Handleiding
Debian GNU/Linux 2.2

Voor IBM-compatible PC's

Achtergronden, Installatie en Gebruik

debian

J.A. Bezemer

Handleiding Debian GNU/Linux 2.2
Versie 22-09-2000

Elektronische versies op http://panic.et.tudelft.nl/debian/handleiding

Voorwoord

Linux is niet moeilijk.

Linux is anders.

Linux is beter.

Heb ik je nog niet overtuigd? Dat was ook niet de bedoeling. Je moet het zelf proberen. Deze handleiding zal je een flink eind op weg helpen.

Linux is niet iets voor mensen die hun computer alleen gebruiken om zo af en toe eens een briefje te typen. De gemiddelde Linux-gebruiker is meer te typeren als een avonturier, iemand die op zoek is naar iets wat hij (of zij natuurlijk) nog niet kent en er geen probleem mee heeft eens iets anders te doen dan in een handleiding staat aangegeven. Ik ga ervan uit dat je al iets van DOS en/of Windows weet; ik leg bijvoorbeeld niet uit wat bestanden en directory's zijn.

Deze handleiding is totaal niet compleet. Ik zal je regelmatig doorverwijzen naar andere informatiebronnen. Je zult zelf actief met Linux aan de slag moeten gaan, want door te doen leer je het meeste. En zeker in het begin is "heel veel lezen" het motto.

De onderdelen in deze handleiding zijn cumulatief opgebouwd. Dat wil zeggen dat je (in ieder geval de eerste keer) bij het begin moet beginnen en tot het einde doorwerken. Het is de bedoeling dat je deze handleiding daarna in de kast zet en er niet meer naar omkijkt, simpelweg omdat de dingen zo simpel zijn dat je na een paar keer oefenen alles uit je hoofd weet.

De enige uitzondering op de regel "alles lezen" is als je gaat werken op een computer waarop Debian al geïnstalleerd is. In dat geval lees je hoofdstukken 1, 2, 4, 6.3 en 6.4, en verder de gedeelten uit hoofdstuk 7 tot 14 die relevant zijn. Als je systeembeheerder gaat worden, lees je ook hoofdstuk 5 op de juiste plaats in de reeks.

Om ruimte te besparen is er geen index; als je naar iets specifieks op zoek bent, kun je met de inhoudsopgave gemakkelijk de juiste weg vinden. Bovendien kun je zoeken in de 'elektronische' versie, wat meestal betere resultaten geeft dan een index.

Versie '2.2' van deze handleiding is niet alleen qua layout veranderd ten opzichte van versie '2.1', maar ook in de inhoud zijn verschillen merkbaar. De eerste installatie is vereenvoudigd, maar ook installeren van extra software gebeurt nu aanzienlijk efficiënter. Verder is er in deze versie veel meer aandacht voor toepassingen en software van allerlei aard, die laten zien dat Debian GNU/Linux een enorm veelzijdig systeem is waarin je al je dagelijkse bezigheden met plezier kunt uitvoeren.

Ik wil mijn broer Maarten bedanken voor het corrigeren van een concept-versie en vele lezers voor het geven van commentaar en het stellen van de nodige vragen.

Papendrecht, september 2000 Anne Bezemer

Inhoud

1. Inleiding

Er is iets grondig aan het veranderen in de software-wereld. Waar vroeger uitsluitend grote commerciële instanties de totale markt in handen hadden, is er nu een andere ontwikkeling zichtbaar. Steeds meer mensen en bedrijven willen af van monopolisten en gebruiken 'vrije' software. 'Vrije' software mag je gebruiken zonder ervoor te betalen. Je krijgt zelfs de broncode erbij, die je naar believen mag aanpassen en verder distribueren.

Linux is ongetwijfeld het grootste vrije-software project tot nu toe. Een compleet, gratis en vooral goed operating system. Origineel voor PC's vanaf de '386, nu ook voor Alpha's, Sparcs, Amiga's, (Power)Macs, NetWinders, PalmPilots en diverse andere. Gemaakt door honderden vrijwilligers, stuk voor stuk software-experts.

Debian GNU/Linux biedt naast het complete Linux operating system een enorm aantal programma's voor elke denkbare toepassing -- allemaal vrije software. Alle programma's zijn ondergebracht in uniforme pakketjes, die gemakkelijk geïnstalleerd en verwijderd kunnen worden. Debian wordt net als Linux verzorgd door honderden vrijwilligers.

Deze handleiding geeft een complete introductie in Debian GNU/Linux. Als eerste wordt de noodzakelijke basiskennis over onder andere Linux behandeld. Dan volgt een uitvoerige beschrijving van de installatie van Debian. Tenslotte wordt de 'Linux-newbie' op een zelfstudie-achtige manier ingeleid in de basis-bezigheden op een Debian systeem.

Debian GNU/Linux draait op elke IBM-compatible PC, vanaf de 80386. Er is minimaal 4 MB intern geheugen nodig en 65 MB harde-schijf-ruimte; maar let op: tijdens de installatie moet er tenminste 12 MB intern geheugen beschikbaar zijn (wanneer nodig kun je misschien ergens extra geheugen lenen, of de harde schijf tijdelijk in een andere computer hangen). Voor tekstmode kan elke videokaart/monitor gebruikt worden; voor grafische mode is minimaal een VGA-kaart en -monitor nodig. Aanbevolen voor "lekker" werken: minstens Pentium (of kloon), minstens 16 MB werkgeheugen en minstens 500 MB harde-schijf-ruimte. Maar als je niet al te veel eisen stelt, werkt een '486 met 8 MB ook nog heel aardig. Als je wilt 'upgraden', is het werkgeheugen het eerste dat aandacht verdient -- uitbreiding tot zo'n 32 MB geeft meer snelheidswinst dan je met een snellere processor zou krijgen. Dat komt, omdat Linux al het niet-gebruikte werkgeheugen als disk-cache reserveert, wat schijftoegang enorm kan versnellen.

Iets over de typografie: als er tekst **`op deze manier`** is aangegeven, gaat het om een programma-naam of een commando wat je kunt intypen. Om papier (en dus kosten) te besparen, staan de commando's niet op aparte regels, maar in de lopende tekst.

```
Tekst die in een bestand staat
(of in een bestand
gezet moet worden),
wordt op deze manier aangegeven.
```

Dan nog iets over de uitspraak van Debian. Die is namelijk "dèbbiejun" en niet "díébiejun" zoals het op het eerste gezicht lijkt. De naam Debian is een samentrekking van Debra en Ian Murdock, die het Debian-project zijn gestart.

Tenslotte nog dit: lees deze handleiding van voor tot achter door, en **sla niets over**. Als je zomaar hier en daar wat leest, krijg je gegarandeerd een systeem dat maar half werkt en moet je dus opnieuw beginnen. Sommige stukjes zijn aan de lastige kant, en die zul je twee of meer keer moeten lezen. In het algemeen: lees eerst een hele paragraaf door, voordat je gaat doen wat erin staat. Als je hulp nodig hebt, kijk dan in paragraaf 4.21.

2. Achtergronden en basiskennis van Linux en Debian

Om verstandig te kunnen omgaan met een willekeurig ding, is het noodzakelijk er het één en ander vanaf te weten. Dat geldt zeker bij de installatie en het gebruik van Debian GNU/Linux (kortweg 'Debian'). Ook als je (bijvoorbeeld op het Internet) een vraag stelt of deelneemt aan een discussie, is het beter dat je niet compleet 'ongeletterd' voor de dag komt. Vandaar dat ik hier wat basiskennis wil aanreiken. Omdat deze tekst niet alleen is bedoeld voor hoogbegaafden, zal ik van onderaf beginnen. (Dat wil niet zeggen dat geleerde heren en dames het onderstaande kunnen overslaan -- correcties en aanvullingen zijn altijd welkom!)

2.1 De computer

Zoals gezegd, beginnen we nogal heel erg 'basic'. Noodzakelijk voor beginnende computeraars en ook nuttig voor gevorderden, omdat terloops een aantal (ongebruikelijke) Engelse termen aan de orde komen. Zoals je misschien al wel door hebt, is Linux (en daarmee Debian) vrijwel compleet Engels en een aantal vaktermen kunnen je raadselachtig voorkomen. Een beetje Engels behoort ook bij de basiskennis en aangezien Microsoft alles maar naar 'goed Nederlands' vertaalt, kan ik er niet vanuit gaan dat die kennis aanwezig is.

Het gaat er niet om, dat je alle termen begrijpt, het is meer dat je weet in welke richting je moet zoeken als je een term leest. Kom je er zelf niet uit, dan kun je bijvoorbeeld terecht in de nieuwsgroep nl.comp.hardware (zie ook paragraaf 4.21).

CPU

De *CPU*, afkorting van *Central Processing Unit* (NL: *Centrale Verwerkings Eenheid*, *CVE*), ook *Processor*, is het hart van de computer. Door deze grote computerchip worden de programma's uitgevoerd. Processors hebben een bepaald merk en type, zoals *Intel 80486* ("een 486"), *Intel Pentium* (of *80586*) of *AMD K6*. De klokfrequentie bepaalt de snelheid van de computer, bijvoorbeeld 233 MHz (megahertz, miljoen cycli per seconde).

Main Board

De CPU is gemonteerd op het *Main Board* (afgekort: *MB*, of *MoBo*, NL: *Moederbord* of ook *Moederkaart*). Dat is een grote plaat (printplaat, 'PCB') die allerlei chips, geheugen en andere apparaten met de CPU verbindt.

Werkgeheugen

Het programma dat wordt uitgevoerd en de gegevens die het nodig heeft, staan in het (*werk-*)*geheugen* of *Main Memory*, ook aangeduid met *RAM* (*Random Access Memory*, willekeurig toegankelijk geheugen, lees- en schrijfbaar). Tegenwoordig is 32 MB (megabyte, 'miljoen' *bytes* of geheugenplaatsen; 1 MB = 1024 x 1024 = 1048576 bytes) zo'n beetje standaard. Termen die met de uitvoering van de geheugenchips te maken hebben zijn bijvoorbeeld *SIMM*, *DIMM*, *SDRAM*, *EDO RAM*.

Insteekkaarten

Het Main Board biedt plaats aan diverse insteekkaarten, zoals een videokaart, multi-I/O kaart, geluidskaart, netwerkkaart en SCSI kaart. Er is een trend gaande om steeds meer van die uitbreidingskaarten in het Main Board te integreren.

Videokaart

De *videokaart* of *Video Card* zorgt ervoor dat de *monitor* de juiste signalen krijgt, zodat er bijvoorbeeld letters of plaatjes op het scherm verschijnen. In de loop van de tijd zijn diverse standaarden ontwikkeld, die steeds meer konden, zoals *Hercules*, *CGA*, *EGA*, *VGA* en *SVGA* (*Super VGA*). In de grafische mode moet voor elk beeldpunt (*pixel*) worden onthouden welke kleur het heeft. Dat kost best wat geheugen, daarom heeft elke nieuwe videokaart minimaal 1 MB geheugen (*Video RAM*). De videokaart en monitor bepalen samen welke *resoluties* mogelijk zijn, bijvoorbeeld 640 bij 480 beeldpunten of 1280 bij 1024. De hoeveelheid geheugen van de videokaart bepaalt hoeveel kleuren er per beeldpunt mogelijk zijn, bij een gegeven resolutie. Videokaarten die op het Main Board zijn geïntegreerd, snoepen vaak 1 tot 2 MB af van het werkgeheugen.

Multi-I/O Controller

De *Multi-I/O Controller* regelt een heleboel *I/O* (*Input* en *Output*, invoer en uitvoer). Zoals daar zijn: *harddisk* (*harde schijf*), *floppy disk drive* (*diskettes*, "schijfjes"), een *parallelle poort* (*Parallel Port*) voor aansturing van de printer, en meestal twee *seriële poorten* (*Serial Ports*) voor bijvoorbeeld een *muis* en een *modem*. Tegenwoordig is de I/O Controller meestal op het Main Board geïntegreerd.

Harddisk

Op de *Harddisk* (NL: *harde schijf*) staan programma-bestanden en gegevens-bestanden. Er zijn twee typen (lees: aansluitingsvormen), *IDE* en *SCSI*. IDE's zijn het goedkoopst en kunnen door elke Multi-I/O kaart (eventueel op het Main Board) worden aangestuurd; SCSI versies zijn soms sneller en hebben een aparte *SCSI Controller* (insteekkaart) nodig. Harde schijven van vele GB's (gigabytes, 'miljard' geheugenplaatsen; 1 GB = 1024 MB = 1073741824 bytes) zijn leverbaar.

Andere apparaten

Zowel binnen als buiten de computer'kast' zijn nog een aantal andere apparaten. Te denken valt aan een *geluidskaart* (*sound card*), een *netwerkkaart* (*network adapter*), een *modem*, de *CDROM-speler* (*CDROM drive*), het *toetsenbord* (*keyboard*), de *muis* (*mouse*), de *monitor* en de *printer*. Een aantal hiervan zal in het vervolg nog ter sprake komen.

2.2 Programma's op de computer, het operating system, de kernel

Op een computer draaien *programma's* (dat is de bedoeling tenminste). Een programma is niets meer dan een hele lange rij van allemaal kleine opdrachtjes, zoals 'verhoog de inhoud van geheugenplaats 6314 met 3', 'kijk of het groter is dan nul', 'spring 41 opdrachten terug'. Al die duizenden kleine opdrachtjes bij elkaar vormen bijvoorbeeld een tekstverwerker.

Programma's staan meestal in (programma-)bestanden op de harde schijf. Maar om ze uit te kunnen voeren, moeten ze in het werkgeheugen staan. Op de één of andere manier moet het programma dus van de harde schijf naar het werkgeheugen worden verplaatst. Dat noemen we het *laden* (*loading*) van het programma.

Het laden en starten van bijvoorbeeld een tekstverwerker wordt geregeld door een *operating system*. Dat is een collectie van programma's die op zichzelf niet zo nuttig is, maar wel onmisbaar. Het *operating system* laadt een programma, wijst er voldoende werkgeheugen aan toe voor gegevensopslag, geeft programma's toegang tot bijvoorbeeld documenten op de harde schijf, en ruimt alles netjes op als het programma afgelopen is. Het levert daartoe een hele reeks van programma-delen, of *routines*, die door andere programma's kunnen worden gebruikt (aangeroepen). Bijvoorbeeld 'open bestand XYZ', 'lees het volgende blok gegevens uit dat bestand', 'laad en start nu programma ABC', maar ook 'zet een j op die-en-die plaats op het scherm'. Deze routines maken deel uit van de *kern* van het operating system, in het Engels: *kernel*.

Omdat het operating system natuurlijk niet weet welk programma je wilt gaan gebruiken, wordt als eerste een standaard-programma gestart, waarin je kunt aangeven welk programma je wilt starten. Zo'n programma presenteert bijvoorbeeld een *prompt* waarachter je de naam van het te starten programma(-bestand) moet intypen. Dat eerste 'doorverwijs-programma' noemen we een *shell*. De shell hoort bij het operating system, maar niet bij de kernel, het is een apart programma. Mogelijkheden om bijvoorbeeld bestanden en directory's te bekijken, kopiëren en verplaatsen horen ook bij het operating system; ze kunnen als interne commando's in de shell beschikbaar zijn, of als aparte programma's.

Laten we eens een bekend voorbeeld analyseren, zodat we een beter idee krijgen waar we het over hebben.

Het bekendste operating system is ongetwijfeld MSDOS (Microsoft Disk operating system). De kernel hiervan bestaat uit de bestanden **`MSDOS.SYS`** en **`IO.SYS`**. Deze bestanden worden als eerste geladen als de computer wordt aangezet (niet helemaal waar, eerst de bootsector nog -- daar gaan we het nog over hebben). Als de MSDOS kernel is gestart, wordt eerst **`CONFIG.SYS`** gelezen, vervolgens wordt de shell **`COMMAND.COM`** gestart, die op zijn beurt de opdrachten uit **`AUTOEXEC.BAT`** gaat uitvoeren. Als dat alles achter de rug is, komt de DOS-prompt op het scherm en kunnen er programma's worden gestart. **`COMMAND.COM`** is trouwens niet de enige shell voor MSDOS, Norton heeft bijvoorbeeld **`NDOS.EXE`**. Het commando '**`COPY A B`**' is in **`COMMAND.COM`** ingebakken; **`XCOPY`**, dat ietsje uitgebreider is, is een apart programma. Beiden behoren tot het operating system.

In DOS kan maar één programma tegelijk draaien: zolang de tekstverwerker actief is, kun je niets meer met **`COMMAND.COM`** (of een ander programma) doen. In Windows-varianten draaien een aantal programma's tegelijkertijd, dat heet *multitasking*, letterlijk 'meerdere taken'. Je kunt die programma's 'door elkaar' gebruiken, zonder ze steeds af te hoeven sluiten.

2.3 Het operating system met de naam Linux

Het operating system (vaak afgekort tot *OS*) waar we het in het vervolg over gaan hebben, heet *Linux*. Hoewel, dat is niet helemaal juist. De *kernel* van het operating system heet Linux, maar in de volksmond wordt dat vaak door elkaar gehaald. De officiële naam is GNU/Linux. Dat komt omdat praktisch alle *utility* programma's ('gereedschap') voor bijvoorbeeld kopiëren en verplaatsen, maar ook de veel gebruikte C en C++ compiler, van het GNU project komen. De *GNU utility's* en de *Linux kernel* vormen *samen* een operating system, vandaar GNU/Linux.

Een beetje geschiedenis

Om een lang verhaal kort te houden: Linux is een *UNIX*-achtige kernel waar de Finse informatica-student Linus Torvalds aan is begonnen en dat onder zijn leiding van de grond af is opgebouwd door honderden vrijwilligers.

UNIX is een OS dat vanaf de eerste computers (zogenaamde *mainframes*) is gebruikt, en dat als voorbeeld diende voor de diverse DOS-versies van vandaag. UNIX staat bekend om zijn zeer flexibele multitasking, maar ook omdat het een *multi-user* OS is. Dat wil zeggen dat er veel mensen tegelijkertijd op één enkele computer met één enkele CPU kunnen werken. In vroeger dagen was dat noodzakelijk, omdat één computer eigenlijk al onbetaalbaar was. Deze multi-user mogelijkheden werden benut door 'gewoon' een aantal extra beeldschermen en toetsenborden op dezelfde computer aan te sluiten. Een combinatie van monitor en toetsenbord wordt een *terminal* genoemd. Zo'n terminal zit met een seriële kabel aan de computer vast, op dezelfde connector als bijvoorbeeld een modem. De terminal geeft aan de computer door welke toetsen er worden ingedrukt, en de computer vertelt de terminal welke letters er op de monitor moeten verschijnen. De monitor en het toetsenbord die aan de computer vastzitten op de 'normale' manier, worden geen terminal, maar *console* genoemd. Linux heeft dezelfde mogelijkheden -- en meer zelfs.

Linus heeft alleenmaar de kernel 'geschreven'. Maar een kernel op zich kan nog niets, daar zijn utility's, een shell, en andere programma's voor nodig. Nu wil het geval, dat net toen Linux iets begon te worden, de GNU utility's beschikbaar kwamen. De GNU utility's zijn nieuw geprogrammeerde versies van alle standaard UNIX commando's / utility's (inclusief een shell), en zijn vrij gebruikbaar, ze komen zelfs met volledige programmacode (*source code*). Omdat ook Linux 'vrij' is, lag het voor de hand om de GNU utility's te gebruiken bij de Linux kernel. Aldus is GNU/Linux ontstaan.

Distributies

Naast de Linux kernel en de GNU utility's zijn veel andere programma's geschreven voor Linux. Net als de kernel en de GNU utility's, worden de meeste Linux programma's uitsluitend in de oorspronkelijke (niet-uitvoerbare) programmacode geleverd (meestal C of C++). Gebruikers moesten deze programmacode zelf omzetten (*compileren*) naar een uitvoerbaar bestand. Dat compileren is tijdrovend en er kunnen gemakkelijk fouten bij optreden.

Verschillende personen kwamen tot de conclusie dat steeds zelf compileren niet ideaal was. Ze besloten *distributies* te gaan uitgeven, dat is een pakket met een kernel, utility's, shells, en vele programma's, allemaal 'voorgecompileerd' dus direct gebruiksklaar. In die begintijd waren bijvoorbeeld de *Slackware Linux* en *SLS Linux* distributies goed bekend. Meer recente distributies zijn *Caldera OpenLinux*, *Red Hat Linux*, en de daarvan afgeleide *Linux Mandrake* en *SuSE Linux*. Hoewel tijdens het gebruik weinig verschillen merkbaar zijn (het is tenslotte allemaal GNU/Linux), lopen installatie- en configuratiemogelijkheden sterk uiteen.

2.4 Debian GNU/Linux

Debian GNU/Linux is een relatief nieuwe distributie, die in 1993 is gestart door Ian Murdock. De belangrijkste reden om met Debian te beginnen was ontevredenheid met alle bestaande distributies. Die werden namelijk door één of enkele personen (een gesloten groep) samengesteld. Allerlei fouten (*bugs*) en onnauwkeurigheden traden op, simpelweg omdat de distributeurs (*distribution maintainers*) geen tijd hadden om alle fouten op te sporen en te verhelpen. Bovendien waren er vaak problemen met het installeren, weghalen en vervangen van software, omdat een echt goed systeem hiervoor ontbrak.

Vanaf de start is Debian ontwikkeld zoals de Linux kernel: iedereen mag meehelpen, het is geen 'gesloten groepje'. Er zijn momenteel ruim 450 *developers*, meer dan bij welke andere Linux distributie ook.

Voor Debian is een compleet nieuw en zeer uitgebreid *packaging system* (pakket-systeem) ontwikkeld. Ieder programma of groepje programma's zit in een uniform *package* of *pakketje*. In elk package is ook vastgelegd hoe het geïnstalleerd moet worden, hoe het weggehaald moet worden, welke andere pakketjes het nodig heeft, en welke pakketjes er totaal niet mee kunnen samenwerken. Debian 2.2 heeft meer dan 4400 pakketjes. De programma's **`dpkg`** en **`apt`** kunnen pakketjes moeiteloos installeren en weghalen.

Om de beste kwaliteit te garanderen, wordt Debian continu getest. Vele honderden gebruikers bekijken alle mogelijkheden van alle pakketjes zeer grondig, voordat een nieuwe versie van de distributie wordt uitgegeven. Hierdoor zijn de 'officiële' Debian *releases* zo goed als foutvrij.

Elke Debian developer 'beheert' één of meer pakketjes. Hij/zij kent de betreffende programma's door en door en is ook goed thuis in de source code. Als er door Debian gebruikers fouten worden gemeld, probeert de Debian developer ze eerst zelf op te lossen. Vaak lukt dat en is er binnen 48 uur een nieuw pakketje beschikbaar. De gevonden oplossing wordt dan doorgegeven aan de oorspronkelijke auteur van het betreffende programma. Als er niet-direct-verhelpbare problemen zijn, zoeken de Debian developer en de oorspronkelijke auteur samen naar een oplossing, maar dat neemt dan meestal wat meer tijd in beslag.

Kortom, Debian GNU/Linux behoort ongetwijfeld tot de beste Linux distributies. Het is bovendien de enige 'grote' distributie die compleet gratis is (behalve de materiaalkosten voor onder andere de CD's).

Debian heeft z'n eigen versienummer (*release number*). Deze handleiding is gebaseerd op Debian versie 2.2. Maar elke versie heeft ook een (werk-)naam, een naam uit de film *Toy Story*. Debian 2.2 heet *Potato*. De vorige versie, 2.1, heette *Slink* en versie 2.3 (of misschien 3.0) zal *Woody* gaan heten.

2.5 Linux is een beetje anders

Als je voor het eerst met Linux in aanraking komt, zal het je opvallen dat er nogal wat anders gaat dan je gewend bent met bijvoorbeeld DOS of Windows. De meeste Linux-gebruikers vinden dat Linux de dingen beter regelt dan ieder ander OS, maar dat terzijde. Voordat je Linux gaat installeren, is het noodzakelijk een paar van deze verschillen te weten. Hier vertel ik alleen het globale verhaal; details komen verderop ruimschoots aan de orde.

2.5.1 Partities op de harde schijf

Het eerste dat opvalt, is dat Linux niet op een DOS-schijf kan worden geïnstalleerd. (Nou ja, het *kan* wel, maar dat is veel langzamer en wordt daarom niet ondersteund door de standaard installatie-procedure.) Linux heeft z'n eigen *partitie* nodig.

Een harde schijf is in de DOS-wereld vooral bekend als "de C:-schijf", omdat die schijf aangeduid wordt met de letter C:. Meestal staat die C: voor de hele harde schijf, maar dat kan ook anders. Een harde schijf kan namelijk worden onderverdeeld in een aantal onafhankelijke stukken, die *partities* heten. Elke partitie krijgt onder DOS z'n eigen letter. Als een harde schijf bijvoorbeeld drie partities heeft, zitten de schijven C:, D: en E: 'naast elkaar' op één en dezelfde harde schijf. Zo lijkt het net of je drie harde schijven hebt, terwijl er maar één is. De C:-schijf is gewoon het eerste stuk, de D:-schijf het tweede en de E:-schijf het laatste stuk. Een 1.6 GB (= 1638 MB) harde schijf kun je bijvoorbeeld opsplitsen in een C: van 500 MB, D: van 830 MB en E: van 308 MB. Op de harde schijf staat (helemaal aan het begin) een tabel waarin wordt bijgehouden hoeveel partities er zijn en waar die zich bevinden: de *partitietabel* (*partition table*).

In DOS hebben meerdere partities op één harde schijf niet zoveel zin. Dat wordt anders als er meerdere operating systems naast elkaar worden gebruikt. Linux heeft een eigen partitie nodig, en OS/2 bijvoorbeeld ook.

Dat is nog niet het hele verhaal. Toen men de partities 'uitvond', werd er gedacht dat vier partities per harde schijf wel ruim voldoende zou zijn. Vandaar dat in de partitietabel slechts ruimte is voor vier partities. Maar vrij snel daarna bleek dat dat toch wel wat weinig was. Om de compatibiliteit niet in gevaar te brengen, heeft men toen besloten om 'sub-partities' te maken (net als subdirectory's): binnen één 'echte' partitie kunnen meerdere sub-partities worden gemaakt. De namen zijn wat verwarrend geworden: een *primaire* partitie is een 'echte' partitie (één van de vier uit de partitietabel) waar werkelijk gegevens in staan, een *uitgebreide* (*extended*) partitie is een 'echte' partitie waar sub-partities in staan (en dus geen gegevens), en een *logische* partitie is zo'n sub-partitie, waar alleen gegevens in kunnen staan. Eén uitgebreide partitie bestaat dus (per definitie) uit een aantal logische partities. Er is trouwens maar één sub-niveau mogelijk, sub-sub-partities bestaan niet. Linux kan zowel op een primaire als op een logische partitie worden geïnstalleerd.

2.5.2 Het ext2 filesystem

De belangrijkste oorzaak waarom Linux een eigen partitie nodig heeft, is dat Linux het *Extended Filesystem versie 2* gebruikt, kortweg het *ext2 filesystem*. Een *filesystem* is de manier waarop bestanden op de harde schijf worden opgeslagen en geordend in directory's. De naam voor het filesystem dat door DOS/Windows wordt gebruikt, is *FAT*, in de varianten *FAT12* (voor bijvoorbeeld 360kB floppy's), *FAT16* (voor harde schijven) en *FAT32* (ook harde schijven, sinds Windows 95). Het systeem met driveletters als C: hoort in principe ook bij het FAT filesystem.

Het ext2 filesystem is totaal anders dan het FAT filesystem. Aan de buitenkant zijn er diverse verschillen te merken, die ik nu zal behandelen.

In het ext2 filesystem mogen bestanden zeer lange namen hebben, maximaal 255 tekens. Die bestandsnamen mogen alle letters, cijfers en andere tekens bevatten; hoofdletters en kleine letters zijn verschillend (*case sensitive*). Ook punten *mogen* in de bestandsnaam voorkomen (eventueel meerdere keren), maar extensies (als **`.exe`**) hebben geen bijzondere betekenis. Een programma 'playmidi' heet gewoon **`playmidi`**, niet **`playmidi.exe`**. Als het echt moet, mogen bijvoorbeeld '**`*`**', '**`?`**', '**`,`**', '**`+`**' en '**`>`**' ook in een bestandsnaam gebruikt worden; '**`/`**' is de enige uitzondering.

Waar in het FAT filesystem de '**`\`**' (*backslash*) wordt gebruikt voor het scheiden van directory-namen en bestandsnamen (als in **`\dir1\dir2\filenaam.ext`**), gebruikt ext2 de '**`/`**' (*slash*). Een bestandsnaam is dan bijvoorbeeld **`/home/ikke/testfile`**.

Het ext2 filesystem kent geen driveletters, en toch kunnen er meerdere schijven aangesproken worden. Er is altijd één 'hoofdschijf', die het *root filesystem* wordt genoemd. Op die schijf staat de *root directory*, waaraan alle andere directory's 'hangen'. De root directory wordt aangeduid met één enkele slash, dus **`/`**. Een aanduiding **`/System.map`** slaat dus op het bestand **`System.map`** dat in de root directory staat. Als een andere schijf aangesproken moet worden, moet die schijf ge*mount* worden ('vastschroeven'). Bij dat mounten wordt de root directory van de te mounten schijf 'geplakt over' een andere, al bestaande, directory. Verderop wordt nog een voorbeeld gegeven van het mounten van een schijf (of eigenlijk partitie).

Bestanden op een ext2 filesystem hebben allemaal een eigenaar en een groep. De eigenaar is bijvoorbeeld 'ikke' en de groep 'users'. De eigenaar, de groep en 'alle anderen' hebben bepaalde rechten, namelijk lees-, schrijf- en uitvoerrechten. Het kan voorkomen, dan een bestand lees-, schrijf- en uitvoerbaar is voor de eigenaar, alleen leesbaar voor de groep, en totaal niet toegankelijk voor alle anderen (en dan kunnen ze er ook *echt niet bij* -- dat is Linux). De rechten (of permissies, *file permissions*) worden vastgelegd in de *mode bits* van het betreffende bestand. Deze zaken komen later uitgebreid aan de orde.

2.5.3 Devices

De Linux kernel heeft tot taak met alle apparaten (*devices*) te praten, zoals harde schijven, floppy drives, cdrom drives, de printer, de muis, de geluidskaart en noem maar op. Om op een eenvoudige manier te kunnen aangeven over welk device het gaat, hebben alle devices een apart nummer gekregen. Of eigenlijk twee nummers, een soort-nummer (*major device number*) en een volgnummer (*minor device number*). Maar mensen hebben veel liever namen dan nummers, dus er is een hele rij bestanden voorhanden met een 'normale' naam, die 'doorverwijzen' naar het juiste *device number*. Die bestanden heten de *device files* en ze staan allemaal bij elkaar in de **`/dev`** directory.

Een paar voorbeelden:
`/dev/audio`: één van de manieren om naar de geluidskaart te refereren.
`/dev/lp0` t/m **`lp3`**: de parallelle printerpoorten (*line printers*);
`/dev/fd0` en **`fd1`**: de floppy drives;
`/dev/hda1` t/m **`hdh20`**: de IDE harddisks, de echte harde schijven worden aangeduid met de letters **`a`** t/m **`h`**, de partities op die schijven met de getallen **`1`** t/m **`20`**. Er kan dus verwezen worden naar de derde partitie op de tweede harde schijf met **`/dev/hdb3`**;
`/dev/sda1` t/m **`sdh16`**: de SCSI disks, aanduidingen hetzelfde als de IDE harddisks;
`/dev/scd0` t/m **`scd7`**: de SCSI CDROM drives.

Nog even wat over de harde schijven: de 'echte' partities (de vier uit de partitietabel) hebben altijd de aanduidingen **`/dev/hd.1`** t/m **`/dev/hd.4`**, het maakt niet uit of dat primaire of uitgebreide partities zijn, of dat ze helemaal niet bestaan (de vier *entries* in de partitietabel hoeven niet opeenvolgend gevuld te worden). Logische partities beginnen altijd vanaf **`/dev/hd.5`** en worden in 'oplopende volgorde van optreden' genummerd. Bijvoorbeeld: op een computer heeft de eerste harde schijf een primaire partitie **`/dev/hda1`**, en een uitgebreide partitie **`/dev/hda3`** (van de vier 'regels' in de partitietabel zijn dus alleen de eerste en de derde gebruikt). Binnen die uitgebreide partitie bevinden zich twee logische partities, **`/dev/hda5`** en **`/dev/hda6`**. De aanduidingen **`/dev/hda2`** en **`/dev/hda4`** worden dus niet gebruikt.

Een harde schijf kunnen we trouwens ook 'als geheel' aanduiden: met bijvoorbeeld **/dev/hda** of **/dev/sdb** worden de apparaten 'harde schijf' *an sich* bedoeld. We kunnen dus zeggen dat **/dev/hda1** de eerste partitie voorstelt van de harde schijf **/dev/hda**.

Voor IDE schijven geldt de volgende 'nummering':
/dev/hda - eerste schijf (master) op de eerste (primaire) controller;
/dev/hdb - tweede schijf (slave) op de eerste (primaire) controller;
/dev/hdc - eerste schijf (master) op de tweede (secundaire) controller;
/dev/hdd - tweede schijf (slave) op de tweede (secundaire) controller, enz.
Eén of meer hiervan kunnen eventueel IDE/ATAPI CDROM drives voorstellen. Een vaak voorkomende situatie is een harde schijf **/dev/hda** en een CDROM **/dev/hdc**. (In het 'dagelijks spraakgebruik' wordt **/dev** vaak weggelaten en heeft men het over "hda").

SCSI harde schijven worden opeenvolgend genummerd, beginnend bij de eerste harde schijf op de eerste SCSI-bus. In een systeem met twee bussen met elk één harde schijf gaat het dus simpelweg om **/dev/sda** en **/dev/sdb** (dat is dus anders dan bij IDE drives, waar eigenlijk de aansluiting een nummer heeft). Dezelfde redenatie geldt voor de SCSI CDROMs, die vanaf **/dev/scd0** opeenvolgend worden genummerd.

Nu we weten hoe we een (partitie op een) harde schijf moeten 'aanwijzen', kunnen we ook aangeven hoe er gemount moet worden, namelijk met een commando als **mount /dev/hdb3 /mnt**. Hiermee wordt de derde partitie van de tweede harde schijf 'geplakt' over de directory **/mnt**. Als er bijvoorbeeld een bestand **/mud/players** op hdb3 zou staan, kan dat nu worden gebruikt als **/mnt/mud/players**. Als er voor het mounten een bestand **/mnt/testfile** bestond, blijft dat wel bestaan, maar we kunnen er niet meer bij -- totdat hdb3 weer ge-*unmount* wordt met **umount /dev/hdb3** (inderdaad, 'umount' zonder 'n').

2.5.4 Consoles, terminals en tty's

Zoals al gezegd, Linux is een multi-user multi-tasking OS. Meerdere gebruikers kunnen tegelijkertijd meerdere dingen doen. Elke gebruiker heeft zijn (of haar natuurlijk) eigen scherm-met-toetsenbord (*terminal*), waarmee hij communiceert met de computer. Eén scherm-met-toetsenbord zit het dichtst bij de computer: de *console*.

Wat de console betreft, heeft Linux een extraatje. Er zijn namelijk meerdere consoles met maar één scherm en één toetsenbord: de *virtuele consoles*. Met de toetscombinaties LinkseAlt-F1 t/m LinkseAlt-F6 kan worden gewisseld tussen de diverse schermen.

Linux kan ook echte terminals aansturen. Een DEC VT100 terminal kan bijvoorbeeld worden aangesloten op een seriële poort, en dan kan daarmee gewerkt worden, op dezelfde manier als met de console.

Een derde manier om te werken met de computer, is een terminal-verbinding via een netwerk. Dan gedraagt een normale computer zich als terminal, maar gaan de signalen niet over een seriële kabel, maar over het netwerk. Er wordt dan gesproken van *virtuele terminals*.

Ook de (virtuele) consoles en (virtuele) terminals zijn apparaten en hebben dus aanduidingen in de **/dev** directory:
/dev/console: de algemene aanduiding voor de console, wordt praktisch nooit gebruikt;
/dev/tty1 t/m **tty63**: de virtuele consoles (standaard alleen **tty1** t/m **tty6** in gebruik);
/dev/ttyS0 t/m **ttyS3**: de 'echte' terminals die aan de seriële poorten (COM1 t/m COM4) kunnen worden aangesloten;
/dev/pts/0...xx: de virtuele terminals die via een netwerkverbinding kunnen worden opgezet.

Opmerking: als er een muis is aangesloten op bijvoorbeeld COM1, dan wordt daar ook **/dev/ttyS0** voor gebruikt.

2.6 Netwerken, IP adressen, namen en domeinen

Linux is een operating system dat 'gemaakt is' voor gebruik in een netwerk (zoals het Internet). Daarbij komen een aantal nieuwe begrippen om de hoek kijken. Het is niet erg als je het één en ander niet begrijpt, in paragraaf 2.6.7 wordt verteld welke instellingen je zou kunnen kiezen.

2.6.1 Protocol: TCP/IP

De communicatie over een netwerk verloopt volgens een aantal vastgestelde regels. De verzameling van al die regels wordt een *protocol* genoemd. Het protocol dat 'standaard' door Linux wordt gebruikt heet *TCP/IP*, van *Transmission Control Protocol / Internet Protocol*. Eigenlijk is dit een verzamelnaam voor een hele *suite* protocols. TCP/IP is het protocol dat ook op het Internet wordt gebruikt.

Om gegevens over een netwerk te versturen, moeten ze daar eerst geschikt voor worden gemaakt. Bij TCP/IP worden de gegevens eerst in kleine stukjes geknipt. Elk stukje wordt 'ingepakt' in een pakketje (*packet*), waar het bestemmingsadres op wordt gezet. Al die pakketjes worden op het netwerk 'uitgezonden', in de hoop dat de bestemmings-computer 'luistert'. Als er een pakketje is ontvangen, zal de bestemmings-computer een pakketje met "ik heb het ontvangen" terugsturen naar de zendende computer.

2.6.2 IP adres

Elke computer die met een TCP/IP netwerk is verbonden, moet een uniek nummer hebben, waarmee hij aangesproken kan worden -- het bestemmingsadres op de pakketjes. Dat nummer heet het *IP adres*. Het IP adres bestaat uit vier getallen van 0 tot en met 255, gescheiden door een punt, bijvoorbeeld 153.67.1.218.

2.6.3 Netmask

In een lokaal (bedrijfs-)netwerk zal het vaak zo zijn dat een computer door een netwerk is verbonden met een aantal andere computers. Die (interne) verbinding is direct, dat wil zeggen zonder tussenkomst van een andere computer. Als er gegevens moeten worden overgestuurd tussen twee computers aan dat netwerk, zijn alleen die twee computers daarvoor nodig (en de netwerkkabel natuurlijk; een eventuele 'hub' wordt beschouwd als onderdeel van de kabel).

Het kan ook zijn dat gegevens moeten worden uitgewisseld tussen een computer aan het bedrijfsnetwerk en een computer in het buitenland. Heel vaak zal die 'verbinding' via het Internet verlopen. Daartoe heeft één computer aan het bedrijfsnetwerk een 'buitenverbinding' naar het Internet, die computer noemen we een *router*. Pakketjes die bestemd zijn voor een 'buitenland-computer' moeten door de router worden opgepikt en naar het Internet 'doorgesluisd'. De afzender moet daarvoor aan de router vragen of de betreffende pakketjes alsjeblieft doorgestuurd mogen worden -- de pakketjes krijgen daarvoor een tweede laag pakpapier en het buitenste bestemmingsadres wordt de router. Als ze bij de router zijn aangekomen, scheurt die het buitenste pakpapier eraf en stuurt het pakketje naar z'n echte bestemming.

De afzender-computer moet wel weten wanneer iets via de router moet worden verzonden. Als het op het lokale netwerk blijft, hoeft dat immers niet. Ofwel: er moet bekend zijn welke IP adressen bij 'streekpost' horen; de rest hoort bij 'overige bestemmingen'. Om het simpel te houden, is bepaald dat bij lokale netwerken alleen de meest rechtse getallen van het IP-adres mogen verschillen. Bijvoorbeeld 164.75.250.5 en 164.75.250.8 zullen op hetzelfde lokale netwerk zitten. Bij hele grote netwerken kan echter ook 164.75.253.56 nog lokaal zijn. Hoeveel van de meest rechtse getallen mogen veranderen, is bij elk netwerk verschillend; dat wordt bepaald door het *netmask*. Bij kleine netwerken is het netmask meestal 255.255.255.0, grotere netwerken hebben soms 255.255.240.0 (maar er zijn talloze andere mogelijkheden). Om dat te begrijpen moeten we het binair gaan bekijken. (Als je er niet veel van begrijpt is dat ook niet erg.)

Binaire getallen worden genoteerd met énen en nullen, die een 'gewicht' hebben afhankelijk van hun plaats. Die gewichten zijn altijd machten van 2, dus 1, 2, 4, 8, 16, 32, enz. Bijvoorbeeld:

```
gewichten:    128     64     32     16      8     4     2    1
binair getal:  1       0      1      0      0     1     0    0
decimaal dus: 1x128 + 0x64 + 1x32 + 0x16 + 0x8 + 1x4 + 0x2 +0x1 = 164
```

Dus het binaire getal 10100100 is hetzelfde als het 'normale' decimale getal 164.

Terug naar de IP adressen en het netmask. De genoemde IP adressen en netmasks zijn in binair:

```
 decimaal          binair
164. 75.250. 5:   10100100 01001011 11111010 00000101
164. 75.250. 8:   10100100 01001011 11111010 00001000
164. 75.253.56:   10100100 01001011 11111101 00111000
255.255.255. 0:   11111111 11111111 11111111 00000000
255.255.240. 0:   11111111 11111111 11110000 00000000
```

Het is afgesproken dat op een lokaal netwerk dát stuk van het IP adres steeds hetzelfde is, waar er énen in het netmask staan. Bij het netmask 255.255.255.0 zijn dus steeds de eerste drie getallen van de IP adressen hetzelfde, bij dat netwerk behoren dus de adressen 164.75.250.0 t/m 164.75.250.255. Soms zijn er echter veel meer computers aangesloten aan een netwerk, dan wordt bijvoorbeeld een netmask van 255.255.240.0 gebruikt. In dat geval zijn de linkse tweeënhalf getallen steeds hetzelfde en we krijgen een netwerk van adressen 164.75.240.0 t/m 164.75.255.255.

Ofwel, als een computer een pakketje wil versturen naar een adres dat afwijkt van zijn eigen adres op plaatsen waar het netmask een één heeft staan, zal het naar de router gestuurd moeten worden.

2.6.4 Network address

Er is afgesproken dat ook het lokale netwerk zelf een adres heeft, het *network address*. Dat is gelijk aan het allereerste adres wat dat netwerk kan 'bedienen'. In ons voorbeeld dus 164.75.250.0 bij het netmask 255.255.255.0, en 164.75.240.0 bij het netmask 255.255.240.0.

2.6.5 Broadcast address

Het komt regelmatig voor, dat er een bericht naar alle computers op een netwerk tegelijk gestuurd moet worden. Daar is een speciaal adres voor gereserveerd, het *broadcast address*, waar alle aangesloten computers naar luisteren. (Ze sturen echter geen bevestiging, dat zou een warboel worden.) De keuze voor dat broadcast address is vrij, maar meestal wordt het laatste 'bedienbare' adres genomen, in een enkel geval ook het eerste (dan zijn network address en broadcast address dus hetzelfde). Als we de trend volgen, zijn de broadcast addressen in ons voorbeeld dus 164.75.250.255 bij netmask 255.255.255.0, en 164.75.255.255 bij netmask 255.255.240.0.

2.6.6 Gateway address

De pakketjes voor 'overige bestemmingen' moeten (als eerste) naar de router worden gestuurd, die het daarna verder stuurt. De router wordt ook wel *gateway* genoemd. Het adres van de router, of gateway, moet bij de zendende computer bekend zijn -- dat heet het *gateway address*. Vaak wordt aan de router/gateway het eerste 'normale' adres gegeven, in ons voorbeeld dus 164.75.250.1 bij netmask 255.255.255.0, en 164.75.240.1 bij netmask 255.255.240.0.

2.6.7 Welke adressen?

Tijdens de installatie van Debian zal gevraagd worden om de adressen die ik net behandeld heb. De vraag is dan welk adres waarvoor genomen moet worden. Er zijn verschillende situaties denkbaar, waarvoor ik nu een mogelijke oplossing zal beschrijven.

Bedrijfsnetwerk (met systeembeheerder)
In een bedrijfsnetwerk liggen network address, netmask, broadcast address en gateway address al vast, ze zijn voor iedere aangesloten computer hetzelfde. Het IP adres is voor elke computer verschillend. Al deze gegevens kun je van de systeembeheerder te weten komen. Wil je hem daar niet mee lastig vallen, en installeer je op een computer waar al een ander OS (bijv. Windows) op staat, dan kun je door goed zoeken in de bestaande instellingen alles wel te weten komen.

Mocht er geen gateway address zijn, dan is het lokale netwerk simpelweg niet aan het Internet verbonden. Het kan ook zijn dat Debian wordt geïnstalleerd op de computer die zelf gateway gaat worden, in dat geval is er ook geen gateway address (de kernel zoekt zelf uit naar welke verbinding de pakketjes moeten worden gestuurd -- maar dat vereist nog wel wat meer werk, zie paragraaf 12.3).

Als in het netwerk automatische adres-toewijzing plaatsvindt met DHCP of BOOTP, is dat nog makkelijker; dat wordt al tijdens de installatie ondersteund.

Eigen netwerk (je bent zelf systeembeheerder)
In het geval van een netwerk dat je zelf (thuis?) gaat opzetten, mag je alles zelf bepalen (tenzij je een reeks vaste IP adressen hebt gekocht, behandel het dan als bedrijfsnetwerk). In principe zou je alle IP adressen mogen nemen. Maar stel je voor dat je een computer het adres 1.2.3.4 geeft, terwijl dat adres ook al op het Internet in gebruik is. Als je dan vanaf je lokale netwerk een WWW-pagina van computer 1.2.3.4 wilt ophalen, krijg je de computer op je eigen lokale netwerk en niet die van het Internet. Dat is waarschijnlijk niet de bedoeling. Vandaar dat er een aantal nummers zijn gereserveerd die niet op het Internet voor (mogen) komen. Dat zijn de volgende:

10.0.0.0 - 10.255.255.255
172.16.0.0 - 172.31.255.255
192.168.0.0 - 192.168.255.255

Vaak wordt het bereik 192.168.1.0 t/m 192.168.1.255 gekozen, netmask 255.255.255.0, network address dus 192.168.1.0, broadcast address 192.168.1.255. Nummer de computers opeenvolgend vanaf 192.168.1.1. Als je een inbel of vaste Internet verbinding hebt, zal het modem / de verbinding aan één bepaalde computer zitten (die heeft dus twee netwerkverbindingen, een 'interne' en een 'externe'), het (interne!) adres van die computer is dus het gateway address; op de gateway zelf is er geen gateway address. Als je geen Internet verbinding hebt, dan heb je vanzelfsprekend ook geen gateway.

NB: als je een inbelverbinding hebt, kan het zijn dat je ISP jou een nummer geeft uit één van de bovengenoemde bereiken (als je in Debian inbelt, is dat tijdens het inbellen te zien met het programma **`/sbin/ifconfig`**). Je kunt dan voor je interne netwerk beter één van de andere bereiken kiezen.

Geen lokaal netwerk
Als er geen (lokaal) netwerk is, dan hoef je ook geen IP address enzovoort op te geven. Een eventueel inbel-account bij een Internet Service Provider wordt pas geregeld als de Debian installatie is voltooid en het systeem 'normaal draait'. Beschouw een vaste Internet-verbinding met een netwerkkaart (sommige kabelmodems) echter als een bedrijfsnetwerk, zoals boven beschreven.

2.6.8 Naam van de computer

Omdat IP adressen zo lastig te onthouden zijn, heeft men besloten de computers ook maar een naam te geven. Want 'elektron.et.tudelft.nl' is toch veel aansprekender dan 130.161.33.15. De namen zijn volgens een hiërarchisch systeem opgebouwd, van links naar rechts steeds algemener. De computer die wordt aangeduid met 'elektron.et.tudelft.nl' heeft de *naam* 'elektron'. Hij staat op de faculteit Elektrotechniek, afgekort ET, van de Technische Universiteit Delft, TUDelft, in Nederland, NL. Het gedeelte na de eerste punt wordt het *domein* genoemd, in het voorbeeld dus 'et.tudelft.nl'. 'Elektron' is één van de vele computers die 'hangen onder' het domein 'et.tudelft.nl'. Eigenlijk zijn 'tudelft.nl' en 'nl' zelf ook (grote) domeinen. In namen van computers en domeinen maken kleine letters en hoofdletters geen verschil (*case insensitive*).

Een nieuwe naam
Tijdens de installatie van Debian zal worden gevraagd om de naam van de computer (*hostname*) en, als er een netwerk is, ook het domein (*domain name*). In een bedrijfsnetwerk ligt in ieder geval het domein vast en vaak ook de hostname. In andere gevallen (met en zonder netwerk) zul je zelf wat moeten verzinnen. Het domein van alle computers in het netwerk zal wel steeds hetzelfde moeten zijn.

Het blijkt vaak lastig om originele namen te bedenken. Een paar ideeën:
- namen van (niet) beroemde filmsterren, sportmensen, presidenten, wetenschappers;
- namen van (grote/kleine) plaatsen, bergen, rivieren in een bepaalde provincie, streek, land, werelddeel;
- namen van (niet) bekende planten, bomen, diersoorten.

Het is aan te raden voor de domeinnaam een niet-bestaand domein te kiezen (als je een eigen nameserver gaat gebruiken kun je anders in de problemen komen). Een idee dat vaak werkt is een combinatie van achternaam, straat en/of woonplaats. Bijvoorbeeld 'dorpsstraat.onsdorp.nl', maar 'dorpsnet.nl' is dan origineler. Als totale naam krijgen we bijvoorbeeld 'kennedy.dorpsnet.nl' -- niet gek.

Let op: de hostname en de domain name zijn twee van de weinige dingen die na installatie heel lastig te veranderen zijn, ze worden namelijk op veel verschillende plaatsen opgeslagen. Denk er dus goed over na!

Nameservers
Om namen aan nummers te koppelen, zijn in Linux twee mechanismen beschikbaar. De eerste is het bestand **`/etc/hosts`**. Daarin staat een tabelletje met IP adressen en de bijbehorende namen. Als je bijvoorbeeld in een webbrowser een naam intypt, wordt eerst in **`/etc/hosts`** gekeken. Als de naam daar niet in staat, wordt de tweede zoektocht gestart. Die houdt in dat er aan speciale computers wordt gevraagd of de naam misschien bekend is. Die computers heten *nameservers* en houden automatisch grote reeksen namen en nummers bij. Elke Internet Service Provider heeft één of meer nameservers, en de meeste bedrijfsnetwerken hebben er ook één.

Tijdens de installatie van Debian wordt om een nameserver gevraagd (ook wel *DNS server* genoemd, van *Domain Name Service*). Als je nooit aan Internet verbonden zult zijn, is er waarschijnlijk geen nameserver nodig en volstaat de **`/etc/hosts`** file. In andere gevallen zal er een nameserver moeten worden opgegeven. In bedrijfsnetwerken kun je het IP adres (niet de naam!) aan de systeembeheerder vragen (of zelf opzoeken), bij een inbel of vaste Internetverbinding heb je dat van je Internet Service Provider gehoord. In geval van nood kun je 130.161.180.1

gebruiken (nameserver van de TUDelft). Merk op dat de nameserver niet in het lokale netwerk hoeft te liggen; vragen aan de nameserver worden eventueel netjes via de gateway gestuurd. Je kunt ook zelf een nameserver opzetten, maar dat komt pas later ter sprake (paragraaf 12.6). Gebruik eerst een bestaande (goed werkende) nameserver.

2.7 Gebruikersnamen en wachtwoorden

Linux is een multi-user operating system. Meerdere gebruikers kunnen een enkele computer gebruiken. Bij Linux staat veiligheid voorop. Vandaar dat je bijvoorbeeld geen bestanden van andere gebruikers kunt weggooien, en je kunt ze ook niet inkijken als de toestemming daarvoor is geweigerd. Daarom moet je je identificeren en legitimeren voordat je in Linux kunt gaan werken -- dat heet *inloggen*. Identificeren is 'zeggen wie je bent'. Elke gebruiker heeft een gebruikersnaam (*username*) ofwel *inlognaam* (*login name*), die bij het inloggen moet worden ingetypt. Door je te legitimeren weet het systeem zeker dat je bent wie je zegt te zijn. Voor dat doel heeft iedere gebruiker ook een (geheim!) wachtwoord (*password*). Dat moet ook worden ingetypt tijdens het inloggen, maar dat komt niet op het scherm te staan (ook niet in sterretjes o.i.d.) -- stel je voor dat iemand over je rug meekijkt.

Een inlognaam bestaat over het algemeen uit alleen maar kleine letters. Een password daarentegen is meestal een combinatie van kleine en grote letters, cijfers en leestekens. De leestekens zijn die, die op een normaal toetsenbord zijn terug te vinden; maar het hekje ('#') wordt meestal afgeraden, omdat dit bij 'echte' UNIX systemen de backspace voorstelt (de backspace-toets werkt daar bij het inloggen (nog) niet). In Linux is het hekje wel te gebruiken, en werkt de backspace-toets normaal. Zowel loginnaam als password mogen maximaal acht tekens lang zijn; een password van minder dan zes tekens wordt als te kort (=onveilig) beschouwd.

Een voorbeeld: gebruiker Karel Appel zal bij een 'klein' systeem waarschijnlijk de inlognaam 'karel' hebben. Bij bedrijven lopen wellicht meer Karels rond, 'onze' Karel wordt dan 'karela' of 'kappel'. Een redelijk goed password is bijvoorbeeld 'KaApDs13', van Karel Appel die woont aan de Dorpsstraat, nummer 13.

Er is één gebruiker die alles kan: de systeembeheerder (*system administrator* of *sysadmin*, ook *system manager* of *sysman*, *system operator* of *sysop*). In Linux heet de systeembeheerder *root* (="belangrijkste", vergelijk het root filesystem). De gebruikers kunnen de instellingen van het systeem niet veranderen, root wel. De gebruikers kunnen geen essentiële programma's weggooien, root wel. En, nog erger, root is de enige die *alle* bestanden met *één* commando zonder vraag om bevestiging kan wissen... Kortom: root zijn is een extreem gevaarlijke bezigheid. Vandaar dat je *zo weinig mogelijk* root moet zijn. En het password van root moet dus ook *zo geheim mogelijk* zijn en *absoluut niet raadbaar*.

Een goed root-password is bijvoorbeeld 'hDy*d6%'. Ik heb dit 'verzonnen' door een hele ris karakters al dan niet met Shift ingedrukt, willekeurig op het toetsenbord te 'rammen', met m'n ogen dicht. Uit die lange rij heb ik er willekeurig een groot aantal weggeknipt, totdat ik er voldoende over hield. Aan jou de taak om net zo'n sterk wachtwoord te 'genereren' -- maar neem natuurlijk *NIET* dit voorbeeld!! Om het niet te vergeten, mag je het best ergens opschrijven, maar zet dan *niet* op het papiertje wat die tekenreeks betekent, en berg het goed op (*niet* bij de computer in de buurt).

3. Installatie van Debian GNU/Linux 2.2

Nu we een beter overzicht hebben gekregen van de 'wereld van Linux', kunnen we gaan beginnen met de installatie van Debian GNU/Linux. We zullen zo af en toe nog wel wat 'theorie' moeten behandelen, om alles tot een goed einde te brengen. Het is verstandig dit hele hoofdstuk eerst goed door te lezen, voor je met installeren gaat beginnen. Dan weet je ongeveer hoe het één en ander gaat verlopen.

In dit hoofdstuk wordt gebruik gemaakt van informatie uit *Installing Debian GNU/Linux 2.2 For Intel x86* door Bruce Perens e.a. (het bestand **`install/doc/install.en.html`** of **`.txt`** op de Debian Binary-1 CDROM, of **`/debian/dists/potato/main/disks-i386/current/doc/install.en.html`** of **`.txt`** op een Debian FTP site). Het is een goed idee ook die (Engelse) tekst door te lezen voor je met installeren gaat beginnen.

De installatie gaat het eenvoudigst als je tenminste een (officiële) Debian 2.2 Binary-1 CDROM hebt. De bestandsnamen die hieronder genoemd worden, zijn daarvoor direct geschikt. Het is echter ook mogelijk om alles te downloaden met FTP; meer daarover staat op http://www.debian.org onder "Download FTP". Beslis wel eerst welke installatie-methode je gaat gebruiken (lees verder...), je hebt namelijk maar een beperkt aantal bestanden nodig. Bij elkaar zijn de 'eerste installatie'-bestanden maximaal ongeveer 25 MB.

