

Grunnleggende kunnskap i nettverk.



Johan Futsæther
28.07.01 Red. 22.03.07

Kilder:

TechRepublic,
Network Essentials,
Innføring i nettverk (Ragnar Iversen),
Datakommunikation (Ewert)

For de fleste som arbeider med datanettverk, er det viktig å forstå hvordan TCP/IP virker. Det er særlig viktig å kunne forstå hvordan vi "adresserer oss" på internett. Med "adressering" i denne sammenheng, menes systemet for hvordan vi kan få sendt en datapakke fra en maskin til en annen maskin på et nettverk/Internett. Ser vi litt stort på det, så kan vi godt tenke oss systemet for hvordan et vanlig brev kan bli sendt fra en person til en annen. På konvolutten må vi oppgi personens navn, adresse, by og land. IP-adressering virker i prinsippet på samme måte. De datapakkene vi ønsker å sende, "pakkes" inn i en "konvolutt", merkes med en adresse, og sendes ut på nettverket/Internett. Siden absolutt alle maskiner (host) som er på nettverket/Internett teoretisk har en helt unik adresse, vil pakken til slutt komme frem til mottakeren. En adresse på internett består av et 32-bits ord som er oppdelt i 4 oktetter. En IP-adresse kan dermed se slik ut: 11000001.11011001.10110010.00010011. I praksis oppgives vanligvis ikke adressen binært, men desimalt. Den samme adressen vil da se slik ut 193.217.178.19.

Grunnleggende kunnskaper.

Før vi går videre, må vi ha noen begreper på plass:

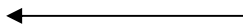
Bit: Ett siffer som er enten 1 eller 0.
Byte: Åtte bit. Kalles også oktett.
Oktett: Åtte bit. Kalles også byte.

Siden adresseformatet på nettet er i binær form, kan det være fornuftig å lære seg hvordan vi kan konvertere en binær adresse til en desimal adresse og omvendt.

Det desimale tallsystemet.

I det desimale tallsystem bruker vi 10 som grunntall. Dette vil si at vi i det **desimale tallsystem øker "verdien" at ett tall med en faktor på 10** for hver plass vi flytter fra høyre mot venstre.

Eksempel: Det desimale tallet **2463**



1000	100	10	1	Faktor.
2	4	6	3	Desimal

$$1000*2 + 100*4 + 10*6 + 1*3 = 2463$$

Skrevet på en annen måte, har vi i dette eksempelet:

$$\begin{array}{r} 3 \text{ enere} \quad 3 * 1 \quad = \quad 0003 \\ + \quad 6 \text{ tiere} \quad 6 * 10 \quad = \quad 0060 \\ + \quad 4 \text{ hundrede} \quad 4 * 100 = \quad 0400 \\ + \quad 2 \text{ tusener} \quad 2 * 1000 = \quad 2000 \\ \hline \end{array}$$

Total sum: 2463 desimalt (To tusener + fire hundrede + seksti tiere + tre enere)

Det binære tallsystemet.

I det **binære tallsystem øker vi verdien med en faktor på 2** for hver plass vi flytter fra høyre mot venstre.

Eksempel: Det binære tallet: 001000101

128	64	<u>32</u>	16	8	<u>4</u>	2	<u>1</u>	Faktor. (Desimal)
0	0	1	0	0	1	0	1	Binært tall

$$0*128 + 0*64 + 1*32 + 0*16 + 0*8 + 1*4 + 0*2 + 1*1 = 37$$

Det binære tallets desimalverdi er: $32 + 4 + 1 = 37$

Det binære eksempelet skrevet på en annen måte:

1 ener	1 * 1 = 1
+ 0 toere	0 * 2 = 0
+ 1 firer	1 * 4 = 4
+ 0 åttene	0 * 8 = 0
+ 0 sekstener	0 * 16 = 0
+ 1 trettitoere	0 * 32 = 32
+ 0 sekstifirere	0 * 64 = 0
+ 0 hundreogtyveåttene	0 * 128 = 0
<u>Total sum</u>	<u>37 desimalt. (En ener + en firer + en trettitoer)</u>

Omgjøring av binære tall til desimale tall og omvendt.

IP-adresser og klasser.

