

Korte uitleg: IP adressen

Hoewel we in het dagelijks gebruik van een computer, verbonden met een TCP/IP netwerk, meestal geen IP adressen gebruiken, maar de host-/domeinnamen (dus niet **130.57.66.6**, maar **forums.opensuse.org**) is enige kennis over IP adressen handig, vooral bij netwerk problemen.

Overall waar een netwerk wordt gebruikt is een unieke adressering van de bestemmingen in dat netwerk nodig, anders weten we niet waar de zaken heengestuurd moeten worden. Er mag dus maar één postadres bestaan:

Dam 1
1012 JS Amsterdam
Nederland

Dit zou je ook kort kunnen schrijven: NL.1012JS.1. Van meest significant (land) naar minst significant (brievenbus).

Klassieke IP, nu IPv4

IP adressen

In een TCP/IP netwerk (en dat geldt dus zowel voor je thuisnetwerk/Local Area Network/LAN al voor het Internet) is dat niet anders. Destijds is besloten om voor de adressering 4 bytes (32 bits) te gebruiken. Dat levert dus $256 \times 256 \times 256 \times 256$ (meer dan 4 miljard) mogelijke adressen op. Dat zou genoeg moeten zijn voor altijd en altijd. Niet dus. Vandaar IPv6. Maar eerst verder over IPv4 (dat toen natuurlijk nog niet zo werd genoemd). We schrijven zo'n adres als decimale notatie van de waardes van de vier bytes gescheiden door punten. Dus bijvoorbeeld **130.57.66.6**. Het eerste (meest significante) byte heeft de waarde 130, enz. De waarde van een byte kan uiteraard alle waardes van 0 - 255 zijn.

Netwerk/computer hiërarchie

Het is natuurlijk niet handig om die adressen willekeurig uit te delen. Net als bij de post is het beter als Dam 2, Amsterdam, Nederland vlak bij Dam 1, Amsterdam, Nederland ligt. Een IP adres bestaat daarom uit twee delen: het netwerk gedeelte and het computer/host gedeelte. Het netwerk gedeelte bepaalt dan welke richting een pakket moet worden gestuurd om bij het juiste netwerk (LAN) aan te komen. In het LAN wordt dan het host gedeelte gebruikt om de individuele computer te vinden.

Maar waar leggen we de grens tussen de twee gedeeltes? Oorspronkelijk werden alle mogelijke adressen daarvoor in drie groepen verdeeld. Klas A, klas B en klas C netwerken. Bij een klas A netwerk is het eerste byte het netwerk gedeelte. Dat levert 128 klas A netwerken op met ieder $16.777.216$ host mogelijkheden. Die netwerken waren gedacht voor zeer grote

organisaties. Klas B netwerken hebben een netwerk gedeelte van 2 bytes: 16.384 netwerken, ieder met 65.536 hosts. Klas C netwerken hebben de grens na 3 bytes en dat levert 2.097.152 netwerken met ieder 256 mogelijkheden. Dit was veel te rigide. Veel bedrijven vroegen een klas A of klas B netwerk aan en gebruikten daarvan maar een gedeelte. Tegenwoordig wordt er bij een adres gewoon gezegd hoeveel bits het netwerk gedeelte is. De rest is dan de host. De notatie is **10.0.0.54/24**. Dit betekent dat de eerste 24 bits (3 bytes: **10.0.0**) het netwerk gedeelte is en de rest (**54**) is het host adres binnen dat netwerk. Overigens is dit niet beperkt tot de grens tussen twee bytes, netwerken als **172.16.0.0/12** zijn ook mogelijk.

Gereserveerde adressen in een netwerk

Niet alle mogelijkheden uit het hostgedeelte zijn bruikbaar als echt adres. Het laagste adres (alle bits van het hostgedeelte op **0**) is de aanduiding van het netwerk zelf. Dus **10.0.0.0/24** betekent: het netwerk **10.0.0**. Het hoogste adres wordt gebruikt als "broadcast", een manier om alle adressen in het LAN tegelijk een pakket te sturen. Dus **10.0.0.255/24** is het broadcast adres van het netwerk **10.0.0.0/24**. Er blijven in ons voorbeeld dus 254 adressen over om computers aan te sluiten (1 - 254).

Publieke en privé adresgroepen

Toen men oplossingen begon te zoeken voor het steeds dreigender opraken van IP adressen, realiseerde men zich dat veel adressen helemaal nooit met het Internet willen praten. Bijvoorbeeld machines in een fabriek, die alleen met elkaar moeten kunnen werken. Er zijn daarvoor een aantal adresblokken gereserveerd. Deze adressen worden door de routers in het Internet niet behandeld/gerouteerd. Het zijn: **10.0.0.0/8**, **172.16.0.0/12** en **192.168.0.0/16**. De laatste doet je misschien denken aan het netwerk bij je thuis. Dat is vaak **192.168.1.0/24** (een zogenaamd sub-net van **192.168.0.0/16**), met **192.168.1.1** als je modem/router en de andere systemen in dat LAN hebben dan dus ook adressen **192.168.1.n**.

