

Korte uitleg: Domain Name System (DNS)

Het telefoonboek van het internet.

In *Korte uitleg: IP adressen* hebben we gezien dat computers een IP adres hebben zoals **130.57.66.6**. Voor computers geen probleem, maar de meeste mensen onthouden dat niet zo makkelijk. Wij werken liever met namen. In het begin waren computers alleen binnen bedrijven aan een netwerk gekoppeld en was het voldoende om de relatie tussen de computernaam en het adresnummer lokaal bij te houden. In een Unix/Linux computer gaat dat met het configuratiebestand `/etc/hosts`. Uiteraard waren systeembeheerders zo slim om ergens een moederbestand te hebben dat bijv. met een scriptje met `rsync` commando zeer snel naar alle systemen binnen het bedrijf verspreid kon worden.

Een stap verder werd gedaan toen SUN Microsystems het op afstand aankoppelen (mounten) van file system mogelijk maakte met NFS. Het ging vergezeld van NIS, dat een centrale administratie van diversie configuratie zoals het `/etc/hosts` bestand (maar ook gebruikers/groepen configuraties) implementeerde.

Voor het snel groeiende Internet was iets beters nodig.

Host/domain names

Ten eerste moesten de namen van computers gestandaardiseerd worden (de IP adressen waren dat al). Gekozen werd voor een hiërarchische aanpak, waarbij het idee is dat een computer (host) met een aantal verwante computers (bijv. in hetzelfde bedrijf) een domein (domain) vormt. Een aantal domeinen vormt dan weer een groter domein en zo door tot het root domein (alweer root en deze root heeft zelfs geen naam) ieder domein heeft dus een aantal sub-domeinen en/of computers. De schrijfwijze is (in tegenstelling tot IP adressen) van minst significant (host) naar meest significant (top-level domain). De velden worden gescheiden door punten. Alleen letters, cijfers en het - teken uit de ASCII lijst mogen worden gebruikt. Boven- en onderkast letters ("hoofdletters" en "klein letters") zijn gelijk aan elkaar. Dus `forums.opensuse.org` is hetzelfde als `forums.openSUSE.org`. Er zijn beperkingen o.a. in lengte van deze namen.

Om gebruik van andere Unicode tekens mogelijk te maken is er een methode om die om te zetten naar de bovengenoemde eisen. De namen zelf bevatten dus geen "rare" tekens, maar zijn min of meer onleesbaar. Een goede applicatie (browser) zet ze weer om in bijv. Arabisch in de adresbalk. Een voorbeeld (uit de Wikipedia pagina):

De gewenste naam is: `bücher.ch`

De URL wordt: `xn--bcher-kva.ch`

Deze laatste wordt dus "intern" gebruikt, maar de gebruiker tikt in de adresbalk (of ziet in de adresbalk) de eerste.

Let op! Je ziet dus de Unicode tekens en in veel fonts lijken sommige Unicode tekens sterk op elkaar. Zo lijkt de Cyrillische **р** (U+0440) heel erg op de Latijnse **p** (U+0070, ASCII:x'70'=112). Je kunt dus denken dat je op wikipedia.org zit terwijl je in werkelijkheid op xn--wkd-8cdx9d7hbd.org bent aangeland.

Gedistribueerd

Het wereldwijd bijhouden van domein- en computernamen kan niet centraal gedaan worden. Daarom wordt deze taak verdeeld. ICANN registreert de "top level domains", dus alle meest rechtse domeinen zoals .com, .net, .org en ook die voor alle landen zoals .nl voor Nederland. Het beheer van die domeinen zelf wordt uitbesteed aan aparte organisaties. Voor Nederland is dat de Stichting Internet Domeinregistratie Nederland (SIDN) en voor België DNS Belgium.

De regels binnen een domein worden door de beheerder bepaald. Zo worden binnen .nl domeinen direct op het hoogste mogelijke niveau verstrekt (bijv. ah.nl en xs4all.nl) terwijl in het Verenigd Koninkrijk eerst een laag voor verdere verdeling is gemaakt met bijv. .org.uk, .co.uk en een bedrijf wordt dan bijv. marksandspencer.co.uk.

Een bedrijf dat zijn domeinnaam beheert kan zelf eventueel sub-domeinen bij zichzelf registreren.

DNS is een gedistribueerde database

Dit geheel vormt de gedistribueerde database die DNS eigenlijk is. De gegevens van ieder (sub-)domein zijn de verantwoordelijkheid van de bezitters (uitbesteden mag). Er is dus een DNS server die de gegevens van ah.nl bevat en een die de gegevens van xs4all.nl bevat en een die de gegevens van .nl bevat. Die laatste weet ook welk systeem de ah gegevens en welk systeem de xs4all gegevens bevat.

