

Linux Gebruikers Handleiding

Versie 1.6-pre2

Jarkko Huijts
januari 2009

Over deze handleiding

De nieuwste versie van deze handleiding is altijd te vinden op <http://jarkko.nl>

De Linux Gebruikers Handleiding (LGH) behandelt alle aspecten en onderdelen van een gemiddeld Linux-systeem (distributie). Waar het interessant is, wordt achtergrondinformatie en geschiedenis gegeven. Verder wordt basiskennis behandeld die van toepassing is op alle distributies en soms zelfs alle Unices. Deze handleiding is niet distributiespecifiek. De nadruk ligt op begrip.

Anders dan in een gemiddeld boek geeft deze handleiding vrij veel informatie op weinig ruimte, zodat je weinig hoeft te lezen om veel te weten te komen. Alles wordt beknopt uitgelegd. Er wordt aangegeven waar je meer informatie kunt vinden indien nodig.

Het publiek dat deze handleiding (hopelijk) nuttig zal vinden, omvat mensen die willen beginnen met het gebruiken van Linux, maar ook mensen die er meer over willen weten.

De auteurs

Deze handleiding is begonnen door Jarkko Huijts toen hij nog informatica studeerde aan de UT in Enschede. Martin Herrman, destijds student aan de TU/e in Eindhoven, heeft meegewerkt met het schrijven van de eerste versies. Sinds versie 1.4 schrijft Martin niet meer actief mee. Aan versie 1.5 heeft Peter de Jong meegewerkt.

Licentie en contact

Tekst uit deze handleiding mag in elke vorm worden gekopieerd, vertaald, uitgebreid en/of gedistribueerd. Kopieer of gebruik je de handleiding in zijn geheel, dan zou ik het waarderen als je een berichtje stuurt naar jarkko@gmx.net. Dan weet ik ook wat er allemaal met dit document gebeurt. Heb je een fout gevonden, opmerkingen of wil je graag je lof betuigen, twijfel dan ook niet om een mailtje te sturen.

Inhoudsopgave

1	Inleiding	11
1.1	Wat is Linux?	11
1.2	Enkele begrippen	11
1.3	Geschiedenis van Linux	12
1.3.1	Unix	12
1.3.2	Het GNU copyleft	13
1.3.3	Linux	13
1.4	Andere Unices	14
1.5	Sterke en zwakke punten	14
1.6	Waar Linux gebruikt wordt	15
1.7	Tux de pinguïn	15
1.8	Vrije software vs. open source	16
1.9	Licenties	16
2	Het vinden van informatie	17
2.1	Linux Documentation Project	17
2.2	HOWTO	17
2.3	Manual pages	17
2.4	Info pages	17
2.5	/usr/share/doc	18
2.6	Website van de distributie	18
2.7	Nieuwsgroepen en forums	18
2.8	Het web	18
2.9	Boeken en tijdschriften	18
3	Hardwareondersteuning	19
3.1	Lijsten	19
3.2	Videokaart	19
3.3	Geluidskaart	19
3.4	Printer en scanner	19
3.5	Tv-kaart	19
3.6	Digitale camera	20
4	Distributies	21
4.1	Verschillen	21
4.2	Enkele bekende	21
4.3	Live-cd's	22
4.4	Verkrijgen	22
4.5	Je aanmelden als Linux-gebruiker	23
5	Partities	25
5.1	Soorten	25
5.2	Bestandssystemen	26
5.3	Partitioneren	26
5.4	Partities voor Linux	27
6	Basiskennis	29
6.1	Gebruikers	29
6.2	Consoles en terminals	30
6.3	Werken met consoles en terminals	31
6.4	Basiscommando's	31
6.5	Pipelines, redirecting en backquotes	33
6.6	Permissies en eigenaren	34
6.7	Variabelen	36
6.8	Aliases	36
6.9	Bestanden bekijken en editen	36
6.9.1	Vim	37

6.9.2	Emacs	37
6.9.3	Pico en Nano	38
6.9.4	Less	38
6.10	Directoryhiërarchie	38
6.11	Archiveren en comprimeren	39
6.12	Mounten	39
6.13	Geheugenbeheer	40
6.14	Processen	41
6.15	Gebruikers in de gaten houden	42
6.16	Printen	42
7	GNU-utility's	43
7.1	Cat	43
7.2	Grep	43
7.3	Strings	44
7.4	Sort en uniq	44
7.5	Cut	44
7.6	Sed	44
7.7	Awk	44
7.8	Tr	45
7.9	Tee	45
7.10	Locate	45
7.11	Find	45
7.12	Xargs	46
7.13	File	46
7.14	Head en tail	46
7.15	Ldd	46
7.16	Touch	46
7.17	Split	47
7.18	Diff	47
7.19	Patch	47
8	Systeemconfiguratie	49
8.1	Het opstartproces en runlevels	49
8.2	Library's installeren	49
8.3	Partities op fouten controleren	50
8.4	Tijd en datum instellen	50
8.5	Logbestanden	51
8.6	At	51
8.7	Cron	51
8.8	Het printstelsysteem	52
8.9	Geluid	53
9	Kernel	55
9.1	Wat is de kernel?	55
9.2	Kernel compileren	55
9.3	Modules	56
9.4	Bootloader	56
9.4.1	LILO	56
9.4.2	GRUB	57
10	Netwerken	59
10.1	Ethernet en UTP	59
10.2	Netwerkbegrippen	59
10.3	Handige programma's	61
10.4	Hosts en resolv.conf	61
10.5	Masquerading firewall	62
10.6	Samba	62
10.7	Sendmail en fetchmail	63

11 Security	65
11.1 Introductie	65
11.2 Rootkits	65
11.3 Wachtwoorden	65
11.4 Inloggen zonder wachtwoord	65
11.5 (X)inetd, hosts.allow en hosts.deny	66
11.6 Firewall	66
11.7 Sudo	67
11.8 SSH	67
12 Software	69
12.1 Overzichten en hosting	69
12.2 Source code	69
12.3 Het systeem van Slackware	69
12.4 Het systeem van Debian	70
12.5 RPM	71
12.6 Het systeem van Gentoo	71
13 X windowing system	73
13.1 Introductie	73
13.2 Drivers	73
13.3 Configuratie	73
13.4 Window managers en desktopomgevingen	75
13.5 Toolkits	76
13.6 X starten	77
13.6.1 Startx	77
13.6.2 Display manager	78
13.7 Xset	78
13.8 X resources	78
13.9 Fonts	78
13.9.1 Core X11 font system	79
13.9.2 Xft	79
13.9.3 Goede fonts	80
14 Op afstand werken	81
14.1 X over het netwerk	81
14.2 SSH	81
14.3 Screen	81
15 Multimedia	83
15.1 Cd's en dvd's branden	83
15.1.1 SCSI-emulatie	83
15.1.2 Tools voor cd's	83
15.1.3 Tools voor dvd's	84
15.2 Muziek afspelen	84
15.3 Video afspelen	84
15.4 Dvd-films afspelen	85
15.5 Dvd's rippen	85
15.6 Tv-kaart	86
15.7 Video editen	86
15.8 Patenten en andere wettelijke restricties	86
16 Internationalisering	89
16.1 Karaktersets en encodings	89
16.2 Locales	89
16.3 Iconv	91
16.4 Xmodmap	91
16.5 Input methods	92
16.6 Andere invoermethoden	92

16.7	Het euroteken	93
17	Windows-gerelateerd	95
17.1	Windows-applicaties draaien onder Linux	95
17.2	DOS-applicaties draaien onder Linux	95
17.3	Windows-partities benaderen vanuit Linux	96
17.4	Linux-partities benaderen vanuit Windows	96
17.5	Defragmenteren	96
18	Bash scripting	97
19	Errors	101
19.1	Bash: naam: command not found	101
19.2	Segmentation fault en bus error	101
19.3	Use "exit" to leave the shell	101
20	Tips & Trucs	103
20.1	Beeld vol vreemde tekens	103
20.2	Core dumps	103
20.3	Dingen tijdens het opstarten laten uitvoeren	103
20.4	Firefox: snel een URL openen	103
20.5	Functie ctrl-alt-del	104
20.6	Geheugen: er wordt minder gebruikt dan beschikbaar is	104
20.7	GNOME: instellingen buiten de desktopomgeving om	104
20.8	ISO en andere archiefformaten mounten	104
20.9	Knippen en plakken	104
20.10	Magic SysRq	105
20.11	Maximal mount count	105
20.12	Melding bij het inloggen veranderen	105
20.13	MPlayer: stream opnemen	105
20.14	Muisknoppen omdraaien in X	106
20.15	Norton Commander-achtige file managers	106
20.16	Num-lock in alle consoles inschakelen	106
20.17	Omhoog scrollen	106
20.18	Output van ls in kleur	107
20.19	Prompt aanpassen	107
20.20	Screenshot maken	107
20.21	Scripts in je home directory	108
20.22	Sendmail: starten duurt erg lang	108
20.23	Signature automatisch laten updaten	108
20.24	Standaard editor	109
20.25	Tekstbestanden van/naar DOS-formaat omzetten	109
20.26	Terughalen van verwijderde bestanden	109
20.27	Titel van een terminal veranderen	110
20.28	Uitschakelen van de pc-speaker	110
21	Links	111
21.1	Beveiliging	111
21.2	Distributies	111
21.3	Documentatie	111
21.4	Emulators, virtuele machines e.d.	111
21.5	Graphics	112
21.6	Hardware	112
21.7	Internet en netwerken	112
21.8	Kernel	112
21.9	Multimedia	113
21.10	Nieuws	113
21.11	Office	113
21.12	Softwareoverzichten	113

21.13	TeX en L ^A TeX	114
21.14	Utility's	114
21.15	Windows-gerelateerd	114
21.16	X	114

1 Inleiding

1.1 Wat is Linux?

Om bij het begin te beginnen: wat is Linux? Linux is een kernel, de kern van een besturingssysteem, waar Linus Torvalds aan is begonnen. De eerste versies heeft hij geheel zelf gemaakt. In de loop der tijd kreeg hij echter steeds meer hulp van allerlei mensen van over de hele wereld, bij elkaar gebracht dankzij het internet. De bekendste "kernel hacker" naast Linus is Alan Cox.

Een besturingssysteem (of operating system, OS) is basissoftware voor een computer die de gebruiker de mogelijkheid biedt het systeem te gebruiken. De kernel is het deel dat als eerste wordt geladen en dan aanwezig blijft in het geheugen. Het vervult essentiële taken, zoals geheugenmanagement, scheduling om taken te verdelen over de processortijd, en communicatie met de hardware via device drivers.

Unix was een van de eerste besturingssystemen, maar werd niet gebruikt op computers voor het grote publiek. De pc van IBM en de Macintosh van Apple waren computers die de mensen in de jaren 1980 thuis gingen gebruiken. Voor de pc was het eerste OS MS-DOS. MS-DOS is een typisch voorbeeld van een tekstgebaseerd OS dat je via een command line bestuurt. Zo werkten alle OS'en tot dan toe, ook Unix. Een nieuwer paradigma is de desktop, een grafische interface die je met de muis bedient. Voorbeelden van grafische OS'en zijn Windows voor de pc en Mac OS voor de Macintosh.

Er bestaan vele verschillende hardwareplatformen en OS'en. Niet elk OS werkt op elke hardware. De meeste mensen hebben thuis een IBM-compatible pc en sommigen een computer van Apple. Andere hardware is veel minder gangbaar. Als je een pc koopt, zit daar vrijwel standaard Windows bij. Dat is echter niet het enige OS dat je op een pc kunt gebruiken. Andere mogelijkheden zijn NetBSD, FreeBSD en Linux, alle Unix-achtige OS'en. Hiervan is vooral Linux bekend geworden.

Strikt gesproken is Linux de naam van de kernel. Op zichzelf is de kernel niet zo interessant. Het biedt geen mogelijkheid voor de gebruiker om de computer te bedienen. Er is meer nodig. Wat vaak met "Linux" wordt bedoeld, is een Linux-distributie. Dat is een samengesteld geheel van de kernel en andere software. Samen vormt het wel een volledig OS, en meer. Belangrijk zijn de GNU-utility's. GNU is een recursief acroniem dat staat voor GNU's Not Unix. Het is een set utility's die is gemaakt door de FSF (Free Software Foundation) en een alternatief vormt voor de utility's die bij Unices te vinden zijn. (Meer over de FSF en GNU komt verderop.) Bij deze utility's hoort o.a. een shell, software die een command line toont zoals bij MS-DOS. De kernel en GNU-utility's vormen samen een volledig functionerend OS dat dezelfde mogelijkheden biedt als Unix, maar dan op een pc. Omdat de software van het GNU-project net zo belangrijk is als de kernel, is het eigenlijk correcter het systeem GNU/Linux te noemen.

De moderne gebruiker wil graag een grafische desktop. Ook dat is beschikbaar bij Linux-distributies. Ten grondslag aan alle grafische software voor Linux (en andere Unices) ligt X. Software die een desktopomgeving vormt, maakt gebruik van X. Er zijn twee belangrijke projecten voor desktopsoftware: GNOME en KDE.

Je merkt dat een Linux-distributie is opgebouwd uit een hoop verschillende onderdelen. Naast de hier genoemde onderdelen is er nog veel meer software te vinden bij een distributie. Wat al deze software gemeen heeft, is dat het open source is. Verder wordt het door vele verschillende individuen en groepen gemaakt.

1.2 Enkele begrippen

Met open-sourcesoftware sta je dicht bij de broncode. Veel wat je bij Linux ziet, zul je beter begrijpen als je iets weet over hoe software gemaakt wordt en in elkaar zit. De volgende begrippen zijn belangrijk.

Broncode of source code Om het eenvoudiger te maken software te ontwikkelen zijn programmeertalen uitgevonden. Broncode of source code is het programma in de een of andere programmeertaal. Het moet door een compiler (vertaler) heen om er machinecode van te maken, een serie instructies die de

computer direct kan uitvoeren.

Assembler Programmeertaal op het laagste niveau. Het staat net iets boven machinecode, maar niet veel. Je beschrijft er instructies mee die de processor moet uitvoeren.

C Een programmeertaal van een hoger niveau. C is gemaakt voor Unix en zodoende is veel software voor Linux (en de kernel en GNU-software zelf ook) in C geschreven. In C kun je functies of routines maken.

Executable Een bestand dat direct kan worden uitgevoerd door de computer. Broncode is dat niet. Na het te hebben gecompileerd zal een bestand in objectformaat worden aangemaakt. Dat bevat code die wel door de computer is uit te voeren. In Linux hebben deze bestanden een speciale flag die aangeeft dat het uitvoerbaar is en verder geen extensie. In Windows hebben ze exe als extensie.

Library of bibliotheek Een (shared) library is een collectie voorgecompileerde routines die door een programma gebruikt kunnen worden. Een library is ook in objectformaat opgeslagen. De bestandsnamen eindigen op .so in Linux en .dll in Windows. Een library is handig voor routines die in meer software nodig zijn. Zo zijn er library's voor het decoderen van MP3 en het tekenen van GUI-componenten.

Porten Een stuk software naar een ander platform (hardware/OS) omzetten. Dit is het eenvoudigste als het geschreven is in een hogere programmeertaal zoals C. Het kan dan door een andere compiler gehaald worden om code voor het nieuwe platform te maken. Tussen Unix-achtige OS'en is porten vrij eenvoudig, zodat de meeste open-sourcesoftware onder alle Unices te gebruiken is.

1.3 Geschiedenis van Linux

De tekst in deze paragraaf is onder het GNU "copyleft" gedoneerd door Hans Paijmans. Het geeft een goed beeld van hoe open source, GNU en Linux zijn ontstaan. Er zijn enkele toevoegingen gemaakt.

1.3.1 Unix

Linux is voor alles een Unix en Unix is van oudsher een systeem voor en van universiteiten. En als zodanig is het stevig verankerd in de eeuwenoude wetenschappelijke traditie van open communicatie en uitwisseling. In deze omgeving worden systemen en ideeën niet, of niet alleen, ontwikkeld om er geld mee te verdienen, maar eerst en vooral om aanzien te verwerven binnen de eigen groep: de peer recognition. Zowel Unix als Linux zijn voortgekomen uit een zeer sterke en constante drang in de mens: de wens voor vol aangezien te worden door je collega's. En zolang de aard van het beestje niet verandert, is het voortbestaan van Linux hierdoor waarschijnlijk beter verzekerd dan dat van andere computersystemen, die afhankelijk zijn van de altijd grillige economische marktwerking. Unix werd aan het einde van de zestiger, begin zeventiger jaren ontwikkeld als een "klein" operating system dat in staat moest zijn op de meest uiteenlopende platforms meerdere gebruikers tegelijkertijd verschillende taken uit te laten voeren. Vanaf het begin werd de source (of broncode) ter beschikking gesteld aan iedereen die er maar aan wilde sleutelen. Dat gebeurde natuurlijk vooral aan de universiteiten en onderzoeksinstituten, en het is geen overdrijving om te zeggen dat het uiteindelijke Unix het resultaat was van honderden computerdeskundigen en hackers. (Hackers zijn begaafde computergebruikers en -programmeurs. De populaire pers verwacht ze ten onrechte met "crackers", ofwel mensen die inbreken in computersystemen. Maar de populaire pers weet nu eenmaal zelden waar zij het over heeft...)

Het is kenmerkend voor deze anarchistische omgeving van wetenschappers en studenten dat er al snel een ware cultuur van humor en sterke verhalen om dit operating system ontstond en het kwam dan ook voor velen als een klap dat in het begin van de tachtiger jaren Unix opeens werd gecommercialiseerd. Het was afgelopen met het inspecteren van de source, het experimenteren met alternatieve algoritmes. Computers waren "big business" geworden en de sources van programmatuur werden even jaloers geheimgehouden als het recept van Coca Cola. Het logische gevolg was dat de softwarebedrijven steeds meer een wurggreep kregen op hun klanten: elke wijziging in de code, elke aanpassing aan een programma moest duur worden betaald, zonder dat de klant enig zicht had op wat zich achter de schermen afspeelde.

Microsoft en Bill Gates zijn de voortbrengselen van deze commerciële cultuur. Programmatuur was alleen nog maar een van de vele manieren om geld te maken en alles werd ondergeschikt gemaakt aan de winst.

De gebruiker was volledig afhankelijk van de grillen van de producent, want hij had geen keus. Alleen had Bill Gates begrepen wat de eigenaars van Unix nooit hadden ingezien: dat je beter duizend systeempjes van honderd dollar kon verkopen dan vijf van tienduizend dollar. De bezitters van Unix hadden dat niet door en de commercialisering van Unix leidde in korte tijd tot de versplintering van dit operating system in evenveel peperdure varianten als er hardwareplatforms voor bestonden. En ondertussen haalde Microsoft de markt van de huiscomputer naar zich toe.

1.3.2 Het GNU copyleft

Deze commercialisering van wat in zijn ogen vrij zou moeten zijn voor iedere gebruiker stootte een zekere Richard Stallman zozeer tegen de borst dat hij besloot actie te ondernemen. Met de typisch Amerikaanse mengeling van naïviteit en dadendrang richtte hij in 1984 de "Free Software Foundation", de FSF op. Het essentiële verschil tussen de producten van de FSF en die van de commerciële bedrijven was nadrukkelijk *niet* dat de FSF-software per definitie gratis zou moeten zijn. In tegenstelling tot wat veel mensen denken wordt er over het verkopen van de software niets bepaald in de legale definitie van het "GNU copyleft" (een parodie op copyright), zoals het in de wandelgangen wordt genoemd. Maar wel is iedereen verplicht om de source van de betreffende software altijd mee te leveren en geen beperkingen te leggen op de verdere verspreiding ervan. Het woord "free" moet hier dan ook worden gezien zoals in de uitdrukking "free speech", niet "free beer". Om Richard Stallman zelf aan te halen: "Een programmeur mag best geld verdienen. Hij en zijn gezin moeten goed kunnen eten als ieder ander, maar hij hoeft dat niet elke dag in een driesterrenrestaurant te doen."

Natuurlijk werden de ideeën van Richard Stallman door de rest van de wereld weggehoond. Dat verhinderde hem en zijn geestverwanten echter niet om een groot aantal uitstekende softwarepakketten te schrijven die op praktisch elke Unix konden draaien. Vooral op universiteiten gooiden de systeemadministrators zo snel mogelijk de commerciële versies van hun respectievelijke Unixen overboord om daarvoor in de plaats de GNU-utility's te kunnen draaien. En dat gebeurde niet uit het oogpunt van kostenbesparing, want de commerciële applicaties waren immers al betaald, maar omdat de source open was en door honderden programmeurs over de hele wereld kon worden bekeken en verbeterd. Daardoor stak de GNU-software met kop en schouders boven de andere programmatuur uit.

1.3.3 Linux

Niettemin zag het er in het begin van de negentiger jaren slecht uit voor Unix. De massamarkt was door Microsoft veroverd en de hoge prijs van de verschillende Unixen verhinderde dat er serieus aan een Unix voor pc's werd gewerkt. Alleen Microsoft had geëxperimenteerd met een eigen Unix voor pc's, Xenix geheten, en die was als SCO Unix een eigen leven gaan leiden. Unix op de pc kon dus wel, alleen kostte een volledige installatie toen nog (omgerekend) zeventuizend euro en dat kon natuurlijk geen enkele thuisgebruiker opbrengen.

In 1991 veranderde dat dramatisch. Een Finse student, Linus Torvalds, ergerde zich aan het feit dat zijn favoriete besturingsstelsel, Unix, niet thuis op zijn pc kon draaien. Toen hij hoorde van Minix, een door Unix geïnspireerd OS gemaakt door Andy Tanenbaum op de Vrije Universiteit in Amsterdam, was hij direct geïnteresseerd. Hij miste echter functionaliteit, die Tanenbaum niet wilde toevoegen. Het moest klein blijven om het te kunnen gebruiken als onderwijsmateriaal. Hierop besloot Linus zelf aan de slag te gaan. Dat was echter niet zijn grootste verdienste. Het geniale was dat hij het op universiteiten toen al lang populaire internet gebruikte om medestanders te vinden die samen met hem aan een Unix voor pc's wilden werken en dat wilden uitbrengen onder het GNU copyleft. Het befaamde usenetbericht van Linus uit 1991 begon als volgt: "Hello everybody out there using minix - I'm doing a (free) operating system (just a hobby, won't be big and professional like gnu) for 386(486) AT clones." We moeten niet vergeten dat op een typisch Linux-systeem met honderden megabytes aan "vrije" programmatuur, slechts een miniem gedeelte van vijf- à zeshonderd kilobytes (de kernel) echt "Linux" is. De rest bestaat uit software die door anderen in het kader van het GNU copyleft is geschreven en aan de rest van de wereld ter beschikking gesteld. En daarin ligt de grote verdienste van Linus Torvalds: het bij elkaar brengen van al deze bestaande stukjes tot een coherent besturingsstelsel.

Hiermee is Unix eindelijk weer terug waar het thuishoort: in de wereld van openheid, onderwijs en experiment, van samenwerking en peer recognition. En het is hoopgevend dat deze onbetaalde aandring om erkend te worden binnen het wereldje van hackers, programmeurs en internet erin is geslaagd waar miljoenenfirma's als IBM, Sun, Novell en al die andere hebben gefaald: het vormen van een serieuze bedreiging voor het gigantische Microsoft.

Ongeacht wat de toekomst brengt, één ding is zeker. Met Linux is de geest van vrije software definitief uit de fles en niemand zal hem er ooit weer in terug kunnen stoppen. De GNU-filosofie van open source, die iedereen kan bekijken en die door iedereen kan worden aangepast en verbeterd, bestaat nu al 20 jaar en anders dan de commerciële bovenbazen graag beweren: zij heeft niet tot versnippering geleid. In tegendeel: de GNU-software is verreweg de beste software die er bestaat.

