å innføre bestemte tidsintervall, med forskjellig lengde.. En pågående kommunikasjon gjør seg ferdig. Neste pakke i kommunikasjonen venter den korteste intervalltiden; SIFS, før neste pakke sendes. Når kommunikasjonen er ferdig, kan neste begynne å kommunisere. Da har PCF fortrinn. PCF kan starte sending etter tiden PIFS, som er litt lenger enn SIFS. Når den kommunikasjonen er ferdig, kan neste DCF kommunikasjon starte. De må vente DIFS tid, som er litt lenger enn PIFS tiden.



b) Anta at du har et ZigBee nett. Beskriv hvordan det er bygd opp, og virkemåten. Husk også å beskrive de forskjellige ZigBee-delene som kan inngå.

ZigBee bruker tre forskjellige noder: ZC(coordinator), ZR(router) og ZED(end device). I et nett finnes det kun en ZC. En ZR kan virke som en ruter som sender trafikken videre i nettverket, i tillegg til at den kan være en aktiv node. ZR kan også bli en ZC. ZED er den enkleste enhet, som kun kan sende eller motta data. Den kan ikke rute trafikk videre. ZC kalles også PAN coordinator. ZR kalles FFD(Full Function Device) og ZED kalles RFD(Reduced Function Device). I figuren om topologier over, er mørk blå en ZC, rød er en ZR og lys blå er en ZED.

Det kan være forskjellige typer nettverkstopologier, - star, tree og. mesh,



ZigBee bruker 16 kanaler i 2,4 GHz båndet. I tillegg brukes det i USA, og noen andre land, 10 kanaler i 915 MHZ båndet. I Europa brukes det 1 kanal i 868 MHz båndet, i tillegg til kanalene i 2,4 GHz båndet.

Enhver node har 64 bit adresse, som er unik for den noden. Ingen andre noder har den adressen. Det finnes også en kort adresse, som er på 16 bit, og som brukes i et nettverk. Så et ZigBee nettverk kan da ha litt over 65000 noder.

c) *Du skal dimensjonere et fiberoptisk anlegg, med bruk av SM fiber. Senderen har en innkoblet effekt i fiberen på +2,0 dBm. Fiberkabelen har en dempning på 0,3 dB/km, og en dispersjon på 3,5 ps/(nm·km). Lyskilden (laseren) har en spektral båndbredde på 2,0 nm. Det er ingen skjøter, og ingen kontakter. Du kan regne med innkoblingstap ved mottageren på 1,0 dB. Hva blir maksimal fiberstrekning når mottageren har en følsomhet på – 40,0 dBm, og det skal sendes data med en (ukodet) bithastighet på 5,0 Gbit/s?*

Vi må regne på både effekt og båndbredde for å finne ut hvilken som setter begrensningen.Regner først på effekten:
Setter systemmarginen til 3,0 dB (mellom 3,0 og 7,0)
$P_{inn} – P_{fiber} – P_{innk} – P_{syst} = P_m$
+2,0 - x· 0,3 – 1,0 – 3,0 = - 40,0
x· 0,3 = 40,0 –1,0 – 3,0 + 2,0 = 38,0
<u>x = 126,7 km</u>

Ved bruk av systemmargin på 7,0 dB blir det
+2,0 – x· 0,3 – 1,0 – 7,0 = - 40,0
x· 0,3 = 40,0 –1,0 – 7,0 + 2,0 = 34,0
<u>x = 113,3 km</u>

Regner så på båndbredden:
Med en ukodet bithastighet på 5,0 Gbit/s, kreves det en båndbredde på 2,5 GHz.
Det gir en maks dispersjon på: $\tau = 0{,}44/(2{,}5 \cdot 10^9) = 176 \cdot 10^{-12}$ s
Den dispersjonen fås ved y km:
$176 \cdot 10^{-12} = 3{,}5 \cdot y \cdot 2{,}0 \cdot 10^{-12}$
y = ~~88·10^{-12} / 7,5 ·10^{-12}~~ = <u>25,1 km (feil i tidligere utgave)</u>
$y = 176 \cdot 10^{-12} / 7{,}0 \cdot 10^{-12}$ = <u>25,1 km</u>