NB: dit hoofdstuk gaat dus *niet* over het upgraden van een oude Debian versie naar versie 2.2. Kijk daarvoor in de **`/upgrade`** directory op de Debian Binary-1 CDROM, of in de **`/debian/dists/potato/main/upgrade-i386`** directory op een Debian FTP server.

Het is overigens geen enkel probleem om de installatie-procedure een paar keer 'uit te proberen'. Op een gegeven moment is alles naar wens, en ga je echt beginnen. Vanaf die tijd geldt de regel "nooit her-installeren, behalve bij een defecte harddisk". Het is ook aan te raden om tijdens het installeren en het eerste gebruik een kladblaadje bij de hand te houden, waar je aandachtspunten, instellingen enz. op kunt noteren.

3.1 Voorbereiden

Mijn ervaring is dat het installeren van Debian niet veel ingewikkelde voorbereiding vergt naast het vrijmaken van ruimte op de harde schijf voor de partities waar Debian op moet komen. Daarnaast moeten er één of meer diskettes worden beschreven.

3.1.1 Backup van alle harde schijven

Er bestaat een kleine kans dat je, ofwel door fouten in het installatieprogramma, danwel door eigen fouten, alle gegevens op alle harde schijven kwijtraakt. Het is daarom noodzakelijk dat je, voor met de Debian installatie te beginnen, veiligheidskopieën maakt van alle bestanden op alle harde schijven. Dit kan worden gedaan door bijvoorbeeld een tweede harde schijf te lenen, daar alle bestanden naartoe te kopiëren, en deze tweede harde schijf voorafgaand aan de Debian installatie los te koppelen en op een veilige plaats te bewaren. Een andere mogelijkheid is de eigen harde schijf mee te nemen naar iemand met een CD-recorder, die dan een backup op CD-R kan maken. Zorg er in ieder geval voor dat alle ongebruikte harde schijven tijdens de Debian installatie zijn losgekoppeld.

Zo. Nu kun je mij in ieder geval niets meer maken. Laat ik je echter gerust stellen: dit heb ik nog nooit iemand zien doen en alles is altijd goed gegaan. Maar ik heb je gewaarschuwd.

3.1.2 Ruimte op de harde schijf

Debian heeft tenminste twee partities op een harde schijf of schijven nodig. Een partitie voor het root filesystem en een *swap partitie*. De swap partitie is net zoiets als het wisselbestand (virtuele geheugen) van Windows: een stuk harde schijf waar stukken werkgeheugen worden gedumpt als ze even niet worden gebruikt, zodat het echte (snelle) werkgeheugen voor andere doeleinden kan worden gebruikt.

De vraag is altijd, hoe groot die partities moeten zijn. Het root filesystem moet tenminste 65 MB zijn, maar daar past alleen een zeer minimaal functioneel systeem op. 200 MB is nodig voor een aardig werkend systeem, en ik ga er hier van uit dat je tenminste zoveel hebt. Met 750 MB kun je praktisch alle nuttige zaken installeren en dan nog wat ruimte overhouden voor jouw eigen bestanden. In het algemeen geldt: hoe meer, hoe beter. (Als je alles wat je nooit nodig hebt ook wilt installeren, heb je aan 4 GB nog niet genoeg.)

Wat de swap partitie betreft, is een goede vuistregel: tenminste twee maal de grootte van het werkgeheugen, tenminste 30 MB, maar maximaal 100 MB (regels in die volgorde behandelen). Een paar voorbeelden: 8 MB werkgeheugen, dan 30 MB swap; 32 MB werkgeheugen, dan 64 MB swap; 128 MB werkgeheugen, dan 100 MB swap. Meer dan 100 MB swapruimte mag natuurlijk ook, maar dat heeft waarschijnlijk niet veel zin bij normaal gebruik. Als je aan 100 MB swap niet genoeg hebt, kun je beter over een fikse werkgeheugen-uitbreiding gaan denken.

Gevorderde Linux gebruikers adviseren soms om de **/usr** en **/tmp** en eventueel **/home** directory's op aparte partities te zetten. Dat raad ik beginnende Linux-gebruikers echter niet aan. 'Losse' partities blijken al gauw óf te klein, óf te groot, wat tot allerlei ingewikkelde 'schuifacties' gaat leiden.

Als er meerdere harde schijven beschikbaar zijn, kan het interessant zijn de swap partitie op een andere harde schijf te zetten dan het root filesystem (danwel het filesystem waar **/usr** op staat). Dat bevordert de snelheid van het systeem, omdat de zoektijden tussen de twee partities dan verdwijnen.

Om Linux te installeren, moet je dus op zoek gaan naar (bijvoorbeeld) 750+50=800 MB vrije ruimte (niet door een bestaande partitie gebruikt) op een harde schijf. Het eenvoudigste is het gebruiken van een hele nieuwe harde schijf, of een stuk van een harde schijf dat nog niet door een partitie is bezet (je kunt bijvoorbeeld alle andere OS'en weggooien). Er is echter ook een manier om een bestaande DOS, Windows of OS/2 partitie 'in te krimpen', zodat Debian op de vrijgekomen ruimte gezet kan worden. De procedure hiervoor staat in de volgende paragraaf beschreven; als je al voldoende ongepartitioneerde ruimte hebt, kun je de volgende paragraaf dus overslaan.

3.1.3 Knippen in een bestaande partitie

Als je een stuk van een DOS, Windows of OS/2 partitie wilt gebruiken voor Debian, zul je natuurlijk eerst een schijf moeten vinden die genoeg vrije ruimte over heeft (denk ook eens aan een 'grote schoonmaak'). Als je die gevonden hebt, moeten alle bestanden achter elkaar op het eerste stuk van die partitie gezet worden. Dan is het achterste gedeelte van die partitie helemaal leeg en kan het van de partitie worden 'afgeknipt'.

Voor het 'afknippen' moet het programma **fips** worden gebruikt. Dat staat op de Debian Binary-1 CD in **tools/fips20.zip** en op een Debian FTP site in dezelfde directory. Je kunt het 'unzippen' in een lege directory met bijvoorbeeld WinZip vanuit Windows of **pkunzip** vanuit DOS, maar ook met **unz512x3.exe** uit diezelfde directory (is een zelf-uitpakkend archief). Het verdient de aanbeveling de bijgeleverde documentatie goed door te lezen, zeker als je OS/2 gebruikt, of een schijfcompressieprogramma als Stacker, SuperStor, DoubleSpace, of als je een driver-programma voor je harde schijf gebruikt als OnTrack Disk Manager. Ik zal er in het vervolg van uitgaan dat je genoemde zaken *niet* gebruikt. Als je problemen tegenkomt, *moet* je eerst de fips documentatie lezen, voor je gaat klagen.

NB: fips kan alleen primaire partities 'knippen' en geen logische partities. Als je er niet met fips uitkomt, kun je eens kijken naar de freeware 'Ranish Partition Manager' voor Windows (http://www.users.intercom.com/~ranish/part), of de commerciële 'Partition Magic' (http://www.powerquest.com/partitionmagic) of 'Partition It' (http://www.symantec .com/region/au_nz/product/partition-it/main.html). Gebruik echter in geen geval 'Disk Druid', dat bij Red Hat Linux werd geleverd; vele schijven zijn daar al door vernield.

Het is gebruikelijk voor het 'fipsen' op te starten van een floppy, om te voorkomen dat allerlei device drivers of systeemprogramma's de harde schijf benaderen tijdens het 'knippen', wat tot verlies van bestanden kan leiden. In het betreffende OS is een mogelijkheid om een opstartdiskette te maken, waar altijd een nieuwe, geformatteerde diskette voor nodig is. Geef in DOS de opdracht **sys a:** (als A: de gebruikte floppy drive is). In Windows 95/98 gaat het via het Configuratiescherm (Control Panel), optie 'Software', tabblad 'Opstartdiskette' ('Boot disk').

Als de diskette 'opstartbaar' is gemaakt, kopieer je de bestanden **RESTORRB.EXE**, **FIPS.EXE** en **ERRORS.TXT** ernaar toe (uit de directory waarin je **fips20.zip** 'ge-unzipt' hebt). Probeer vervolgens of je er goed mee kunt opstarten, maar doe verder nog niets; start je oorspronkelijke OS weer op. Als het opstarten van diskette niet lukt, probeer dan een hele nieuwe diskette te maken.

Voordat je fips kunt gebruiken, moet je eerst de harde schijf 'schoonvegen'. Daarvoor gebruik je het *defragmenteerprogramma* dat (als het goed is) is meegeleverd met het operating system waar de betreffende partitie voor wordt gebruikt. In DOS (vanaf versie 6) heet dat programma **defrag.exe**, maar je kunt ook Nortons **speedisk.exe** (of **sd.exe**) gebruiken. In Windows 95/98 selecteer je in de Verkenner (Explorer) de betreffende harde schijf en kies je menu 'Bestand' ('File'), 'Eigenschappen' ('Properties'), dan tabblad 'Extra' en knop 'Nu defragmenteren' ('Defragment now').

Probeer nooit een Windows "lange bestandsnamen"-partitie met een DOS programma te defragmenteren! Sluit vóór het defragmenteren alle andere programma's af. Kies altijd voor een volledige defragmentatie of volledige optimalisatie (*niet* 'defragment files only' of 'defragment free space'; in Windows 95/98 met de knop 'Geavanceerd' of 'Advanced').

Let erop, dat er achteraan de schijf (partitie) misschien onverplaatsbare ('unmovable') bestanden staan. Dit is meestal te zien in het defragmentatieprogramma. Gebruik een optie als 'Walk Map' om uit te vinden om welke bestanden het gaat. Die bestanden hebben het *hidden* of *system* attribuut, om dat uit te zetten gebruik je (in DOS) het commando **ATTRIB -h -s <filenaam>** (voor <filenaam> moet je dus de bestandsnaam invullen; <...> is een gebruikelijke manier om "hier iets invullen" aan te geven). Als het de bestanden **image.idx** en/of **mirrorsav.fil** betreft, kun je die (*na* het defragmenteren) gewoon weggooien; geef voor het weggooien wel eerst het commando **ATTRIB -r -s -h <filenaam>**. Ook wisselbestanden (*swap files*) van bijvoorbeeld Windows zouden op deze manier verwijderd moeten kunnen worden.

Als het defragmenteren is voltooid (kan enkele uren duren), sluit dan alles zo snel mogelijk af en start opnieuw op met de fips opstartdiskette. (Als je was vergeten de fips diskette in de drive te doen, sluit dan alles normaal af en start weer opnieuw op; het is echter mogelijk dat je dan nog een keer zult moeten defragmenteren.) Je zult waarschijnlijk de mogelijkheid krijgen de tijd en datum te veranderen, geef daar gewoon Enter ('niet veranderen').

Start nu **fips.exe** (van de opstartdiskette dus). Fips is op elk moment af te breken met de toetscombinatie Ctrl-C. Als je meerdere harde schijven hebt, vraagt fips waar de partitie op staat die 'geknipt' moet worden. Vervolgens wordt een overzicht getoond van de partities op de betreffende harde schijf. Alleen de meest linkse en meest rechtse kolom zijn van belang (allemaal nullen betekent een regel in de partitietabel die niet gebruikt is). Als er meerdere 'splitsbare' partities zijn, vraagt fips het nummer van de te 'knippen' partitie, gebruik de getoonde tabel om dat te bepalen. Daar worden vervolgens weer meer gegevens over afgebeeld, die je over mag slaan. Fips bekijkt tot waar de partitie gevuld is, en staat alleen toe een stuk 'af te knippen' dat ongebruikt is. Je kunt nu aangeven op welke plaats er geknipt moet gaan worden. De grootte van het eerste stuk en het 'afgeknipte' stuk worden getoond en zijn te veranderen met de cursortoetsen links/rechts, en in grote stappen met op/neer. Als je tevreden bent (knip niet alles af!), druk je op Enter. Druk als je het overal mee eens bent op 'c' en op 'y'. De oorspronkelijke partitie is nu ingekrompen en er is een nieuwe DOS/Windows-partitie gemaakt van het afgeknipte stuk. Die nieuwe partitie zullen we tijdens de Debian installatie weer weggooien en de zo vrijgekomen ruimte gebruiken voor de twee benodigde Linux-partities. Wacht nu zeker tien seconden en start de computer opnieuw op naar het gebruikelijke OS. Als alles gelukt is, zou je niets moeten merken, behalve dat je veel minder vrije schijfruimte over hebt (die is er immers net afgeknipt).

Nogmaals: als er problemen zijn (foutmeldingen van fips bijvoorbeeld), lees dan eerst de fips documentatie!

3.1.4 Gang van zaken tijdens de installatie

De bestanden die nodig zijn om Debian te installeren, kunnen van diskettes, de harde schijf, een CDROM of het netwerk worden gehaald, of een combinatie hiervan. Ik zal hieronder de installatieprocedure globaal beschrijven; in de volgende paragraaf worden de benodigde onderdelen bij elke stap toegelicht. Daarna volgt een veel gedetailleerdere uiteenzetting van de precieze procedures.

Tijdens de installatie zijn de volgende onderdelen nodig:
opstartmethode - het starten van de Linux kernel;
installatie-systeem - het filesystem met de bestanden die tijdens de installatie worden gebruikt;
device drivers - de vele stuurprogramma's voor allerlei (rand-)apparaten;
Base System - het 'basis' filesystem dat op de harde schijf zal worden geïnstalleerd.

Tijdens het installeren wordt een tijdelijk filesystem gebruikt. Dat filesystem bevat een aantal bestanden en programma's die voor de installatie zorgen: het installatie-systeem. Er moet een apart filesystem worden gebruikt, omdat ook de installatie zelf al in Linux plaatsvindt. Het installatie-filesystem wordt meestal niet 'op de diskette' gebruikt, maar op een *ramdisk*. Dat is een stuk werkgeheugen dat op dezelfde manier wordt behandeld als een diskette (of harde schijf), je kunt bestanden ervan lezen en ernaar schrijven (maar wijzigingen gaan natuurlijk verloren als je de computer uit zet).

Als Debian geïnstalleerd moet worden, moet dus eerst de Linux kernel worden gestart. Dat kan bijvoorbeeld gebeuren door het systeem op te starten (te *booten* of *rebooten*) van een Linux opstartdiskette. Om diskruimte te besparen, is de kernel gecompress (gecomprimeerd). Een 'kernel op disk' wordt trouwens een *kernel image* genoemd -- het is een 'beeld' van de kernel, die 'normaal' in het geheugen hoort te zitten. Op de opstartdiskette staat dus een *compressed kernel image*. Aan het begin daarvan staat een klein stukje niet-gecompresste code. Als de kernel moet worden gestart, wordt het hele gecompresste kernel image in het werkgeheugen geladen en wordt het kleine stukje niet-gecompresste code gestart. Dat zorgt ervoor, dat de kernel “uitgepakt” (*gedecomprimeerd*, *geunzipt*, *gedecompresst*) wordt en gestart (vergelijkbaar met zelf-uitpakkende (“*self-extracting*”) archieven/programma’s).

Het eerste wat de kernel doet, is het starten (*initialiseren*) van allerlei apparaat-aansturingen, zoals de videokaart, IDE en SCSI interfaces. Er kunnen een aantal foutmeldingen komen omdat apparaten niet gevonden zijn, maar dat is niet erg -- de kernel moet een heleboel (ook ongebruikelijke) apparaten aan kunnen sturen, en die heb jij waarschijnlijk niet allemaal.

Als alles in de kernel geïnitialiseerd is, moet er een programma gestart worden. Dat programma staat in dit geval op het tijdelijke filesystem dat op de ramdisk moet komen. Tussen het laden van het gecompresste kernel image en het starten ervan, werd daarom ook het gecompresste *ramdisk image* in het geheugen geladen. Na de initialisaties ziet de kernel dat er een ingepakt ramdisk image in het geheugen staat, dat hij gaat uitpakken en als ‘gewone’ disk gaat behandelen.

Het programma dat altijd als eerste wordt gestart, is **`/sbin/init`**. In het geval van de Debian installatie staat dat programma dus op het tijdelijke filesystem, dat ondertussen op de ramdisk staat. Dat programma start de installatie-procedure, waar je aan de hand van een menustructuur doorheen wordt geloodst. Bij een aantal onderdelen kun je weer diskettes nodig hebben, of andere media.

Als er een partitie voor het root filesystem (genoemd de *root partitie*) en een swappartitie zijn gemaakt, moeten de kernel en de device drivers op die nieuwe root partitie gezet worden. De kernel kan bijvoorbeeld van de opstartdiskette gelezen worden, en de device drivers van de “driver” diskettes. De device drivers kunnen daarna eventueel geconfigureerd en geladen worden, zodat bijvoorbeeld het netwerk toegankelijk wordt.

Als laatste wordt het basis filesystem van bijvoorbeeld de “Base System” diskettes op het nieuwe root filesystem gezet. Op die diskettes staan opeenvolgende stukken van een ‘normaal’ archief (ingepakt dus) van het formaat .tar.gz (“tar-gzip”, *tape archive* dat ingepakt is met *GNU zip*, het meest gebruikte archiefformaat in UNIX en Linux, zie ook paragraaf 4.19). Ze staan er echter niet normaal op, maar in een speciaal formaat dat de hele diskette gebruikt (inclusief waar normaal de directory staat) -- weer een plaatsbesparing. Deze diskettes zijn dus in DOS/Windows niet leesbaar.

3.1.5 Manieren om te installeren

Alle bestanden die hieronder ter sprake komen, inclusief **`rawrite2.exe`**, zijn te vinden op de Debian Binary-1 CDROM, onder de directory **`/dists/potato/main/disks-i386/2.2.16-2000-07-14`**, in DOS **`\dists\potato\main\disks-i3\2_2.16-`**. Op de Debian FTP site is het dezelfde directory, alleen meestal met **`/debian`** ervoor. Dit is correct voor de oorspronkelijke release van Debian 2.2; eventueel volgende revisies kunnen een iets andere laatste component hebben.

Overzichtstabel

De onderstaande tabel geeft een overzicht van de bestanden die bij elke stap van de installatie-procedure gebruikt worden. Daaronder volgt gedetailleerdere uitleg. Merk op dat diverse bestanden tijdens meerdere stappen gebruikt worden; als het om diskettes gaat, hoef je die natuurlijk maar één keer te schrijven.

Medium	Opstarten + Installatie-systeem	Kernel + Device Drivers	Base System
3½″ diskettes (1.44 MB)	images-1.44/rescue.bin images-1.44/root.bin	-1.44/rescue.bin -1.44/driver-1.bin t/m -3	-1.44/base-1.bin t/m -11
5¼″ diskettes (1.2 MB)	images-1.20/rescue.bin images-1.20/root.bin	-1.20/rescue.bin -1.20/driver-1.bin t/m -4	-1.20/base-1.bin t/m -13
Harddisk	install.bat linux dosutils/loadlin.exe images-1.44/root.bin	-1.44/rescue.bin drivers.tgz	base2_2.tgz
CDROM [IDE/SCSI]	Opstarten van CD of install.bat linux dosutils/loadlin.exe images-1.44/root.bin	-1.44/rescue.bin drivers.tgz	base2_2.tgz
CDROM [Andere]	--	--	base2_2.tgz
Netwerk [HTTP]	--	--	base2_2.tgz

Flavours

Het installatie-systeem kent verschillende ‘smaken’ (*flavours*), elk met specifieke hardware-ondersteuning:

vanille (*vanilla*, standaard) - ondersteuning voor alle hardware, behalve de Promise Ultra66 controller voor UDMA66 harddisks.
compact - alleen ondersteuning voor IDE en SCSI harddisks en een beperkt aantal netwerkkaarten, verder geen ondersteuning voor andere apparaten.
idepci - alleen ondersteuning voor IDE harddisks en een beperkt aantal PCI netwerkkaarten, verder geen ondersteuning voor andere apparaten.
udma66 - ondersteuning voor alle hardware, plus de Promise Ultra66 controller, maar niet gegarandeerd werkend op alle systemen.

Om te bepalen welke smaak je moet gebruiken gelden de volgende vuistregels:

1) als je diskettes gebruikt: de ‘compact’ smaak;
2) alle andere methodes: de ‘vanille’ smaak;
3) als één smaak niet blijkt te werken, probeer je een andere.

In de bovenstaande tabel zijn steeds de ‘vanille’ bestanden aangegeven. Bestanden voor de ‘compact’, ‘idepci’ en ‘udma66’ smaken staan steeds één directory dieper. Dat gaat dan alleen om de eerste twee kolommen, want het Base System is in alle gevallen identiek. Merk op dat verschillende smaken meestal een ander aantal driver disks hebben.

Als je van de Debian Binary-1 CD opstart, krijg je de ‘vanille’ smaak. De Binary-2 CD heeft ‘compact’, maar voor de overige onderdelen moet je weer de Binary-1 CD gebruiken.

Diskettes

Diskettes zijn de meest universele manier om te installeren, in die zin dat ze altijd en overal werken. Dat wil zeggen, als ze goed geschreven zijn. Het kan gebeuren, dat tijdens de installatie blijkt dat een diskette niet goed leesbaar is. Breek de installatie dan netjes via het menu af en schrijf een compleet nieuwe diskette. Het kan helpen om een andere computer/drive te gebruiken voor het schrijven van de diskettes. Als je de installatie weer opnieuw start, hoef je de meeste instellingen niet opnieuw op te geven.

Voor elk onderdeel van de installatie zijn de benodigde bestanden in de tabel weergegeven. In deze bestanden staan *complete* diskettes (dus met complete directorystructuur enzovoort). Deze *image files* moeten, voor zover je ze wilt gebruiken, met het programma **rawrite2.exe** op diskettes gezet worden. Als je Windows gebruikt, sluit dat dan af en keer terug naar DOS mode. Ga in DOS met **cd** naar de juiste directory op de CDROM (bijvoorbeeld **...\images-1.44**) en geef daar een commando als:

```
..\dosutils\rawrite2 -f rescue.bin -d a:
```

(waarin je dus de juiste filename moet invullen, en **a:** de letter van de gebruikte floppy drive is).

Als je al ergens UNIX of Linux draait, gebruik daar dan bijvoorbeeld

```
dd if=rescue.bin of=/dev/fd0 bs=512 conv=sync ; sync
```

(waar **/dev/fd0** de floppydrive aanduidt).

Wacht in beide gevallen tot het drivelampje uit is en de diskette gestopt is met draaien, voor je de diskette uit de drive haalt. (Je moet natuurlijk wel geformatteerde diskettes gebruiken; formatteren kan eventueel gewoon vanuit DOS/Windows.)

Voorzie de diskettes van het juiste label; er is een *rescue disk*, een *root disk*, diverse genummerde *driver disks* en *base system disks*. En om ze later weer te kunnen identificeren kun je er het beste ook “Debian 2.2” opschrijven.

Tip: als je nog een tweede computer beschikbaar hebt, kun je ook met slechts één of twee diskettes werken, door ze op de andere computer steeds opnieuw te beschrijven.

Harddisk
Een andere mogelijkheid is het gebruik van de harde schijf. Linux opstarten kan namelijk ook zonder de computer te rebooten. Het batch-bestand **install.bat** gebruikt het programma **loadlin.exe** om de kernel (het bestand **linux**) en de ramdisk (het bestand **root.bin**) in het geheugen te laden. Het Base System wordt uit één groot **.tar.gz** (= **.tgz**) archief gelezen.

De precieze bestanden voor elk onderdeel staan weer in de tabel. Zet alle benodigde bestanden in een aparte directory, bijvoorbeeld **C:\debian.22**. Let erop, dat de beschikbare **install.bat** bestanden uitgaan van dezelfde directory-structuur (met sub-directory’s) als op de CDROM of de FTP site; als je die indeling niet aanhoudt, moet je de gekopieerde **install.bat** even wijzigen met een tekst-editor (als DOS **edit** of Notepad/Kladblok). Merk op dat de betreffende bestanden op een DOS FAT16 of Windows FAT32 filesystem mogen staan.

Vergeet niet de naam te noteren van de directory waar je alles in hebt gezet (in paragraaf 3.1.6), deze heb je namelijk later nodig.

CDROM
Vanaf CDROM kan het ook en wel op een tweetal manieren. Het eenvoudigste is om van de CD op te starten (kan alleen met nieuwere computers). Zet daarvoor in de *BIOS Setup* (meestal te bereiken door de DEL-toets in te houden tijdens het opstarten) de *Boot Sequence* op (bijvoorbeeld) “CDROM, A, C”. Vergeet het niet terug te zetten als Debian eenmaal draait!

Als je niet direct van CDROM kunt opstarten, gebruik dan (net als bij de harde schijf) het batch-bestand **install.bat** direct uit de bovengenoemde directory (of één van de ‘smaak’-subdirectory’s).

Let op: deze methoden werken alleen als je een IDE (ATAPI) of SCSI CDROM drive hebt, die op hetzelfde datakabeltje aangesloten is als een harde schijf. Als je een (oude) CDROM drive hebt met z’n eigen interface-kaartje, of één die op de geluidskaart aangesloten moet worden, wordt het wat moeilijker. Zo’n drive heeft namelijk een aparte device driver nodig, en die is pas beschikbaar als de kernel en de device drivers naar het nieuwe root filesystem zijn gekopieerd. Oftewel: in zo’n geval moet je dus de kernel en de device drivers ergens anders vandaan halen: van diskettes of van de harde schijf (zie boven). Het Base System wordt pas gekopieerd (uit **base2_2.tgz**) als de drivers geladen zijn; dat kun je dus wel (als enige) van CD lezen met een dergelijke CDROM drive.

Netwerk
Voor het netwerk geldt hetzelfde als voor de niet-IDE/SCSI CDROM drive: je hebt een driver voor je netwerk(kaart) nodig en die is er pas als de device drivers zijn ‘gepasseerd’. Ook hier moet je dus alles behalve het Base System op diskettes of op de harde schijf hebben (zie boven). Op het netwerk kan wel het Base System aanwezig zijn, in het bestand **base2_2.tgz**. Dat bestand moet via HTTP beschikbaar zijn, dus op een normale web-server. Als je via een netwerkkaart (dus niet een modem) verbinding hebt met Internet, kan het Base System van één van de vele Debian FTP sites worden gehaald, mits die sites het archief ook per HTTP beschikbaar stellen. Zonodig kan ook een web-proxy

worden ingesteld. De communicatie gebeurt met het TCP/IP protocol; LanManager, Windows netwerk (SMB), Netware en Appletalk worden tijdens het installeren niet ondersteund (wel als Linux eenmaal ‘normaal’ draait). Automatische adres-toewijzing via DHCP/BOOTP werkt al wel tijdens de installatie.

Combinaties
Merk op dat verschillende methoden te combineren zijn. Je kunt bijvoorbeeld opstarten met de rescue en root diskettes, de kernel en device drivers van de Debian CDROM laden en het Base System via HTTP over het netwerk binnenhalen. Het gaat er dus om dat voor elk van de drie kolommen in de overzichtstabel één bijpassend medium wordt gekozen.

Ik prefereer zelf de volgende volgorde van mogelijkheden:
1) **install.bat** van IDE/SCSI CDROM en verder alles van CD halen (scheelt een reboot en het veranderen van de CMOS setup);
2) booten van IDE/SCSI CDROM en verder alles van CD halen;
3) alle benodigde bestanden (zie boven) kopiëren naar een DOS/Windows partitie op harde schijf en starten met **install.bat**;
4) of, als 1 t/m 3 niet werken, alles van floppy.

Verder blijkt het in de praktijk heel handig te zijn als je een rescue disk en een root disk bij de hand hebt, ook als het systeem al geïnstalleerd is en goed werkt. Daarom adviseer ik om in ieder geval zo’n setje te maken, ongeacht de installatiemethode die je kiest. En houd ook nog een lege (eventueel ongeformatteerde) diskette bij de hand, die in paragraaf 3.2.16 de ‘bootfloppy’ (opstartdiskette) van het nieuw geïnstalleerde systeem gaat worden.

3.1.6 Gegevens die nodig zijn

Tijdens de installatie zijn een aantal gegevens nodig. Je kunt ze hier noteren (liefst met potlood), zodat je niet teveel hoeft te zoeken.

- Bij installatie van harddisk: partitie met kernel/drivers/Base System (bijv. /dev/hda1): /dev/________

- Bij installatie van harddisk: directory met kernel/drivers/Base System: ______________________

- Harde schijf voor root partitie (bijv. /dev/hda, zie par. 2.5.3): /dev/_________

- Grootte van de root partitie (zie par. 3.1.2): _________ MB

- Harde schijf voor swap partitie (bijv. /dev/hda, zie par. 2.5.3): /dev/_________

- Grootte van de swap partitie (zie par. 3.1.2): _________ MB

- De naam (hostname) van je computer (zie par. 2.6.8): ______________________

Onderstaande alleen als je met een netwerkkaart aangesloten bent aan een lokaal netwerk (als er een DHCP/BOOTP-server aanwezig is, hoef je sommige instellingen wellicht niet op te geven):

- De domeinnaam (domain name) (zie par. 2.6.8): ____

- Het IP adres van je computer (zie par. 2.6.7): ________ . ________ . ________ . ________

- Het netmask (zie par. 2.6.7): ________ . ________ . ________ . ________

- Het broadcast address (zie par. 2.6.7): ________ . ________ . ________ . ________

- Het IP adres van de gateway/router (zie par. 2.6.7): ________ . ________ . ________ . ________

- Het IP adres van de nameserver (zie par. 2.6.8): ________ . ________ . ________ . ________

Instellingen van insteekkaarten enzovoort (niet nodig voor PCI kaarten):

- Netwerkkaart (voor zover van toepassing) IRQ: ______ I/O Adres: __________

- Niet-IDE/SCSI-CDROM (voor zover van toepassing) IRQ:______ I/O Adres:__________ DMA:______

Verder nog:
- Password voor root (zie paragraaf 2.7)
- Naam voor 'eerste' gewone gebruiker
- Password voor 'eerste' gewone gebruiker

NOTEER DEZE ***NIET*** HIER maar eventueel op aparte papiertjes!

3.2 Installatie - Deel 1

Als je ruimte voor de twee Linux partities hebt, als je gekozen hebt voor een installatiemethode en eventueel diskettes hebt geschreven, en als je alle benodigde gegevens bij de hand hebt, dan kun je beginnen met het installeren van Debian GNU/Linux 2.2. In paragraaf 3.1.4 heb je de globale procedure gelezen.

3.2.1 Starten van de installatie

Lees vóór je met de installatie gaat beginnen, eerst de rest van dit hoofdstuk, én de officiële installatiehandleiding, het bestand **`/install/doc/install.en.html`** of **`.txt`** op de Binary-1 CDROM of **`/debian/dists/potato/main/disks-i386/current/doc/install.en.html`** of **`.txt`** op een Debian FTP site. Ga dan pas echt aan de slag.

Opstarten van diskette
Sluit alle draaiende programma's af, doe de rescue disk in de floppy drive en start het systeem opnieuw op.

Starten met install.bat van harddisk of CDROM
Sluit eventueel Windows af en keer terug naar 'kaal' DOS. Vergewis je ervan dat alle diskcache programma's afgesloten zijn. Type voor de zekerheid **`smartdrv /c`** en **`ncache2 /reset`** (negeer foutmeldingen).
Als je **`install.bat`** van CDROM gebruikt, ga je naar de juiste directory, bijvoorbeeld **`D:\dists\potato\main\disks-i3\2_2.16-`** (waarin je voor **`D:`** de drive letter van je CDROM drive in moet vullen -- let op: die kan veranderd zijn als je ge-fips-t hebt!).
Als je **`install.bat`** van de harde schijf gebruikt, ga je naar de directory toe, waar je de installatiebestanden in hebt gezet.
Wacht nog tien seconden en type dan het commando **`install`**.

Opstarten van CDROM
Sluit alle draaiende programma's af, doe de Debian 2.2 Binary-1 (of eventueel -2) CDROM in de CDROM drive en start de computer opnieuw op. Druk tijdens het geheugentesten op DEL en verander de 'Boot Sequence' (onder 'BIOS Features') zodanig, dat CDROM als eerste staat. 'Save' de settings in het CMOS en reboot.

3.2.2 Lezen van informatie, keuze van bootmogelijkheden

Opgestart van diskette of CDROM
Er komt een introductie op het scherm. Gebruik eventueel de toetsen F1 t/m F10 om meer informatie te lezen. Geef (achter de "boot:" prompt, waar de cursor altijd staat) Enter om verder te gaan.

Let echter op: IBM ThinkPad gebruikers moeten geen Enter geven, maar typen "linux floppy=thinkpad". En als je een monochrome monitor hebt, type dan "linux mono". Als je de 'compact' of 'idepci' smaak gebruikt en de 'grafische tekstmode' niet blijkt te werken (of als je die niet mooi vindt), type dan "linux video=vc:8". Hier is "linux" steeds de te-laden kernel; de rest zijn opties die eventueel gecombineerd kunnen worden. Deze opties kun/moet je steeds meegeven als je Linux opstart, zie paragraaf 13.3 voor meer informatie hierover.

Gestart met install.bat van harddisk of CDROM
Niets. Door naar de volgende stap.

3.2.3 Laden van Linux kernel en ramdisk image, starten van Linux

De Linux kernel en het ramdisk image worden ingelezen: 'Loading...' Dat kan een tijdje in beslag nemen. Daarna verschijnt de melding 'Uncompressing Linux...' Vrij snel daarna komt er een hele berg tekst langs. Het kan gebeuren dat het systeem even 'stilstaat' tijdens het testen van verschillende soorten randapparaten.

Als je te weinig geheugen (minder dan 12 MB) hebt voor de installatie, word je daarvan op de hoogte gesteld, of krijg je een 'out of memory' melding, waarna de computer niets meer doet. Je kunt dan op dat systeem Debian niet zomaar *installeren* (maar wellicht wel draaien na de installatie). Voor de installatie kun je misschien wat geheugen lenen uit een ander systeem, of tijdelijk de harde schijf in een andere computer hangen en daarop installeren. Dat is geen probleem, omdat de installatie "hardware-ongevoelig" is.

Als alles okee is, wordt de ramdisk geunzipt en wordt het installatieprogramma gestart.

Opgestart van diskette
De ramdisk moet geladen worden van de root disk. Als daarom gevraagd wordt, even na het starten van de Linux kernel, doe je die diskette in de drive en druk je op Enter.

3.2.4 Welkom!

Na het starten van de kernel komt een welkomst-tekst in beeld. Lees het door (gebruik pijltjestoetsen op/neer) en druk als je klaar bent op Enter.

Let op: Linux heeft een ingebouwde screen-saver, die het scherm na 10 minuten op zwart zet. Als dat gebeurt moet je *geen letter/cijfer/spatie/Enter-toets* indrukken, want die wordt gewoon als invoer behandeld. Gebruik liever de Shift-, Ctrl- of Alt-toets.

3.2.5 Het hoofdmenu

Voordat je in het installatie-hoofdmenu terecht komt, kun je heel even een 'tussenscherm' zien: "The installation program is determining the current state of your system". Dan wordt nagegaan wat je al hebt gedaan en wordt bekeken wat de volgende stap moet zijn. De eerstgenoemde opties in het hoofdmenu zijn namelijk afhankelijk van de vorderingen in het installatieproces. De bovenste optie is altijd 'Next': als je die kiest, vervolg je de installatie 'zoals het zou moeten'.

3.2.6 Configureren van het toetsenbord

Naast 'Next' staat nu 'Configure the Keyboard'. Kies deze optie door op Enter te drukken.

Uit de lijst van beschikbare toetsenborden moet de juiste (of beste) gekozen worden. Loop op en neer met de pijltjestoetsen en druk op Enter als je op de juiste *layout* staat. Waarschijnlijk is dat de bovenste, 'U.S. English (QWERTY)'.

Na het selecteren van het toetsenbord kom je weer in het hoofdmenu terecht.

3.2.7 Partitioneren van de harde schijf

Als je nog geen backups hebt gemaakt, haal dan diskettes en CDROMs uit de drives, reset de computer en maak backups van alle harde schijven. Het volgende kan namelijk flink misgaan. Laatste kans!
(Persoonlijk doe ik dat nooit. Maar ja, wie ben ik...)

Je moet nu twee partities gaan maken. Eén voor het root filesystem, en een swap partitie. In paragraaf 3.1.6 heb je alle gegevens bij elkaar gezet, gebruik ze hier. (Als je al een keer eerder geïnstalleerd had, bestaan de benodigde partities al, en wordt deze stap automatisch overgeslagen.)

Kies in het hoofdmenu voor 'Next : Partition a Hard Disk'. Je moet dan kiezen welke harddisk je wilt partitioneren. Kies die harddisk waar de partitie met het root filesystem op moet komen. Lees vervolgens de informatie over grote harddisks, die waarschijnlijk niet echt relevant is. Na Enter kom je in het programma **`cfdisk`**, waarmee je partities kunt aanmaken en/of verwijderen.

NB: Als je zelf al verstand hebt van partitioneren, hoef je je niet persé aan onderstaande te houden (af en toe is het motto: "wees inventief").

Als je met een nieuwe harde schijf bezig bent, of als je een stuk ongepartitioneerde ruimte hebt, zie je een regeltje met onder 'FS Type' (filesystem type) de aanduiding 'Free Space'. Als je net gefipst hebt, staan er twee 'DOS FATxx' regels onder elkaar. De laatste is het stuk dat er 'afgeknipt' is. Controleer de grootte!! Dat afgeknipte stuk moet geen DOS blijven, ga daarom met de pijltjestoetsen op/neer op die regel staan, kies met pijltjes rechts/links '[Delete]' en druk op Enter. De regel wordt nu vervangen door 'Free Space'.

Ga op de regel 'Free Space' staan, kies '[New]' en druk op Enter. Kies nu '[Logical]' (als dat gevraagd wordt) en druk op Enter. Geef de gewenste grootte van het root filesystem op (in MB), gevolgd door Enter. Kies (als dat gevraagd wordt) '[Beginning]' en druk op Enter. Je ziet dat de lege regel wordt vervangen door één met type 'Linux'. Dat wordt je root partitie. Merk op dat de grootte iets kan afwijken, omdat die deelbaar moet zijn door een bepaald schijf-afhankelijk getal.

Als de swap partitie op dezelfde schijf moet komen, kun je die gelijk maken. Ga weer op de 'Free Space' staan, kies '[New]' en dan weer '[Logical]'. Voor de grootte kun je waarschijnlijk op Enter drukken, dat betekent 'de rest', dan is Beginning of End ook geen vraag meer. Het type staat nu ook op 'Linux', maar dat is niet de bedoeling. Blijf op de swap-regel staan en kies '[Type]'. Het type moet 82 ('Linux Swap') zijn.

Als je tevreden bent, kijk je alles nog eens grondig na. Kies dan '[Write]'. Kijk alles voor de laatste keer na. Als er iets niet klopt, druk je op Escape. Als alles OK is, typ je "yes" (voluit), gevolgd door Enter. De partitietabel wordt nu naar de harde schijf geschreven. Als je een waarschuwing krijgt, is dat niet erg. Kies '[Quit]' om terug te keren naar het hoofdmenu.

Als de swap partitie op een andere schijf moet komen dan de root partitie, moet je weer voor 'Partition a harddisk' kiezen in het hoofdmenu, dit hoeft geen 'Next' meer te zijn. Kies de gewenste schijf en herhaal de bovenstaande procedure. Let erop dat het type 'Linux Swap' (nummer 82) moet zijn.

3.2.8 Swap partitie activeren

Bij 'Next' staat nu dat de swap partitie geïnitialiseerd en geactiveerd moet worden. Na Enter moet de juiste partitie gekozen worden (waarschijnlijk is er maar één keus en dat is de goede). Kies er voor om de slechte-blokken scan *niet* over te slaan, 'No' dus. Kies tenslotte het juiste antwoord op de vraag of je het wel zeker weet.

Let op bij het maken van 'Yes'/'No' keuzes: de 'knop' met de *blauwe* kleur is geselecteerd, en daar staat ook de cursor bij. Je kunt kiezen met de pijltjestoetsen links/rechts of de TAB toets, en bevestigen met Enter.

Het initialiseren en checken van de swap partitie kan enige tijd in beslag nemen. Er is geen voortgangs-rapportage te zien, maar het harddisk-ledje is wel aan. Daarna kom je weer in het hoofdmenu terecht.

3.2.9 Linux partitie initialiseren

De volgende stap is het initialiseren van het root filesystem (de Linux partitie), in DOS-termen 'formatteren'. Als deze optie wordt geselecteerd, wordt gevraagd welke partitie geïnitialiseerd moet worden. Let op: selecteer *niet* een partitie waar al een filesystem op staat, want dat zal dan worden verwijderd!! Selecteer alleen die partitie die je zojuist met **`cfdisk`** hebt aangemaakt (in paragraaf 3.2.7). Bij de vraag om compatibiliteit met 2.0 kernels te behouden kun je het beste 'Yes' kiezen. Kies er weer *niet* voor om de bad-block scan over te slaan, dus 'No'. Geef tenslotte het juiste antwoord op de vraag of je het zeker weet.

Bij grote schijven/partities kan het checken enige tientallen minuten in beslag nemen. Na afloop wordt gevraagd of de zojuist geïnitialiseerde (geformatteerde) partitie als root filesystem gemount moet worden. Kies daar 'yes' (zie ook paragrafen 2.5 en 3.1.2).

Als je de installatie had afgebroken en opnieuw gestart, heb je de root partitie wellicht al geïnitialiseerd. Dan is opnieuw initialiseren natuurlijk niet gewenst. Kies dan voor het 'Alternatief' om de root partitie alleen te mounten. Als je al eens eerder had geïnstalleerd en je wilt alles helemaal opnieuw doen, dan moet je hier *wel* initialiseren, want daarmee verdwijnt de oude inhoud van de partitie.

Als je meerdere partities moet initialiseren en/of mounten (bijvoorbeeld voor **`/usr`** of **`/var`**), moet dat gebeuren direct na het initialiseren/mounten van de root partitie. Daarvoor zijn dan 'Alternates' voor beschikbaar. Je kunt de optie 'View the Partition Table', onderin het hoofdmenu, gebruiken om een overzicht van de partities te krijgen.

3.2.10 Installeren van de Kernel en Modules (device drivers)

Bij 'Next' staat nu dat de kernel en modules (dat zijn de device drivers) geïnstalleerd moeten worden. Dat betekent dat ze van bijvoorbeeld floppy's of de harde schijf worden gekopieerd naar het zojuist aangemaakte root filesystem. Als deze optie wordt gekozen, moet worden aangegeven waar één en ander vandaan moet worden gehaald.
(Als je de installatie had afgebroken en opnieuw gestart, heb je deze stap wellicht al gehad. Hij wordt dan automatisch overgeslagen.)

Kernel/drivers op diskette
Kies na Enter voor de eerste of tweede floppy drive. De rescue disk (opstart-diskette) moet in de drive gedaan worden, maar waarschijnlijk zit die er nog in. Controleer het toch even. Als de kernel is gekopieerd, zijn de device drivers aan de beurt -- die staan op de driver disks, die je dus in de floppy drive zult moeten doen als daarom gevraagd wordt.

Sommige oudere computers hebben een floppy drive die niet doorgeeft dat de diskette is gewisseld. Als je daar problemen mee hebt, kan het zinvol zijn de gevraagde diskette te laten lezen zonder dat die in de drive zit.

Kernel/drivers op harddisk
Kies voor 'harddisk'. Er moet vervolgens worden aangegeven op welke schijf/partitie de kernel en drivers zich bevinden, waarschijnlijk op een DOS partitie, zie paragraaf 3.1.6. Vervolgens moet je de directory opgeven, waar de kernel in staat ('Archive path'). Let erop, dat je overal slashes ('**`/`**') moet gebruiken in plaats van backslashes ('**`\`**'). Bijvoorbeeld: als je de directory **`C:\debian.22`** gebruikt, dan moet je **`/debian.22`** opgeven. Kies in het volgende venster voor 'list'. Je krijgt dan de keuze uit één optie en dat is de directory die je net hebt opgegeven (met **`/instmnt`** ervoor); dat is goed, dus druk op Enter. Vervolgens worden de kernel en de device drivers van de harde schijf gelezen en geïnstalleerd.

Kernel/drivers op CDROM
Kies voor 'cdrom'. Als je meerdere CDROM drives hebt, krijg je een lijstje waar je de juiste van moet kiezen (voor de duidelijkheid: first drive = master, second drive = slave). Na Enter wordt je gevraagd de Debian Binary-1 CDROM in de drive te doen; doe dat en geef Enter. Vervolgens moet je het pad aangeven waar het Debian archief zich bevindt. Standaard staat daar al **`/instmnt`** en dat is juist; druk dus op Enter. Kies in het volgende venster voor 'default'. Daarna worden zowel het kernel image als de drivers van de CD gelezen en geïnstalleerd.

3.2.11 Configureren van de Device Drivers

Na de kernel en de device drivers, zou het Base System op de nieuwe partitie moeten worden gezet. Omdat het Base System bijvoorbeeld ook 'over het netwerk' kan worden gehaald, moet dat netwerk wel eerst geconfigureerd worden. De eerste stap daarvoor is het laden van de device driver voor de betreffende netwerkkaart. En als we dan toch bezig zijn, waarom dan niet gelijk alle device drivers langs lopen. Daarom staat bij 'Next' dat 'de' (lees 'alle') device drivers geconfigureerd moeten (lees 'kunnen') worden.

PCMCIA drivers
Als je PCMCIA kaarten ('PC-card', creditcard-formaat, bij de meeste laptops) wilt gaan gebruiken, moet je *eerst* voor 'Alternate : Configure PCMCIA Support' kiezen. Eerst moet je dan (met de spatiebalk) opgeven welke dingen je wilt gaan configureren -- meestal alleen de PCMCIA controller. Als je de juiste optie(s) geselecteerd hebt, ga je met de TAB-toets naar <Ok> en druk je op Enter. Je wordt dan gevraagd welke PCMCIA controller in je laptop zit, meestal is dat **`i82365`**. Bij de daaropvolgende vragen hoeft meestal niets te worden ingetypt (maar lees wel de uitleg).

In een enkel geval zou het kunnen dat PCMCIA tijdens de installatie nog niet werkt. Als het systeem in paragraaf 3.3 opnieuw is opgestart, zou PCMCIA in ieder geval wél moeten werken.

'Gewone' device drivers
Als je Enter hebt gegeven op 'Configure Device Driver Modules', komt er een lijstje op het scherm met categorieën. Onder die categorieën zitten weer hele lijsten met de echte device drivers. Je kunt ze allemaal langs lopen, maar de meesten zul je niet direct begrijpen -- doe er dan ook niets mee. De volgende drivers zijn misschien interessant (maar niet gaan 'spelen' met drivers voor dingen die je niet hebt!):

cdrom/.... - als je een niet-IDE/SCSI CDROM hebt, moet je hier de juiste uitkiezen. Hebben vaak I/O adressen en/of IRQ's nodig, die je in paragraaf 3.1.6 hebt genoteerd. NB: Noteer ergens de naam die helemaal links staat, dat is ook de **`/dev`** naam, en die heb je straks nodig (bijv. **`/dev/sbpcd`**).

fs/.... - als je bijvoorbeeld een HPFS of NTFS partitie hebt, kun je hier de driver ervoor laden.

misc/serial en **misc/lp** - seriële en parallelle poorten (hebben beide *geen* parameters nodig); worden automatisch geladen, je hoeft ze hier dus *niet* te installeren.
misc/.... - diverse merken 'abnormale' muizen ('normale' = seriële muizen hebben *geen* driver nodig). NB: Noteer ergens de naam die helemaal links staat -- dat is het muis device. De PS/2 muis (**`/dev/psaux`**) is meestal reeds in de kernel 'gebakken', daar is dus geen aparte driver voor nodig.
misc/.... - vele geluidskaarten en aanverwante zaken; ik raad je aan om die *niet nu* te installeren, dat kan altijd later nog en ze zijn op dit moment totaal niet nodig.
net/.... - als je een (gewone, dus niet PCMCIA) netwerkkaart hebt, moet je hier de juiste uitkiezen (*ook* als je het Base System niet van het netwerk wilt halen), bijvoorbeeld 'ne' of 'ne2k-pci' voor NE2000-compatible ISA resp. PCI kaarten. Niet-PCI kaarten hebben meestal I/O adressen en/of IRQ's nodig, die je in paragraaf 3.1.6 hebt genoteerd.

Als je Enter geeft op een bepaalde driver, zal worden gevraagd of die geïnstalleerd moet worden (of verwijderd, als die al geïnstalleerd was). Er komt een venster op het scherm met wat informatie over de driver, en een invoerveld. In dat veld kunnen opties worden meegegeven. Welke dat kunnen zijn, staat erboven aangegeven, vaak inclusief een voorbeeld op de bovenste regel. Soms hoef je geen opties (parameters / 'command-line arguments') op te geven als je de 'default' instellingen prima vindt (of als er helemaal geen parameters zijn natuurlijk). Let erop, dat de meeste parameters (zoals I/O adressen) in hexadecimale vorm kunnen worden opgegeven. Ik heb voor mijn NE2000-ethernetkaart de volgende opties: "**`io=0x280 irq=12`**", ofwel I/O adres 280 (hexadecimaal, = 640 decimaal) en IRQ 12 (decimaal, = 0x0C hexadecimaal). Een enkele keer staat er iets als 'xyz.o must be loaded manually'. Dan moet je dus allereerst op zoek gaan naar die driver 'xyz' en daar een 'Install'-opdracht voor geven. Dit gebeurt echter niet vaak, want als een driver een andere nodig heeft, wordt die meestal volautomatisch geladen.

Als je de juiste (of geen) opties/parameters hebt ingetypt, geef je Enter. (Ga in geval van vergissing met TAB naar 'Cancel' en druk op Enter.) De resultaten van het laden van de driver worden op het scherm getoond. Als er problemen zijn, heb je waarschijnlijk óf de verkeerde driver geselecteerd, óf onjuiste parameters opgegeven. Het kan gebeuren dat het systeem lijkt te 'hangen' (minstens 2 minuten wachten!); druk in zo'n geval op LinkseAlt-F2, geef Enter, type 'reboot', gevolgd door Enter, en begin opnieuw met de installatie. Als een driver weigert te installeren, sla die dan (voorlopig) over; als Debian eenmaal draait, kun je alles alsnog installeren.

Als je met een categorie, of met het geheel, klaar bent, gebruik je het item 'Exit' (bovenin) om terug te springen naar het vorige niveau.

Het instellen (of 'bijstellen') van de device drivers kan ook na de installatie nog gebeuren, met het programma **`/usr/sbin/modconf`**. De instellingen worden bijgehouden in de bestanden **`/etc/modules`** en **`/etc/modutils/modconf`**; na verandering van die laatste moet altijd **`update-modules`** worden gedraaid.

3.2.12 Netwerkinstellingen

De volgende stap is het configureren van het (lokale) netwerk, als je dat hebt. Dit is niet veel meer dan het invullen van de gegevens uit paragraaf 3.1.6. NB: bij de eerste vraag om de hostname moet je het domein er nog *niet* achter typen, dat komt later -- of helemaal niet, als je geen lokaal netwerk hebt.

Als je in de vorige stap één of meer netwerkkaart-drivers hebt geïnstalleerd, wordt hier per stuk gevraagd om de juiste instellingen. Als er om een 'DNS server' wordt gevraagd, wordt daarmee de nameserver bedoeld.

De instellingen die je hier opgeeft worden bijgehouden in het bestand **`/etc/network/interfaces`** en **`/etc/resolv.conf`** (en voor een PCMCIA netwerkkaart in **`/etc/pcmcia/network.opts`**).

3.2.13 Installeren van het Base System

Dan is het nu zover dat het Base System kan worden geïnstalleerd. Dat wil zeggen dat het (bijvoorbeeld van diskettes) wordt gekopieerd en 'uitgepakt'.

Base System van diskettes
Kies voor de eerste of tweede floppy drive. Er wordt achtereenvolgens gevraagd om de elf (of dertien) diskettes 'base 1', 'base 2', enz. Als deze diskettes zijn gelezen, wordt het Base System uitgepakt en op de root partitie gezet. Dit kan een aantal minuten in beslag nemen. Mochten er fouten in een diskette zitten, dan moet die diskette opnieuw worden geschreven. Als dat nodig is, kun je de installatieprocedure afbreken met de optie 'Reboot the system', onderin het hoofdmenu -- dus *NIET* zomaar resetten!

Base System van harddisk
Kies 'harddisk'. Kies vervolgens de juiste partitie en geef de naam van de directory waarin de installatiebestanden staan (heb je genoteerd in paragraaf 3.1.6). Als die directory bijvoorbeeld **`C:\debian.22`** is, moet je **`/debian.22`** opgeven (let op: '**`/`**' in plaats van '**`\`**'). Kies in het volgende venster voor 'list' en vervolgens de enige optie die je gepresenteerd krijgt (de opgegeven directory met **`/instmnt`** ervoor). Daarna wordt het Base System van de harde schijf gelezen en geïnstalleerd.

Base System van CDROM
Kies 'cdrom' en vervolgens, als dat gevraagd wordt, het juiste apparaat (waarbij first drive = master en second drive = slave). Als je een niet-IDE/SCSI drive hebt, moet je waarschijnlijk voor 'proprietary' kiezen, maar dat heb ik zelf niet kunnen testen.
Er wordt gevraagd de Debian Binary-1 CDROM in de drive te doen; doe dat (als het nog niet gebeurd is) en druk op Enter. Het 'archive path' staat al op **`/instmnt`**; dat is goed, druk dus op Enter. Kies tenslotte 'default'. De installatie wordt dan gestart en je komt na enige tijd weer in het hoofdmenu terecht.

Base System via HTTP van een netwerkserver
Het netwerk is geconfigureerd en dus toegankelijk. Kies 'network'. In het venster vul je nu de locatie in van het bestand **`base2_2.tgz`**, behalve die naam zelf. De voor-ingestelde locatie is Debian's hoofd-server in de VS. Als je een web-proxy moet gebruiken, kun je die hier ook instellen, spring tussen de velden met de TAB-toets. (NB: die web-proxy instelling wordt alleen voor deze stap gebruikt, en daarna weer vergeten.) Wanneer de instellingen naar wens zijn, loop je met de TAB-toets naar "<Ok>" en druk je op Enter. Het Base System (ongeveer 15 MB) wordt vervolgens opgehaald en geïnstalleerd.

3.2.14 Configureren van het Base System - de klok

Als het Base System is gekopieerd, zijn er nog wat instellingen nodig. Kies dus voor het configureren van het Base System. Dat bestaat uit het ingeven van de instellingen voor de klok.

Als eerste moet de tijdzone worden geselecteerd, waarin de computer zich bevindt. Loop met de pijltjestoetsen op/neer naar 'Europe' en druk op Enter. Voor Nederland kies je vervolgens (aan de rechtse kant) 'Amsterdam', en voor België 'Brussels'.

Vervolgens wordt gevraagd of de systeemklok op *GMT* staat (*Greenwich Mean Time = Universal Time*) of op de lokale tijd. Als je uitsluitend Linux draait, kun je het beste voor GMT kiezen ('yes' dus), anders voor Local Time ('no'). Meer informatie over de tijd en de klok staat in paragraaf 13.1.

Na het installeren van Debian is de tijdzone eventueel in te stellen met het programma **`/usr/sbin/tzconfig`**.

3.2.15 Linux direct van de harde schijf laten opstarten

Bij 'Next' staat nu dat Linux direct *bootable* gemaakt kan worden. Dat houdt in dat Linux direct kan worden opgestart als de computer wordt aangezet. Kies altijd voor deze optie, ook als je voorlopig niet van de harddisk wilt/kunt opstarten, er wordt hiermee namelijk een simpele LILO-configuratie gemaakt, die je in paragraaf 5.5 goed kunt gebruiken. In de vervolg-vragen kun je aangeven of je nu al direct Linux wilt laten opstarten.

Als eerste wordt gevraagd waar LILO moet worden geïnstalleerd, op de 'master boot record' van de harddisk of op de boot-sector van de Linux root partitie. In praktisch alle gevallen is dat laatste de beste keuze, zeker als je nog een ander OS op je computer hebt staan dat je wilt kunnen blijven opstarten.

Als de Linux root partitie een logische partitie is, wordt gevraagd om een alternatieve partitie, bijvoorbeeld **`/dev/hda2`**. Dat is de bijbehorende uitgebreide (extended) partitie, zie ook paragraaf 2.5.1.

De volgende vraag gaat over het maken van een Master Boot Record. Als je OS/2 draait, of een ander OS met een *boot manager*, kies je hier 'No', anders 'Yes' (ook als je niet weet wat een boot manager is). Er volgt een scherm met informatie die ik hieronder zal toelichten.

Als laatste kun je aangeven of je standaard Linux wilt laten opstarten ('default boot partition'). Als je Linux altijd direct wilt laten opstarten, kun je hier 'yes' kiezen; met 'no' blijft het opstarten zoals je gewend bent. Merk op, dat je de 'default boot partition' = 'actieve partitie' altijd kunt wijzigen, met bijvoorbeeld **`cfdisk`** of **`activate`** in Linux (zie ook paragraaf 5.5) of **`fdisk`** in DOS/Windows.