1.4 Andere Unices

Wat waarschijnlijk wel duidelijk is geworden, is dat Unix belangrijk is geweest in de geschiedenis van besturingssystemen. Vele OS'en zijn ervan afgeleid of erdoor geïnspireerd. Unix is oorspronkelijk ontwikkeld in Bell Labs in 1969, door (onder anderen) Ken Thompson en Dennis Ritchie. De broncode van het besturingsstelsel werd aan universiteiten geleverd. Gebruikers konden hierdoor gemakkelijk zelf aanpassingen maken. Er ontstonden een hoop verschillende varianten van het stelsel, die Bell Labs tussen 1977 en 1982 tot een enkel stelsel combineerde. Hierdoor ontstond UNIX System III. Er werden steeds meer mogelijkheden toegevoegd, wat leidde tot UNIX System V.2, ook wel bekend als SysV.

Een andere belangrijke tak is die van BSD, de Berkeley Software Distribution van de Universiteit van Californië, Berkeley. Afgeleid van BSD zijn FreeBSD (<http://www.freebsd.org>), NetBSD (<http://www.netbsd.org>), OpenBSD (<http://www.openbsd.org>), DragonFly BSD (<http://www.dragonflybsd.org/main/>) en Darwin (<http://developer.apple.com/darwin/>), wat de basis vormt voor Mac OS X.

1.5 Sterke en zwakke punten

Sterke punten van Linux en open-sourcesoftware zijn:

- Een Linux-distributie is voor geen of weinig geld (legaal) te verkrijgen.
- Hiermee samenhangend: er is veel open-sourcesoftware beschikbaar, u aangeboden door een wereldwijd netwerk van programmeurs. Deze software is vaak kwalitatief beter dan commerciële.
- Linux is leuk om te gebruiken en de mogelijkheden zijn vrijwel onbeperkt. De grootste beperking vormt de gebruiker vaak zelf.
- Linux is stabiel. De meest voorkomende reden van vastlopers is hardwareproblemen. Er zijn veel mensen die het maanden of zelfs jaren achter elkaar draaien zonder een enkele reboot.
- Er bestaan geen echte virussen of andere malware voor Linux. Een reëler gevaar vormen crackers (mensen die inbreken in computersystemen).
- Het is prima te gebruiken op een server. Het is goed in multitasking, biedt geavanceerde firewallmogelijkheden en er is software voor webservers tot nieuws- en FTP-servers.

Sommige Linux-gebruikers zien Linux als hét ultieme OS. Ik zal je maar gauw uit die droom helpen: Linux kent ook wel zijn zwakke punten. Zoals eigenlijk met alles, is er geen OS dat voor iedereen en voor alle doeleinden perfect is.

- Wat de gemiddelde pc-gebruiker lastig vindt, wordt door veel technische mensen (die vaak achter open-sourcesoftware staan) zwaar onderschat.

- Een besturingssysteem als Linux echt goed leren kennen kost wel wat tijd en inspanning.
- Softwarefabrikanten richten hun software vaak op wat de meeste klanten zullen gebruiken, wat vandaag de dag nog altijd Windows is. Erg veel spellen zijn er bijv. niet voor Linux.
- Voor hardwarefabrikanten geldt hetzelfde. Het kan zijn dat ze alleen drivers voor Windows maken. Of er drivers voor Linux zijn, hangt er dan vanaf of er specificaties beschikbaar zijn van de hardware of dat iemand de moeite heeft genomen de hardware te gaan reverse-engineeren (meestal geen eenvoudige klus).
- Het enorme aanbod aan open-sourcesoftware heeft als keerzijde dat je wel eens door de bomen het bos niet meer zou kunnen zien.
- De snelle ontwikkeling heeft ook een keerzijde: informatie en kennis die nu van toepassing is, kan over een jaar al behoorlijk achterhaald zijn.

1.6 Waar Linux gebruikt wordt

- Door mensen in technische vakgebieden en educatieve instellingen.
- Door steeds meer overheden vanwege de veiligheid, controleerbaarheid en kosten.
- Door Google, de populairste zoekmachine van het moment.
- In film- en animatiestudio's zoals Disney, DreamWorks en Pixar.
- Voor servers, o.a. bij internetproviders en webhosters.
- Voor embedded systemen.
- Door een groeiend aantal bedrijven en thuisgebruikers.

Open-sourcesoftware die erg veel gebruikt wordt:

- Apache, een webserver (software), wordt gebruikt op een goed deel van alle webserver (hardware).
- MySQL en PHP, een database en een scripttaal, worden samen veel gebruikt voor dynamische websites.
- Sendmail, een mailserver, wordt gebruikt op vele mailservers.
- BIND (Berkeley Internet Name Daemon) wordt gebruikt op het merendeel van alle DNS-servers.

Deze software is naast onder Linux ook onder verschillende andere OS'en te gebruiken.

1.7 Tux de pinguïn

De officiële mascotte en het logo van Linux is Tux de pinguïn, ontworpen door Larry Ewing in 1996 voor een logocompetitie. Linus vond dat het een pinguïn moest worden. De inzending van Larry won. Je komt hem overal tegen, soms enigszins verkleed voor de gelegenheid.

1.8 Vrije software vs. open source

Vrije software slaat op het recht van de gebruiker om de software te gebruiken, kopiëren, verspreiden, bestuderen, veranderen en verbeteren. Vrije (Engels: free) software wordt vaak verward met "gratis". De kostprijs is echter een natuurlijk gevolg. De volgende uitspraak van Richard Stallman wordt vaak aangehaald om het duidelijk te maken: "Free as in speech, not as in beer." Stallman en zijn geestverwanten zien vrije software als een sociaal proces: de wereld verbeteren door het verspreiden van vrije software. Software die niet vrij is, zien ze als suboptimaal. Dit is de visie van de FSF, de Free Software Foundation, opgericht door Stallman.

Er is een andere groep die de open-sourcebeweging genoemd wordt. Een van de bekendste personen die hiertoe behoort is Eric S. Raymond, die het Open Source Initiative (OSI) heeft opgericht. Zij hebben geen ethische bezwaren tegen software die niet open source is. Door dit verschil in visie keuren FSF en OSI andere soorten licenties goed. (Maar dit maakt uiteraard niets uit voor de geldigheid van de licentie.)

Meer informatie is te vinden op de websites van GNU en FSF (<http://www.gnu.org>) en OSI (<http://www.opensource.org>).

De term "open source" wordt vaak gebruikt om aan te geven dat de broncode van de software vrijelijk beschikbaar is. Het hoeft niet per se te slaan op de open-sourcebeweging.

1.9 Licenties

Er zijn verschillende licenties die gebruikt kunnen worden voor open-sourcesoftware. De meest gebruikte zijn GPL (General Public License) en LGPL (Lesser General Public License) van GNU en de BSD- en MIT-licenties.

De GPL-licentie bepaalt dat iedereen de software mag gebruiken (ook voor commerciële toepassingen) en het zonder restricties verder verspreid mag worden. Het is bij verdere verspreiding wel verplicht de broncode te leveren. Bijzonder aan de GPL-licentie is dat het verplicht stelt dat software waar GPL-software in gebruikt wordt, ook onder de GPL-licentie wordt uitgebracht. Als gevolg moet de broncode er ook van worden vrijgegeven. Op die manier komt het ter beschikking voor de maatschappij. Sommigen, vooral makers van propriëtaire software, zien dit echter als het "infecteren" van de nieuwe software en zullen iets wat onder de GPL-licentie valt niet snel gebruiken. Alle GNU-software en Linux zijn voorbeelden van software die onder de GPL-licentie verspreid worden.

LGPL is een iets "lichtere" variant. Het stelt niet verplicht dat software waar LGPL-software in gebruikt wordt ook onder de LGPL-licentie wordt uitgebracht. Wel verplicht het wijzigingen in de LGPL-delen vrij te geven. LGPL was oorspronkelijk bedoeld voor library's, omdat die bij uitsteking door veel andere software gebruikt worden.

De BSD-licentie is heel eenvoudig. Het vereist dat je de naam van de auteur vermeldt als je de software gebruikt in een ander product. Alle vormen van gebruik en verspreiding zijn zonder meer toegestaan. De besturingssystemen FreeBSD en OpenBSD en de webserver Apache dragen deze licentie. De MIT-licentie is vergelijkbaar. Die vereist enkel dat een verklaring wordt opgenomen waarin alle aansprakelijkheid van de auteur voor schade, in welke vorm dan ook, wordt afgewezen.

2 Het vinden van informatie

Dit hoofdstuk geeft een aantal mogelijkheden die je kunt aflopen als je op zoek bent naar informatie over een bepaald Linux-gerelateerd onderwerp.

2.1 Linux Documentation Project

De hoofdpagina van TLDP, The Linux Documentation Project, is <http://www.tldp.org>. (Er zijn vele mirrors, dus er zijn nog wel meer URL's voor.) Je kunt er o.a. HOWTO's, guides en manual pages op vinden. Veel informatie die op het TLDP te vinden is, kan ook op de harde schijf geïnstalleerd worden.

2.2 HOWTO

In een HOWTO staat informatie over een specifiek onderwerp. Zoek je bijv. informatie over fonts, kijk dan eens of je een font HOWTO kunt vinden. Let wel op de datum van de documenten! Sommige zijn jaren oud. De informatie kan dan achterhaald en niet meer van toepassing zijn.

Installeer je de HOWTO's, dan worden ze geplaatst in `/usr/share/doc/HOWTO/` of (bij distributies die zich niet zo goed aan de Filesystem Hierarchy Standard houden) in `/usr/doc/HOWTO`. Het zijn eenvoudige tekstbestanden.

2.3 Manual pages

Een manual page bevat volledige informatie over een bepaald programma, bestand, libraryfunctie e.d. De manual pages zijn ingedeeld in acht secties, te weten:

1. Executable programs or shell commands
2. System calls
3. Library calls
4. Special files
5. File formats and conventions
6. Games
7. Miscellaneous
8. System administration commands

Het is redelijk standaard dat de manual pages geïnstalleerd worden. Deze bestanden bevatten een bepaalde opmaak. Bekijk ze met het programma `man`. Wil je bijv. weten hoe het configuratiebestand `/etc/crontab` in elkaar zit, typ dan **man 5 crontab**. Als je geen onderdeel opgeeft, dan krijg je de manual page met het laagste onderdeelnummer te zien. Voor verdere informatie over het gebruik van manual pages typ **man man**.

2.4 Info pages

Deze zijn vergelijkbaar met de manual pages, maar de informatie is vaak meer up-to-date. Mits geïnstalleerd, zijn de info pages te vinden in `/usr/share/info` of `/usr/info`. Gebruik `info` om ze te bekijken. Zo toont **info cp** de info page van `cp`. Hoe je `info` gebruikt, staat in **info info**.

2.5 /usr/share/doc

Hier vind je de documentatie van geïnstalleerde software. Heb je bijvoorbeeld inkscape geïnstalleerd, dan staat de bijbehorende documentatie in /usr/share/doc/inkscape. Is het daar niet te vinden, probeer dan /usr/doc.

2.6 Website van de distributie

Vaak is op de website van de distributie informatie te vinden. Voor informatie over distributiespecifieke zaken is dit de aangewezen plaats om te kijken. Gentoo heeft werkelijk uitstekende online documentatie. De delen die niet specifiek zijn voor Gentoo, kun je ook gebruiken voor een andere distributie. Vaak is er ook wel een forum waar gebruikers vragen kunnen stellen.

2.7 Nieuwsgroepen en forums

Als je de benodigde informatie niet kunt vinden of als je er echt geen hout van begrijpt, kun je ook terecht bij nieuwsgroepen over Linux, zoals nl.comp.os.linux.overig. De hele nl.comp.os.linux-boom staat bekend als NCOL. Daarnaast zijn er ook allerlei webforums, o.a. op de websites van sommige distributies. Let bij het posten van berichten op de volgende punten.

1. Gebruik platte tekst en geen HTML (in de meeste software in te stellen). HTML neemt meer ruimte in beslag en niet alle usenetsoftware ondersteunt het.
2. Neem als onderwerp een korte beschrijving van het probleem en niet iets als "HELP!!!!".
3. Zet een reply onder de quote (aangehaalde tekst) en quote niet meer dan nodig is.
4. Vraag niet om het antwoord via e-mail te verzenden, maar controleer regelmatig de nieuwsgroep. Op die manier kunnen andere mensen ook nog wat opsteken van jouw problemen.

Je kunt tussen oude nieuwsberichten rondzoeken op <http://groups.google.com>. Wellicht heeft iemand al eens een interessant bericht gepost.

2.8 Het web

Informatie die niet direct in een van de vorige bronnen te vinden is, kan nog wel op andere locaties op het web staan. Raadpleeg Google. Op <http://www.google.com/linux> kun je specifiek zoeken op Linux-gerelateerde sites. De online encyclopedie Wikipedia (<http://www.wikipedia.org>) bevat ook veel nuttige (achtergrond)informatie. Het meeste is te vinden op de Engelstalige pagina's.

2.9 Boeken en tijdschriften

Er zijn veel algemene boeken over Linux geschreven. Ik zou niet kunnen zeggen welke daarvan goed zijn. Over specifiekere onderwerpen, zoals Bash of Sendmail heeft O'Reilly verschillende goede boeken uitgebracht. Je herkent ze direct aan hun witte voorkant met het een of ander beest erop.

Een tijdschrift is een leuke manier om wat nieuwtjes en tips op te steken. Er zijn verschillende Linux-specifieke tijdschriften in boeken- en tijdschriftenwinkels te vinden, zoals Linux magazine (<http://www.linuxmag.nl>) en Linux Format (<http://www.linuxformat.co.uk>).

3 Hardwareondersteuning

Linux is oorspronkelijk ontwikkeld voor het x86-platform, IBM-compatible pc's. Linus dacht eerst dat het niet erg portable zou worden door de assemblercode (erg hardware-specifiek) die hij gebruikte in de kernel. Het tegendeel is echter waar gebleken: het is naar vele hardwareplatformen geport. Het is o.a. ook te gebruiken op computers van Apple, Sun en Playstations.

De rest van dit hoofdstuk richt zich op IBM-compatible pc's, aangezien dat de meest voorkomende hardware is. Een aantal onderdelen, zoals de processor, geheugen of optische drives (zoals dvd-branders), zal geen problemen opleveren. Er worden enkele aandachtspunten gegeven. Voor de meeste hardware is een device driver in de kernel nodig. De videokaart is een uitzondering: die moet X ondersteunen.

3.1 Lijsten

Op websites van distributies staat vaak een lijst met ondersteunde hardware. Daar staat vaak niet echt alle ondersteunde hardware op, maar wel de meest voorkomende. Staat een stuk hardware dat je hebt niet op zo'n lijst, geef het dan niet meteen op. Het is wel een makkelijke manier om een overzicht te krijgen.

Het omgekeerde is de Linux Incompatibility List op <http://leenooks.com> met een overzicht van hardware die helemaal niet of niet volledig onder Linux werkt. Informatie over Linux op laptops is te vinden op <http://linux-laptop.net>.

3.2 Videokaart

Elke videokaart zal in de console werken en kan sowieso met de Vesa-driver werken onder X. Zie <http://wiki.x.org/wiki/Projects/Drivers> voor een overzicht van hardware die echt ondersteund wordt. Hierbij is het belangrijk op welke chipset je videokaart is gebaseerd, niet welk merk of type videokaart je precies hebt. De marktleiders van videochipsets zijn NVIDIA en ATI. Beide bedrijven bieden binary drivers aan naast de beschikbare open-sourcedrivers. Zie ook paragraaf 13.2.

3.3 Geluidskaart

Tot de kernelreeks 2.4.x werd gebruikgemaakt van OSS, het Open Sound System. Sinds kernel 2.6.x is ALSA (Advanced Linux Sound Architecture) echter de standaard geworden. Omdat sommige software alleen met OSS overweg kan, is er nog wel OSS-emulatie voor ALSA. Zie <http://www.alsa-project.org> voor kaarten die door ALSA ondersteund worden.

3.4 Printer en scanner

Een mooi overzicht van wat wel en niet werkt en welke driver nodig is, is te vinden op <http://www.linuxprinting.org>. Wat je nodig hebt om je scanner te gebruiken is SANE (Scanner Access Now Easy). Zie <http://www.sane-project.org> voor een overzicht van ondersteunde scanners.

3.5 Tv-kaart

De meeste analoge tv-kaarten worden wel ondersteund. Het deel van de kernel dat dit soort kaarten ondersteunt, gebruikt een API die bekend staat als V4L of video4linux. Je kunt ook digitale tv ontvangen met

je pc als je een DVB-kaart hebt (Digital Video Board). Ondersteuning voor DVB-kaarten zit in de kernels 2.6.x inbegrepen. De website van het project is <http://www.linuxtv.org>.

3.6 Digitale camera

Ondersteuning voor vele camera's kan verkregen worden met de library libgphoto2. Dat werkt in user-space, zodat er geen extra kernel driver voor nodig is. Een lijst met ondersteunde camera's is te vinden op <http://www.gphoto.org/proj/libgphoto2/support.php>. Staat je camera niet in de lijst, dan kan het alsnog mogelijk zijn de data te benaderen door de camera als USB mass storage device te gebruiken (waarbij je de geheugenkaart net als een USB-stick benadert). Ook kun je de geheugenkaart van de camera altijd nog met behulp van een kaartlezer uitlezen.

4 Distributies

In het begin moest je Linux (de kernel), de GNU-utility's en andere software zelf compileren om een systeem op te bouwen. Men moest dus eigenlijk een compleet besturingssysteem in elkaar zetten uit losse onderdelen. Het is daarvoor nodig een al werkend systeem te hebben waarop je dit kunt doen. Om het allemaal wat eenvoudig te maken, zijn er distributies gemaakt.

4.1 Verschillen

Wat alle distributies gemeen hebben, is dat ze de Linux-kernel en de gebruikelijke open-sourcesoftware bevatten. Hoe alles precies in elkaar zit, verschilt per distributie. Welke filosofie de makers van de distributie erop nahouden kun je vaak op hun website terugvinden. Als twee uitersten zijn sommige distributies gericht op eenvoud van gebruik, terwijl andere zijn gericht op eenvoud van systeemopbouw. Het laatste zorgt ervoor dat configuratie en aanpassen met de hand juist eenvoudiger is. Een voorbeeld van het eerste soort is Ubuntu, van het tweede soort Slackware.

Verder verschillen de installatieprocedure en het pakkeetsysteem dat gebruikt wordt om software te installeren, updaten en verwijderen. Als twee grote groepen heb je RPM en DEB. RPM was oorspronkelijk het pakkeetsysteem van Red Hat en DEB dat van Debian. De meeste distributies gebruiken een van beide. Bij Slackware wordt een heel simpel pakkeetsysteem gebruikt.

4.2 Enkele bekende

De bekendste distributies zijn:

- Slackware: <http://www.slackware.com>
- Gentoo: <http://www.gentoo.org>
- Fedora: <http://fedoraproject.org>
- Mandriva: <http://www.mandrivalinux.com>
- SUSE Linux: <http://www.opensuse.org>
- Debian: <http://www.nl.debian.org>
- Ubuntu: <http://www.ubuntulinux.org>

Een goed overzicht is te vinden op <http://www.distrowatch.com>. Daar is meer informatie te vinden over de verschillende distributies.

Ik kan niet vertellen welke distributie het "beste" is en ik heb ook niet met elke distributie even veel ervaring. Ik kan wel wat algemene informatie geven over enkele ervan. Slackware en Gentoo zijn het meest geschikt voor ervaren gebruikers of mensen die echt in het diepe willen duiken. Fedora, Mandriva, SUSE en Ubuntu zijn er meer op gericht alles "out of the box" te laten werken. Vooral Ubuntu lijkt erg populair op het moment. Debian staat ergens tussen beide in.

Slackware is de oudste van allemaal en wordt onderhouden door Patrick Volkerding. Het is in al die jaren weinig veranderd. Het installatieprogramma is nog altijd tekstgebaseerd en er zijn weinig voorzieningen om configuratie eenvoudiger te maken. Je zult alles met de hand moeten doen. Het voordeel is overzichtelijkheid.

De grootste commerciële distributies zijn Red Hat, SUSE en Mandriva. Red Hat is wellicht de bekendste distributie van allemaal. Het bedrijf werd in 1995 opgericht door Robert Young en Marc Ewing. In het verleden was Red Hat gericht op thuisgebruikers en gratis te downloaden, maar tegenwoordig is hun distributie

(RHEL, Red Hat Enterprise Linux) gericht op het bedrijfsleven. Voor de thuisgebruiker is er eind 2004 het door Red Hat gesponsorde Fedora gekomen.

De situatie van SUSE is vergelijkbaar. Dit Duitse bedrijf is eind 2003 opgekocht door Novell en richt hun SUSE Linux Enterprise op bedrijven. Een aparte tak, openSUSE, is te gebruiken voor thuisgebruikers.

Mandriva (voorheen Mandrake) is begonnen als verbeterde versie van Red Hat. De distributie is ook commercieel, maar ook enige tijd na uitgave vrijelijk te downloaden. De naamsverandering is gekomen nadat ze met Conectiva (een andere distributie) zijn samengegaan.

Debian is vernoemd naar de oprichter en zijn vrouw: Ian en Deborah en wordt onderhouden door een team vrijwilligers. Het pakkeetsysteem van Debian verdient een vermelding. Het was het eerste systeem dat echt goed met afhankelijkheden tussen pakketten omging. Debian bestaat altijd uit drie verschillende takken: stable, testing en unstable. Ieder draagt een codenaam van een figuur uit Disneys animatiefilm Toy Story. Stable bevat alleen software die lange tijd getest is. Het is altijd behoorlijk verouderd, maar wel geschikt voor servers. Wat de meeste mensen willen is testing. Daar zit software in die nog getest wordt, maar dezelfde versies worden over het algemeen zonder meer toegepast in andere distributies. Unstable, ten slotte, wil je waarschijnlijk niet. Software uit deze tak kan wel eens niet goed werken.

Ubuntu is een vrij nieuwe distributie die snel is opgekomen in populariteit. Het is gebaseerd op het systeem van Debian. Ubuntu wordt ook door een groep vrijwilligers gemaakt en gesponsord door het Afrikaanse bedrijf Canonical Software, opgericht door ondernemer en multimiljonair Mark Shuttleworth. Het motto van Ubuntu is dat software vrijelijk en gratis beschikbaar moet zijn voor iedereen. Het kan gedownload worden en er zijn gratis cd's te verkrijgen. Een cd met Ubuntu kan zelfs naar je toegezonden worden, waar je ook woont en zonder verzendkosten. Er komt elke 6 maanden een nieuwe versie uit. Dat is dezelfde periode als die aangehouden wordt voor GNOME, de standaard desktopomgeving van Ubuntu. (Je kunt ook andere desktopomgevingen achteraf installeren of een van de andere variant gebruiken, zoals Kubuntu voor KDE.)

Gentoo is ook vrij nieuw en geniet ook een zekere populariteit, zij het dan waarschijnlijk bij een ander publiek. Bijzonder is dat het source-gebaseerd is. Dat wil zeggen dat je geen voorgecompileerde pakketten installeert, maar de broncode downloadt en die zelf compileert. Dit doe je met een systeem genaamd Portage, wat geïnspireerd is door het systeem van BSD. Gentoo biedt vooral grote flexibiliteit. Je moet vrij veel zelf met de hand doen, maar de uitstekende documentatie helpt je er goed doorheen. Daardoor is het erg geschikt voor iemand die beter wil begrijpen hoe zijn Linux-systeem in elkaar zit. Het grootste nadeel van Gentoo is dat het compileren van software meer tijd in beslag neemt dan het installeren van voorgecompileerde pakketten.

4.3 Live-cd's

Naast de normale distributies die je op de harde schijf moet installeren, zijn er ook wat live-cd's: Linux-systemen die je kunt opstarten vanaf cd. Er is geen installatie voor nodig. Een bekende is Knoppix (<http://www.knopper.net/knoppix/index-en.html>), gebaseerd op Debian. Knoppix gebruikt KDE als desktopomgeving. Er is ook nog Gnopix (<http://www.gnopix.org>) met GNOME.