Maksimal fiberstrekning blir **25,1 km**

d) *Nå skal du dimensjonere et fiberoptisk anlegg med bruk av MM fiber. Fiberen har en dempning på 2,5 dB/km, og en båndbredde på 250 MHz·km. Senderen er en LED, med utstrålt effekt på -15,0 dBm. Fiberstrekningen er på 5,0 km. Det er en kontakt ved senderen og en ved mottageren. Du kan sette kontakttapet til 1,5 dB per kontakt. I tillegg er det et innkoblingstap ved senderen på 5,0 dB. Innkoblingstapet ved mottageren kan settes til 0, da den er innbakt i kontakttapet.*

1) *Hvilken følsomhet må mottageren ha?*

Setter opp effektbudsjettet:
$P_o - P_{innk} - P_{kont} - P_{fiber} - P_{syst} = P_m$

Ved å bruke en systemmargin på 3,0 dB får vi:

– 15,0 dBm – 5,0 dB – 2·1,5 dB – 2,5·5,0 dB – 3,0 dB = – 38,5 dBm

Ved å bruke en systemmargin på 7,0 dB får vi:

– 15,0 dBm – 5,0 dB – 2·1,5 dB – 2,5·5,0 dB – 7,0 dB = – 42,5 dBm

Vi bør velge en mottager med bedre følsomhet enn -38,5. F.eks **– 39,0 dBm**
Ved bruk av en systemmargin på 7,0, bør følsomheten være bedre enn – 42,5 dBm.
F.eks **– 43,0 dB**

2) *Hva er maksimal datahastighet på forbindelsen. Regn med at det skal brukes Manchster koding på datasignalet.*

Ved bruk av Manchester koding måbredden i Hz, være like stor som bithastigheten i bit/s
Båndbredden er (250 MHz·km) / 5,0 km = 50 MHz
Maksimal bithastighet er 50 Mbit/s