Als je met de tweede vraag de simpele Linux boot manager had geïnstalleerd, zie je tijdens het opstarten de aanduiding 'MBR' voorbij komen. Als je tijdens het opstarten de Shift ingedrukt houdt (begin daarmee tijdens de geheugentest), komt daar iets als '2FA:' achter te staan. Je kunt dan de weergegeven letters/cijfers kiezen. '2' staat dan voor het opstarten van hda2 ("standaard", de 'actieve' partitie), 'F' voor opstarten van een floppy en 'A' voor 'advanced'. Met de 'A' krijg je weer een andere keuze, namelijk '1234F', waar '1', '2', '3' en '4' voor hda1 t/m hda4 staan (je ziet nu dus alle partities op hda).

Met deze simpele boot manager kun je dus kiezen van welke parititie je wilt opstarten. Als je net LILO hebt laten installeren op **`/dev/hda2`**, en 'yes' hebt gekozen bij de laatste vraag, zal hda2 actief gemaakt zijn, en zal Linux dus direct opstarten. Als DOS op hda1 staat, kun je (Shift inhouden tijdens opstarten) met 'A' en '1' alsnog in DOS terecht komen. Maar als je 'no' had gekozen bij de laatste vraag, zal DOS nog steeds opstarten van hda1, en kun je met 'A' en '2' in Linux terecht komen. Met deze boot-manager kun je dus altijd alles opstarten, alleen moet je voor de niet-standaard opties even de Shift ingedrukt houden.

Zie paragraaf 5.5 voor meer informatie over manieren om Linux op te starten. Als je van de simpele boot manager af wilt, kun je in DOS het commando **`fdisk /mbr`** geven.

3.2.16 Bootfloppy maken

Ik adviseer om *altijd* een bootfloppy (opstartdiskette) te laten maken. In ieder geval als je niet hebt gekozen voor het direct opstarten (vorige optie), want dan is een bootfloppy de enige manier om weer in Linux te komen. Maar als het opstartbaar maken fout is gegaan, of als het automatisch starten niet werkt, heb je ook een bootfloppy nodig.

Selecteer daarom 'Make a Boot Floppy' en doe een, eventueel ongeformatteerde, 3½" (1.44MB) of 5¼" (1.2 MB) diskette in de eerste diskdrive. Het installatieprogramma zal de diskette formatteren en opstartbaar maken. Vergeet niet deze diskette 'Custom Boot' te labelen en hem tegen schrijven te beveiligen (*write protect*).

Ik heb zelf één keer een floppy gehad, die niet geformatteerd wilde worden. Een nieuwe diskette van een ander merk loste dat probleem op.

3.3 Installatie - Deel 2

Het eerste deel van de installatie, dat gebruik maakte van de boot- en rootfloppy's, is voltooid. Het wordt tijd om Linux eens 'echt' te gaan starten, vanaf het root filesystem dat zojuist is geïnstalleerd.

3.3.1 Het systeem rebooten

In het installatie-hoofdmenu is de volgende stap is het opnieuw opstarten (*rebooten*) van het systeem. Kies daar dan ook voor.

De bedoeling is dat je na de reboot op de één of andere manier weer in Linux terecht komt. Als je had gekozen om Linux direct te laten opstarten van de harde schijf, zal dat vanzelf gebeuren. Anders moet je vóór het rebooten de Custom Boot floppy in de diskdrive doen. Als er nog een CDROM in de drive zat, haal die er dan eerst uit.

Let op: als je zojuist van CDROM bent opgestart, moet je de Boot Sequence weer terug zetten zoals die daarvoor stond. Druk tijdens het geheugentesten weer op DEL, zet de juiste instellingen terug en kies voor 'Save and Exit'.

Als alles goed gaat, zal Linux opnieuw opstarten, ofwel van de harde schijf, danwel van de Custom Boot floppy. Bij de Custom Boot floppy mag je Enter geven bij de 'boot:' prompt, je kunt ook een paar seconden wachten. Als het opstarten van de harde schijf niet werkt, reboot dan nog een keer (bijvoorbeeld met de reset-knop op de systeemkast) en doe de Custom Boot floppy in de diskdrive; booten van die diskette zou altijd moeten werken.

Net als bij de eerste keer, zul je een veelheid van meldingen over het scherm zien vliegen. Zodra dat gebeurt, mag je de Custom Boot floppy (als je die had gebruikt) uit de drive halen, die is pas weer nodig bij de volgende keer opstarten. Vervolgens worden er weer een aantal instellingen gevraagd, op dezelfde manier als tijdens het eerste deel van de installatie.

Als je linux op geen enkele 'normale' manier in kunt komen, kun je ook weer van de rescue floppy of CDROM opstarten, en dan geen Enter bij de "boot:" prompt, maar "rescue root=/dev/hdXX ro" (en voor hdXX je root partitie invullen).

3.3.2 MD5 encryptie voor passwords

Passwords kunnen 'normaal' of 'extra' ge-encrypt ('versleuteld') worden opgeslagen. De veiligste methode heet 'MD5' encryptie, maar die is niet geheel compatible met een aantal netwerk-toepassingen. Bij de vraag om 'md5 passwords' kies je dus 'Yes' als je geen lokaal netwerk hebt, en (vaak) 'No' als je wel een lokaal (bedrijfs-)netwerk hebt.

3.3.3 Shadow passwords

Vervolgens wordt gevraagd of je *shadow passwords* wilt gebruiken. Kies daar altijd 'Yes', dat is namelijk goed voor de veiligheid. Als het systeem draait kun je (als root) shadow passwords aan- en uitzetten met de commando's **`shadowconfig on`** en **`shadowconfig off`**.

3.3.4 Het root password

Na een introductie-scherm over het root-password moet je jouw eigen, zelf verzonnen, onkraakbare, uiterst geheime tekenreeks intypen; zie paragraaf 2.7, en misschien heb je 'm in paragraaf 3.1.6 wel op een apart briefje genoteerd. En nog een keer, zodat typefouten vermeden worden. Merk op dat ingetypte letters op geen enkele manier in beeld verschijnen.

3.3.5 Het eerste gebruikersaccount

Dan moet je ook een eerste gebruikersaccount aanmaken, ofwel je moet jezelf "een login geven" (kies dus 'Yes' bij de vraag 'create a normal user account'). Je hebt in paragraaf 3.1.6 een leuke username verzonnen (of gewoon je voornaam) en een password dat redelijk gemakkelijk te onthouden is. Bij de vraag om de 'Full Name' kun je je volledige naam intypen. Tijdens het intypen van het password is wederom niets op het scherm te zien.

Dit gebruikersaccount gebruik je om al je werk te doen. Het root-account gebruik je alleenmaar voor die dingen die je als gewone gebruiker niet mag. Veel Linux-gebruikers vinden het prettig dat ze als gewone gebruiker niet zo erg hoeven op te letten en alles kunnen doen zonder het systeem in gevaar te brengen (hoogstens hun eigen bestanden). Let op: dat geldt dus *niet* voor root!

Je kunt (als root) meer nieuwe gebruikers aanmaken met **`adduser <nieuweusernaam>`**, zie paragraaf 5.4 en 13.2.

3.3.6 Opruimingswerkzaamheden

Als je geen PCMCIA gebruikt, kun je 'Yes' antwoorden op de vraag of de ondersteuning daarvoor verwijderd mag worden. NB: de melding "het schijnt dat je geen PCMCIA gebruikt" kan foutief zijn! Als je wel PCMCIA nodig hebt, moeten de 'pcmcia' pakketjes natuurlijk *niet* verwijderd worden.

3.3.7 PPP connectie opzetten

Het kan gebeuren dat wordt gevraagd of je een PPP-verbinding naar een Internet Service Provider (ISP) wilt opzetten. Daarover zal dan het hele systeem worden geïnstalleerd. Als je geen Debian Binary-1 CDROM hebt, én een heleboel geld wilt uitgeven, is dit misschien een optie. Maar aangezien er nogal veel (50 tot 200 MB) moet/kan worden opgehaald, lijkt deze optie me niet erg zinvol. Ik kies daar dus altijd 'No'.

3.3.8 Opgeven van installatie-bronnen

Dan verschijnt een keuze-menu met mogelijkheden voor het verkrijgen van de te-installeren software.

Als je één of meer Debian CDROM's wilt gebruiken, doe er dan één in de drive (volgorde maakt niet uit), wacht tot de drive de CD herkend heeft en kies dan 'cdrom' gevolgd door Enter. De 'Package Indexes' worden gelezen. Daarna wordt gevraagd om de volgende CD; als je er meerdere hebt, haal dan de eerste CD uit de drive, doe de volgende erin, wacht tot de drive de CD herkend heeft, kies dan voor 'Yes' en druk op Enter. Herhaal dat voor alle Debian CD's die je hebt, Source CD's mogen ook maar hoeven niet persé. Kies 'No' na het scannen van de laatste CD. Met 'Yes' kun je eventueel nog meer bronnen toevoegen (ook andere CD's als je per ongeluk 'No' had gekozen).

Als je tijdens de installatie nog geen CD's had gebruikt, kan het zijn dat de juiste CDROM drive niet automatisch gevonden wordt. Er wordt dan gevraagd om de juiste device naam; er staat **`/dev/cdrom`** maar dat werkt blijkbaar niet. De meest gangbare devices zijn:

/dev/hda, **/dev/hdb**, **/dev/hdc**, **/dev/hdd** - IDE/ATAPI harddisks of CDROM drives
/dev/scd0 - eerste SCSI CDROM drive
/dev/sbpcd en dergelijke - “andere” CDROM drives waarvoor je eerder een device driver hebt geïnstalleerd.

Je kunt de software ook automatisch van het Internet laten downloaden. Kies dan ‘http’ of ‘ftp’. Meestal zal het antwoord ‘Yes’ zijn op de vragen of je ‘non-US’, ‘non-free’ en ‘contrib’ software wilt kunnen installeren. Uit de lijst met landen kun je vervolgens de meest geschikte kiezen; voor Europa zijn ‘Finland’ en ‘Sweden’ meestal het snelst, maar ‘United States’ is ook een goede optie. Als je een web-proxy nodig hebt voor een buiten-verbinding, kun je die (alleen bij ‘http’) in het volgende venster opgeven, bijvoorbeeld “http://proxy.provider.nl:8080/”. Hierna wordt deze configuratie direct geprobeerd, wat natuurlijk zal mislukken als het netwerk niet beschikbaar is.

Als je (een gedeelte van) het Debian archief op je eigen harddisk hebt, kun je die met de keuze ‘filesystem’ toegankelijk maken.

Merk op dat je met alle methodes tegelijk kunt werken; het systeem zal steeds de nieuwste versies van de software-onderdelen installeren die op één of andere manier beschikbaar zijn.

Als je geen bronnen meer wilt toevoegen, kies je ‘No’ bij “Add another apt source”, of ‘Cancel’ (gebruik de TAB-toets) bij de methode-keuzelijst.

3.3.9 Selectie en installatie van extra onderdelen

Vervolgens wordt gevraagd om een ‘simple’ of ‘advanced’ installatie. Kies hier ***altijd*** voor ‘simple’, want anders kom je hopeloos vast te zitten en moet je weer helemaal opnieuw beginnen.

Debian heeft alle beschikbare programma’s ‘ingepakt’ in pakketjes. Als je nieuw bent in Debian of Linux, kan de veelheid daarvan een beetje overweldigend zijn: je hebt geen idee welke pakketjes je wel en niet moet installeren. Om daaraan tegemoet te komen, zijn een aantal ‘standaardlijstjes’ (zogenaamde *taken* of *tasks*) samengesteld, die vast een voor-selectie maken van de meest gangbare software.

Je krijgt een lange lijst met standaard-setjes gepresenteerd. Loop naar ‘Newbie Help’, druk daar op de spatie (zet een ‘*****’ in het vakje) en vervolgens op de ‘F’ van Finish. Kies op dit moment liever *geen* andere opties, want dan klopt de beschrijving hier niet meer goed; je kunt later met **tasksel** alsnog iets anders kiezen.

Hierna wordt een bevestiging gevraagd voor de lijst met pakketjes die geïnstalleerd moeten worden, druk op ‘y’ en Enter. Vervolgens worden de pakketjes automatisch van het netwerk, de harde schijf of de CDROM(’s) gehaald en geïnstalleerd. Bij gebruik van meerdere CDROM’s wordt wellicht een enkele keer gevraagd om een andere CD in de drive te doen; doe dat dan, wacht tot de drive de CD gedetecteerd heeft en druk op Enter.

Enkele pakketjes zullen vragen om een nadere configuratie; soms is dit ‘full-screen’ zoals eerder in de procedure. Ik zal de pakketjes even langslopen (de precieze volgorde kan trouwens van geval tot geval verschillen):

‘Lynx’: “What is lynx’s homepage?”: Enter (tenzij je iets beters weet).
‘Debconf-tiny’: “What frontend should be used?”: kies ‘Dialog’, Enter. Lees eventueel introductie-scherm, Enter. “Ignore questions with a priority”: kies ‘medium’, Enter.
‘Exim’: Enter na introductie-tekst. “Select a number from 1 to 5”: directe Internet-verbinding of lokaal netwerk: ‘1’; inbel-verbinding en geen lokaal netwerk: ‘2’; geen Internet en geen lokaal netwerk: ‘4’. Blijf dan Enter geven totdat één bepaalde vraag steeds weer terug komt, type daar je eigen username gevolgd door Enter. En dan nog één Enter om alles te bevestigen.

NB: als het systeem stil lijkt te staan, *geen toets drukken*. Als er een toets nodig is, wordt dat aangegeven; eventuele ‘extra’ toetsen worden in een keyboard-buffer gezet en als antwoord ingevuld bij volgende vragen (en dat is dus niet de bedoeling). Als het scherm door de automatische screensaver ‘op zwart’ gaat, *niet* een lettertoets indrukken, maar bijvoorbeeld Shift, Ctrl of Alt.

Aan het einde van de installatie wordt meestal gevraagd: “Erase the downloaded .deb files?”, druk daar op ‘y’ en Enter, en nog een Enter bij “Press enter to continue”.

3.3.10 De login prompt

Misschien komt er nog een mededeling dat de installatie gereed is; in een aantal gevallen moet je ook nog een tijdje wachten terwijl er niets lijkt te gebeuren (NB: zoals altijd niet op een toets drukken als dat niet wordt gevraagd).

Daarna verschijnt er een regel met ‘login:’ en daarachter een knipperende cursor. Dat is de *login prompt*. Daarachter kun je je eigen username intypen, en dan wordt om je password gevraagd. Linux is ‘up and running’! Tijd om er iets meer over te weten te komen... in het volgende hoofdstuk.

Let op: je mag de computer *niet* zomaar uitzetten, zelfs niet als de login prompt op het scherm staat en er niemand bezig is. Meer hierover staat in het begin van het volgende hoofdstuk, paragraaf 4.4. Voorlopig kun je de Ctrl-Alt-Delete toetscombinatie gebruiken, dat mag namelijk wel (niet alle Ctrl’s, Alt’s en Del’s hoeven te werken, probeer zolang tot je een werkende combinatie vindt). Er komen dan diverse meldingen over het scherm heen, die laten zien hoever het afsluiten van Linux vordert, en dan wordt de computer opnieuw opgestart. Tijdens het opnieuw opstarten (bijvoorbeeld tijdens het testen van het geheugen) mag je de computer uitzetten.

4. Gebruik van Linux: de basis

In dit hoofdstuk zal ik een groot aantal dingen bespreken die jou als beginnende Linuxer een beetje op weg zullen helpen. Een aantal zaken zijn geldig voor Linux in het algemeen, andere zijn specifiek voor Debian (andere distributies doen het soms op andere manieren).

Tussen de theorie door zal ik een groot aantal voorbeeld-commando's aangeven. Ik stel voor dat je die als eerste probeert, om wat grip op de zaken te krijgen. Als je zo een paragraaf hebt doorgewerkt, kun je zelf ook nog wat dingen proberen. Trouwens, als je geen root bent, ben je alleen een 'bedreiging' voor je eigen bestanden. En omdat je voorlopig nog geen belangrijke bestanden hebt, kun je alles doen -- je hoeft niet al te bang te zijn voor typefouten, er kan nog niets echt misgaan. Experimenteer gerust naar hartelust.

Om te beginnen een aantal hele fundamentele dingen, die je ook in elk ander operating system tegenkomt. Zoals het werken met bestanden. Linux heeft de meeste van deze zaken overgenomen uit UNIX, dus de hier besproken dingen zijn meestal niet specifiek voor Linux, maar werken ook in de meeste andere UNIX varianten.

4.1 Inloggen

De login prompt is altijd het eerste wat je ziet als je met een Linux systeem wilt gaan werken. Je moet je eigen username en password intypen. Als dat niet correct is, mag je het nog eens proberen, maar daarvoor moet je wel even wachten; dat is om krakers het leven wat zuurder te maken. Als je username en password kloppen, komt er een welkomst-melding op je scherm, die je even door mag lezen. Dat heet de *message of the day* ('bericht van de dag'), afgekort met *motd*. Helemaal onderaan staat iets wat eindigt met een dollar-teken ('$') en daarachter de knipperende cursor.

(NB: als je daar een hekje ('#') ziet, met daarachter de cursor, dan ben je als root ingelogd. Dat was dus *NIET* de bedoeling. Type **`exit`** gevolgd door Enter, en log in als gewone gebruiker.)

4.2 De shell

Die aanduiding met het dollar-teken aan het eind is de *prompt*, net als in DOS de overbekende **`C:\>`** prompt. Achter de prompt kun je opdrachten intypen, net als in DOS. Het programma dat die prompt op het scherm zet en dat jouw opdrachten interpreteert, heet de *shell*. In Debian is de shell met de naam **`bash`** de standaard shell, net als **`COMMAND.COM`** in DOS.

Wat invoer van commando's betreft, werkt het net als in DOS. Eerst de naam van een programma, dan eventueel een aantal *argumenten*. Als je DOS kent, zul je daar niets moeilijks aan vinden. Bovendien heeft **`bash`** standaard een mogelijkheid om de commandoregel (*command line*) te bewerken (pijltjestoetsen etc.) en ook een *history*, waarmee je vorige commando's kunt terughalen (pijltjes op en neer).

4.3 Uitloggen

Als je klaar bent met werken, geef je het commando **`logout`**. Of **`exit`**, wat hetzelfde effect heeft. Dit heet *uitloggen*. Daarna verschijnt de login prompt weer in beeld en kun jij, of een ander, weer inloggen.

4.4 De computer uitzetten

Als je de computer uit wilt gaan zetten, zijn er een aantal dingen die heel anders gaan dan in DOS (maar wel vergelijkbaar met Windows-varianten).

Als je bent aangesloten aan een lokaal netwerk (dat zelf eventueel aan Internet 'hangt'), kunnen er op elk willekeurig moment (ook als er niemand op de console ingelogd is!) vele verbindingen zijn met jouw computer. Zoals al opgemerkt in paragraaf 2.5.4, kunnen er mensen via het netwerk ingelogd zijn op / werken met jouw computer. Maar ook zijn verbindingen voor bestandsoverdracht mogelijk. Voor de computer uitgezet wordt, moeten al deze verbindingen netjes worden afgesloten, anders ontstaat er een kleine chaos op het netwerk.

Iets anders waar je rekening mee moet houden, is dat Linux alle harde schijven, cdroms en eventueel floppy's *cachet*, zowel voor lezen als voor schrijven (vergelijk **`smartdrv`** en **`ncache`** voor DOS). De *lees-cache* is een stuk

werkgeheugen dat wordt gebruikt om even te 'onthouden' wat er als laatste van de schijven is gelezen. Het komt namelijk heel vaak voor dat één bepaald stukje harde schijf meerdere keren vlak achter elkaar wordt gelezen, en dan is lezen uit het werkgeheugen veel sneller dan van de betreffende schijf (bijvoorbeeld een single-speed cdrom). Het schrijven naar een bestand gebeurt vaak in kleine stukjes met steeds een hele korte tussentijd. Het schrijven van kleine stukjes naar een harde schijf is bijzonder tijdrovend, vandaar dat er eerst een tussenopslag in het geheugen plaatsvindt, in de *schrijf-cache*. Als er genoeg kleine stukjes zijn 'gebufferd', worden ze in één keer achter elkaar naar de harde schijf geschreven; de tijd die daarvoor nodig is, is praktisch gelijk aan die voor het schrijven van één klein stukje. De cache wordt helemaal door de Linux kernel geregeld, zelfs de cache-grootte, die aangepast wordt aan het geheugengebruik van andere programma's -- met 64 MB werkgeheugen kan er gemakkelijk 50 MB gebruikt worden voor cache. Als de computer wordt uitgezet, is het wel de bedoeling dat die schrijf-cache eerst wordt 'geleegd', anders kunnen er gegevens verloren gaan.

Dat zijn de twee belangrijkste redenen waarom je *nooit* een computer die Linux draait zomaar mag uitzetten. Voor de *shutdown* is een speciale procedure ontwikkeld, die voor allerlei afsluit-bezigheden zorgdraagt. Die procedure is te starten door de Control, Alt en Delete toetsen tegelijkertijd in te drukken (aangeduid met Ctrl-Alt-Del). Niet alle toetsen met die opschriften hoeven trouwens te werken, dat vraagt wat experimenteren; meestal werken de linkse Ctrl en Alt, en de meest rechtse Del (de punt van het nummerblok) wel. Als je die toetscombinatie hebt gebruikt, zullen er diverse meldingen over het scherm komen en zal het systeem tenslotte rebooten. Na die reboot kun je de computer uitzetten, bijvoorbeeld tijdens het testen van het geheugen.

Trouwens, als je de computer niet netjes uitzet, zul je dat merken ook... Als je Linux dan weer opnieuw start, krijg je een melding als '/dev/hda5 was not cleanly unmounted, check forced', en dan wordt de hele harde schijf getest op fouten, die zouden kunnen zijn ontstaan omdat de schrijf-cache nog niet was geleegd. En dat testen kan bij grotere schijven meer dan tien minuten gaan duren.

4.5 Je eigen naam

Als je ingelogd bent, kun je aan de computer vragen wie hij denkt dat jij bent. Type achter de prompt **`whoami`** -- wie ben ik? Als antwoord verschijnt de naam waarmee je bent ingelogd.

(NB: Als daar staat 'root', ben je *FOUT*. Type **`exit`**, log in als gewone gebruiker, en ga terug naar AF).

Je ziet je eigen naam trouwens ook in de prompt staan, samen met de naam van de computer. Zo weet je direct wie je bent en waar je werkt, heel handig als je met veel verschillende computers over het netwerk bezig bent.

4.6 Het password veranderen

Als je je wachtwoord te zwak vindt, of als je iemand verdenkt van bepaalde zaken, is het verstandig je wachtwoord te veranderen. Dat gaat met het commando **`passwd`**. Je moet eerst je oude password intypen en dan twee keer het nieuwe. Je zult merken dat het nieuwe wachtwoord aan een aantal eisen moet voldoen, anders krijg je klachten (zie ook paragraaf 2.7). Je mag maximaal drie keer proberen.

4.7 De directory en de inhoud

In DOS is **`dir`** vaak het eerste commando dat je geeft. Dat kan ook in Linux, probeer maar. Je ziet alleen helemaal niets. Dat komt omdat de directory waarin je staat, leeg is.

Helaas geeft de prompt weinig informatie over de directory waar je in staat. Maar dat kan met een commando uitgevonden worden. In DOS gaat dat met (alleen) **`cd`** maar dat heeft in Linux een andere functie. Gebruik het commando **`pwd`**, van print working directory (huidige werkdirectory). Je ziet iets als **`/home/username`**. Ja, elke gebruiker krijgt z'n eigen directory, zijn *home directory* ("thuis directory"). Dus niet 'home' zoals in **`C:\`** "C-home" van DOS, want **`C:\`** zou in Linux "C-root" heten (als er in Linux een **`C:`**-schijf was geweest, maar er zijn geen schijfletters, zie paragraaf 2.5).

De home directory is het privé-domein van de gebruiker: alleen hij kan er bestanden inzetten en weghalen. Bestanden zijn standaard leesbaar voor iedereen, maar dat kan veranderd worden (zie paragraaf 4.11). Er is ook een tijdelijke directory, **`/tmp`**, waar iedereen in mag schrijven. Let op: bij het opnieuw opstarten van Debian zal **`/tmp`** worden geleegd!

Een directory waar wat meer in te bezichtigen is, is de root directory. Geef het commando `dir /` om dat te bekijken (gebruik '`/`' overal waar je in DOS '`\`' zou gebruiken, zie paragraaf 2.5). Je ziet dat de uitvoer wat lijkt op die van `dir /w` in DOS.

En daar duikt weer een verschil op. Probeer maar eens `dir /w` in Linux: het bestand `/w` bestaat niet. Linux gebruikt namelijk niet de slash ('`/`') maar het minteken ('`-`') om een schakeloptie (*switch* of *option*) aan te geven.

Een gevorderde Linuxer zal zijn wenkbrauwen fronsen als je het commando `dir` in Linux gebruikt. Het echte Linux commando is namelijk `ls`, van 'list contents of directory'. Met `ls /` krijgen we hetzelfde te zien als met `dir /`.

Laten we eens een paar switches proberen, met het commando `ls -a -l -F /` . Daarmee krijg je opeens een heleboel meer informatie. Meestal mag je die switches achter elkaar plakken, zoals in `ls -alF /` . Dat is het standaardcommando om in Linux een directory te bekijken. Het werkt trouwens in elke UNIX variant, omdat het in een standaard is vastgelegd.

De uitgebreide directory-listing van `ls -alF` bestaat uit de volgende velden (kolommen, *fields*), we komen later uitgebreid op alle aspecten terug:

1. De *mode bits*, zoals drwxr-xr-x. De eerste letter geeft het type bestand aan, als dat bijzonder is. 'd' staat bijvoorbeeld voor directory en '-' staat voor 'gewoon bestand'. De volgende drie groepen van drie letters zijn de read, write en execute (uitvoer) rechten (*permissions*) voor respectievelijk de eigenaar, de groep en 'alle anderen'. Als een bepaald recht niet is verleend, staat er een '-'. Bij directory's betekent het execute bit dat er toestemming is om naar die directory toe te gaan (met het commando `cd`).
2. Het aantal *hard links* naar het bestand (vergeet dat maar even);
3. De eigenaar van het bestand;
4. De groep van het bestand;
5. De grootte van het bestand in bytes;
6. De datum en tijd waarop het bestand voor het laatst is gewijzigd;
7. De bestandsnaam zelf;
8. Aan de bestandsnaam vast een teken wat 'specialiteiten' duidelijk maakt, bijvoorbeeld een '/' achter een directorynaam en een '*' achter een uitvoerbaar bestand (programma).

We kunnen ook naar de root directory toe gaan, met het commando `cd /` ("cd root"), en controleer met `pwd` of het ook gelukt is. Nu kun je met `ls -alF` (zonder de `/`) de inhoud van de root directory bekijken. Je zou weer terug kunnen gaan met **cd /home/<username>**, maar alleen `cd` heeft hetzelfde resultaat. Met `pwd` kun je het resultaat weer bekijken.

Geef ook in je home directory eens het commando `ls -alF` . Dan blijken er opeens toch wel bestanden in die directory te staan. Maar die beginnen allemaal met een punt. In Linux worden punt-bestanden (*dot files*) standaard niet in een directorylisting neergezet (net als de verborgen bestanden, *hidden files*, van DOS), maar `ls -alF` laat ze wel zien.

Net als DOS kent ook Linux het vraagteken en het sterretje, die voor respectievelijk één en meerdere willekeurige tekens staan. Probeer maar eens `cd /etc` om naar de `/etc` directory te gaan (en let op de prompt!). Dan `ls -alF` om alle bestanden te bekijken, en vervolgens `ls -alF ser*` en `ls -alF ?o?t*`. Maar ook na het sterretje mogen nog karakters staan, bijvoorbeeld `ls -alF *g` of `ls -alF *b*on*`. Ga met een enkele `cd` weer terug naar je home directory.

In de prompt staat blijkbaar de directory, en het 'golfje' ('tilde') '`~`' is een afkorting van je home directory. Ofwel `~/bestand` is precies hetzelfde als `/home/ikke/bestand`.

4.8 Het bekijken van een bestand

Om een bestand te bekijken, kun je het commando `cat` gebruiken. Probeer eens `cat /etc/motd`, en je ziet de message of the day nog eens. (Vergelijk het DOS-commando `type`.) `/etc` is trouwens de directory waar praktisch alle configuratiebestanden in staan.

Doe ook eens `cat /etc/inetd.conf`, en je ziet één van de configuratiebestanden (je hoeft niet te snappen wat er staat). Dat bestand is wel lang en het is van je scherm 'afgelopen'. Maar daar heeft Linux een trucje voor: met Shift-PageUp en Shift-PageDown kun je het scherm omhoog en omlaag schuiven, *scrollen* heet dat, een half scherm per keer. Je kunt misschien wel terugbladeren tot waar je ingelogd bent.

De uitvoer van `cat /etc/fb.modes` is wel heel erg lang, het zal waarschijnlijk niet meer helemaal te bekijken zijn door terug te scrollen, het past niet in de *scrollback buffer*. Voor zulke lange bestanden gebruik je het commando `less`, het 'betere' broertje van `more` in DOS.

Dus: `less /etc/fb/modes`. Met de Enter ga je één regel verder, met de spatie een heel scherm, of gebruik de pijltjestoetsen op/neer, of PageUp en PageDown. Met Home en End, of 'g' en 'G', ga je naar het begin en het einde van het bestand. Maar `less` kan meer. De toets 'q' bijvoorbeeld sluit meteen af, je hoeft niet eerst de hele tekst te bekijken. En met '/' (slash) kun je een zoekterm opgeven, 'n' is 'zoek volgende' en 'N' 'zoek vorige'. Probeer bijvoorbeeld eens '/50', dan krijg je de eerstvolgende '50' in de tekst te zien; gebruik 'n' en 'N' om andere '50's te vinden. Geef een 'h' voor meer hulp -- waar je ook weer doorheen kunt bladeren en 'q' om eruit te komen.

Probeer `cat bestaatniet` en je krijgt netjes een melding dat het commando `cat` het bestand `bestaatniet` niet kan vinden.

Type dan `cat /etc/ppp/pap-secrets` (de '`-`' is gewoon de min) en je ziet een typische Linux beveiliging: je hebt geen toestemming om dit bestand te lezen. Er kunnen namelijk geheimen in staan, in dit geval passwords voor inbel-accounts bij een Internet Service Provider. En het is niet de bedoeling dat elke willekeurige gebruiker die wachtwoorden te weten komt en op andermans kosten gaat bellen. Daarom is dat bestand alleen leesbaar voor root, en voor niemand anders. En dan is er voor jou als gewone gebruiker totaal geen enkele manier om de inhoud te weten te komen. Typisch Linux *security*. Om dat wat nader te onderzoeken, type je `ls -alF /etc` . Scroll eventueel met Shift-PageUp terug tot de regel met `ppp/`, en je ziet dat de mode bits op drwxr-x--- staan, eigenaar root en groep dip. Ofwel jij als niet-root en niet-dip mag helemaal niets met of in `/etc/ppp`.

Het commando `cat /etc/debian_version` geeft '2.2' terug. Maar die hele lange bestandsnaam intypen is lastig. En ook daar is wat op gevonden. Type alleen het eerste stuk, `cat /etc/deb`, en druk dan op de TAB-toets. En zie daar, de shell vult de naam compleet aan; een druk op Enter is voldoende om het bestand te bekijken. Dat scheelt dus 10 toetsaanslagen! Deze *file name completion* is trouwens een mogelijkheid (*feature*) die in bepaalde minder geavanceerde shells ontbreekt.

Probeer ook eens `cat /usr/bin/less`. Dat was dus een programma (namelijk `less`). En type nu eens 'abcdefg'. Ja, het scherm is een beetje 'stuk'. Als je dat ooit nog eens overkomt, kun je dat opheffen door (blind) het commando `reset` te geven. Probeer dat ook nu. Het scherm wordt schoongemaakt, er verschijnen een aantal mededelingen, en typen werkt weer zoals het hoort. In een aantal gevallen kun/moet je ook `echo` [Ctrl-V][Escape]`c` gebruiken.

4.9 Bestanden kopiëren, verplaatsen en verwijderen

Veel gebruikte commando's hebben in Linux vaak korte namen, om zo typewerk te besparen en efficiency te vergroten. Kopiëren en verplaatsen gaat met de commando's `cp` en `mv`, afkortingen van 'copy' en 'move' (bekend uit DOS). Verwijderen gaat met `rm`, van 'remove' (`del` of `erase` in DOS).

Laten we weer eens wat proberen. Kijk met `pwd` of je in je home directory staat; als dat niet zo is kun je daar met enkel `cd` terugkomen. Type dan `cp /etc/motd .` (inclusief de punt), daardoor wordt het bestand `/etc/motd` gekopieerd naar de huidige directory, die met een enkele punt wordt aangeduid. Kijk met `ls -alF` of het bestand is aangekomen; type `cat motd` om het te bekijken.

We kunnen het zojuist gekopieerde bestand ook een andere naam geven, bijvoorbeeld met `mv motd testbestand`; kijk met `ls -alF` en `cat testbestand` wat ervan terecht is gekomen.

Als we zijn uitgekeken op het bestand, kunnen we het weggooien met `rm testbestand`, verifieer de correcte werking met `ls -alF`.

4.10 Bestanden bewerken

In Linux zijn de meeste instellingen vastgelegd in tekstbestanden. Met een simpele *text editor* kunnen ze veranderd worden. Er zijn vele editors beschikbaar, maar op het standaard-systeem staat er maar één, namelijk **ae**, van Anthony's Editor (andere zijn later gemakkelijk toe te voegen).

We hebben een bestand nodig dat we kunnen bewerken. Laten we de message of the day weer eens pakken. Met het commando **cp /etc/motd ae_test_file** wordt de motd gekopieerd naar het bestand **ae_test_file** in de huidige directory (die je home directory zou moeten zijn).

(NB: ik ga er in het vervolg van uit, dat je zelf zo slim bent om met **ls -alF** en **cat** of **less** de resultaten van je acties te bekijken.)

Start de editor met **ae ae_test_file** (of **ae ae**<TAB>). Bovenaan het scherm zie je een beknopte hulptekst, onderin staat het bestand dat je aan het bewerken bent. Helemaal bovenin staat een regel met status-informatie. Loop met de cursortoetsen heen en weer en verander hier en daar iets.

Gebruik Ctrl-X gevolgd door Ctrl-W (vaak ook aangeduid met ^X^W, '^' betekent dan 'Ctrl-') om het bestand op te slaan. Geef Enter op de bestandsnaam, en **ae** meldt dat het bestand is opgeslagen.

Om **ae** af te sluiten gebruik je Ctrl-X gevolgd door Ctrl-C (ofwel ^X^C). Als je het bestand niet hebt opgeslagen, wordt je gevraagd of je echt wilt stoppen. Als je dan alsnog wilt opslaan, moet je dus 'n' antwoorden en apart ^X^W gebruiken.

Als je op het bestand **ae_test_file** bent uitgekeken, kun je het met **rm ae_test_file** weggooien.

4.11 Permissies veranderen

Zoals al bleek bij de directory-listing (paragraaf 4.7), heeft elk bestand een aantal mode bits, die aangeven wie wat met het bestand mag doen. Je kunt die permissies ook veranderen, met het commando **chmod**, van change mode.

Maak een testbestand met **cp /etc/motd modetestfile**, en bekijk de directory met **ls -alF**. De mode bits zullen op -rw-r--r-- staan. Dat wil zeggen dat jij als eigenaar zowel mag lezen als schrijven, en dat de groep en 'alle anderen' alleenmaar mogen lezen. Het is geen uitvoerbaar bestand (programma) dus er staat nergens een **x**. Merk trouwens op dat jouw groepnaam momenteel gelijk is aan je username, maar dat kan eventueel ook anders (zie paragraaf 13.2).

Om het bestand voor jezelf niet meer leesbaar te maken, doe je **chmod u-r modetestfile** (gebruik de TAB-toets! zie paragraaf 4.8). Met **ls -alF** blijkt dat de mode bits nu op --w-r--r-- staan, en dat was ook de bedoeling. Probeer het bestand eens te lezen met **cat modetestfile**: het mag niet, en zo hoort het ook.

De eerste switch (ook genoemd het eerste *argument*) van **chmod** heeft een bijzonder formaat: een letter, dan een **+** of **-**, en dan weer een letter. De eerste letter duidt op de persoon/personen om wie het gaat, **u** van user (eigenaar), **g** van group, **o** van others (alle anderen) en **a** van all (allemaal tegelijk). De laatste letter staat voor de permissie die aan- of uitgezet moet worden, **r** van read (lezen), **w** van write (schrijven) en **x** van execute (uitvoeren). Middenin staat een **+** als de permissie moet worden aangezet en een **-** als hij moet worden uitgezet. Er mogen eventueel meerdere letters achter elkaar worden opgegeven.

Met **chmod a-r modetestfile** (zelfde als **ugo-r**) trekken we alle leestoestemming in. En **chmod g+wx modetestfile** geeft de groep schrijf- en uitvoerrechten. De mode bits zijn nu als het goed is --w--wx---, iets wat in de praktijk zelden voor zal komen.

Naast de letter-teken-letter reeks is er ook nog een andere manier om mode bits aan te geven, namelijk met drie cijfers van 0 tot 7. Elk bit heeft namelijk een gewicht, zoals hieronder aangegeven.


```
cijfer            1       2       3
                /   \   /   \   /   \
gewicht         4 2 1   4 2 1   4 2 1
mode bit    -   r w x   r w x   r w x

        special   user   group   others
```

De reeks -rw-r--r-- is dus 644 in deze zogenaamde *octale* representatie, en --w--wx--- is 230. En ook daarmee kan **chmod** overweg, probeer **chmod 644 modetestfile** en **ls -alF**. De mode bits staan nu weer zoals ze oorspronkelijk stonden. Met de alphanumerieke notatie kon dat ook, namelijk met '**a+r,go-wx**', maar dat is wat omslachtiger.

4.12 Links

Een bestand wordt opgeslagen op een bepaalde plaats op de harde schijf. In de directory staat niet veel meer dan de naam van dat bestand en de plaats waar het op de harde schijf te vinden is. In Linux is het ook mogelijk om twee directory-items naar dezelfde plaats op de harde schijf te laten wijzen -- dan zijn er dus twee namen voor één bestand. We zeggen dan dat er twee *links* zijn naar dat bestand. Het commando dat daarvoor gebruikt wordt, is **ln**, van link.

Ga in je home directory staan (met alleen **cd** bijvoorbeeld) en type **cp /etc/motd origfile**. In plaats van **cp origfile secondfile** doe je nu **ln origfile secondfile**. Als je **ls -alF** geeft, lijkt het alsof er gewoon gekopieerd is. Wanneer je echter naar de tweede kolom van de listing kijkt, zie je daar 2 staan. Er zijn blijkbaar twee links en dat klopt ook. Verander nu met **ae** iets in **secondfile** (commando: **ae secondfile**), en bekijk **origfile** (met **cat origfile**). Je zult zien dat beide bestanden in feite hetzelfde zijn. Als je het bestand **secondfile** weggooit (**rm secondfile**), zal **origfile** gewoon blijven bestaan, met de gewijzigde inhoud.

Doe nog eens **ln origfile secondfile**; **secondfile** is dan de (eventueel gewijzigde) message of the day. Als je nu achtereenvolgens **mv origfile oldorigfile** en **cp /etc/debian_version origfile** doet, zal **secondfile** nog steeds de message of the day zijn. Dat hoort ook zo, want **secondfile** verwijst naar een *plaats op de harde schijf* en niet naar een bestand met de naam **origfile**. Dit heet een *hard link*.

Om toch naar '*het bestand met de naam* **origfile**' te verwijzen, is er een andere soort link, namelijk de *symbolic link* of *symlink* (enigszins te vergelijken met de snelkoppeling in Windows 95/98). Zo'n link maken gaat met de switch **-s** van **ln**.

Gooi om te beginnen alle 'oude' bestanden even weg, met **rm origfile oldorigfile secondfile**. Geef dan weer **cp /etc/motd origfile**, en nu **ln -s origfile secondfile**. Met **ls -alF** kun je het resultaat zien: **secondfile** is een symlink (de '**l**' links in de mode bits) en verwijst naar de *naam* **origfile**. Je kunt weer één van beide bestanden editen en het effect op de ander zien. Maar als je weer **mv origfile oldorigfile** en **cp /etc/debian_version origfile** doet, zal **secondfile** opeens de inhoud '2.2' hebben! Dat komt omdat het bestand met de *naam* **origfile** een andere inhoud heeft gekregen. Als je nu **rm origfile** geeft en vervolgens **cat secondfile**, zie je dat **secondfile** op zich eigenlijk helemaal niet bestaat, het is op dit moment een verwijzing naar niets. Ruim tenslotte met **rm oldorigfile secondfile** de testbestanden weer op.

In het Linux systeem wordt veel gebruik gemaakt van symlinks, kijk maar eens in **/sbin** (met **ls -alF /sbin**). Op deze manier is het mogelijk om nieuwe versies van bestanden en programma's te installeren, zonder de links te hoeven veranderen. Zolang de naam van het geupdate bestand hetzelfde blijft, wijst de symlink naar de geupdate versie.

4.13 Directory's maken en verwijderen

Met het commando **mkdir testdirectory** (van make directory) kun je een nieuwe subdirectory aanmaken (bekijk 'm met **ls -alF**). Met **cd testdirectory** 'duik' je erin en met **cd ..** ga je weer terug (nee, **cd..** aan elkaar werkt alleen in DOS).

Het weghalen van een lege(!) directory gaat met **rmdir testdirectory** (van remove directory).

Als je de directory niet zelf helemaal wilt leeghalen (bijvoorbeeld als er veel subdirectory's zijn) kun je **rm -rf testdirectory** gebruiken. LET OP: dan worden *zonder waarschuwing* alle bestanden en subdirectory's in die directory verwijderd! Denk dus *heel* goed na voor je als root **rm -rf** gebruikt, daar zijn namelijk al talloze systemen mee vernield!

4.14 Virtuele consoles

Linux is een multi-tasking operating system. Eén van de manieren om daarvan gebruik te maken, zijn de *virtuele consoles*.

Zorg dat je ingelogd bent en doe iets als **less /etc/fb.modes**, en blader er een stukje doorheen. Toets dan LinkseAlt-F2, ofwel ga naar de tweede virtuele console. De tekst die je aan het bekijken was verdwijnt en er verschijnt een login prompt. Als je weer LinkseAlt-F1 toetst, verschijnt de tekst weer -- op de eerste virtuele console.

Ga met LinkseAlt-F2 weer naar de tweede virtuele console en geef gewoon je eigen username en password (let ook op 'tty2' net boven de login prompt -- dat is de naam van die virtuele console; zie ook paragraaf 2.5.4). Je krijgt gewoon weer een shell gepresenteerd. Type daar bijvoorbeeld **ls -alF /**. En je hebt een directory-listing gekregen terwijl je (op de andere virtuele console) een bestand aan het bekijken was: dat is multi-tasking.

Met LinkseAlt-F3 t/m LinkseAlt-F6 kun je nog vier andere virtuele consoles bereiken; dat zou voldoende moeten zijn voor relatief zware multi-tasking (je kunt later eventueel meer toevoegen).

Je kunt zien wie er allemaal ingelogd zijn met het commando **who**, of wat uitgebreider met **who -HTi**. Probeer dat eens als je een aantal virtuele consoles 'open hebt staan'. Je kunt ook zien wat iedereen aan het doen is, met het commando **w**.

4.15 Processen

Elk programma dat 'draait', wordt een *proces* (of *process*) genoemd. Elk proces krijgt een uniek nummer, genoemd *process identification*, *process ID* of (meestal) *pid*. Elk proces draait op één bepaalde terminal (bijvoorbeeld een virtuele console; zie ook paragraaf 4.14).

Met het commando **ps** kun je een lijst krijgen van processen die jij hebt draaien. Daarop zie je bijvoorbeeld de shell(s) en ook **ps** zelf. In de kolom 'PID' staat het proces-nummer en onder 'TT' staat de terminal (plak **/dev/tty** ervoor voor de complete naam). Als er een vraagteken ('?') onder 'TT' staat, is er geen *controlling terminal* en draait het programma 'in de achtergrond' (vergelijk TSR's in DOS).

Gebruik **ps aux** om een uitgebreide lijst te zien van *alle* processen die op dit moment draaien (de **aux** is een switch, maar dan zonder '**–**' ervoor). En dat blijken er nogal wat te zijn. Ik zal ze hier niet allemaal behandelen, maar neem van me aan dat ze allemaal nodig zijn om het systeem correct te laten functioneren. In ieder geval is er een proces met de naam **init**, dat pid 1 heeft. Dat is het programma **/sbin/init** dat als eerste door de kernel is gestart (zie ook paragrafen 3.1.4 en 13.4). **Init** heeft er bijvoorbeeld voor gezorgd dat er zes virtuele consoles beschikbaar zijn gekomen.

Niet alle processen doen trouwens altijd iets. Sterker nog, de meeste processen 'liggen te slapen' en wachten tot ze 'wakker' gemaakt worden. De shell **bash** doet niets anders dan wachten tot jij eens een toets in gaat drukken, of wachten tot het programma wat erdoor gestart is, is afgelopen. Dit slapen kun je zien aan de letter 'S' (van sleep) in de kolom 'STAT' (van *state* of status, toestand). De 'R' daar is van running, dan is een proces ook echt iets aan het doen -- zoals **ps** zelf, die de lijst van processen aan het presenteren is. Onder 'TIME' staat het aantal minuten:seconden dat een programma totaal al heeft ge-run-d sinds het gestart is (kolom 'START'), en meestal is dat niet zo lang.

4.16 Process en job control

De toestand van een proces is te veranderen. Dat wil zeggen dat een proces even gepauzeerd kan worden, dan wel helemaal afgebroken. Dat gebeurt door een signaal (*signal*) naar het proces te sturen, bijvoorbeeld het 'stop ermee' of het 'pauze' signaal. Dat sturen van een signaal kan gebeuren met het commando **kill**.

Log in op twee virtuele consoles en geef in één het commando **more /etc/fb.modes** (dit is een minder krachtige variant van **less**, die voor deze demonstratie beter werkt). Zoek op de andere virtuele console uit, wat de pid is van **more**, bijvoorbeeld 274 (gebruik **ps u** of **ps aux**). Geef vervolgens het commando **kill 274** (vul de goede pid in)

en kijk op de andere virtuele console wat er is gebeurd. Juist, het proces **more** is beëindigd, (*terminated*) ofwel *gekilled*. Als je geen signaal specificeert, stuurt kill standaard het TERM signaal (van terminate, ofwel beëindigen -- daar komt de naam 'kill' vandaan).

Het beëindigen van een programma kan ook vanaf dezelfde virtuele console gebeuren en wel met Ctrl-C. Start weer eens **more /etc/fb.modes** (pijltje omhoog in de shell!) en breek het af met Ctrl-C. Dat werkt trouwens niet altijd, er zijn programma's die de Ctrl-C zelf ergens voor gebruiken of compleet negeren (zoals **less**). Dan moet je vanaf een andere terminal **kill** gebruiken. (Ctrl-C stuurt trouwens een INT signaal, dat ongeveer hetzelfde doet als een TERM signaal).

Er zijn ook programma's die niet op het TERM signaal reageren. Om zo'n programma toch te stoppen, gebruik je het KILL signaal, met **kill -KILL <pid>**. Met KILL krijgt het programma echter geen gelegenheid om zichzelf netjes af te sluiten (met TERM en INT wel), dus gebruik KILL alleen in noodgevallen.

Om een programma te pauzeren (*stoppen* of *suspenden*) gebruik je Ctrl-Z of **kill -STOP <pid>**. Start de **more** weer eens en druk op Ctrl-Z. Er komt een regeltje met 'stopped...' op het scherm en je krijgt de bash-prompt weer terug. Met **ps u** kun je zien dat het proces nog wel bestaat, maar het heeft nu de status 'T' (van stopped). Nu kun je bijvoorbeeld even 'tussendoor' een directory-listing opvragen met **ls -alF**. Ga dan weer terug naar **more** met het commando **fg** (van foreground), geef een paar keer Enter om verder te scrollen. Met **fg** wordt het laatst gepauzeerde programma weer opnieuw gestart (**kill -CONT <pid>** doet het zelfde, maar dat vindt de shell niet leuk).

Er kunnen trouwens meerdere programma's tegelijk gepauzeerd zijn. Pauzeer de **more** weer met Ctrl-Z, start nog een **more**, weer Ctrl-Z, nog een **more** en weer Ctrl-Z. Vraag nu met het commando **jobs** een lijstje op van alle processen die gepauzeerd zijn. Elke gestopte opdracht heeft een job-nummer (*job number*) gekregen, dat tussen blokhaken is weergegeven, zoals [2]. Als daarachter een plus staat, kan die job met **fg** naar de voorgrond gehaald worden. Met een pseudo-commando als **%2** kan ook bijvoorbeeld job 2 'geforceerd' naar de voorgrond gehaald worden. Met iets als **kill %3** wordt job 3 'afgemaakt', zonder op de voorgrond gekomen te zijn. Hier neemt **%<jobno>** dus de plaats in van **<pid>**. Om alle **more** jobs netjes af te sluiten, geef je een aantal keer **fg** gevolgd door 'q' (van quit).

4.17 In- en uitvoer van programma's

De meeste programma's zullen iets op het scherm zetten, zoals bijvoorbeeld **cat**. Andere programma's verwachten ook dat je wat op het toetsenbord intypt, zoals **less**. Die uitvoer en invoer gaan normaal naar het scherm en van het toetsenbord (de *standard output* en *standard input*, afgekort met *stdout* en *stdin*), maar dat kan ook anders. Zowel in- als uitvoer kunnen namelijk worden 'omgeleid'. Dat gaat met de 'kleiner dan', 'groter dan' en '*pipe*' tekens '**<**', '**>**' en '**|**' (meestal Shift-'****').

Bijvoorbeeld met **cat /etc/motd > uitvoer_van_cat** wordt de uitvoer van **cat** niet op het scherm getoond, maar naar een bestand **uitvoer_van_cat** geschreven. Als je daarna **cat /etc/debian_version > uitvoer_van_cat** doet, zul je zien dat z'n oorspronkelijke inhoud verdwenen is. Dat is te verhelpen door twee **>**'s te gebruiken: **cat /etc/motd >> uitvoer_van_cat** voegt de uitvoer toe aan het bestand (dat overigens nieuw gemaakt wordt als het nog niet bestond).

Een andere handige toepassing voor omleiding is het snel maken van een klein bestandje met tekst. Daarvoor gebruiken we het commando **echo**, dat simpelweg z'n argumenten op het scherm zet. Probeer maar eens **echo Dit is een test**. Ook deze uitvoer kunnen we omleiden en zo een kort bestandje maken, bijvoorbeeld met **echo Dit wordt omgeleid > uitvoer_van_echo**. Bekijk het bestandje met **cat uitvoer_van_echo**.

Geef eens alleen het commando **cat**. Er lijkt niets te gebeuren, maar type eens wat tekens gevolgd door Enter. Je ziet dat ze gedupliceerd worden. Type nog een paar regels en sluit af met Ctrl-D op een lege regel (einde-van-tekst teken). Met een commando **cat > uitvoer_van_cat** kun je dus op een primitieve manier ook langere tekstbestanden maken (vergelijk **copy con** in DOS).

Maar als cat invoer accepteert van het toetsenbord, kunnen we ook dat omleiden. Probeer eens **cat < uitvoer_van_echo**, wat natuurlijk hetzelfde resultaat heeft als wanneer er geen '**<**' tussen had gestaan.

Bij '**>**', '**>>**' en '**<**' ging het steeds om uit- en invoer naar/uit bestanden. Met de *pipe* ('sluis') kan er ook uitvoer naar een ander programma worden gestuurd. Probeer eerst eens **ls -alF /dev**, wat wel erg lang is. Maar als we die uitvoer nu eens met **less** zouden bekijken, zou het een stuk beter gaan. Geef daarom **ls -alF /dev | less**. Ofwel, de uitvoer van het commando **ls** wordt de invoer van het commando **less**.

In Linux worden deze omleidingen zeer vaak gebruikt. Dat maakt namelijk deel uit van de filosofie van UNIX systemen: er zijn een hele berg kleine programmaatjes die allemaal een klein beetje kunnen, en door ze met pipes achter elkaar te 'hangen' kan praktisch elk gewenst resultaat worden bereikt.

Iets wat in Linux vaak voorkomt, is dat we de uitvoer dan een programma niet willen zien, bijvoorbeeld omdat die veel te uitgebreid is en we er toch niets wijzer van worden. Dan wordt die uitvoer omgeleid naar **/dev/null**, een 'apparaat' dat als een zwart gat functioneert. Bijvoorbeeld **ls -alF /etc > /dev/null** (vergelijk **dir > nul** in DOS).

4.18 Processen in de achtergrond

Stel je voor dat je het commando **cat /usr/bin/* | gzip | od | gzip | od | gzip | md5sum > /tmp/testchecksum** zou willen geven (waarom doet er trouwens niet toe). Dat commando doet er flink lang over, zeker een paar minuten. En ondertussen wil je wel doorwerken. Dan zou je het natuurlijk op een aparte virtuele console kunnen doen, maar dan moet je daar ook weer inloggen. Er is echter een makkelijkere manier, namelijk het uitvoeren in de achtergrond (*background*). Dat kan, omdat het programma toch geen schermuitvoer heeft, alles wordt omgeleid naar een bestand.

Eén manier om dat te doen, is het programma gewoon starten door het in te typen. Suspend het dan (als je het te lang vindt gaan duren) met Ctrl-Z. Geef dan het commando **bg** (van background, vergelijk **fg**) en je krijgt een melding dat het tijdrovende commando weer is gestart. Maar tegelijk krijg je de prompt weer terug en kun je andere nuttige dingen gaan doen. Vraag bijvoorbeeld de **jobs** op en je ziet dat de job 'Running' is. Als de job klaar is, wacht de shell totdat er weer een prompt op het scherm gezet moet worden (bijvoorbeeld als je op Enter drukt), en dan wordt er eerst een mededeling afgedrukt dat de job afgelopen is.

Als je van tevoren weet, dat het commando lang gaat duren, plak er dan een ampersand ('**&**', en-teken) achter, dus **....testcksum &**. De job wordt dan meteen in de achtergrond gestart. Het jobnummer wordt afgedrukt, en de pid van het eerste proces in de job. De rest gaat zoals boven beschreven.

Ook een job die in de achtergrond draait, kun je met **%1** of **fg** weer in de voorgrond halen; met Ctrl-Z en **bg** weer terug in de achtergrond.

Om te zien wat er op het systeem gebeurt, wordt vaak het commando **top** gebruikt. Dat commando geeft elke paar seconden een lijst zoals **ps**, maar dan gesorteerd op processor-belasting. Tijdens ons voorbeeld-commando blijkt dat de **gzip**'s het meeste "CPU-tijd vreten", zie de kolom '%CPU'. Je kunt de update-snelheid van **top** veranderen door de 's' in te typen, gevolgd door een aantal seconden (probeer eens 0.01 en kijk naar de '%CPU' van **top** zelf). Je kunt **top** afsluiten door 'q' in te typen.

Merk op, dat achtergrond-processen gewoon blijven draaien als je uitlogt. Het jobnummer gaat dan wel verloren; als je ze wilt killen moet je hun pids gebruiken. Bovendien kun je ze op geen enkele manier meer in de voorgrond krijgen.

4.19 Het gebruik van .tgz archieven, tar, gzip en bzip2

In Linux is het standaardformaat voor archieven (*archives*) het **.tgz** formaat. Dat is eigenlijk een combinatie van twee formaten, namelijk **.tar** en **.gz**; er zijn in principe ook twee programma's voor nodig, **tar** en **gzip** (vandaar dat je ook vaak **.tar.gz** ziet). Met **tar** (van tape archive) wordt eerst een on-ingepakt (niet-gecompresst) archief gemaakt, door simpelweg alle te archiveren bestanden achter elkaar te plakken in één groot bestand. Vervolgens wordt dat éne grote bestand met **gzip** ingepakt (gecompresst).