4.4 Verkrijgen

Je kunt commerciële distributies in de winkel aanschaffen. Sommige commerciële en alle niet-commerciële zijn ook te downloaden of er ergens op goedkope cd/dvd te krijgen. Zoals eerder vermeld kun je een cd van Ubuntu op hun site bestellen om kosteloos toegezonden te krijgen.

Als je installatie-cd's downloadt, zijn ze vaak in ISO-formaat. Een ISO-bestand is een image of archief. Het is even groot als de bestanden die het bevat. Het voordeel van een ISO is dat het eenvoudiger te downloaden is dan alle bestanden afzonderlijk. Bovendien zouden zaken als permissies en eigenaren van bestanden niet behouden blijven als je alle bestanden afzonderlijk downloadt onder Windows, met alle gevolgen van dien. De meeste brandsoftware heeft wel ondersteuning voor ISO's. Let erop dat je aangeeft het als image te branden.

Als je gewoon het ISO-bestand op de cd zet, zul je na afloop op de cd ook alleen het grote ISO-bestand aantreffen.

4.5 Je aanmelden als Linux-gebruiker

Er kan niet via licenties worden bijgehouden hoeveel Linux-gebruikers er op de wereld zijn. De mensen van <http://counter.li.org> proberen het aantal te schatten. Ga je Linux gebruiken, meld je dan ook even aan. Je kunt zowel jezelf als je machines aanmelden. Ook wordt er gevraagd naar je woongebied, zodat de spreiding kan worden onderzocht.

5 Partities

5.1 Soorten

Omdat men het handig vond om een harde schijf in meerdere stukken te kunnen opsplitsen, zijn partities verzonnen. Een partitie is een stuk van je harde schijf. Er zijn verschillende redenen te bedenken waarom je partities zou willen gebruiken. Je kunt het bijvoorbeeld handig vinden om verschillende data op verschillende partities te plaatsen. Als je verschillende besturingssystemen, zoals Windows en Linux, naast elkaar wilt gebruiken, dan zul je ieder op een eigen partitie moeten zetten. Meerdere besturingssystemen op één computer hebben heet ook wel dual-booten.

Doordat men ooit heeft verzonnen dat vier partities wel genoeg zou zijn, kun je maar vier zogenaamde primaire partities aanmaken. Veel besturingssystemen kunnen alleen vanaf een primaire partitie opstarten. Linux is een van de weinige besturingssystemen die je ook kunt opstarten vanaf een logische partitie (zie verderop).

Er is altijd één primaire partitie actief (zo wordt het genoemd in fdisk van DOS) of bootable (volgens fdisk van Linux). Vanaf die partitie wordt het OS opgestart. Een bootmanager is een programma waarmee je bij het opstarten kunt kiezen welk besturingssysteem je wilt gebruiken. Je kunt de bootmanager aan het begin van een partitie zetten die bootable is. Eenvoudiger is echter het in de MBR (Master Boot Record, het eerste stuk van de harde schijf) te zetten. Dan maakt het niet meer uit welke partitie bootable is. De bootmanager in de MBR kan eventueel weer andere bootmanagers op andere partities starten. Bootmanagers die veel gebruikt worden, zijn LILO (Linux Loader) en het nieuwere GRUB. Je gebruikt ze om tijdens het booten te kunnen kiezen of je Linux of Windows wilt starten (als beide geïnstalleerd zijn). Het is ook nodig om uit verschillende kernelversies te kunnen kiezen.

Vier partities is geen luxe. Vaak wil je er meer aanmaken. Daarop heeft men het volgende bedacht. In plaats van een primaire partitie maak je een uitgebreide (Engels: extended) partitie. Je kunt dus in totaal maximaal vier primaire of uitgebreide partities aanmaken. Binnen een uitgebreide partitie kun je andere partities aanmaken. Zo'n partitie binnenin een uitgebreide partitie heet een logische (Engels: logical) partitie.

Onder DOS en Windows hebben alle partities (die herkend worden) een andere letter, stations genaamd. Partities op harde schijven beginnen vanaf C. In Linux gaat dit heel anders. Alle harde schijven en partities worden daar weergegeven als bestand onder /dev (van device). Voor harde schijven van het type IDE geldt:

Bestand	Omschrijving
/dev/hda	primaire master
/dev/hdb	primaire slave
/dev/hdc	secundaire master
/dev/hdd	secundaire slave

Er zijn meestal twee kabels op het moederbord waar harde schijven (en andere IDE-apparatuur) op kunnen worden aangesloten. De ene is de primaire kabel, de andere de secundaire. Op elke kabel kunnen weer twee apparaten worden aangesloten. De ene is de master, de andere de slave. Wat waar op zit aangesloten, kun je waarschijnlijk in je BIOS zien tijdens het opstarten.

Voor SCSI-schijven heb je sd in de naam in plaats van hd: /dev/sda, /dev/sdb, etc. De meeste mensen zullen geen SCSI hebben. Moderne moederborden hebben wel weer een ander (nieuw) type aansluiting: SATA (Serial ATA). Je kunt het duidelijk herkennen aan de veel smallere datakabels die naar de schijven lopen. Voor SATA-schijven werd bij kernel 2.4.x ook hda etc. gebruikt. Vanaf kernel 2.6.x is het echter sda etc. (dus identiek aan SCSI) geworden.

Deze devices staan voor de complete harde schijf. De partities die op hda staan heten hda1, hda2, etc. Om te zien welke partities werkelijk bestaan, kun je onder Linux **fdisk -l /dev/hda** gebruiken. Partities binnen een uitgebreide partitie worden genummerd vanaf de 5. Zo is de tweede logische partitie (binnen een

uitgebreide partitie) op hda hda6. De primaire of uitgebreide partities zijn genummerd van 1 tot en met 4. (Meer is toch niet mogelijk.)

5.2 Bestandssystemen

Partities moeten worden geformatteerd voordat je er bestanden op kunt opslaan. Daarbij wordt er een bestandssysteem op aangemaakt. Tijdens de installatieprocedure van de distributie zal waarschijnlijk wel gevraagd worden om nieuw aangemaakte partities voor Linux te formatteren. Je kunt het in Linux zelf doen met `/sbin/mkfs`. Dat is een frontend voor verschillende programma's in `/sbin` genaamd `mkfs.type`, waarbij `type` het bestandssysteem is dat het kan aanmaken. Standaard wordt er `ext2` genomen, maar met `mkfs -t type` kun je een andere kiezen. Je kunt ook direct `mkfs.type` gebruiken. Alleen voor Reiser is er een afwijkend commando: `mkreiserfs`.

Er bestaan vele verschillende bestandssystemen. De volgende tabel toont een aantal veelvoorkomende en het besturingssysteem waarbij het gebruikt wordt.

Besturingssysteem	Bestandssysteem
Linux	ext2, ext3 of Reiser, swap
MS-DOS	FAT16
Windows 9x	FAT16 of FAT32
Windows NT en nieuwer	NTFS

Ext2 is het klassieke bestandssysteem voor Linux. Ext3 is identiek aan ext2, maar dan met een zogenaamd journaal erbij. Het idee van een journaal is dat bij elke verandering aan het bestandssysteem eerst in het journaal geschreven wordt wat er gaat gebeuren. Daarna wordt het echt gedaan en ten slotte wordt de entry uit het journaal gehaald. Als het systeem uitvalt voordat de verandering voltooid is (bijvoorbeeld door een stroomstoring), dan staat er nog altijd in het journaal wat er moet gebeuren. Die actie kan alsnog worden uitgevoerd als het systeem weer opstart. Dit gaat sneller dan het moeten uitvoeren van een controle door `e2fsck` (het equivalent van `scandisk` onder Windows), waardoor een betere uptime te verzekeren is. Dit is bijv. goed voor web servers. Een ander voordeel is dat het niet nodig is het bestandssysteem om de zoveel tijd te laten controleren. Bij ext2 gebeurt dat standaard om de 180 dagen of na 29 keer reboorten (welke er eerder komt). De controle van grote bestandssystemen kan enige tijd in beslag nemen. Ext3 is daarom boven ext2 aan te raden voor de meeste gebruikers.

Reiser is net als ext3 een nieuwer bestandssysteem met een journaal. Het zou efficiënter moeten zijn als je veel kleine bestanden hebt, maar ik heb er zelf weinig ervaring mee.

5.3 Partitioneren

Heb je op dit moment Windows geïnstalleerd, dan is de kans groot dat er slechts één grote partitie op je harde schijf aanwezig is, de C in Windows. Een veelgemaakte denkfout is: "ik heb veel ruimte over op C, dus daar kan ik Linux wel op installeren". Die vrije ruimte is ruimte op de Windows-partitie en is niet beschikbaar om Linux-partities in aan te maken. Je moet er op een of andere manier voor zorgen dat er echt ruimte vrijkomt op de harde schijf, bijvoorbeeld door de schijfvullende partitie voor Windows te verkleinen.

Partitioneren kan met een aantal verschillende programma's:

- Je gebruikt `fdisk`. Er is een `fdisk` van DOS en een van Linux. Voor beide geldt dat alle data op de partities die je gaat veranderen verloren zal gaan. Natuurlijk doet dit niet ter zake als je net een nieuwe harde schijf hebt. En als je een goede backup hebt van alles, dan kun je die naderhand weer terugzetten. Bij sommige distributies kun je `fdisk` voor Linux ook tijdens de installatie gebruiken. De `fdisk` van

Linux is anders dan die van DOS. Het is wat veiliger omdat wat je verandert pas werkelijk wordt uitgevoerd als je dat opgeeft. Verder kent de Linux-versie veel meer verschillende bestandssystemen.

- Je gebruikt FIPS (<http://bmrc.berkeley.edu/people/chaffee/fips/fips.html>). Je kunt hiermee vanuit DOS een bestaande FAT16- of FAT32-partitie verkleinen. Voordat je FIPS gaat gebruiken, moet je je Windows-partitie defragmenteren. De data van de partitie die je gaat verkleinen, zal behouden blijven.
- Ntfsresize (<http://linux-ntfs.sf.net/info/ntfsresize.html>) is een dergelijke tool voor NTFS-partities. Deze tool is voor Linux. Ook hiermee raak je je gegevens niet kwijt. Vooraf defragmenteren van de NTFS-partitie is niet nodig.
- Je gebruikt een speciaal programma waarmee je de partities op eenvoudige wijze en zonder gegevensverlies kunt veranderen onder DOS of Windows. Een bekend programma is PowerQuest Partition Magic (commercieel).
- GNU parted (<http://www.gnu.org/software/parted/>) is een goed partitioneringsprogramma voor Linux. Het kan partities met een ext2-, ext3-, Reiser-, swap-, FAT16- of FAT32-bestandssysteem (net als Partition Magic) van grootte veranderen. Ondersteuning voor het resizen van NTFS heeft het niet. Parted is een command-lineprogramma.
- GParted (<http://gparted.sf.net>) en QtParted (<http://qtparted.sf.net>) zijn hier grafische frontends voor, resp. voor KDE en GNOME. Beide maken naast van GNU parted ook gebruik van Ntfsresize om NTFS-partities te kunnen resizen. De tools lijken sterk op Partition Magic.

Het spreekt voor zich dat het maken van een backup een goed idee is voordat je met partities gaat knutselen.

Je zou de Linux-tools hierboven kunnen gebruiken vanaf een live-cd of installatie-cd waar ze op aanwezig zijn. De installatieprocedure is verschillend per distributie. Het verschilt dan ook wat er tijdens de installatie mogelijk is om te (her)partitioneren. De hele harde schijf opnieuw partitioneren (en eventueel bestaande data overschrijven) is altijd mogelijk en de eenvoudigste optie. Het verkleinen van FAT-partities kan ook al langer en is daarom wel in alle installatiesoftware verwerkt. Je hebt waarschijnlijk een NTFS-partitie als je Windows 2000 of nieuwer gebruikt. Of het verkleinen daarvan mogelijk is zal verschillen, maar ondertussen ook wel in de meeste installaties verwerkt zijn.

5.4 Partities voor Linux

Voor Linux is het aan te raden op zijn minst twee partities te maken: een met het ext2-bestandssysteem (of ext3 of Reiser) waarop software wordt geïnstalleerd en een swap-partitie.

De swap-partitie wordt gebruikt als virtueel geheugen, geheugen op de harde schijf. Dat is veel langzamer dan het echte geheugen, maar kan wel veel groter zijn. Data die lange tijd niet meer gebruikt is, wordt verhuisd naar de swap. Ook wordt het gebruikt als er meer geheugen nodig is dan beschikbaar. Hoe groot de swap-partitie moet zijn, ligt eraan hoe groot je gewone geheugen is en wat je van plan bent te gaan doen. De totale hoeveelheid geheugen die voor programma's beschikbaar is, is het echte geheugen + de swap. Dus als je 256 MB aan RAM hebt en 256 MB aan swap, dan is er in totaal 512 MB beschikbaar voor programma's.

Als je naast de swap-partitie slechts één grote partitie met het ext2-, ext3- of Reiser-bestandssysteem maakt, dan moet die grote partitie gemount worden onder / (de root-directory). Op deze zogenaamde root-partitie komen alle bestanden voor Linux te staan. Veel mensen vinden het prettig meerdere partities aan te maken. Voor al die partities moet je een mount point opgeven. Als je bijv. /usr opgeeft, dan komen alle bestanden die je in /usr zet op die partitie terecht en niet op de root-partitie. Om het goed te kunnen indelen, moet je wel weten hoeveel ruimte er voor welke directory nodig gaat zijn. Daar kom je zelf wel na een tijdje achter, maar als je net begint, heb je nog geen idee. Daarom volgt er hier een overzicht van directory's die handig kunnen zijn om op een aparte partitie te plaatsen en hoeveel ruimte er typisch voor nodig is.

- /home** Hier komen alle persoonlijke bestanden van gebruikers, zoals persoonlijke configuratiebestanden, documenten, muziek, films, etc. Gebruik hier alle ruimte voor die over is na het indelen van de andere partities.
- /usr** Hier wordt veruit de meeste data in gezet, zoals niet-essentiële programma's (dat wil zeggen: niet vitaal voor het überhaupt kunnen opstarten of om heel elementaire dingen te kunnen doen), library's en documentatie. Hoeveel ruimte je hiervoor nodig hebt, verschilt enorm met wat voor software je wilt installeren. Je kunt wel aannemen dat een systeem met veel desktopsoftware nog niet snel de 10 GB zal overschrijden. Ik heb nu een Ubuntu-systeem met zowel GNOME als KDE met 5 GB in deze directory.
- /usr/local** Alles dat niet tot de distributie behoort, hoort hier te worden geplaatst om alles overzichtelijk te houden. Als je later wilt upgraden of van distributie wilt wisselen, kun je deze partitie laten staan om zo al je eigen toevoegingen te behouden. Ook hier is de grootte volledig afhankelijk van hoeveel software je wilt installeren. Denk je dat je later nog zelf software toe wilt voegen, dan zal iets van 2 GB waarschijnlijk wel voldoende zijn.
- /var** Hier staan bestanden onder die vaak veranderen, zoals spool voor printer en mail, logbestanden, locks, tijdelijke bestanden en opgemaakte manual pages (wordt gemaakt als je er een opvraagt). Ben je van plan een nieuws- of e-mailserver te installeren, dan kun je deze directory op een aparte partitie plaatsen. Daarmee voorkom je dan dat de root-partitie volloopt. Bij Gentoo zal deze directory groter worden dan normaal, omdat alle software hier wordt gecompileerd.
- /boot** Hier staat voornamelijk de kernel in. Hier is niet zo gek veel ruimte voor nodig. Een enkele kernel is misschien hooguit 2 MB groot.
- / De root-partitie. Hier komt dus alles op te staan dat niet op een andere partitie komt. Als je de bovenstaande directory's (met name /home en /usr) aanhoudt om op aparte partities te zetten, zal veruit de meestse data op andere partities komen te staan. Enkele GB's is hier dan ruim voldoende voor. Gebruik je alleen een grote root-partitie, dan is voor een desktopsysteem iets van 10 GB wel voldoende.

6 Basiskennis

6.1 Gebruikers

Tijdens de installatie van de meeste distributies wordt gevraagd om een normale gebruiker aan te maken naast root. Als het installatieprogramma van jouw distributie dat niet vraagt, moet je dat zelf naderhand doen. De gebruiker root, ook wel superuser genoemd, mag alles doen. Hij mag alles bekijken en veranderen. Daarom is het ook gevaarlijk om de hele tijd als root te werken. Je kunt o.a. gemakkelijk iets fout doen. Maak voor jezelf een normale gebruiker aan met:

- `useradd jarkko`
- `passwd jarkko`

Verander hierin `jarkko` door de gebruikersnaam. `Useradd` is een script om het proces van een nieuwe gebruiker aanmaken te automatiseren. Als `useradd` niet bestaat, probeer dan eens `adduser`. Zowel GNOME als KDE hebben ook grafische tooltjes om gebruikers toe te voegen. Wat ook altijd werkt, is het met de hand doen. Maak daarvoor een home directory aan in `/home` voor de nieuwe gebruiker. In die directory mag de gebruiker al zijn configuratie- en andere persoonlijke bestanden zetten. Het is gebruikelijk (en overzichtelijk) de home directory hetzelfde te noemen als de loginnaam. (Om de directory te mogen aanmaken, zul je root moeten worden met `su`, zoals verderop wordt uitgelegd.) Voeg vervolgens een regel toe aan `/etc/passwd` en eventueel `/etc/group`. Wat scripts als `useradd` daarnaast nog doen is de inhoud van `/etc/skel` kopiëren naar de nieuwe home directory. Zodra de gegevens zijn toegevoegd aan `passwd` en eventueel `group`, moet de home directory ook eigendom gemaakt worden van deze gebruiker. Dit kan met `chown`, wat wordt uitgelegd in paragraaf 6.6.

Regels in `/etc/passwd` zitten als volgt in elkaar:

```
jarkko:x:500:500:Jarkko Huijts:/home/jarkko:/bin/bash
```

Van links naar rechts staat er de gebruikersnaam, zijn geëncrypteerde wachtwoord, de UID (user-ID), de GID (group-ID), een veld met informatie over de gebruiker (meestal zijn volledige naam), zijn home directory en ten slotte zijn shell. De populairste shell is Bash. Naast Bash bestaan er o.a. `ash`, `bsh`, `tcsh`, `csh` en `zsh`. De shell moet in `/etc/shells` staan, anders mag de gebruiker niet inloggen. Alle moderne distributies maken gebruik van shadow passwords. Daarbij staat het wachtwoord in `/etc/shadow` (een bestand dat naast root door niemand gelezen mag worden) en staat in `/etc/passwd` enkel `x` op de plaats van het wachtwoord. Om het wachtwoord van een gebruiker te veranderen, gebruik je **`passwd gebruiker`**. (Als je geen gebruiker opgeeft, verander je je eigen wachtwoord.) Een normale gebruiker kan alleen zijn eigen wachtwoord veranderen. Root kan alle wachtwoorden wijzigen.

De user-ID en group-ID zijn nummers die de gebruiker resp. de groep waar hij toe behoort aangeven. Gebruik een UID (en GID) van 500 of hoger als je er zelf een wilt kiezen. De GID's kun je terugvinden in `/etc/group`. Wil je een nieuwe gebruiker tot een bestaande groep laten behoren, dan kun je de optie `-g group` gebruiken bij `useradd`. Vervang hierin `group` door de GID of naam van de groep. (Wil je de gebruiker nog bij extra groepen laten behoren, dan moet je ook de optie `-G group2,group3` gebruiken. Scheid de groepen met komma's.)

Een regel in `/etc/group` zit als volgt in elkaar:

```
jarkko:x:505:jarkko
```

Van links naar rechts staat er de naam van de groep, een wachtwoord voor de groep (meestal geen), de GID (group-ID) en achteraan een lijst met gebruikers die tot die groep behoren. Zijn er meerdere gebruikers, dan

moet je een komma tussen de namen zetten. Met dit veld kun je een gebruiker bij meerdere groepen laten horen.

Log altijd in als normale gebruiker i.p.v. root. Je moet wel root zijn als je software wilt installeren of bestanden buiten je home directory wilt aanpassen. Je kunt van normale gebruiker naar root overgaan met het commando **su** (substitute user). Je moet dan vanzelfsprekend het root-wachtwoord opgeven. In plaats van **su** kun je soms beter **su -** gebruiken, omdat daarmee ook het PATH voor root wordt gezet overeenkomstig met wat er in `/etc/login.defs` gedefinieerd staat. Dat kan nodig zijn om `/sbin` en `/usr/sbin` meteen in het PATH te hebben staan. (Daar staan binary's in die alleen voor root bestemd zijn.) Je kunt achter **su** ook een gebruikersnaam zetten om een andere gebruiker te worden. Stop met root (of een andere gebruiker) zijn met het commando **exit** of `ctrl-d`. Dat commando moet ook gebruikt worden om een shell die binnenin een andere shell is gestart te verlaten. Een shell binnen de huidige shell starten kan door bijv. **bash** in te tikken.

Ubuntu is op dit punt afwijkend. Daar is er standaard geen wachtwoord voor root ingesteld, zodat je niet kunt inloggen als root of **su** kunt gebruiken om het te worden. Commando's die root moet uitvoeren, moeten vooraf gegaan worden door **sudo**. Daarbij wordt om een wachtwoord gevraagd, maar dat is het wachtwoord van de huidige gebruiker. Zie ook paragraaf 11.7.

Grafische software die handelingen uitvoert die alleen door root gedaan mogen worden (zoals een programma om pakketten met software te installeren), moeten worden gestart als root. Dat kan door in een terminalvenster root te worden (met **su**) en vervolgens het programma te starten op de command line. Bij GNOME en KDE zit hiervoor ook een klein tooltje dat je op de desktop grafisch om het root-wachtwoord vraagt als je een dergelijk programma probeert te starten, of je eigen wachtwoord als Sudo gebruikt wordt.

6.2 Consoles en terminals

De basisinterface van Linux is de console: een shell in tekstmode waar je commando's in kunt geven, zoals bij DOS. Als je bent ingelogd in de console, krijg je meestal Bash als shell. De meeste distributies starten tegenwoordig al bij het booten een grafische inlogprompt (display manager genaamd), waarna je een desktop onder X krijgt voorgeschoteld.

Je kunt onder meerdere consoles tegelijk inloggen. Gebruik `alt +` een functietoets om van console te wisselen. Je kunt ook met `alt +` de pijltjes naar links en rechts door de consoles wandelen. Met `alt-F7` kom je onder X als dat gestart is. Als je vanuit X naar een console terug wil, moet je naast `alt` ook de `ctrl`-toets indrukken.

Onder X kun je ook een shell in een window krijgen door een terminalemulator te starten. De traditionele is `xterm`, maar er zijn verschillende modernere. In alle terminals krijg je exact dezelfde shell als in de console.

Met **exit** of **logout** sluit je de huidige shell af. Je kunt ook `ctrl-d` intikken. Als je in de console zit, zul je hierdoor uitloggen. Doe je het in een terminal in X, dan zal alleen die terminal sluiten.

Is X nog niet gestart tijdens het booten, dan kan dat vanuit de console met **startx**. Meer hierover kun je vinden in hoofdstuk 13. Of er bij het booten al een display manager wordt gestart ligt aan de standaard runlevel waartoe geboot wordt (zie ook paragraaf 13.6.2).

Wil je de computer uitschakelen, voer dan in de console **shutdown -h now** uit. Heb je X gestart met **startx**, sluit dat dan eerst af. Vervang de **h** (van halt) door **r** om te rebooten. Shutdown mag alleen door root worden uitgevoerd. Bij de meeste distributies wordt de toetsencombinatie `ctrl-alt-del` opgevangen. Druk je die toetsen in, dan wordt shutdown uitgevoerd. Met andere woorden: je kunt `ctr-alt-del` gebruiken om het systeem af te sluiten. Hierbij maakt het niet als welke gebruiker je bent ingelogd. (Je kunt de uit te voeren actie ook veranderen in `/etc/inittab`.) Zit je in X, dan zal deze zelfde toetsencombinatie eerst door X opgevangen worden en sluit je X op een abrupte manier af. Ben je ingelogd met een display manager, log dan uit van je desktop en kies afsluiten/herstarten in de display manager.

