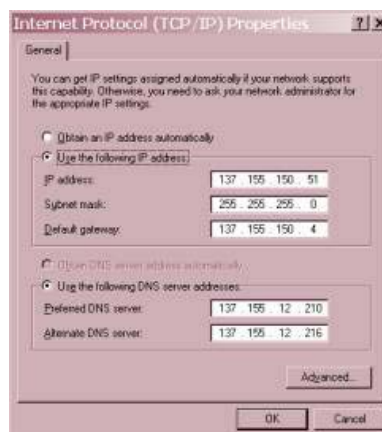


## Subnettmasker og IP adresser:

En introduksjon til subnetting for elever ved Oslo By Steinerskole,  
IKT driftsfag, VK1.

Del 1:  
route or shoute.

*Alle noder (node = host) på et TCP/IP nettverk MÅ ha en subnetmaske.*



Vi skal snakke litt om hva subnettmasker er og hvorledes TCP/IP bruker subnettmasken til å bestemme om dette er en datapakke som er til en lokal maskin eller om den skal til et annet nettverk, eventuelt ut på Internet. Litt senere skal vi se på hvordan vi kommer frem til hvilken subnettmaske en maskin, eller et nettverk skal ha i en gitt situasjon.

En subnettmaske er et 32 bits adresse, akkurat på samme måte og i samme format som en IP adresse. Subnettmasken brukes til å "blokkere" eller "maskere" en bestemt del av IP adressen for å kunne se hva som egentlig er nettverksadressen, og hva som er hostadressen. Dette er helt nødvendig å gjøre for å kunne si med sikkerhet om en bestemt IP adresse er på det lokale nettverket, eller om den tilhører et annet nettverk.

Alle noder på nettverket MÅ ha en subnettmaske. Det finnes to typer subnettmasker; default og custom. Det kan godt hende at det er IP adresseklassens default subnettmaske. Det kan også være en annen subnettmaske som du har laget (beregnet altså en custom subnettmaske) selv for å dele opp nettverket i flere små nettverk. Dette gjør man ofte for å redusere nettverkstrafikken, eller av sikkerhetshensyn.

**Default subnettmasker:**

En default subnettmaske brukes på nettverk som IKKE er delt opp i mindre enheter (mindre nettverk). Default subnettmaske er forskjellig for de forskjellige nettverksklassene (klasse A, B og C vi holder klasse D og E utenfor diskusjonen foreløbig).

I en subnettmaske (binær form!!) er alle bits i nettverksadressen satt til 1 og alle bits i hostadressen (IP adressen som noden har) er satt til 0.

klasse	1oktett	2oktett	3oktett	4oktett	desimal
A	11111111	00000000	00000000	00000000	255.0.0.0
B	11111111	11111111	00000000	00000000	255.255.0.0
C	11111111	11111111	11111111	00000000	255.255.255.0

Et eksempel på en klasse B adresse med standard (default) subnettmaske:

IP adresse	131.107.16.200
Subnettmaske	255.255.0.0

Dette gir oss følgende:

Nettverks ID:	<i>131.107.y.z</i>
Host ID:	<i>w.x.16.200</i>

## Hvordan bestemmes om en IP pakke er på vei til en lokal maskin eller til et annet nettverk?

Dette foregår ved bruk av en *logisk operator* og vi snakker om boolsk matematikk, en *logisk AND operasjon*. AND operasjoner beskrives vi med tegnet " & ". Den logiske operasjonen, den boolske matematikken blir utført av TCP/IP, normalt trenger du ikke gjøre dette selv. Det må du dersom det er noe galt og du skal forsøke å finne feilen, såkalt "trøbbelskyting".

### Hva skjer?

Når TCP/IP blir startet, og det blir det når maskinen starter (booter), vil nodens egen IP adresse bli ANDet med sin egen subnettmaske.

Før noden sender ut IP pakker vil den ANDe IP adressen denne pakken skal sendes til med den samme subnettmasken (sin egen!). Hvis resultatet av en AND med egen IP adresse og egen subnettmaske OG resultatet av en AND med mottakerens IP adresse og avsenderens egen subnettmaske blir 100 % maks vil maskinen konkludere med at mottakerens IP adresse er på det samme lokalenettverket. Hvis de to operasjonene gir forskjellig svar; om det så er bare en eneste bit av de 32 bitene i adressen vil det resultere i at maskinen ikke sender pakken ut på lokalt nett, den sendes rett til gatewayens IP adresse. En gateway er ofte en ruter, men det kan godt være et modem eller ISDN kort.

For å ANDe IP adressen og subnettmasken vil TCP/IP sammenligne adressen og subnettmasken bit for bit. Husk at alt dette foregår med binære tall!! Det er den eneste tallformen maskinen kan bruke. Den sammenligner disse binære tallene etter et bestemt system som altså kalles en logisk AND og er en boolsk, logisk operasjon, ikke et matematisk regnestykke. Dette er ikke algebra (pust ut ☺).

### Meget kort repetisjon:

Systemet for logisk AND er slik:

Bitkombinasjon	Resultat
1 AND 1	1
1 AND 0	0
0 AND 1	0
0 AND 0	0

Altså: 1 AND 1 er 1, alt annet blir 0.