Een voorbeeld. Ga met **cd /etc** naar de **/etc** directory. Met **tar cvf /tmp/testtarfile.tar *** wordt een **.tar**-bestand gemaakt met alles wat in **/etc** staat (de **cvf** is weer een '**-**'-loze switch); inpakken gaat met **gzip /tmp/testtarfile.tar**. (De foutmelding van **tar** komt omdat je niet alles in **/etc** mag lezen.) Als je nu met **ls -alF /tmp** de directory bekijkt, zie je dat er een bestand **testtarfile.tar.gz** staat -- **gzip** plakt namelijk altijd **.gz** achter de naam van het ingepakte bestand. De naam kun je eventueel met **mv** veranderen, bijvoorbeeld in **testtarfile.tgz**. In Debian kun je de twee stappen ook in één keer doen, omdat **gzip** 'ingebakken' zit in **tar**. Geef daarvoor het commando **tar czvf /tmp/testtarfile2.tgz ***, en de **.tgz** wordt in één keer gemaakt. Met **ls -alF /tmp** zie je dat de groottes van de archieven enkele bytes verschillen. Dit gebeurt omdat er een paar bytes in de ingepakte **.tar**'s verschillend zijn, namelijk de tijd waarop de archieven zijn gemaakt.

Je kunt de inhoud van het **.tgz** archief (ook wel "tarball" genoemd) bekijken met **tar tzvf /tmp/testtarfile2.tgz**. Als het om een ongecompresst **.tar** archief gaat, moet je steeds de '**z**' uit de optie-lijst weglaten.

Het archief uitpakken doen we in een nieuwe directory. Geef **cd /tmp**, gevolgd door **mkdir etcuittar** en **cd etcuittar**. Pak dan het archief uit met **tar xzvf /tmp/testtarfile2.tgz**. Bekijk het resultaat met **ls -alF**.

Met **gzip** kun je slechts één bestand per keer compressen (en dat is af en toe ook heel handig), daarom moeten we eerst **tar** gebruiken als we meerdere bestanden in één archief willen krijgen. **gzip** zonder bestandsnaam compresst de standard input en geeft het resultaat aan de standard output, ideaal om 'om te leiden'; **tar** met een min ('**-**') als bestandsnaam leest danwel schrijft van/naar standard input en output. We kunnen ze dus ook 'aan elkaar koppelen': **tar cvf - /etc/* | gzip > /tmp/testtarfile3.tgz** om in te pakken en **gunzip < /tmp/testtarfile3.tgz | tar tvf -** om te bekijken, bijvoorbeeld. Dat zijn handige manieren om op andere UNIX systemen met **.tgz**-bestanden te werken, omdat **tar** daar meestal geen compressie heeft ingebouwd. Je ziet trouwens, dat hier **etc/** in de bestandsnamen wordt meegenomen (en hiervoor niet) -- die heb je immers ook opgegeven! Let wel: dat eerste pad-stuk wordt ook meegenomen bij het uitpakken, en dat hoeft niet altijd te zijn wat je bedoelt; daar moet je dus bij het inpakken al rekening mee houden.

Een recente ontwikkeling is **bzip2**, een 'vervanger' voor **gzip** die vooral tekstbestanden aanzienlijk kleiner inpakt. Het grote nadeel is echter dat **bzip2** langzamer werkt en (ook tijdens het uitpakken) enorme hoeveelheden geheugen gebruikt (enkele megabytes); het is daarom minder geschikt voor oudere computers. Gebruik van **bzip2** en **bunzip2** is identiek aan **gzip** en **gunzip**, de extensie is **.bz2**. In combinatie met **tar** wordt dat **.tar.bz2** of **.tbz2**, voor directe **bzip2**-compressie met **tar** vervang je de **z** in de switch door **I**, dus bijvoorbeeld **cIvf** in plaats van **czvf**.

In Debian is trouwens ook ondersteuning voor **.arj** en **.rar** (alleen uitpakken), **.zip** en **.zoo** bestanden (en misschien nog meer), maar dan moeten wel de betreffende pakketjes zijn geïnstalleerd. NB: 'zip' en 'unzip' zijn twee aparte pakketjes, en voor de ARJ-uitpakker naar het pakketje 'unarj' zoeken.

4.20 De indeling van het filesystem

De bestanden in een Linux filesystem zijn gerangschikt op een manier waaraan je even moet wennen, maar die eigenlijk best wel makkelijk is. Ik zal een overzicht geven van een aantal veelgebruikte directory's; de informatie komt gedeeltelijk uit de *Linux Filesystem Structure Standard*, het document **/doc/package-developer/fsstnd-1.2.txt.gz** op de Debian 2.2 Binary-1 CDROM, dezelfde plaats op een Debian FTP site. Bepaalde termen zul je (nog) niet begrijpen; een aantal komt in de volgende hoofdstukken nog ter sprake.

Programma's

/bin: elementaire/essentiële programma's, bijv. **/bin/ls**.
/usr/bin: andere, niet-essentiële programma's, bijv. **/usr/bin/top**.
/usr/bin/X11 (= **/usr/X11R6/bin**): programma's die specifiek onder het X Window System draaien, bijv. **/usr/bin/X11/fvwm95** (zie hoofdstuk 6).
/usr/games: alle spellen bij elkaar, bijv. **/usr/games/tetris**.

/sbin: elementaire/essentiële programma's die alleen voor root bedoeld zijn (gewone gebruikers kunnen er niets mee/hebben er niets aan), bijv. **/sbin/reboot**.
/usr/sbin: andere, niet-essentiële programma's die alleen voor root bedoeld zijn, bijv. **/usr/sbin/adduser**

Met "essentiële programma's" worden die programma's bedoeld, die nodig zijn om een systeem minimaal te kunnen opstarten en eventueel andere partities (waar bijvoorbeeld **/usr** op staat) te kunnen repareren, in geval van nood.

Configuratie

/etc: praktisch alle configuratiebestanden, voor alle mogelijke programma's, bijv. **/etc/syslog.conf**. De meeste configuratiebestanden hebben een manual page (zie ook paragraaf 4.21).

/etc/X11: configuratie voor het X Window System, bijv. **/etc/X11/Xresources**.

Gebruikersbestanden

/home: de home directory's van alle gebruikers (behalve root).

/root: de home directory van root.

/tmp: tijdelijke bestanden (en eventueel directory's) van alle gebruikers door elkaar.

/var/tmp: praktisch gelijk aan **/tmp**; wordt zelden gebruikt.

Let op: de **/tmp** directory wordt tijdens het opstarten compleet geleegd -- het is een directory voor korte-termijn (*tijdelijke*, *temporary*) bestanden; **/var/tmp** wordt niet gewist.

Mount-punten

De directory's **/mnt**, **/floppy** en **/cdrom** zijn speciaal bedoeld als *mount point*, dat wil zeggen dat er 'iets algemeens', een floppy danwel een cdrom 'overheen gemount' kan worden (zie ook paragraaf 5.8).

Software die niet in een Debian pakketje zit

Als je software hebt (maakt, of download) die niet in een Debian pakketje zit, zet die dan eerst in je home directory (ik prefereer zelf iets als **/home/ikke/src/programmanaam** , met **src/** van source). Als alles goed blijkt te werken zou je het kunnen kopiëren naar een directory onder **/usr/local**, meestal **/usr/local/bin**. Die 'lokale' directory's worden namelijk nooit door een pakketjes-systeem gebruikt.

Let op: NOOIT gaan rommelen in **/bin**, **/usr/bin**, **/sbin** en **/usr/sbin**, want daar kan het pakketjes-beheer-systeem niet tegen; upgraden wordt daardoor onmogelijk.

De rest

Onder **/usr** zit nog veel meer, bijvoorbeeld:

/usr/share/doc: documentatie van bijna alle pakketjes op het systeem.

/usr/src: 'system-wide' sourcecode, bijvoorbeeld van de Linux kernel.

/usr/share/man: de 'broncode' van de manual pages.

/usr/share/info: de 'broncode' van de info pagina's.

Dan zijn er ook nog:

/boot: dingen die met opstarten te maken hebben, zoals het kernel-image **/boot/vmlinuz-2.2.17**.

/dev: alle devices bij elkaar, bijv. **/dev/audio**.

/lib: *library's* ("programmadeel-bibliotheken"); in **/usr/lib** staan er nog meer.

/lost+found: hier komen eventueel de gevonden 'lost clusters'.

/proc: informatie over de status van de computer en alle processen die draaien; probeer eens **cat /proc/cpuinfo** en **cat /proc/self/status**.

/var: een hele boel andere bestanden die 'variabel' zijn, bijvoorbeeld de print- en mail-spooler. In **/var/log** staan verschillende zeer interessante log-bestanden, het bekijken waard. Uit veiligheidsoogpunt kun je veel logfiles alleen als root bekijken.

Uitspraak

De korte directory-namen worden soms heel verschillend uitgesproken. De trend is om ze zo kort mogelijk te houden, dus **/usr/sbin** niet als 'slesj-uu-es-er-slesj-es-bee-ie-en' maar gewoon 'joesur-esbin' ("Nederlandse fonetische notatie"). De '**/**' wordt slechts zeer zelden uitgesproken, omdat die vrijwel altijd duidelijk is. **/var/tmp** wordt 'var-temp' en **/etc/X11** *meestal* 'ets-iks-elf'.

4.21 Meer hulp

Programma-hulp

De behandelde commando's kunnen veel meer dan ik hier uit de doeken heb gedaan. Het voert me echter te ver om dat allemaal te gaan behandelen. En dat is ook niet echt nodig, omdat de commando's ook zelf heel wat hulp kunnen bieden. Probeer maar eens alleen **cp** en op diens aanwijzing **cp --help**. Een hele lijst met mogelijke switches

verschijnt op het scherm, compleet met een beknopte uitleg. De meeste commando's hebben een switch **-h** of **--help**, die een hulptekst laat verschijnen.

Manpages

Er zijn in Linux ook vele uitgebreide handleidingen (*manuals*) beschikbaar. Het commando om een *manual page* te bekijken, is **man <programmanaam>**. Met **man -k <zoekwoord>** (van keyword) wordt in de lijst van manual pages gezocht naar titels die dat zoekwoord bevatten. De manual pages zijn onderverdeeld in een aantal secties (of hoofdstukken), genummerd van 1 tot 9. Wat in welke sectie thuishoort kun je lezen in de manpage van **man** zelf (met **man man** dus). Het komt voor dat er in twee secties een manpage met dezelfde naam staat -- bijvoorbeeld van **man** zelf, in sectie 1 en sectie 7. Om de juiste te kunnen bekijken, geef je **man 1 man** en/of **man 7 man**, in het algemeen **man <sectie> <programmanaam>**. Gebruik de '/'-toets (slash) om in een lange manpage naar een bepaalde term te zoeken.

Een leuk voorbeeld van het opzoeken van een manual page: je wilt weten hoeveel ruimte op een harde schijf in gebruik is. Ofwel, je wilt de 'disk usage' weten. Om het commando te zoeken wat daarbij hoort, geef je **man -k disk | less**. Om het commando snel te vinden zoek je naar 'usage' (type '/usage' gevolgd door Enter). Met **man df** lees je dan wat dat commando allemaal kan.

De 'source code' van de manpages kun je herkennen aan de bestandsnaam, iets als **bash.1.gz** (met **1** dus het sectie-nummer). Standaard kijkt **man** alleen in speciale directory's voor manual pages, maar soms kom je ergens een 'losse' manpage tegen die je dan niet zomaar met **man** kunt bekijken. Gebruik in dat geval de **-l** switch, bijvoorbeeld **man -l /usr/share/man/man1/bash.1.gz**.

Texinfo

Wat uitgebreidere manual pages staan in een ander formaat, het Texinfo ("tech-info") formaat. Geef het commando **info <programmanaam>** om het te bekijken, of alleen **info** voor een lijst. Texinfo is een 'hypertext'-systeem, wat wil zeggen dat er in de tekst *links* zijn naar andere punten in de tekst. Meestal kom je binnen in een inhoudsopgave-pagina; met de pijltjestoetsen kun je naar een item lopen, en met Enter spring je daarheen. Links staan meestal tussen een sterretje en twee dubbele punten, bijvoorbeeld "*note: More Features::"; als de link-tekst over meerdere regels doorloopt, werkt de Enter alleen op de eerste regel. Met de 'L'-toets spring je terug naar de vorige pagina. Als je een voetnoot tegenkomt die het halve scherm in beslag neemt, Ctrl-X 1 (^X 1 dus). Je kunt een heel Texinfo document ook in volgorde van voor naar achter doorlezen door steeds aan het einde van een pagina op de spatiebalk te drukken. Zoeken gaat met Ctrl-S; de zoekterm wordt gezocht terwijl je 'm intypt, weer Ctrl-S zoekt de volgende, Escape stopt het zoekproces. In **info** geeft 'H' of '?' meer informatie over het gebruik. Als er geen specifieke info-pagina van een bepaald commando is, wordt de normale manual page weergegeven. Verlaat **info** met 'q' of Ctrl-X Ctrl-C (^X^C).

HOWTO's

Er staat ook enorm veel informatie in de HOWTO's. Dat zijn documenten die allerlei ingewikkelde zaken voor nieuwelingen uit de doeken doen. Als je het 'doc-linux-text' pakketje hebt geïnstalleerd, staan ze in **/usr/share/doc/HOWTO/en-txt**. Ze zijn ook online op het Internet te raadplegen, op http://metalab.unc.edu/pub/Linux/docs/HOWTO. Maar let erop, dat die documenten niet Debian-specifiek zijn; sommige dingen gaan op een Debian systeem heel anders dan in de HOWTO's wordt uitgelegd.

Documentatie

Specifieke informatie/documentatie van de meeste pakketjes staat gegroepeerd in de **/usr/share/doc** directory.

Als je een **.gz** bestand wilt lezen, kun je dat bijvoorbeeld kopiëren naar de **/tmp** directory, en dan **gunzip**-pen en met **less** bekijken, maar **gunzip < bestand.gz | less** werkt handiger. Het makkelijkste is echter simpelweg **zless bestand.gz**.

Sommige documentatie is alleen in HTML-vorm beschikbaar, gebruik dan (voorlopig) **lynx bestand.html(.gz)**. Een goed voorbeeld is de 'Debian Guide', **lynx /usr/doc/debian-guide/html/noframes/index.html**. Merk op dat dat document enigszins verouderd is (dat is ook te zien aan **/usr/doc** zonder **share/**).

Internet
Op het Internet staan bergen informatie over van alles en nog wat. Interessante adressen:

http://www.debian.org - de Debian homepage
http://www.debian.org/doc - Debian-specifieke documentatie, met onder andere:
http://www.debian.org/doc/ddp - het Debian Documentation Project
http://www.linuxdoc.org - het Linux Documentation Project, met documentatie over zegmaar alles.
http://www.linux.org - met heel veel links
http://www.linuxhq.com - informatie over de allernieuwste kernels
http://www.nllgg.nl - Nederlandse Linux Gebruikers Groep
http://counter.li.org - hier kun je jezelf en je computer opgeven als Linux-gebruiker - aanrader!
http://www.slashdot.org - verzamelplaats voor allerlei Free Software-gerelateerde nieuwsberichten en discussies

Hier geldt ook weer: als het niet Debian-specifieke documentatie is, zijn bestandsnamen en procedures wellicht niet geheel of geheel niet van toepassing! Lees in zo'n geval eerst het hele document door, onthoud de belangrijkste dingen, leg het document dan terzijde, en zoek alles op je Debian systeem zelf uit.

Vergeet trouwens ook de vele Internet-zoekmachines niet. Met Google, AltaVista en Deja kun je meestal heel snel een antwoord vinden op een specifieke vraag. URL's: http://www.google.com, http://www.altavista.com en http://www.deja .com.

Nieuwsgroepen
Ook via het USENET news systeem (de 'nieuwsgroepen' of 'discussiegroepen') kun je vanalles te weten komen. De belangrijkste Linux-gerelateerde nieuwsgroepen zijn nl.comp.os.linux, nl.comp.os.linux.installatie, comp.os.linux.misc en comp.os.linux.annonce. Als je het vermoeden hebt een 'domme' beginnersvraag te stellen, zet dan 'NEWBIE' aan het begin van de Subject(=onderwerp)-regel. Wees bovendien zo nauwkeurig mogelijk: vermeld de distributie die je gebruikt (Debian 2.2 dus), eventueel de specificaties van je hardware, en geef foutmeldingen altijd volledig en correct weer. Hoe duidelijker je vraag is, hoe eerder er een bruikbaar antwoord komt.

Mailing lists
Een alternatieve nieuwsgroep-vorm is de mailing list. Daarmee worden alle artikeltjes gewoon naar alle abonnees gemaild. Er zijn een groot aantal actieve Debian-specifieke mailing lists, zie de Debian homepage, http://www.debian.org onder "Support". De 'debian-user' mailing list is de algemene list voor alle gebruikersvragen; hier gelden soortgelijke 'gedragsregels' als in de nieuwsgroepen.

Het gebruik van de verschillende Internet-gerelateerde zaken in Debian komt verderop uitgebreider aan de orde, vooral in hoofdstuk 12.

5. Het domein van de systeembeheerder

Zo af en toe zul je iets aan het systeem moeten veranderen, bijvoorbeeld netwerk-instellingen. Dat kun je niet als gewone gebruiker doen, maar daar moet je systeembeheerder voor zijn. Dat is ook nodig als je nieuwe software-pakketjes wilt installeren. Hoe dat allemaal gaat, zal ik nu enigszins uit de doeken doen.

Als je (als systeembeheerder) iets wilt veranderen wat je tijdens de Debian installatieprocedure hebt ingesteld (bijvoorbeeld het netwerk), kijk dan in de betreffende paragraaf van de installatie (hoofdstuk 3); meestal staat daar aangegeven naar welke configuratie-bestanden je moet zoeken.

5.1 Inloggen als root, of su -

Je kunt je op een aantal manieren als systeembeheerder (ofwel root) aanmelden. Het meest voor de hand liggende is om gewoon 'root' in te typen bij de login prompt, gevolgd door het root-password. Dat werkt echter alleen als je op de console zit en niet als je een netwerkverbinding met de betreffende computer hebt.

Een manier die wel altijd werkt, is als gewone gebruiker in te loggen en dan het commando **`su -`** te geven (inclusief het min-teken). Ook dan wordt om het root-password gevraagd. Een voordeel van deze methode is, dat ook wordt bijgehouden ('gelogd') wie er root is geworden -- in het bestand **`/var/log/auth.log`**.

Als je als root bent ingelogd (of ge-su-d), krijg je ook weer een shell. De prompt ziet er een beetje anders uit; de username (root) staat er niet, en aan het eind staat een hekje ('#'). Dat hekje is het 'universele' teken dat je root bent (en dat je dus moet uitkijken wat je doet). Je had waarschijnlijk al door dat het golfje de directory voorstelt, het is een afkorting voor de home directory, die **`/root`** is voor root (probeer maar eens **`pwd`**). Na **`cd /etc`** komt daar de complete directory te staan. Met alleen **`cd`** (of **`cd ~`**) ga je weer terug naar de home directory (van root in dit geval).

5.2 Voorzichtig!

Als root kun je alles doen wat een gewone gebruiker ook kan, dus bijvoorbeeld alle commando's die ik tot nu toe besproken heb. Je moet als root echter wel zeer voorzichtig handelen, omdat je als root simpelweg alles mag en alles kan.

Ik zal één voorbeeld geven van iets wat ongewild heel erg fout kan gaan. **MET DIT COMMANDO ZUL JE ALLE BESTANDEN OP JE HARDE SCHIJF WEGGOOIEN -- GA DIT VOORBEELD DUS NIET IN DE PRAKTIJK PROBEREN!!!** Stel je voor dat je per ongeluk een directory met de naam **`xyz`** in de root directory hebt aangemaakt. Dat was dus ook al fout gegaan, maar niet desastreus. Stel nu dat je als root in je home directory staat, dan moet je dus **`rm -rf /xyz`** geven om die 'foute' directory te verwijderen. Nu ben je de TAB-completion zo gewend, dat je **`rm -rf /x`** wilt typen en dan op de TAB-toets wilt drukken. Maar stel nu dat je niet hard genoeg op de 'x' drukte. Dan staat er dus **`rm -rf /`**, maar de TAB-toets doet nu niets, omdat niet duidelijk is welke bestandsnaam je wilt completeren. Er blijft dus **`rm -rf /`** staan, en in je haast druk je meteen op Enter. Als je merkt dat je harde schijf wel erg lang blijft ratelen, is het al te laat. Snel Ctrl-C drukken zal niet veel meer helpen -- je bent je hele filesystem aan het weggooien... Nogmaals: **GA DIT NIET PROBEREN!!!**

Ik hoop, dat dit voorbeeld voor zichzelf spreekt. Als je je er niets van aantrekt, zul je het wel merken. NB: daar ben je dan helemaal zelf schuldig aan -- ik heb je gewaarschuwd.

Nog iets anders. Ik ken mensen, die zomaar bestanden/programma's weggooien, waarvan ze denken dat ze die niet (meer) nodig hebben. **DOE DAT NOOIT IN LINUX!** Simpelweg, omdat je geen idee hebt waar welke bestanden voor gebruikt worden. Daarbij komt, dat het pakketjes-beheer-systeem enorm in de war kan raken als jij alles naar jouw eigen inzichten gaat veranderen. Een systeem waaraan gerommeld is, is niet meer te upgraden. De enige plaatsen waar jij heer en meester bent, zijn **`/home`**, **`/root`**, **`/tmp`** en **`/usr/local`**. Verander verder nergens iets, tenzij je heel zeker weet wat je aan het doen bent (dat geldt ook voor de configuratiebestanden in **`/etc`**!).

Let wel: iets wat je als gewone gebruiker niet mag, is *per definitie gevaarlijk*.

5.3 Het systeem halten of rebooten

Eén van de zaken waarmee een systeembeheerder zich nuttig kan maken, is het afsluiten van Linux. Dat hoeft namelijk niet persé met Ctrl-Alt-Del, maar daar zijn ook commando's voor. Bijvoorbeeld het commando **reboot**, wat hetzelfde doet als Ctrl-Alt-Delete, ofwel Linux afsluiten en het systeem opnieuw opstarten. Of het commando **halt**, dat alleen Linux afsluit en dan verder totaal niets doet -- de computer reageert nergens meer op en kan worden uitgezet.

Er is een meer multifunctioneel programma, met de naam **shutdown**. Dat kan naar keuze wel of niet rebooten en ook eventueel een tijd wachten voordat de *shutdown sequence* gestart wordt. De switch **-h** zorgt voor alleen halt-en (dus niet rebooten) en **-r** voor (wel) rebooten. Het laatste argument geeft aan wanneer gehalt of gereboot moet worden. Daarvoor zijn drie formaten, namelijk **uren:minuten** voor 'shutdownen' op de aangegeven tijd, of **+minuten** voor shutdownen na het aangegeven aantal minuten, of **now** als er niet gewacht hoeft te worden.

Een paar voorbeelden: **shutdown -r 04:00** zorgt ervoor dat de komende nacht om vier uur gereboot gaat worden (bijvoorbeeld zodat er een nieuwe kernel gestart kan worden zonder andere gebruikers te hinderen); **shutdown -h now** zal direct beginnen met halten, het is een alternatief voor het **halt** commando; **shutdown -h +10** tenslotte zal over tien minuten gaan halten. Een **shutdown** commando zal wachten tot de aangegeven tijd, om dan met de shutdown sequence te beginnen. Als je het in de voorgrond laat draaien, kun je een 'uitgestelde shutdown' afbreken (*cancelen*) door op Ctrl-C te drukken; als shutdown in de achtergrond draait (bijvoorbeeld gestart met **shutdown -r +10 &**) kun je afbreken met **shutdown -c** (van cancel). Natuurlijk kun je ook **kill** gebruiken, maar dan moet je eerst de pid van de draaiende shutdown uitzoeken.

Om de gebruikers te waarschuwen, zullen er meldingen over hun schermen worden 'geplakt' (desnoods dwars door alles heen) dat het systeem binnenkort 'down' gaat. Dat gebeurt een aantal keren, vanaf vijftien minuten voor de geplande shutdown.

5.4 Een nieuwe gebruiker aanmaken, gebruikers 'weggooien'

Het kan zo af en toe gebeuren dat er iemand naar je toe komt met de vraag of hij of zij even van je computer gebruik mag maken. Dat is natuurlijk geen probleem (als die persoon tenminste iets van Linux afweet). Maar je laat hem of haar natuurlijk niet aan jouw waardevolle bestanden komen. Daarom maak je speciaal voor die persoon een nieuw account aan -- met een eigen username en password, en een eigen home directory. Dat doe je (als root natuurlijk) met het commando **adduser <nieuwe-username>**. Als er dan gevraagd wordt om een password, verdient het de aanbeveling om die persoon zelf even achter het toetsenbord te laten plaatsnemen en het zelf verzonnen password twee maal te laten intypen (terwijl jij natuurlijk oplet of hij/zij niet stiekem iets ongeoorloofds doet). Volledige naam en andere gegevens kun jij dan zelf weer intypen, op de manier zoals je gewend bent.

Een gebruiker 'weggooien' gaat eenvoudig met **userdel -r <username>**. Met dit commando wordt alle informatie van de gebruiker gewist, inclusief de home directory, *zonder verdere bevestiging*. Dus: voorzichtig hiermee!

Uitgebreidere informatie over gebruikers en groepen staat in paragraaf 13.2.

5.5 Gebruik van loadlin en LILO

Loadlin
Met het DOS-programma **loadlin.exe** kun je Linux opstarten vanuit DOS; het staat op Debian 2.2 Binary-1 CDROM, **/install/loadlin.exe**, en ingepakt in **/tools/lodlin16.zip** op die CD of op een Debian FTP site. Start zonder parameters voor hulp. Normaal gebruik is bijvoorbeeld **loadlin vmlinuz root=/dev/hda2 ro**; de kernel (het bestand **/vmlinuz** in Linux) moet je dus eerst op de DOS-partitie bij **loadlin.exe** zetten, en je moet de juiste Linux root partitie opgeven. De **ro** betekent dat de root partitie in eerste instantie read-only gemount moet worden. Je kunt ook vanaf je Windows 95/98 partitie starten, maar dan moet je eerst uit Windows terug naar 'kaal' DOS (**loadlin** geeft anders netjes een foutmelding). Let op: laat in geen geval een snelkoppeling aanmaken.

LILO
Met LILO kun je Linux gelijk starten als je de computer aanzet. LILO is erg krachtig, maar gebruik ervan *kan* ingewikkeld zijn. De LILO documentatie in **/usr/share/doc/lilo** is uitgebreid en heel helder; gebruik **zless /usr/share/doc/lilo/Manual.txt.gz** om het te lezen. Er is waarschijnlijk al een **/etc/lilo.conf** gemaakt tijdens de installatie van Debian, daar staat nuttig commentaar bij. Het commando **liloconfig** kan een (nieuwe) simpele LILO-configuratie maken, gebaseerd op jouw systeem, met wat ander commentaar in de **lilo.conf** (de oude wordt gebackupt in **lilo.conf.OLD**). Let wel: **liloconfig** is niet zo slim, en geeft vaak geen juist resultaat. Met LILO kun je naast Linux ook DOS, Windows en andere operating systems opstarten, zie daarvoor de documentatie.

MBR
De boot manager die is besproken in paragraaf 3.2.15 (met de '2FA:'-prompt), is geen LILO, maar een apart programmaatje, dat door **liloconfig** op de harde schijf kan worden gezet. Je kunt dat ook met de hand doen, met het commando **install-mbr /dev/hda --reset** (**/dev/hda** vervangen door de aanduiding van je eerste harde schijf; dus *niet* een aanduiding van een partitie op die schijf!). Met **install-mbr** kun je het één en ander aanpassen, vooral veiligheid-opties; zie de manpage voor meer informatie.

Activate
Het 'bootable' (='actief') maken van een partitie gaat het makkelijkste met een commando als **activate /dev/hda 4**, hiermee wordt hda4 actief gemaakt (let op de spatie tussen 'hda' en '4'!). Je kunt natuurlijk ook **cfdisk /dev/hda** gebruiken (het [Bootable] commando), maar daarmee kun je geen uitgebreide (extended) partities actief maken -- en daar staat juist vaak LILO in. De DOS/Windows **fdisk** kan alleen DOS/Windows partities bootable maken.

5.6 Debian pakketjes en hun gebruik

De kracht van Debian ligt in de manier waarop de software wordt aangeboden, namelijk in pakketjes of *packages*. In zo'n pakketje staat echter niet alleen de software (het programma) zelf, maar nog flink wat meer. Zoals bijvoorbeeld een aantal scripts (vergelijkbaar met batch-bestanden in DOS) die het installeren in goede banen moeten leiden -- de **preinst**, **postinst**, **prerm** en **postrm** scripts, die respectievelijk voor en na de installatie, en voor en na het weghalen van het pakketje worden uitgevoerd.

5.6.1 Informatie

Maar het meest belangrijke 'extraatje' is wel wat ik samenvat als 'informatie' over het pakketje. Niet alleen wordt gespecificeerd welke bestanden er allemaal inzitten (handig voor het verwijderen) en welke daarvan configuratiebestanden zijn (die dus meestal juist *niet* weggegooid mogen worden), maar ook van welke andere pakketjes het betreffende pakketje 'afhankelijk' is -- de zogenaamde *dependency's*. Die 'afhankelijkheden' zijn voor vitaal belang als er zomaar willekeurig pakketjes worden geïnstalleerd, want daarmee wordt verzekerd dat altijd alle programma's in de geïnstalleerde pakketjes werken. Deze informatie staat in de pakketjes zelf, maar ook verzameld in het bestand **/dists/potato/main/binary-i386/Packages(.gz)** op de Debian Binary CDROMs en de FTP sites.

5.6.2 Versies

Praktisch alle programma's hebben een versienummer, bijvoorbeeld de Linux kernel met versie 2.2.17. Maar van een bepaald programma wordt soms herhaaldelijk een nieuw pakketje gemaakt, waarbij er bijvoorbeeld wat meer bugs zijn verwijderd, of betere install-scripts zijn gemaakt, terwijl het nog steeds om dezelfde versie van het betreffende programma gaat. Om toch onderscheid te houden, krijgt ieder pakketje een eigen Debian-volgnummer, dat bij 1 start als er weer een 'echte' nieuwe versie van het betreffende programma uitkomt. Bijvoorbeeld de schermaansturing-bibliotheek 'ncurses' versie 5.0 is al zes keer in een Debian pakketje gezet, dus het huidige Debian-volgnummer is 6. Het totale versienummer is een combinatie van die twee en wel beide versies gescheiden door een minteken: 5.0-6. Ook die versienummers worden vaak gebruikt bij het aangeven van dependency's.

5.6.3 Pakket-namen en bestandsnamen

Pakketjes hebben praktisch altijd de naam van het programma 'dat erin zit', maar dat hoeft niet (soms is het een verzamelnaam). Als er veel verschil zit tussen opeenvolgende versies van een programma (of als twee versies naast elkaar kunnen worden gebruikt), kan ook het versienummer deel uitmaken van de pakket-naam, bijvoorbeeld 'libncurses5'. De bestandsnamen van de pakketjes zijn nog ingewikkelder en volgen het patroon **<pakketnaam>_<totale-versienummer>.deb**, bijvoorbeeld **libncurses5_5.0-6.deb**.

5.6.4 Voorbeelden van dependency's

Een paar andere voorbeelden. De editor **ae** zit in het pakketje met de naam 'ae', in het bestand **ae_962-26.deb**; het gaat dus om **ae** versie 962, die voor de 26ste keer in een Debian pakketje is gezet. Het heeft een regel 'Depends: libc6 (>= 2.1), slang1 (>> 1.3.0-0)'. Dus **ae** zal alleen werken als de pakketjes 'libc6' en 'slang1' zijn geïnstalleerd. Beide

zijn bibliotheken (*library's*) met programma-delen die tijdens het gebruik van **ae** worden aangeroepen, de éne met algemene standaard-functies, de andere met specifieke scherm-aansturing. Van 'libc6' moet tenminste versie 2.1 aanwezig zijn en voor 'slang1' is een versie groter dan 1.3.0-0 vereist.
Het programma **chmod** zit samen met een aantal 'vriendjes' in het (verzamel-)pakketje 'fileutils', in het bestand **fileutils_4.01-8.deb**. Het heeft een regel 'Pre-Depends: libc6 (>= 2.1.2)', wat betekent dat dit pakketje niet eens mag worden geïnstalleerd als er geen geschikte versie van 'libc6' is.
Bij het pakketje 'freetype-tools' staat de regel 'Conflicts: freetype (<= 0.beta.1997.12.25-1)'. Dat betekent dat 'freetype-tools' *niet* zal werken als de opgegeven versie of een eerdere versie van het 'conflicterende' pakketje is geïnstalleerd. Het gaat hier om een gesplitst pakketje, waarbij het laatste 'complete' pakketje 'freetype 0.beta.1997.12 .25-1' was, dat toen gesplitst is in 'freetype-tools' en een paar andere. Door het op deze manier op te geven, moet eerst het oude pakketje worden verwijderd voordat een nieuwe kan worden geïnstalleerd. (Weghalen van oude versies gebeurt normaal automatisch; dit soort constructies zijn alleen nodig als de namen van de pakketjes gewijzigd zijn.)

Naast 'Depends', 'Pre-Depends' en 'Conflicts' regels, zijn er ook 'Recommends' en 'Suggests' regels. Daarmee wordt aangegeven dat het betreffende pakketje *beter* zal werken als ook een aantal andere pakketjes geïnstalleerd zijn. 'Recommends' is een sterkere aanbeveling dan 'Suggests'. Het is over het algemeen zinvol om 'Recommends' op te volgen; 'Suggests' heeft meer het karakter "misschien vind je dat ook wel handig."

5.6.5 Virtuele pakketjes

Met het pakketje 'apache-doc' lopen we tegen een ander probleem aan. Dat pakketje bestaat namelijk uit een groot aantal documenten in het HTML-formaat. Voor het bekijken van die documenten heb je dus een web-browser nodig, bijvoorbeeld Netscape. Maar als er alleen 'Recommends: netscape' zou staan, zou bijvoorbeeld de browser mosaic buiten de boot vallen. Goed, dan kan ook 'Recommends: netscape ¦ mosaic' nog en is één van de twee al goed ('¦' staat voor 'of'). Maar ook mozilla, lynx en emacs kunnen html-teksten bekijken -- en misschien zijn of komen er nog wel meer. Het is heel wat werk om daarvoor steeds de 'Recommends:'-regel te wijzigen. Ook daar is dus iets op gevonden: het *virtuele pakketje* (*virtual package*). Dat is een pakketje dat niet bestaat en toch ook weer wel. In feite is het een type-aanduiding, zoals 'www-browser' in ons voorbeeld. Elk pakketje dat een web-browser 'levert', heeft een regel 'Provides: www-browser', en 'apache-doc' heeft dan 'Recommends: www-browser'. Op die manier wordt aangegeven dat het aanbevolen is om één of meerdere web-browsers te installeren, zonder ze allemaal bij name te noemen. Als er dan nog een web-browser bijkomt, hoeft die alleen in z'n eigen pakketje een 'Provides' regel op te nemen, en alles werkt zoals het hoort. Andere virtuele pakketjes zijn bijvoorbeeld 'mail-reader' en 'c-compiler'.

5.6.6 Installeren en verwijderen

Om een pakketje te installeren en te verwijderen moeten er door het *Package Management System* (pakketjes-beheer-systeem) een aantal handelingen worden verricht, dit gebeurt allemaal volautomatisch. Ik zal hiervan nu een beknopt overzicht geven, zodat je weet wat er ongeveer gebeurt.

Nog voor de werkelijke installatie begint, wordt eerst gekeken of er Pre-Dependency's zijn. Als het pakketje vooraf-afhankelijk is van één of meer pakketjes die nog niet geïnstalleerd zijn, zal het installatieproces worden afgebroken.

Wanneer aan de Pre-Depend-voorwaarden zijn voldaan, kunnen diverse pakketjes reeds geconfigureerd worden met behulp van de 'debconf' faciliteit. De antwoorden worden in een database opgeslagen, en later gebruikt. Dan wordt het **preinst** script uitgevoerd. Hierin kunnen bijvoorbeeld netwerk-diensten even uitgezet worden, zodat daar zonder problemen nieuwe versies voor kunnen worden geïnstalleerd. Omdat dit soort zaken voor ieder pakketje verschillend zijn, heeft ieder pakketje z'n eigen scripts (zoals **preinst**).

Vervolgens wordt het pakketje daadwerkelijk uitgepakt. Hiermee komen alle bestanden uit dat pakketje gelijk op hun goede plaats terecht, bijvoorbeeld in **/usr/bin** of **/etc**.

Als laatste stap van de installatie wordt het **postinst** script uitgevoerd. Daarin worden soms vragen gesteld aan de gebruiker, om zo het pakketje te kunnen configureren; pakketjes die 'debconf' gebruiken kunnen de antwoorden uit de database lezen. Daarnaast kunnen bijvoorbeeld ook de netwerk-diensten weer opnieuw opgestart worden, waarbij de zojuist geïnstalleerde nieuwe versies van de programma's worden gebruikt.

Het verwijderen van pakketjes begint met het uitvoeren van het **prerm** script. Hierin worden bijvoorbeeld netwerk-services, die door het betreffende pakketje geleverd werden, permanent uitgezet -- het pakketje wordt immers verwijderd.

Vervolgens worden alle bestanden die bij het pakketje horen weggegooid (de namen van die bestanden staan in een lijstje). Naar keuze worden de configuratiebestanden bewaard, of ook weggegooid; dat bewaren is handig als er binnenkort een nieuwe versie zal worden geïnstalleerd, die dan niet meer geconfigureerd hoeft te worden.

Na het verwijderen wordt tenslotte het **postrm** script uitgevoerd. Dat kan nog wat meer pakket-specifieke opruimwerkzaamheden verrichten.

5.6.7 Indelingen: secties en prioriteiten

Om wat meer overzicht aan te brengen in de veelheid van pakketjes, is er een indeling gemaakt in een aantal *secties*. Zo zijn er bijvoorbeeld secties 'admin', 'devel', 'mail', 'sound', 'editors' en 'games'.

Voor nieuwe Linuxers is echter vaak de vraag welke pakketjes nu echt nodig zijn, welke pakketjes op een Linux systeem 'horen' en welke alleen maar bij zeer specifieke interesses gebruikt worden. Daarom is er ook een indeling gemaakt in *prioriteiten*, namelijk 'required', 'important', 'standard', 'optional' en 'extra'. Die indeling loopt dwars door de secties heen; er zijn bijvoorbeeld 'important' pakketjes in secties 'editors' en 'mail', maar in diezelfde secties zijn ook 'optional' pakketjes.

De prioriteiten betekenen het volgende:
'Required': essentiële pakketjes die nodig zijn voor het juist functioneren van het systeem; als ze verwijderd worden zal het systeem niet meer werken;
'Important': belangrijke pakketjes die op elk UNIX-achtig systeem aanwezig zijn en ook vaak door andere pakketjes gebruikt worden;
'Standard': de meest gangbare pakketjes die een gevorderde Linuxer op elk systeem verwacht (alleen tekstmode, geen X);
'Optional': veruit de grootste sectie, met allerlei andere pakketjes die 'nuttig of aangenaam' kunnen zijn, bijvoorbeeld het X Window System, maar ook alle spellen;
'Extra': pakketjes die je alleen nodig zult hebben als je heel specifieke behoeften hebt.

5.6.8 Actie en status

Met pakketjes kun je verschillende dingen doen, en er kunnen een aantal dingen mee aan de hand zijn. Je kunt een pakketje installeren, verwijderen met of zonder behoud van configuratiebestanden, en wel of niet automatisch upgraden. Een pakketje zelf kan nieuw zijn, geïnstalleerd, wel of niet geconfigureerd, verwijderd met of zonder behoud van configuratiebestanden, en er kan ook nog het één en ander mee misgegaan zijn. Voor elk pakketje wordt bijgehouden wat de laatste actie was en wat de huidige toestand (status) is. Ik zal de acties en toestanden nog wat verduidelijken.

Als eerste de acties die je met een pakketje kunt ondernemen:
'New package', ook 'Keep new': voor nieuwe, nooit geïnstalleerde pakketjes, die nieuw moeten blijven -- dit is de 'standaard actie';
'Install': als het om een nieuw pakketje gaat, wordt het geïnstalleerd; met een reeds geïnstalleerd pakketje wordt niets gedaan, behalve als er een nieuwe versie beschikbaar is (dan wordt het automatisch ge-upgrade);
'Hold': de huidige versie van het pakketje geïnstalleerd laten en geen nieuwe versie installeren, ook al is die beschikbaar;
'Remove': het pakketje wordt verwijderd, maar de configuratiebestanden blijven bestaan;
'Purge': het pakketje wordt compleet verwijderd, inclusief configuratiebestanden.

De verschillende toestanden:
'Not installed': het is een nieuw pakketje, dat nog nooit geïnstalleerd is geweest, of het pakketje is verwijderd inclusief configuratiebestanden ('gepurged');
'Half-installed': tijdens het installeren (uitpakken) is er iets misgegaan;
'Unpacked (not set up)': installeren was succesvol, maar het pakketje is (nog) niet geconfigureerd (meestal vanwege dependency-problemen);
'Half-configured': tijdens het **postinst** script (de configuratie) is er iets misgegaan;
'Installed': het pakketje is succesvol geïnstalleerd en geconfigureerd;
'Removed (configs remain)': het pakketje is verwijderd, maar de configuratiebestanden zijn blijven staan.

5.6.9 Contrib, non-free, non-US

Naast de hierboven genoemde secties als ‘sound’, ‘editors’ en ‘games’, zijn er nog drie ‘speciale’ secties. Dat zijn de secties ‘contrib’, ‘non-free’ en ‘non-US’. Dat vraagt wat meer uitleg.

De Debian distributie bestaat in principe alleen uit software die vrij verspreidbaar/gebruikbaar is. Dat wil zeggen, dat je niet hoeft te betalen om het te gebruiken, en dat je met de programma’s alles mag doen wat je wilt, inclusief veranderen. Verreweg de meeste Linux-software is van dat type. Maar er zijn ook diverse programma’s die je niet mag veranderen, of waarvoor je moet betalen als je ze in een bedrijf gebruikt. Die ‘niet-vrije’ programma’s mogen officieel niet bij de echte Debian distributie horen. Er zijn echter wel veel mensen die ze willen gebruiken. Een oplossing is gevonden door die niet-vrije pakketjes in een sectie ‘non-free’ te groeperen en die ‘non-free’ sectie *niet* bij de standaard Debian distributie te voegen. Je kunt ze echter wel via het Internet downloaden en meestal hoef je voor individueel gebruik (‘thuisgebruik’) niet te betalen. Er staan zoveel programma’s in de sectie ‘non-free’, dat men heeft besloten om daarin sub-secties te maken, die dezelfde namen hebben als de ‘normale’ secties, dus ‘non-free/graphics’, ‘non-free/mail’, ‘non-free/web’, enzovoort.

Dan zijn er ook nog programma’s die niet uit de Verenigde Staten geëxporteerd mogen worden. Dat gaat meestal om cryptografie-programma’s (geheimschrift, ‘versleuteling’), waarvoor in de VS een wet-tegen-export bestaat. Die programma’s mogen echter wel in de VS ingevoerd worden. Bij Debian horen een paar van deze soort programma’s, die in de sectie ‘non-US’ staan. Die sectie hoort ook niet bij de standaard Debian distributie, omdat Debian dan niet meer uit de VS geëxporteerd zou mogen worden. De oplossing is om die ‘non-US’ pakketjes alleen op computers buiten de VS te zetten (bijvoorbeeld in Duitsland) en ze dan via het Internet te downloaden. Op die manier kunnen ze zowel binnen als buiten de VS gebruikt worden. In ‘non-US’ zitten ook een aantal niet-vrije pakketjes, in ‘non-US/non-free’. De rest zit in ‘non-US/main’.

Dan is er nog een bijkomend probleem: bepaalde (op zichzelf vrije) pakketjes uit de ‘normale’ Debian distributie hebben ‘non-free’ pakketjes nodig (in hun dependency’s dus). Als je dat vrije pakketje wilt gebruiken, zul je dus ook dat niet-vrije pakketje nodig hebben. En het is mogelijk dat je voor dat niet-vrije pakketje zult moeten betalen (daarom zit ’t ook in ‘non-free’). Om de gebruikers bijvoorbaat hierop alert te maken, zijn die ‘verdachte’ vrije pakketjes bij elkaar gezet in de sectie ‘contrib’. Maar in de sectie ‘contrib’ staan ook nog andere pakketjes die om de één of andere reden niet ergens anders pasten. Omdat alle programma’s in ‘contrib’ wél vrij zijn, is de sectie ‘contrib’ gewoon onderdeel van de standaard Debian distributie. Ook de ‘contrib’ sectie is zo groot, dat er sub-secties gemaakt zijn.

Tenslotte: alle ‘normale’ pakketjes zitten dus niet in ‘non-free’, niet in ‘non-US’, niet in ‘contrib’, maar in ‘main’.

5.6.10 De Debian CD’s

De standaard (officiële) Debian distributie bestaat dus uit de ‘normale’ secties (de ‘main secties’) en de sectie ‘contrib’. Van al die secties wordt ook nog de sourcecode bijgeleverd, zodat je (als je dat leuk vindt) alles kunt aanpassen en/of opnieuw compileren. Alles bij elkaar past het op zes CDROMs: de ‘main’ en ‘contrib’ secties op de Debian Binary-1, -2 en -3 CDROMs, en de sourcecode op de Debian Source-1, -2 en -3 CDROMs. Er zijn bovendien twee varianten van de Binary-1 en Source-1 CD’s, namelijk met en zonder de ‘non-US/main’ pakketjes. De meeste verkopers leveren de ‘non-US’ versie, behalve verkopers uit de VS die aan buitenlandse klanten moeten leveren.

Omdat veel mensen het lastig vinden om ‘non-free’ en ‘non-US/non-free’ pakketjes via het Internet op te halen, zijn er diverse CDROM-distributeurs die ook CD’s verkopen met daarop die ‘non-free’ en ‘non-US/non-free’ pakketjes.

5.7 Gemakkelijk pakketbeheer met APT

Software op een Debian systeem is eenvoudig te beheren met APT, de *Advanced Package Tool*, in combinatie met een aantal ondersteunende low-level programma’s. APT (spreek uit ‘ept’) kan software zelfstandig van verschillende locaties ophalen en vervolgens installeren of upgraden.

5.7.1 APT’s bronnen: sources.list

De locaties waar APT software-pakketjes vandaan kan halen staan opgesomd in **/etc/apt/sources.list**. Hier staan waarschijnlijk al wat bronnen in, die je tijdens de Debian installatie hebt opgegeven. Het formaat is simpel, maar staat in **man sources.list** op een moeilijke manier uitgelegd. Je kunt een onbeperkte hoeveelheid bronnen opgeven, die allemaal andere locaties kunnen aanwijzen. APT zal zelf uitzoeken waar de nieuwste versies van pakketjes te krijgen zijn; als dat op meerdere locaties is, gaat de voorkeur naar de eerstgenoemde in **sources.list**.

FTP en HTTP

Op www.debian.org onder ‘Download FTP’ kun je adressen vinden van meer dan honderd FTP en HTTP sites met (een deel van) het Debian archief. Als je pakketjes van zo’n server wilt gebruiken, kijk dan eerst met een web-browser wat het precieze pad op die server is. De structuur is bijvoorbeeld als volgt:

```
ftp://ftp.xyz.nl/pub/debian/dists/potato/main/binary-i386/...
ftp://ftp.xyz.nl/pub/debian/dists/potato/contrib/binary-i386/...
```

Merk op dat je (zolang je Debian 2.2 wilt draaien) voor ‘potato’ moet kiezen, en niet voor ‘woody’, ‘sid’ of iets anders. Voor deze (niet-bestaande) server moet je deze regel aan **/etc/apt/sources.list** toevoegen:

```
deb ftp://ftp.xyz.nl/pub/debian potato main contrib
```

De ‘deb’ duidt aan dat er **.deb** pakketjes op die locatie te vinden zijn. Dan volgt het eerste stuk van de locatie, tot aan ‘dists’. De ‘dists’ plakt APT er zelf tussen, gevolgd door het derde item op de regel. De vierde en volgende items worden gebruikt om het pad uit te breiden met verschillende takken, elk item een nieuwe tak. APT plakt er zelf weer ‘binary-i386’ en de juiste sectie-naam achter.

Voor een compleet archief, met main, contrib, non-free en non-US moet je meestal twee regels opgeven, zoals:

```
deb ftp://ftp.xyz.nl/pub/debian potato main contrib non-free
deb ftp://ftp.xyz.nl/pub/debian-non-US potato/non-US main contrib non-free
```

Hoewel je veel servers kunt opgeven, heeft het geen zin om meer dan één ‘officiële’ Debian mirror op te geven, want die zijn (als het goed is) allemaal even up-to-date en worden slechts één keer per dag ververst. Niet alle mirrors hebben trouwens non-US; in dat geval kun je ftp://non-us.debian.org gebruiken, die staat in Nederland.

Lokaal archief

Als je op je eigen harddisk een Debian pakketjes-mirror hebt, kun je die opgeven op een manier die erg lijkt op FTP/HTTP hierboven. Als de structuur bijvoorbeeld het volgende is:

```
/var/ftp/debian/dists/potato/main/binary-i386/...
/var/ftp/debian/dists/potato/contrib/binary-i386/...
```

wordt de regel in **sources.list**:

```
deb file:/var/ftp/debian potato main contrib
```

Ook hier wordt ‘dists’ er dus weer tussen geplakt, en de vierde en volgende items gebruikt voor de verschillende sub-directory’s.

CDROM’s

Je kunt Debian CDROM’s niet met de hand aan **sources.list** toevoegen, want voor CD’s moeten er nog meer andere zaken bijgehouden worden. Een CD kun je toevoegen met **apt-cdrom add**, die de CD scant en een regel toevoegt aan de **sources.list**. Je kunt dat commando herhalen voor alle CD’s die je wilt toevoegen.

Merk op dat hiervoor het mountpunt **/cdrom** moet bestaan, en dat dat in **/etc/fstab** gekoppeld moet zijn aan **/dev/cdrom**, die op zijn beurt weer een symlink moet zijn naar het echte CDROM-drive device (bijvoorbeeld **/dev/hdc**). Zie paragraaf 5.8 voor details daarover.

Als je een CDROM-regel wilt weghalen, kan dat wel gewoon door **/etc/apt/sources.list** te editten.

Wijzigingen doorgeven, Packages files ophalen

Als je de **sources.list** hebt gewijzigd, en/of als er nieuwe pakketjes beschikbaar (zouden kunnen) zijn op een FTP/HTTP mirror of je lokale archief, moet je dat aan APT doorgeven met apt-get update. Die gaat alle **Packages(.gz)** files ophalen en combineren. Pas daarna weet APT wat de allernieuwste versies zijn van alle pakketjes.

Speciale APT bronnen
Op het Internet zijn talloze minder-officiële en niet-officiële plaatsen waar je Debian pakketjes vandaan kunt halen; ze worden wel aangeduid met 'APT repositories'. Merk op dat deze minder geschikt zijn voor modem-gebruikers, omdat enkel het ophalen van de **Packages(.gz)** files al flink wat (dure) tijd kan kosten.

Het is altijd een goed idee om APT te vertellen waar de laatste security-updates te vinden zijn. Die staan namelijk niet in het standaard-archief, maar op een speciale plaats. De juiste **sources.list** regel is op dit moment:

```
deb http://security.debian.org/ potato/updates main contrib non-free
```

Meer informatie daarover staat trouwens op http://security.debian.org.

Andere updates, die nog niet in het officiele archief zijn toegelaten, omdat ze eerst nog getest moeten worden staan in de 'proposed updates'. Hier komen na enige tijd ook de security-updates terecht (maar die blijven ook op security .debian.org staan). Verder staan er andere pakketjes met belangrijke bug-fixes en functionaliteits-uitbreidingen, maar het blijft enigszins onbetrouwbaar zolang het nog niet stabiel verklaard is. Een regel voor APT is

```
deb ftp://ftp.xyz.nl/pub/debian potato-proposed-updates/
```

want bijna alle sites mirroren ook de 'proposed updates'.

Dan is er nog een hele lijst met bronnen te vinden op http://www.internatif.org/bortzmeyer/debian/apt-sources/, maar wees daar wel voorzichtig mee, want er staan heel wat 'onstabiele' zaken bij. En vergeet niet dat je altijd 'potato' moet hebben.

5.7.2 Pakketjes installeren en verwijderen

Als de **sources.list** eenmaal juist is ingesteld (en **apt-get update**!), is het installeren van een nieuw pakketje een fluitje van een cent: **apt-get install <pakketnaam>**. Bijvoorbeeld **apt-get install bsdgames**, en probeer dan, weer als gewone gebruiker, **tetris-bsd** (gebruik j, k, l, spatie) of **worm** (pijltjes). Je zult merken dat **apt** automatisch alle dependency's erbij installeert, in dat geval wordt eerst een bevestiging gevraagd. En als je een pakketje opgeeft dat al geïnstalleerd is, wordt dat netjes overgeslagen.

Een pakketje weghalen is net zo simpel, bijvoorbeeld **apt-get remove bsdgames**. Hierbij wordt meestal een bevestiging gevraagd. Tijdens het verwijderen kunnen er meldingen komen dat bepaalde directory's niet konden worden verwijderd; dat is normaal en komt omdat in die directory's nog andere bestanden staan (meestal configuratie-bestanden die niet worden verwijderd).

Een pakketje compleet verwijderen inclusief configuratie-bestanden gaat niet met **apt-get**. Daar is het low-level programma **dpkg** (spreek uit 'die-pekidzj') voor nodig, bijvoorbeeld **dpkg --purge bsdgames**.

Dat low-level programma kan trouwens ook installeren, maar dan moet de volledige filenaam opgegeven worden. Dus als je bijvoorbeeld (als gewone gebruiker) **bsdgames_2.9-3.deb** van een FTP site hebt gedownload (**dists/potato/main/binary-i386/games**) en in **/tmp** gezet, kun je (als root) installeren met **dpkg --install /tmp/bsdgames_2.9-3.deb**. En direct zo'n file installeren is iets wat **apt-get** weer *niet* kan.

5.7.3 Pakketjes upgraden

Als er een nieuwe versie van een programma beschikbaar is, en jij weet dat toevallig, kun je (na **apt-get update**!) dat éne pakketje upgraden met hetzelfde commando als de installatie, dus bijvoorbeeld **apt-get install bsdgames** of **dpkg --install /tmp/bsdgames_2.9-4.deb**. Je hoeft het pakketje niet eerst te verwijderen, dat gebeurt automatisch.

Maar gelukkig hoef je niet allemaal zelf bij te houden wat nieuw is. Met **apt-get upgrade** (let op, up<u>date</u> is wat anders) worden automatisch alle beschikbare nieuwe versies (gedownload en) geïnstalleerd.

'Upgrade' alleen kan echter geen al-te-ingewikkelde dingen doen; hiermee kun je niet upgraden naar de volgende Debian versie. Dat moet met **apt-get -f dist-upgrade**, die wat slimmer is. Dat is trouwens expres gedaan, want bij 'dist-upgrade' is meestal (veel) meer aandacht nodig; gewoon 'upgrade' kun je over het algemeen zonder nadenken uitvoeren.

Als er een pakketje is wat je niet automatisch wilt laten upgraden, bijvoorbeeld omdat je het zelf op een andere manier gecompileerd hebt, kun je het op 'Hold' zetten. De makkelijkste manier is iets als **echo bsdgames hold | dpkg --set-selections**. De 'Hold' weer opheffen met **echo bsdgames install | dpkg --set-selections**.

5.7.4 Pakket-info

Er is op diverse manieren informatie te krijgen over de pakketjes. Je hoeft daar trouwens nooit root voor te zijn. De eenvoudigste is het vragen van een lijst van alle geïnstalleerde pakketjes met **dpkg --get-selections**.

Met iets als **apt-cache show bsdgames** krijgen we informatie over één pakketje.

Als je alle pakketjes wilt bekijken die APT kent, gebruik dan **apt-cache dumpavail | less**. Zoeken naar een specifiek pakketje gaat dan met bijvoorbeeld '/e: bsdgames', de '/' om te zoeken en de 'e:<spatie>' om (hopelijk) alleen op het 'Package:'-veld te matchen. Probeer het ook eens zonder de 'e:<spatie>'; wel eerst met 'g' of Home terug naar het begin, 'n' voor de volgende match.

Er is een meer visuele manier om alle beschikbare pakketjes te zien. Geef daarvoor eerst als root het commando **/usr/lib/dpkg/methods/apt/update**. Daarna kun je *als gewone gebruiker* (altijd!!) het programma **dselect** starten. Kies met de pijltjestoetsen gelijk voor '[S]elect', Enter, druk op de spatie en toets de 'magische reeks' "ooOOvD" in. Dan krijg je een alfabetische lijst met alle pakketjes, waar je doorheen kunt bladeren. Met een aantal keer 'o' en 'O' kun je ook op acht andere manieren sorteren. De beschrijving onderin kan met 'd' en 'u' omlaag en omhoog, en met 'i' kun je ook andere soorten beschrijvingen onderin krijgen. Op een andere virtuele console kun je dan (als root) met **apt-get** de gewenste pakketjes installeren. **dselect** verlaat je met Ctrl-C. Ik raad je sterk af om **dselect** voor enige andere taak te gebruiken.