6.3 Werken met consoles en terminals

Om jezelf heel wat typewerk te besparen, is er een ontzettend handige toets voor Bash, namelijk de TAB-toets. Probeer maar eens de eerste letters van een bestandsnaam in te tikken en op TAB te drukken. Dan zal de rest van de naam automatisch worden aangevuld! Zijn er meerdere mogelijkheden, dan volgt een overzicht door nogmaals op TAB te drukken. Naast bestandsnamen kan Bash ook aliasnamen, hostnamen (na een @) en namen van variabelen (na een \$) afmaken. In andere software werkt de TAB-toets soms ook om iets af te maken. Verder kun je alle vorige commando's opvragen met de pijltjes omhoog/omlaag.

Wat niet met Bash te maken heeft, maar werkt in elke VT100-compatible terminal zijn de volgende toetsen. De console emuleert een VT100, terminal emulators voor X de VT220. Maar ook daar werken dezelfde toetsen.

Toets	Functie
ctrl-c	breek huidig proces af
ctrl-d	voer EOF-karakter (end of file) in, sluit een shell af
ctrl-l	refresh het scherm
ctrl-a	naar begin van de regel
ctrl-b	karakter naar links (zelfde als pijltje naar links)
ctrl-d	delete het karakter onder de cursor (zelfde als delete)
ctrl-e	naar einde van de regel
ctrl-f	karakter naar rechts (zelfde als pijltje naar rechts)
ctrl-h	delete het karakter voor de cursor (zelfde als backspace)
ctrl-k	delete alles van de cursor tot het einde van de regel
ctrl-t	verwissel het karakter onder de cursor met dat ervoor
ctrl-u	delete alles van de cursor tot het begin van de regel
ctrl-w	delete het vorige woord
ctrl-s	bevries de output
ctrl-q	laat de output weer doorgaan
ctrl-v	karakter dat je hierna invoert wordt ingevoerd als escape code
ctrl-y	voeg in wat je het laatste verwijderd hebt

Als je per ongeluk op ctrl-s druk, lijkt het alsof alles vast zit. Je kunt dan simpelweg op ctrl-q drukken om alles weer door te laten gaan.

6.4 Basiscommando's

De meest gebruikelijke commando's hebben allemaal slechts 2 letters:

Commando	Functie
ls	directoryinhoud bekijken (list)
cd	van directory veranderen (change directory)
rm	bestand verwijderen (remove)
cp	bestand kopiëren (copy)
mv	bestand verplaatsen of hernoemen (move)
ln	link aanmaken (link)
du	bestandsgrootte opvragen (disk usage)
df	vrije schijfruimte opvragen (disk free)

Om te zien welke opties je achter deze commando's kunt zetten, kun je `--help` achter het commando zetten. Je ziet dan twee rijen met mogelijke opties. De linker kolom bevat enkele letters met een liggend streepje ervoor, de rechter kolom bevat woorden met twee liggende streepjes ervoor. Beide opties doen hetzelfde.

De ene is korter, maar bij de andere is het gemakkelijker om te zien wat het doet. Dit idee kom je bij veel GNU-software tegen. De korte versie is volgens de POSIX-standaard, de lange versie is toegevoegd bij de GNU-software.

Veel gebruikte opties voor **ls** zijn: **-l** voor meer informatie en **-a** voor het bekijken van verborgen bestanden en directory's. Alle bestanden en directory's die beginnen met een punt zijn in Linux verborgen. En laten nou toevallig alle configtiebestanden en directory's met configuratiebestanden in je home directory met een punt beginnen. Daardoor zie je ze normaal gesproken niet, wat wel zo overzichtelijk is. Een andere zeer handige optie is **--color=auto** om een verschil te kunnen zien tussen een normaal bestand, directory, symlink etc. (Zie ook paragraaf 20.18.) Met **ls -lR** krijg je een compleet overzicht van alle bestanden in de huidige en alle subdirectory's.

Bestandsnamen worden als strings, een reeks karakters, gezien. Je kunt zonder problemen meerdere punten in een bestandsnaam zetten. Spaties in de bestandsnaam zijn ook toegestaan. Een bestandsnaam kan maximaal 256 tekens lang zijn. Doordat bestandsnamen als strings worden gezien, is het niet nodig om **ls *.*** te gebruiken om alles te zien te krijgen. Het zal zelfs niet het gewenste effect opleveren. Dan worden namelijk alleen die bestanden getoond waar een punt in de naam voorkomt. Met **ls *** worden alle bestanden in de huidige en een directory "dieper" getoond. Het *-teken geeft een reeks tekens van willekeurige lengte aan. Naast * bestaat er ook ?, wat staat voor één enkel karakter. De tekens * en ? worden wildcards genoemd. Met **ls a*** worden alle bestanden getoond waarvan de naam begint met een a.

Wildcards (* en ?) worden niet door de programma's ls, mv, cp en mv begrepen, maar worden door de shell verwerkt. Die vervangt een uitdrukking met een wildcard erin door alle bestandsnamen die eraan voldoen en geeft dat door aan het programma. Naast wildcards kent de shell (Bash in ieder geval) ook enkele simpele reguliere expressies:

- [str] voor elk karakter in de string str
- [a-b] voor elk karakter tussen a en b

Voorbeelden:

- **ls a[abc]** toont de bestanden aa, ab en ac indien ze bestaan
- **ls a[0-9]** toont de bestanden a0, a1 tot en met a9 indien ze bestaan

Linux is, net als alle andere Unices, hoofdlettergevoelig. Hierdoor is file een ander bestand dan File, terwijl dat onder Windows hetzelfde bestand zou zijn. Verder is het belangrijk op te merken dat in Linux en andere Unices / gebruikt wordt waar in DOS en Windows \ gebruikt wordt.

Om een directory terug te gaan moet je **cd ..** gebruiken en niet **cd..**, dus met een spatie tussen cd en de puntjes. Bij sommige distributies is echter een alias gemaakt zodat het laatste ook werkt. (Waarschijnlijk voor mensen die DOS gewend zijn.) Je kunt snel teruggaan naar je home directory met **cd ~** of zelfs alleen maar **cd**. Je kunt in alle commando's je home directory aangeven met een tilde (~). Met **cd -** ga je naar de laatst bezochte directory terug. Als je niet zeker weet in welke directory je op dit moment zit, kun je **pwd** (print working directory) gebruiken.

Bij **rm** kun je de optie **-f** gebruiken als je geforceerd bestanden wilt verwijderen (je hoeft dan geen bevestiging te geven) en **-r** om directory's compleet (recursief) te verwijderen. Deze twee kun je combineren tot **rm -rf dirnaam** om een complete directory weg te halen zonder enige bevestiging. Je hoeft dus niet twee aparte streepjes te gebruiken (**-r -f** is niet nodig).

Met **cp -r vanlocatie naarlocatie** kun je een hele directory (recursief) kopiëren.

Met **mv** kun je zelfs hele directory's verplaatsen naar waar je maar wil. Het is dus veel krachtiger dan ren en move onder DOS.

Een symlink (of softlink) kun je maken met **ln -s vanlocatie naarlocatie**. Wordt deze symlink benadert, dan wordt eigenlijk de data uit het bestand waar het naartoe wijst benadert. Je kunt ook een symlink naar een directory maken. Je moet een symlink zien als een doorverwijzer naar een ander bestand (of directory). Zonder de optie **-s** wordt een hard link aangemaakt. Door een hard link naar een bestand aan te maken, zijn er twee bestandsnamen gekoppeld aan dezelfde data. Het is alleen mogelijk een hard link te maken naar een ander bestand op dezelfde partitie en een hard link naar een directory maken is onmogelijk. Delete je een bestand waarvan meerdere hard links bestaan, dan blijft de data benaderbaar via de andere bestandsnaam of -namen. Pas als alle bestanden die naar dezelfde data verwijzen zijn verwijderd, is de data gedeleteet. Delete je een symlink, dan is de data nog benaderbaar via het bestand waar de symlink naar verwees. Delete je het bestand waar een symlink naar verwijst, dan wijst de symlink naar iets dat niet meer bestaat. In het overzicht van **ls -l** kun je zien waar een symlink naartoe wijst. Het eerste getal dat je ziet in de uitvoer van dit commando laat zien hoeveel hard links er naar het bestand bestaan. Gebruik je **ls -l symlinknaam** en deze symlink verwijst naar een directory, dan zul je een overzicht krijgen van de bestanden in de directory waarnaar verwezen wordt. Wil je alleen zien naar welke directory de symlink wijst, dan moet je de optie **-d** gebruiken bij **ls**, dus **ls -ld symlinknaam**.

Met **du directorynaam** wordt van alle subdirectory's de grootte getoond. Met de optie **-s** krijg je het totaal te zien. Met de optie **-h** is het formaat human readable, d.w.z. in kB, MB of GB aangegeven.

Met **df** krijg je een overzicht van alle gemounte partities en hoeveel ruimte er gebruikt c.q. vrij is. Ook hier bestaat de optie **-h**. Dit commando is samen met **mount** ook handig om informatie over partities te verkrijgen.

Achter commando's als **cp** en **mv** kun je meerdere bestandsnamen zetten: **mv file1 file2 tmp** om **file1** en **file2** beide naar de directory **tmp** te verplaatsen. Je kunt hierbij uiteraard ook wildcards gebruiken: **mv *.ext tmp**.

Als een bestandsnaam spaties of speciale tekens zoals een (bevat, zul je aanhalingstekens om de naam moeten zetten. Alternatief kun je deze karakters een voor een "escapen" door er direct vóór een backslash (\) op te nemen. Zo kun je de grootte van een bestand genaamd "rare (bestandsnaam)" opvragen met een van de volgende commando's:

- **du rare \ (bestandsnaam \)**
- **du "rare (bestandsnaam)"**

Dit is nodig omdat sommige karakters een speciale betekenis hebben in Bash en het anders niet eenduidig is wat je bedoelt. Alle karakters met een speciale betekenis zijn: \$ # & ? * < > [] () ' en |. Verder wordt een spatie normaal gesproken gezien als scheiding tussen bestandsnamen. Het is meestal het makkelijkste lastigere bestandsnamen automatisch te laten afmaken met TAB.

Begint de bestandsnaam met een minteken, dan krijg je ook problemen, omdat het zal worden aangezien als optie. Je moet voor dergelijke bestandsnamen **--** zetten om aan te geven dat wat erna komt een bestandsnaam is en niet als optie gezien moet worden. Alternatief kun je **./** voor het streepje aan het begin van de bestandsnaam zetten. De punt daarin geeft de huidige directory aan (net zoals in DOS). Op die manier laat je de bestandsnaam niet met een streepje beginnen en heb je ook geen problemen. Een voorbeeld: **rm -- -file** of **rm ./-file**.

6.5 Pipelines, redirecting en backquotes

Met **|** kun je de output van het ene programma laten gebruiken als invoer voor het volgende. Dit heet een pipeline. Voorbeeld: **ps aux | grep firefox**. Het eerste commando (**ps aux**) geeft als output alle processen die op dit moment draaien. Deze output dient als input voor **grep firefox**, die ervoor zorgt dat alleen de regels waar **firefox** in voorkomt overblijven. De invoer van een programma noem je **stdin**, de uitvoer **stdout**. Er is ook nog **stderr**, de error output. Die laatste wordt gebruikt voor foutmeldingen. Beide worden hetzelfde getoond in de console/terminal, maar je kunt ze wel apart filteren bij het redirecten.

Met `>` kun je de output van een commando naar een bestand laten redirecten. Zo zorgt `echo hoi > testfile` ervoor dat de output van `echo hoi` (de tekst "hoi") in het bestand `testfile` wordt gezet. Bestaat dat bestand al, dan wordt het overschreven! De output krijg je bij het redirecten niet op het beeld te zien. Je kunt van ongewenste output afkomen door het te redirecten naar `/dev/null`. Dan verdwijnt het in de digitale prullenbak. De `stderr` wordt normaal gesproken niet meegenomen bij het redirecten. Wil je die ook meenemen, gebruik dan `&> file` of `> file 2>&1`.

Met `>>` kun je ook output naar een bestand redirecten, maar in dit geval wordt het bestand niet overschreven, maar wordt de output achteraan in het bestand toegevoegd (append).

Met `<` wordt wat erna komt als input gebruikt voor het programma dat ervoor komt.

Met `<<` kun je zelf input intikken totdat je een bepaald woord intikt. Voorbeeld: `cat > test << stop`. Je kunt na dit commando gegeven te hebben van alles intikken, totdat je op een lege regel alleen `stop` intikt. Dan wordt de ingetikte tekst (zonder het woord `stop`) in het bestand `test` gezet. Waarschijnlijk ga je dit niet zo vaak gebruiken.

Ten slotte nog iets handigs dat in ieder geval in Bash werkt: backquotes (ook wel backticks genoemd). Een backquote is de quote die links naast de 1 op je toetsenbord zit. De output van wat tussen backquotes staat, wordt letterlijk zo ingevuld op die plaats. Een voorbeeld: `vim `which startx``. Het commando `which startx` geeft de exacte locatie (complete pad + bestandsnaam) van het script `startx` als output. Dat wordt letterlijk ingevuld achter `vim`, dus zo open je het script `startx`, zonder dat je precies weet waar het staat.

6.6 Permissies en eigenaren

Als je `ls -l` intikt, dan zul een regel als de volgende zien:

```
-rw-rw-r-- 1 jarkko jarkko 2786 Aug 8 20:19 index.html
```

Van links naar rechts staan er de permissies, het aantal hard links naar het bestand, de eigenaar van het bestand, de groep waar hij toe behoort, de grootte in bytes, de datum, de tijd en ten slotte de bestandsnaam. Bij de datum zie je de tijd als het bestand in het huidige jaar gemaakt is (zoals in het voorbeeld hier) of anders een jaartal.

De permissies zitten als volgt in elkaar. Helemaal links staat een letter die het type van het object aangeeft.

Letter	Omschrijving
-	bestand
b	block device, komt voor in /dev
c	character device, komt voor in /dev
d	directory
l	symlink
p	named pipe
s	socket

Daarachter staan drie blokken met ieder drie tekens waar dezelfde letters in kunnen staan, variërend van `---` tot `rxw`. De `r` staat voor read, de `w` voor write en de `x` voor execute. Het eerste blok geldt voor de eigenaar van het bestand, het tweede voor de groep en het derde voor de rest van de wereld. Symlinks hebben altijd `lrwxrwxrwx`. Read, write en execute spreken voor zich bij gewone bestanden, maar bij directory's heeft het een ander effect. Het volgende overzicht geeft aan wat je kunt met welke permissies.

Permissie	Bestand	Directory
---	niks	onbenaderbaar
r--	inhoud bekijken	bestandsnamen bekijken
rw-	inhoud zien en wijzigen	inhoud zien en wijzigen
rwx	alles (incl. uitvoeren script/binary)	alles (incl. naar directory cd'en)
r-x	bekijken, script uitvoeren	kunt niks verwijderen/toevoegen, wel zien en cd'en
--x	binary uitvoeren	kunt binary uitvoeren als je de exacte locatie kent

Merk op dat je voor het uitvoeren van een binary alleen maar execute nodig hebt, terwijl voor het uitvoeren van een script zowel execute als read nodig is. Dat komt doordat een script door een interpreter gehaald moet worden en die moet de tekst van het script kunnen inlezen.

Om de permissies te veranderen, is het commando **chmod**. Zet daar eerst achter voor wie de verandering geldt, dan of het aan of uit moet en ten slotte wat er moet veranderen. Een **u** staat voor user, **g** voor group en **o** voor other (de rest van de wereld). De volgende twee voorbeelden maken het waarschijnlijk meteen duidelijk.

- **chmod ugo-rwx file**: Haal voor User, Group en Others Read Write en eXecute weg.
- **chmod u+x file**: Maak het voor User eXecutable.

In plaats van letters kun je ook cijfers gebruiken. Bij andere Unices is dit vaak ook de enige optie. Om het getal te bepalen dat je moet hebben, moet je de drie posities waar rwx kan staan zien als drie bits (het zijn ook eigenlijk drie bits, die zo letterlijk in het bestandssysteem staan opgeslagen). Zet dit binaire getal om in een octaal getal. In binair stelt de meest rechter bit $2^0 = 1$ voor. De bit er links naast $2^1 = 2$ en die daarnaast $2^2 = 4$. Als alle permissiebits aan staan, dus rwx, levert dit $4+2+1=7$ op. Hier volgt weer een tweetal voorbeelden ter verduidelijking.

- **chmod 777 file**: Geef iedereen read-, write- en execute-permissie.
- **chmod 750 file**: Geef de eigenaar alle rechten, de rest van de groep alleen read- en execute-permissie en de rest niks.

Naast de permissies zijn er nog drie bits die je per bestand in kunt stellen: SUID, SGID en de sticky bit. Even iets vooraf: Er wordt niet uitgelegd wat er gebeurt als de execute bit niet is gezet samen met een van deze bits. Je kunt de sticky bit voor bestanden zetten en SUID en SGID voor directory's. Al deze dingen worden zelden gedaan. Wat er in die gevallen gebeurt, hangt af van welke Unix-variant je precies gebruikt.

De sticky bit wordt normaal alleen gebruikt voor directory's. Is deze bit gezet, dan mogen bestanden toegevoegd worden door iedereen die er schrijfrechten heeft, maar niet iedereen die schrijfrechten heeft, mag zomaar bestanden hernoemen of verwijderen (wat normaal wel zou mogen). Dat mag dan alleen als de gebruiker ook de eigenaar is van het bestand. De sticky bit wordt typisch gezet voor /tmp. De sticky bit, SUID en SGID zijn verbonden aan execute. In de output van **ls -l** verandert de laatste x in een t als de sticky bit is gezet. De sticky bit zet je aan met **chmod +t file** of door een extra 1 voor de permissiewaarde te zetten (bijv. 1777).

Als je de SUID-bit (waarbij de S staat voor saved) zet voor een binary of script, dan verander je de effectieve UID (user-ID) van alle gebruikers die het uitvoeren. Het effect is hetzelfde als wanneer de eigenaar van het programma het uitvoerde. Is root de eigenaar, dan is het dus net alsof root het uitvoerde en het programma heeft daarmee dezelfde rechten als root. Op die manier kun je alle gebruikers programma's laten uitvoeren die normaal gesproken alleen door root uitgevoerd mogen worden. Dit wordt SUID-root genoemd.

De SUID-bit kun je zetten door een extra 4 voor de permissiewaarde te zetten. Hiermee samen hangt de SGID-bit. Is die gezet, dan is de effectieve groep de groep die eigenaar is van het bestand. Je kunt de SGID-bit zetten door een extra 2 voor de permissiewaarde te zetten. Met **chmod +s file** kun je de SUID- en SGID-bit zetten (als je **u+s** gebruikt zet je SUID alleen, met **g+s** SGID alleen en zonder aanduiding voor wie het moet gelden beide tegelijk). Een alternatief voor SUID-root is Sudo gebruiken. Zie paragraaf 11.7 voor meer daarover.

Net zoals bij de bits voor read, write en execute kun je de waarden van de sticky bit, SUID en SGID bij elkaar optellen. Zo kun je de sticky bit en SUID tegelijk zetten door een $1+4=5$ voor de permissiewaarde te zetten. Er wordt vaak een extra 0 vooraan de permissiewaarde gezet voor chmod om er zeker van te zijn dat de sticky bit, SGID en SUID weggehaald worden.

Om de eigenaar en groep te veranderen is het commando **chown**. Zet daar eerst de nieuwe eigenaar, dan een dubbele punt en dan de nieuwe groep achter. Laat voor of achter de dubbele punt de eigenaar of groep leeg om de huidige eigenaar of groep te behouden. Zo verandert **chown jarkko: file** de eigenaar van file in jarkko. Bij oudere versies van chown moest je een punt gebruiken i.p.v. een dubbele punt. De eigenaar van een bestand veranderen zorgt ervoor dat SUID en SGID weggehaald worden, tenzij root het doet.

6.7 Variabelen

Veel software kijkt naar bepaalde variabelen. Elk programma dat je start heeft omgevingsvariabelen. Je kunt in Bash alle huidige variabelen bekijken met **export**. Je kunt nieuwe zetten met **export VAR=waarde**. Om een variabele te zetten voor een enkel programma, kun je voordat je het programma start export gebruiken, maar wat ook kan is de variabele voor dat programma alleen opgeven door het te starten met **VAR=waarde programmaam**.

Variabelen worden bijgehouden per instantie van Bash. Ze gelden dus alleen voor programma's die je start onder die shell (alle zogenaamde kindprocessen van die shell). Als je een Bash-script uitvoert, wordt voor het uitvoeren daarvan een nieuwe instantie van Bash gestart. Om die reden kun je geen Bash-script maken waar export-commando's in staan voor de huidige shell.

Je kunt variabelen die je elke keer bij het starten van Bash wilt laten zetten opnemen in `~/bashrc` (gebruik complete export-commando's).

6.8 Aliases

De shell Bash kent aliases. Hiermee kun je commando's maken om minder te hoeven typen. Je kunt bijv. **alias ll='ls -l'** gebruiken om **ll** te definiëren als snelle versie van **ls -l**. Je kunt het ook gebruiken om bepaalde opties standaard te laten meegeven. Een voorbeeld is **alias df='df -h'**. Hierdoor wordt altijd de optie **-h** gebruikt bij **df**.

Je kunt aliases net als variabelen opnemen in `~/bashrc`.

6.9 Bestanden bekijken en editen

Bestanden kunnen editen is wel iets wat je snel onder de knie moet krijgen, want anders kun je geen enkel (configuratie)bestand wijzigen. Er zijn verschillende editors voor Linux. De meest uitgebreide zijn Vim en Emacs. Pico en Nano zijn enkele simpelere editors, die de beginnende gebruiker misschien meer aanspreken. Daarnaast zijn er ook vele editors voor X beschikbaar.

6.9.1 Vim

Vi kun je op bijna elk Unix-systeem vinden en tegenwoordig commercieel. Een open-sourceversie is Vim (VIsual editor iMproved), gemaakt door Nederlander Bram Molenaar. Vim vereist enige gewenning als je het nog niet kent, maar als je daar eenmaal doorheen bent, is het handig en vooral ontzettend snel in gebruik. Het is zo gemaakt dat je je vingers niet van het toetsenbord af hoeft te halen en je nooit toetsen nodig hebt die "ver weg" zitten. Dat komt vooral doordat Vi op terminals gebruikt moest kunnen worden met toetsenborden waar minder toetsen op zitten dan die gebruikelijk zijn bij een pc. Zelfs als je Vim niet wilt gebruiken voor je normale editwerk, is het erg nuttig op zijn minst een basis te leren. Op een rescue disk of een Unix-systeem kom je wellicht geen enkele andere editor tegen.

In Vim werk je in twee modes: de command mode en de insert mode. Standaard zit je in de command mode. De meeste toetsen op het toetsenbord hebben dan een functie. Zo haal je met x het karakter op de huidige cursorpositie weg. Om ex-commando's in te kunnen voeren, moet je eerst : intikken. Openen van bestanden, opslaan en de editor verlaten zijn allemaal ex-commando's. Om tekst toe te voegen moet je overschakelen naar de insert mode. Dat doe je met i. Om vervolgens weer terug te gaan naar command mode gebruik je ESC of ctrl-]. (De laatste mogelijkheid is vaak weer iets sneller in te tikken.)

De belangrijkste functies:

Toets	Functie
:q	quit
:q!	quit zonder opslaan
:w	write
:wq	write and quit (kan ook met ZZ)

De helpfunctie van Vim is erg uitgebreid. Gebruik :help of F1 om die te bekijken. Op de officiële website van Vim, <http://www.vim.org>, staat ook veel informatie, met name vele tips van gebruikers.

Naast de command-lineversie is er ook een voor X: gVim. De editor zelf werkt hetzelfde, maar je hebt er een menu bij.