### Et eksempel:

Vi ser på hvordan dette ville se ut i desimal form (husk det er binære tall som "ruler"!)...

IP adresse:	159	224	7	129
subnettmaske:	255	255	0	0
-----				
"Resultat"	159	224	0	0

Dette ser jo greit ut, hvis det bare hadde betydd noe...

Vi tar et eksempel: (i binær form selvsagt, dette er vanskelig å se desimalt!)

IP adresse:	10011111	11100000	00000111	10000001
subnettmaske:	11111111	11111111	00000000	00000000
-----				
resultat A:	10011111	11100000	00000000	00000000

La oss si at dette er "første runde" i prosessen med ANDing, den delen som skjer i det TCP/IP starter opp.

La oss nå si at vi vil sende noen datapakker til en annen maskin, en maskin med en IP adresse i samme nettverk som vårt eget. Vår IP adresse (med default subnettmaske) er:

159.224.7.129

mottagerens IP adresse er:

159.224.7.84

hvis vi nå ANDer mottageradressen med vår egen får vi (binært):

mottagers IP:	10011111	11100000	00000111	000000
egen subnettmaske:	11111111	11111111	00000000	000000

-----  
resultat B:            10011111    11100000    00000000    00000000

La oss kalle dette "runde to" i prosessen.

Nå skal vi sammenligne resultat A med resultat B.

resultat A:	10011111	11100000	00000000	00000000
resultat B:	10011111	11100000	00000000	00000000

Vi skal IKKE ANDe dem, eller utføre noen annen logisk operasjon, vi bare ganske enkelt ser på de binære tallene og ser om de er like eller ikke, og i dette tilfellet er de 100 % makne. Det betyr at mottagermaskinen er på vårt lokale nettverk og vi kan sende datagrammene (pakkene) direkte til mottageren. Hadde de vært forskjellige ville avsender måtte sende rammene til den lokale ruter (gateway), enten det er en "ekte" ruter eller det er en vanlig PC med to nettverkskort og IP forwarding (NAT) som dermed fungerer som en ruter. Det siste er for så vidt også en "ekte" ruter, men det er ikke uvanlig å tenke på en liten, dedikert boks (fra Cisco? ☺ ) når vi ser for oss en ruter. Alle rutere som er *backbone rutere* på Internet er spesielt bygget for oppgaven, det er ikke "*multi-homed computers*", altså datamaskiner med mer enn et nettverkskort.

### Et annet eksempel:

Vi ser på en situasjon hvor mottageren ikke er på det lokale nettverket, men på en maskin et helt annet sted i verden.

Først vår egen "runde 1" ANDing:

IP adresse:	10011111	11100000	00000111	10000001
subnettmaske:	11111111	11111111	00000000	00000000
-----				
resultat A:	10011111	11100000	00000000	00000000

Normalt vil vi finne frem til mottagerens IP adresse via et DNS oppslag, vi tar utgangspunkt i at det er ferdig, og at det gikk greit. Mottagerens IP adresse er:

63.125.54.3

i binær form: 10011111 11100000 00000111 000000

Først har vi ANDet vår egen IP adresse med vår egen subnettmaske:

egen IP adresse:	10011111	11100000	00000111	10000001
subnettmaske:	11111111	11111111	00000000	00000000
-----				
resultat A:	10011111	11100000	00000000	00000000

Nå må vi ANDe mottagerens IP adresse med vår egen subnettmaske (default klasse B) og herfra tar vi dette binært:

mottagers IP:	00111111	01111101	00110110	00000011
egen subnettmaske:	11111111	11111111	00000000	00000000
-----				
Resultat B:	00111111	01111101	00000000	00000000

La oss sammenligne resultatene og se om de er like:

Resultat A:	10011111	11100000	00000000	00000000
Resultat B:	00111111	01111101	00000000	00000000

Vi ser, med en gang at dette ikke er 100 % likt, det er flere bits som ikke stemmer, og det ville ha vært nok med bare en eneste bit som er forskjellig. Husk på at vi skal ikke utføre noe matematisk eller logisk på dette, bare sammenligne og se om det er forskjell.

Denne pakken er ikke lokal, og derfor sender vi den til ruterens, ikke ut på LAN. Hvis vi skulle forsøke å gjøre den siste operasjonen om igjen, desimalt ville den se slik ut:

IP adresse:	159	224	7	129
subnettmaske:	255	255	0	0
-----				
”Resultat A”	159	224	0	0
IP hos mottager:	63	125	54	3
egen subnettmaske:	255	255	0	0
-----				
”Resultat B”	63	125	0	0

Nå må vi sammenligne resultat A med resultat B:

”Resultat A”	159	224	0	0
”Resultat B”	63	125	0	0

Når vi sammenligner disse to tallene ser vi med en gang at de er ulike, og derfor er mottager på et annet nettverk enn vårt og datagrammet må sendes til ruterens.

Vi ser nå (håper jeg) at dette er mye lettere å ”se” dersom det er binært, så sant vi kan å ANDe da...

I del 2 av dette kompendiet skal vi se på den andre årsaken til å bruke subnetmasker, de såkalt spesielle subnetmaskene, de som brukes til å segmentere et nettverk, enten av sikkerhetskyn, eller for å redusere nettverkstrafikken.