De bestanden die bij een geïnstalleerd pakketje horen, zijn te zien met **dpkg -L bsdgames**. Het zoeken van het pakketje dat verantwoordelijk is voor een bepaald bestand kan met **dpkg -S tetris**. Je kunt een bestand ook zoeken in de complete lijst van alle pakketjes, bijvoorbeeld op de Binary-1 CD, met een commando als **zgrep tetris /cdrom/dists/potato/Contents-i386.gz**, de naam van het pakketje staat in de rechter-kolom.

5.7.5 Een eigen pakketjes-archief

Als je zelf pakketjes hebt gemaakt of verzameld, kun je ze individueel met **dpkg -i** installeren, maar ook allemaal in een 'archief' zetten dat door APT gebruikt kan worden. Als je dat in een **public_html/** directory in je homedir zet, is het meteen ook via HTTP voor andere computers toegankelijk, zie ook paragraaf 12.5.2.

Installeer eerst 'dpkg-dev', maak dan bijvoorbeeld een directory **/home/ikke/public_html/DebPack**, en zet daarin de pakketjes neer, eventueel in subdirectory's met willekeurige namen als je dat makkelijk vindt. Ga dan één directory 'hoger' staan, in **/home/ikke/public_html** dus, en geef daar het commando **dpkg-scanpackages DebPack /dev/null | gzip > DebPack/Packages.gz** (en dat steeds doen als er nieuwe pakketjes bijkomen). Voor de **/dev/null** kun je ook een 'override'-file opgeven, die je op een Debian FTP-server in **/debian/project/indices/** kunt vinden, of zelf maakt.

De juiste regels voor **/etc/apt/sources.list** worden dan (let op de spaties en slashes in dit speciale geval):

```
deb file:/home/ikke/public_html/ DebPack/
deb http://kennedy.dorpsnet.nl/~ikke/ DebPack/
```

voor respectievelijk dezelfde computer en een andere computer via HTTP over het netwerk. Vergeet niet **apt-get update**, die waarschijnlijk klaagt dat de **Release** file niet bestaat, maar dat is niet erg.

5.7.6 Meer tips voor APT

Op computers met weinig geheugen (minder dan 32 MB in dit geval) kan het zijn dat het heel lang duurt voordat pakketjes worden geïnstalleerd nadat je het commando daarvoor hebt gegeven, en de harde schijf ratelt dan flink. De oplossing is eenvoudig (als root): **`echo '' > /var/lib/dpkg/available`**. Het is op zulke machines ook niet aan te raden **`dselect`** te gebruiken om de pakketjes te bekijken, maar *als* je dat doet, moet je eerst de 'available file' weer vullen met het 'update' commando uit paragraaf 5.7.4.

Met enkele schakelopties van **`apt`** kun je informatie krijgen wat er precies gaat gebeuren. **`apt-get --simulate install task-gnome-apps`** geeft de volgorde van handelen (maar bij gebruik van CD's is die meestal toch anders). **`apt-get --print-uris install task-gnome-apps`** geeft precies aan welke bestanden zullen worden gebruikt, inclusief lengte en **`md5sum`**. Installeer 'task-gnome-apps' trouwens liever niet nu, dat komt in paragraaf 6.4.1.

Die 'task'-pakketjes zijn trouwens speciaal, omdat ze zelf leeg zijn maar wel een flink aantal dependency's hebben, die dan netjes door APT worden 'meegezogen'. De 'newbie-help' selectie aan het einde van de installatie was ook zo'n 'task'-pakketje, met **`apt-cache show task-newbie-help`** zie je dat die ervoor gezorgd heeft dat **`less`** beschikbaar is. Je kunt 'task'-pakketjes installeren met **`apt-get`** als ieder ander pakketje, maar met **`tasksel`** kan dat ook op een meer 'visuele' manier. Let er echter op dat (ook) hiervoor het 'update' commando uit paragraaf 5.7.4 nodig is.

Als je met **`apt-get`** pakketjes hebt laten downloaden via HTTP of FTP, worden ze in **`/var/cache/apt/archives/`** bewaard, totdat je ze met **`apt-get clean`** laat weghalen. Dus: doe zo af en toe **`apt-get clean`** als je HTTP/FTP gebruikt!

Je kunt ook pakketjes met FTP of HTTP ophalen voor een computer die geen Internet-toegang heeft, maar dat gaat wat omslachtiger. De truc is dat je een 'leeg' archief maakt op de harddisk, met een complete directory-structuur voor alle relevante onderdelen zoals je die op een FTP-site ziet, maar daar slechts één file daadwerkelijk in zet, namelijk de **`Packages.gz`**, die je zelf moet downloaden. Als je deze 'nep-mirror' toevoegt aan **`/etc/apt/sources.list`**, zal **`apt-get update`** gewoon werken, maar installeren daarvan natuurlijk niet. Als je nu bij **`apt-get install`** de **`--print-uris`** optie meegeeft, zullen alle benodigde bestanden in een lijstje worden weergegeven; je kunt die bestanden dan zelf gaan ophalen, op de juiste plaats in de 'nep-mirror' op de harddisk zetten, en dan zal **`apt-get install`** wel werken.

5.8 Mounten van partities, CD's en NFS, /etc/fstab

Partities op de harddisk

In paragraaf 2.5.3 heb ik al kort uitgelegd hoe je een partitie moet mounten. Bijvoorbeeld de eerste logische partitie van de tweede IDE harddisk met **`mount /dev/hdb5 /mnt`**. Maar als je wat meer partities wilt mounten, heb je te weinig mountpunten. Om daar iets aan te doen, moeten er dus wat meer directory's (mountpunten) worden gemaakt. Een mogelijkheid is om bijvoorbeeld extra directory's 'onder' **`/mnt`** te maken, zoals **`/mnt/hdb5`**, maar ik vind het makkelijker om die directory's gelijk in de root te maken, dus **`/hdb5`**. Dan wordt het genoemde commando **`mount /dev/hdb5 /hdb5`**. Op die manier blijft **`/mnt`** vrij voor 'algemeen gebruik'.

Over het algemeen is de Linux kernel prima in staat om vast te stellen welk filesystem op een partitie staat, bijvoorbeeld een ext2 of een msdos filesystem. In een aantal gevallen zul je dat echter moeten specificeren, bijvoorbeeld voor een *vfat* filesystem (lange bestandsnamen in Win95), dat niet te onderscheiden is van een 'normaal' msdos filesystem. Zoiets geef je op met de **`-t`** (type) switch van **`mount`**, bijvoorbeeld **`mount -t vfat /dev/hda2 /hda2`**. Voor meer filesystem-types, zie **`man mount`**.

Het is natuurlijk niet ideaal om iedere keer na het opstarten alle partities met de hand te gaan mounten. Daarom is er een bestand **`/etc/fstab`** waarin alles staat wat automatisch moet worden gemount (inclusief je root- en swappartitie). In dat bestand moet je wel altijd het type filesystem aangeven. Zet onder 'options' steeds 'defaults'. In de kolom 'dump' mag je steeds '0' invullen (tenzij je backups maakt met **`dump`**); in kolom 'pass' vul je *alleen* voor een ext2 filesystem een nummer groter dan 1 in (0 bij niet-ext2), bij het opstarten worden de ext2 filesystems in 'oplopende nummer-volgorde' gescand voor fouten. Zie **`man fstab`** en **`man mount`** voor meer informatie.

Let op: als je bijvoorbeeld een vfat-type partitie bij het opstarten wilt laten mounten (in **`/etc/fstab`** dus), moet je er wel voor zorgen, dat de juiste module dan geladen is. Je kunt dat doen door de juiste module te selecteren met

`modconf` (of met de hand aan **`/etc/modules`** toevoegen), maar je kunt ook het hekje ('#') voor 'auto' in **`/etc/modules`** weghalen, waardoor de modules al eerder op 'auto-load' gezet worden.

CDROM's

Om een CDROM te gebruiken, moet je hem ook mounten. Maak voor het gemak eerst een symlink van **`/dev/cdrom`** naar het echte cdrom-device, bijvoorbeeld door in **`/dev`** het commando **`ln -s hdc cdrom`** te geven (dat is voor het geval de CDROM drive de master is van de secundaire IDE controller, zie ook paragraaf 2.5.3). NB: als je Debian (gedeeltelijk) van CDROM had geïnstalleerd, is de **`/dev/cdrom`** link al automatisch voor je gemaakt en hoef je dat dus niet meer zelf te doen. Doe een CDROM in de drive en geef het commando **`mount /dev/cdrom /cdrom`**. Dan zouden de bestanden op de CD toegankelijk moeten zijn onder de **`/cdrom`** directory.

Tijdens het gebruik van de CDROM zullen de meeste CDROM-drives 'geblokkeerd' worden, zodat je de CD er niet meer uit kunt krijgen. Daarvoor moet je eerst het commando **`umount /cdrom`** of **`umount /dev/cdrom`** gebruiken (hier mag je zowel het device als het mountpunt opgeven). Merk op, dat dit niet zal werken zolang er iemand met de CD bezig is (melding "device busy"); de werkdirectory (**`pwd`**) mag zelfs niet 'op de CDROM' staan.

Veel mensen vinden het lastig om steeds root te moeten worden om een CD te kunnen mounten. Daarom is er een manier om aan alle gebruikers toestemming te geven om een bepaalde schijf (bijvoorbeeld een CD) te mounten. Daarvoor moet je een (normale) regel aan **`/etc/fstab`** toevoegen, maar dan onder 'options' niet alleen 'defaults' maar 'defaults,user,noauto'. Om schrijven tegen te gaan (aanrader bij CD's) kun je ook 'defaults,user,noauto,ro' (van read only) gebruiken. Het filesystem type van een CDROM is trouwens 'iso9660'. Als zo'n regel erin staat, kan iedere gebruiker met **`mount /cdrom`** mounten en met **`umount /cdrom`** unmounten -- de andere benodigde gegevens (zoals het juiste device) worden automatisch in **`/etc/fstab`** opgezocht.

ZIP-disks

Een Iomega ZIP-drive (parallelle poort) gebruiken gaat ook heel goed in Linux. Als de drive aangesloten is, doe je eerst als root **`modprobe ppa`** (of **`imm`** voor nieuwere drives), dan wordt de driver geladen. Het mounten van een ZIP-disk gaat dan met een commando als **`mount -t vfat /dev/sda4 /zip`** (waarbij je natuurlijk eerst de directory **`/zip`** hebt gemaakt). Als de disk gemount is, kan hij niet uit de drive, totdat je 'm weer **`umount`**. NB: haal nooit de kabel los als een disk gemount is! Als je zeker weet dat de drive altijd beschikbaar is als je opstart, kun je 'ppa' toevoegen aan **`/etc/modules`**, dan hoef je de driver niet meer met de hand te laden, en kun je ook een 'user'-regeltje aan **`/etc/fstab`** toevoegen. Als de kernel geen andere SCSI-drivers 'ingebakken' heeft (de standaard-kernel heeft dat wel!), gaat het nog makkelijker: maak een bestand **`/etc/modutils/zip`** met daarin

```
alias scsi-hostadapter ppa
```

(of 'imm', en niet vergeten daarna **`update-modules`** draaien), en geen 'ppa' in **`/etc/modules`**; dan wordt de ZIP-drive-driver pas geladen als je **`/dev/sda4`** aanspreekt bij het mounten, en hoeft de drive dus niet meer aanwezig te zijn tijdens het opstarten.

NFS

Je kunt ook een directory van een NFS-server mounten, met een commando als **`mount server.ergens.com:/dir1/dir2 /mnt`**. De te-mounten-directory moet wel bij de NFS-server in **`/etc/exports`** staan (of iets wat daarmee overeenkomt); als je dat bestand veranderd hebt, moet je dat doorgeven met het commando **`/etc/init.d/nfs-server reload`**. Tip: gebruik bij het mount-commando **`-o rsize=8192,wsize=8192`**, dan wordt het netwerk-transport (veel) sneller. Ook een NFS-mount-regel kun je op een voor de hand liggende manier in **`/etc/fstab`** zetten, het filesystem type is 'nfs'. Als de NFS-server een Debian 2.2 systeem is, moet het pakketje 'nfs-server', waar **`nfsd`** en **`mountd`** in zitten, geïnstalleerd zijn.

Een voorbeeld **`/etc/fstab`**:

```
# /etc/fstab: static file system information
#
# <filesyst><mountpt><type>   <options>                   <d><p>
/dev/hda5   /         ext2     defaults,errors=remount-ro  0  1
/dev/hda6   none      swap     sw                          0  0
/dev/hda1   /hda1     msdos    defaults                    0  0
/dev/hda2   /hda2     vfat     defaults                    0  0
/dev/cdrom  /cdrom    iso9660  defaults,ro,user,noauto     0  0
/dev/fd0    /floppy   auto     defaults,user,noauto        0  0
/dev/sda4   /zip      vfat     defaults,user,noauto        0  0
proc        /proc     proc     defaults                    0  0
```

Hier is hda5 de root partitie en hda6 de swappartitie; als er fouten op het root filesystem worden gesignaleerd, wordt dat automatisch readonly gemount. Op hda1 staat een DOS versie, op hda2 staat Windows 95. De CDROM is mountbaar door iedere gebruiker; dat geldt ook voor de floppy drive en de ZIP-disk. Het ‘proc’ filesystem is een virtueel filesystem, waarin de kernel allerlei proces-informatie neerzet. (NB: ‘proc’ is verplicht, en kost geen schijfruimte.)

Lees voor (veel) meer informatie de manpages van **`mount`** en **`fstab`**.

5.9 Inlogzaken, issues en motd

Na een tijdje zul je de message of the day wel uit je hoofd kennen en is die lap tekst overbodig geworden. Je kunt **`/etc/motd`** dan drastisch inkorten. De bovenste regel wordt echter elke keer dat je de computer aanzet, automatisch bijgewerkt, zodat steeds de actuele kernel-versie wordt gemeld. Dat kun je uitzetten door in **`/etc/default/rcS`** de regel EDITMOTD=yes te veranderen in EDITMOTD=no. De motd is de ideale plaats om algemene mededelingen te doen (bijvoorbeeld onderhouds-planning).

Je kunt ook de systeem-aanduiding boven de login-prompt wijzigen. Standaard staat daar iets als “Debian GNU/Linux 2.2 kennedy tty1”. De bestanden die je kunt wijzigen zijn **`/etc/issue`** (voor de virtuele consoles) en **`/etc/issue.net`** (voor virtuele terminals via het netwerk met **`telnet`**). Ik heb zelf het volgende in **`/etc/issue`**:

```
<lege regel>
Debian GNU/\s 2.2 \n (\l)
<lege regel>
```

en in **`/etc/issue.net`** iets als:

```
<lege regel>
Debian GNU/%s 2.2 kennedy (pts/%t)
<lege regel>
```

(/dev/pts/3 bijvoorbeeld is de derde pseudo-terminal, zie ook paragraaf 2.5.4). Meer hierover in **`man issue`**, **`man getty`** (onder ‘ISSUE ESCAPES’) en **`man issue.net`**.

Wat het inloggen betreft, kun je nog veel meer wijzigen in **`/etc/login.defs`**. Het is de moeite waard om dat bestand eens door te lezen; zie ook **`man login.defs`**.

6. De grafische omgeving: het X Window System

Het *X Window System* is de grafische omgeving van Linux. Door middel van vensters, knoppen en invoervelden wordt de communicatie met de computer wat vergemakkelijkt. Vroeger was het vaak een hele toer om X aan de praat te krijgen. Tegenwoordig is er wat dat betreft gelukkig flink wat verbeterd, al moet je niet verwachten dat je totaal ongeschonden door de configuratie heen zult komen. Ik zal uitleggen hoe X ongeveer in elkaar zit, hoe je het moet installeren en wat je er zoal mee kunt doen.

6.1 Overzicht van X: servers, window managers en applicaties

Eigenlijk kun je X net zo ingewikkeld uitleggen als je maar wilt. Ik zal het hier heel simpel houden, maar daarmee vertel ik dus lang niet alles.

X is een *client-server* omgeving. De *server* biedt een aantal *services* (diensten) aan. Die services bestaan grofweg uit de mogelijkheid om ‘dingen’ in een grafische omgeving te doen. Een *client* (‘klant’) maakt van die services gebruik om een bepaald doel te bereiken. Alle acties van een client vinden plaats in één of meer *windows* (*vensters*), die eventueel over het scherm verplaatst kunnen worden. Zo’n actie (of service) is bijvoorbeeld het trekken van een lijn, of het neerzetten van een reeks letters. De client moet aan de server vragen om bijvoorbeeld een lijn met dikte *x* en kleur *y* neer te zetten van punt *a* tot punt *b*. De server kent alleen basis-opdrachten; er zijn library’s beschikbaar die het voor programma’s makkelijk maken om bijvoorbeeld *buttons* (‘knoppen’) en *pull-down menu’s* te gebruiken -- de library vertaalt een opdracht als ‘zet een OK-button daar-en-daar neer’ naar een reeks lijn- en tekst-opdrachten waar de server iets mee kan. De server spreekt de videokaart-hardware direct aan, daarom hebben diverse videokaart-families een eigen X-server, bijvoorbeeld de S3 kaarten, maar er is ook een algemene SVGA server die praktisch alle SVGA-kaarten aankan (maar niet altijd optimaal).

Zoals gezegd, ‘leeft’ een programma (of applicatie), bijvoorbeeld een tekstverwerker of een rekenmachine, in een window. Dat programma kan z’n eigen ‘input’ verwerken, zoals ingetypte letters en een ‘klik op de knop’. Maar een programma heeft geen mogelijkheden om het venster ‘interactief’ te laten verplaatsen, of bijvoorbeeld te minimaliseren/maximaliseren. Dat soort mogelijkheden worden door de *window manager* georganiseerd. Een window manager voorziet alle windows van een rand (*border*, voor het vergroten/verkleinen), een titelbalk (*title bar*, voor het verplaatsen) met titel en vaak knopjes (voor bijvoorbeeld minimaliseren/maximaliseren). Als er bijvoorbeeld op de minimaliseer-knop wordt gedrukt, zorgt de window manager dat dat ook gebeurt; de applicatie merkt daar totaal niets van. Omdat smaken verschillen, zijn er vele verschillende window managers beschikbaar, bijvoorbeeld **`fvwm`**, **`twm`**, **`9wm`**, **`afterstep`**, **`enlightenment`** en **`windowmaker`**. Alle window-managers hebben een verschillende look-and-feel en je kunt kiezen welke je het prettigste vindt werken. Als je een Windows 95-achtige omgeving wilt, kun je het beste **`fvwm95`** gebruiken. Merk op dat window managers ook gewone programma’s zijn -- je kunt een window manager bijvoorbeeld killen en een andere opstarten zonder uit X te hoeven gaan. Als er geen window manager draait, zijn de windows simpelweg ‘border-loos’ en kunnen ze niet verplaatst worden.

6.2 Installeren van X

In het onderstaande wordt de installatie van het X Window System behandeld. Dit is één van de mogelijke manieren, en zou in de meeste gevallen vrijwel direct tot resultaat moeten leiden. Het is het verstandigste om deze keer *niet* iets anders te doen dan hier beschreven staat, want daarmee maak je het alleenmaar lastiger. Mocht de installatie onverhoopt toch helemaal fout gaan, begin dan opnieuw vanaf hier (je hoeft dus niet heel Debian opnieuw te installeren).

6.2.1 Installeren van de basis-onderdelen

Log in als root, en geef het commando **`apt-get install xfonts-75dpi xf86setup xserver-vga16 xfonts-scalable xterm xcontrib xviddetect twm pciutils menu gpm`**; als je de Binary-2 CD hebt, of van een FTP site installeert, zet er dan **`fvwm95 fvwm95-icons xaw95g`** achter. Na een bevestiging wordt geïnstalleerd; op een ‘kaal’ systeem is dat 25 tot 30 MB.

Tijdens de configuratie, die na de installatie automatisch begint, zal ‘xserver-vga16’ waarschijnlijk vragen of “VGA16 de default X server moet worden”, antwoord ‘y’ gevolgd door Enter. Op de vraag “create the XFree86 configuration file” geef je ‘n’ en Enter. Als ‘gpm’ vraagt “change anything?”, geef dan ‘y’; “where is your mouse”: type ‘/dev/ttyS0’ voor seriële muis op COM1, ‘/dev/psaux’ voor PS/2 muis; “what type”: ‘bare’ voor seriële muis, ‘ps2’ voor PS/2 muis; Enter, Enter Enter, ‘n’ Enter.

Dan moeten we nog twee bestanden aanpassen (als dat nodig is), en enkele andere administratieve zaken regelen. De eerste twee regels van **/etc/X11/Xserver** moeten luiden:

```
/usr/bin/X11/XF86_VGA16
Console
```

Verander het bestand zonodig, en let op de hoofdletters (voor het gebruik van de editor **ae**, zie paragraaf 4.10). Als het bestand **/etc/gpm.conf** bestaat, moet je de regel

```
repeat_type=<iets>
```

geheel verwijderen, en vervolgens het commando **/etc/init.d/gpm restart** geven. Doe dan **update-alternatives --config x-window-manager**, en kies het nummer dat voor **/usr/bin/X11/fvwm95** staat, of (als **fvwm95** er niet is) dat voor **/usr/bin/X11/twm** staat, of misschien is er zelfs geen keuze.

Tenslotte moet je met **mv /etc/X11/XF86Config /etc/X11/XF86Config.old** een eventueel bestaande oude configuratie verwijderen, omdat die flink roet in het eten kan gooien (let op de hoofdletters).

6.2.2 Configuratie van de voorlopige VGA16 server

We gaan nu beginnen met de configuratie van de VGA16 server, die we voorlopig even zullen gebruiken.

Geef als root het commando **XF86Setup** (let op de hoofdletters!). Misschien zal er gevraagd worden of je het bestaande XF86Config bestand wilt gebruiken voor de 'default' waarden, antwoord daar 'n'. (NB: dat mag eigenlijk niet gevraagd worden, want dat bestand zou je zojuist verwijderd moeten hebben.)

Na één Enter (ook al wordt dat schijnbaar twee keer gevraagd) zal de grafische modus worden opgestart. Dat kan enige tijd duren (vooral op langzame computers), maar zou nergens problemen op mogen leveren. Na een tijdje zie je een gestippeld-grijze achtergrond met een groot kruis in het midden en daarna verschijnt een menu-achtig scherm met groot 'The XFree86 Project, Inc.' bovenin. In het midden staat een soort 'I', dat is de muispointer, maar die hoeft nog niet te werken.

Ga met de TAB-toets (Shift-TAB gaat de andere kant op) naar de knop 'Mouse' linksboven en druk daar op Enter. Het updaten van de selectie kan vrij langzaam zijn; misschien loop je 'Mouse' voorbij, dan met Shift-TAB terug. Lees eventueel het introductiescherm, loop dan met TAB naar 'Dismiss' (onderin) en druk op Enter. Bovenin zie je nu een flinke lijst met muis-protocollen. Ga met de TAB naar het juiste protocol en druk daar op Enter. Voor de meeste 'normale' muizen (ook bijv. Logitech Pilot Mouse) is het Microsoft protocol geschikt; voor PS/2 muizen natuurlijk PS/2. Loop dan met TAB naar de lijst (*niet* het invoerveld) onder 'Mouse Device' (links midden), kies met de pijltjestoetsen op/neer het juiste device en geef Enter. Het device zal meestal **/dev/ttyS0** (COM1) of **/dev/psaux** (PS/2 muis) zijn. Na Enter staat de selectie al op de knop 'Emulate3Buttons', die je het beste aan kunt zetten met nog een Enter. 'ChordMiddle' mag uit blijven, 'Resolution' mag op 'Medium' blijven staan. De 'Baud Rate' is voor seriële muizen meestal 1200, maar dat kan voor jouw muis misschien anders zijn, selecteer dan de juiste optie. 'Buttons' laat je op '3' staan, tenzij je er meer hebt. 'Sample Rate' mag op '0' blijven staan en 'Emulate3Timeout' op '50'. Loop met de TAB naar de 'Apply' knop en druk dan op Enter. Probeer je muis door te bewegen en in het 'muisvakje' (rechtsmidden) op de muis-knoppen te drukken (probeer ook rechtse knop en linkse knop tegelijk...). Als het niet werkt, zul je iets aan de instellingen moeten veranderen (en dan steeds weer 'Apply'!); als alleen de middelste muisknop niet werkt, probeer dan een ander protocol, of eventueel 'ChordMiddle' bij de protocols die dat ondersteunen. Oudere muizen met een MS/PC keuze-knopje kun je het beste op 'PC' zetten, gebruik dan het 'MouseSystems' protocol, meestal zonder 'Emulate3Buttons'. Merk op dat bij sommige 'goedkope' muizen de middelste knop niet werkend te krijgen is.

Als de muis werkt, gaat het allemaal wat makkelijker. Klik op de knop 'Keyboard' (bovenin). Onder 'Model:' staat 'Generic 101-key PC' met daarnaast een knop met een pijltje naar beneden. Klik op dat pijltje, en er verschijnt een pull-down keuzelijst. Kies het juiste model, meestal 'Generic-101-key PC' (voor niet-Win95 toetsenbord) of 'Generic 104-key PC' (voor een toetsenbord met Win95 toetsen). De 'Layout' kan waarschijnlijk op 'U.S. English' blijven staan en de 'Variant' op niks. Voor de 'Available options' (rechts) kun je alles op 'default' laten staan. Klik op de grote 'Apply' knop (onder het plaatje) om dit onderdeel af te sluiten.

Kies dan bovenin voor 'Card'; zoek in de lijst 'Generic VGA compatible' en klik op die regel. (De speciale kaartserver gaan we later installeren).

Kies nu bovenin voor 'Monitor' en klik op 'Standard VGA, 640x480 @ 60 Hz' (ook als je een andere monitor hebt).

De volgende optie bovenin is 'Modeselection'. Klik in het lijstje middenin alleen op '640x480', en in het rijtje onderin op '8bpp'.

Onder 'Other' hoeft niets veranderd te worden. Kies dus midden-onder voor 'Done'. Klik op de 'Okay' knop in het venster om de configuratie te beëindigen.

Vervolgens wordt er even teruggegaan naar tekstmode, en wordt meteen weer de grafische mode opgestart. Kies daar voor 'Save the configuration and exit' en weer eens 'Okay'.

6.2.3 Voor het eerst in X

De configuratie van de voorlopige X server is nu voltooid, je kunt als root uitloggen.

Log vervolgens in als gewone gebruiker en geef het commando **startx**. Je krijgt dan weer de inmiddels bekende stippeltjes-achtergrond te zien en het kruis (de muispointer) in het midden. Even later wordt het scherm lichtblauw, en verschijnt een Start-balk onderin het scherm, met daarboven een grote-knoppen-balk. Klik op de 'Start'-knop en kies het eerste item, 'xterm'. En ziedaar, je eeste venster! [Geen fvwm95 maar alleen twm: het scherm wordt niet blauw en er is geen Start-balk. Druk de linkse muisknop in (ergens op de stippel-achtergrond) en houd hem ingedrukt. Uit het keuzemenu kies je 'Debian', 'XShells' en 'Xterm' door er overheen te bewegen, laat de knop los als 'Xterm' geselecteerd is. Sleep het verschijnende lijnenpatroon tot het helemaal zichtbaar is en klik het dan vast.]

Als X niet wil starten ("no valid modes found"), moet je (als root) het bestand **/etc/X11/XF86Config** editten. Zoek naar Section "Monitor" en maak van de betreffende regels "HorizSync 30-33" en "VertRefresh 59-61". Log daarna weer uit als root, en probeer het opnieuw (als gewone gebruiker).

Je eerste venster is dus een *xterm*, een <u>term</u>inal-emulator voor <u>X</u>. Bovenin zie je een prompt verschijnen. Probeer eens een commando (bijvoorbeeld **pwd**) in te typen. Als er geen letters op het scherm komen, is het venster blijkbaar niet geselecteerd; klik dan ergens op het verster. [Twm: zet de muispointer in het venster.]

Je kunt een tweede xterm opstarten zoals je net gedaan hebt, via de menu's, maar ook door in de eerste xterm het commando **xterm &** te geven. De nieuwe xterm wordt 'in de achtergrond' gestart, waardoor je ze beide kunt blijven gebruiken.

Een xterm sluit je af door **exit** in te typen. Als je alle xterms hebt afgesloten, blijft X nog steeds actief, omdat de window manager (momenteel bijvoorbeeld **fvwm95**) nog draait. Die kun je afsluiten door op de 'Start'-knop te klikken, dan het onderste item, 'Exit Fvwm', gevolgd door 'Yes, Really Quit'. Dan wordt X afgesloten en kom je weer terug in tekstmode. [Twm: op de stippeltjes-achtergrond links-klikken en "Exit" - "Yes, really quit" kiezen.]

In ieder geval werkt X blijkbaar -- maar nog lang niet optimaal, omdat we nog de zeer beperkte VGA16-server gebruiken. Dat moet anders.

6.2.4 Installeren van de echte server

Log weer in als root en geef het commando **SuperProbe -no16**. Dat zal eerst vijf seconden wachten en vervolgens proberen te bepalen welke videokaart je hebt. Het scherm kan daarbij korte tijd zwart blijven. Als er geen videokaart gevonden wordt, of als de computer 'hangt', kun je **SuperProbe** nog een keer zonder **-no16** proberen. Ook **xvid-detect** en **lspci** kunnen resultaat opleveren. Als er iets gevonden is, schrijf dan alles (Chipset, Memory, RAMDAC etc.) ergens op een papiertje; het is niet erg als er niets gevonden is, maar dan moet je zelf weten welke videokaart je hebt.

Geef dan het commando **XF86Setup**. Antwoord nu 'y' op de vraag "Would you like to use the existing XF86Config file for defaults?", en na nog één Enter kom je weer in de grafische configuratie terecht. Kies daar (met de muis, die nog steeds zou moeten werken) gelijk voor 'Card' (bovenin), en klik dan op de knop 'Card List' (rechts onder). Zoek in de grote lijst naar jouw videokaart, die je zojuist met bijvoorbeeld **SuperProbe** hebt gedetecteerd (of zelf hebt moeten uitzoeken). Klik op die kaart, en je zou onderin een regeltje moeten zien dat meldt dat de server voor jouw kaart niet is geïnstalleerd. Daar staat ook welke server er nodig is: "Please abort, install the XXX server..." -- schrijf die naam ("XXX" dus) ergens op. Klik op de knop 'Abort' (onderin), en je komt weer in tekstmode terecht, zonder iets veranderd te hebben.

Het kan zijn, dat je een hele nieuwe videokaart hebt, die nog niet in het lijstje voorkomt. Kijk dan op http://www.debian.org/~branden, waar misschien Debian pakketjes staan van een nieuwere X versie, die wellicht nog 'onstabiel' is, maar waar wel diverse nieuwe kaart-servers bij zitten. Let er wel op dat je 'potato' versies hebt, en geen 'woody' of 'sid'; de juiste URL direct naar de pakketjes is waarschijnlijk http://www.debian.org/~branden/potato/. Stel eventueel vragen op de 'debian-user' mailing list, zie paragraaf 4.21. Als je dat niet wilt, kun je ook de VGA16-server blijven gebruiken, en de rest van deze paragraaf overslaan.

Geef vervolgens het commando **apt-get install xserver-<servernaam>**, <servernaam> heb je zojuist genoteerd, maar moet nu met kleine letters geschreven worden (dus voor 'SVGA' bijvoorbeeld **apt-get install xserver-sgva**). Als gevraagd wordt of de nieuwe server de default server moet worden, moet je 'y' antwoorden.

Het kan gebeuren, dat de nieuwe server niet standaard (default) aan wordt gezet. Kijk even met **cat /etc/X11/Xserver**, daar mag geen **/usr/bin/X11/XF86_VGA16** op de eerste regel staan, maar **/usr/bin/X11/XF86_<servernaam>** (met servernaam nu weer in hoofdletters), bijvoorbeeld **/usr/bin/X11/XF86_SVGA**. Zo nodig moet je dat met **ae** met de hand veranderen (zie ook paragraaf 4.10).

Start weer **XF86Setup** (weer 'y' bij die 'defaults' vraag) en klik bij 'Card' - 'Card List' weer op jouw videokaart; dat zou nu geen problemen meer op moeten leveren.

Kies dan bovenin voor 'Monitor', zoek in de lijst in het midden jouw type monitor, en klik daarop. Als je niet weet wat je moet kiezen, is 'Standard VGA, 640x480 @ 60 Hz' de optie die altijd zou moeten werken, of anders 'Super VGA, 800x600 @ 56 Hz'. In de gebruiksaanwijzing van je monitor staan vaak verticale en horizontale frequenties, in Hz resp. kHz (meestal in een "specificaties tabel"), deze staan ook boven in het scherm genoemd. Let erop, dat de ingestelde waarden de specificaties niet overschrijden. Je kunt eventueel ook met de hand de waarden opgeven in de "Horizontal" en "Vertical" invoervelden; druk dan wel steeds op Enter na het veranderen van de waarden (en de pijltjes-toetsen werken niet). In het algemeen is het verstandig om een range op te geven, en geen aparte waarden; en een paar tienden buiten de specs kan meestal geen kwaad. Als je later toch niet tevreden mocht zijn met de instelling, kun je die weer met **XF86Setup** wijzigen.

De volgende optie bovenin is 'Modeselection'. In het lijstje middenin kun je de modi aanklikken die je wilt gaan gebruiken -- meestal zullen dat de bovenste twee of drie zijn. Onderin kun je aangeven welke kleurmode je standaard wilt gaan gebruiken: '8bpp' voor 256 kleuren, '16bpp' voor 65536 kleuren, '24bpp' en '32bpp' voor 16.7 miljoen kleuren ('true color'). Let op: als je 'teveel bpp' opgeeft, kan het gebeuren, dat de X server niet wil starten; draai dan opnieuw **XF86Setup** en kies een lagere waarde.

Klik dan op 'Done' (onderin) en 'Okay'. Er wordt weer teruggegaan naar tekstmode en vervolgens weer naar grafisch (als de server goed werkt, tenminste).

Mocht de monitor vreemd gaan doen (rollen enz.), direct op Ctrl-Alt-Backspace drukken, en terug in tekstmode gelijk Ctrl-C. Start **XF86Setup** opnieuw en probeer minder extreme waarden voor de horizontale en verticale frequenties (en waarschijnlijk moet je de videokaart ook weer goed zetten). Denk er ook om, dat gebruiksaanwijzingen van monitoren niet altijd geheel correct zijn (soms negatief, soms positief!); soms loont het de moeite om diverse vertalingen met elkaar te vergelijken. Als de server helemaal niet start, is er waarschijnlijk een probleem met de 'bpp' instelling (of de foute videokaart gekozen?). Een hoogst enkele keer kan het nodig zijn om in **XF86Setup** bij 'Card' expliciet de Chipset, RAMDAC of ClockChip op te geven, deze worden normaal automatisch gedetecteerd, maar soms gebeurt dat niet correct.

Als je weer in 'de betere' grafische mode zit, kun je het scherm eventueel nog wat verschuiven/vergroten/verkleinen. LET OP: DIT IS GEVAARLIJK! Klik daarvoor op de knop 'Run xvidtune'. In het 'WARNING'-venster klik je gelijk linksonder op 'OK'. Nu kun je met de knoppen 'Left' t/m 'Taller' het één en ander aanpassen. Die aanpassingen worden niet direct uitgevoerd, maar kunnen met de 'Test'-knop voor 5 seconden uitgeprobeerd worden. Als alles naar wens is, klik je op 'Apply' en 'Quit'; om zonder wijziging te verlaten: geen 'Apply' maar gelijk 'Quit'. Als je meerdere resoluties hebt gekozen, kun je wisselen met Ctrl-Alt-NumeriekePlus en Ctrl-Alt-NumeriekeMin. Ook die kun je eventueel met 'Run xvidtune' nog iets aanpassen.

Klik op de knop 'Save the configuration and exit' en de nieuwe X server is geconfigureerd.

Als je uit de grafische mode terug komt in tekstmode, kan het gebeuren dat de letters er een beetje vreemd uitzien; geef dan het commando **consolechars -f iso01.f16** (eventueel 'blind' intypen).

Probeer als niet-root met **startx** of alles goed werkt. Als de X server niet wil starten of als één van resoluties niet werkt, terwijl alles tijdens de configuratie goed ging, probeer dan de "HorizSync" en "VertRefresh" waardes in **/etc/X11/XF86Config** een beetje uit te breiden, zie ook de vorige paragraaf.

Als je later nog eens wat aan de configuratie wilt veranderen, kun je weer **XF86Setup** gebruiken.

6.2.5 Andere configuratiemogelijkheden

NB: Als alles al goed werkt, kun je deze paragraaf waarschijnlijk overslaan.

Het kan nodig zijn om de tekstmode-configuratie te gebruiken. Dat gaat (als root) met het programma **xf86config** (kleine letters). De vragen die gesteld worden, zijn grotendeels gelijk aan die van **XF86Setup**. Antwoorden zul je zelf moeten verzinnen, maar dat zal niet al te moeilijk zijn door de duidelijke uitleg.

Als je een hele nieuwe videokaart hebt, waarvoor nog geen server in een Debian package zit, moet je die in **/usr/bin/X11/** zetten, bij de andere **XF86_XXX** servers. Pas de eerste regel van **/etc/X11/Xserver** aan en configureer dan met **xf86config**. Kies er *niet* voor om de videokaarten-lijst gepresenteerd te krijgen, maar kies voor de algemene 'accelerated' driver (optie 4). Kijk bij http://www.xfree86.org voor de allernieuwste servers, maar merk op dat die niet gegarandeerd werken met Debian 2.2!

Eventueel kun je ook met een tekst-editor in het bestand **/etc/X11/XF86Config** het één en ander wijzigen, de uitleg erbij is zeer uitgebreid (die uitleg is er *alleen* als je eerst **xf86config** hebt gedraaid; **XF86Setup** haalt die uitleg weer weg).

Uitleg over opties in **XF86Config** staat in **man XF86Config**; specifieke documentatie en opties voor veel videokaarten staan in **/usr/share/doc/xserver-common/README.*.gz**; het is altijd verstandig de relevante documentatie eens door te lezen.

Tenslotte: als de muispointer er niet goed uitziet, 'ernaast' lijkt te staan, vreemd lijkt te verspringen, of buiten het normale beeld blijkt te kunnen gaan, is er waarschijnlijk sprake van een veel-voorkomend probleem met de hardware pointer (*hardware cursor* in de documentatie). Forceer dan het gebruik van de software pointer door een regel

```
Option   "sw_cursor"
```

toe te voegen aan de Section "Device" van **/etc/X11/XF86Config**.

6.3 Gebruik van X

In deze paragraaf worden kort een aantal aspecten van het gebruik van het X Window System besproken.

6.3.1 Starten van X

Je hebt X al eens opgestart, met **startx**. Daarmee wordt de X server gestart en een window manager. Je kunt ook starten met **xinit**, daarmee wordt de window manager 'overgeslagen', maar wel een xterm gestart; een window manager mag je dan zelf starten (bijvoorbeeld **fvwm95 &** in de xterm).

X 'draait' op de eerste vrije virtuele console, meestal dus tty7. Als je in X Ctrl-Alt-F1 t/m Ctrl-Alt-F6 geeft, kun je terug naar de tekst-consoles (X blijft gewoon draaien). Vanuit een tekst-console kun je met LinkseAlt-F7 weer naar X terug. (Let op het verschil: Ctrl-Alt-F*x* en LinkseAlt-F*x*.)

Het wordt overigens sterk afgeraden om als root X te draaien. Als gewone gebruiker **su** gebruiken vanuit een xterm verdient de voorkeur. Dit omdat root in principe zo weinig mogelijk programma's moet/mag draaien.

6.3.2 Programma's en nieuwe vensters

X is ideaal om multitasking mee te doen. Dus als je vanuit een xterm een programma start, dat een eigen window heeft, wil je dat de prompt na het starten van dat programma meteen weer terugkomt, zodat je weer andere programma's kunt starten. Ofwel, die programma's draaien 'in de achtergrond' (in Linux betekent dat *niet* dat ze langzamer draaien!) en je moet ze starten met **programma &** (zie paragraaf 4.18). Een tweede xterm start je dus met **xterm &**.

Voor wat andere leuke programma's geef je bijvoorbeeld **xeyes &**, **xcalc &**, **xclock &**, **xload -update 1 &** en **xman &**.

Je kunt de kleur van de achtergrond wijzigen met bijvoorbeeld **xsetroot -solid darkblue**; de kleuren staan opgesomd in **/usr/lib/X11/rgb.txt**. Als je zelf de RGB-waarde weet, gebruik dan iets als **'#23CD89'** (inclusief quotejes) als kleur.

Vaak kun je aan programma's opgeven waar ze hun venster moeten neerzetten en hoe groot dat moet zijn. Dat gaat met de **-geometry** switch. Bijvoorbeeld: **xterm -geometry 80x24+0+0 &** zet een xterm met 80 kolommen en 24 regels neer op positie (0,0), dus helemaal linksboven. En **xeyes -geometry 50x50+700+500 &** zet kleine oogjes ergens rechtsonder op het scherm (bij 800x600-mode). **xeyes -geometry 50x50-50-50 &** heeft hetzelfde effect, maar werkt ook op andere schermen dan 800x600, omdat nu van de rechter- en de onderrand wordt geteld.

6.3.3 Vensters verplaatsen, veranderen en sluiten

Acties met vensters worden door de window manager geregeld, en elke window manager pakt dat anders aan. Met **fvwm95** gaat dat min of meer zoals je in Windows gewend bent. Voor verplaatsen werken echter ook de venster-randen (en niet alleen de titelbalk), en voor vergroten/verkleinen werken alleen de hoekpunten. Voor minimaliseren/ maximaliseren zijn de gebruikelijke knoppen beschikbaar. Een window sluiten kan met de sluit-knop of dubbelklikken op het icoontje linksboven, maar als het programma een eigen afsluit-optie heeft, verdient die de voorkeur (bijvoorbeeld **exit** in een xterm of 'File' - 'Exit' in Netscape). Als een venster niet dicht wil, moet je het proces wat erin loopt, stoppen met **kill**, bijvoorbeeld vanuit een (andere) xterm.

6.3.4 Knippen en plakken

Misschien heb je al gemerkt dat je met de muis kunt selecteren door over de tekst in een xterm te bewegen met de linkse muisknop ingedrukt. Je kunt een bestaande selectie uitbreiden met de *rechtse* muisknop. De tekst die geselecteerd is, wordt meteen in de *primary paste buffer* van X geplaatst (vergelijkbaar met Klembord/Clipboard van Windows), daar is geen extra handeling voor nodig. Plakken kan met de middelste muisknop (of links en rechts tegelijk), maar vaak ook met Shift-Insert.

Soms werkt selecteren/plakken niet in een xterm, omdat het programma dat erin draait zelf al iets met de muis doet. Houd in dat geval de Shift-toets ingedrukt tijdens de muisacties.

De primary paste buffer kan alleen tekst opslaan. Om willekeurige data op te slaan zijn er ook nog een aantal *cut buffers*, waarvan normaal alleen de eerste (CUT_BUFFER0) wordt gebruikt. Deze zijn niet direct met de muis te bedienen, maar lopen meestal via een 'Edit' - 'Copy' / 'Paste' menu van tekstverwerkers. Als je tekst via de cut buffer transporteert, is die tekst meestal eerst geselecteerd geweest, en staat die dus ook in de primary paste buffer, maar dan zonder enige opmaak.

6.3.5 Instellingen van een xterm

Een xterm heeft ook diverse instellingen. Een aantal kun je wijzigen door te Ctrl-klikken in het venster; Ctrl-Rechtse-Knop wordt het meest gebruikt om tijdelijk een andere lettergrootte te krijgen.

Andere instellingen moeten worden opgegeven als de xterm wordt gestart. Als je in de xterm **ls -alF /dev** geeft (kleur niet mooi? dan voorlopig **--color=no** erachter) kun je ook met Shift-PageUp/PageDown scrollen, maar niet zover. Als je xterm had gestart met **xterm -sb -sl 1000** heb je wat meer scroll-back buffer. De **-sb** zorgt voor een schuifbalk (scrollbar), met de *middelste* muisknop kun je die slepen (links en rechts werken ook, maar anders). **xterm -ls** geeft een login shell die iets anders wordt behandeld als een 'normale' shell (zie paragraaf 7.9), de directory is nu niet meer gekleurd. Meer uitleg met **xterm -help** of **man xterm**.

6.3.6 Afsluiten van X

Als je X met **startx** hebt gestart, wordt X afgesloten als je de window manager stopt, in fvwm95 met de 'Exit Fvwm'-optie uit het 'Start'-menu. Let op: alle 'openstaande' vensters worden dan automatisch gesloten. Merk trouwens op dat het 'Start'-menu ook bereikbaar is door ergens op de achtergrond te klikken.

Als je **xinit** hebt gebruikt, wordt X afgesloten als je in de allereerste xterm het commando **exit** geeft.

Een andere, minder nette manier is Ctrl-Alt-Backspace. Die toetscombinatie kun je bijvoorbeeld gebruiken als je monitor problemen heeft met een bepaalde videomode, of als de window manager niet meer op de muis reageert.

Als je in tekstmode ziet dat de letters er vreemd uitzien, heeft de X server blijkbaar wat karakter-geheugen gebruikt en incorrect teruggezet (bekend probleem). Zoiets kun je herstellen met het commando **consolechars -f iso01.f16** (eventueel 'blind' intypen).

6.3.7 Vensters vanaf andere computers

De X server en de clients hoeven niet op dezelfde computer te draaien, de verbinding ertussen kan heel goed via een netwerk verlopen. (De namen 'server' en 'client' kunnen hier wat verwarrend zijn. In deze context wordt met 'server' de computer bedoeld waar de X server draait, en met 'cliënt' de computer waar de applicaties draaien die van de X server gebruik maken. Het kan dus best gebeuren dat de 'client' een "zware groeps-*server*" is.)

Start aan de server-kant X normaal op en geef in een xterm het commando **xhost + <client-computer-naam>**, bijvoorbeeld **xhost + elektron.et.tudelft.nl**. Start dan (nog steeds aan de server-kant) een telnet-verbinding naar die andere computer. Je kunt dan daar (aan de client-kant) **xterm &** doen. Maar dat hoeft niet te werken, omdat de client niet altijd weet welke computer de server is. Dat kun je aan de client doorgeven met het commando **DISPLAY=<hostname>:0.0; export DISPLAY** (in **bash**-achtige shells) of **setenv DISPLAY <hostname>:0.0** (in **csh**-achtige shells). Voorbeeld: **DISPLAY=kennedy.dorpsnet.nl:0.0; export DISPLAY** en dan werkt **xterm &** zoals het hoort. Zolang de **xhost** 'open staat', mag *iedereen* van de betreffende computer iets met jouw scherm (of display) doen -- niet echt veilig. Geef **xhost - <client-computer-naam>** om de toestemming in te trekken; daarbij worden bestaande clients echter niet afgebroken. Voor deze authorisatie bestaat een veiliger alternatief met **xauth**, maar dat is vrij ingewikkeld en het voert te ver om dat hier helemaal te bespreken.

Als je aan de server-kant X had gestart met **xinit**, had je daar nog geen window manager. Op de bovenstaande manier kun je ook een window manager op een andere computer draaien dan de X-server.

Een programma wat je op deze manier een andere computer draait heet trouwens een *remote* programma, zoals een 'remote xterm' of een 'remote window manager'. Dat in tegenstelling tot een *local* (lokaal) programma, dat op de server-computer zelf draait.

Als je de DISPLAY goed zet, kun je ook vanaf een lokale tekst-mode virtuele console een X programma starten, maar daarvoor hoef je geen **xhost** te doen. Meer informatie over **export** en dergelijke staat in paragraaf 7.1.

6.3.8 Instellingen voor X in bestanden

Elke gebruiker heeft de mogelijkheid om X naar zijn persoonlijke smaak in te richten, en dat automatisch te laten doen als X start. Het programma **startx** kijkt namelijk of er een uitvoerbaar bestand **.xsession** in je home directory staat. Als dat er niet is, wordt de default window manager (bijvoorbeeld **fvwm95**) gestart; anders wordt **.xsession** uitgevoerd. Een **.xsession** ziet er bijvoorbeeld zo uit:

```
#! /bin/sh
(sleep 5s; xsetroot -solid darkblue) &
xterm -geometry 80x25+20+20 &
xeyes -geometry 75x75-10+10 &
fvwm95
```

Let er wel op, dat dat bestand uitvoerbaar is voor jezelf (**chmod u+x .xsession**, zie ook paragraaf 4.11).

De tweede regel zorgt voor de achtergrond. De haakjes om de opdrachten zorgen dat die worden uitgevoerd in een sub-shell, en die 'tweede' shell wordt in de achtergrond gezet. Ofwel **xterm**, **xeyes** en **fvwm95** kunnen rustig opstarten, en pas na vijf seconden wordt de achtergrond goed gezet. Omdat **fvwm95** bij z'n startup zelf de achtergrond 'goed' zet, moeten wij eerst vijf seconden (of eventueel langer) wachten voor we iets met de achtergrond doen, want wie 't laatst komt...

Je kunt ook een plaatje in de achtergrond zetten. In plaats van het **xsetroot**-commando kun je dan **xloadimage -onroot -fullscreen <plaatje.jpg>** of **xv -quit -root -max <plaatje.jpg>** neerzetten (naast **.jpg** worden vele formaten ondersteund). In beide gevallen wordt het plaatje 'uitgerekt' tot maximale grootte; lees de manpages voor meer details. Hiervoor moet je natuurlijk wel pakketjes 'xloadimage' en/of 'xv' installeren.

X wordt pas afgesloten als de **.xsession** is afgelopen. In dit geval wordt dus gewacht tot de gebruiker **fvwm95** afsluit (daarom staat er ook geen '&' achter). Dit bestand is een *shell script*, zie paragraaf 7.1 voor meer informatie hierover.

Ook de instellingen van de window manager (bijvoorbeeld de kleuren) zijn te veranderen. De window manager **fvwm95** haalt z'n configuratie standaard uit **/etc/X11/fvwm95/system.fvwm95rc**, tenzij er een **.fvwm95rc** in je homedirectory staat. Om iets aan te passen, kopieer je de **system.fvwm95rc** naar **.fvwm95rc** in je eigen homedirectory en pas je jouw eigen **.fvwm95rc** aan met een teksteditor (bijvoorbeeld **ae**). Zoek naar de "DefaultColors" (ergens bovenin dat bestand) en verander het tweede item van "#cccccc" bijvoorbeeld in "red" (de "#" dus weghalen). Herstart **fvwm95** door in het 'Start'-'Exit Fvwm'-menu voor 'Restart' te kiezen. Dat ziet er opeens heel anders uit.

Er zijn vele andere window manager-pakketjes beschikbaar, die je allemaal kunt uitproberen. De 'default' window manager kun je als root instellen met **update-alternatives --config x-window-manager**.

Praktisch elke window manager is op één of andere manier aan ieders individuele wensen aan te passen. Kijk voor (veel) meer informatie in de betreffende manual page. Ook de manpages van **X**, **startx**, **xinit**, **XF86_<server-naam>** en **xterm** bevatten interessante informatie.

De system-wide X-instellingen staan in **/etc/X11** en **/usr/lib/X11**; als je wat wilt veranderen moet je daar eens gaan zoeken. Maar: wees voorzichtig!

6.3.9 Een X login scherm

Als je altijd in X wilt werken, is het misschien verstandig om altijd X te starten, en 'grafisch' in te loggen: **apt-get install xdm**. Als je dat hebt geïnstalleerd, kun je het de eerste keer opstarten met **/etc/init.d/xdm start**; daarna wordt **xdm** bij elke reboot automatisch gestart. Je kunt praktisch alles van **xdm** aanpassen in de directory **/etc/X11/xdm** -- aan jou de taak om er iets moois van te maken. Kijk eerst naar de ***_0** bestanden, want die gaan over de lokale X server (**xdm** kan ook op niet-lokale X servers displayen). **xdm** tijdelijk stoppen gaat met **/etc/init.d/xdm stop**, permanent stoppen door het pakketje te removen.

Als je een mooie login-achtergrond wilt, moet je eens kijken naar het pakketje 'xbanner'; ook daar is wat smaak-afhankelijke configuratie nodig (en dat maakt het juist zo leuk!).

Een alternatieve grafische login kun je krijgen met **apt-get remove xdm ; apt-get install wdm**.

Je kunt nog steeds zelf X starten vanaf een tekstmode virtuele console; gebruik dan **startx -- :1**, X wordt dan gestart op tty8. Je kunt zo ook met een andere 'kleurdiepte' (aantal bits-per-pixel) starten, bijvoorbeeld **startx -- -bpp 8**.

6.3.10 De bel en de screensaver

Als je de standaard-bel ('pieptoon') in X niet mooi vindt (test met de backspace in een xterm), kun je die met bijvoorbeeld **xset b 100 100 100** veranderen. De 'b' is van bell, de drie getallen zijn volume (altijd 100), toonhoogte en tijdsduur. Of probeer **xset b 100 5000 10**.

xset is een commando dat diverse X instellingen kan wijzigen die niet in configuratie-bestanden voorkomen. De tweede belangrijke instelling is de screensaver. Met **xset s 1** gaat het scherm steeds na 1 seconde op zwart. Vaak

wordt 300 gebruikt, voor 300 seconden = 5 minuten; standaard staat hij echter op 0, dus uitgeschakeld, en dat is niet erg fijn voor de monitor.

De **xset s** is niet milieu-vriendelijk, want de monitor blijft gewoon aan. Dat kan ook anders, als je monitor 'DPMS' ondersteunt (praktisch alle monitoren doen dat). Met **xset s 300 dpms 360 420 480** gaat de monitor na 5 minuten op zwart, 6 op 'standby', 7 op 'shutdown' en 8 op 'off'. Met **xset dpms force off** gaat de monitor meteen uit; **xset -dpms** schakelt de DPMS-screensaver uit.

Andere mogelijkheden van **xset** kun je zien met xset zonder opties, of in de manpage. De huidige instellingen worden weergegeven met **xset q**. Het is een goed idee om **xset**-commando's in je **.xsession** te zetten.

Meer processor-intensieve screen-'savers' zijn te vinden in het pakketje 'xscreensaver', in te stellen met 'Screen' - 'Save' - 'ScreenSaver Preferences' in het (Debian) menu. Dit programma kan in de achtergrond draaien en het scherm na een in te stellen tijd voor andere dingen gaan gebruiken.

6.4 De GNOME desktop-omgeving

De laatste tijd zijn een tweetal 'desktop-omgevingen' ontwikkeld, KDE (*K Desktop Environment*) en GNOME (*GNU Network Object Model Environment*). KDE heeft momenteel effectief geen licentie, en mag dus niet verspreid worden; daarom zit KDE niet bij Debian. Bovendien is er uit 'de industrie' meer interesse in GNOME, dus zal ik me hier daartoe beperken.

De aanduiding 'desktop-omgeving' is een beetje vaag. GNOME bestaat uit 1) een aantal library's voor grafische zaken, en 2) een 'panel'. De zogenoemde "GNOME programma's" gebruiken allemaal die library's, en krijgen daardoor hetzelfde uiterlijk. Daar is niets bijzonders aan, want er zijn diverse andere soortgelijke library-sets. Het 'panel' is ook zo'n GNOME programma, maar heeft het specifieke doel om een taakbalk te zijn (met extra's). De 'desktop-omgeving' GNOME is dus niets anders dan een stel losse programma's die er allemaal hetzelfde uitzien. Ze werken gewoon onder X, en met elke willekeurige window manager. Ze hebben het GNOME 'panel' ook totaal niet nodig. Het enige 'probleem' is dat sommige window managers beter met (alleen!) het panel overweg kunnen dan andere.

Ik zal nu de meest 'fancy' GNOME setup beschrijven, compleet met een 'betere' window manager en veel GNOME programma's. Let op: dit is alleen geschikt voor Pentium-processors en hoger, want op een '486 kan het starten van X, window manager en panel zeker een kwartier gaan duren (en bovendien kost het behoorlijk veel geheugen).

6.4.1 Installatie van GNOME

GNOME draait onder X, dus je moet eerst X werkend geïnstalleerd hebben, zie eerder in dit hoofdstuk.

Installatie van de benodigde pakketjes gaat eenvoudig als root met **apt-get install task-gnome-desktop task-gnome-apps task-gnome-games task-gnome-net enlightenment gnome-help gnotepad+-help**; dit is zo'n 70 MB. Bij de configuratie vraagt 'esound' of je een (onder Linux) werkende geluidskaart hebt, geef het juiste antwoord. Na afloop moet je waarschijnlijk nog één of twee keer **dpkg --configure --pending** doen om alles te configureren.

Tenslotte moet je de standaard window manager aanpassen, **update-alternatives --config x-window-manager**, en kies het getal dat voor **/usr/bin/enlightenment** staat.