6.9.2 Emacs

Emacs is een behoorlijk uitgebreide editor. Het is een tegenhanger van Vim. Mensen die Vim gebruiken zijn typisch wat anti-Emacs en vice versa. Een typische opmerking van een Vim-aanhanger: "Emacs is a great operating system. It lacks a good editor, though." Het eerste deel van die opmerking is niet eens zo gek, want je kunt er e-mail mee versturen, nieuwsgroepen, websites en manual pages mee bekijken en tetris spelen!

Emacs kent niet twee modes, zoals Vi(m). Je kunt direct beginnen tekst in te tikken. Veel commando's geef je door ctrl + een bepaalde toets in te drukken. De belangrijkste zijn:

Toets	Functie
ctrl-x, ctrl-s	opslaan
ctrl-x, ctrl-c	quit

Ook van Emacs is er een versie voor X: XEmacs.

6.9.3 Pico en Nano

Pico hoort bij Pine (een mail- en nieuwsclient). Een clone van Pico is Nano. Het werkt in grote lijnen hetzelfde als Pico, maar is nog kleiner. Het gebruik van deze editors spreekt voor zich. De toetsen staan onder in beeld.

6.9.4 Less

Dit is geen editor, maar kan wel gebruikt worden om bestanden (of willekeurige output) handig te bekijken. Gebruik **less file** of **cat file | less**. Met less kun je door lange teksten heen scrollen met de pijltjestoetsen. Naast naar beneden en naar boven scrollen, kun je ook naar links en rechts scrollen. Gebruik ctrl-v om een pagina omlaag te gaan en ctrl-u voor een pagina terug. (Of page up en page down.) Als je meerdere bestanden tegelijk opent (met **less *.txt** bijvoorbeeld), dan kun je naar het volgende bestand met :n en naar het vorige bestand met :p. Wil je een bestand editen met Vim, tik dan :v. Druk op h voor help en q om te stoppen.

Er is een soortgelijk programma genaamd more. Less is beter dan more. Met less kun je ook terugscrollen en less start sneller op doordat hij niet eerst het complete bestand (of alle output) inleest.

6.10 Directoryhiërarchie

De complete Filesystem Hierarchy Standard, een overzicht van alle directory's en hun functie, kan gevonden worden op <http://www.pathname.com/fhs/>. Hier volgt de hoofdindeling.

/bin Binary's die nodig zijn tijdens het opstarten en andere belangrijke binary's die ook door normale gebruikers gebruikt mogen worden. Hier staan alle basiscommando's zoals ls, cp, df, ln en mv in.

/boot Normaal gesproken staat hier de kernel in. Soms staat die echter ook meteen in /.

/dev Device files. Elk device (zoals harde schijven, seriële poort, parallelle poort en geluidskaart) wordt weergegeven als bestand.

/etc System-wide configuratiebestanden.

/home Bestanden van gebruikers, zoals persoonlijke configuratiebestanden en documenten. De configuratiebestanden, of soms directory's met configuratiebestanden, beginnen altijd met een punt. Bij Unices zijn dat verborgen bestanden/directory's.

/lib Belangrijke gedeelde (shared) library's en kernelmodules (in /lib/modules).

/lost+found Naamloze bestanden die gevonden worden door e2fsck worden hier gedumpt. Je hebt een directory genaamd lost+found op elke partitie.

/media Plaats voor mount points van verwisselbare media, zoals cd's.

/mnt Plaats voor mount points van andere tijdelijke bestandssystemen, zoals partities op harde schijven.

/proc De bestanden hieronder staan niet op je harde schijf, maar zijn louter fictief (al kun je wel hun grootte opvragen). Als je ze bekijkt, krijg je informatie die in het geheugen wordt opgeslagen. Het kan gebruikt worden om informatie over je systeem en lopende processen te krijgen. Verder staan er onder de subdirectory sys bestanden waarmee je direct kunt "praten" met de kernel.

/root De home directory van de systeemadministrator (superuser of root).

/sbin Systeembinary's. Deze zijn over het algemeen niet voor normale gebruikers bedoeld. Daarom staat deze directory ook niet standaard in de PATH-variabele voor normale gebruikers.

/tmp Tijdelijke (temporary) bestanden. Er staan ook tijdelijke bestanden in `/var/tmp`. Het verschil is dat de bestanden daar tussen reboots behouden moeten blijven.

/usr Staat niet voor user, maar voor Unix System Resources. Hier wordt veruit de meeste data (zoals binary's, library's en documentatie) in gezet. Deze data is niet essentieel tijdens het opstarten. Je zult hier behoorlijk wat subdirectory's onder aantreffen, o.a. weer opnieuw directory's genaamd lib, bin en sbin.

/usr/local Alles dat niet tot de distributie behoort, hoort hier te worden geplaatst om alles overzichtelijk te houden. Als je later wilt upgraden of van distributie wisselen, dan kun je deze partitie en daarmee je zelf toegevoegde software behouden.

/var Hier staan bestanden onder die vaak veranderen, zoals spools (voor printer, mail en nieuws), logbestanden en lockbestanden (worden aangemaakt om aan te geven dat een programma gestart is) en opgemaakte manual pages (gebeurt als je er een opvraagt).

6.11 Archiveren en comprimeren

Tar is een programma waarmee je bestanden kunt archiveren. Tar stopt losse bestanden bij elkaar in één groot bestand. Dat werd vooral vroeger veel gebruikt om backups te maken op tape. Als je bestanden archiveert met tar, blijven de permissies behouden als je het kopieert naar een partitie met een ander bestandssysteem (zoals FAT of NTFS).

Gzip is een programma dat een bestand comprimeert/inpakt. Dit wordt vaak gebruikt bij bestanden die niet direct gebruikt worden, maar wel bewaard moeten blijven. Bij sommige distributies worden manual pages en fonts gecomprimeerd bijgehouden. Het wordt ook gebruikt om bestanden gemakkelijker te kunnen transporteren via bijv. een website.

Om een partij bestanden in een groot gecomprimeerd bestand samen te voegen, moeten tar en gzip samen gebruikt worden. De taak van archiveren en comprimeren is daarmee gescheiden. Er ontstaat uiteindelijk een .tar.gz- of .tgz-bestand, ook wel tarball genaamd. Je kunt een complete directory archiveren en comprimeren met het volgende commando: **tar zcvf file.tar.gz directory/**

Heb je zo'n tarball gedownload, dan decomprimeer en dearchiveer je het met: **tar zxvf file.tar.gz**

Vaak wordt er dan een subdirectory aangemaakt met dezelfde naam als de tarball zonder de extensie .tar.gz. De volgorde van de opties (**zxvf**) is niet belangrijk, maar in deze volgorde tikt het het snelst. Een minteken voor de opties kan, maar is niet nodig. De **x** staat voor extract, de **c** voor create en de **v** voor verbose (laat de bestandsnamen zien). De **z** zorgt ervoor dat gzip of gunzip wordt gebruikt. Analoog hieraan kan de optie **j** gebruikt worden voor bzip2 en bunzip2. Daarmee kunnen .tar.bz2-bestanden worden in- en uitgepakt. Bzip2 kan betere compressie geven, maar het duurt langer om weer te decomprimeren.

Een oud compressieprogramma is **compress**. Daarmee ingepakte bestanden eindigen op **.Z**. Je kunt ze weer uitpakken met **uncompress file.Z**.

6.12 Mounten

Onder Linux bestaan geen letters die een station aanduiden, zoals A voor floppy en C voor de eerste Windows-partitie, maar alles hangt onder dezelfde boom. Elke partitie moet worden gemount op een bepaald punt aan deze boom (mount point). Elke partitie, cd of dvd moet worden gemount om benaderbaar te zijn. Mount points voor partities die tijdelijk gemount worden, behoren in `/mnt` (bijv. `/mnt/dosc`). Die voor verwisselbare media, zoals cd en dvd, behoren in `/media` (bijv. `/media/cdrom`).

Een Windows-partitie kan een FAT- of NTFS-bestandssysteem hebben, terwijl een Linux-partitie typisch van het type ext2, ext3 of Reiser is. Stel je wilt bij een FAT-partitie voor Windows, zeg `/dev/hda1`. Je kunt die mounten met **mount -t vfat /dev/hda1 /mnt/dosc**. (Weet je niet welke partitie je precies nodig

hebt, dan kun je een overzicht krijgen met **fdisk -l /dev/hda** als root, met i.p.v. hda je harde schijf.) Je kunt het type opgeven met de optie **-t**. Zoals je hier ziet wordt FAT vfat genoemd. Dat is de naam van een uitbreiding die sinds Windows 95 gebruikt wordt om langere bestandsnamen mogelijk te maken. NTFS heet simpelweg ntfs. Meestal kan mount het type zelf ook wel goed ontdekken.

Als je wilt dat elke keer bij het opstarten de partitie automatisch gemount wordt, kun je een regel als de volgende toevoegen aan `/etc/fstab`:

```
/dev/hda1 /mnt/dosc vfat auto,user,quiet,umask=0000,rw 0 0
```

Van links naar rechts staan er de partitie, de mount point, het bestandssysteem, opties, of de partitie gedumpt moet worden als je **dump** gebruikt en een getal dat aangeeft in welke volgorde de partities gecontroleerd moeten worden. Een 0 in de laatste kolom geeft aan dat de partitie niet gecontroleerd hoeft te worden. Belangrijk bij de opties is dat ze gescheiden worden door een komma en geen spaties bevatten. Spatiëring wordt gebruikt om de velden van elkaar te scheiden. De opties die hier staan, zijn nodig om FAT-partities voor alle gebruikers lees- en schrijfbaar te maken. Auto geeft aan dat de partitie automatisch gemount moet worden en user dat normale gebruikers de partitie mogen (un)mounten.

Je kunt regels in `fstab` opnemen voor partities die je niet automatisch wilt laten mounten, maar die wel aangeven welke opties gebruikt moeten worden en wat de mount point is. Gebruik dan de optie `noauto`. Dit is bijvoorbeeld nuttig voor een dvd-speler. Je kunt dan achter mount alleen de devicenaam of de mount point meegeven, bijv. **mount /media/dvd**. Voor alle mogelijke opties zie **man fstab**.

Een partitie weer ontkoppelen van de boom kan met **umount** (dus niet `unmount`). Geef als argument de te ontkoppelen mount point of device aan. Voor een overzicht van op dit moment gemounte partities, tik **mount** zonder argumenten. Het zal niet lukken een partitie te unmounten als een gebruiker nog in die directory aanwezig is. Dan zal een melding als "umount: /media/cdrom: device is busy" verschijnen. Ga als root en alle andere gebruikers uit de directory weg, dan zal het unmounten wel lukken. Bij problemen kan de optie `-f` van force nog wel eens uitkomst bieden.

De bestandssystemen die worden ondersteund, hangt af van de kernel. Ondersteuning voor FAT (zoals gebruikt onder oudere versies van Windows) is perfect. Die voor NTFS (ingevoerd vanaf Windows NT en ook gebruikt door Windows 2000, XP en Vista) is minder eenvoudig. Dit komt doordat het geen triviaal bestandssysteem is en compleet met reverse engineering uitgeplozen moet worden. Er is geen publieke documentatie over. Voor meer over de huidige stand van zaken rond NTFS zie <http://www.linux-ntfs.org>.

Om een cd of dvd niet elke keer te hoeven mounten en naderhand weer te unmounten zijn verschillende oplossingen gemaakt, zoals automount, supermount en submount. De basisfunctionaliteit is dat het medium vanzelf gemount wordt als het nodig is. Alle drie de oplossingen zijn inmiddels weer verouderd. Er is nu een beter alternatief beschikbaar onder de verzamelnaam project Utopia. Het project richt zich op het automatisch detecteren van hardware, met name verwisselbare media zoals USB-sticks, digitale camera's, geheugenkaarten in kaartlezers en cd/dvd's. De basisonderdelen zijn D-BUS (een communicatiesysteem voor applicaties) en HAL (Hardware Abstraction Layer). HAL biedt een interface aan naar de hardware. Het zorgt niet voor het automatisch mounten ervan. Bij GNOME doet `gnome-volume-manager` dat. Naast mounten kan het ook bepaalde programma's starten als een medium gevonden is. bijv. een mediaspeler om een dvd af te spelen of een programma om foto's op te halen van je digitale camera. KDE heeft vergelijkbare ondersteuning voor HAL (in de vorm van een `ioslave`). Een alternatief dat te gebruiken is zonder een van deze desktopomgevingen is `Ivman` (<http://ivman.sf.net>). Gebruik je HAL, dan hoef je voor verwisselbare media niet zelf regels in `fstab` op te nemen.

6.13 Geheugenbeheer

Linux zal het beschikbare geheugen zo veel mogelijk verdelen over de draaiende processen. Daardoor zal je geheugen vrijwel altijd goed gevuld zijn. Linux gebruikt het geheugen o.a. voor disk cache. D.w.z. dat er

gegevens in worden opgeslagen die ook op de harde schijf staan. Gegevens zijn veel sneller uit het geheugen te benaderen dan van een harde schijf. Je kunt dat goed merken als je tweemaal achter elkaar een zware applicatie start. De tweede keer bevinden de gegevens zich in het geheugen en is het (veel) sneller gestart.

Als je wilt weten hoeveel geheugen er in gebruik is, tik je **free** in. Met de optie **-t** wordt ook het totaal getoond. Om de uitvoer in MB's te zien, kun je **-m** gebruiken. Je krijgt bijvoorbeeld de volgende uitvoer. In de tweede regel zie je hoeveel geheugen er gebruikt en vrij is afgezien van disk cache en buffers. Disk cache en buffers zullen worden vrijgemaakt als een nieuw proces geheugen nodig heeft.

	total	used	free	shared	buffers	cached
Mem:	2026	1875	150	0	58	644
-/+ buffers/cache:		1172	853			
Swap:	1027	408	619			
Total:	3053	2284	769			

6.14 Processen

Een programma kan meerdere keren gestart worden. Elke gestarte versie is een losstaand proces. Verschillende gebruikers kunnen zo tegelijkertijd hetzelfde programma gebruiken.

Wil je dat een programma in de achtergrond wordt uitgevoerd, zet dan een **&**-teken achter de naam als je het start. Op die manier zorg je ervoor dat een console of terminal niet in beslag genomen wordt. Heb je een programma al in de voorgrond gestart (dus geen **&**-teken gebruikt), dan kun je hem naar de achtergrond verplaatsen door op **ctrl-z** te drukken en vervolgens **bg** in te tikken. Om het vervolgens weer naar de voorgrond te verplaatsen, kun je **fg** gebruiken. Als je meerdere programma's in de achtergrond hebt draaien, dan kun je daar een overzicht van krijgen met **jobs**. Om een van deze programma's weer op de voorgrond te krijgen, kun je **fg #** gebruiken, met in plaats van **#** het jobnummer dat je bij de uitvoer van **jobs** zag.

Start je een programma voor X via een terminal, dan kunnen meldingen in de terminal getoond worden, ook als het programma in de achtergrond draait. Wil je die meldingen per se weg hebben, gebruik dan **programmanaam &> /dev/null &**. Start je het vanuit een menu van je window manager of desktopomgeving, dan zullen de meldingen in de console verschijnen van waaruit X gestart is of misschien nergens. Werkt een programma niet, dan kun je eens proberen het vanuit een terminal te starten om eventuele foutmeldingen te kunnen zien.

Een PID is een Process-ID. Elk proces heeft een eigen nummer. Met **ps aux** krijg je een compleet overzicht van draaiende processen. Het commando **pidof proces**, waarin je proces vervangt door de naam van het programma, geeft de PID van dat proces. Als je de naam niet exact weet, kun je ook **ps aux | grep stukvannaam** gebruiken.

Met het hulpprogramma **top** krijg je een live overzicht van op dit moment actieve processen. Je ziet daarbij hun processor- en geheugengebruik. Het is handig dit in een terminal in X te starten, zodat er veel in beeld past.

Je kunt een proces meer CPU-tijd geven met **renice n pid**. Als je voor **n** **-20** invult krijg het proces de hoogst mogelijke prioriteit, met **0** normale en met **20** de laagst mogelijke. Een normale gebruiker mag alleen positieve getallen gebruiken (dus vertragen). Alleen root mag ook negatieve getallen gebruiken. Om een programma meteen met een alternatieve prioriteit te starten, gebruik je **nice n programmanaam**.

Je kunt een programma dat niet reageert afsluiten met **kill pid**, waarin je **pid** vervangt door de PID van het proces. Je kunt ook **killall naam** gebruiken. Daarbij moet je niet de PID invullen, maar de naam van het proces. Je kunt een programma in X vrij letterlijk afschieten door **xkill** te starten en vervolgens het gewenste venster aan te klikken. In **top** kun je een proces killen door op **k** te drukken en vervolgens de PID in te tikken.

Je kunt het kill-commando verschillende signalen laten sturen. Zie **man 7 signal** voor een compleet overzicht. Met **-9** wordt het proces absoluut afgebroken door de kernel. Het is geen nette manier om processen te stoppen. Het programma krijgt dan geen tijd meer om lock-bestanden te verwijderen en gedeeld geheugen vrij te geven. Wil het programma echt niet weg, dan is dit de definitieve oplossing. Werkt dit niet, dan werkt niets meer. Met **-15** wordt vriendelijk aan het proces verzocht of hij wil stoppen. Een goed programma roept dan de exit-procedure aan en sluit netjes af. Dit signaal wordt gestuurd als je geen andere opgeeft. Shells en daemons kun je herstarten met de optie **-1** of **-HUP**. Dit is het zogenaamde SIGHUP-signaal. Dit signaal zegt dat de gebruiker de lijn opgehangen heeft (uit de modemtijd).

Processen kunnen met behulp van de system call `fork()` kindprocessen creëren. Het proces zelf is dan een parent. Een zombie is een proces dat overleden is, maar dat nog niet heeft doorgegeven aan z'n parent. Het gebruikt geen CPU of geheugen meer, maar staat nog wel in de process table. Ze verdwijnen als je hun parent killt.

Als je een proces hebt gestart vanuit een terminal, dan zal dat proces vaak worden afgebroken als je die terminal sluit. Om dat te voorkomen, moet je het programma starten met **nohup prog &**. De output van het programma wordt dan naar het bestand `nohup.out` in de huidige directory gestuurd.

Het commando **uptime** zegt naast hoe lang je computer al aan staat ook ruwweg hoeveel processen de processor graag compleet bezet zouden houden. Het is een pseudo-meetmethode van de belasting van je systeem. Als 2 processen 100% willen, is de load 2. Als 3 processen elk 25% CPU-tijd willen, is de load 0.75. Uptime geeft de load voor de afgelopen 1, 5, en 15 minuten weer.

6.15 Gebruikers in de gaten houden

Je kunt zien welke gebruikers zijn ingelogd met **who**. Met **w** krijg je ook te zien wat ze op het moment aan het doen zijn. Is iemand met zaken bezig die je liever niet hebt, dan kun je hem van je server af schoppen door zijn shell (waarschijnlijk Bash) te killen.

6.16 Printen

Er zijn twee oude systemen onder Unices om te printen: die van BSD (Berkeley Line Printer Daemon of LPD) en die van System V (AT&T Line Printer system). Beide zijn, zoals al aan de namen te zien is, ontworpen voor lijnprinters. In het verleden werden bij Linux systemen gebruikt die waren gebaseerd op de eerste (LPD), zoals LPRng. Je kunt hierbij een document printen met **lpr file**. De queue of spool is te bekijken met **lpq** en een printopdracht kan uit de queue verwijderd worden met **lprm job**, waarin je job vervangt door het nummer van de job dat je met `lpq` kunt opvragen. Bij het printstelsel van System V gebruik je resp. **lp**, **lpstat** en **cancel**. Tegenwoordig wordt onder Linux gebruikgemaakt van het modernere printstelsel CUPS. CUPS heeft ondersteuning voor beide sets van commando's, zodat oudere software hiermee nog kan printen. Je kunt het ook zelf gebruiken voor het printen van o.a. tekstbestanden of PostScript-documenten. Vanuit grafische software voor KDE of GNOME kun je gebruikelijk printen kiezen in het menu File/Bestand.

Er zijn enkele handige tooltjes voor tekstbestanden. Met **pr** kun je een tekst opmaken en voorzien van header, footer en paginanummer. Met **enscript** kun je zo ongeveer hetzelfde, maar dat zet de tekst om naar PostScript. Dat kan of naar de printer spool gestuurd worden of opgeslagen in een bestand. Je kunt er ook source code mee pretty-printen. Zie de manual pages voor meer informatie.

7 GNU-utility's

GNU is een recursief acroniem en staat voor "GNU's not UNIX". De GNU-software is Unix-compatible software die wordt uitgegeven door de Free Software Foundation, de FSF. De echte kracht in de GNU-utility's zit hem in het feit dat complexe taken kunnen worden opgebroken in snelle, kleine stukjes. Niet alle GNU-utility's worden hier genoemd, alleen de meest gebruikte. Voor een compleet overzicht zie <http://www.gnu.org>. Van de GNU-utility's die hier worden besproken, worden ook lang niet alle mogelijkheden genoemd. Die kunnen in manual pages en andere bronnen gevonden worden.

Zie ook paragraaf 6.5 over hoe je met input en output werkt.

7.1 Cat

Een van de simpelste programmaatjes. Het toont de inhoud van een bepaald bestand dat je opgeeft: `cat file`. Je kunt het ook gebruiken om bestanden mee aan elkaar te plakken: `cat file1 file2 > filesamen`.

7.2 Grep

Grep is een afkorting voor General Regular Expression Parser. Het is een programma dat met behulp van reguliere expressies ergens naar kan zoeken. Met reguliere expressies kun je zoekpatronen maken om naar bepaalde stukken tekst te zoeken. (Maar er zijn wel dingen te bedenken die onmogelijk zijn om in een "regex" te formuleren.) Naast grep kunnen regexps ook in een aantal andere programma's gebruikt worden. Een tweetal simpele voorbeelden van het gebruik van grep:

- `grep -i foo file.txt`
- `ps aux | grep firefox`

Met het eerste commando wordt naar de string `foo` gezocht in het bestand `file.txt`. De optie `-i` zorgt ervoor dat er niet op hoofd- en kleine letters gelet wordt (ignore case). Met het tweede commando wordt in de output van `ps` gezocht naar het woord `firefox`.

In grep kun je de volgende reguliere expressies gebruiken. Er wordt telkens aangegeven waar een expressie mee zal matchen.

Expressie	Betekenis
<code>^</code>	begin van de regel
<code>\$</code>	eind van de regel
<code><</code>	begin van een woord
<code>></code>	eind van een woord
<code>.</code>	elk enkel karakter
<code>[str]</code>	elk karakter in de string <code>str</code>
<code>[^str]</code>	elk karakter dat niet in de string <code>str</code> voorkomt
<code>[a-b]</code>	elk karakter tussen <code>a</code> en <code>b</code>
<code>\</code>	heft speciale betekenis van het teken erna op (escapen)
<code>*</code>	0 of meer herhalingen van het vorige item

Als je `grep -E` of `egrep` gebruikt, kun je zelfs nog meer expressies vormen:

Expressie	Betekenis
+	1 of meer herhalingen van het vorige item
?	0 of 1 herhaling van het vorige item
{j}	exact j herhalingen van het vorige item
{j,}	j of meer herhalingen van het vorige item
{,k}	maximaal k herhalingen van het vorige item
{j,k}	j tot k herhalingen van het vorige item
s t	s of t
(exp)	behandel exp als een enkel item

7.3 Strings

Strings zoekt in een opgegeven bestand naar strings. Een string is een willekeurig serie karakters, zoals dryl of a&3?. Je kunt dit gebruiken om in bestanden die geen tekstbestand zijn te zoeken naar leesbare informatie. Het gebruik is eenvoudig: **strings bestand**. Standaard wordt er naar strings gezocht van minstens 4 tekens lang, maar met de optie `--bytes=n` kun je dit veranderen in n tekens.

7.4 Sort en uniq

Sort alfabetiseert de input. Je kunt het in combinatie met **uniq** gebruiken. Uniq verwijdert dubbele regels uit gesorteerde input.