Alle dataenheter på ett nettverk må ha en unik IP-adresse for å kunne kommunisere med hverandre. Som nevnt tidligere, så består en slik adresse av ett 4 bytes "tall". Dette tilsvarer ett 32 bits "ord". Rent teoretisk skulle vi da kunne ha 4294967300 forskjellige kombinasjonsmuligheter av 0-ere og 1-ere. adresser. Altså 4294967300 forskjellige adresser. Slik er det ikke i virkeligheten. I utgangspunktet er det tilgjengelige adresseområdet oppdelt i "IP-klasser" ("adresseklasser"), slik som dette:

IP-klasser:

Klasse A	Adresseområdet 0-127 i det første byte. Brukt i store nettverk.
Klasse B	Adresseområdet 128-191 i det første byte. Brukt i mellomstore nettverk
Klasse C	Adresseområdet 192-223 i det første byte. Brukt i små nettverk

En IP-adresse kan for eksempel se slik ut: 128.16.30.4

Vi har også klasse D og E, men dette er spesialtilfeller som vi ikke skal gå nærmere inn på akkurat nå:

- Klasse D: Adresseområdet 224-239 i første byte. Reserverte adresser. Brukt som multicastadresser. Disse adressene er ikke tilgjengelig som hostadresser.
- Klasse E: Adresseområdet 240-255 i første byte. Reserverte adresser. Brukt i forskning. Disse adressene er ikke tilgjengelige som hostadresser.

Hierarkisk system.

En IP-adresse er ett hierarkisk system, som er delt opp i to deler. Vi vet at en IP-adresse består av et 32-bits ord, som er oppdelt i 4 byte. For eksempel 128.16.30.4. Det hierarkiske systemet deler opp IP-adressen i en nettverksadressedel (N) og en maskinadressedel som benevnes host (H). Viss vi går tilbake til sammenligningen med et brev som sendes med posten, så ser vi jo at også der er adressen oppdelt. På et brev som skal fra Norge til England, så er ikke navnet på adressaten viktig for det norske postvesenet, men til hvilket land brevet skal til. Postvesenet i England sender brevet videre til riktig by, der postkontoret sender brevet videre til riktig adresse og postkasse. Det samme gjelder en IP-adresse. Nummeret på maskinen (host) som datapakken skal til, er ikke viktig i første omgang, men til hvilket nettverk pakken skal til. En Ip-adresse er dermed oppdelt slik:

1. Nettverksadresse. (N, n)
2. Hostadresse. (H, h, maskinadresse, Enhetsadresse, nodeadresse (kjært barn, mange navn))

For å "se" hva som er nettverksadresse og hva som er hostadressen, må vi kunnskap om hvordan det hierarkiske systemet for IP-adresser er bygd opp. Det er her "IP-klasser" kommer inn. En IP-klasse er egentlig ikke annet enn en ett område av det totale adresseringsområdet. Vi skiller mellom klasse A, B og C. Klasse D og E er spesialtilfeller.

Klassene defineres slik:

Klasse A (N.H.H.H)	Nettverk	Host	Host	Host
Klasse B (N.N.H.H)	Nettverk	Nettverk	Host	Host
Klasse C (N.N.N.H)	Nettverk	Nettverk	Nettverk	Host

Men hvordan vet en maskin hva som er **Nettverksdelen** og hva som er Hostdelen? Da TCP/IP ble utviklet ble klassene spesifisert slik:

De første bittene i første oktett:	Klasse
0	A
10	B
110	C

Når vi nå vet hvordan vi skal finne ut av hvilke klasse en adresse tilhører, kan vi dermed også finne ut av hvor mange Nettverk (N) og Hosts (H) hver klasse kan "supportere". I Nettverks-adressedelen må vi ta hensyn til hvor mange bit som "går vekk" for å identifisere hvilken klasse adressen tilhører.

Antall nettverk og nettverkadresseområdet:

Klasse	Nettverksdelen består av (bit)	Antall nettverk:	Nettverksadresseområdet. (desimaltallet i første oktett)
Klasse A (0)	(8 - 1) bit = 7 bit	$2^7 = 128$	0 - 127
Klasse B (10)	(16 - 2) bit = 14 bit	$2^{14} = 16384$	128 - 191
Klasse C (110)	(24 - 3) bit = 22 bit	$2^{22} = 4194304$	192 - 223

Tar vi for eksempel IP-adressen 20.207.40.2, så ser vi at dette er en klasse A-adresse fordi 20 i første oktett er innenfor Nettverksadresseområdet for Klasse A. (0 – 127)

Antall host:

Klasse	Hostdelen består av (bit)	Teoretisk tilgjengelig adresseområdet. Antall Hosts:
Klasse A	24	$2^{24} = 16777216$
Klasse B	16	$2^{16} = 65536$
Klasse C	8	$2^8 = 256$

Tar vi for eksempel IP-adressen 20.207.40.2, så ser vi at dette er en klasse A-adresse fordi 20 i første oktett er innenfor Nettverksadresseområdet for Klasse A. (0 – 127) og at i dette nettverket kan det være 16777216 hosts fordi hostdelen består av 24 bit.