VEDLEGG

$$B = \frac{0{,}44}{\tau}$$

$$U = \frac{L/R}{RTT + L/R}$$

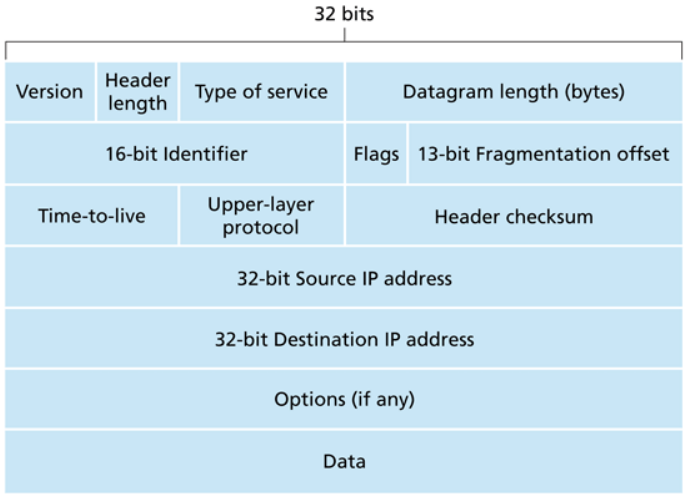


Figure 4.13 ♦ IPv4 datagram format

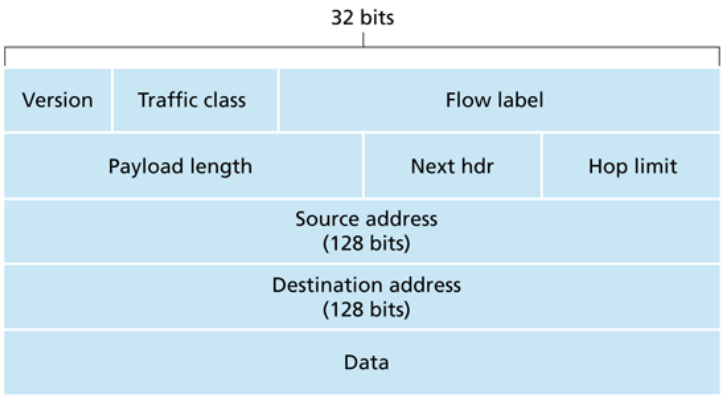


Figure 4.24 ♦ IPv6 datagram format

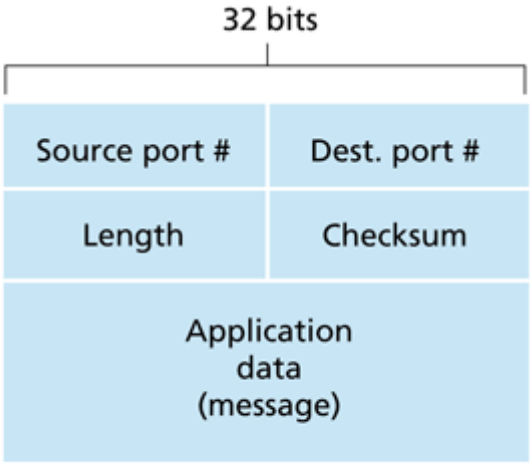


Figure 3.7 ♦ UDP segment structure

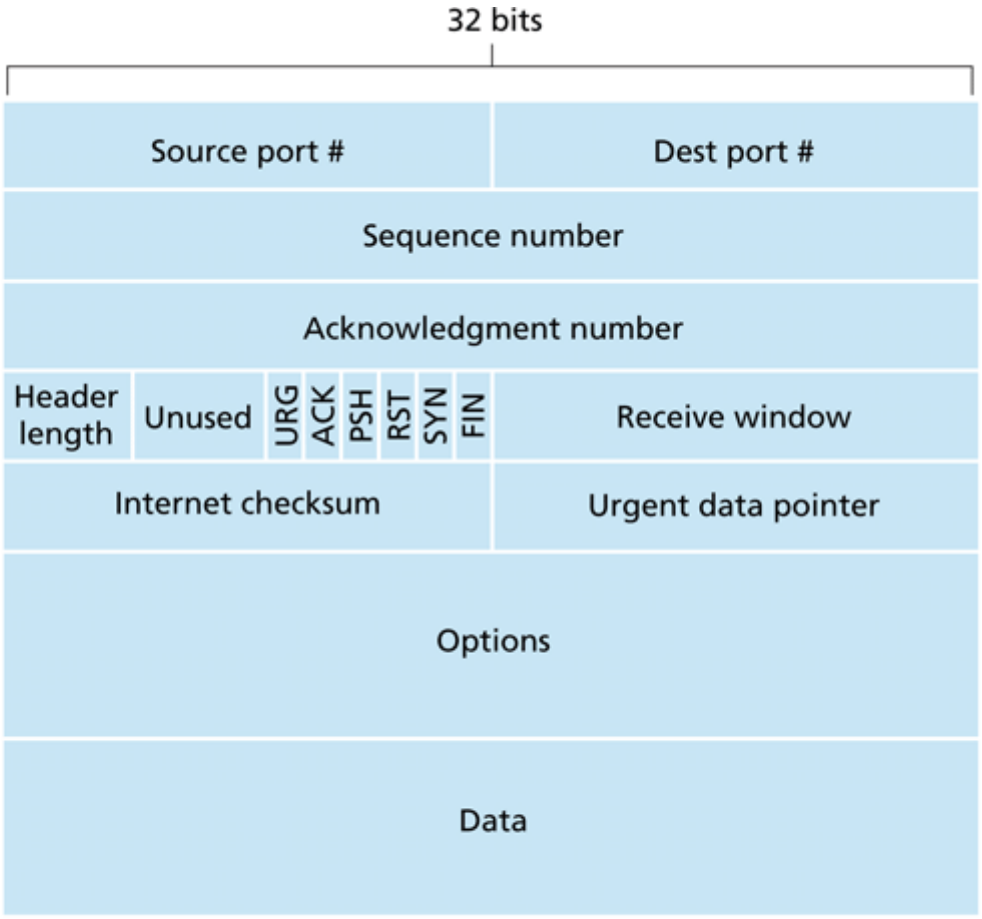


Figure 3.29 ♦ TCP segment structure