7.5 Cut

Cut kan een bepaald stukje uit elke regel knippen. Om bijvoorbeeld van byte 2 tot en met byte 4 van elke regel uit een bestand te tonen, gebruik je **cut -b 2-4 foo**. Door geen bestandsnaam op te geven, kun je cut ook in een pipe gebruiken om een stuk uit de standaard invoer te knippen. Je kunt ook bepaalde kolommen uitknippen als je aangeeft welk teken de kolommen van elkaar scheidt: **cut -d , -f 12 foo**. Dit toont kolom 12 in een bestand waarin de kolommen gescheiden worden door komma's.

7.6 Sed

Sed is een zogenaamde "streaming editor". Hij kan tekst bewerken tijdens het doorwandelen van die tekst. Sed heeft vele mogelijkheden. Een voorbeeld: **cat foo | sed -e "s/oud/nieuw/"** Dit vervangt alle voorkomens van de tekst "oud" door "nieuw" in de uitvoer van cat.

Naast in de manual page is ook veel informatie te vinden in de info page, te bekijken met **info sed**.

7.7 Awk

Awk is vooral sterk in het werken met tabellen. Net zoals bij sed en grep kun je reguliere expressies gebruiken om bepaalde tekst te vinden. Je kunt de eerste kolom uit een bestand (bestaande uit een tabel gescheiden door spaties) halen met **awk '{print \$1}' file**. Alle kolommen zijn op te vragen met \$1, \$2, etc.

Veel werk dat gedaan kan worden met een combinatie van verschillende andere GNU-utility's achter elkaar, kan met awk in een keer. Je kunt bijvoorbeeld het IP-adres van eth0 uit de output van ifconfig filteren met:

```
/sbin/ifconfig eth0 | awk '/inet addr/ {sub("addr:", "", $2); print $2}'
```

Alle mogelijkheden van awk staan beschreven in de manual page: **man awk**.

7.8 Tr

Met tr (translate) kun je een serie tekens of letters laten vervangen door een andere serie tekens of letters. Beide series moeten tussen enkele quotes geplaatst worden. Met **tr '!?";[]{}(),.' ' ' < file.txt** worden alle leestekens uit file.txt verwijderd. Om alle hoofdletters in kleine letters om te zetten, kun je **tr 'A-Z' 'a-z' < file.txt** gebruiken. Maar tr kent nog wat meer functies. Zo kun je om meerdere spaties om te zetten in enkele **tr -s ' ' < file.txt** gebruiken.

7.9 Tee

Stuurt output zowel naar een bestand als naar de standard output, oftewel naar het beeldscherm. Een voorbeeld: **ls | tee test** laat een bestandsoverzicht op het scherm zien, maar zet het ook in bestand test.

7.10 Locate

Dit is het simpelste en snelste commando om naar bestanden te laten zoeken. Daarbij wordt niet echt op je harde schijf gezocht, maar in een database met bestandsnamen. Die wordt normaal gesproken 's nachts geüpdatet. (Dus wees gerust, dat geratel in het holst van de nacht zijn geen crackers!) Om het handmatig te updaten gebruik je **updatedb**. Als je bijv. **locate bash** intikt, worden alle bestandsnamen waar "bash" in voorkomt getoond. Er kunnen ook wildcards en reguliere expressies gebruikt worden. Voorbeeld: **locate [mM]akefile**. Bijbehorend zijn **whereis** en **whatis**. Whereis zoekt naar binary's, source code en manual pages. Whatis toont de omschrijving uit de manual page van een bepaald programma.

7.11 Find

Dit is een utility waarmee echt op je harde schijf wordt gezocht. Eerst moet de begindirectory opgegeven worden, daarna de zoekopties. Voorbeeld: **find / -name Makefile -print** Hiermee worden alle Makefile-bestanden in / en in alle subdirectory's daarvan getoond. Door **-iname** te gebruiken wordt niet op hoofd- en kleine letters gelet. Door de optie **-ls** te gebruiken i.p.v. **-print** krijg je de bestanden te zien in de vorm van **ls -li**.

Als je weet hoe groot het gezochte bestand is, kun je de optie **-size** gebruiken. Zet daar een getal achter en de eenheid (c voor bytes, k voor kilobytes). Voorbeeld: **find /usr -size 100k**.

Als je een bepaald type bestand zoekt, kun je de optie **-type** gebruiken. Zet daar een letter achter om aan te geven wat voor type bestand je zoekt. Een b staat voor block (buffered) special, c voor character (unbuffered) special, d voor directory, p voor een named pipe (FIFO), f voor een normaal bestand, l voor een symbolic link en s voor een socket. Deze letters zie je ook helemaal links als je **ls -l** gebruikt, met als enige verschil dat de f daar een streepje is. Voorbeeld: **find /tmp -type l**.

Om iets met de gevonden bestanden te laten doen, kan de optie **-exec** gebruikt worden. Zet daar een commando achter dat voor alle bestanden uitgevoerd moet worden. Gebruik daarin '{}' (met de enkele quotes) om de bestandsnaam te laten invullen op die plaats en ';' om aan te geven dat het einde van het commando is bereikt. Om bijvoorbeeld naar alle .c-bestanden in /usr/src/linux met de tekst foo erin te zoeken, gebruik je:

```
find /usr/src/linux -name "*.c" -exec grep -l foo '{}' ';'
```

7.12 Xargs

De `exec`-optie van `find` voert per gevonden bestand een bepaald commando uit. `Xargs` daarentegen voert een bepaald commando uit op vele bestanden. Dat gaat aanzienlijk sneller. Het voorbeeld bij `find` kan met `xargs` als volgt worden gedaan:

```
find /usr/src/linux -name "*.c" | xargs grep -l foo
```

Een andere mogelijkheid die je zou kunnen bedenken om hetzelfde te doen is het commando:

```
grep -l foo `find /usr/src/linux -name "*.c" `
```

Dat zal echter mislukken als het commando tussen backquotes een langere output heeft dan de maximale lengte die een commando mag hebben.

Bestandsnamen (en directorynamen) met spaties erin zullen `xargs` verwarren, omdat hij spaties ziet als een teken dat de volgende bestandsnaam eraan komt. Om dat probleem te verhelpen, moet je de optie `-print0` (met het cijfer nul) gebruiken i.p.v. `-print` bij `grep` (zodat er een null character tussen de bestandsnamen komt) en de optie `-0` (wederom een nul) bij `xargs` (zodat `xargs` het null character ziet als bestandsnaamonderscheider).

Normaal gesproken werkt `xargs` automatisch, maar door commando's als `rm` kan om bevestiging gevraagd worden. Gebruik dan de optie `--interactive` of `-p` om `xargs` interactief te maken.

7.13 File

Met `file` kun je laten zien wat voor soort data een bestand bevat. Erg handig voor bestanden waarvan je niet direct weet wat het voor moet stellen. Voorbeeld: `file 5x8.pcf`

7.14 Head en tail

Met `head` en `tail` kun je de eerste cq. laatste zoveel regels van een bestand of standaard uitvoer bekijken. Standaard worden er 10 regels getoond, met `head -n 20 foo` toon je de eerste 20 regels van bestand `foo`. Een leuke optie is `-f` om een groeiend bestand te volgen. Met `tail -f logfile` kun je handig een logbestand blijven volgen.

7.15 Ldd

Met `ldd` kun je laten zien welke gedeelde (shared) library's een bepaald programma nodig heeft. Voorbeeld: `ldd /usr/bin/perl`.

7.16 Touch

Met `touch file` verander je de access time van een bestand in de huidige datum en tijd. Als het bestand file nog niet bestaat, maakt het een nieuw leeg bestand aan. Daar wordt `touch` ook vaak voor gebruikt.

7.17 Split

Split kan een groot bestand opsplitsen in kleinere stukken. Een voorbeeld: **split -b1400k JDK.tar.gz** splitst JDK.tar.gz op in blokken van 1,4 MB. De opgehakte bestanden eindigen op aa, ab, ac enz. Om ze weer aan elkaar te plakken kun je simpelweg **cat file* > file** gebruiken.

7.18 Diff

Diff laat de verschillen zien tussen twee bestanden. Het gebruik is eenvoudig: **diff file1 file2**. Geef je twee directory's op, dan worden verschillen getoond tussen elk tweetal gelijknamige bestanden in de directory's. Met de optie **--recursive** worden ook subdirectory's meegenomen. Met dit programma worden patches gemaakt.

7.19 Patch

Kan gebruikt worden om bijv. source code te upgraden tot de nieuwste versie. Als je bijvoorbeeld de source van kernel 2.6.20 hebt en je downloadt een patch (gemaakt met diff) met de verschillen tussen 2.6.20 en 2.6.21, dan kun je je code upgraden naar 2.6.21. Patches zijn klein en zorgen ervoor dat je niet de hele source code opnieuw hoeft te downloaden. Het gebruik: **patch origineel patchfile**.

8 Systeemconfiguratie

8.1 Het opstartproces en runlevels

De kernel wordt geladen met een bootloader (LILO of GRUB). Daarna wordt het proces `init` geactiveerd. Wat `init` doet, wordt geregeld in `/etc/inittab`. Er worden runlevels gedefinieerd en een van de runlevels wordt als standaard aangewezen. Verder voert `init` initialisatiescripts uit en stelt het consoles beschikbaar om in te loggen.

Runlevel 0 is altijd de status waarin het systeem gestopt is en 6 is een reboot. De rest kan verschillen per distributie. Je kunt vaak in `/etc/inittab` terugvinden welke runlevel waar goed voor is. Een mogelijke indeling:

Runlevel	Omschrijving
0	halt
1	single-user mode
2	multi-user, without NFS
3	full multi-user mode
4	unused
5	X11
6	reboot

Met `init` of `telinit` kun je van runlevel veranderen.

Ook bij de LILO-prompt kun je de runlevel opgeven. Zet het nummer van de gewenste runlevel dan achter het gewenste label (bijv. `linux 3`). In plaats van een 1 (voor single-user mode) kun je ook het woord `single` (dus bijv. `linux single`) gebruiken. Bij GRUB kunnen deze opties worden opgegeven door op de regel van de gewenste kernel te gaan staan en op `e` te drukken. Je kunt dan opties toevoegen.

In `/etc/init.d` of `/etc/rc.d/init.d` staan een aantal scripts om services te starten en stoppen (met de argumenten `start` en `stop`.) Vaak werkt `restart` ook om iets te herstarten en `status` om de status op te vragen. Er bestaan ook verschillende directories `/etc/rc#.d` of `/etc/rc.d/rc#.d`, waarbij `#` de runlevel aangeeft. In deze directories staan symlinks naar de scripts in de `init.d`-directory. De namen van deze symlinks zien eruit als `Sxxnaam` of `Kxxnaam`. Hierbij geeft de `S` start aan en `K` kill (stop). Wordt de runlevel ingegaan, dan worden de symlinks die met een `S` beginnen gebruikt, bij het verlaten die met een `K`. De `xx` na de `S` of `K` is een nummer dat de volgorde bepaalt (01 wordt eerder uitgevoerd dan 10). Er mogen best gaten in de nummers die voorkomen zitten (niet elk nummer hoeft te bestaan) en als de volgorde waarin drie dingen worden gestart niet uitmaakt, mogen ze ook alle drie hetzelfde nummer hebben. De naam na het nummer is alleen om gemakkelijk te kunnen zien waar de symlink naar wijst en is verder niet van belang.

8.2 Library's installeren

Als je nieuwe library's hebt gecompileerd en zelf ergens neerzet, dan moet je ze nog laten opnemen in de database van library's. De desbetreffende directory moet daartoe in de variabele `$LD_LIBRARY_PATH` opgenomen zijn of in `/etc/ld.so.conf` staan. De directory's `/lib` en `/usr/lib` worden echter altijd al doorgekeken. Voer (als root) `/sbin/ldconfig` uit om de database van library's te updaten als je `ld.so.conf` aanpast.

Als je `make install` gebruikt voor het installeren van zelf gecompileerde library's, dan staat in de Makefile vaak al wat code zodat een eventuele benodigde toevoeging aan `/etc/ld.so.conf` en het uitvoeren van `ldconfig` automatisch worden gedaan. Hetzelfde is het geval bij het installeren van library's via het pakkeetsysteem van je distributie (in die pakketten zitten ook de benodigde scripts). Dus meestal zul je dit niet handmatig hoeven te doen.

8.3 Partities op fouten controleren

Doe dit als X niet gestart is en er geen programma's driftig met de harde schijf in de weer zijn. Het beste is het systeem in runlevel 1 (single user) brengen. Je kunt dit bereiken met **init 1** of tijdens het booten zoals hiervoor uitgelegd. Eenmaal in single-user mode moet je de te controleren partitie nog unmounten met **umount /dev/xxx** (zie ook paragraaf 6.12). Dit gaat zonder problemen voor partities zonder belangrijke gegevens. De root-partitie kun je echter niet zomaar unmounten. Wat je daarmee kunt doen, is het hermounten als read-only met **mount -o remount,ro /**.

Nu alles veilig is en echt niet meer naar de partitie geschreven kan worden, kun je het controleren laten beginnen met **e2fsck -f /dev/xxx** met in plaats van xxx de te controleren partitie. Het is in veruit de meeste gevallen verstandig y(es) te antwoorden op vragen van e2fsck om fouten te repareren. Als e2fsck verloren data vindt, wordt die in de subdirectory lost+found op de desbetreffende partitie gezet.

Na de controle kun je de partitie weer mounten. Je kunt de root-partitie hermounten als read-write met **mount -o remount,rw /**. Ten slotte kun je weer terug naar de oorspronkelijke runlevel. Bij de meeste distributies is runlevel 3 multi-user mode: **init 3**.

E2fsck kan alleen partities met ext2 of ext3 controleren. Gebruik voor ReiserFS **reiserfsck** met de optie **--fix-fixable**. Zie de manual page voor meer informatie.

8.4 Tijd en datum instellen

Er zijn twee klokken: de hardware- of CMOS-klok en de softwareklok. De softwareklok wordt vastgesteld door het OS, dat de tijd van de hardwareklok bekijkt tijdens het booten. De hardwareklok loopt altijd door, ook als je pc uit staat. Daarvoor zit er een batterijtje op het moederbord. De reden waarom er een softwareklok wordt bijgehouden, is dat deze veel nauwkeuriger is. De hardwareklok wordt gezet op de tijd van de softwareklok als je Linux afsluit.

Je kunt de softwareklok bekijken met **date** en de hardwareklok met **hwclock**. De softwareklok verander je met een commando als:

```
/bin/date --set '10/24/2004 14:35:14'
```

De hardwareklok met een commando als:

```
/sbin/hwclock --set --date="10/24/2004 14:35:14"
```

De hardwareklok kan ook in je BIOS ingesteld worden. Denk eraan dat de dag en maand in het Engels omgekeerd zijn!

Linux corrigeert de hardwareklok automatisch als die te snel of te langzaam loopt. Het is een goed idee om, nadat je de hardwareklok veranderd hebt, **echo " > /etc/adjtime** (twee keer een enkele quote achter echo) te gebruiken. Daarmee wordt de oude informatie vergeten en wordt de tijd meestal nauwkeuriger bijgehouden.

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 of UTC is dat het geen zomer- en wintertijd kent, en dus nooit een uur vooruit of terug gezet hoeft te worden.

Als de hardwareklok op GMT draait (en als je Linux-systeem dat weet), zal Linux je nog steeds de lokale tijd (CEST dus) geven als je de tijdzone juist hebt ingesteld. Maar Windows gaat ervan uit dat de hardwareklok op de lokale tijd staat. Als je naast Linux nog een ander besturingssysteem gebruikt, kun je de hardwareklok het beste de lokale tijd aan laten geven.

Voor elke tijdzone (eigenlijk van elk land) staat informatie in `/usr/share/zoneinfo`. Eén van die dingen is de zomer- en wintertijd. Als de hardwareklok op GMT draait, zul je merken dat de zomer- en wintertijd automatisch wordt aangepast (in de softwareklok, de GMT van de hardwareklok kent immers geen zomer- en wintertijd). Als de hardwareklok op de lokale tijd draait, gaat het soms wel, en soms ook niet goed. Ook als je af en toe Windows draait, wil de tijd nog wel eens verkeerd gezet worden. Het kan nooit kwaad om op dergelijke momenten te kijken of alles nog goed loopt, en het zonedig aan te passen.

Bij alle distributies is `/etc/localtime` een symlink zijn naar een bestand onder `/usr/share/zoneinfo/`. Er zijn vaak tooltjes om de tijdzone in te stellen. Aangeven of de hardwareklok op GMT/UTC loopt of niet kan meestal ook in een bestand onder `/etc`. Raadpleeg de documentatie van de distributie voor meer informatie.

8.5 Logbestanden

In `/var/log` staat een aantal logbestanden met meldingen van de kernel en verschillende daemons. Daar kun je gebruik van maken om meer informatie over problemen met hardware te verzamelen of om te kijken wie er in je systeem heeft ingebroken (als die sporen niet zijn gewist).

Je wilt waarschijnlijk niet dat de logbestanden maar blijven groeien. Daarom wordt vaak gebruikgemaakt van een programma als `logrotate`, dat door middel van cron regelmatig oude logbestanden backupt (en hele oude verwijdert) en nieuwe aanmaakt.

Het commando `dmesg` laat de meldingen zien die van de kernel afkomstig waren tijdens het opstarten. Daarbij ontbreken de meldingen die door andere opstartscripts gegenereerd werden.

8.6 At

At is een service om een opdracht uit te laten voeren op een bepaald tijdstip. Zo kun je je bijvoorbeeld de volgende dag om 7 uur gewekt laten worden door een cd of een MP3 te laten afspelen. Lees voor de details `man at`.

8.7 Cron

Cron is een service waarbij je met vaste regelmaat een opdracht kunt laten uitvoeren. Dit is bijvoorbeeld handig voor het maken van een backup of het opruimen van tijdelijke bestanden.

Bij veel distributies zijn er subdirectory's in `/etc` zoals `cron.daily` en `cron.weekly`. In het algemene configuratiebestand `/etc/crontab` staat dat alles in die directory elke dag resp. week moet worden uitgevoerd. Wil je dus iets toevoegen dat regelmatig moet worden uitgevoerd, zet de opdracht dan in een script in een van deze directory's en je bent klaar.

Een voorbeeldinhoud van `/etc/crontab`:

```
SHELL=/bin/sh
PATH=/sbin:/bin:/usr/sbin:/usr/bin
HOME=/

# minute, hour, day of month, month, day of week
*/10 * * * * root run-parts /etc/cron.d
14 * * * * root run-parts /etc/cron.hourly
00 19 * * * root run-parts /etc/cron.daily
06 19 * * 0 root run-parts /etc/cron.weekly
10 19 0 * * root run-parts /etc/cron.monthly
```

Het gaat hier vooral om de laatste 5 regels. Een uitleg per regel:

- Elke 10 minuten worden de scripts uitgevoerd die in de directory `/etc/cron.d/` staan.
- Elke 14e minuut van een uur worden de scripts in `/etc/cron.hourly/` uitgevoerd.
- Elke dag om 19:00 uur worden de scripts in `/etc/cron.daily/` uitgevoerd.
- Elke eerste dag van de week om 19:06 uur worden de scripts in `/etc/cron.weekly/` uitgevoerd.
- Elke eerste dag van de maand worden om 19:10 uur de scripts in `/etc/cron.monthly/` uitgevoerd.

Het is mogelijk per gebruiker extra regels voor crontab op te geven, bijvoorbeeld om opdrachten op afwijkende tijden uit te voeren. Doe dit door **crontab -e** in te tikken. Er wordt dan een editor gestart waarin je eigen regels kunt opgeven. Het crontabbestand zal ergens worden opgeslagen onder `/var/spool/cron`. (Zie paragraaf 20.24 hoe je kunt bepalen welke editor zal worden gestart.)

Stel dat je elke werkdag om 7 uur 's ochtends gewekt wilt worden met een MP3-bestand, dan kun je de volgende regel gebruiken:

```
0 7 * * 0-5 mpg123 /home/piet/mp3/*
```

Anacron zorgt ervoor dat cronjobs altijd worden uitgevoerd. Stel dat je pc om 3 uur 's nachts uit staat en dat er dan een cronjob zou zijn om de logfiles te rotaten, dan zou deze met de gewone cron niet worden uitgevoerd. Met anacron gebeurt dat wel zodra de pc weer aan gezet wordt.

Voor meer informatie zie **man cron** en **man crontab**.

8.8 Het printstelsysteem

De standaard om te printen onder Unices is PostScript, een formaat/programmeertaal gespecialiseerd in het plaatsen van tekst en plaatjes op een pagina. Er zijn printers die dit formaat direct begrijpen, zogenaamde PostScript-printers. De PostScript-data kan direct naar zulke printers worden gestuurd. Er zijn echter weinig mensen die zo'n printer hebben.

CUPS (Common Unix Printing System, <http://www.cups.org>) is het standaard printstelsysteem geworden voor Linux. Je kunt het configureren via een website die draait op poort 631 van localhost. Dus je kunt in een willekeurige browser naar <http://localhost:631> gaan. De gebruikersnaam en het wachtwoord die je daar moet invoeren, moeten die van root zijn. Zowel KDE als GNOME hebben ook configuratietooltjes.

CUPS bevat filters om verschillende formaten, zoals PostScript, platte tekst en jpeg, om te zetten naar een voor de printer begrijpbaar formaat. Er wordt gebruikgemaakt van een eigen versie van Ghostscript voor niet-PostScript-printers. Ghostscript is software van GNU om de PostScript-taal te interpreteren. Welke driver het beste is voor je printer, kun je vinden op <http://www.linuxprinting.org>. Enkele speciale sets van drivers die je via CUPS kunt gebruiken zijn:

Gimp-Print (<http://gimp-print.sf.net>) - Voor verschillende printers van Canon, Epson, Lexmark, Sony, Olympus en PCL-printers van HP (PCL is Printer Command Language, uitgevonden door HP). De drivers zijn bedoeld om hoogwaardige (fotokwaliteit) prints te kunnen maken. Ze kunnen ook worden gebruikt door Ghostscript en direct door GIMP (een tekenprogramma).

HP Linux Printing Project (<http://hpinkjet.sf.net>) - Voor printers van HP (de hpijs-driver). Omvat ook ondersteuning voor gecombineerde printer-scanners en kaartlezers, die vaak in de apparaten ingebouwd zitten.

PNM2PPA (<http://pnm2ppa.sf.net>) - Een kleine serie printers van HP maakt gebruik van een systeem genaamd PPA (Printing Performance Architecture). Ze worden nu niet meer geproduceerd. Deze printers hebben minder hardware en de driver moet meer voorkauwen. Voor PPA-printers is PNM2PPA als filter nodig.

Een geïnstalleerde printer is via het netwerk met behulp van Samba te bereiken. Meer daarover vind je in de documentatie van Samba. Meer over Samba vind je in paragraaf 10.6.

Een gedetailleerd overzicht van hoe het printstelsel werkt, is te vinden in de Printing HOWTO en de documentatie van CUPS. Zie ook paragraaf 6.16.

8.9 Geluid

In de kernel zitten drivers met ondersteuning voor je geluidskaart. Vroeger was dit OSS (Open Sound System), maar tegenwoordig wordt ALSA (Advanced Linux Sound Architecture) gebruikt. Als je geluidskaart ondersteuning heeft voor hardwarematig mixen van geluid, dan is het met ALSA-drivers mogelijk meerdere processen geluid naar de kaart te laten sturen. Bij OSS kon dat niet. Veel geluidskaarten, vooral goedkopere, kunnen niet hardwarematig mixen. In dat geval kan maar één proces tegelijk van de geluidskaart gebruikmaken.