6.4.2 Opstarten van GNOME, eerste configuratie

Na de installatie van de pakketjes log je weer in als gewone gebruiker. Dan maak je een **.xsession** in je homedirectory met enkel

```
#! /bin/sh
gnome-session
```

erin (de rest kun je eventueel uitcommentaren door er hekjes ('#') voor te zetten). Start dan X (en Gnome) met **startx**, zoals je gewend bent. Eerst zie je Enlightenment starten, de eerste keer is dat vrij langzaam. Dan verschijnen er een aantal windows, en het panel onderin het scherm.

Voor je echt met alles aan de slag gaat, moet er nog het één en ander ingesteld worden zodat je er als beginner prettiger mee kunt werken. Sluit eerst alle openstaande vensters met de 'X'-knop rechtsboven. Rechts-klik dan op de achtergrond om het 'Settings' menu te krijgen. In 'Focus Settings' kies je 'Focus follows mouse clicks', 'Clicking in a window always raises it' en 'All new windows first get the focus', dan 'OK'. Vervolgens weer met rechts-klik naar 'Settings', en in 'Pager Settings' zet je 'Enable pager display' *uit*, dan 'OK'. In 'Settings' - 'Multiple Desktop Settings' sleep je de schuifbalk naar precies 1 desktop. Bij 'Settings' - 'Virtual Desktop Settings' sleep je de schuifbalken naar '2 x 2', en zet je 'Enable edge flip' *uit*. Vervolgens rechts-klik je op het zwarte vakje rechtsonder op het scherm en kies je 'Close Iconbox'; dat moet je misschien twee keer doen. Dan rechts-klik je op een leeg stuk van de taakbalk (= het panel), kies 'Add applet' - 'Utility' - 'Gnome Pager'. Tenslotte nog een keer rechts-klik op een leeg stuk van het panel, kies 'System menus' - 'Settings' - 'Desktop' - 'Background', en in het venster kies je 'Disable background selection' en dan 'OK', sluit het venster met de 'X'-knop rechtsboven.

Om deze instellingen te bewaren, klik je nu direct, dus zonder vensters op het scherm, op de Gnome-logo knop, de gestileerde voetafdruk links op het panel, en kies je 'Log out', de onderste optie. In het venster selecteer je eerst 'Save current setup' en dan 'Yes'.

6.4.3 Werken in GNOME

Je start X plus Gnome zoals gebruikelijk, met **startx** of een **xdm**-achtige grafische login. Als je de muis even op de achtergrond laat staan, verschijnt een wolkje met aanwijzingen. Als je dat niet meer wilt zien, rechts-klik je op de achtergrond naar het 'Settings' - 'Tooltip Settings' menu, en zet daar 'Display Root Window Tips' uit, 'OK'.

Door te dubbel-klikken op het 'Home directory' icoontje linksboven op de achtergrond kun je een 'verkenner-achtige' file manager starten, maar dat kan ook met 'File Manager' in het Gnome-knop menu. Je kunt bestanden en directory's tussen file managers slepen, maar ook naar de achtergrond ('desktop'). Zonder toets is meestal verplaatsen, met Ctrl ingedrukt is kopiëren, en Alt (of MiddenKnop-slepen) vraagt wat er moet gebeuren: verplaatsen, kopiëren of symlinken (snelkoppeling maken). In de 'Settings' - 'Preferences' van de file manager kun je het beste onder 'Desktop' de optie 'Snap icons to grid' aanzetten.

Een vertrouwde xterm start je uit het Gnome-knop menu met 'Xshells' - 'Xterm'. Het hulp-systeem kun je inkomen met 'Help system' in het Gnome-knop menu. En er zijn nog talloze andere programma's, trucs en instellingen, maar die zal ik hier niet uitleggen. Zelf proberen is veel leuker!

Uitloggen gaat als eerder gedaan met 'Log out' in het Gnome-knop menu. Als je 'Save current setup' kiest, zullen alle openstaande file managers, xterms enzovoort de volgende keer dat je X/Gnome start automatisch weer geopend worden. En dat blijft zo tot je weer een keer 'Save current setup' kiest tijdens het uitloggen.

7. Meer over de shell

De shell kan veel meer dan het simpelweg starten van programma's. Het is in feite een complete programmeertaal en een werk-omgeving in één. In dit hoofdstuk uitgebreidere informatie over de shell **bash**, de 'visuele' shell **mc** en utility's die vaak in shells worden gebruikt.

7.1 Shell scripts

In DOS was je wellicht batch-bestanden gewend om allerlei zaken automatisch te laten verlopen. Dat kan in Linux ook, maar dan heet het een *shell script*. Dat betekent, dat het een reeks opdrachten (script) is, dat door een shell wordt uitgevoerd. Shell scripts zijn (net als batch-bestanden) gewone 'plain text' ASCII-files. In een shell script kunnen, naast het domweg opstarten van programma's, ook allerlei opdrachten worden gebruikt (bijvoorbeeld if, while en case). De notatie van die opdrachten is echter voor iedere shell anders. Daarom beperk ik me hier tot **bash**, kijk voor andere shells in hun respectievelijke manpages.

Geef voor je het onderstaande gaat proberen, eerst het commando **export PATH=$PATH:.** zodat 'de huidige directory' aan je zoekpad wordt toegevoegd. Een shell zoekt uitvoerbare bestanden, zoals shell scripts, namelijk *alleen* in het zoekpad en niet in de huidige directory. Het is een aardig idee om dat zoekpad-commando toe te voegen aan je **.bashrc** (zie paragraaf 7.9). Het originele zoekpad wordt gezet in **/etc/profile** (zie ook paragraaf 7.9).

Als je niets aan je zoekpad wilt veranderen, moet je voor elk programma dat niet in het zoekpad staat de juiste directory opgeven, en '**./**' staat voor 'de huidige directory'; bijvoorbeeld **/home/ikke/bin/testscript** of **./testscript** als je in **~/bin** staat.

Een shell script begint altijd (bovenste regel) met een aanduiding van de shell die gebruikt moet gaan worden. Dat gaat met een regel '#! /dir1/dir2/shell', bijvoorbeeld:

```
#! /bin/bash
```

Maar niet elke UNIX variant heeft een **bash**. En omdat **bash** zich toch min of meer gedraagt als **sh** (in Linux is **sh** een symlink naar **bash**) en elke UNIX wél een **sh** heeft, schrijven we meestal

```
#! /bin/sh
```

Onder die eerste regel komen de opdrachten en commando's die we willen laten uitvoeren. Een voorbeeld:

```
#! /bin/sh
# Achter een hekje mag je commentaar neerzetten
# en lege regels mogen ook.

# Zet een tekst op het scherm en voer het commando 'date' uit
echo Het is nu...
date
```

Deze tekst kun je met bijvoorbeeld **ae** in een bestand **zietijd** in je homedirectory zetten. Als je dat probeert uit te voeren door het commando **zietijd** te geven, zal dat niet lukken. Dat komt, omdat je nog niet hebt aangegeven dat het bestand **zietijd** een uitvoerbaar bestand is. Dat doe je met **chmod u+x zietijd** (de '**u**' van user, je mag het alleen zelf uitvoeren; met '**a+x**' mag iedereen het; zie ook paragraaf 4.11). Na het aanzetten van het 'executable'-bit werkt het commando **zietijd** zoals het hoort (tenminste, als je het **$PATH** goed hebt gezet in de vorige paragraaf).

7.2 Shell variabelen

Af en toe wil je een waarde opslaan in een stukje geheugen, om dat dan later (wellicht meerdere keren) te gebruiken. Dat heet dan een *shell variable*. In DOS kende je iets als **set TEMP=C:\TMP** en kon je de inhoud zien met bijvoorbeeld **echo %TEMP%**. In **bash** gaat dat zo:

```
#! /bin/sh

# Vul een variabele met een waarde
TestVar=TestInhoud

# Laat de inhoud van de variabele zien
echo Dit staat in de test-variabele: $TestVar
```

Merk op dat de naam (en ook de inhoud) van een variabele *case sensitive* is (hoofdletters en kleine letters zijn verschillend).

7.3 Quotejes en dergelijke

Probeer eens:

```
#! /bin/sh
echo Hier staan                veel spaties          tussen.
echo Hier staat een sterretje: *
```

Voer dit eens uit vanuit je home directory en bijvoorbeeld vanuit de root directory (dan wel de volledige padnaam naar het script opgeven). Er vallen twee zaken op: de spaties zijn verdwenen en op de plaats van het sterretje worden bestandsnamen ingevuld. Dat komt doordat de shell eerst de commando's 'voorbewerkt'. Het sterretje wordt vervangen door 'passende' bestandsnamen (dat gebeurde ook al met **ls** in paragraaf 4.7!). De *shell* bepaalt wat de argumenten van een commando zijn -- en argumenten "zijn gescheiden door één of meer spaties". De bovenste **echo** krijgt dus vijf argumenten, namelijk 'Hier', 'staan', 'veel', 'spaties' en 'tussen.', en **echo** drukt alle argumenten achter elkaar af met precies één spatie er tussen.

Er is echter een manier om aan te geven dat de shell geen 'voorbewerking' mag doen, namelijk door quotejes te gebruiken. Er zijn enkele quotejes en dubbele quotejes (aanhalingstekens), die een iets andere functie hebben. Enkele quotejes stoppen alles wat er tussen zit onveranderd in één argument; dubbele quotejes maken ook één argument, maar staan sommige 'voorbewerkingen' nog wel toe. Probeer maar eens het volgende en bestudeer de uitvoer:

```
#! /bin/sh
echo Hier staan                veel spaties          tussen.
echo Hier staat een sterretje: *
echo 'Hier staan                veel spaties          tussen.'
echo 'Hier staat een sterretje: *'
echo "Hier staan                veel spaties          tussen."
echo "Hier staat een sterretje: *"

TestVar=TestInhoud
echo Dit is $TestVar
echo 'Dit is $TestVar'
echo "Dit is $TestVar"
echo Dit is \$TestVar en een sterretje: \*
```

Je kunt ook één enkel karakter 'quoten' door er een backslash voor te zetten, zoals in de laatste regel is aangegeven. Om een backslash zelf af te drukken, zet je er twee achter elkaar ('\\'), of je zet er één binnen *enkele* quotejes.

7.4 De uitvoer van een programma 'vangen'

Stel je voor dat je de uitvoer van het commando date in een variabele wilt hebben. Daarvoor moet je backquotes gebruiken (de '`' dus, meestal links van de '1'-toets):

```
#! /bin/sh
# Uitvoer van date naar een variabele
datum=`date`
# Zet het op het scherm
echo In \$datum staat: $datum
# Het kan ook zo:
echo De datum is `date`
```


7.5 Meer opdrachten voor bash-scripts

Bestudeer het volgende scriptje om wat gevoel te krijgen voor de opdrachten die je in een **sh**-shell script kunt gebruiken:

```
#! /bin/sh

# Kijk of een bestand bestaat
if [ -e /etc/motd ] ; then
    echo Het bestand /etc/motd bestaat
fi
# Algemene syntax:
# if <commando> ; then <commando's> else <commando's> fi
# Commando `[' is een algemeen test-commando,
# -e : test of bestand bestaat, -r : leesbaar, -x : uitvoerbaar
# Zie verder `man ['

# We genereren een lijst en lopen die langs
for i in `ls /` ; do
    echo De variabele i is nu $i
done
# Algemene syntax: for <var> in <lijst> ; do <commando's> done
# Probeer ook eens met:  for i in /* ; do  en let op het verschil

# Wat rekenwerk met de opdracht `let'
a=1
b=2
c=a+b              # hier wordt nog niet gerekend
let d=a+b          # hier wel
let e=$a+$b        # ook goed, $'s mogen bij `let' eventueel ook
let f=(5*a+b)/3    # 7/3 = 2.3333 dus afgerond 2
echo a=$a b=$b c=$c d=$d e=$e f=$f

# Knip een stuk uit een tekst
string="Dit is heel leuk"
stuk=`expr substr "$string" 8 4`
echo Het stuk is \"$stuk\"
# `expr' kan nog veel meer, zie `man expr'
# Let ook op gebruik van quotejes e.d. (laat ze bijv. eens weg)
```

7.6 Environment variabelen

Doe eens het volgende in de shell zelf:
TestVar1=TestInhoud1
TestVar2=TestInhoud2
export TestVar1

En maak dan het onderstaande scriptje en voer het uit:

```
#! /bin/sh
echo Dit is TestVar1: $TestVar1
echo Dit is TestVar2: $TestVar2
```

Je ziet, dat TestVar2 op de één of andere manier 'verdwenen' is. Dat komt omdat shell-variabelen standaard niet worden doorgegeven aan programma's die vanuit de shell worden gestart (en een shell script is ook een programma, het wordt uitgevoerd door een tweede shell). Na een **export**-opdracht wordt de variabele wél doorgegeven (geëxporteerd). Je kunt in **bash** ook een **export** combineren met een waardetoekenning, zoals in **export TestVar1=TestInhoud1**, maar dat werkt meestal *niet* in de **sh**'s van andere UNIXen. Een geëxporteerde variabele wordt ook wel *environment variabele* genoemd.

In andere shells werken shell variabelen en environment variabelen anders. In (**t**)**csh** zet je een shell variabele met **set var=inhoud** en een environment variabele met **setenv var inhoud** (zonder '**=**'). Zie de manpage van de betreffende shell voor meer details.

In **bash** kun je een overzicht van alle bekende variabelen krijgen met het commando **set** (zonder argumenten). Een variabele kun je verwijderen met **unset <variabele>**.

7.7 Aliasjes

Er zijn een aantal ingewikkelde commando's die je zeer vaak zult gebruiken, bijvoorbeeld **ls -alF**. Vele Linuxers willen dat afkorten tot een simpel commando **ll** (van long list). Nu zou je natuurlijk een scriptje **ll** kunnen maken met daarin

```
#! /bin/sh
ls -alF
```

maar dat is een beetje overdreven, omdat het makkelijker kan. Dan hebben we het over een *alias*. Een alias is simpelweg een afkorting van één of enkele simpele commando's.

Om een commando **ll** te maken, type je **alias ll='ls -alF'**, en probeer of **ll** werkt. Je kunt ook **ll /** geven -- de **ll** wordt vervangen door **ls -alF** en de **/** blijft er gewoon achter staan.

Je zou deze lijst met 'handige aliasjes' aan je **.bashrc** (zie ook paragraaf 7.9) kunnen toevoegen:

```
alias cd..='cd ..'
alias copy='cp -i'
alias del='rm -i'
alias ll='ls -alF'
alias md=mkdir
alias move='mv -i'
alias quit=exit
alias rd=rmdir
alias ren='mv -i'
alias type=cat
alias x=startx
```

Je kunt commando's scheiden door een puntkomma (';'), zodat je ons allereerste voorbeeld-script ook met een aliasje kunt realiseren: **alias zietijd_alias='echo Het is nu... ; date'**, en probeer het commando **zietijd_alias**.

Het commando **alias** zonder argumenten geeft een overzicht van alle gedefinieerde aliasjes; met **unalias <aliasnaam>** kun je een aliasje opheffen.

7.8 De prompt in bash

Veel gebruikers vinden de standaard-prompt niet optimaal. Dat is echter geen probleem, want net als in DOS kan de prompt veranderd worden en nog heel uitgebreid ook. Een veelgebruikte prompt kun je bereiken met het commando **PS1='[\u@\h $PWD]\$ '** gevolgd door **export PS1**. Met deze commando's zet je een environment variabele, zie paragraaf 7.6 voor meer info daarover.

Het is aan te raden om die twee opdrachten (of een variant ervan) op te nemen in de **.bashrc** in je homedirectory, zie ook paragraaf 7.9.

Die zelfde prompt werkt trouwens ook voor root -- en daar staat toch een # in plaats van een $... Zie **man bash** voor de uitleg (zoek naar 'PROMPTING').

7.9 De opstartbestanden van bash

Er is zoiets als een **AUTOEXEC.BAT** in **bash**, ofwel een mogelijkheid om bij het starten van de shell een reeks commando's automatisch uit te voeren. Daarin maakt **bash** onderscheid tussen een login shell en een niet-login shell. Een shell is een login shell als het het eerste programma is dat gestart wordt als je net ingelogd bent. De shell waarin je tot dusver werkte, was dus een login shell. Maar je kunt ook een nieuwe shell starten met het commando **bash**. Die tweede (sub-)shell is geen login shell (bekijk 't ook eens met **ps**, een '-bash' is een login shell).

Voor login shells leest **bash** eerst het 'system-wide' opstartscript **/etc/profile** en daarna het script **.bash_profile** uit jouw eigen homedirectory. Beide scripts zijn gewone shell scripts, maar hoeven geen '#!-regel' te hebben, omdat ze toch al door **bash** worden uitgevoerd.

Voor niet-login shells leest bash alleen het script **.bashrc** in je eigen homedirectory (de 'rc' betekent iets als runtime configuration).

In de files **.bash_profile** en **.bashrc** kun je dus commando's zetten die allerlei dingen speciaal voor jou goedzetten, zoals de prompt (zie paragraaf 7.8). Maar over het algemeen wil je zowel in login shells als niet-login shells dezelfde instellingen. Het is daarom goed gebruik om in de **.bash_profile** (vrijwel) alleen

```
if [ -f ~/.bashrc ] ; then
    source ~/.bashrc
fi
```

te hebben staan. Hiermee wordt eerst getest op het bestaan van een **.bashrc** in je home directory, en die wordt dan uitgevoerd (met het '**source**' commando). Instellingen voor de shell hoef je dan alleen in je **.bashrc** te zetten, want ze komen effectief ook in je **.bash_profile** terecht.

Als je iets wilt laten gebeuren als je uitlogt, kan dat in een **.bash_logout** in je home directory.

Tijdens het aanmaken van een nieuwe gebruiker met **adduser** worden voorbeeld-bestanden met standaard-instellingen uit **/etc/skel/** gekopieerd naar de nieuw gemaakte home directory. Dus als je elke volgende nieuwe gebruiker 'betere' standaard-instellingen wilt geven, moet je ze daar veranderen.

Je zult merken dat in de standaard **.bashrc** commando's staan die **ls** een 'gekleurde' directory listing laten geven. Persoonlijk vind ik dat niet prettig werken, dus haal ik de '**eval `dircolors`**' en '**alias ls=**...' weg uit mijn eigen **.bashrc** en **/etc/skel/.bashrc**.

Als je meer wilt weten over alles wat met de shell te maken heeft, kijk dan in **man bash** en/of **info bash**; de opstart-bestanden staan bijvoorbeeld beschreven onder 'INVOCATION'.

7.10 Midnight Commander, een 'visuele' shell

Naast shells als **bash**, **ash**, (pd)**ksh**, **csh**, **tcsh**, en **zsh**, die allemaal regel-georiënteerd zijn, zijn er twee shells die 'visueel' werken: **git** (*GNU Interactive Tools*) en **mc** (*Midnight Commander*). Beide zijn gebaseerd op de bijzonder populaire *Norton Commander* voor DOS. Omdat **git** veel minder geavanceerd is dan **mc** (maar daarom wel veel makkelijker compileert op andere UNIX'en), zal ik er hier geen aandacht aan besteden.

De Midnight Commander installeren gaat als altijd eenvoudig met **apt-get install mc file** (als root).

Als je (als gewone gebruiker, maar ook als root) **mc** (**mc -c** in een xterm) voor het eerst start, staat nog niet alles optimaal voor 'de beginner'. Het is daarom verstandig de volgende handelingen even uit te voeren: druk op F9, kies met de pijltjestoetsen 'Options' - 'Configuration', zet daar zowel 'show Backup files' als 'show Hidden files' aan met de spatiebalk, en 'Auto save setup' juist uit, dan met de pijltjestoetsen naar '[Save]' en Enter. Weer F9, nu 'Options' - 'Layout', zet alle opties in het 'Highlight...'-vakje uit, en weer '[Save]'. Type dan **cd** en Enter, ga met de TAB-toets naar het linker-venster, en type weer **cd** en Enter. Dan staat je home-directory dus aan beide kanten. Kies tenslotte F9, 'Options' - 'Save setup', Enter, Enter. Met F10 verlaat je **mc** weer.

Als je nu **mc** (**-c**) opnieuw start, zal de huidige directory in het linker-venster staan, en je home-directory rechts. Je kunt aan beide kanten (TAB) met de pijltjestoetsen door de bestanden heen bladeren. Enter op een directory of uitvoerbaar bestand heeft het verwachte effect, maar werkt ook op bijvoorbeeld een **.tar**(**.gz**) archief en een **.deb** pakketje. En zo kun je ook met F9 - 'Left'/'Right' - 'FTP link...' over een FTP-site lopen (vul bijvoorbeeld eens 'ftp.debian.org' in). En ook dan blijven praktisch alle features werken.

Op de onderste regel kun je zien welke functies de functietoetsen hebben, probeer ze allemaal eens uit (voorzichtig!). Als je toetsenbord geen functietoetsen heeft, kun je ook bijvoorbeeld Escape, 1 gebruiken. Een opdracht cancelen gaat met dubbel-Escape. De verschillende functies werken ofwel op het éne bestand waar de selectie-balk op staat, of op meerdere bestanden die met de Insert-toets zijn geselecteerd. Ook met '+', '-' en '*' kan de selectie veranderd worden. Copy/Move kopieert/verplaatst standaard van het éne naar het andere venster, maar in het dialoogvenster kun je dat wijzigen.

Je kunt, net als in een gewone shell, ook commando's intypen. Die verschijnen op de regel onderin, probeer maar eens `ls -alF`. Met `cd` kun je als gebruikelijk naar andere directory's springen, maar als het pad ingewikkeld of lang is werken een paar Enters in het 'visuele' gedeelte stukken sneller. Je kunt de momenteel-geselecteerde bestandsnaam op de commando-regel plakken met Escape, Enter. Ga bijvoorbeeld op je `.bashrc` staan en en type `cat`, Spatie, Escape, Enter, en nog een Enter om het uit te voeren. Je kunt door je vorige commando's bladeren met Escape, p en Escape, n, maar als je een werkende muis hebt kun je ook op het '[^]'-teken aan het einde van de commando-regel klikken.

De Midnight Commander heeft nog talloze andere opties die bestandsbeheer extreem efficiënt maken; je kunt er alles over lezen in de uitgebreide 'online' hulpteksten die met F1 altijd bereikbaar zijn.

7.11 Shell-utility's en tekst-utility's

De UNIX filosofie is gebaseerd op kleine programma's die allemaal een specifieke opdracht kunnen uitvoeren. Door ze handig in scripts te combineren kan elk gewenst resultaat worden bereikt. De meeste commando's betreffen bestandsbeheer en de verwerking van tekst-bestanden. Ik zal de meest-gebruikte noemen, soms met een klein voorbeeld. Ze hebben allemaal een uitgebreide manpage met meer informatie.

`awk` (`mawk`, `nawk`) - tekst-processing taaltje, `echo a b c | awk '{print $2}'`
`basename`, `dirname` - tonen van alleen bestandsnaam of directorynaam, `basename /a/b/c`
`bc` - rekenmachine-taaltje. Probeer eens '5', '1+2', 'a', 'a=8', 'a', 'a=a+4', 'a', 'quit'
`cal` - tonen van een kalender van een opgegeven maand/jaar
`cksum`, `md5sum`, `sum` - berekenen van een checksum van bestanden, ter controle van een goede overdracht
`clear` - schoonmaken van het scherm
`cut` - knippen van bepaalde velden uit regels, `cut -d ':' -f 5 /etc/passwd`
`diff`, `sdiff`, `cmp`, `comm` - zoeken van verschillen in (tekst-)bestanden
`dmesg` - tonen van de (opstart)berichten van de kernel
`ed`, `sed`, `red` - command-line text editors, `sed -e 's/o/hallo/g' /etc/passwd`
`factor` - bepalen van priemfactoren van getallen, `factor 12345`
`free` - tonen van hoeveelheid gebruikt geheugen/swap, vergelijk `cat /proc/meminfo`
`grep` - tonen van regels waar een tekenreeks (niet) in voorkomt, `grep home /etc/passwd`
`head`, `tail` - bekijk de eerste/laatste regels uit een bestand (evt. in volg-modus), `tail -5 /etc/passwd`
`hexdump`, `od` - dumpen van bestanden in decimaal, hexadecimaal, octaal etc, `od -tx1 -Ax /etc/passwd`
`hostname`, `dnsdomainname` - tonen van machinenaam en domeinnaam
`killall`, `skill` - killen van een aantal processen tegelijk door hun naam/gebruiker te specificeren
`last`, `lastlog` - bekijken van alle/laatste inlogtijden van alle gebruikers
`mktemp`, `tempfile` - produceren van een unieke (nog niet gebruikte) naam voor een tijdelijk bestand
`nice`, `renice`, `snice` - uitvoeren van commando's / programma's met een lagere prioriteit (= een hogere nice-waarde) en veranderen van de prioriteit van draaiende programma's
`nl` - toevoegen van regelnummers aan tekstbestanden, `nl /etc/passwd`
`seq` - genereren van een reeks getallen, `seq 1 10`
`setleds` - aan en uit zetten van de LED's op het toetsenbord, alleen in tekstmode, `setleds +caps`
`sleep` - wachten van een een te specificeren aantal seconden, `sleep 5`
`sort` - sorteren van tekstbestanden (kan veel schijfruimte gebruiken), `sort /etc/passwd`
`split`, `csplit` - opdelen van bestanden in kleinere bestanden, `split -b 100 /etc/passwd /tmp/passwd.`
`tac` - druk bestanden ondersteboven af, `tac /etc/passwd`
`tee` - kopiëren van standaard input naar standaard output en naar een bestand, `<opdracht> | tee <bestand>`
`touch` - aanmaken van een leeg bestand, de datum/tijd van een bestand actueel maken/wijzigen
`tr` - wijzigen of verwijderen van tekens in bestanden, `tr tro gva < /etc/passwd`
`tty` - tonen van de device naam van de huidige terminal
`uname` - tonen van informatie over OS-versie, `uname -a`
`uniq` - tonen van (niet) unieke regels in een tekstbestand
`uptime` - tonen van de tijd dat Linux 'up' is, het aantal gebruikers en de gemiddelde system load

`wc` - tellen van regels, woorden en bytes in een tekstbestand, `wc /etc/passwd`
`which` - zoeken van de volledige naam van een commando/programma, `which bash`
`yes` - afdrukken van een oneindige rij 'y<Enter>', voor redirection naar commando's die bevestigingen willen, `yes | tr y n` voor het andere antwoord, of `yes | sed -e 's/y/yes/g'`; stop met Ctrl-C

7.12 Reguliere expressies

Commando's als `grep` en `sed`, maar ook het zoeken met '/' in `less` en de `mc`-viewer, maken gebruik van *reguliere expressies* om te zoeken in tekstbestanden. Deze hebben hetzelfde doel als het vraagteken en sterretje in de shell, maar zijn veel krachtiger. Dezelfde tekens worden voor andere doelen gebruikt, en dat kan verwarrend zijn. Ik zal een aantal eenvoudige voorbeelden geven, meer diepgaande informatie staat in `man grep`. Reguliere expressies moeten (praktisch) altijd binnen enkele quotejes opgegeven worden, anders zou de shell ze gaan interpreteren. Regexps 'matchen' altijd op hele regels; `grep` bijvoorbeeld drukt een regel af als een gedeelte daarvan gematcht wordt. Als test-data gebruiken we een directory-listing, met `ls -1d`, die alle bestanden op een aparte regel weergeeft en de inhoud van subdirectory's overslaat. Voor het beste resultaat moet je vooraf in een directory gaan staan met veel bestanden, zoals `/etc` of `/usr/bin`.

`'a'` - een karakter of cijfer matcht zichzelf, `ls -1d * | grep 'a'` is equivalent met `ls -1d *a*`
`'.'` - een punt matcht één willekeurig teken, `ls -1d * | grep 'a.s'` = `ls -1d *a?s*`
`'\.'` - om een punt letterlijk te matchen moet je 'm escapen, `ls -1d * | grep '\.'` = `ls -1d *.*`
`'a*'` - een sterretje matcht het voorgaande item 0 of meer keer, `ls -1d * | grep 'pa*s'`
`'.*'` - matcht dus 0 of meer willekeurige tekens, `ls -1d * | grep 'p.*s'` = `ls -1d *p*s*`
`'a\+'` - een plus-teken matcht het voorgaande item 1 of meer keer, `ls -1d * | grep 'as\+w'` = `ls -1d * | grep 'ass*w'`
`'[abc]'` - blokhaak-reeks matcht één van die tekens, `ls -1d * | grep '[abc]'` = `ls -1d *[abc]*` = `ls -1d *a* *b* *c* | uniq`
`'^'` - dakje ('caret') matcht het begin van de regel, `ls -1d * | grep '^a'` = `ls -1d a*`
`'$'` - dollar-teken matcht eind van de regel, `ls -1d * | grep 't$'` = `ls -1d *t`
`'^a$'` - matcht dus precies de hele regel, `ls -1d * | grep '^a$'` = `ls -1d a` en `ls -1d * | grep '^.*a.*$'` = `ls -1d *a*`
`'\(iets\)'` - haakjes maken er één item van, `ls -1d * | grep 'a\(ss\)*d'` = `ls -1d *ad* *assd* *assssd* *assssssd*` ... `| uniq`
`'\|'` - 'or' teken matcht linkse of rechtse item, `ls -1d * | grep 'a\(ss\)\|\(do\)w'` = `ls -1d *assw* *adow* | uniq`

Merk op dat sommige tekens hier ge-escaped worden met '\'; het verschilt van programma tot programma of en wanneer dat nodig is.

8. Opslagmedia

Je weet inmiddels hoe je harddisks moet mounten (paragraaf 5.8). Nu meer over andere aspecten, en over diverse manieren om diskettes te gebruiken. Procedures voor CD-R's en CD-RW's besluiten dit hoofdstuk.

8.1 Werken met diskettes

Mounten
Diskettes zou je kunnen mounten met iets als `mount /dev/fd0 /floppy` (eerste floppydrive) of `mount -t vfat /dev/fd1 /floppy` (tweede floppydrive, lange bestandsnamen), zie ook paragraaf 5.8. Dat heeft echter wel een paar nadelen. Je moet in ieder geval root worden om te mouten (tenzij je 'user' hebt gespecificeerd in `/etc/fstab`). En je mag een gemounte flop *nooit* zomaar uit de drive halen -- *altijd* eerst unmounten met `umount /floppy`.

Mtools
Er is echter een veel eenvoudigere manier om floppy's aan te spreken, namelijk met mtools (van msdos tools). Na `apt-get install mtools fdutils` zal het commando `mdir a:` de directory van de diskette in de eerste floppy drive laten zien (ja, je moet hier de vertrouwde DOS-driveletters gebruiken).

Merk op dat je niet van/naar een diskette kunt lezen/schrijven als je daar geen toestemming voor hebt. Je kunt iedereen schrijftoestemming geven door als root het commando `chmod a+rw /dev/fd0` (of `fd1`) te geven. Je kunt ook bepaalde gebruikers 'selecteren', door ze lid te maken van de groep 'floppy' (als root `adduser <username> floppy`; zie ook paragraaf 13.2), dan hoef je geen `chmod` te gebruiken.

Een bestand van flop naar `/tmp` kopiëren gaat bijvoorbeeld met `mcopy a:iets.txt /tmp`, en omgekeerd werkt ook. Iets ingewikkelder wordt het als je alles van diskette naar de harde schijf wilt krijgen, want `mcopy a:* /tmp` werkt dan soms niet. Dat komt omdat de shell (`bash` bijvoorbeeld) het sterretje kan gaan 'invullen' met alle namen in de huidige directory die met 'a:' beginnen, en die zijn er natuurlijk niet. Zoiets kun je voorkomen door aanhalingstekens te gebruiken, zoals `mcopy 'a:*' /tmp`. De andere kant op mogen er weer geen aanhalingstekens gebruikt te worden, dus bijvoorbeeld `mcopy /tmp/* a:`. Directory's op een floppy kun je het beste scheiden met een Linux-achtige slash ('`/`') en niet met een DOS-achtige backslash ('`\`'), dus bijvoorbeeld `mcopy a:/dir1/dir2/bestand.ext /tmp`.

Naast `mdir` en `mcopy` bestaan de volgende commando's, die wel heel bekend voorkomen: `mattrib`, `mdel`, `mdeltree`, `mlabel`, `mmd`, `mrd`, `mmove`, `mren`, `mtype` en `xcopy` (zonder 'm'). Lees meer over deze commando's in hun respectievelijke manpages en/of de manpage of `info`-documentatie van mtools.

Formatteren
Je kunt in Linux ook diskettes formatteren in DOS-formaat. Dat gaat met een commando als `superformat /dev/fd0`. Daarmee wordt de diskette geheel geformatteerd en wordt er een DOS filesystem op gezet. Een snelle 'quick format', waarbij alleen een nieuwe directory en FAT wordt geschreven, gaat met iets als `mformat a:`. NB: wees heel erg voorzichtig met deze commando's -- ze vragen niet om een bevestiging! Zie `man superformat` of `info fdutils` voor meer info.

Als je echt iets bijzonders wilt doen, kun je ook een ext2 filesystem op een (geformatteerde) floppy krijgen, met `mke2fs /dev/fd0`. De mtools werken dan natuurlijk niet meer, en je zult de floppy moeten mounten. Voordelen zijn bijvoorbeeld dat er dan mode bits zijn en dat je symlinks kunt maken. NB: altijd unmounten voor je de floppy uit de drive haalt!

Fdflush
Sommige oudere computers hebben een 'goedkope' floppy drive, die het niet doorgeeft als de diskette gewisseld is. En als Linux geen signaaltje krijgt, zal bij de volgende lees-opdracht nog gewoon uit de cache worden gelezen, en niet van de (inmiddels nieuwe) diskette. Als je dit probleem tegenkomt, moet je steeds `fdflush` uitvoeren als je een diskette gewisseld hebt.

Direct lezen en schrijven
De meeste apparaten kun je ook als 'rauw' device aanspreken, zo ook floppy's door direct van/naar `/dev/fd0` te lezen/schrijven. Bijvoorbeeld `less -f /dev/fd0`, of een floppy image op een diskette zetten met `cat rescue.bin > /dev/fd0`. Een floppy compleet lezen en er een image-file van maken zou kunnen met `cat < /dev/fd0 > /tmp/flopimage`, maar dat geeft soms problemen omdat de grootte van de diskette niet altijd bekend is. Daarvoor is het commando `dd` geschikt, dat precies-gespecificeerde groottes kan kopiëren. Voor een 1.44MB diskette: `dd if=/dev/fd0 of=/tmp/flopimage bs=512 count=2880`, met de juiste input file, output file, block size en aantal-blokken (`bs=2304 count=640` geeft hetzelfde resultaat, maar sectors op floppy's zijn nu eenmaal 512 bytes).

8.2 Harddisks

8.2.1 Een nieuwe harddisk gaan gebruiken

Partitioneren
Voordat je een harddisk kunt gebruiken, moet je er eerst partities op aanmaken. Dat kan met bijvoorbeeld `cfdisk` of de 'primitievere' `fdisk` in Linux. Bij lege harddisks kan `cfdisk` echter geen partitietabel vinden, en stopt er dan gelijk mee; gebruik in dat geval de `-z` optie.

Let op: als je met de Linux `fdisk`/`cfdisk` een DOS partitie aanmaakt of vergroot/verkleint, moet je direct daarna `dd if=/dev/zero of=/dev/hdXX bs=512 count=1` doen, anders is er een kans dat DOS `format` de schijf te klein of te groot(!!) formatteert.

Filesystem maken
Op een nieuwe partitie kun je een Linux ext2 filesystem maken met `mke2fs /dev/hdXX`. Let op: eventuele oude gegevens worden zonder bevestiging overschreven! Als je de disk gelijk wilt checken, voeg dan de `-c` optie toe.

Je kunt ook vanuit Linux een DOS-partitie maken met `mkdosfs` (`apt-get install dosfstools`). Gebruik `-v` voor meer informatie tijdens het werk, `-c` om eerst de disk te checken en bijvoorbeeld `-F32` voor een FAT32 filesystem. Merk op dat zo'n filesystem zowel onder DOS als Windows (lange bestandsnamen) gebruikt kan worden, en in Linux als type 'msdos' of 'vfat' gemount kan worden.

Als je op een ext2-partitie de `/lost+found` directory perongeluk had weggegooid, niet zelf opnieuw maken, maar gebruik `mklost+found`, anders werkt `fsck` niet meer goed.

Een nieuwe swappartitie initialiseren gaat met `mkswap`; als je een toepasselijke regel aan `/etc/fstab` hebt toegevoegd zal `swapon -a` zorgen dat 'ie gebruikt gaat worden (anders na de volgende reboot).

Inhoud van een harddisk kopiëren
Als je een compleet filesystem, of bijvoorbeeld alleen `/home` of `/usr`, wilt kopiëren naar een andere harddisk/partitie, moet je niet zomaar `cp` gebruiken, want dan verlies je informatie. Het juiste commando is bijvoorbeeld (als root) `cp -av /home /mnt`, dat maakt een nieuwe `/mnt/home` met alle informatie van `/home`, en laat ondertussen zien hoe ver één en ander gevorderd is. Merk op dat het in het algemeen het verstandigste is om bij het kopiëren van een compleet systeem alle directory's uit de root directory apart op te geven, bijvoorbeeld `cp -av /bin /dev /root ... /vmlinuz /mnt`, maar dan moet je `/mnt`, `/proc` en `/lost+found` natuurlijk niet meenemen (`/mnt` en `/proc` zelf op de nieuwe partitie aanmaken). Vergeet na het kopiëren niet de `/etc/fstab` aan te passen.

Je kunt ook kopiëren met twee `tar`'s in een pijp, bijvoorbeeld `tar cvf - /home | (cd /mnt ; tar xvpf -)`. Let op de '`p`' bij de tweede `tar`. Je kunt een '`z`' toevoegen voor compressie, en als je de pijp splitst en een filenaam invult voor de '`-`' kan het ook via een bestand gaan. Interessant is dat je dit ook over een netwerk kunt doen (`apt-get install netcat`); op de 'ontvangende' computer eerst in de juiste directory gaan staan(!) en `nc -l -p 12345 | tar xzvpf -` en dan op de 'zendende' computer `tar czvf - /home | nc kennedy.dorpsnet.nl 12345` (de 12345 is een willekeurig poort-nummer).

Natuurlijk kun je ook harddisks met `dd` kopiëren, maar omdat het zelden exact dezelfde harddisks zullen zijn is dat minder gebruikelijk.

8.2.2 Bestands-informatie

Disk-gebruik

Met `ls` kun je zien hoe groot elk bestand is; met `du *` kun je zien hoeveel diskruimte elk bestand daadwerkelijk gebruikt, in kB's. Enkel `du` geeft informatie over alle sub(-sub...)-directory's, het getal ervoor is hoeveel diskruimte *totaal* door die directory + eventuele subdirectory's wordt gebruikt. `du -s` laat de subdirectory's weg en geeft alleen de totaal-telling voor z'n argumenten, bijvoorbeeld `du -s /home`.

Een overzicht van beschikbare en totaal-gebruikte ruimte op alle gemounte filesystems krijg je met `df`. Als je `/dev/hdXX` of een bestaande bestandsnaam aan `df` meegeeft, krijg je alleen informatie over dat bepaalde filesystem, of het filesystem waar het opgegeven bestand op staat. De informatie is hier weer in kB's (1024 bytes), maar let erop dat sommige UNIX systemen bij `df` (en ook `du`) werken met 512-byte blokken.

Bestanden zoeken

Als je een bestand kwijt bent, kun je het zoeken met bijvoorbeeld `ls -alFR / | less`, maar `find / | less` is meestal sneller. Als je een stuk van de naam weet, kun je ook `find / -name '*naam*'` gebruiken; voor '`/`' kun je ook de start-directory invullen als je maar een gedeelte van het filesystem wilt afzoeken.

Een veel snellere manier is bijvoorbeeld `locate name`. Dat maakt gebruik van de 'locate database' die elke ochtend om half zeven wordt ge-update, *als* je computer dan tenminste aan staat. Je kunt 'm echter ook altijd met de hand updaten, door (als root) `/etc/cron.daily/find` uit te voeren. (Meer over automatische zaken: zie `/etc/crontab` en `man cron`.)

8.2.3 Een kapotte harddisk

Heel af en toe komt het voor, dat een harde schijf problemen geeft. Dat is heel gemakkelijk te herkennen, op de console komen dan talloze foutmeldingen met "... { DriveReady SeekComplete DataRequest Error } ..." en de harde schijf maakt (meestal) vreemde geluiden. In de meeste gevallen zal de betreffende partitie dan automatisch read-only gemount worden, zodat de schade niet nog groter kan worden. Die problemen kunnen trouwens zowel door hardwarefouten (slecht schijfoppervlak) of softwarefouten (computer uitzetten zonder nette shutdown) komen.

Dit soort problemen kunnen vaak worden opgelost met het programma `fsck` (van filesystem check). Maar `fsck` mag *alleen op een niet-gemount filesystem*. Dus als je root-partitie (met het root-filesystem) problemen heeft, moet je van de rescue en root floppy opstarten (of van de Binary-1 CD), alsof je opnieuw wilt gaan installeren, en dan met LinkseAlt-F2 naar de tweede virtuele console gaan. Een niet-root-partitie (met een 'sub-filesystem') moet je eerst unmounten, maar dan kun je in je 'normale' omgeving blijven.

Geef (*na unmount of rescue-boot*) het commando `fsck <devicenaam>`, bijvoorbeeld `fsck /dev/hda6`. In een aantal (langdurige) 'passes' wordt het hele filesystem gecheckt. Het is aan te bevelen voor elke gestelde vraag het 'default'-antwoord te kiezen (steeds op Enter drukken dus). Met deze test worden problemen op het filesystem-niveau verholpen; eventuele 'verdwaalde' bestanden worden onder `/lost+found` geplakt. `fsck` is trouwens een algemeen programma, dat uitzoekt om welk filesystem het gaat en dan een daarvoor gespecialiseerd programma als `e2fsck` of `fsck.minix` aanroept. Het verdient aanbeveling om de bestandsnamen die 'langskomen' ergens op te schrijven en later even na te kijken of daarmee alles nog in orde is (zo niet, dan het betreffende pakketje removen en opnieuw installeren).

Problemen op het fysieke niveau kunnen gedetecteerd worden met het commando `badblocks`, maar daarvoor moet je wel het aantal blocks op de partitie (of de hele harde schijf) weten. Als het om een ext2 filesystem gaat, kun je beter `fsck -c <devicenaam>` gebruiken, dan wordt de bad-blocks-tabel gelijk ge-update (met `badblocks` moet je dat ook zelf uitzoeken).

Meer informatie in de manpages van `fsck`, `e2fsck`, `fsck.minix` en `badblocks`.

8.3 CD-R en CD-RW

Linux kan uitstekend overweg met beschrijfbare CD's, zowel lezen (mounten, of `dd`) als schrijven. Praktisch alle CD-recorders worden ondersteund, maar voor IDE/ATAPI modellen moet je een iets aangepaste kernel compileren.

ISO9660 filesystem maken

Op een data-CD staat meestal een *ISO9660* filesystem. Dat heeft standaard alleen korte (8.3) bestandsnamen, maar met de *Joliet extensies* kunnen er langere namen op; dit is wat Windows 95/98 gebruikt. Met de *RockRidge extensies* krijgt iso9660 een "UNIX gevoel" en worden naast lange bestandsnamen ook eigenaar, groep en access bits opgeslagen, en worden bijvoorbeeld symlinks en devices ondersteund. Deze drie formaten kunnen naast elkaar voorkomen, elk OS kan dan pakken waar hij het beste mee overweg kan.

Een 'ISO9660 image' maken van een bestaande directory-structuur is eenvoudig met `mkisofs` (`apt-get install mkisofs`). Als `/tmp/naarcd` bijvoorbeeld de top-directory van de CD moet worden, geef je een commando als `mkisofs -o /tmp/cdimage.iso -J -r -a -D -v -V 'CD-Naam' -p 'Maker-Naam' -P 'Uitgever-Naam' /tmp/naarcd`. Hiermee worden zowel Joliet als RockRidge extensies toegevoegd, alle files meegenomen, 'te' diepe files niet teruggezet, en de gegeven namen gebruikt. Na afloop staat een mooi CD-image in `/tmp/cdimage.iso`, klaar om naar CD geschreven te worden. `mkisofs` heeft talloze andere opties, waaronder het bootable maken van de CD (lastig), zie de manpage voor details.

Voor je gaat schrijven kun je een image eerst bekijken door te 'loop-mounten' met `mount -o loop /tmp/cdimage.iso /mnt`. Met `-o loop,norock` of `nojoliet` (of beide) kun je een Windows of DOS-emulatie krijgen. Vergeet niet te un-mounten voor je gaat schrijven.

Merk tenslotte op dat je natuurlijk met `dd` een image direct van een CDROM kunt lezen als je 'm letterlijk wilt kopiëren. Daarvoor moet je de grootte van de CD weten, gebruik `isosize /dev/cdrom` (`apt-get install cdwrite`); dit is in bytes en altijd een veelvoud van 2048 (`bs=2k`), maar met `factor` kun je de `bs`/`count` wellicht beter indelen (ideale `bs` is 50k tot 100k).

Schrijven van data en audio CD's

CD-R/RW's schrijven doe je (als root) met `cdrecord` (`apt-get install cdrecord`). Kijk eerst met `cdrecord -scanbus` of het SCSI-systeem goed werkt (IDE/ATAPI schrijvers: zie eerst hieronder), en met `cdrecord -inq dev=0,0,0` of de drive aan te spreken is (0,0,0 is SCSI-busnummer, apparaatnummer, subapparaat, zoals in de 'scanbus' uitvoer).

Het schrijven van een complete CD, single-session, track-at-once, met één data-track en twee audio-tracks gaat met een commando als `cdrecord -v speed=4 -eject dev=0,0,0 -data -pad /tmp/cdimage.iso -audio -pad /wavs/track2.wav -audio -pad /wavs/track3.wav`. Je kunt voorin een `-dummy` toevoegen om alles eerst uit te proberen. De `.wav`'s zijn standaard geluidsbestanden, 44.1kHz 16 bit stereo; als je CD-speler daar een enorme herrie van gaat maken, moet je aan het begin nog `-swab` toevoegen, maar dat hangt af van hoeveel `cdrecord` weet over jouw CD-schrijver (gebeurt meestal automatisch als dat nodig is). De `-pad` zorgt er steeds voor dat de tracks blijven voldoen aan de standaard, zodat elke CD-speler/CDROM-drive ermee overweg kan. Als je alleen data of alleen audio wilt schrijven, of meer dan drie tracks, moet je de commando-regel natuurlijk overeenkomstig aanpassen. Zoals gebruikelijk kun je ook `cdrecord` compleet naar je hand zetten, zie de manpage voor meer informatie.

Voor de meer visueel ingestelden is er `xcdroast`, die als back-end ook `cdrecord` gebruikt, maar persoonlijk vind ik de command-line sneller en makkelijker werken.

Aangepaste kernel voor IDE/ATAPI CD-recorders

CD-recorders met IDE-aansturing werken niet met de standaard kernel. Je moet daarvoor een nieuwe kernel compileren die de CD-apparaten niet met IDE, maar met SCSI-over-IDE aanstuurt (SCSI-ATAPI emulation). Zie paragraaf 9.6 voor informatie over de kernel. Voor SCSI-over-IDE moeten de volgende opties zijn ingesteld:

'Block devices' - 'Include IDE/ATAPI CDROM support': *uit*
id. - 'SCSI emulation support': aan
'SCSI support' - 'SCSI support': aan
id. - 'SCSI CD-ROM support': aan
id. - 'SCSI generic support': aan
id. - 'Verbose SCSI error reporting': aan

9. Programmeren en compileren

Bij Debian zitten talloze compilers en interpreters voor evenzoveel programmeertalen. In dit hoofdstuk kan ik je natuurlijk geen programmeertaal aanleren, maar wel vertellen hoe één en ander op een Debian (Linux) systeem werkt. Omdat alle source-code bij Debian meegeleverd wordt, kun je de **.deb** pakketjes zelf (opnieuw) compileren, maar ook niet-Debian software. Het compileren van een nieuwe Linux kernel blijkt tenslotte niet al te moeilijk te zijn.

9.1 Programmeren in C, C++ en Pascal

Je kunt in Debian programmeren in C, C++, Objective-C, Pascal, Java, Perl, Tcl/Tk, Fortran, Basic, Lisp, Scheme, Forth, Ada, Python, Mercury, Prolog, Rexx, Eiffel, Intercal en nog een paar andere. Elke programmeertaal heeft z'n eigen pakketje(s).

Ik zal hier een kort voorbeeldje geven hoe je een C programma compileert; vooraf **apt-get install gcc gcc-doc libc6-dev**.

Maak een tekstbestand **hallo.c** als volgt:

```
#include <stdio.h>

int main (void)
{
    printf("Hallo wereld!\n");
    return 0;
}
```

Compileer dat met **gcc -Wall -O2 -o hallo hallo.c** en probeer dan **./hallo** (de directory **./** moet er waarschijnlijk bij, omdat die niet in je pad zit). De **-o** geeft de output filenaam aan, **-Wall** laat alle warnings zien en **-O2** geeft een optimaliserende compilatie. **info gcc** geeft meer informatie over zaken die specifiek zijn voor deze compiler. Na **apt-get install manpages-dev glibc-doc** staat je met **info libc** een enorme hoeveelheid documentatie ter beschikking over letterlijk alle C standaard-functies, compleet met talloze voorbeelden (meer dan 1000 pagina's als je het uit zou printen), en is beknopte informatie via bijvoorbeeld **man strlen** bereikbaar.

Voor C++ compileren eerst **apt-get install g++ gcc-doc libstdc++2.10-dev**. Weer een voorbeeldje, **hallo2.cpp**:

```
#include <iostream.h>

int main (void)
{
    cout << "Hallo wereld!\n";
    return 0;
}
```

Compileer nu met **g++ -Wall -O2 -o hallo2 hallo2.cpp** en probeer **./hallo2**.

Ook Pascal is mogelijk (maar het gaat niet allemaal hetzelfde als Turbo Pascal), **apt-get install gpc gpc-doc libc6-dev**. Voorbeeldje, **hallo3.pas**:

```
program HalloWereld;

begin
    Writeln('Hallo wereld!');
end.
```

Compileer met **gpc -Wall -O2 -o hallo3 hallo3.pas** en probeer **./hallo3**. Meer informatie en voorbeelden staan in **/usr/doc/gpc**. Een misschien meer Delphi-compatible omgeving is 'Free Pascal', in 'fp-compiler' en andere 'fp-' pakketjes.

Merk op, dat er geen echte IDE (Integrated Development Environment) beschikbaar is; een simpele tekst-editor en **gcc**/**g++**/**gpc** volstaan. Als je toch een Borland-achtige IDE wilt hebben, kijk dan eens naar het pakketje 'xwpe'. Ook 'emacs20'/'xemacs21' worden vaak gebruikt als programmeer-omgeving. Als je veel source-code voorbeelden wilt zien, moet je eens op de Debian Source CD's kijken, of in het source-gedeelte van de FTP sites.

Voor grotere programma's heb je wellicht nog meer pakketjes nodig, zoals 'make', 'patch', 'libncurses5-dev', en vele andere '...-dev' pakketjes.

Programmeren van X toepassingen kan op veel verschillende manieren, van de low-level X-library tot high-level omgevingen als tcl/tk, visual-tcl, tix, gtk. Er zijn ook verschillende cross-platform toolkits die zowel onder Linux als Windows werken.

9.2 Make en Makefiles

De source-code van grote programma's staat vaak niet in één file, omdat dat niet meer overzichtelijk is. Er zijn dan meestal meerdere **.c**-files met programma-code en **.h**-files (*header files*) met definities van functies en macro's. Nu zou je bij het compileren alle **.c** en **.h** files achter elkaar kunnen plakken en in één keer door de compiler halen, maar dat is niet zo efficiënt. Je kunt ook alle **.c**-files per stuk compileren naar een **.o** *object file*, en dan alle **.o**-files aan elkaar plakken (*linken*). Als er maar één **.c**-file veranderd is, hoef je alleen die éne bijbehorende **.o**-file opnieuw te maken (en dat kost relatief veel tijd) en dan kun je alle **.o**'s weer linken tot een uitvoerbaar programma (en dat is heel snel).

Natuurlijk wil je die hele administratie van **.c** en **.h** en **.o**-files die wel of niet veranderd zijn, of opnieuw gemaakt moeten worden, en dan opnieuw linken, niet zelf bij gaan houden. Dat hoeft gelukkig ook niet, daar hebben we **make** voor (**apt-get install make**). In een **Makefile** zet je dat **programma** gemaakt moet worden van **deel1.o** en **deel2.o** en **deel3.o**, en dat **deel1.o** gemaakt moet worden van **deel1.c** en **deel1.h** en **global.h**, enzovoort. Dan zoekt **make** zelf uit welke files veranderd zijn (de laatst-veranderd datum/tijd in de directory-listing!) en gaat dan al het nodige werk doen om het programma te maken.

Makefiles kun je heel ingewikkeld maken, of juist heel simpel door alle ingebouwde slimmigheidjes van **make** te gebruiken. Ik zal hier een heel eenvoudig voorbeeldje geven zodat je weet hoe het eruit ziet, meer informatie kun je lezen in **info make**, en er zijn vele duizenden voorbeeld-Makefiles te vinden op de Debian Source CD's en het Internet. Zorg dat je de C-development pakketjes en 'make' hebt geïnstalleerd; zet **hallo.c** uit de vorige paragraaf in een nieuwe en verder lege directory. Maak dan in die directory een file **Makefile** met inhoud:

```
hallo: hallo.o
	gcc -o hallo hallo.o

hallo.o: hallo.c
	gcc -Wall -O2 -c -o hallo.o hallo.c
```

Let op: het inspringen moet met een TAB! Geef gan het commando **make hallo** of gewoon **make** (maakt standaard het eerst-genoemde bestand). Compileren naar een **.o**-file gebeurt met de **-c** switch, en **gcc** ziet zelf dat 'ie moet linken als er enkel object files opgegeven worden. Doe dan **rm hallo hallo.o** en maak verander de **Makefile** in:

```
CC=gcc
CFLAGS=-Wall -O2

hallo: hallo.o
```

Weer enkel **make** maakt opnieuw het programma, en **make** is zelf zo slim dat hij precies weet wat er moet gebeuren.

9.3 Debian pakketjes (her-)compileren

Debian pakketjes komen altijd compleet met source-code, en je kunt ze bijzonder eenvoudig zelf compileren. Dat kan echter wel behoorlijk veel disk-ruimte kosten door alle benodigde development-pakketjes. Eerst als root **apt-get install task-debian-devel** om de essentiële pakketjes te installeren. Bij de configuratie van 'debconf' kies je 'dialog' en 'medium'; voor 'cvs' kies je '/var/cvs', 'ignore', en 'pserver' *niet* aanzetten; na afloop nog een keer 'ignore' bij 'cvs'.

Debian source pakketjes bestaan meestal uit een drietal bestanden, een **.orig.tar.gz** (de originele *upstream* source code), een **.diff.gz** (Debian-specifieke aanpassingen) en een **.dsc** (index-file voor de **.orig.tar.gz** en **.diff.gz**). Je kunt die (meestal) drie bestanden downloaden uit het 'source'-gedeelte van een Debian FTP site, maar ook direct van de Debian Source CD's halen. Je pakt de bestanden uit (als gewone gebruiker) met bijvoorbeeld **dpkg-source -x /cdrom/dists/potato/main/source/text/less_346-7.dsc**, dan verschijnt een nieuwe sub-directory **less-346/** in de huidige directory.

Na **cd less-346** kun je met **less debian/control** kijken naar de "Build-Depends(-Indep):" regels, de daargenoemde pakketjes moeten eerst nog geïnstalleerd worden, dus in dit geval weer even root worden en **apt-get install libncurses-dev debhelper** doen. Als alles klaar staat geef je (weer als gewone gebruiker) het commando **fakeroot debian/rules binary**, en er wordt gecompileerd. Na afloop staat er een **.deb** pakketje in de bovenliggende directory (**cd .. ; ls -alF**). Je kunt het pakketje als root installeren met **dpkg -i less*.deb**.

Het ophalen van de source-code kan nog makkelijker als je ook de Source CD's met **apt-cdrom add** hebt gescand, of 'deb-src' regels voor een FTP server hebt toegevoegd aan **/etc/apt/sources.list**. Dan kun je als gewone gebruiker(!) in een lege directory **apt-get source less** doen, en wordt de source automatisch 'ergens' vandaan gehaald en uitgepakt. Je kunt op deze manier ook in één keer compileren met **fakeroot apt-get source -b less**, maar dan moeten alle benodigde pakketjes ('libncurses5-dev' dus) vooraf al geïnstalleerd zijn.

Merk op dat niet alle source-pakketjes juiste "Build-Depends:" regels hebben; in dat geval moet je gewoon proberen te compileren en zien wat er mis gaat. Dan het juiste pakketje erbij installeren, **fakeroot debian/rules clean** en opnieuw proberen. Tenslotte: deze procedure werkt niet of niet goed bij ingewikkelde pakketjes; meestal is er dan een README die meer uitleg geeft.