Meget viktig regel:

Når vi nå kjenner til hvordan er IP-adresse er bygd opp og vet hvor mange Nettverk og Noder vi teoretisk kan ha i hver klasse, så er det meget viktig å kjenne til følgende regler:

1. Det er i utgangspunkt ikke tillatt med alle bit "1" eller alle bit "0" i Nettverksdelen av en IP-adresse MEN, viss ruterer kan har mulighet til å konfigureres til å bruke "IP subnet zero", så kan dette likevel la seg gjøre.
2. Det er ikke tillatt å ha alle bit "1" eller alle bit "0" i Hostdelen av en IP-adresse.

Alle bit satt til "1" i hostdelen (H) er avsatt til "broadcastadresse". Broadcastadressen brukes når en host ønsker å sende en melding til alle andre hosts innenfor nettverket.
 Alle bit satt til "0" i hostdelen er avsatt til adressen til nettverket selv. Av det teoretiske tilgjengelige adresseområdet må vi derfor alltid trekke fra 2 adresser for å få det praktisk tilgjengelige adresseområdet.

Eksempel Klasse C (N.N.N.H):

Praktisk tilgjengelig hosts (H) = $2^8 - 2 = 254$ hosts.

Hvordan finne Nettverksadresse (NettverksID), broadcastadresse og praktisk tilgjengelig adresseområdet.?

Klasse A-nettverk.

Nettverksadresse, broadcastadresse og lovlige adresser finnes slik:

Nettverksadresse: Sett alle hostbits (H) til 0. Du får da nettverksadressen. For eksempel for én klasse A-nettverk hvor IP-adressen er 20.138.43.16, (N.H.H.H) så er nettverksadressen 20.0.0.0. (Binært 00010100.00000000.00000000.00000000) Adressen 20 i første byte er innenfor det lovlige området 1-126 som gjelder for klasse A-nettverk.

Broadcastadresse: Sett alle hostbits (H) til 1. For eksempel for én klasse A-nettverk hvor IP-adressen er 20.138.43.16 (N.H.H.H), så er broadcastadressen 20.255.255.255. (Binært 00010100.11111111.11111111.11111111)

Lovlige IP-adresser i én klasse A-nettverk: De lovlige ip-adresser i ett klasse A-nettverk er adressene mellom nettverksadressen og broadcastadressen. I dette tilfelle, klasse A-nettverk, vil de lovlige adressene være 20.0.0.1 til 20.255.255.254 som er $2^{24}-2 = 16777216252$ hostadresser (H)

20.255.255.255	Broadcastadressen (siste adresse i det teoretiske tilgjengelige adresseområdet.) Binært: 00010100.11111111.11111111.11111111
1 - 254	Det praktisk tilgjengelige adresseområdet. N.H.H.H.H $2^{24} - 2 = 16777216252$ (H)

Klasse B-nettverk.

Klasse B-nettverk går fra 128 – 191. Hvis du finner en adresse som for eksempel begynner på 160, da vet du at dette er ett klasse B-nettverk. Klasse B-nettverk bruker de to første byte for å identifisere nettverket (N). De to siste byte identifiserer hosts. (H)

Slik:

Nettverk.Nettverk.Host.Host. (N.N.H.H)

I adressen 160.94.9.3, så er 160.94 identifiseringen av nettverket, og 9.3 identifiseringen av hosts.

Vi kan bruke samme metode som for klasse A-nettverk for å finne nettverksadresse, broadcastadresse og lovlige hosts-adresser. Husk bare at i ett klasse B-nettverk identifiseres nettverket ved de to første byte – og hosts-identifiseringen ved de to siste byte.

Slik:

Nettverksadresse: 160.94.0.0 Alle hostbits (H) er slått av.

Broadcastadresse: 160.93.255.255. Alle hostbits (H) er slått på.

Lovlige IP-adresser i ett klasse B-nettverk: De lovlige ip-adresser i ett klasse B-nettverk er adressene mellom nettverksadressen og broadcastadressen. I dette tilfelle, klasse B-nettverk, vil de lovlige adressene være 160.94.0.1 til 160.255.255.254

Klasse C-nettverk.

Klasse C-nettverk går fra 192 – 223 i første byte. I ett klasse C-nettverk, identifiserer de tre bytes nettverket (N), og den siste byten antall hosts (H).

Slik: Nettverk.Nettverk.Nettverk.Host (N.N.N.H)

Adressen 201.10.10.46 er en klasse C-adresse.

Vi kan bruke samme metode som for klasse A-nettverk for å finne nettverksadresse, broadcastadresse og lovlige hosts-adresser. Husk at i ett klasse C-nettverk identifiseres nettverket ved de tre første byte – og hosts-identifiseringen ved det siste byte.