Ruwweg gaat het dus zo:

Als vanuit mijn systeem aan de DNS server (ook "name server") van XS4ALL wordt gevraagd om het IP adres van www.ah.nl, vraagt die XS4ALL server een stapje hogerop, bij de server van .nl. En die verwijst weer naar de name server van AH en die weet het antwoord (hopelijk). Overigens, om niet steeds weer die hele weg te bewandelen, wordt op veel plaatsen de vragen/antwoorden bewaard gedurende een bepaalde tijd (caching).

Belangrijk in een systeem is dus de verwijzing naar een DNS server om je vragen naar toe te sturen. Het maakt niet veel uit welke, maar als die " dicht bij" is scheelt dat netwerk verkeer en dus tijd. Die DNS server is geconfigureerd in [/etc/resolv.conf](http://etc/resolv.conf). Bij mij staat er dit in (commentaar regels verwijderd):

```
henk@boven:~> grep -v '^#' /etc/resolv.conf
search xs4all.nl
nameserver 194.109.6.66
```

```
nameserver 194.109.9.99
nameserver 194.109.104.104
henk@boven:~>
```

Daar zie je drie DNS servers van mijn provider. Uiteraard zijn dat IP adressen en geen namen. Bij veel thuiscomputers zul je daar overigens het adres van je router vinden. Je router dient dan als DNS doorgeefluik en je krijgt dat adres geconfigureerd door DHCP te gebruiken.

Als DNS het "niet doet", controleer dan `/etc/resolv.conf`. Als daar niets instaat kun je het adres van je router proberen. Je kunt ook een adres dat je ISP ter beschikking stelt proberen. Je kunt altijd

```
nameserver 8.8.8.8
```

proberen, dat is een nameserver die Google ter beschikking stelt.

Zelf proberen

```
henk@boven:~> host forums.opensuse.org
forums.opensuse.org has address 130.57.66.6
forums.opensuse.org has IPv6 address 2600:806:310::100
forums.opensuse.org mail is handled by 42 mx2.suse.de.
henk@boven:~>
```

Je ziet het IPv4 adres, het IPv6 adres en ook een ander gegeven: als er mail wordt gestuurd `@forums.opensuse.org`, wordt dat doorgestuurd naar `mx2.suse.de`.

```
henk@boven:~> host 130.57.66.6
6.66.57.130.in-addr.arpa domain name pointer forums.opensuse.org.
6.66.57.130.in-addr.arpa domain name pointer www.opensuse.org.
henk@boven:~>
```

Omgekeerd zoeken gaat ook. Je ziet hier dat er zelfs twee namen zijn voor dit IP adres. Eén is een zogenaamde alias. Interessant is hoe een IP adres, dat hiërarchiesch van links naar rechts is, wordt vervangen door een constructie die omgekeerd is en dan als sub-domein van `.in-addr.arpa` wordt gebruikt. Dit om de hele manier van werken voor een IP adres gelijk te trekken aan die voor een host-domeinnaam.

Een ander tool (oud, maar bestaat nog steeds):

```
henk@boven:~> nslookup 130.57.66.6
6.66.57.130.in-addr.arpa domain name pointer forums.opensuse.org.
6.66.57.130.in-addr.arpa domain name pointer www.opensuse.org.
henk@boven:~> nslookup forums.opensuse.org
Server:          194.109.6.66
Address:         194.109.6.66#53
```

```
Non-authoritative answer:  
Name:   forums.opensuse.org  
Address: 130.57.66.6
```

```
henk@boven:~>
```

Dit geeft weer wat andere extra's. Je ziet hier welke DNS server is gebruikt. Handig als wilt weten of je configuratie in `/etc/resolv.conf` wel werkt.

Toch handig: `/etc/hosts`

Intussen hoef je voor het configureren van een lokaal telefoonboek niet direct zelf een DNS server te bouwen. Het bestand `/etc/hosts` werkt nog steeds aanvullend aan DNS. Je kunt dat via YaST > Netwerkservices > Hostnamen doen, of het bestand met een editor wijzigen (moeilijk is het niet, maar lees eerst de `man` pagina!). Handig als je een paar systemen in huis hebt. Uiteraard zorg je ervoor dat de `/etc/hosts` uit de verschillende systemen elkaar niet tegenspreken.

Overigens wordt in `/etc/nsswitch.conf` bepaald in welke volgorde gezocht wordt. De default:

```
henk@boven:~> grep hosts /etc/nsswitch.conf  
hosts:          files mdns_minimal [NOTFOUND=return] dns  
henk@boven:~>
```

betekent dat eerst in `files` (`/etc/hosts`) gekeken wordt en als daar geen oplossing wordt gevonden wordt `dns` (DNS) gebruikt (vergeet even het stukje `mdns_minimal [NOTFOUND=return]`, het is in dit voorbeeld niet van belang). Het is beter om dit niet te veranderen.