9.4 Compileren van pine en pico

Het zeer gebruikers-vriendelijke e-mail programma **pine** en de editor **pico** zijn speciaal, omdat die alleen in sourcecode-vorm verspreid mogen worden en door de gebruiker zelf gecompileerd moeten worden. Dat gaat vergelijkbaar met een standaard-pakketje, maar toch iets anders omdat de sourcecode niet in source-files staat maar in een **.deb** pakketje. Ik zal hier een procedure aangeven die op zichzelf staat (en zo min mogelijk diskruimte gebruikt), daar heb je de vorige paragraaf dus niet bij nodig.

Installeer als root eerst de nodige pakketjes met **apt-get install libncurses5-dev mime-support gcc dpkg-dev fakeroot pine4-src pine4-diffs**. De laatste twee komen uit de 'non-free' sectie en staan dus niet op de officiële CD's; je kunt ze ook weglaten uit de **apt-get** regel, van een Debian FTP site downloaden en met **dpkg -i** installeren.

Log dan in als gewone gebruiker en ga bijvoorbeeld in **/tmp** staan. Geef daar het commando **dpkg-source -x /usr/src/pine4/pine4_4.21-1.dsc**. Dan **cd pine4-4.21** en compileer met **fakeroot debian/rules binary** (dit kan een tiental minuten duren). Na het compileren staan in **/tmp** vier nieuwe .deb-pakketjes, die je (als root) met bijvoorbeeld **dpkg -i *.deb** kunt installeren.

Als je een zojuist gemaakt pakketje op meerdere computers nodig hebt, is er niemand die je belet ze (intern!) te distribueren; haal ze dan wel even uit **/tmp** weg en zet ze op een veilige plaats.

9.5 Niet-Debian source code

Het kan gebeuren dat je op het Internet een Linux-programma tegenkomt dat nog niet in Debian zit (maar die kans wordt steeds kleiner). In zo'n geval gaat het meestal om een **.tar.gz** of **.tar.bz2** archief. Pak dat (als gewone gebruiker!) uit in een lege directory, meestal wordt er dan een nieuwe sub-directory gemaakt met de sourcecode. Als daar een programma **configure** bestaat, doe dan eerst iets als **./configure --prefix=/tmp** (**./configure --help** voor meer info). Daarna zou **make** moeten compileren. Het is misschien even zoeken naar de executables; als je ze niet zomaar kunt vinden, of als ze niet zomaar werken, kan **make install** soms uitkomst bieden; als je de **--prefix** had opgegeven, komen de programma's meestal in **/tmp/bin** of iets dergelijks. Je kunt de gecompileerde

programma's in je eigen **~/bin/** directory zetten, maar ook system-wide toegankelijk maken door ze in **/usr/local/bin** te zetten.

9.6 Het compileren van een nieuwe kernel

Tegenwoordig is het in de meeste situaties niet meer nodig om zelf een nieuwe kernel te compileren. Nieuwe kernels kun je meestal zo uit het 'woody' of 'sid'-archief op de Debian FTP-sites halen en installeren (ook al heb je een 'potato' systeem), dat zijn de 'kernel-image' pakketjes. Alleen als je een erg niet-standaard apparaat hebt (of een IDE/ATAPI CD-recorder) kan her-compileren van de kernel noodzakelijk zijn; hiervoor heb je een 'kernel-source' pakketje nodig.

De manier van kernel-compileren die ik hier behandel, is niet de standaard manier (uit de Kernel-HOWTO) maar een Debian-specifieke manier, die veel makkelijker werkt. Eventueel kun je ook de standaard-manier gebruiken, maar dan geef ik geen enkele garantie voor het welslagen van die onderneming.

Er zijn diverse hardware-fabrikanten, die Linux kernel-add-on's leveren bij hun produkten. Wees daar heel voorzichtig mee! Over het algemeen zitten die drivers namelijk al in de standaard kernel-source, en krijg je enorme problemen als je de add-on's wilt gebruiken.

De 'kernel-source-xxx'-pakketjes zijn bij wijze van uitzondering gewone pakketjes; ze staan op de *Binary* CD's (dus niet op de Source CD's), en in het 'binary-i386'-gedeelte van de Debian FTP-sites.

Kijk vooraf met **df** of je nog voldoende schijfruimte beschikbaar hebt. De uitgepakte kernel-source is tenminste 85 MB, daar komt nog 35 MB bij als je de benodigde pakketjes nog niet had geïnstalleerd, en tijdens het compileren is daar bovenop nog maximaal 15 MB nodig.

Installeer als root de benodigde pakketjes met **apt-get install kernel-source-XXX kernel-package libc6-dev bin86 libncurses5-dev gcc fakeroot dpkg-dev bzip2**; zet daar nog **tk8.2-dev** achter als je de X configuratie wilt gebruiken. Voor de 'XXX' moet je dus de juiste kernel-versie opgeven, bijvoorbeeld '2.2.17'. Als je een 'kernel-source' pakketje van een andere locatie wilt gebruiken, bijvoorbeeld de nieuwste versie uit het 'woody' of 'sid' archief, kan dat (meestal) ook; laat hem dan weg in de **apt-get** regel, download 'm apart en installeer 'm met **dpkg -i kernel-source-...deb** (de volledige filenaam dus). Let wel: deze procedure werkt *alleen* met Debian kernel-source pakketjes, en *niet* met de officiële on-aangepaste **.tar.gz**/**bz2** archieven.

Procedure, als niet-root uitvoeren ("<versie>" is bijvoorbeeld "2.2.17"; commentaar hieronder):
1) **cd /tmp**
2) **tar xIvf /usr/src/kernel-source-<versie>.tar.bz2**
3) **cd kernel-source-<versie>**
4) **make-kpkg clean** (negeer eventuele foutmeldingen)
5) **make menuconfig** (of **make xconfig** als je de X pakketjes hebt geïnstalleerd; *geen* **make dep** erachter!)
6) **fakeroot make-kpkg --revision=9:mijnkernel.1.0 kernel_image** (en wacht 10 minuten tot een paar uur)

Na stap 6 staat er een nieuw 'kernel-image' pakketje in **/tmp**.
7) maak een kopie van **/tmp/kernel-image-<versie>_mijnkernel.1.0_i386.deb** op een veilige plaats

Installeer het nieuwe 'kernel-image' pakketje, nu wel als root:
8) als er een directory **/lib/modules/<versie>** bestaat: **mv /lib/modules/<versie> /lib/modules/<versie>.old**
9) **dpkg -i /tmp/kernel-image-<versie>_mijnkernel.1.0_i386.deb** (vraagt om eventuele LILO-configuratie te updaten)
10) bij gebruik van een bootfloppy of LOADLIN: zet de nieuwe kernel (**/vmlinuz**) op de juiste plaats
11) reboot computer
12) als je stap 8 had gedaan: **rm -rf /lib/modules/<versie>.old**

In stap 1 t/m 6 (die je als niet-root uitvoert) wordt de kernel-source uitgepakt en gecompileerd in **/tmp**. Dat mag ook in een willekeurige andere directory (zo kun je meerdere kernels tegelijk compileren), maar let erop, dat er vooraf tenminste 100 MB vrije ruimte op de betreffende schijf moet zijn. NB: **/tmp** wordt geleegd bij elke reboot!
Voor stappen 8 t/m 12 moet je wel root zijn.

Stap 5, het configureren van de kernel, is het enige 'echte werk' (de rest stelt niets voor). Je moet het gewoon eens proberen en vooral overal de helptekst van opvragen (en ook het configuratie-gedeelte van de Kernel-HOWTO raadplegen). Als er in zo'n helptekst gezegd wordt dat je een optie 'als module' kunt configureren, dan staat daar meestal bij dat je een bepaald tekstbestand moet lezen. Bij Debian hoeft dat echter niet, want modules werken al. Een module is gewoon een stukje kernel dat niet altijd in het geheugen zit, maar 'op aanvraag' ('*on demand*') wordt bijgeladen; het zijn de 'device drivers' die je tijdens de installatie bent tegengekomen. Je kunt modules ook met de hand laden en 'uitladen', zie daarvoor de manpages van **modprobe** en **lsmod** (en eventueel van **depmod**, **insmod** en **rmmod**). Meestal is het aan te bevelen zoveel mogelijk als module te laten maken, *behalve* het filesystem van de root partitie, ext2 ('Second extended fs support') dus. De allereerste keer kun je de voorbeeld-configuratie aanhouden uit de volgende paragraaf.

De naam 'mijnkernel.1.0' in stap 6 (en verder) mag je zelf verzinnen. Ik gebruik meestal <computernaam>.<versie> met versie oplopend, 1.0, 1.1, 1.2 enz. (en 2.0 als de hardware verandert), bijvoorbeeld 'kennedy.1.6'. Er wordt een Debian pakketje gemaakt met die opgegeven naam, zie stap 7 en 9.

Stappen 8 t/m 12 kunnen trouwens ook op een andere computer uitgevoerd worden, of op een hele serie computers. De compilatie wordt meestal op de snelste computer gedaan; dan heeft een langzame computer alleenmaar het nieuw gefabriceerde 'kernel-image' pakketje nodig: veel sneller.

Tussen stap 5 en 6 maak ik altijd een backup van de **.config** file, die de complete configuratie van de kernel bevat. Om precies diezelfde kernel te maken hoef ik alleen maar de juiste 'kernel-source' uit te pakken en de juiste **.config** erheen te kopiëren. Heel handig als je op één computer kernels voor veel verschillende machines compileert. De naam van de backup is dan bijvoorbeeld **/home/ikke/kernel/.config.2.0.36.mijnkernel.1.0**.

Stap 8 en 12 zijn nodig als je een nieuwe versie maakt van de kernel die je op dat moment al draait. De modules van de draaiende kernel kunnen namelijk niet door **dpkg** worden verwijderd. Nieuwe modules zouden er dus overheen geïnstalleerd worden, en dan blijft er een mengelmoes van oude en nieuwe modules achter. Om dat te voorkomen, geven we de modules-directory eerst een andere naam, en verwijderen we hem met de hand als de nieuwe kernel draait.

Meer informatie over 'de Debian manier' van het 'bakken' van een kernel staat in **/usr/share/doc/kernel-package/README.gz** (en andere bestanden in die directory). Ook de directory's **/usr/share/doc/kernel-source-<versie>** en **Documentation/** (van een uitgepakte kernel source) bevatten veel lezenswaardige informatie.

Als er vreemde problemen zijn met compileren (bijv. 'signal 11' van **gcc**), kan het zijn dat je slechte geheugenchips hebt, of een ander hardwareprobleem. Zie daarvoor de Internetpagina http://www.bitwizard.nl/sig11/.

9.6.1 Voorbeeld-configuratie

Hieronder staat een volledige configuratie voor een bijzonder minimale kernel. Deze kernel heeft ondersteuning voor floppy drives, IDE harddisks en CDROM drives, maar niet voor parallelle poort, seriële poort ('gewone' muizen), PS/2 muizen, SCSI apparaten, 'vreemde' CDROM drives en geluidskaarten, kan niet netwerken en kan geen DOS/Windows partities mounten.

Bij deze configuratie geldt: alles is uit (toets 'n') behalve wat hieronder is aangegeven (aan = toets 'y', 'M'odulair = toets 'm'). Dit is alleen gecontroleerd met een 2.2.17 kernel, vooral 2.4.xx kernels hebben waarschijnlijk enkele andere opties nodig.

'Processor type and features' - 'Processor family': kies juiste waarde, zie <Help>; '386' werkt altijd.
id. - 'Maximum Physical Memory': 1GB is meestal voldoende
id. - 'Math emulation': aan
id. - 'MTRR (Memory Type Range Register': aan
'Loadable module support' - Alle opties: aan
'General setup' - 'Networking support': aan
id. - 'PCI support': aan
id. - 'PCI access mode': kies Any
id. - 'PCI quirks': aan
id. - 'Backwards-compatible /proc/pci': aan
id. - 'System V IPC': aan
id. - 'Sysctl support': aan

id. - 'Kernel support for ELF binaries': aan
'Block devices' - 'Normal PC floppy disk support': aan
id. - 'Enhanced IDE/MFM/RLL disk/cdrom/tape/floppy support': aan
id. - 'Include IDE/ATA-2 DISK support': aan
id. - 'Include IDE/ATAPI CDROM support': aan
id. - 'CMD640 chipset bugfix/support': aan
id. - 'RZ1000 chipset bugfix/support': aan
id. - 'Generic PCI IDE chipset support': aan
id. - 'Generic PCI bus-master DMA support': aan
'Networking options' - 'Unix domain sockets': aan
id. - 'TCP/IP networking': aan
'Character devices' - 'Virtual terminal': aan
id. - 'Support for console on virtual terminal': aan
id. - 'Unix98 PTY support': aan
id. - 'Maximum number of Unix98 PTYs in use': 256 is voldoende
'Filesystems' - 'ISO 9660 CDROM filesystem support': 'M'odulair
id. - 'Microsoft Joliet CDROM extensions': aan
id. - '/proc filesystem support': aan
id. - '/dev/pts filesystem for Unix98 PTYs': aan
id. - 'Second extended fs support': aan
id. - 'Native Language Support' - 'Codepage 437 (United States, Canada)': 'M'odulair
id. - id. - 'Codepage 850 (Europe)': 'M'odulair
id. - id. - 'NLS ISO 8859-1 (Latin 1; Western European Languages)': 'M'odulair
'Console drivers' - 'VGA text console': aan

9.6.2 Compileren van extra kernel modules: PCMCIA, ALSA

Als je PCMCIA-modules of ALSA-modules gebruikt, moet je die opnieuw compileren voor de nieuwe kernel. Ik zal hier PCMCIA noemen, maar ALSA werkt precies hetzelfde. Installeer eerst als root de source, die net als de kernel-source in een standaard **.deb**-pakketje zit, met **apt-get install pcmcia-source**. Ook hier is er meestal geen probleem als je de versie uit 'woody' of 'sid' gebruikt op een 'potato' systeem. De procedure is (alles als niet-root):

1) **cd /tmp**
2) **tar xzvf /usr/src/pcmcia-cs.tar.gz**
3) **cd modules/pcmcia-cs**
4) **fakeroot debian/rules KSRC=/tmp/kernel-source-<versie> binary-modules**
5) **fakeroot debian/rules KSRC=/tmp/kernel-source-<versie> binary-cs**

In stap 4 en 5 moet je het juiste pad naar de zojuist-gecompileerde kernel-source opgeven. Na deze stappen staan in **/tmp/modules** twee nieuwe pakketjes, één met modules en één met ondersteunende programma's. Je kunt ze beide installeren op de doel-computer.

9.6.3 Oude kernel terugzetten bij problemen

Als de nieuwe kernel niet wil opstarten, moet je een oude kernel terugzetten. Als je nog naar Linux kunt opstarten, en bijvoorbeeld de Binary-1 CD kunt mounten, kun je met **dpkg -i /cdrom/dists/potato/main/binary-i386/base/kernel-image-2.2.17_2.2.17*.deb** de standaard-kernel weer installeren.

Je kunt ook opstarten van een rescue floppy of de Binary-1 CD, en dan geen Enter geven op de 'boot:'-prompt, maar **rescue root=/dev/hdXX ro** (vul de juiste root-partitie in). Dan kom je met de standaard-kernel op je eigen systeem terecht, en kun je met bovenstaand commando de kernel weer van CD terugzetten. Alleen werkt dat waarschijnlijk niet als je geen IDE/SCSI CDROM-drive hebt; in dat geval moet je eerst (met een ander Operating System) het kernel-pakketje in gedeelten op diskettes zetten, die je dan kunt mounten, kopiëren (of **mcopy**) en aan elkaar plakken (**cat a b c > totaal**), en dan installeren.

In het uiterste geval kun je een nieuw Debian basis-systeempje installeren (of de harde schijf in een ander Debian-systeem hangen), en daar op één of andere manier de **kernel-image...deb** toegankelijk krijgen. Dan kun je je 'normale' root partitie mounten op bijvoorbeeld **/mnt** (van het nieuwe systeempje) en vervolgens met **dpkg --root=/mnt -i kernel-image...deb** de standaard-kernel op de 'alternatieve root partitie' installeren.

10. Teksten, plaatjes en printen

De meest voorkomende computer-toepassingen zijn wel het schrijven van teksten en het tekenen van plaatjes, die dan (nog steeds) vaak worden afgedrukt op papier. Debian is voorzien van vele programma's voor deze toepassingen, die ik in dit hoofdstuk beknopt zal toelichten. De meeste genoemde programma's zitten in gelijknamige pakketjes, waar dat niet zo is staat de juiste pakket-naam erbij.

10.1 PostScript

Eerst iets over PostScript, omdat dit vaker zal terugkomen. PostScript is van origine een taal voor de aansturing van laserprinters. Een PostScript-bestand (**.ps**), of een PostScript-datastroom naar een printer, is in feite een programma in de programmeertaal 'PostScript', die de processor van de printer instructies geeft hoe de te-printen pagina's eruit moeten gaan zien, aan de hand van data die in dat bestand/die datastroom wordt meegestuurd. Een PostScript-file is altijd een gewoon tekst-bestand, dat je met bijvoorbeeld **less** kunt bekijken, of eventueel zelfs wijzigen als je daar behoefte aan hebt. In PostScript kun je bitmap-plaatjes kwijt (later meer daarover), maar liever vector-plaatjes. Die zijn in principe oneindig scherp, en dat is belangrijk voor printers met een zeer hoge resolutie.

Er is een goede PostScript-viewer voor X, namelijk **gv** (van ghostview), die gebruik maakt van **gs** (ghostscript), dé PostScript interpreter voor 'normale computers' (in plaats van printers). Installeer als root met **apt-get install svgalib-dummyg1 gv gsfonts**. Als 'libpaperg' vraagt om de 'Default paper name', antwoord dan 'a4'.

Geef dan (weer als gewone gebruiker) uit bijvoorbeeld een xterm het commando **gv /usr/share/doc/gs/examples/alphabet.ps.gz**, en bekijk het alfabet. Je kunt de tekst met de muis bewegen; inzoomen door slepen met de middelste muisknop en na het loslaten een factor kiezen. Met de menu's bovenin kun je bijvoorbeeld anti-aliasing inschakelen, waardoor de letters er veel minder ruw uit gaan zien (houd de muisknop ingedrukt en 'sleep' naar de keuze in het menu). Ook kun je de papier-grootte, rotatie en standaard-vergroting aanpassen. Er staan meer voorbeelden in de genoemde directory, bijvoorbeeld **colorcir.ps.gz** (bekijk die ook eens met **zless**), **escher.ps.gz** en **tiger.ps.gz**.

Naast het 'pre-viewen' op het scherm, kan **gs** ook PostScript-files her-formatteren voor praktisch alle niet-PostScript printers (matrix, inktjet enz., zie verderop). Omdat je in PostScript alles kunt uitdrukken wat printers kunnen, en omdat het moeiteloos bruikbaar is voor verreweg de meeste printers, en omdat het bovendien een zeer gemakkelijk uitvoerformaat is voor tekstverwerkers en tekenprogramma's, is PostScript het de-facto 'tussen-formaat' geworden voor printen op UNIX systemen.

10.2 Platte tekst

'Gewone tekst', '*plain text*', 'platte tekst' ofwel 'ASCII tekst' wordt overal gebruikt waar speciale opmaak niet nodig of gewenst is, bijvoorbeeld bij alle configuratie-bestanden op UNIX-achtige systemen, bij programma-broncodes en e-mail berichten. Je hebt het al bekeken met **cat** en **less**, en ge-edit met **ae**. Veel gevorderde Linuxers gebruiken óf **vi** ('nvi') óf (**x**)**emacs** ('emacs20' of 'xemacs21'), maar die zijn vrij lastig om aan te leren. (Tip: vi verlaat je met <Escape>:q! en emacs met ^X^C.)

Persoonlijk vind ik **pico** (zie paragraaf 9.4) of diens kloon **nano** gemakkelijk werken, juist omdat die zo weinig overbodige features heeft. Korte gebruiksaanwijzing: ^K (Ctrl-K) knipt een regel, evt. herhalen, ^U zet ze weer terug, evt. herhalen. Met ^6 - pijltjes - ^K kun je ook blokken knippen/plakken. Als pijltjes niet werken is ^A: home, ^E: end, ^Y: PgUp, ^V: PgDn, ^W^Y: bovenaan, ^W^V: onderaan. ^G voor hulp.

Maar er zijn talloze andere editors beschikbaar, zoals **ee**, **jed**, **joe**, **vim** en **zed** voor gebruik op de textmode virtuele console, en bijvoorbeeld **cooledit**, **gedit**, **gnp** ('gnotepad+'), **xcoral** en **xjed**. Het is aan jou om de leukste te vinden, of in ieder geval de editor waarmee jij het beste overweg kunt.

Sommige programma's starten automatisch een editor als dat nodig is (bijvoorbeeld om een e-mailtje te schrijven), dan wordt de standaard-editor gebruikt. Die kun je als root 'system-wide' instellen met **update-alternatives --config editor**; je kunt hem ook zelf kiezen door in zowel de **EDITOR** als **VISUAL** environment variabele de juiste programmanaam in te stellen (zie ook paragraaf 7.6).

Platte tekst omzetten naar PostScript gaat met bijvoorbeeld 'mpage', iets als **mpage -2 -f -H /etc/fb.modes > /tmp/fb.modes.ps**. Krachtiger is 'a2ps', bijvoorbeeld **a2ps -2 /etc/fb.modes -o /tmp/fb.modes.ps**; **a2ps** zal bijvoorbeeld **.c**-files automatisch van 'highlighting' voorzien.

10.3 Opgemaakte tekst

Als je brieven of boeken schrijft, gebruik je meestal geen 'platte' tekst, maar één of andere vorm van opmaak, zoals die bijvoorbeeld door een moderne tekstverwerker kan worden verzorgd.

Plain-text source

Eén manier om tekst op te maken is door in 'platte' tekst opmaak-aanwijzingen te geven. Het bekendste voorbeeld is wel HTML, waar webpagina's mee gemaakt worden. Je kunt dit soort bestanden met elke 'gewone' tekst-editor wijzigen; sommige editors hebben bijvoorbeeld een speciale HTML-mode die je nuttig kunt gebruiken.

Om boeken of wetenschappelijke publikaties te schrijven wordt vaak LaTeX gebruikt (spreek uit: "Latech"), **apt-get install latex telex-extra svgalib-dummyg1 gv gsfonts**. Snel voorbeeld: **cd /tmp**, **gunzip < /usr/share/doc/texmf/tetex/TETEXDOC.tex.gz > TETEXDOC.tex**, twee maal **latex TETEXDOC.tex** (en schrik niet). Dan **xdvi TETEXDOC.dvi** om in X te pre-viewen (langzaam), of **dvips -f TETEXDOC.dvi > TETEXDOC.ps** en dan bijvoorbeeld **gv TETEXDOC.ps** (veel sneller) of de **.ps** printen. Voor een zeer complete (en veel grotere) LATEX-omgeving: **apt-get install task-tex hyperlatex**. Op ftp://ftp.funet.fi/pub/TeX/tex-archive/info/lshort/english/ kun je een eenvoudige introductie vinden in de wereld van LaTeX.

Voor Debian documentatie wordt vaak een 'vorm' van SGML gebruikt, DebianDoc SGML. Andere projecten gebruiken DocBook. SGML kan naar veel verschillende formaten worden geconverteerd, waaronder natuurlijk PostScript. Met **apt-get install task-sgml** krijg je een complete SGML-omgeving, die (onder andere) beide formaten aan kan.

What-You-See-Is-What-You-Get

De meest aansprekende manier om 'opgemaakte' teksten te produceren is met een 'visuele' omgeving, maar dat biedt vaak niet de flexibiliteit van de 'plain-text source' manieren. In Debian zit bijvoorbeeld Lyx (spreek uit: "Liks"), een X front-end voor LaTeX (dus naast 'lyx' ook de LaTeX-pakketjes installeren, zie boven). Een andere editor is **ted**, die **.rtf**-files (Rich Text Format) gebruikt en **.ps** kan produceren. Ook **abiword** is de moeite van het proberen waard; hoewel deze tekstverwerker pas in een vroeg ontwikkel-stadium verkeert, kun je er al heel aardige documenten mee maken.

Naast deze 'vrije' software zijn er ook nog diverse niet-vrije ('*proprietary*') office-pakketten voor Linux. De belangrijkste zijn *WordPerfect Office* van Corel (http://linux.corel.com) en *StarOffice* van Sun Microsystems (http://www.sun.com/products/staroffice). Deze pakketten zijn meestal vrij duur, maar in het algemeen krijg je wel waar voor je geld. Er is een oudere versie (8.0) van WordPerfect voor Linux verkrijgbaar op het Internet (bestanden als **GUILG00.GZ** of **.TAR**) die gratis is voor persoonlijk gebruik, maar daarvoor moet je wel even een (gratis) registratienummer halen op http://linux.corel.com.

10.4 Plaatjes

Bitmap

Foto's en dergelijke worden opgeslagen in *bitmap*-formaat; van elk beeldpunt (*pixel*) is de precieze kleur bekend. Maar om de bestanden niet al te groot te laten worden, is het aantal beeldpunten (*resolutie*) meestal vrij beperkt. Je kunt bitmap-plaatjes maken met de Gimp (*GNU Image Manipulation Program*), pakketjes 'gimp' of 'gimp1.1' (de 'onstabiele' development-versie). Dit zeer populaire programma werkt met 'layers', die je je het beste kunt voorstellen als transparanten (overhead sheets) die over elkaar schuiven en die je onafhankelijk kunt bewerken. Een (grote!) handleiding zit in 'gimp-manual'. Ook 'imagemagick' wordt veel gebruikt, de programma-naam is **display**; probeer alle muisknoppen. Bitmap-viewers zijn onder andere **eeyes**, **xli**, **xloadimage**, **gtksee**, **qiv** en de 'non-free' **xv**. De meeste viewers kun je met de 'q'-toets verlaten. Het printen (=omzetten naar PostScript) van bitmap-plaatjes vind ik persoonlijk het makkelijkste met **xv**. Bepaalde scanners kun je gebruiken met SANE (*Scanner Access Now Easy*), pakketjes 'sane', 'sane-gimp1.1', 'xsane' en/of 'xsane-gimp1.1'. Verder ondersteunt **gphoto** een groot aantal digitale fotocamera's, maar voor verschillende camera's en webcams zijn (ook) aparte pakketjes beschikbaar.

Vector
Technische tekeningen, plattegronden en grafieken bestaan meestal enkel uit lijnen en tekens. In die gevallen is het veel eenvoudiger om alleen die *vectoren* op te slaan in plaats van de kleur van elk beeldpunt. Een bijkomend voordeel is dat de verschillende onderdelen van de tekening dan onafhankelijk kunnen worden gewijzigd, en dat de resolutie onbeperkt is. Traditioneel wordt hiervoor **xfig** veel gebruikt, **apt-get install xfig xfig-doc transfig**. Met 'File' - 'Export...' kun je een PostScript of Escapsulated PostScript plaatje krijgen (tekstverwerkers kunnen vaak ruimte voor **.eps**-bestanden reserveren, en ze dan letterlijk kopiëren in hun **.ps**-uitvoer). Met **pstoedit** kun je veel **.(e)ps**-bestanden in **.fig**-formaat omzetten zodat je ze kunt bewerken. Andere vector-tekenprogramma's zijn bijvoorbeeld **sketch** en **tkpaint**.

10.5 Printen

In Linux kun je op veel verschillende manieren iets uit de printer krijgen. De meest 'basic' manier is om direct iets te sturen naar de printerpoort, met een commando **echo Hallo daar > /dev/lp0**, maar dat mag alleen root.

Een 'nettere' manier is het installeren van een *print spooler*, dat is een programma dat alle printopdrachten even in een wachtrij zet, en na elkaar naar de printer stuurt; **apt-get install lprng**, wijzig in **/etc/printcap** de aanduiding 'lp=/dev/lp1' in 'lp=/dev/lp0', en geef tenslotte het commando **/etc/init.d/lprng restart**. Je kunt nu (als niet-root) met **echo Hallo daar | lpr** of **lpr <bestand>** printen. Met **lpq** kun je de status van de printer-wachtrij zien, en met **lprm <jobnummer>** kun je een lopende printopdracht uit de wachtrij verwijderen (het jobnummer is te zien met **lpq**). Deze manier van printen is echter alleen geschikt voor tekstfiles.

Het kan gebeuren, dat de printer geen 'terugloop' heeft (het *staircase effect*), dat wil zeggen dat de regels schuin onder elkaar worden geprint, en niet allemaal aan de linkerkant van het papier beginnen. Je printer is dan ingesteld op een DOS-achtige modus, want DOS zendt de 'nieuwe regel' en 'terugloop' commando's apart. Linux gebruikt alleen het 'nieuwe regel' commando, en verwacht dat de printer dan automatisch een 'terugloop' uitvoert. Als jouw printer dat niet doet, installeer dan 'sysutils' en print met een commando als **cat <bestand> | todos | lpr**. Een directory-uitdraai krijg je bijvoorbeeld met **ls -alF /home | todos | lpr**.

Een veel 'slimmere' manier van printen kan met MagicFilter, **apt-get install lprng magicfilter svgalib-dummyg1 gs gsfonts**. Tijdens de configuratie kan 'libpaperg' vragen naar het papierformaat dat wordt gebruikt, in Europa is dat "a4". Magicfilter geeft je waarschijnlijk ook gelijk de gelegenheid om de printer(s) te configureren (zo niet, gebruik dan **magicfilterconfig --force** als **apt-get** klaar is). Elke printer heeft een lange en een korte naam (die je zelf mag verzinnen), een device (**/dev/lp0** voor de eerste printerpoort!), en een inputfilter. Dat inputfilter is afhankelijk van het printer-type, en er wordt een lijst met mogelijkheden getoond; kies diegene die het dichtst in de buurt komt.

Als je nu **ls -alF /home | lpr** doet (zonder **todos** ertussen), zou alles opeens netjes onder elkaar moeten komen. MagicFilter doet namelijk wat printer-afhankelijke *pre-processing* op de invoerfile, voor die naar de printer gestuurd wordt. Met **lpr -P<kortenaam>** kun je op de tweede, derde, enz. printer printen, als je die had geconfigureerd.

Nu je MagicFilter hebt geïnstalleerd, zou je ook PostScript-bestanden moeiteloos moeten kunnen printen; probeer bijvoorbeeld eens **lpr /usr/share/doc/gs/examples/alphabet.ps.gz** (en let erop, dat je niet eens zelf hoeft te **gunzip**-pen!). **lpq** en **lprm** werken trouwens nog steeds, **-P<kortenaam>** erachter om met andere printers te werken.

Magicfilter heeft een 'concurrent' in apsfilter, **apt-get install lprng apsfilter a2ps svgalib-dummyg1 gs gsfonts**. Kies er *niet* voor om **apsfilterconfig** tijdens de configuratie te draaien, maar doe dat 'met de hand' als **apt-get** klaar is. In het configuratie-hoofdmenu moet je in ieder geval de opties 1 t/m 5 doorlopen. In stap 2 moet je **/dev/lp0** kiezen voor de (eerste) parallelle printerpoort. Na de configuratie weer **/etc/init.d/lprng restart**. Met **-Plp** print je nu 'automatisch goed', **-Pascii** voor 'geforceerd ASCII' en **-Praw** voor 'filterloos'.

De printer-instellingen staan in **/etc/printcap**. Veel meer interessante informatie kun je lezen in de manual pages, in **/usr/share/doc** en in diverse HOWTO's. Je kunt ook printers delen/gebruiken met/van Windows machines, installeer daarvoor het pakketje 'samba' en lees de documentatie (onder andere **man smb.conf**).

11. Audio en video

Multi-mediale toepassingen van de computer komen steeds meer in de belangstelling. Ook onder Linux zijn hiervoor diverse mogelijkheden, die ik hier kort zal bespreken.

11.1 Driver voor de geluidskaart

Als je geluid wilt horen, heb je een geluidskaart nodig. Linux ondersteunt vele geluidskaarten, maar het is vaak een kunst om ze aan de praat te krijgen. In alle gevallen moet je als root met **modconf** één (of misschien meer) modules uit de sectie 'misc' (of misschien 'sound' in heel nieuwe kerels). PCI kaarten zijn het makkelijkst, je kunt met **lspci** (pakketje 'pciutils) zien wat voor type het is, en de bijbehorende module zonder enige parameter installeren.

ISA kaarten zijn vaak lastiger. Je moet eerst te weten zien te komen om wat voor audio-chip het gaat; meestal helpt de tekst op de chip, maar soms moet/kun je ook informatie halen uit de bijgeleverde documentatie en/of de Windows driver. Vervolgens moet je het I/O-adres(sen), IRQ's (interrupts) en DMA-kanalen (*Direct Memory Access*) te weten komen. Soms kun je dat met jumpers op de kaart instellen. Bij andere kaarten moet je dat met een DOS-driver-programma doen, soms eenmalig, soms na elke reboot (dan moet je na het instellen van de geluidskaart Linux starten met loadlin.exe, zie paragraaf 5.5). En als het gaat om een Plug-and-Play kaart moet je die vanuit Linux met **isapnp** instellen (**pnpdump > /etc/isapnp.conf** om een 'template' te maken specifiek voor jouw systeem, dat editten = haakjes weghalen en eventueel waarden aanpassen, ook haakje weg voor de (ACT Y), dan **isapnp /etc/isapnp.conf**; bij volgende keren opstarten wordt **isapnp** automatisch gedraaid). Wanneer je alle informatie hebt, kun je met **modconf** de juiste module (proberen te) laden; waarden als io=0x220 irq=11 dma=1 moet je opgeven als daarom gevraagd wordt. Met iets als **nm /lib/modules/2.2.17/misc/opl3sa.o | grep 'D'** (pakketje 'binutils') kun je meestal een idee krijgen van de parameters die er zijn.

Als je zelf een kernel compileert, moet je de juiste opties wel aanzetten (als 'm'odule). In plaats van de standaard drivers in de kernel kun je ook de ALSA modules gebruiken (*Advanced Linux Sound Architecture*), meer daarover in paragraaf 9.6.2.

Als alles lijkt te zijn gelukt, kun je het geluid uitproberen met bijvoorbeeld **cat /bin/cat > /dev/audio** (speakers eerst zacht zetten!!), en met **cat /dev/sndstat** kun je zien wat er zoal beschikbaar is. Als de device-files nog niet bestaan, dan als root **cd /dev ; ./MAKEDEV audio**.

Je kunt de geluidsfuncties standaard alleen als root gebruiken; voeg andere gebruikers toe aan de groep 'audio' om ze ook daarvoor toestemming te geven (zie paragraaf 13.2).

11.2 Geluid beluisteren

Geluidsbestanden zijn er in soorten en maten, en elk formaat heeft z'n eigen afspeel-programma.

Mixer
Eerst iets over programma's waarmee je de mixer van de geluidskaart kunt instellen. Voor tekstmode en commandline-toepassingen is **aumix** populair, in X kun je bijvoorbeeld **tkmixer** gebruiken.

CD-speler
Er zijn talloze tools om de CDROM-drive aan te sturen bij het spelen van audio CD's. Voor commandline-toepassingen is 'cdtool' geschikt, commando's zijn bijvoorbeeld **cdplay 4**, **cdstop** en **cdeject**; bij sommige drives werken die echter niet goed. **workbone** is een aardige 'visuele' tool voor de console. In X kun je **gcd** gebruiken, of **xplaycd** uit het 'mctools-lite' pakketje (zit ook een **xmixer** in). Als je de CD's door gewone gebruikers wilt laten afspelen, is **chmod 666 /dev/hdc** (geef het juiste device voor de CDROM-drive) een goed idee.

'Wave' en 'sound'
Bestanden met extensies **.wav**, **.voc**, **.snd** en **.au** bevatten voor elk tijdstip de preciese positie van de luidspreker, ze zijn vergelijkbaar met bitmap-bestanden voor grafische toepassingen. Ook hier wordt het aantal tijdstippen beperkt tot een bepaalde hoeveelheid per seconde, van 8000 voor telefoon-kwaliteit tot 44100 (44.1 kHz, kilo-Hertz) voor CD-kwaliteit. Dit soort bestanden zijn af te spelen met **bplay**, en met **brec** uit hetzelfde pakketje kun je geluid opnemen;

opnemen met visuele volume-meter kan met **gramofile**. Met **sox** kun je formaten converteren, editten kan met **mxv** ('mixviews') of **snd** in X.

MP3
Het MPEG-1 (of -2) Layer 3 formaat is niets anders dan ingepakte 'wave'-data; het is bijzonder populair op het Internet omdat het bijzonder klein is en toch hoge kwaliteit oplevert. Voor afspelen in tekstmode is **mpg123** (sectie 'non-free') populair, maar er zijn ook **maplay** ('maplay3') en **splay**. **mp3blaster** is een 'visuele' tool voor de console. In X is wordt **xmms** (*X MultiMedia System*) veel gebruikt; dit is een 'Winamp' kloon met zeer uitgebreide mogelijkheden. Voor het omzetten van **.wav** bestanden in **.mp3** formaat is er 'lame' of 'notlame', maar die mogen vanwege copyright en patenten niet vrij gedistribueerd worden, en zitten daarom niet bij Debian; zie http://www.mp3dev.org en http://hive .me.gu.edu.au/not_lame voor meer informatie. Het 'rippen' van audio-CD's gaat gemakkelijk met **cdparanoia**. Met **id3**, **id3ed** of **mp3info** kun je de MP3-'tags' zien en/of wijzigen.

MIDI
In tegenstelling tot het **.wav**-formaat geven MIDI bestanden (**.mid**) alleen informatie over wanneer welke toon op welk instrument moet worden gespeeld; het is in feite een notenbalk waar de geluidskaart of computer muziek bij moet gaan maken. Met **playmidi** (of **xplaymidi**) kun je deze files af laten spelen door je geluidskaart, de **-e** switch stuurt het naar de externe MIDI-poort of soms de hardware-wavetable synthesizer. Met 'timidity' en 'timidity-patches' heb je een compleet software-wavetable systeem, dat beter is dan de meeste geluidskaarten. Topkwaliteit met **timidity -a -Es -s48000 -Od1lsS file.mid** (als je geluidskaart 48 kHz aankan); je kunt hiermee ook een **.wav**-file als uitvoer krijgen, die je dan weer met **bplay** kunt afspelen. **rosegarden** is een MIDI-editor voor X, en **vkeybd --device midi** is een 'virtueel keyboard'.

MOD
Een **.mod** bestand (ook diverse andere extensies) lijkt op een MIDI-bestand, maar bevat naast de notenbalk ook de geluidsfragmenten die voor de verschillende instrumenten gebruikt moeten worden. Afspelen op top-kwaliteit gaat met bijvoorbeeld **mikmod -f48000 -i -hq file.mod**, **s3mod -sbf 44100 file.mod** of **xmp -b 16 -f 48000 file.mod**. Een goede MOD-editor voor X is **soundtracker**.

Geluid in X
Enlightenment en Gnome hebben diverse mogelijkheden om geluid te koppelen aan muis-acties. Misschien staat dit nog niet aan en kan **dpkg-reconfigure esound-common** helpen.

11.3 Beeld

Video-bestanden zijn er in soorten en maten, en er komen steeds nieuwe varianten bij. Daarbij wordt echter veel gebruik gemaakt van ofwel geheime ofwel gepatenteerde coderingstechnieken, zodat afspelen in Linux vaak een probleem is. De 'non-free' **xanim** kan veel formaten aan (waaronder AVI, QuickTime en MPEG), maar sommige 'decoders' moeten apart worden opgehaald, wat automatisch kan gebeuren als je het 'xanim-modules' pakketje installeert. Voor **.mpg** en **.mov** formaten kunnen meestal ook **mpeg_play** (uit 'ucbmpeg-play') en **plaympeg** (uit 'smpeg-plaympeg') worden gebruikt. Het coderen van een reeks plaatjes naar een MPEG video kan met **mpeg_encode** uit 'ucbmpeg'.

12. Netwerken

Linux is gemaakt voor netwerken, en veel netwerk-toepassingen zijn beschikbaar. In dit hoofdstuk aandacht voor 'interne' en 'externe' netwerken, het koppelen van netwerken door routing en masquerading, en een groot assortiment programma's die netwerken gebruiken.

12.1 Aanleg van een Ethernet netwerk

Tegenwoordig kun je wel heel goedkoop NE2000-kloon netwerk-kaartjes krijgen (meestal een RealTek chipset), waardoor het aanleggen van een lokaal netwerkje aantrekkelijk wordt. Er zijn twee verschillende netwerk-structuren mogelijk. De eerste (en oudste) werkt met coax-kabels (RG58-type, 10Base2) en ronde BNC-connectors. Op elke netwerkkaart zit een T-stuk aangesloten met daaraan twee coax-kabels of één kabel en een 50-Ohm 'terminator'. Dit resulteert in één lange lijn met hier en daar een aftakking; de maximale snelheid is 10 Mbps (megabit per seconde), ofwel ongeveer 1 MB per seconde. De andere manier is met UTP-kabels (*Unshielded Twisted Pair*, 10BaseT, 100Base...) die 'vierkante' stekkertjes hebben (RJ45, 'western' stekkers). Van elke computer gaat dan één kabel naar een centrale *hub*, die de signalen van de zendende computer ofwel doorstuurt naar alle andere aansluitingen (*shared network*) of alleen naar de geadresseerde computer (*switched network*); zo'n hub is meestal vrij duur. Met deze ster-structuur zijn zeer grote snelheden haalbaar, 100 Mbps wordt vaak in nieuwe installaties gebruikt, maar ook 1 Gbps is beschikbaar. Meer informatie staat in de Ethernet-HOTWO en de NET3-4-HOWTO (zie ook paragraaf 4.21).

12.2 Een nieuwe netwerkkaart installeren

Als je een nieuwe netwerkkaart wilt gaan gebruiken, moet je eerst de driver daarvoor laden. Dit kan als root met **modconf**, kijk in de sectie 'net'. Voor PCI kaarten (die je met **lspci** kunt zien, pakketje 'pciutils') hoef je nooit opties op te geven, voor ISA moet dat wel, bijvoorbeeld io=0x300 irq=10. Als je meerdere kaarten hebt van hetzelfde type moeten ze op andere adressen ingesteld zijn; dat geef je op met bijvoorbeeld io=0x280,0x300. Op sommige kaarten moet je het adres en het IRQ-nummer met jumpers instellen; bij modernere 'jumperless' kaarten moet je éénmalig een DOS-programma draaien. Een enkele kaart die volledig Plug-and-Play is vergeet de instellingen als de computer uitgezet wordt; deze kun je met **isapnp** instellen, zie paragraaf 11.1.

De 'netwerk-gegevens' van de kaart(en) kunnen worden ingesteld in **/etc/network/interfaces**. Daar staat bijvoorbeeld in (inspringen mag met spaties of TAB's, hoeft niet):

```
iface lo inet loopback

iface eth0 inet static
        address 192.168.1.5
        netmask 255.255.255.0
        network 192.168.1.0
        broadcast 192.168.1.255
        gateway 192.168.1.1

iface eth1 inet dhcp
```

Met de loopback-interface kun je je eigen computer bereiken; dat is altijd adres 127.0.0.1 (eigenlijk 127.*x*.*y*.*z*, met *x*, *y* en *z* willekeurig). De eerste Ethernet-kaart is statisch geconfigureerd, de tweede krijgt de parameters via DHCP (*Dynamic Host Configuration Protocol*) met **pump**. Zie de **man interfaces**, **man ifconfig** en **man pump** voor meer informatie. Met **ifup eth1** wordt de tweede kaart aangezet, en met **ifdown -a** gaan ze allemaal uit.

12.3 Inbellen naar een ISP

Linux heeft uitgebreide mogelijkheden om via *Internet Service Providers* "het net op" te gaan. Veel analoge modems, ISDN-adapters en kabelmodems worden ondersteund.

12.3.1 Analoog modem met PPP

De standaard-manier voor internet-verbindingen over analoge modems is *PPP*, het *Point-to-Point Protocol*. In deze paragraaf behandel ik hoe het opzetten van zo'n verbinding in z'n werk gaat. Alle handelingen zijn uit veiligheids-overwegingen alleen door root uit te voeren, met uitzondering van het 'opdracht geven tot inbellen' zelf.

Er zijn analoge modems op de markt, die Linux niet kan gebruiken. Dat gaat om zogenaamde “WinModems” ofwel modems met *Host Signal Processing* (*HSP*). Dat zijn ‘uitgeklede’ (en daarom heel goedkope) modems, die eigenlijk niets meer zijn dan een geluidskaart. De processor van de PC moet al het zware werk doen. De manier van communiceren tussen deze modems en processor is echter geheim en voor elke fabrikant anders. Er worden Windows-drivers bijgeleverd, maar Linux is open-source “en dus taboe”. Als je zo’n modem hebt, kun je ’m beter weggeven aan een Windows-fan en zelf een echt modem kopen. Merk op dat *externe* modems geen WinModems kunnen zijn, en dus altijd bruikbaar zijn in Linux. USB-modems worden waarschijnlijk ook ondersteund, maar daar moet je een onstabiele development-kernel voor draaien. PCMCIA modems (‘credit card’ formaat) werken over het algemeen ook goed.

Interne modems (niet-WinModems) gedragen zich als derde seriële poort, en moeten meestal apart ingesteld worden. Tijdens de installatie van Debian werden maximaal vier seriële poorten automatisch gedetecteerd, maar niet altijd correct. Als je opnieuw wilt laten auto-detecten moet je “###AUTOSAVE-ONCE###” als eerste regel in **/etc/serial.conf** zetten, en dan **/etc/init.d/setserial stop** uitvoeren. De instellingen komen onderin **/etc/serial.conf** te staan, je kunt ze daar eventueel aanpassen; ze worden effectief na **/etc/init.d/setserial start** of na de volgende reboot. Als je meer dan vier seriële poorten hebt, moet je de device files met de hand aanmaken, bijvoorbeeld **cd /dev ; ./MAKEDEV ttyS5 ttyS6**.

Ik ga hier uit van de standaard-manier om PPP te configureren. Er schijnen een paar ISP’s te zijn, waarbij dat niet helemaal perfect werkt. In dat geval moet je het één en ander met de hand wijzigen, de betreffende bestanden staan onder aangegeven. Je kunt ook het pakketje ‘wvdial’ proberen, maar daar heb ik geen ervaring mee. Als je een (Telix-achtig) programma wilt om direct met het modem te ‘praten’ (AT-commando’s etc.) en ASCII-toegang te krijgen, probeer dan het pakketje ‘minicom’ (eerst als root **minicom -s** om te configureren).

Configuratie van een analoge PPP-verbinding

Bij Debian kunnen een aantal PPP-verbindingen tegelijk geconfigureerd zijn, die allemaal een eigen naam hebben. Bij het opzetten van de verbinding wordt dan met de juiste naam aangegeven welke ISP gebeld moet worden. Namen kunnen bestaan uit hoofdletters, kleine letters, cijfers en eventueel sommige leestekens (zoals ‘.’ of ‘_’). Eén naam is speciaal, namelijk ‘provider’. Die naam is voor de standaard-provider, die ‘snel’ gebeld kan worden.

Configureren van een PPP-verbinding gaat met het programma **pppconfig**; daarin moet je meestal de TAB-toets gebruiken om naar de ‘<Ok>’ knop te gaan. De uitleg in **pppconfig** is zeer uitgebreid, dus die zal ik hier niet herhalen, je moet steeds de gegevens invullen die je van je ISP hebt gekregen. De meeste ISP’s gebruiken CHAP voor authenticatie; als dat niet blijkt te werken moet je PAP proberen.

Wel iets over de modemsnelheid. De snelheid tussen de computer en het modem is namelijk meestal anders (groter) dan de snelheid tussen de twee modems (‘door de telefoonkabel’). Dat komt enerzijds omdat het dataverkeer over de telefoonkabel gecompresst kan zijn (en tussen computer en modem is dat nooit zo), anderzijds omdat de mogelijke snelheden over de telefoonkabel anders zijn dan tussen de computer en het modem (over telefoonkabel kan bijvoorbeeld 14400 bps, maar tussen computer en modem kan niets tussen 9600 en 19200 bps). De snelheid wordt weergegeven in *bps*, van bits per seconde (vaak ten onrechte *baud* genoemd). De snelheid in bytes per seconde is grofweg een tiende van de bps-waarde, maar dat kan ook meer zijn als je modem compressie toepast.

Een vuistregel is dat modems met (telefoonkabel-)snelheden tot 9600 bps niet kunnen compressen. Bij die langzame modems moet je de snelheid tussen computer en modem gelijk kiezen aan de telefoonkabel-snelheid. Bij snellere modems (12000 bps, 14400 bps enz.) kun je voor de snelheid tussen computer en modem het beste 115200 bps nemen, dan weet je zeker dat alles zo snel mogelijk gaat. Als je modem (of je seriële poort) geen 115200 bps aankan, probeer dan 38400 of 19200 bps. In **pppconfig** moet je alleen de snelheid tussen computer en modem opgeven (meestal dus 115200 bps); het modem onderhandelt zelf bij iedere verbinding met het andere modem wat de hoogst haalbare telefoonkabel-snelheid is.

Starten en stoppen van een verbinding

Als je een PPP-verbinding met **pppconfig** geconfigureerd hebt, kan die gestart worden met het commando **pon** (van ppp on). Gewoon **pon** zonder opties belt naar de ‘standaard’ ISP, die de verbinding-naam ‘provider’ heeft. Om een andere verbinding te starten geef je **pon <verbindingnaam>**.

De PPP-verbinding wordt verbroken met het commando **poff** (van ppp off), eventueel met **<verbindingnaam>** erachter.

Het is (zeker in het begin) een goed idee om tijdens het inbellen de logfile **/var/log/ppp.log** in de gaten te houden. Doe dit bijvoorbeeld met **tail -f /var/log/ppp.log** (afbreken met Ctrl-C).

Zoals al opgemerkt, is het niet nodig om root te worden voor het starten van de verbinding. Gebruikers die lid zijn van de groep ‘dip’ mogen ook **pon** en **poff** gebruiken (zie paragraaf 13.2 voor meer informatie over gebruikers en groepen.)

Wijzigen van verbindingsinstellingen

Je kunt met **pppconfig** de instellingen van een bestaande verbinding wijzigen, maar met de hand editten kan ook. Daarvoor moet je weten hoe één en ander werkt en welke bestanden gebruikt worden. Door **/usr/bin/pon** wordt **/usr/sbin/pppd** aangeroepen. **pppd** leest eerst de system-wide opties uit **/etc/ppp/options**. Dan worden verbinding-specifieke opties gelezen uit **/etc/ppp/peers/<verbindingnaam>**. In die file staat ook een doorverwijzing naar een ‘chat-script’ **/etc/chatscripts/<verbindingnaam>**, die het eigenlijke bel-proces (opdrachten aan het modem e.d.) regelt. Als **chat** merkt dat het andere modem heeft opgenomen, gaat de controle weer terug naar **pppd**, die inlogt (username in **peers/**, wachtwoord in **/etc/ppp/{chap,pap}-secrets**) en onderhandelt over enkele andere opties, waaronder het IP-adres van beide modems. Zodra de PPP-connectie actief is, wordt door **pppd** het script **/etc/ppp/ip-up** gestart, dat op zijn beurt weer alle scripts/programma’s in de directory **/etc/ppp/ip-up.d** uitvoert. Als je de connectie wilt verbreken, wordt door **/usr/bin/poff** een signaal naar de draaiende **pppd** gestuurd, die na het ‘ophangen’ **/etc/ppp/ip-down** start, die op zijn beurt weer alle scripts/programma’s in de directory **/etc/ppp/ip-down.d** uitvoert.

In **/etc/ppp/ip-up.d** kun je bijvoorbeeld een (uitvoerbaar!) script **do_mail** zetten waarin iets met **runq** en/of **fetchmail** wordt gedaan; of iets met masquerading, zie paragraaf 12.4.

Meer informatie

Er staat zeer uitgebreide informatie in de manual pages van **pppd** en **chat**. Kijk ook eens in de diverse bestanden in **/etc/ppp** en **/etc/chatscripts**.

12.3.2 Digitaal ISDN modem

Configuratie

ISDN gebruiken gaat aanzienlijk anders dan analoge modems. Na **apt-get install isdnutils** moet je eerst de driver voor je ISDN-kaart laden. Dat is altijd de ‘hisax’ module, met **modconf** in de ‘misc’ sectie; het probleem is het vinden van de juiste parameters. In ieder geval is dat protocol=1 voor Duitsland en protocol=2 voor de rest van Europa. Dan is het type van je kaart nodig, en eventueel een I/O-adres en IRQ-nummer. Je kunt dit opzoeken in de tabel in **/usr/share/doc/isdnutils/README.HiSax.gz**, bijvoorbeeld een Gazel ISA kaart heeft type=34 en daarbij moet ook een io=0x*XXX* en irq=*XX* worden opgegeven. Van PCI-kaarten kun je de naam zien met **lspci** (pakketje ‘pciutils’), en daar zijn nooit io- en irq-waarden voor nodig. ISA-kaarten moet je soms met jumpers instellen, met een DOS programma of **isapnp** (zie paragraaf 11.1); PCMCIA-kaarten (‘credit card’-formaat) kun je met de card services instellen en/of uitlezen.

Als de driver eenmaal geladen is, start je als root **isdnconfig**, waarmee standaard-configuratiefiles kunnen worden gemaakt. Kies ‘1’ (‘network interfaces’) en type ‘ippp0’ en Enter. Dan ‘2’ (‘synchronous ppp deamon’) en type weer ‘ippp0’ en Enter. Tenslotte ‘4’ (‘isdnlog configuration’) en type ‘isdnctrl0’ en Enter. In het hoofdmenu ‘q’ om te stoppen. Op dit moment staan in **/etc/isdn** een aantal bestanden die nog nader aangepast moeten worden. In **device.ippp0** eerst de ‘Warning!’-regel weghalen. Bij ‘LOCALMSN’ vul je één van je eigen nummers in, zonder het netnummer (een fout nummer is niet erg, dan komt het op de rekening van het ‘hoofdnummer’). Achter ‘REMOTE-MSN’ komt het nummer van je ISP, nu met het netnummer erbij (zonder streepjes of haakjes achter elkaar). Zet ‘LEADINGZERO’ op **''**, en ‘DIALMODE’ voorlopig op ‘manual’. De timeout kun je wat ophogen naar bijvoorbeeld ‘isdnctrl huptimeout ${device} 600000’. Het is ook een goed idee om ‘isdnctrl verbose 2’ aan te zetten. Ook in **ipppd.ippp0** eerst de ‘Warning!’-regel compleet verwijderen. Verder kun je de opties ‘debug’, ‘lock’, ‘noauth’, ‘defaultroute’ en ‘noproxyarp’ het beste aanzetten (hekjes weghalen); zet dan achter ‘name’ de inlognaam die je van je ISP hebt gekregen. In **isdn.conf** wordt ‘AREACODE’ je eigen netnummer, maar dan zonder ‘0’. Dan haal je in **isdnlog.isdnctrl0** alleen de ‘Warning!’-regel weg. Tenslotte moet je zowel in **/etc/ppp/pap-secrets** als **/etc/ppp/chap-secrets** je ISP-password aan je ISP-loginnaam koppelen door aan beide een regel “naam * password” toe te voegen. Als dat allemaal is gebeurd, zou **/etc/init.d/isdnutils start** zonder problemen moeten werken.

Merk op dat je ook hier verschillende verbindingen kunt configureren, dat worden dan bestanden met **ippp1**, **ippp2** enzovoort.

Starten en stoppen van een verbinding
Zodra je alles geconfigureerd hebt, kun je als root met **isdnctrl dial ippp0** de verbinding starten en met **isdnctrl hangup ippp0** stoppen. In logbestanden in **/var/log** kun je precies zien wat er gebeurt, zelfs informatie over andere in- of uitgaande gesprekken.

Als je de ‘manual’ aanduiding (weer) vervangt door ‘auto’, zal automatisch verbinding worden gemaakt zodra pakketjes met een ‘buiten-bestemming’ gesignaleerd worden. Het is verstandig om de ‘timeout’ dan weer wat lager te zetten, zodat ook automatisch kan worden opgehangen als er geen netwerkverkeer meer is. Let er wel op dat programma’s als Netscape soms/vaak schijnbaar onverklaarbaar de verbinding kunnen openen (of open houden), bijvoorbeeld als een webpagina code bevat die zichzelf na een tiental minuten opnieuw gaat laden.

12.3.3 Kabelmodem en ADSL/xDSL

Praktisch alle kabelmodems en xDSL-modems werken ofwel als analoog modem (intern of extern) danwel als/met een gewone netwerkkaart. Beide gevallen moet je behandelen als eerder beschreven.