Als oplossing hiervoor zijn sound servers of sound daemons gemaakt. Deze programma's mixen het geluid softwarematig. Bekende zijn ESD of Esound en aRts. ESD staat voor Enlightenment Sound Daemon (het was oorspronkelijk voor de window manager Enlightenment gemaakt) en werd gebruikt bij GNOME. KDE maakte lange tijd gebruik van aRts (analog real-time synthesizer). Naast mixen kunnen sound servers ook nog een andere taak verzorgen: het kunnen verzenden van geluid over het netwerk. ALSA heeft ook een oplossing om geluid (softwarematig) te mixen: de dmix-plugin. Hiermee kun je alleen geen geluid over het netwerk sturen. Een andere overweging die door ontwikkelaars in het achterhoofd wordt gehouden, is dat ALSA er alleen voor Linux is en niet voor andere Unices, zoals de BSD-varianten.

Applicaties die geluid willen afspelen, moesten lange tijd of direct OSS of ALSA gebruiken, of een sound server. Oude applicaties hebben wellicht alleen ondersteuning voor OSS. Hiervoor zit OSS-emulatie in ALSA. Als een sound server in gebruik is, zal het de geluidskaart bezet houden. Als een programma niet zelf met de sound server overweg kan, zal het met een wrapperprogramma gestart moeten worden. Voor aRts is er **artsdsp** en voor ESD is er **esddsp**. Geef als argument het programma mee.

Eigenlijk waren zowel ESD als aRts al enige tijd aan vervanging toe. Ze zijn lang geleden ontwikkeld en worden niet langer meer onderhouden. GNOME maakt steeds meer gebruik van GStreamer. GStreamer is geen sound server, maar een multimediaframework. Het kan gebruikt worden om media-applicaties te bouwen. Dat kan een mediaspeler zijn, maar ook een video-editor. Codecs om audio en video te en- of decoderen zijn als plugin toe te voegen. (Zie ook paragraaf 15.3.) Voor het afspelen heeft GStreamer o.a. outputplugins voor OSS, ALSA en de sound servers ESD en aRts. Door GStreamer te gebruiken, is er geen directe afhankelijkheid meer met de specifieke "backend" (direct ALSA, een sound server of weer een ander multimediaframework). Ook KDE is vanaf versie 4 afgestapt van de oude sound server. Daarvoor in de plaats is Phonon gekomen, wat ook werkt als laag tussen de applicatie en de backend.

Er zijn nog enkele andere sound servers en multimediaframeworks die minder wijd verspreid zijn. Zo zijn er de sound servers PulseAudio (<http://pulseaudio.org>) en JACK (<http://jackaudio.org>, ontworpen voor low-latency) en de multimediaframeworks NMM (Network-integrated Multimedia Middleware, <http://www.networkmultimedia.org>) en MAS (Media Application Server, <http://www.mediaapplicationserver.net>).

9 Kernel

9.1 Wat is de kernel?

De kernel is de basis van het besturingssysteem en het deel dat echt "Linux" heet. Het vervult essentiële taken, zoals geheugenmanagement, scheduling om taken te verdelen over de processortijd, en communicatie met de hardware via device drivers. Device drivers kun je in de kernel zelf meebakken of als module compileren. Een module is een stukje code dat je naast de kernel kunt laden als je het nodig hebt. De kernel kan niet onbeperkt groot zijn. Het is daarom verstandig om zaken die niet permanent nodig zijn als module te compileren. Daarnaast kunnen modules uit het geheugen gehaald worden om geheugen te besparen.

Het versienummer van de kernel heeft altijd de vorm a.b.c. Als de b even is, gaat het om een stabiele kernel, bij oneven getallen om een developmentversie. De nieuwste stabiele reeks is 2.6.x.

Je kunt de versie van de kernel die je nu gebruikt opvragen met **uname -r**.

9.2 Kernel compileren

De standaard locatie voor de broncode van de kernel `/usr/src/linux`. Als je meerdere versies bijhoudt, is het handig om ze in directory's als `/usr/src/linux-2.6.10` en `/usr/src/linux-2.6.11` te zetten. Vervolgens kun je een symlink `/usr/src/linux` maken met een commando als **ln -s /usr/src/linux-2.6.11 /usr/src/linux**. Ga naar `/usr/src/linux` en typ **make menuconfig** of **make xconfig** om een tekstgebaseerd resp. X-gebaseerd configuratieprogramma te starten. Bij de meeste keuzemogelijkheden kun je help opvragen. Doe dat als je niet weet wat je moet kiezen.

Als je klaar bent met het configureren, moet je voor kernel 2.4.x **make dep** doen. Voor 2.6.x is dit niet nodig. Vervolgens kun je eventueel **make clean** gebruiken om alles van een vorige compilatieslag weg te halen. Doe je dat niet, dan zullen reeds gecompileerde onderdelen niet opnieuw gecompileerd worden. Nu kun je de kernel zelf maken met **make zImage**. Als de kernel te groot blijkt te zijn (je zult vanzelf een foutmelding krijgen bij het opstarten), kun je de kernel ook compileren met **make bzImage**. Je kunt eigenlijk net zo goed altijd **bzImage** gebruiken. Het zorgt er niet voor dat de kernel met `gzip` wordt ingepakt, zoals soms wordt gedacht. De kernel wordt met `gzip` ingepakt, net zoals bij **zImage**. De `b` staat voor `big`. Met **bzImage** zorg je ervoor dat er "staggered loading mode" gebruikt wordt. Normaal gesproken kan er slechts 512 kB geheugen geadresseerd worden voor de kernel tijdens het opstarten. Met deze methode kan er 1024 kB worden gebruikt. Daardoor zijn grotere kernels te gebruiken.

Maak de modules met **make modules**. Als je een aangepaste versie maakt van dezelfde kernerversie als die je al gebruikt, dan zullen de oude modules overschreven worden bij de volgende stap. Wil je ze bewaren, verplaats ze dan met een commando als **mv /lib/modules/2.6.11 /lib/modules/2.6.11-working**. Installeer de modules met **make modules_install**. De nieuwe modules komen dan te staan onder `/lib/modules/kernerversie`.

Kopieer het bestand `/usr/src/linux/arch/i386/boot/bzImage` (de zojuist gecompileerde kernel) naar `/boot` onder een andere naam dan de huidige kernel, zodat je de oude kernel nog bij de hand hebt voor als het fout gaat. Noem het bijvoorbeeld `kernel-2.6.11-custom`.

Als laatste stap moet de bootloader worden aangepast om de nieuwe kernel te kunnen laten booten. Dit staat verderop uitgelegd. Laat de oude kernel ook staan tussen de mogelijkheden van de bootloader. Dan kun je de oude kernel gebruiken in het geval dat de nieuwe niet werkt.

9.3 Modules

Gebruik **insmod** of **modprobe** om een module te laden, **lsmod** om te zien welke geladen zijn en **rmmod** om een module te verwijderen. Achter **insmod** moet je een bestandsnaam opgeven van een module (eventueel met het hele pad ervoor als het niet in de huidige directory staat), achter **modprobe** alleen de naam zonder extensie (.o voor kernel 2.4.x, .ko voor kernel 2.6.x). **Modprobe** zoekt zelf uit of er nog andere modules nodig zijn en laadt die indien nodig. Meer informatie over een module kun je opvragen met **modinfo modulenaam**. Gebruik hier ook de basisnaam zonder extensie. Zie de manual pages voor meer informatie.

De modules staan altijd in `/lib/modules/kernelversie`.

9.4 Bootloader

Een bootloader is nodig om tijdens het opstarten te kunnen kiezen welk OS geboot moet worden. Het is nodig om uit verschillende kernels te kunnen kiezen, maar ook om zowel Windows en Linux te kunnen booten als je beide op hetzelfde systeem hebt geïnstalleerd.

Het eenvoudigste is de bootloader in de MBR (master boot record) van je harde schijf zetten. Wat ook kan is het aan het begin van een partitie zetten. Die partitie moet dan wel als bootable gemarkeerd zijn. Dat kun je wijzigen met **fdisk**. (Als je de bootloader aan het begin van een partitie zet en er is een andere bootloader in de MBR, dan zal die laatste worden gestart tijdens het booten.)

9.4.1 LILO

LILO staat voor Linux Loader en is een oude bootloader. Het configuratiebestand van LILO is `/etc/lilo.conf`. Een voorbeeldinhoud is:

```
boot=/dev/hdc
map=/boot/map
install=/boot/boot.b
vga=normal
lba32
prompt
timeout=50
message=/boot/message
default=Linux

image=/boot/kernel-2.6.11
    label=Linux
    root=/dev/hdc5
    read-only

other=/dev/hdc1
    label=Windows
    table=/dev/hdc
```

Met de regel `lba32` zorg je ervoor dat de kernel ook gestart kan worden als hij na de 1024e cylinder staat. Wat achter `boot` staat, is de plaats waar LILO geïnstalleerd gaat worden. Door de device van de harde schijf op te geven, wordt dat de MBR. Wat achter `install` staat, is het bestand dat in de boot sector gaat komen. Dit kan zowel een normale tekstgebaseerde LILO-prompt zijn als een enigszins grafisch menu. Wat achter `default` staat, is wat na de timeout gekozen zal worden. Voor een overzicht van configuratiemogelijkheden zie **man lilo.conf**.

Alle kernels die je met LILO wilt kunnen starten, moeten in een image-blok gezet worden. Met label geef je een woord dat je moet intikken om die kernel te starten. Om DOS of Windows van een bepaalde partitie op te starten, gebruik je niet image, maar other. Hierbij is het genoeg om een label en de partitie om van op te starten aan te geven. Op de LILO-prompt kun je met TAB een overzicht van mogelijkheden krijgen.

Per kernel moet je de root-partitie opgeven en eventuele extra parameters met een append-regel. In het voorbeeld hier worden geen extra opties meegegeven. Om ide-scsi te laten gebruiken kun je bijv. de regel `append="hda=ide-scsi"` gebruiken. (Zie ook paragraaf 15.1.1.) Het is normaal om de root-partitie eerst read-only te laten mounten. Later in het opstartproces wordt het dan geremount als read-write.

Als je dit configuratiebestand aanpast, gebeurt er nog helemaal niks. Om LILO echt opnieuw in de MBR te laten zetten, moet je (als root) `/sbin/lilo` uitvoeren.

Bij problemen is het aantal letters dat verschijnt bij het booten is een indicatie voor hoe ver LILO komt. De meest voorkomende oorzaken van een probleem zijn:

- LILO is niet opnieuw uitgevoerd na een nieuwe kernel te hebben gecompileerd.
- `/etc/lilo.conf` bevat fouten.
- De MBR (Master Boot Record, eerste stuk van de harde schijf) is corrupt.

Wat vaak helpt is LILO opnieuw installeren. Dit kan door het systeem te booten m.b.v. van een bootdisk en vervolgens `/sbin/lilo` uit te voeren.

9.4.2 GRUB

Een alternatieve bootloader is GRUB (<http://www.gnu.org/software/grub/>), GRand Unified Bootloader. In veel distributies wordt dit standaard gebruikt i.p.v. het oudere LILO, omdat het een aantal voordelen heeft t.o.v. zijn voorganger. Om te beginnen is het OS-neutraal. LILO kan alleen vanuit een Linux-systeem geconfigureerd en geïnstalleerd worden. Bij GRUB is dat vanuit verschillende OS'en mogelijk. Verder biedt GRUB een interactief en configureerbaar menu. Er is altijd de mogelijkheid een command line te krijgen waarmee je je systeem kunt booten. Als LILO niet goed is geconfigureerd, is het niet mogelijk te booten totdat je met een bootdisk weer in je Linux-systeem bent, het goed configureert en herinstalleert. En ten slotte heeft het veel meer mogelijkheden.

Het configuratiebestand van GRUB staat in `/boot/grub/grub.conf` of `menu.lst`. Een voorbeeldinhoud is:

```
timeout 10
default 0
splashimage = (hd0,1)/grub/splash.xpm.gz

title Gentoo Linux
root (hd0,1)
kernel /kernel-2.6.9 root=/dev/hde3
boot

title Windows
rootnoverify (hd0,0)
makeactive
chainloader +1
```

GRUB gebruikt niet de devicenamen voor harde schijven en partities van Linux. De eerste harde schijf is altijd (hd0), waar die ook op zit aangesloten. De tweede is (hd1) etc. De eerste partitie van de eerste harde schijf is (hd0,0), de tweede (hd0,1) etc.

Per Linux-kernel of ander OS maak je een blok dat begint met de titel die getoond moet worden in het menu van GRUB. Voor Windows is altijd hetzelfde blok nodig. Let alleen op de partitie waar het geïnstalleerd is.

Kijk bij de Linux-kernel uit hoe je naar de kernel verwijst. De root-regel geeft niet de root-partitie aan, maar de root waaronder GRUB de kernel moet zoeken. In het voorbeeld hier is (hd0,1) een partitie die in Linux gemount wordt onder /boot. In deze partitie staat de kernel direct. Als de data voor /boot in de root-partitie staat en de root-partitie (hd0,1) is, dan zou je naar de kernel moeten wijzen met (hd0,1)/boot/kernel-2.6.9.

Eventuele extra opties voor de kernel zet je direct achter de naam van de kernel (zoals `root=/dev/hde3` in het voorbeeld hier).

Gebruik **grub-install /dev/xxx** om GRUB te installeren. Vervang xxx door de devicenaam van je harde schijf. (Eventueel kun je ook een partitie aangeven, maar zoals gezegd is het eenvoudiger het in de MBR te zetten.) Je hoeft GRUB niet te herinstalleren na veranderingen. Het configuratiebestand wordt bij het opstarten ingelezen.

10 Netwerken

Voor het configureren van het netwerk zijn er verschillen tussen de distributies. Raadpleeg de documentatie voor details. In dit hoofdstuk worden alleen algemeen geldende zaken behandeld.

10.1 Ethernet en UTP

De meest gebruikte soort netwerkkaart zijn de zg. ethernetkaarten. De fysieke verbinding wordt verzorgd door UTP-kabels (Unshielded Twisted Pair). Een inmiddels verouderde mogelijkheid is coaxkabels. Een coaxverbinding heeft een maximum van 10 Mbit per seconde. Met UTP haal je ook 100 MBit. Nog nieuwer is draadloos (WLAN, Wireless Local Area Network).

Je kunt twee computers door middel van een UTP-kabel rechtstreeks met elkaar verbinden. Zo'n kabel noemt men een crosslink-kabel. Door aan beide computers een statisch IP-adres te geven, kunnen ze met elkaar communiceren. Wil je meerdere computers aan elkaar koppelen, dan heb je een hub of switch nodig. Alle computers hebben dan met een normale UTP-kabel (ook wel patchkabel) rechtstreeks verbinding met de hub of switch. De hub of switch verzorgt het dataverkeer tussen de computers. Een hub biedt data aan aan alle aangesloten computers. Bij een switch gaat de data alleen naar de computer waar het voor is bestemd. Een switch zal over het algemeen data sneller doorvoeren. Een hub of switch kan weer gekoppeld worden aan een andere hub of switch om het netwerk verder uit te breiden.

Wellicht wil je met het hele netwerk een internetverbinding delen. Daarvoor moet één van de computers in het netwerk een verbinding met het internet hebben en fungeren als gateway voor de rest. Die computer als gateway laten gebruiken, doe je met een masquerading firewall (zie sectie Masquerading firewall). In die computer moeten dan twee netwerkkaarten zitten als je voor de internetverbinding een netwerkkaart nodig hebt. Als alternatief kun je ook een router nemen die kan fungeren als gateway. Tegenwoordig is zo'n router al standaard uitgerust met bijvoorbeeld een 4-poortswitch en een draadloos access point.

Er moet ondersteuning voor je netwerkkaart in de kernel zitten of als module aanwezig zijn. De eerste ethernetkaart heet eth0, een tweede eth1, etc. Als de kaart goed is herkend, dan zou je het MAC-adres van de kaart (en andere informatie) moeten kunnen zien met **ifconfig**. (Ben je niet root, dan moet je **/sbin/ifconfig** gebruiken.) Ook in de logs van de kernel, te bekijken met **dmesg**, is het MAC-adres terug te vinden.

10.2 Netwerkbegrippen

Een protocol is een afspraak over wat er over een medium heen en weer gestuurd wordt om tegen elkaar te praten. Het protocol waar op het internet en bij ethernet gebruik van wordt gemaakt heet TCP/IP. Dit protocol is opgebouwd uit verschillende lagen:

IP Dit deel is het laagste niveau en verzorgt het verzenden en ontvangen van IP-pakketjes. Elk IP-pakket heeft o.a. een source (bron) en destination (doel) IP.

TCP Het TCP-deel is een niveau hoger dan de vorige en verzorgt de communicatie over een verbinding. Het zorgt ervoor dat datapakketten worden herkend, altijd aankomen en op de juiste volgorde worden gebruikt.

UDP Dit is te gebruiken i.p.v. TCP. Het grootste verschil met TCP is dat er geen garantie is dat de data aangekomen is of in de juiste volgorde bekeken wordt. Daardoor is het sneller, maar ook minder betrouwbaar. Verrassend genoeg kwam UDP later dan TCP.

SMTP (voor e-mail), HTTP (voor websites), FTP (file transfer) en SNMP (usenet) zijn voorbeelden van protocollen die weer bovenop TCP/IP werken. Als programmeur hoef je je geen zorgen te maken over het oversturen van elk individueel pakketje data. Dat handelt TCP/IP af.

Hier volgt een lijst van begrippen die met netwerken en het internet te maken hebben. Deze uitleg gaat uit van het klassieke IP-protocol, versie 4 (IPv4). Ondertussen wordt er steeds verder overgegaan op versie 6 (IPv6).

IP-adres Elk netwerkdevice, zoals een ethernetkaart of modem, heeft bij het TCP/IP-protocol een uniek nummer: het IP-adres. Het IP-adres bestaat uit vier getallen van tussen 0 en 255, gescheiden door punten, bijvoorbeeld 145.23.443.1. Elk getal is maximaal 255, omdat elk getal door 8 bits wordt weergegeven. Het gehele IP-adres is 32 bits groot. (Bij IPv6 is het 128 bits om meer adressen ter beschikking te stellen.)

Het IP-adres bestaat uit twee delen: een netwerk- en hostgedeelte. Het netwerkgedeelte komt eerst en geeft aan op welk netwerk de machine zit. Het hostgedeelte komt daarachter en geeft aan welke machine het precies is binnen dat netwerk. Waar de scheiding tussen beide delen zit, hangt af van de class en wordt aangegeven met de netmask.

Als je verbinding maakt met een netwerk (bijvoorbeeld het internet), kun je een statisch of dynamisch IP-adres hebben. Een statisch IP-adres is, zoals de naam al zegt, altijd hetzelfde. Dit adres kan worden bijgehouden in de configuratie van de netwerkkaart of worden toegewezen door een zogenaamde DHCP-server. Maar een DHCP-server is vooral bedoeld voor het bepalen van een dynamisch IP-adres.

IP-range Een IP-range is een serie IP-adressen, bijvoorbeeld de IP-adressen 133.155.4.0 t/m 133.155.4.255. De Internet Assigned Numbers Authority (IANA) beheert deze ranges. Er zijn drie IP-ranges gereserveerd die mogen worden gebruikt op een Local Area Network (LAN). Deze adressen worden niet toegekend aan wie dan ook.

- 10.0.0.0–10.255.255.255
- 172.16.0.0–172.31.255.255
- 192.168.0.0–192.168.255.255

Daarnaast wordt 127.0.0.1 altijd voor de lokale machine (localhost) gebruikt. 127.0.0.1 wordt ook wel de loopback device genoemd. Hij is nodig omdat sommige programma's niet werken zonder een netwerkverbinding.

Class A, B en C Als in een IP-range alleen het eerste getal vast (het netwerkgedeelte) is, dan bevat die IP-range erg veel IP-adressen (bijv. 23.0.0.0 t/m 23.255.255.255). Zo'n range wordt class A genoemd. Als de eerste twee getallen het netwerkgedeelte voorstellen, dan bevat het al minder IP-adressen. Zo'n range is class B. Zijn de eerste drie getallen het netwerkgedeelte, dan is het class C.

Netmask Door de netmask kan bepaald worden of een pakketje data voor een computer in het lokale netwerk (LAN) of WAN (Wide Area Network, een verzameling van verschillende LAN's) bestemd is. Is het bestemd voor buiten het interne netwerk, dan moet het pakket naar de gateway. De netmask bepaal je door alle IP's in de IP-range die je ter beschikking hebt binair te bekijken. Op alle plaatsen die niet veranderen staat een 1 in de netmask, op de andere plaatsen een 0. Een voorbeeld:

Je gebruikt de IP-range 192.168.1.0–192.168.1.255.

```
192.168.1.0 is binair 11000000 10101000 00000001 00000000
192.168.1.1 is binair 11000000 10101000 00000001 00000001
```

De eerste 16 binaire posities veranderen nooit in deze IP-range. Op al die plaatsen hoort een 1 in de netmask. Dus de netmask is 11111111 11111111 11111111 00000000. Decimaal is dat 255.255.255.0.

Je krijgt dus 255.0.0.0 voor class A, 255.255.0.0 voor class B en 255.255.255.0 voor class C.

Netwerk- en broadcastadres De laagste IP uit de IP-range is volgens afspraak de IP die bij het hele netwerk hoort, het netwerkadres. Het broadcastadres is vaak de hoogste IP in de range, al is dit adres wel vrij te kiezen. Pakketten die naar het broadcastadres worden gestuurd, komen op alle IP's in het netwerk aan.

Gatewayadres Dit is de IP van de computer die het netwerk aan het WAN (bijvoorbeeld internet) koppelt. Die computer wordt router of gateway genoemd.

Poort Om ervoor te zorgen dat elk programma een eigen verbinding kan maken (die het overige netwerkverkeer niet in de weg zit), gebruikt men poorten. Elke service op een machine heeft een eigen poort. Zo heeft FTP poort 21, SSH poort 22, telnet poort 23, en HTTP poort 80. Alle poortnummers onder de 1024 zijn ergens aan toegewezen. Een lijst van poorten en services kun je terugvinden in `/etc/services`. Programma's die geen poort onder 1024 toegewezen hebben gekregen, gebruiken een hoger poortnummer (tussen 1024 en 65535).

DNS-server Staat voor Domain Name Server. Je weet waarschijnlijk wel dat je in plaats van een IP-adres ook een naam kunt intikken in een webbrowser om een pagina te bekijken. Sterker nog, je doet waarschijnlijk nooit anders. Maar hoe weet de browser met welk IP-adres hij verbinding moet maken? Daar is een DNS-server voor. Die zet namen om in een IP-adres (en vice versa).

DHCP Staat voor Dynamic Host Configuration Protocol, een protocol voor het toekennen van dynamische IP-adressen. De meeste providers hebben een DHCP-server om beschikbare adressen uit te delen aan alle verbindingen. DHCP biedt de provider het voordeel van minder administratie, maar ook voor de gebruiker is dit eenvoudiger. Met DHCP wordt ook vanzelf de netmask ingesteld, de routing table vastgesteld en de DNS-server(s) bepaald.

10.3 Handige programma's

De volgende programma's zijn handig/nodig als je een netwerk of internetverbinding aan het opzetten bent.

ping Om te testen of er een verbinding is kun je het programma ping gebruiken. (Dit programma bestaat ook onder Windows.) Het gebruik van ping is eenvoudig: **ping IP/hostname**. Er worden dan kleine datapakketjes (ICMP echo requests) naar de machine verzonden waarop gereageerd wordt. Als uitvoer geeft ping hoe lang het duurt om een reactie te krijgen. Hier gaat de Linux-versie eindeloos mee door. Je moet het stoppen met `ctrl-c`. Dan worden nog wat statistieken getoond. Je kunt ook een aantal keer opgeven dat een ping moet worden verstuurd met de optie `-c#` (met `#` het aantal keer). Naast testen of een verbinding met een andere computer in het netwerk werkt, kun je er ook mee testen of de internetverbinding en DNS-server goed werken.

ifconfig Zonder argumenten mee te geven, laat dit programma zien welke verbindingen er actief zijn en o.a. welk IP eraan is toegekend. Je kunt verbindingen starten en stoppen met **ifconfig naam up/down**. In plaats van naam gebruik je bijv. `eth0` of `ppp0` (voor een modemverbinding).

nslookup Met dit programma kun je nagaan hoe een "name server lookup" verloopt. Typ je bijvoorbeeld **nslookup www.google.nl**, dan zul je een lijstje krijgen van de weg die jouw request aan de DNS-server doorloopt.