Dus het verschil tussen publieke en privé (public/private) netwerk adressen is dat de private adressen niet over het Internet kunnen werken en dat iedereen dus die adressen thuis kan gebruiken zonder dat ze elkaar storen. Maar hoe kan het dan dan je systeem thuis met een privé adres toch met systemen op het Internet kan praten? Dat wordt opgelost door de router, die immers aan de "buitenkant" een publiek adres heeft. En dit gebeurt met Network Address Translation (NAT) en dat behandelen we in *Korte uitleg: "Network Address Translation" en "Port Forwarding" op je router*.

Local host

Behalve de netwerkkaarten (kabel of wifi) is er ook een niet bestaand apparaat **lo**. Ook wel local loop genoemd. Hierover kan het systeem TCP/IP netwerkverkeer doen met zichzelf. Er is een adres voor gereserveerd: **127.0.0.1/8**. Een client programma op een systeem (bijvoorbeeld een web-browser) kan nu zonder iets bijzonders te doen met een server

prorogramma (bijv. de Apache web-server) op hetzelfde systeem praten. Het TCP/IP pakketje gaat niet echt het netwerk in, maar wordt door de TCP/IP software in de kernel direct doorgegeven van de afzender naar de ontvanger.

Andere adressen uit de **127.0.0.0/8** groep worden soms ook voor speciale zaken gebruikt. Het is meestal een truukje, zoals **127.127.0.0/16** in de NTP (Network Time Protocol) configuratie.

Link-local adress

Ik heb dit maar onvertaald gelaten. Er is een netwerk gereserveerd voor dit soort adressen: **169.254.0.0/16**. Een systeem kan zo'n adres gebruiken als er geen ander adres geconfigureerd is (of kan worden). Het gaat vaak om kleine apparaten (bijv. printers). Zij doen dit met proberen. Ze kiezen een willekeurig adres uit deze groep en kijken of een ander die al gebruikt, enz. Zulke adressen worden door een router niet behandeld/gerouteerd. Dus niet alleen niet naar het Internet, maar ook niet naar andere LANs die direct aan dit LAN zijn verbonden.

Praktijk

Een simpel voorbeeld op een systeem met een local loop en een ethernet kabel:

```
boven:~ # ip -f inet address
1: lo: mtu 65536 qdisc noqueue state UNKNOWN
   inet 127.0.0.1/8 brd 127.255.255.255 scope host lo
      valid_lft forever preferred_lft forever
2: enpls8: mtu 1500 qdisc pfifo_fast state UP qlen 1000
   inet 10.0.0.154/24 brd 10.0.0.255 scope global enpls8
      valid_lft forever preferred_lft forever
boven:~ #
```

IPv6

En toen waren de IP adressen op. Tijd voor iets nieuws. De "oude" adressering heet nu IPv4 en de nieuwe IPv6. IPv6 adressen zijn vier maal zo groot als IPv4 adressen (16 bytes, dus 128 bits). De notatie is anders, hexadecimaal in groepjes van 2 bytes (16 bits) en dus vier hexadecimale cijfers, gescheiden door **:**. Dat zijn dus 8 van die velden. Er zijn allerlei regels over hoe je velden kunt weglaten (zo kun je **::** lezen als een rij **0000** velden achter elkaar en wel zoveel als nodig om het geheel op 8 velden te brengen).

Publieke en privé adresgroepen

Er bestaan ook privé adressen, hier "unique local addresses" (ULA) genoemd: **fc00::/7**.

Local host

Het local host adres is `::1/128`.

Link-local address

En de link-local adressen beginnen met: `fe80:`. In tegenstelling met IPv4 worden link-local adressen altijd aan een netwerkkaart (NIC) geconfigureerd. Een apparaat is kan dus altijd meteen binnen een LAN functioneren, zelfs als de NIC alleen maar is UP gezet zonder verdere configuratie.

Het host gedeelte van de adressen word vaak automatisch toegekend op basis van het MAC adres (een unieke indentificatie) van de netwerkkaart. Er worden vaak meer adressen toegekend voor verschillende doeleinden.

Praktijk

Hetzelfde systeem als boven, maar nu de IPv6 adressen:

```
boven:~ # ip -f inet6 address
1: lo: mtu 65536
   inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: enpls8: mtu 1500 qlen 1000
   inet6 2001:980:91a0:1:bd59:db78:98bd:314b/64 scope global temporary dynamic
       valid_lft 6435sec preferred_lft 3356sec
   inet6 2001:980:91a0:1:21b:fcff:fe7f:c1ef/64 scope global dynamic
       valid_lft 6435sec preferred_lft 2835sec
   inet6 fe80::21b:fcff:fe7f:c1ef/64 scope link
       valid_lft forever preferred_lft forever
boven:~ #
```