12.4 Routeren en masqueraden

Als je een lokaal netwerk hebt, kunnen alle computers tegelijk gebruik maken van het modem aan een Linux-computer. Linux heeft namelijk de mogelijkheid om pakketjes van het netwerk door te sluizen naar het modem, en eventueel kan ook nog gedaan worden alsof ze van de computer-met-modem zelf vandaan komen. Met die *masquerading* lijkt het voor de ISP of er maar één computer inbelt, terwijl in feite het hele netwerk met Internet bezig kan zijn. Gebruik gewone routering (*forwarding*) als je van je ISP een aantal (statische!) IP-adressen hebt gekregen; gebruik masquerading (eigenlijk forwarding-met-iets-extra’s) als je van je ISP slechts één (statisch of dynamisch) IP-adres hebt gekregen; gebruik *geen* forwarding of masquerading als je maar één computer hebt (sla in dat geval de rest van deze paragraaf over).

Instellen van forwarding en/of masquerading gaat met het commando **ipchains** (“IP-ketting”). Daarmee kan zeer precies worden aangegeven wat waarheen mag worden gerouteerd. Je hoeft de interface (netwerkkaart, ppp-verbinding, etc.) echter nooit op te geven, omdat de kernel dat al weet en zelf kan uitzoeken. De ‘routeringsregels’ worden bijgehouden in een aantal tabellen, die aan elkaar gekoppeld worden met sprong-opdrachten. Pakketjes die we zelf sturen komen binnen op de ‘output’-tabel, en worden standaard doorgelaten. Pakketjes die specifiek aan ons gericht zijn komen binnen op de ‘input’-tabel, en worden ook standaard doorgelaten. Pakketjes die niet voor ons zijn bedoeld, maar wel ons als ‘gateway’ (‘router’) gebruiken komen binnen op de ‘forward’-tabel, en daar kunnen we interessante dingen mee gaan doen.

De door **ipchains** gebruikte IP-adressen worden in het algemeen gespecificeerd als reeks met een netmask, in de vorm <adres>/<netmask> [poort(-reeks)]. Bijvoorbeeld: ‘192.168.1.0/255.255.255.0’ betekent ‘alle adressen van 192.168.1.0 t/m 192.168.1.255’ en ‘192.168.12.34/255.255.0.0 20:25’ betekent ‘alle adressen van 192.168.0.0 t/m 192.168.255.255, maar alleen de poorten 20 t/m 25’. Als je geen netmask opgeeft (of 255.255.255.255) wordt één bepaalde computer aangewezen. Als je twee of meer niet-aansluitende poorten (of reeksen) wilt specificeren, moet je twee aparte regels gebruiken.

Standaard staat forwarden en masqueraden uit, zet het eerst aan door in **/etc/network/options** “ip_forward=yes” te zetten en **/etc/init.d/networking start** uit te voeren (gebeurt automatisch bij opnieuw opstarten).

Hierna zullen alle pakketjes onveranderd doorgestuurd worden, maar dat is meestal (zeker bij masquerading) niet de bedoeling. Je zet de ‘standaard-regel’ voor de ‘forward’-tabel op weigeren (‘deny’) met **ipchains -P forward DENY** (‘**-P**’ van standaard-policy).

Onveranderd forwarden gaat dan met **ipchains -b -A forward -s <lokaalnet>/<netmask> -d <verweg-net>/<netmask> -j ACCEPT**, hiermee wordt een bidirectionele regel toegevoegd (append) aan de ‘forward’-tabel, die pakketjes met het gegeven source en destination adres (en vice versa door de **-b**) doorstuurt (jump) naar de speciale ‘ACCEPT’-tabel, die altijd alles accepteert wat erop terecht komt. Voorbeeld voor routering tussen twee subnets binnen een bedrijf, waarbij de router twee netwerkkaarten zal hebben:


```
# routeer van 192.168.1.* naar 192.168.2.* en terug:
ipchains -b -A forward -s 192.168.1.0/255.255.255.0 -d 192.168.2.0/255.255.255.0 -j
ACCEPT
```

Masqueraden is niet bidirectioneel (terugrouteren gaat automatisch), en moet naar de ‘MASQ’-tabel in plaats van ‘ACCEPT’. Voorbeeld voor masquerading bij inbellen naar een ISP:

```
# masquerade alles van 192.168.1.* naar *.*.*.* (en terug):
ipchains -A forward -s 192.168.1.0/255.255.255.0 -d 0.0.0.0/0.0.0.0 -j MASQ
```

Je kunt alle regels uit een tabel weggooien (bijvoorbeeld als de PPP-connectie is opgehangen) met **ipchains -F forward**. Alle tabellen bekijken kan met **ipchains -L -n**.

Als je masquerading wilt automatiseren, maak dan een script **/etc/ppp/ip-up.d/masq_on** met daarin iets als:

```
#! /bin/sh
# gooi voor de zekerheid alle oude regels weg
ipchains -F forward
# standaard niets doorlaten
ipchains -P forward DENY
# masquerade alles van 192.168.1.* naar *.*.*.* (en terug):
ipchains -A forward -s 192.168.1.0/255.255.255.0 -d 0.0.0.0/0.0.0.0 -j MASQ
# verhoog tcp-timeout zodat langzame (FTP-)verbindingen en ongebruikte
# remote shells niet onderbroken worden
ipchains -M -S 3600 0 0
```

en een script **/etc/ppp/ip-down.d/masq_off** met iets als:

```
#! /bin/sh
# gooi alle regels weg
ipchains -F forward
```

Als het niet werkt: heb je de scripts met **chmod** uitvoerbaar gemaakt? En hebben alle computers de computer-met-modem als router=gateway staan (in **/etc/network/interfaces** of equivalent)?

Met **ipchains** kun je talloze andere dingen doen, zie daarvoor de manpage en de IPCHAINS-HOWTO.

12.5 Netwerkprogramma’s

Als je in een netwerk zit (ook bij inbellen naar een ISP), kun je verbinding maken met andere computers. Voor de vele mogelijkheden zijn diverse programma’s beschikbaar. Dat zijn steeds *client* programma’s, en de ‘ver-weg-computer’ is de *server*, die server-programma’s (ook *daemons* genoemd) draait. De client is een programma dat je zelf (als gewone gebruike) start als je dat nodig hebt; de server wordt tijdens het installeren van het server-pakketje gelijk aangezet, zodat de service beschikbaar is als iemand ’m wil gebruiken. Ik zal hier de meest populaire zaken bespreken.

Ook als je niet ‘aan een netwerk hangt’ kun je de meeste programma’s uitproberen, door zowel server als client op dezelfde computer te installeren en voor de servernaam steeds ‘localhost’ in te vullen.

12.5.1 Werken op afstand

Telnet en rlogin
Een veel-gebruikte soort verbinding is de *remote login*: je logt in op een computer (server) terwijl je achter een heel andere computer zit. Ofwel, je krijgt een virtuele terminal op de server (zie ook paragraaf 2.5.4). Eén manier om zo’n verbinding te openen is met **telnet <servernaam>** (client: ‘telnet’, server: ‘telnetd’), bijvoorbeeld **telnet panic .et.tudelft.nl**, of **telnet localhost**. Een andere (oudere) manier is met **rlogin <servernaam>** (client: ‘rsh-client’, server: ‘rsh-server’). Daarmee wordt ervan uitgegaan dat je op de server dezelfde username hebt als op de client waar je ‘achter zit’, en er wordt dus meteen om je password gevraagd. Is dat verkeerd, dan mag je alsnog een andere username opgeven. Zie de manpages van **telnet** en **rlogin** voor meer info.

Als je op een andere server remote bent ingelogd, kan het gebeuren dat programma's klagen dat ze problemen hebben met de terminal of de terminal-definitie. Voor praktisch elk type terminal is namelijk een lijst beschikbaar met dingen die ze wel en niet kunnen en bijvoorbeeld wat voor toetsen erop zitten. Schermvullende programma's als **less** en **mc** gebruiken die informatie om alles netjes op het scherm te krijgen en om bijvoorbeeld functietoetsen goed te verwerken. Omdat een terminal niet zelf zegt welk type hij is, moet de gebruiker dat opgeven en wel in de TERM environment variabele. Nu heeft in Linux de console het type 'linux' en een xterm 'xterm', en een standaard UNIX machine kent zeker 'linux' vaak niet. De oplossing is om dan een ander terminal-type te kiezen, meestal werkt 'vt100' of 'vt220' goed. In een **bash**-achtige remote shell doe je dat met bijvoorbeeld **TERM=vt100; export TERM** (of verkort **export TERM=vt100**), in een **csh**-achtige remote shell met **setenv TERM vt100**; zie ook paragraaf 7.6.

SSH

In het algemeen worden **telnet** en **rlogin** als bijzonder onveilig beschouwd, onder andere omdat de tekst die je typt en op het scherm ziet, letterlijk over het netwerk gaat. Iedereen kan daarbij meeluisteren, en moeiteloos je password of andere persoonlijke gegevens te weten komen. Zelfs het draaien van de telnet en rlogin *servers* wordt meestal ten zeerste afgeraden (tip: **apt-get remove telnetd rsh-server**).

De oplossing is **ssh**, de *Secure Shell*, die alle gegevens (inclusief passwords) versleuteld (ge-encrypt) verzendt. Er zijn hier twee varianten van, 'ssh' is de vrije maar nog wat onstabiele versie, 'ssh-nonfree' is niet-vrij maar stabieler. In beide pakketjes zit zowel de server als de client. Verbinding maken gaat met iets als **ssh -C -v <servernaam>**, dan krijg je gelijk ook compressie erbij. Met de 'vrije' client kun je ook ssh'en naar een 'niet-vrije' server en vice versa.

Een bijkomend voordeel is dat met **ssh** ook (onder andere) remote X-vensters kunnen worden ge-encrypt. Op de server wordt de **DISPLAY** environment variabele automatisch op bijvoorbeeld **<servernaam>:10** gezet; daar hangt dan geen echte X-server aan, maar de ssh, die alles naar jouw originele X-server door-'tunnelt'. Als dit niet meteen werkt, moet je **ssh** starten met een extra **-X** optie.

12.5.2 Bestandsoverdracht

FTP

Om bestanden van de éne naar de andere computer te sturen zijn verschillende mogelijkheden, bijvoorbeeld FTP, het *File Transfer Protocol* (client bijvoorbeeld 'ftp', server: 'wu-ftpd' of 'proftpd'). Met FTP kun je inloggen als jezelf, met password, dan kun je de hele harde schijf van de server benaderen. Meer gebruikelijk is *anonymous FTP* ('anoniem'), waarbij je inlogt als 'anonymous' of 'ftp', en meestal een dummy e-mail-adres als wachtwoord, dan heb je alleen toegang tot een bepaalde directory (meestal **/home/ftp/pub** of **/var/ftp/pub**). De installatie-scripts van de FTP-servers vragen meestal of anonieme toegang aangezet moet worden.

Voorbeeld van gebruik: ga eerst op de client in de directory staan waar je bestanden uit wilt verzenden of in wilt zetten. Geef dan het commando **ftp <servernaam>**. Na het invoeren van je username en password kun je commando's geven. Met **cd** verander je van directory op de server(!), met **dir** of **ls -alF** kun je de inhoud van de remote directory opvragen. Bestanden van de client naar de server kopiëren (*uploaden*) gaat met **put <lokaalbestand>** of **mput <lokalebestanden>** (met 'm' van multiple). Bestanden van de server naar de client kopiëren (*downloaden*) gaat met **get <remotebestand>** of **mget <remotebestanden>**. Bij de 'm'-versies mogen ook wildcards ('**?**' en '*****') in de bestandsnamen gebruikt worden. Als je bij de 'm'-versies niet steeds wilt bevestigen, geef dan vooraf het ftp-commando **prompt**. Als je wilt zien hoe snel de overdracht gaat, geef dan vooraf **hash**. Bestanden kunnen binair (één-op-één) of in tekstmode overgestuurd worden, kies vooraf de juiste mode met **binary** of **ascii**. Let op: in ascii-mode worden Enters en Linefeeds veranderd, en dat is meestal *niet* de bedoeling -- gebruik dus altijd binaire mode. Verlaat FTP met **bye**, **quit** of **exit**. Zie verder de manpage van **ftp**.

Trouwens, de alom geprezen **mc** kan ook heel goed FTP-en. Probeer eens F9 - 'Right' - 'FTP Link...' en geef als machine name <username>@<servernaam> en vervolgens je password.

SCP

Niet-anonieme FTP heeft hetzelfde probleem als **telnet** en **rlogin**: je wachtwoord gaat zichtbaar over het netwerk. Ook daarvoor heeft de **ssh**-'suite' een makkelijke oplossing: het commando **scp**, *Secure Copy*. Met **scp -C -v <servernaam>:file /tmp** kopieer je het bestand **file** uit je home directory op de server naar **/tmp** op de client. En dat kan ook andersom.

HTTP

'Surfen' over het 'web' (browsen van Internet-pagina's) is ook niets anders dan bestands-overdracht van tekst-pagina's, plaatjes en dergelijke. Op Linux systemen wordt praktisch alleen Netscape als client gebruikt, pakketje 'communicator' voor veel functionaliteit of 'navigator' voor minder (en minder schijfruimte en geheugengebruik), of 'communicator-smotif-473' danwel 'navigator-smotif-473' als de korte versies niet beschikbaar zijn. Recenter is **mozilla**, die op termijn de nieuwe **netscape** moet gaan worden maar nu nog vrij onstabiel is. In tekstmode was **lynx** lang de enige optie, maar die heeft nu concurrentie van **w3m** en **links**, die beide meer features hebben (zie de manpages voor de functies van de toetsen!).

Er zijn vele web-servers beschikbaar, maar 'apache' is veruit de populairste. Na installatie kun je met bijvoorbeeld 'http://kennedy.dorpsnet.nl' de tijdelijke 'homepage' van de server zien; je kunt hem aanpassen in **/var/www**. Ook iedere gebruiker kan een 'homepage' opzetten, in de subdirectory **public_html** in z'n home directory; met 'http://kennedy.dorpsnet.nl/~ikke/iets.html' wordt het bestand **/home/ikke/public_html/iets.html** opgehaald. Als geen specifieke bestandsnaam wordt opgegeven, wordt een directory-listing getoond, of het bestand **index.html** als dat in de betreffende directory bestaat. Als je je eigen pagina's niet kunt zien: zijn ze wel leesbaar voor iedereen (**chmod a+r**)?

Voor modemgebruikers en anderen die niet van reclame houden is de *Internet JunkBuster* ('junkbuster') bijzonder geschikt. Stel je browser in op proxy: localhost, poort: 5865 (**export http_proxy='http://localhost:5865/'** voor browsers die daar zelf geen invul-veld voor hebben). Met

```
ad.*.com
ads.*.com
adserver.*.com
/*.*/(ad|ads|adverts)/
/*.*/banner*
```

in **/etc/junkbuster/blockfile** filter je al heel wat troep weg; je kunt natuurlijk ook zelf dingen toevoegen.

File-sharing

Zeer recent zijn diverse 'file-sharing' programma's ontwikkeld, die in korte tijd enorm populair zijn geworden. Vaak zijn daarvoor Linux clients/servers beschikbaar, maar die zijn vaak zo onstabiel dat ze nog niet bij Debian zitten; kijk op de betreffende web-pagina's voor meer informatie. Bij Debian zitten wel enkele Napster-klonen voor X, **gnap**, **gnapster** en **gnome-napster**.

12.5.3 Contact met anderen

Ping, traceroute

Met **ping <servernaam>** ('netbase') kun je zien of een computer bereikbaar is; **fping** is geschikter voor scripts. Met **/usr/sbin/traceroute** of **mtr** (**--curses**) kun je zien welke route pakketjes afleggen.

Finger

Om te kijken of een specifiek persoon op een bepaalde computer is ingelogd kun je **finger <username>@<servernaam>** gebruiken (client: 'finger', server: bijvoorbeeld 'cfingerd'). Een lijst vragen van iedereen die op een computer is ingelogd gaat met **finger @<servernaam>**. Voorbeeld: **finger ikke@kennedy.dorpsnet.nl**. Fingeren kan trouwens ook lokaal, bijvoorbeeld **finger andere_user**. **finger** zonder opties geeft uitvoer die lijkt op die van **w** of **who**.

Talk

Als je iemand heel snel wilt spreken, kun je hem of haar ook 'opbellen'. Geef daarvoor het commando **talk <username>@<servernaam>** (client: 'talk', server: 'talkd'), '@servernaam' mag je weglaten bij een 'lokaal gesprek'. De persoon die jij oproept, krijgt dan een bericht op z'n scherm dat er een talk-request van jou is. Die ander geeft dan ook een commando **talk <jouwnaam>@<jouwserver>** en de verbinding is gemaakt. Je krijgt allebei een 'gesplitst' scherm: bovenin komt de tekst die je zelf typt en onderin de tekst die de ander typt. Op die manier kun je een 'geprek' voeren. Je kunt 'de hoorn weer op de haak leggen' met Ctrl-C. Zoals gebruikelijk: zie verder de manpage van **talk**.

Merk op, dat je ook (tijdelijk of permanent) onbereikbaar kunt zijn voor berichten, door de opdracht **mesg n** te geven (messages: no). Je kunt weer bereikbaar worden met **mesg y**. Het kan zijn, dat je standaard (bij het inloggen) onbereikbaar bent. Vergeet dan niet **mesg y** te geven voor je **talk** gaat uitproberen. Je kunt **mesg y** of **n** natuurlijk ook opnemen in je opstartbestand(en), zie paragraaf 7.9. Trouwens, 'uitgaande' gesprekken zijn altijd mogelijk, onafhankelijk van de **mesg**-status. Met alleen **mesg** kun je de actuele status opvragen. Let op: als je ge-**su**-d bent geweest naar root, kan de **mesg**-status op n gezet zijn door root's **.profile**.

Write en wall

Naast **talk**, dat ook voor 'remote' computers werkt, is er nog een manier om gebruikers op het 'lokale' systeem snel te bereiken. Dat gaat met een opdracht als **write andereuser**. Dan krijgt de andere gebruiker een melding op z'n scherm dat jij hem iets wilt vertellen, en mag jij een bericht intypen. Elke keer dat jij een regel afsluit met Enter zal die regel in z'n geheel op 'het andere scherm' verschijnen. Als die ander ook **write jouwnaam** start, kun je op een primitieve manier talken. Je sluit **write** af met Ctrl-D (het algemene 'einde bericht'-teken), of eventueel (als je dat niet meer weet) met Ctrl-C.

Als je een bericht hebt, dat iedereen zou moeten interesseren, kun je alle lokale gebruikers in één keer bereiken met het commando **wall**. Dat werkt hetzelfde als **write**, maar je hoeft nu natuurlijk geen gebruikersnaam op te geven. En het hele bericht wordt in één keer verstuurd als jij Ctrl-D (geen -C) hebt gegeven, dus niet regel-voor-regel zoals met **write**. Wees wel erg voorzichtig met **wall**, er zijn gebruikers die heel snel geïrriteerd raken als ze ongevraagd onnodige berichten over hun scherm krijgen.

Een manier om **write** of **wall** in een script te gebruiken (maar ook vaak op de commandline gebezigd), is met redirection (zie paragraaf 4.17), zoals **cat /tmp/bericht | write iemand** of **echo 'Hallo allemaal!' | wall**.

Je kunt **write** en **wall** berichten onderdrukken met **mesg n**, zie hierboven. Merk echter op dat root altijd toestemming heeft om iets bij iemand (of iedereen) op het scherm te zetten.

IRC, ICQ, Instant Messenger

Met veel personen tegelijk praten op een centrale server kan met IRC (*Internet Relay Chat*). Er zijn zeer uiteenlopende clients voor Linux, in tekstmode bijvoorbeeld **bitchx <nickname> <server>**, of **xchat** in X. IRC-server programma's zijn er ook, bij Debian zitten 'ircd' en 'ircd-dalnet'. Met ICQ ("I Seek You") kun je je vriendenkring bereiken, Linux-klonen voor tekstmode zijn onder andere **micq** en **zicq**, voor X is er **licq**. Met **everybuddy** kun je naast ICQ ook gebruik maken van sommige Instant Messenger services, **tik** is alleen voor de AOL IM (AIM).

E-mail

Ook voor het lezen en schrijven van E-mail heeft Linux vele mogelijkheden. Persoonlijk vind ik nog steeds **pine** het meest gebruiksvriendelijk, vooral door de uitgebreide hulpteksten en het toetsenoverzicht onderin het scherm. Het zit echter in de 'non-free' sectie en je moet het zelf even compileren, zie paragraaf 9.4. In tekstmode zijn er bijvoorbeeld ook **mutt** en **elm** ('elm-me+'), vooral de toets-functies zijn hierbij even wennen, en beide gebruiken een externe editor die je met de EDITOR environment variabele zelf kunt kiezen (zie paragraaf 10.2). In X is de mail-functie van de Netscape Communicator erg populair (zie ook in paragraaf 12.5.2), daarnaast zijn er ook **spruce**, die mijn Inbox als één enorm bericht ziet, en **tkmail**, die ook nog 'tk8.0' nodig heeft.

Voor de daadwerkelijke afhandeling van de E-mail is een *mail transport agent* nodig. Waarschijnlijk is 'exim' al geïnstalleerd, maar als je speciale wensen hebt kun je bijvoorbeeld ook de "oude (on)vertrouwde" 'sendmail' gebruiken, of 'postfix', of 'zmailer'.

USENET Nieuwsgroepen

pine en de Netscape Communicator kunnen ook worden gebruikt voor het volgen van en deelnemen aan de discussies in de 'nieuwsgroepen' (*USENET newsgroups*). In tekstmode worden ook vaak **nn** en **slrn** gebruikt, of de 'non-free' **tin** of **trn** (pakketje 'trn' of 'strn'). Voor X is er ook nog **knews**. De meeste news-programma's kunnen zowel een 'remote' server benaderen met NNTP (*News Network Transport Protocol*), maar ook een lokaal nieuws-archief direct op de harddisk; kiezen tussen die twee gebeurt vaak met commandline-opties.

Bij Debian zitten diverse news-server-pakketjes; 'inn2' heeft de voorkeur, maar je kunt ook bijvoorbeeld 'leafnode' of 'cnews' gebruiken als je speciale behoeften hebt.

12.5.4 Andere protocollen: SMB, IPX, AppleTalk

Linux is heel goed in het communiceren met allerlei soorten computers. Het Windows netwerk (SMB, 'LanManager'), Novell Netware (IPX) en AppleTalk worden uitstekend ondersteund. Je moet wel een kernel hebben die dat aankan, en ook een aantal client en server programma's. Kijk eens naar de pakketjes 'samba', 'smbclient', 'smbfs', 'samba-doc', 'smb-nat', 'ipx', 'ipxripd', 'ncpfs', 'netatalk' en 'macgate'. Voor Netware-server emulatie kun je ook op Internet zoeken naar 'mars_nwe' (niet als Debian pakketje beschikbaar).

12.6 Nameservice: /etc/hosts en BIND

12.6.1 Namen vertalen naar nummers

Op een netwerk zijn alleen IP-adressen bekend, en geen namen. Als je de opdracht **ssh -C -v host.somewhere.nl** geeft, moet de hostname eerst worden 'vertaald' in een IP-adres, voordat de verbinding kan worden opgebouwd. Voor dat vertalen zijn twee mechanismen beschikbaar: een (simpel) tekstbestand en een (ingewikkeldere) nameserver.

Als eerste wordt altijd dat tekstbestand, **/etc/hosts**, geraadpleegd. In dat bestand staan een aantal regels die een adres verbinden met één of meer namen. Bijvoorbeeld:

```
192.168.1.1      kennedy.dorpsnet.nl        kennedy
192.168.1.2      rocky.dorpsnet.nl          rocky
192.168.1.3      mountains.dorpsnet.nl      mountains
123.45.67.89     andere.host.ergens.com     andere
```

Met deze **/etc/hosts** hebben bijvoorbeeld **ssh -C -v 123.45.67.89**, **ssh -C -v andere.host.ergens.com** en **ssh -C -v andere** allemaal hetzelfde effect.

Als de naam niet in **/etc/hosts** te vinden is, wordt het tweede mechanisme in werking gesteld: de nameserver. In het bestand **/etc/resolv.conf** staan (meestal) een aantal 'nameserver'-regels. Met die nameservers wordt een netwerk-verbinding gemaakt en gevraagd welk IP-adres er bij de betreffende naam hoort. Als een nameserver niet bereikbaar is, of te lang op zich laat wachten, wordt de volgende nameserver geprobeerd.

Als geen enkele nameserver uitkomst biedt, heb je waarschijnlijk een typefout gemaakt in de hostname, en wordt dat gemeld.

Als je veel computers op een netwerk hebt, wordt het een hele klus om steeds bij elke computer de **/etc/hosts** aan te passen. Vandaar dat er dan meestal gekozen wordt voor één centrale nameserver (en vaak nog één extra als 'backup').

12.6.2 De werking van een nameserver

Een nameserver wordt meestal gebruikt om een IP-adres te zoeken bij een gegeven computernaam. Maar eventueel kan het ook omgekeerd. En er kan worden uitgezocht of e-mail naar een bepaalde computer zelf moet, of naar een centrale mail-server. Alle vragen en antwoorden lopen via netwerk-verbindingen.

In principe weet een nameserver alleen alles over z'n eigen domein, daarom wordt hij officieel een *Domain Name Server* (of *DNS*) genoemd. Als er vragen komen over computers in een ander domein, zal de nameserver die vraag doorspelen aan een andere nameserver, die daar meer vanaf zou moeten weten. Dat 'doorvragen' kan op twee manieren gebeuren.

De 'standaard'-manier om bijvoorbeeld het adres van www.tudelft.nl te weten te komen, is om aan een 'top-level' nameserver te vragen om het adres van de nameserver van het domein 'nl'. Dan wordt aan de nameserver van 'nl' gevraagd om het adres van de nameserver van het domein 'tudelft.nl'. En tenslotte wordt aan de nameserver van 'tudelft.nl' gevraagd om het adres van 'www.tudelft.nl'.

Die reeks aanvragen kan soms vrij lang gaan duren en is ook behoorlijk netwerk-intensief. Daarom is er een tweede manier ontwikkeld, de zogenaamde *recursive query* (ook wel *forwarden* genoemd). Dan wordt aan een andere nameserver precies dezelfde vraag gesteld als aan de 'eerste' nameserver en moet die 'tweede' nameserver zelf maar zien hoe hij aan het antwoord komt (dat kan dan weer op beide manieren). Zo kan bijvoorbeeld een inbel-verbinding behoorlijk worden ontlast en zal het opzoeken van IP-adressen meestal ook nog sneller verlopen.

Een ander versnellings-mechanisme is dat de nameserver een heleboel informatie kan 'onthouden' in een *cache*. Als voor de tweede keer het IP-adres wordt gevraagd van een bepaalde computer, zal die informatie meestal nog in de cache staan en kan de nameserver gelijk (zonder doorvragen) het antwoord geven.

12.6.3 Opzetten van een nameserver

De standaard UNIX-nameserver heet het *Berkeley Internet Name Domain*, afgekort *BIND*. Het echte nameserver-programma is **named** (van name daemon). Om een nameserver op een Debian systeem te zetten: **apt-get install bind bind-doc dnsutils**. Bij de configuratie van die pakketjes wordt waarschijnlijk gevraagd of 127.0.0.1 aan **/etc/resolv.conf** moet worden toegevoegd, geef daar 'y'.

De nieuwe nameserver is nu geconfigureerd als *caching-only* nameserver -- hij weet zelf niets, maar kan wel voor een aardige tijdsbesparing zorgen. Tenminste, als je aan alle lokale computers doorgeeft dat die nameserver actief is. Dat doe je door een regel

```
nameserver     192.168.1.1
```

of iets dergelijks (waarschijnlijk IP-nummer aanpassen) toe te voegen aan de **/etc/resolv.conf** van alle computers. Ook bij Windows-computers kun je dit doen, via de gebruikelijke netwerk-configuratie.

De nameserver gebruikt momenteel alleen de eerste ('standaard') methode om namen op te zoeken, forwarding kan worden aangezet door in **/etc/bind/named.conf**, in de sectie 'options', iets neer te zetten als

```
        forwarders {
                130.161.180.1;
                130.161.180.65;
        };
```

en vul de nameservers in waar je naar wilt laten forwarden, de juiste adressen heb je van je ISP gehoord (deze zijn van de TU Delft, en kun je gebruiken als je niets beters weet). Na het veranderen van de instellingen altijd **/etc/init.d/bind reload** draaien.

Het toevoegen van informatie over een domein is vrij eenvoudig. Het volgende voorbeeld gaat uit van de **/etc/hosts** file die boven is afgedrukt, met (alleen) kennedy.dorpsnet.nl als nameserver; je kunt het eenvoudig aanpassen aan jouw situatie.

Maak een bestand **/etc/bind/db.dorpsnet** met de volgende inhoud (alle witruimtes mogen willekeurig lang zijn):

```
; BIND data file for dorpsnet.nl
; (commentaar na een ';')

$TTL    86400
;              Deze nameserver      Nameserver-beheerder
;                 Let op de punt--v     v--'.' i.p.v. '@'
@   IN   SOA   kennedy.dorpsnet.nl. root.kennedy.dorpsnet.nl. (
               1998092601   ; volgnummer - datum+revisie
;                             ophogen bij elke verandering!
                    10800   ; Refresh        )
                     3600   ; Retry          )  standaardwaarden
                  3600000   ; Expire         )   (niet veranderen)
                    86400 ) ; Default TTL    )

; Nameservers voor dit domein
               NS    kennedy
; Eventueel meer NS-jes

; Verklarende tekst (hoeft niet)
               TXT   "This is DORPSNET.NL!"
```



```
; Alle computers in dit domein.
; 'A'-record geeft IP-adres
; 'MX'-record geeft naam van computer waar mail heen moet
kennedy        A     192.168.1.1
               MX    10 kennedy

rocky          A     192.168.1.2
               MX    10 rocky

mountains      A     192.168.1.3
               MX    10 mountains

; kennedy is ook webserver (www.dorpsnet.nl)
www            A     192.168.1.1
               MX    10 kennedy

; Algemene mail-route; mail naar iemand@dorpsnet.nl wordt
; eerst bij kennedy geprobeerd, dan bij rocky en tenslotte
; bij mountains. De nummers 10, 20 en 30 zijn de prioriteiten.
; Je kunt nu ook surfen naar dorpsnet.nl (zonder www ervoor).
;         v--Let op de punt na dorpsnet.nl!
dorpsnet.nl.   A     192.168.1.1
               MX    10 kennedy
               MX    20 rocky
               MX    30 mountains
```

Dan een bestand **/etc/bind/db.dorpsnet-rev** met de volgende inhoud:

```
; BIND data file for dorpsnet.nl (reverse lookup)

$TTL    86400
;              Deze nameserver      Nameserver-beheerder
;               Weer punt achteraan en '.' i.p.v. '@'
@   IN   SOA   kennedy.dorpsnet.nl. root.kennedy.dorpsnet.nl. (
               1998092601   ; volgnummer - datum+revisie
;                             ophogen bij elke verandering!
                    10800   ; Refresh        )
                     3600   ; Retry          )  standaardwaarden
                  3600000   ; Expire         )   (niet veranderen)
                    86400 ) ; Default TTL    )

; Nameservers voor dit domein, volledige namen
; Let op de punt achteraan!
               NS    kennedy.dorpsnet.nl.
; Eventueel meer NS-jes

; Verklarende tekst (hoeft niet)
               TXT   "This is DORPSNET.NL!"

; Alle computers met IP-adressen 192.168.1.*
; Het eerste getal moet je achter '192.168.1.' plakken.
; Let op de punten achter de volledige naam
1              PTR   kennedy.dorpsnet.nl.
2              PTR   rocky.dorpsnet.nl.
3              PTR   mountains.dorpsnet.nl.
```

Tenslotte moet je helemaal onderaan **/etc/bind/named.conf** de volgende twee secties toevoegen (er staan al diverse soortgelijke secties), let op de puntkomma's en andere leestekens:

```
// Dorpsnet naam -> adres
zone "dorpsnet.nl" {
        type master;
        file "/etc/bind/db.dorpsnet";
};
```

```
// Dorpsnet adres -> naam (let op omgekeerde 192.168.1)
zone "1.168.192.in-addr.arpa" {
        type master;
        file "/etc/bind/db.dorpsnet-rev";
};
```

Als je alle bestanden naar je zin veranderd hebt, moet je dat weer aan **named** doorgeven met **/etc/init.d/bind reload**.

Met **nslookup rocky.dorpsnet.nl** en **nslookup 192.168.1.3** kun je nu kijken of heen- en terug-zoeken goed werkt (Ctrl-C als het te lang duurt = fout gaat). Mochten er problemen zijn, dan staat in **/var/log/daemon.log** meestal een foutmelding met een indicatie van de oorzaak.

Je kunt alle vragen aan de nameserver laten loggen in **/var/log/daemon.log**; dat is handig bij problemen, maar kan erg lange logfiles veroorzaken. Daarvoor moet je in **/etc/bind/named.conf** in de sectie 'logging' de regel "category queries { default_syslog; };" toevoegen en weer doorgeven met **/etc/init.d/bind reload**.

Merk op dat je ook meerdere 'zones' met één nameserver kunt beheren (gewoon meer bestanden van het type **db.dorpsnet** en **db.dorpsnet-rev**), maar meestal zal dat niet nodig zijn.

Meer informatie staat in de manpage van **named**, in DNS-HOWTO en in **/usr/share/doc/bind**. Let echter op dat hier en daar nog oude configuratie-opties worden gemeld, die met de nieuwe BIND niet meer werken. Bovendien komen veel bestandsnamen niet overeen met die op een Debian systeem.

13. Andere interessante zaken

In dit hoofdstuk nog enkele zaken die niet echt berekking hebben op de voorgaande hoofdstukken, maar toch van belang kunnen zijn.

13.1 De datum en de tijd

In Linux bekijk je de huidige datum en tijd met het commando **date**, zie ook de manual page. Maar daarmee is nog lang niet alles gezegd.

In een computer zitten twee klokken, een hardware-klok en een software-klok. De hardware-klok werkt altijd, of de computer nu aan of uit staat, en valt in dezelfde categorie als het CMOS Setup-geheugen. De hardware-klok is ook met de CMOS Setup (meestal DEL tijdens het opstarten) te veranderen. De software-klok werkt alleen als er op de computer een programma draait dat die klok bijhoudt. Dat programma moet dus op geregelde tijden ergens een (seconde-)teller ophogen.

De Linux kernel heeft zo'n software-klok, die de 'system time' bijhoudt. Tijdens het opstarten van de kernel wordt de software-klok ingesteld op de waarde die op dat moment van de hardware-klok (ook wel 'CMOS clock' genoemd) af te lezen valt. Vanaf dat moment wordt de hardware-klok 'vergeten' en wordt alleen naar de software-klok gekeken. Dat is omdat de Linux software-klok vele malen nauwkeuriger is dan de hardware-klok. Met **date** zie je dus de software-klok.

13.1.1 De klok goedzetten

Je kunt de software-klok (als root) verzetten met een commando als **date --set '04/02/01 19:55:30'**, dan wordt het 2 april 2001, 4-en-een-halve minuut voor 8 uur 's avonds.

De hardware-klok krijgt automatisch de tijd van de software-klok als je Linux afsluit. Dat is omdat de software-klok meestal nauwkeuriger loopt dan de hardware-klok. Een extraatje is dat Linux de hardware-klok automatisch corrigeert als die te snel of te langzaam lijkt te lopen. In de praktijk blijkt dat echter niet altijd even goed te werken. Het is daarom in het algemeen een goed idee om, steeds als je **date --set** hebt gedaan, ook **echo '' > /etc/adjtime** te doen. Daarmee wordt de 'oude' informatie vergeten en wordt de tijd meestal nauwkeuriger bijgehouden.

Met de hardware-klok hoef je dus zelf nooit iets te doen, dat gaat automatisch. Meer informatie over hardware- en software-klok staat in **man hwclock**.

13.1.2 Tijdzones, zomertijd en wintertijd

De Europese tijd (tijdzone CEST, Central European Standard Time) loopt in de winter één en in de zomer twee uur voor op GMT (Greenwich Mean Time, of UTC, Coordinated Universal Time). Het voordeel van GMT is, dat het geen zomer- en wintertijd kent, en dus nooit een uur vooruit of terug gezet hoeft te worden.

Als de hardware-klok op GMT draait (en als Linux dat weet), zal Linux je nog steeds de lokale tijd (CEST dus) geven. Maar DOS bijvoorbeeld weet niets van tijdzones af en gaat ervan uit dat de hardware-klok op lokale tijd draait. Als je dus naast Linux nog een ander operating system draait, kun je de hardware-klok het beste op lokale tijd laten draaien.

Van elke tijdzone (eigenlijk van elk land) is een aantal zaken bekend, in de directory **/usr/share/zoneinfo**. Eén van die dingen is de zomer- en wintertijd. Als de hardware-klok op GMT draait, zul je merken dat de zomer- en wintertijd automatisch wordt aangepast (in de software-klok dus, de GMT van de hardware-klok kent immers geen zomer- en wintertijd). Als de hardware-klok op lokale tijd draait, gaat het soms wel, en soms ook niet goed; ook als je af en toe Windows 95/98 draait, die zelf ook zomer- en wintertijd kan omschakelen. Het kan nooit kwaad om op dergelijke momenten te kijken of alles nog goed loopt, en het zonodig aan te passen (en vooral een **echo '' > /etc/adjtime** niet vergeten, zie boven).

Voor gebruikers uit een ander land of een andere tijdzone is er nog een leuk trucje: zet de environment variabele TZ (van timezone) op de juiste tijdzone, en je krijgt automatisch de omgerekende tijd. Probeer maar eens **export TZ=US/Central** en dan **date**, of **ls -alF**. Type **unset TZ** en bekijk hoe de datum of de directory er werkelijk uitziet.

Als root kun je de standaard (system-wide) tijdzone instellen met het programma **/usr/sbin/tzconfig**; in /etc/default/rcS kun je aangeven of de hardware-klok op GMT (=UTC) loopt.

13.2 Gebruikers, groepen en passwords: /etc/passwd en /etc/group

Gegevens over gebruikers worden opgeslagen in **/etc/passwd**. Dat is een database-achtig tekstbestand: elke gebruiker heeft één regel en elke regel heeft een aantal *velden* die gescheiden worden door dubbele punten (':'). Die velden zijn in volgorde: gebruikersnaam, password, gebruikersnummer (*uid*, *user id*), 'primair' groepsnummer (*gid*, *group id*), 'informatie', home directory, opstartshell. Het veld 'informatie' is ook weer onderverdeeld in komma-gescheiden subvelden, namelijk volledige naam, kamernummer, werk-telefoonnummer, thuis-telefoonnummer, 'andere informatie'.

13.2.1 Passwords

Het password dat in **/etc/passwd** staat is *ge-encrypt* ('gecodeerd'). Dit encrypten is een één-richtings-actie, dat wil zeggen dat het echte password wel in de ge-encrypte code kan worden omgezet, maar niet andersom. Tijdens het inloggen wordt het door jou ingetypte password ook ge-encrypt en wordt vergeleken of de ge-encrypte codes hetzelfde zijn.

Het is in principe een veiligheidsrisico als het ge-encrypte password in **/etc/passwd** staat. Dat bestand is namelijk publiekelijk toegankelijk en er zijn programma's beschikbaar die hele lijsten met veelgebruikte passwords gaan 'uitproberen'. Daarom is er *shadowing* bedacht: de passwords staan nu in een apart bestand **/etc/shadow**, waar alleen bepaalde programma's (zoals **login** en **passwd**) bij kunnen; in **/etc/passwd** staan x-jes op de plaats van het ge-encrypte password. Je kunt shadowing als root aan- en uitzetten met **shadowconfig on** of **off**; de bestanden **/etc/passwd** en **/etc/shadow** worden daarmee automatisch aangepast.

Je kunt je password veranderen met het commando **passwd**, dat dan vraagt om je oude en (twee keer) je nieuwe password. Als je root bent, kun je het password van een willekeurige gebruiker veranderen met **passwd <username>**.

Je kunt de inhoud van het 'informatieveld' veranderen met het commando **chfn** (of als root eventueel **chfn <username>**) en de opstartshell met **chsh**. NB: verander nooit de opstartshell van root!

13.2.2 Groepen

Iedere gebruiker is lid van precies één '*primaire*' *groep*. Bestanden die door die gebruiker worden aangemaakt, worden van die groep (bekijk dat met **ls -alF**). Maar naast de primaire groep kunnen gebruikers ook lid zijn van andere groepen, dat wil zeggen dat ze dan ook bestanden kunnen bekijken/veranderen die leesbaar/schrijfbaar zijn voor zo'n andere groep.

De groepen worden bijgehouden in **/etc/group**. In het vierde veld (achter de laatste dubbele punt) staan de komma-gescheiden namen van gebruikers die lid zijn van die groep. Het lidmaatschap van de primaire groep wordt echter niet in **/etc/group** aangegeven, maar in het vierde veld van **/etc/passwd**. In **/etc/group** worden ook de groepnamen gekoppeld aan de groepnummers (die bijvoorbeeld in dat vierde veld van **/etc/passwd** gebruikt worden).

Je kunt met het commando **groups** zien, van welke groepen je lid bent; het commando **id** geeft nog wat meer informatie. Merk op dat groeps-veranderingen pas merkbaar zijn als je een volgende keer inlogt.

13.2.3 Nieuwe gebruikers en groepen

Zoals in paragraaf 5.4 al is aangegeven, maak je een nieuwe gebruiker aan met **adduser <nieuwe-username>**. Standaard krijgt die gebruiker een eigen nieuwe (primaire) groep, met zijn eigen naam. Je kunt nieuwe gebruikers ook direct aan een andere, bestaande groep toevoegen (d.w.z. een andere primaire groep geven) door **adduser --ingroup <groepnaam> <nieuwe-username>**.

Een bestaande gebruiker toevoegen aan een bestaande groep gaat met **adduser <username> <groepnaam>**. Gebruikers uit een groep verwijderen die niet hun primaire groep is, kan alleen door met de hand **/etc/group** te editen. Gebruikers een andere primaire groep geven, kan alleen door het juiste veld in **/etc/passwd** te wijzigen.

Om een bestaande gebruiker 'weg te gooien', gebruik je **userdel -r <username>**. Hiermee wordt zonder bevestiging ook de home directory gewist, dus voorzichtig!

Een nieuwe groep maken gaat met **addgroup <nieuwe-groepnaam>**. Een bestaande groep 'weggooien' is over het algemeen geen goed idee.

Instellingen voor adduser en addgroup staan in **/etc/adduser.conf**.

13.2.4 Eigenaar en groep van bestanden

Bestanden hebben één eigenaar en één groep, waarvoor permissies kunnen worden opgegeven, zie paragraaf 4.11.

Als root kun je de eigenaar en/of de groep van een bestand veranderen met het commando **chown <username>:<groupname> <file(s)>**, waarbij je **<username>** of **:<groupname>** ook mag weglaten. Met **chgrp <groupname> <file(s)>** kan alleen de groep van een bestand worden veranderd.

Als je als gewone gebruiker lid bent van een aantal groepen, kun je zelf ook **chgrp** gebruiken om een bestand van groep te veranderen, maar alleen 'jouw' groepen mogen dan gebruikt worden.

Als je van plan bent veel bestanden te gaan maken die een andere groep moeten krijgen dan jouw primaire groep, kun je met **sg <groupname>** tijdelijk een andere primaire groep krijgen. Dit moet natuurlijk wel weer een groep zijn waar jij lid van bent.

13.2.5 Password vergeten

Het kan gebeuren dat je als gewone gebruiker je password vergeet. De procedure is eenvoudig: geef als root het commando **passwd <jouwusername>** en verzin een beter-te-onthouden-password.

Veel ernstiger is het, als je het password van root vergeten bent, maar ook dat kan opgelost worden. Je moet het systeem met Ctrl-Alt-Del rebooten en dan opstarten van een rescue disk (voor het maken daarvan, zie paragraaf 3.1.4), en ga in het installatie-hoofdmenu direct met LinkseAlt-F2 naar de tweede virtuele console. Geef daar een Enter en dan een commando als **mount /dev/hda5 /target** om je root partitie te mounten. Dan kun je met **ae /target/etc/passwd** of **ae /target/etc/shadow** de passwords editen. Je moet van de regel 'root:<passwd>:<getal>:<getal>:...' het tweede veld helemaal leeghalen, dan wordt het dus 'root::<getal>:<getal>:...'. Geef dan de commando's **umount /target** en **reboot** (vergeet niet de rescue disk uit de floppy drive te halen). Zodra je Linux weer normaal opgestart hebt, log je in als root, daar heb je nu geen password meer voor nodig. Met **passwd** geef je root vervolgens een nieuw password.

Meer informatie over alle genoemde commando's staat in hun respectievelijke manual pages.

13.3 Grote geheugens

Als je meer dan 64 MB werkgeheugen hebt, kan het gebeuren dat Linux er maar 64 detecteert, meestal vanwege een 'verouderd' BIOS (sommige Compaq's geven zelfs maar 16 MB door). Als je toch meer geheugen wilt gebruiken, moet je dat zelf bij het opstarten aan de kernel doorgeven.

Bij het gebruik van de Custom Boot floppy, moet je het bestand **syslinux.cfg** op die diskette wijzigen. Dat kan in DOS, maar ook in Linux (met **mcopy** even naar bijvoorbeeld **/tmp** kopiëren). Achter de regel "APPEND root=/dev/hdXX ro" voeg je toe "mem=128MB" of iets dergelijks.

Bij LOADLIN vanuit DOS/Windows moet je bijvoorbeeld "mem=128MB" toevoegen aan de LOADLIN command-line (waar al "root=/dev/hdXX" staat).

LILO-gebruikers moeten **/etc/lilo.conf** aanpassen. Voor de eerste "image=/XX"-regel moet een regel "append="mem=128MB"" komen (de buitenste aanhalingstekens niet overnemen, de binnenste wel). Als er al een "append"-regel bestaat, moet je "mem=128MB" eraan toevoegen. Vergeet niet **/sbin/lilo** te draaien als je **/etc/lilo.conf** veranderd hebt!

Let op: sommige computers bieden niet het totale geheugen ter gebruik aan. Om problemen te voorkomen, kan het verstandig zijn 1 MB te weinig aan de Linux kernel door te geven.

In de BootPrompt HOWTO staat meer informatie over allerlei beschikbare kernel-opties. Zie paragraaf 4.21 voor informatie over de HOWTO's.

13.4 Gang van zaken tijdens opstarten, runlevels en dergelijke

Tijdens het opstarten van Linux gebeurt ongeveer het volgende:

- kernel boot
- **/sbin/init** wordt door kernel gestart
- **init** leest **/etc/inittab**
- **init** zoekt in **inittab** naar regel "si::sysinit:..."
- **init** voert **/etc/init.d/rcS** uit
- **/etc/init.d/rcS** voert **/etc/default/rcS** uit
- **/etc/init.d/rcS** voert alle programma's (symlinks) in **/etc/rcS.d** uit, in oplopende-nummer-volgorde
- **/etc/init.d/rcS** voert alle programma's in **/etc/rc.boot** uit (zijn er meestal niet)
- **init** zoekt in **inittab** naar "l2:2:wait:..." regel (2 is runlevel, aangegeven in regel "id:<runlevel>:initdefault:")
- **init** voert **/etc/init.d/rc 2** uit
- **/etc/init.d/rc** voert alle programma's in **/etc/rc2.d** uit, in volgorde van oplopende nummers (**S<nummer><prognaam>**), met argument '**start**'
- **init** zorgt voor het opstarten van de **getty**'s (login prompts), regels "<tty>:<runlevels>:respawn:..." en je kunt inloggen.

Traditioneel heeft Linux een aantal *runlevels*, die aangegeven worden met de cijfers 0 t/m 6. Elk runlevel biedt een aantal (mogelijk verschillende) services, zoals een aantal **getty**'s of netwerktoegang. Dat wil zeggen, dat bijvoorbeeld in runlevel 3 geen netwerk beschikbaar wordt gemaakt en in runlevel 5 wel. Dat onderscheid is tegenwoordig echter wat verdwenen; de runlevels zijn in Debian als volgt:

0: system halt
1: single-user
2-5: multi-user
6: system reboot

Standaard wordt in runlevel 2 opgestart; runlevels 3 t/m 5 zijn identiek aan 2, maar die kun je eventueel veranderen. Op mijn laptop heb ik bijvoorbeeld in runlevel 2 de PCMCIA- en netwerkservices uitgezet, zodat het 'opstarten onderweg' wat sneller verloopt. Als ik een netwerk nodig heb, moet ik naar runlevel 3 overgaan. Dat gaat met het commando **init <runlevel>**, dus in dit geval **init 3**. Runlevel 1 kan worden gebruikt om systeembeheer-werkzaamheden te verrichten waarbij 'gewone gebruikers' niet mogen storen (dat zijn dan altijd lokale werkzaamheden, want in runlevel 1 is het netwerk niet beschikbaar).

Bij het overstappen naar een ander runlevel gebeurt grofweg het volgende:

- **init** zoekt in **inittab** naar "l<runlevel>:<runlevel>:wait:..." regel
- **init** voert **/etc/init.d/rc <runlevel>** uit
- **/etc/init.d/rc** voert alle K-scripts in **/etc/rc<runlevel>.d** uit, in oplopende-nummer-volgorde, met argument '**stop**'; deze scripts stoppen (killen) mogelijk een aantal services
- **/etc/init.d/rc** voert alle S-scripts in **/etc/rc<runlevel>.d** uit, in oplopende-nummer-volgorde, met argument '**start**', voor zover die services nog niet gestart zijn; deze scripts starten een aantal services
- **init** zorgt eventueel voor het opstarten (of afschieten) van **getty**'s (login prompts).

Deze manier van doen geldt ook voor runlevels 0 (halt) en 6 (reboot), waar dus eerst alle mogelijke services gestopt moeten worden. Het eigenlijke halten/rebooten gebeurt in de scripts **/etc/rc0.d/S90halt** respectievelijk **/etc/rc6.d/S90reboot**. Je zou dus ook **init 0** en **init 6** kunnen gebruiken voor halt en reboot, maar **shutdown** (en afgeleide **halt** en **reboot**) is netter.

Als je iets automatisch wilt laten starten en stoppen tijdens de boot-sequence respectievelijk de shutdown-sequence, kan dat eenvoudig gerealiseerd worden met een scriptje in de betreffende **/etc/rc<runlevel>.d** directory. Ik adviseer de namen **/etc/rc2.d/S99zzz_van_mij** voor zo'n start-script en zowel **/etc/rc0.d/K00aaa_van_mij** (halt)

als **/etc/rd6.d/K00aaa_van_mij** (reboot) voor zo'n stop-script. Het verdient aanbeveling deze scripts in de home-directory van root te zetten, bijvoorbeeld in **/root/init.d/**, en dan symlinks te gebruiken in **/etc/rc<runlevel>.d**. Voor meer over scripts, zie paragraaf 7.1; voor symlinks paragraaf 4.12.

14. Tenslotte

Dit is alweer het einde van deze handleiding. Er zijn (tenminste) honderd-en-één dingen die ik niet heb beschreven. Die dingen zul je zelf uit moeten vogelen. Maar omdat je de basis nu kent, verwacht ik niet al te veel problemen. Kijk nog eens in paragraaf 4.21, waar diverse manieren staan opgesomd waarop je hulp kunt krijgen.

Als je niet meer weet wat je met Debian zou moeten doen, raad ik je aan eens (als niet-root!) met bijvoorbeeld **`dselect`** door alle pakketjes te bladeren -- je zult zien dat er hele interessante zaken bij zitten, bijvoorbeeld 'xephem' of 'rocks-n-diamonds'. Probeer gewoon alles uit wat je leuk lijkt, maar denk er ondertussen wel aan af en toe de ongebruikte pakketjes te **`apt-get remove`**-en (tenzij je vele gigabytes aan schijfruimte hebt).

Ongetwijfeld heb je op- en aanmerkingen over deze handleiding. Bijvoorbeeld dat ik iets niet duidelijk uitleg, dat ik verschrikkelijk veel taal- en stijlfouten maak, dat niet alles werkt zoals ik het beschrijf en dat iets heel belangrijks totaal niet aan de orde komt. Ik hoor die dingen graag, dan kan ik ze in volgende versies verbeteren. Stuur alle (liefst opbouwende) kritiek, commentaar en suggesties naar J.A.Bezemer@ITS.TUDelft.NL, of, als dat adres niet (meer) werkt, costar@panic.et.tudelft.nl of mcbzmr@dds.nl.

Happy Linuxing!

APPENDIX: GNU Free Documentation License

GNU Free Documentation License
Version 1.1, March 2000

0. PREAMBLE
The purpose of this License is to make a manual, textbook, or other written document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.
This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.
We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS
This License applies to any manual or other work that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you".
A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.
A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (For example, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.
The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License.
The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License.
A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, whose contents can be viewed and edited directly and straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup has been designed to thwart or discourage subsequent modification by readers is not Transparent. A copy that is not "Transparent" is called "Opaque".
Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML designed for human modification. Opaque formats include PostScript, PDF, proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML produced by some word processors for output purposes only.
The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

2. VERBATIM COPYING
You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.
You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY
If you publish printed copies of the Document numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.
If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.
If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a publicly-accessible computer-network location containing a complete Transparent copy of the Document, free of added material, which the general network-using public has access to download anonymously at no charge using public-standard network protocols. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.
It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS
You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.
B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has less than five).
C. State on the Title page the name of the publisher of the Modified Version, as the publisher.
D. Preserve all the copyright notices of the Document.
E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.
F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.
G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.
H. Include an unaltered copy of this License.
I. Preserve the section entitled "History", and its title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.
J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.
K. In any section entitled "Acknowledgements" or "Dedications", preserve the section's title, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.

L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.
M. Delete any section entitled "Endorsements". Such a section may not be included in the Modified Version.
N. Do not retitle any existing section as "Endorsements" or to conflict in title with any Invariant Section.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.
You may add a section entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties--for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.
You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.
The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS
You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice.
The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.
In the combination, you must combine any sections entitled "History" in the various original documents, forming one section entitled "History"; likewise combine any sections entitled "Acknowledgements", and any sections entitled "Dedications". You must delete all sections entitled "Endorsements."

6. COLLECTIONS OF DOCUMENTS
You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.
You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS
A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, does not as a whole count as a Modified Version of the Document, provided no compilation copyright is claimed for the compilation. Such a compilation is called an "aggregate", and this License does not apply to the other self-contained works thus compiled with the Document, on account of their being thus compiled, if they are not themselves derivative works of the Document.
If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one quarter of the entire aggregate, the Document's Cover Texts may be placed on covers that surround only the Document within the aggregate. Otherwise they must appear on covers around the whole aggregate.

8. TRANSLATION
Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License provided that you also include the original English version of this License. In case of a disagreement between the translation and the original English version of this License, the original English version will prevail.

9. TERMINATION
You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

10. FUTURE REVISIONS OF THIS LICENSE
The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See http://www.gnu.org/copyleft/.
Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

How to use this License for your documents
To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

> Copyright (c) YEAR YOUR NAME.
> Permission is granted to copy, distribute and/or modify this document
> under the terms of the GNU Free Documentation License, Version 1.1
> or any later version published by the Free Software Foundation;
> with the Invariant Sections being LIST THEIR TITLES, with the
> Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.
> A copy of the license is included in the section entitled "GNU
> Free Documentation License".

If you have no Invariant Sections, write "with no Invariant Sections" instead of saying which ones are invariant. If you have no Front-Cover Texts, write "no Front-Cover Texts" instead of "Front-Cover Texts being LIST"; likewise for Back-Cover Texts.
If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.

Linux is een gratis operating system dat steeds populairder wordt. Het is de meest stabiele multi-user multi-tasking omgeving voor onder andere PC's vanaf de 80386.

Debian GNU/Linux biedt naast het complete Linux operating system een schat aan programmatuur voor elke denkbare toepassing.

Deze Handleiding Debian GNU/Linux 2.2 gaat in op de achtergronden, de installatie en het gebruik van Debian. Allereerst wordt algemene basiskennis aangereikt over operating systems, Linux, devices en netwerken. Vervolgens wordt de installatieprocedure van Debian zeer uitvoerig beschreven. Tenslotte komen de Linux basiscommando's, administratieve zaken, het X Window System en vele toepassingen uitgebreid aan bod.

Kortom, deze Handleiding geeft een complete introductie in de Debian distributie van het Linux operating system, waarna de gebruiker gemakkelijk zijn eigen weg zal kunnen vinden.

De auteur studeert Elektrotechniek aan de Technische Universiteit te Delft. Voor zowel studie- als privédoeleinden maakt hij vrijwel uitsluitend gebruik van Debian GNU/Linux. Recent ontwikkelde hij in het kader van een studieproject het programma GramoFile, dat onder Linux geluidssignalen van grammofoonplaten kan overzetten op schrijfbare CD's. Verder heeft hij een actieve rol in de elektronische en fysieke distributie van Debian CD's.