Slik:

Nettverksadresse: 201.10.10.0 Alle hostbits (H) er slått av.

Broadcastadresse: 201.10.10.255. Alle hostbits (H) er slått på.

Lovlige IP-adresser i ett klasse B-nettverk: De lovlige ip-adresser i ett klasse C-nettverk er adressene mellom nettverksadressen og broadcastadressen. I dette tilfelle, klasse C-nettverk, vil de lovlige adressene være 201.10.10.1 til 201.10.10.254.

Klasse D-nettverk. (Reserverte adresser)

Klasse D-nettverk går fra 224-239 i første byte og er reserverte adresser. Disse adressene kalles multicastadresser. *De kan ikke brukes som hostadresser.* Multicastadresser brukes når en ønsker å sende til en utvalgt gruppe av hosts på ett nettverk.

Klasse E-nettverk. (Reserverte adresser)

Klasse E-nettverk går fra 240 – 255 i første byte og er reserverte adresser. Disse adressene brukes kun i forskningsmiljø.

Nettverksmaske

Men hvordan skal hosts (maskinen) som henger på nettet vite hva som er nettverksadresse (N) og hva som er hostsadresse (H)? Til dette brukes Nettverksmasken eller også kalt subnettmask. IP vil sjekke nettverksmasken for å finne hva som er nettverksadresse (N) og hva som er hostadresse. (H)

Det lar seg ikke gjøre å gi en host en adresse uten å også gi host nettverksmaske -informasjonen.

Nettverksmasken for klasse A, B og C angir du slik:

Klasse A-nettverk: 255.0.0.0
Klasse B-nettverk: 255.255.0.0
Klasse C-nettverk: 255.255.255.0

Default gateway.

Hvis en hosts (H) ikke ligger i samme nettverk som deg selv, så vil systemet sende datapakken til "default gateway", som er en ruter som kan sende datapakker videre ut av nettet.

Adressen for default gateway må du alltid oppgi. Bruker du DHCP, får du den automatisk.

I et windowsnettverk er adressen til "default gateway" vanligvis den første adressen i adresseområdet.
I et linuxnettverk er denne adressen vanligvis den siste adressen i adresseområdet.

Hvordan finner systemet ut av om en IP-adresse finnes på eget nettverk eller ikke?

Systemet finner ut om en IP-adresse er på samme nettverk eller ikke, ved å gjøre følgende logiske operasjon:

1. Kjør en "OG" funksjon på source IP-adresse (Fra IP-adresse) og Nettverksmasken (N)
2. Kjør en "OG" funksjon på destination IP-adresse (Til IP-adresse) og Nettverksmasken. (N)
3. Hvis resultatet på trinn 1 og 2 er helt likt, så er destinationadresse (Til IP-adresse) på samme nettverket som deg selv. Hvis resultatet er ulikt, så er TIL IP-adresse på et annet nettverk enn deg selv. IP-pakken sendes da til "default gateway" på det lokale nettet.

Eksempel.

Din maskin har IP-adresse 159.224.7.129 (Source-adresse, Fra-adresse) med nettverksmaske 255.255.0.0
Du ønsker å sende en datapakke til IP-adresse 159.224.7.84 (Destination-adresse, Til-adresse)

Før vi kan gå i gang med å utføre den logiske "OG"-operasjonen, må vi selvfølgelig først gjøre om de desimale IP-adressene og nettverksmasken til binær notasjon. Hvordan dette gjøres, har vi sett på tidligere.

Source IP-adresse (Fra IP-adresse) 159.224.7.129 = 10011111.11100000.00000111.10000001
Destination IP-adresse (Til-adresse) 159.224.7.84 = 10011111.11100000.00000111.01010100
Nettverksmasken : 255.255.255 = 11111111.11111111.00000000.00000000

Nå starter vi den logiske operasjonen:

Punkt 1.

Fra-IPadresse:	10011111.11100000.00000111.10000001
+ Nettverksmasken	11111111.11111111.00000000.00000000
<u>Resultat A: :</u>	<u>10011111.11100000.00000000.00000000</u>

Punkt 2.

Til-IPadresse:	10011111.11100000.00000111.01010100
+ Nettverksmasken	11111111.11111111.00000000.00000000
<u>Resultat B: :</u>	<u>10011111.11100000.00000000.00000000</u>

Vi ser at resultat A og resultat B er helt likt. Dette betyr at Fra IP-adresse (Source-IPadresse) og Til-IP-adresse (Destination-adresse) er på samme nettverk. Datapakken sendes dermed ikke til default gateway.