10.4 Hosts en resolv.conf

Binnen een lokaal netwerk kun je lokaal te gebruiken hostnamen opgeven per IP in `/etc/hosts`. Een voorbeeldinhoud:

```
127.0.0.1 localhost
192.168.1.1 piet
192.168.1.2 frits
192.168.1.3 karel
```

Hierbij ga je ervanuit dat alle machines een statisch IP-adres hebben. De ranges van IP-adressen die hiervoor geschikt zijn werden in sectie Netwerkbegrippen genoemd. Begin de adressen vanaf `.1` op het eind en niet vanaf `.0`. Dat zou het netwerkadres zijn en dat werkt niet. Je kunt hier ook meerdere namen voor je eigen machine (`localhost`) opgeven.

In het geval van DHCP hoef je het adres van de DNS-server(s) niet zelf op te geven. In overige gevallen zul je de domeinnaam en DNS-server(s) van je provider zelf in `/etc/resolv.conf` moeten zetten. Een voorbeeld:

```
search provider.nl
nameserver 195.96.96.97
nameserver 195.96.96.33
```

10.5 Masquerading firewall

Als je een lokaal netwerk hebt, wil je wellicht ook wel met het hele netwerk gebruik kunnen maken van internet. Als een van de computers in het netwerk een verbinding heeft met je provider, dan kun je die als gateway gebruiken voor de rest. Wat je wilt is een masquerading firewall draaien op die computer. Dan kun je op alle computers in het netwerk gebruiken maken van internet, maar het lijkt van buitenaf alsof al het verkeer van/naar de computer met de internetverbinding komt/gaat. (En in feite is dat ook zo, want alles gaat langs die computer.)

Naast ondersteuning voor een firewall in de kernel heb je nog een programma nodig om daadwerkelijk regels te kunnen opgeven. Bij 2.4.x- en 2.6.x-kernels kun je daarvoor iptables (<http://www.iptables.org>) gebruiken. Een eenvoudige regel:

```
iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
```

Op de plaats van eth0 vul je de device in die bij jou de internetverbinding verzorgt. Waarschuwing: Dit vormt een minimale configuratie die totaal geen veiligheid biedt!

Daarnaast moet er nog tegen de kernel gezegd worden dat IP forwarding moet worden geactiveerd:

```
echo 1 > /proc/sys/net/ipv4/ip_forward
```

Tip: Dezelfde functionaliteit is ook te bereiken met veel routers. Zonder zouden ze niet dezelfde internetverbinding kunnen aanbieden aan verschillende computers binnen hetzelfde huis. Vaak zit er ook een DHCP-server ingebouwd om alle computers een intern IP-adres te geven, zonder zelf iets te hoeven instellen.

10.6 Samba

Samba (<http://www.samba.org>) is software om bestanden (en printers) te kunnen delen met Windows. Het bestaat uit een client en server.

Om een verbinding met een Samba-server te maken (en verbreken) zijn er drie programma's: **smbmount**, **smbumount** en **smbclient**. Een directory die wordt geshared op een Samba-server of Windows-systeem mount je net als een partitie van je eigen harde schijf. Je hebt er dus ook een mount point voor nodig. Maak bijv. `/mnt/samba` aan. Je mount daar dan een gesharede directory onder met **smbmount //machinenaam/share /mnt/samba**. Er wordt dan gevraagd om een wachtwoord. Als er guest access is (dus als er geen wachtwoord nodig is), kun je daar simpelweg op enter drukken. Je kunt in dat geval ook de optie `-o guest` meegeven om niet om een wachtwoord te laten vragen. Om een overzicht van shares op een bepaalde machine te krijgen, kun je **smbclient -L //machinenaam** gebruiken. Wil je een directory weer unmounten, dan gebruik je **smbumount /mnt/samba**. Smbumount mislukt echter als de machine waar de gesharede directory op stond in de tussentijd werd uitgezet of is gecrasht. In dat geval kun je **umount /mnt/samba** proberen.

Naast mounten kun je ook met het programma **smbclient** een share bekijken en bestanden downloaden en uploaden op dezelfde manier als bij een FTP-server.

Met de file managers van KDE en GNOME (resp. Konqueror en Nautilus) kun je veel handiger shares

benaderen zonder ze te hoeven mounten. Voer daarvoor in de adresbalk `smb://machinenaam/share` of `smb://machinenaam` in.

Het configuratiebestand van de server is `/etc/smb.conf` of `/etc/samba/smb.conf`. Wat je in dit bestand moet zetten, wordt uitgelegd in de manual page (**man smb.conf**). Een voorbeeldconfiguratie:

```
[global]
  workgroup = THUISNETWERKJE
  server string = Samba server
  hosts allow = 192.168.1. 127.
  security = share
  socket options = TCP_NODELAY SO_RCVBUF=8192 SO_SNDBUF=8192
  dns proxy = no

[public]
  path = /home/shares
  public = yes
  writable = no
```

Deze configuratie geeft iedereen in de IP-range 192.168.1.x zonder wachtwoord toegang tot `/home/shares`, maar zonder schrijfrechten.

10.7 Sendmail en fetchmail

De klassieke manier om mail te versturen en ontvangen is met Sendmail. Ontvangen mail wordt opgeslagen in `/var/spool/mail/username`. Programma's als Mutt en Pine (beide goede tekstgebaseerde mailprogramma's) gaan uit van zo'n situatie. Veel mensen hebben echter niet hun computer dag en nacht aan en met een actieve internetverbinding, wat nodig is altijd mail te kunnen ontvangen. Verder heeft men meestal ook geen eigen domeinnaam, zodat een IP-adres zou moeten worden gebruikt in het e-mailadres. Daarom gebruiken veel mensen de mailservers van een ander systeem, typisch die van hun ISP (internetprovider). Om mail van die server op te halen, kun je fetchmail (<http://fetchmail.berlios.de>) gebruiken. Fetchmail kun je configureren met **fetchmailconf**. Handmatig configureren is ook niet zo moeilijk. Zet daarvoor in `~/fetchmailrc` iets als:

```
poll pop.gmx.net with proto POP3
user "gebruikersnaam" there with password geheim is jarkko here warnings 3600
```

Vervang hierin `pop.gmx.net` door de mailservers die je wilt, POP3 door het te gebruiken protocol, gebruikersnaam door de gebruikersnaam om in te loggen, geheim door het te gebruiken wachtwoord en `jarkko` door de gebruikersnaam van de persoon waar de mail voor is. Je kunt je mail eenmalig laten ophalen met **fetchmail**. Met de extra optie `--daemon interval` kun je fetchmail in daemon-mode laten werken. Vervang interval door het aantal seconden dat er gewacht moet worden om de mail opnieuw te controleren.

Meer over fetchmail is te vinden in de manual page.

Nu heb je alleen nog het probleem dat de mail die je verstuurt de hostname van de machine waar je op werkt als afzender heeft. En dat wil je juist niet als je computer niet altijd aan staat. (Dan zou mail niet ontvangen worden als je computer toevallig uit staat, omdat de mailservers dan niet bereikbaar is.) Je kunt Sendmail zo configureren dat hij de afzender herschrijft. Hoe je dat precies doet, staat beschreven in de Sendmail address rewrite mini-HOWTO. Iets eenvoudiger (maar niet centraal geregeld voor alle gebruikers) is dit probleem verplaatsen naar het gebruikersniveau. De meeste mailprogramma's kunnen zelf al een andere afzender opgeven dan huidigegebruiker@hostname. Voor Mutt kun je bijvoorbeeld in `~/muttrc` het volgende opnemen:

```
my_hdr From: naam <user@host.com>  
my_hdr Return-Path: naam <user@host.com>
```

Vervang daar user@host.com door de gewenste afzender. Bij Pine kan ditzelfde gedaan worden door de volgende regel op te nemen in ~/.pinerc:

```
customized-hdrs=From: naam <user@host.com>  
customized-hdrs=Return-Path: naam <user@host.com>
```

Bovenstaande is niet van toepassing als je gebruikmaakt van een mailclient als Thunderbird. Daar wordt niet uitgegaan van een lokale mailservers, maar kun je een externe instellen.

11 Security

11.1 Introductie

Virussen zijn niet iets waar je je onder Unices zorgen over hoeft te maken. Een veel reëler probleem vormen bugs in software die uit te buiten zijn met een zg. exploit. Dit is vooral gevaarlijk als de software als root draait, zoals Sendmail en andere daemons. Mogelijk kan iemand dan via de exploit elke gewenste code laten uitvoeren, zodat hij bijv. een root shell kan krijgen.

Een korte checklist van veiligingspunten:

- Houd je software up-to-date.
- Schakel onnodige services/daemons uit.
- Gebruik een firewall om toegang zoveel mogelijk te blokkeren.
- Gebruik goede wachtwoorden.
- Gebruik SSH om remote in te loggen.
- Geef anderen zo weinig mogelijk informatie over je systeem.
- Controleer regelmatig je logbestanden op verdachte zaken.

11.2 Rootkits

Een rootkit is een set programma's om jezelf onzichtbaar te houden na het verkrijgen van root access op een machine. Veelal bevat het alternatieve versies van ls, df, ps e.d. waardoor de eigenaar van de machine niks vreemds opmerkt. Met chkrootkit (<http://www.chkrootkit.org>) kun je je systeem controleren op aanwezigheid van een serie bekende rootkits. Je kunt dit gebruiken als eerste controle als je vermoedt dat er rare dingen aan de hand zijn op je systeem.

11.3 Wachtwoorden

Op oudere systemen werden alle wachtwoorden in het voor iedereen leesbare bestand `/etc/passwd` opgeslagen. Op nieuwere systemen worden ze opgeslagen in een alleen voor root leesbaar bestand: `/etc/shadow`. Op de plaats waar je in `/etc/passwd` het wachtwoord zou verwachten, staat dan alleen nog maar een x.

Het wachtwoord is geëncrypteerd opgeslagen, maar als iemand hem zou weten te bemachtigen is er toch een kans dat het te achterhalen is. Een methode daarvoor is om een hele rij mogelijkheden af te gaan, elke optie te encrypteren en te vergelijken met de bemachtigde versie. Dit is moeilijker te maken door een goed wachtwoord te kiezen en shadow maakt het bemachtigen ook weer wat lastiger.

Slechte wachtwoorden zijn bijv. het woord password zelf, je naam, gebruikersnaam, naam van een familielid, vriend(in) of huisdier, naam van je bedrijf, geboortedatum en elk willekeurig woord dat je in een woordenboek kunt vinden. Gebruik ook het liefst niet hetzelfde wachtwoord op meerdere plaatsen. Dat is wel makkelijker te onthouden, maar is niet erg veilig. Een goed wachtwoord is relatief lang (minstens 6 tekens) en bevat een combinatie van hoofd- en kleine letters en cijfers. Verander je wachtwoord ook om de zoveel tijd.

11.4 Inloggen zonder wachtwoord

Als je het wachtwoord van root vergeten bent (of iemand heeft in je systeem ingebroken en het wachtwoord veranderd), dan kun je het veranderen door achter de LILO-promp `linux init=/bin/sh` in te tikken (vervang

linux door het juiste label, aangegeven in `/etc/lilo.conf`). Bij GRUB kun je op de regel van de gewenste kernel gaan staan en op `e` drukken om de optie toe te voegen. Je start nu een shell zonder in te hoeven loggen! De root-partitie zal read-only gemount zijn. Typ `mount -o remount,rw /` om het te hermounten als read-write. Heb je `/usr` op een aparte partitie staan, mount die dan ook. Bij sommige distributies kan `fstab` niet goed gelezen worden als `proc` niet gemount is. Mislukt het mounten van `/usr`, doe dan eerst nog `mount -t /proc none /proc`. Nu kun je het wachtwoord veranderen met `passwd` (dit programma staat in `/usr/bin`). Log niet meteen uit, want dan zijn de partities niet netjes gemount. Doe eerst `umount /usr` (indien nodig) en dan `mount -o remount,ro /`. Daarna kun je veilig uitloggen en rebooten met `ctrl-d`.

Deze mogelijkheid geeft inderdaad een behoorlijk veiligheidslek als mensen fysiek bij een systeem kunnen.

11.5 (X)inetd, hosts.allow en hosts.deny

Inetd is de "internet super-server" en werkt als volgt. Tijdens het booten wordt inetd gestart en leest daarbij zijn configuratiebestand `/etc/inetd.conf` in. Daarin staat op welke poorten hij moet luisteren en welke server/daemon hij moet starten als er een verbinding wordt gelegd met die poorten. Op die manier hoeven niet alle mogelijke servers constant actief te zijn. Een nadeel van inetd is dat het even tijd kost voordat een server is gestart. Als er dus veel verbindingen binnenkomen (bijvoorbeeld op een druk bezochte webserver), is het beter die server stand-alone te gebruiken. Xinetd is een wat efficiënter en veiliger alternatief. Zijn configuratie staat in `/etc/xinetd.conf`.

Schakel voor de veiligheid alle services die je niet gebruikt uit. Dat doe je door in `/etc/(x)inetd.conf` de desbetreffende regels te verwijderen of uit te commenten met `#`-tekens vooraan de regels. Services die stand-alone gestart worden, worden gestart via de runlevel-scripts. Die uitschakelen kan meestal via een toeltje van je distributie of met de hand door de desbetreffende symlink naar het script te verwijderen. Zie ook paragraaf 8.1.

Toegang tot services regel je met twee bestanden in `/etc`: `hosts.allow` en `hosts.deny`. In `hosts.deny` geef je aan welke hosts geen toegang hebben en in `hosts.allow` wie je wel toelaat. Je kunt alle toegang dichtgooien door in `hosts.deny` de regel `ALL:ALL` op te nemen en vervolgens in `hosts.allow` aan te geven welke uitzonderingen hierop gemaakt mogen worden.

Een aantal voorbeelden voor in `hosts.allow`:

```
# geef localhost en intern netwerk (adressen die beginnen met 192.168.1) toegang
ALL : 192.168.1. 127.0.0.1 localhost
# SSH alleen van te vertrouwde hosts
sshd: 100.200. .trusted.nl
# toegang tot FTP beperken
in.ftpd: .telekabel.euronet.nl localhost 192.168.1
```

Zoals je ziet geeft een punt voor of achter een IP of domeinnaam aan dat daar alles voor of achter mag komen. Daarnaast zijn er nog twee mogelijke constructies: `131.155.72.0/131.155.73.255` (van `131.155.72.0` tot en met `131.155.73.255`) en `.trusted.nl EXCEPT nuker.trusted.nl` (maak een uitzondering).

Meer informatie vind je in de manual pages van `(x)inetd`, `(x)inetd.conf` en `hosts.allow` (is dezelfde als die van `hosts.deny`).

11.6 Firewall

Om een firewall te kunnen opzetten moet daar ondersteuning voor in de kernel zitten of als module beschikbaar zijn. Aast ondersteuning in de kernel heb je nog een programma nodig om daadwerkelijk firewallregels te kunnen opgeven. Voor de oude 2.2.x-kernels was dat `ipchains`. Vanaf kernel 2.4.x gebruik je daarvoor `iptables` (<http://www.iptables.org>).

Regels van een firewall stellen restricties aan de pakketjes die over het netwerk gestuurd en ontvangen worden. Er kan ook mee geregeld worden welke pakketjes doorgestuurd moeten worden naar een achterliggend netwerk. Je kunt een firewall gebruiken om een modem te sharen, maar ook om een webserver wereldwijd bereikbaar te maken terwijl de server niet direct met het internet verbonden is.

Informatie over hoe je de firewall gebruikt, is te vinden op de website van iptables en in verschillende tutorials online.

Met Nmap (<http://www.insecure.org/nmap/>) kun je laten zoeken naar poorten die open staan. Om een betrouwbaar resultaat te krijgen, moet je je door een ander laten testen.

11.7 Sudo

Sommige programma's kunnen normaal gesproken alleen door root uitgevoerd worden. Voorbeelden zijn smbmount en cdrecord. Maar je wilt ze waarschijnlijk ook wel als normale gebruiker kunnen uitvoeren. Een mogelijke oplossing is die programma's SUID-root maken (zie sectie Permissies en eigenaren), maar meer controle kan worden uitgeoefend via Sudo. Met behulp van Sudo kun je normale gebruikers op een veilige manier programma's laten uitvoeren alsof ze door root zouden worden uitgevoerd.

Sudo configureren doe je niet door direct het configuratiebestand `/etc/sudoers` aan te passen, maar door (als root) **visudo** uit te voeren. (Standaard zal dit Vim starten. Zie paragraaf 20.24 over hoe je een andere in kunt stellen.) Een voorbeeldconfiguratie:

```
Host_Alias  HOSTNAME = huijts
User_Alias  USERS = jarkko
Runas_Alias OP = root
Cmd_Alias  CDBURN = /usr/bin/cdrecord
Cmd_Alias  SAMBA = /usr/bin/smbmnt, /usr/bin/smbmount, /usr/bin/smbumount

# Logbestanden
Defaults    syslog=auth
Defaults@HUIJTS log_year, logfile=/var/log/sudo.log

jarkko ALL = NOPASSWD:CDBURN, SAMBA
```

Met deze configuratie kan de gebruiker jarkko door **sudo cdrecord** in te tikken cdrecord uitvoeren alsof root het startte.

Het standaard gedrag van Sudo is om te vragen naar een wachtwoord. Dan moet je je eigen wachtwoord invoeren en niet die van root. Sudo onthoudt een bepaalde tijd dat je het wachtwoord hebt ingevoerd, zodat je het niet nog een keer hoeft in te tikken als je het tweemaal snel achter elkaar gebruikt. Met de optie **NOPASSWD** in het configuratiebestand hoeft de gebruiker geen enkel wachtwoord in te voeren.

Zie de manual pages van Sudo en sudoers voor meer informatie.

11.8 SSH

Als je een telnet-server hebt draaien kun je met een telnet-client ergens anders op je Linux-systeem inloggen en een shell verkrijgen. Bij telnet wordt de informatie die over en weer gezonden wordt niet gecodeerd. Met behulp van zogenaamde packet sniffers is deze informatie door anderen vrij gemakkelijk te onderscheppen. Ook het wachtwoord dat je intikt om in te loggen is op die manier te zien! Veiliger is SSH, Secure SHell. Je kunt er hetzelfde mee als met telnet, maar de data wordt gecodeerd overgezonden. Een goede open-sourcevariant van SSH is OpenSSH (<http://www.openssh.com>). Een goede SSH-client voor Windows is PuTTY (<http://www.chiark.greenend.org.uk/~sgtatham/putty/>).

Naast inloggen en een shell verkrijgen, kun je met SSH ook bestanden kopiëren. Dit maakt het dus ook meteen onnodig hier een FTP-server voor te gebruiken. Gebruik **scp user@host:/path/file .** om een bestand te kopiëren naar de huidige directory. Zie de manual page van scp voor meer informatie. Met de file managers van KDE en GNOME (resp. Konqueror en Nautilus) kun je ook naar het adres sftp://hostname gaan.

12 Software

12.1 Overzichten en hosting

Open-sourceprojecten worden gedaan door verschillende individuen en groepen. Er is daarmee geen centrale plaats waar je alle projecten kunt vinden. Een klassieke overzichtssite is <http://freshmeat.net>.

Tegenwoordig worden aardig wat projecten gehost door SourceForge (<http://sourceforge.net>). Zij bieden goede voorzieningen om open-sourceprojecten te hosten, waar duizenden projecten dankbaar gebruik van maken. De URL's van projectpagina's eindigen op `.sourceforge.net` of `sf.net`.

GNU Savannah is vergelijkbaar, maar heeft een strikter beleid. Alleen vrije software mag er worden gehost. De site is in twee domeinen verdeeld, <http://savannah.gnu.org> en <http://savannah.nongnu.org>, waarbij de tweede duidelijk voor projecten is die niet bij GNU zelf horen.

Verder is een overzicht van software voor GNOME te vinden op <http://gnomefiles.org> en software voor KDE op <http://www.kde-apps.org>.

12.2 Source code

Van alle open-sourcesoftware is de source code te downloaden. Je kunt het onder elke distributie op dezelfde wijze compileren en installeren. Dit wil je niet doen voor alle software op je systeem. Gebruik als het even kan het pakketsysteem van je distributie. De gebruikelijke pakketsystemen worden in de volgende secties behandeld. Dat gezegd hebbende wil je wellicht wel eens iets zelf compileren, bijv. omdat de software niet via je pakketsysteem beschikbaar is.

De meeste source code die je kunt downloaden is in `.tar.gz`- of `.tgz`-formaat, zogenaamde tarballs. Je kunt deze bestanden uitpakken en dearchiveren met `tar zxvf filename.tar.gz`. Meestal is er dan een nieuwe directory met ongeveer dezelfde naam als de tarball aangemaakt. Ga naar die directory. In een bestand als `INSTALL` of `README` staat gebruikelijk de benodigde uitleg en mogelijke afhankelijkheden. De meeste software is in C of C++ gemaakt. Daarvoor moet de GNU C compiler gcc geïnstalleerd zijn. In negen van de tien gevallen komt het op het volgende neer:

- `./configure`
- `make`
- `su`
- `make install`
- `exit`

Configure is een script dat informatie verzamelt die nodig is tijdens het compileren van de software. Make leest het bestand Makefile in en compileert het programma volgens de regels die daarin staan. In makefiles is gebruikelijk ook een regel opgenomen om met `make uninstall` de installatie weer ongedaan te maken. Dan kun je dus, als je de source code houdt, de software op een vrij eenvoudige wijze weghalen.

12.3 Het systeem van Slackware

Het pakketsysteem van Slackware is het eenvoudigste van allemaal. De pakketten zijn tarballs met twee extra bestanden: een installatiescript en een beschrijving van het pakket. Er is geen controle op afhankelijkheden. Daar moet je zelf maar voor zorgen.

Om een pakket te installeren gebruik je **installpkg pakket.tgz**. Om een reeds geïnstalleerd pakket te upgraden gebruik je **upgradepkg pakket.tgz**. Een pakket verwijderen kan met **removepkg pakketnaam**. Bij dit laatste commando hoef je geen bestandsnaam op te geven als argument, maar slecht de basisnaam van het pakket.

Het programma **pkgtool** geeft je toegang tot een menu-interface voor pakketbeheer. De belangrijkste opties zijn **remove** en **setup**. Met **setup** kun je allerlei configuratietooltjes starten.

12.4 Het systeem van Debian

Dit systeem is oorspronkelijk ontwikkeld voor Debian en wordt ook gebruikt bij op Debian gebaseerde distributies zoals Ubuntu.

Het pakketsysteem van Debian maakt gebruik van deb-bestanden. Het basisprogramma om deze pakketten te installeren is **dpkg**. Gebruik **dpkg -i bestand.deb** om een pakket los te installeren. Voor de overige opties zie **man dpkg**. Van oudsher heeft Debian het best werkende pakketsysteem van alle distributies. Dit komt door een laag bovenop dpkg: apt (advanced package tool), waar je normaal gesproken alles mee doet.

In het configuratiebestand `/etc/apt/sources.list` zet je een of meer locaties waar zg. pools met Debian-pakketten gevonden kunnen worden. In deze pools wordt gezocht naar pakketten. Een voorbeeldregel:

```
deb http://ftp.nl.debian.org/debian/ testing main non-free contrib
```

Nadat je een bron hebt toegevoegd, moet je **apt-get update** uitvoeren om een lijst te laten ophalen met wat er in de bronnen te vinden is.

Na de locatie geef je op welke tak je van Debian wilt hebben: **stable**, **testing** of **unstable**. Pakketjes beginnen in **unstable**. Daar staan dus hele nieuwe dingen in. Als het uitgetest wordt, dan wordt het verplaatst naar **testing**. Is het na grondige tests stabiel genoeg gebleken, dan komt het pas in **stable** terecht. De **stable**-tak heeft altijd redelijk verouderde software. Wat bij Debian nog in **testing** staat, zou bij andere distributies zonder meer in een release gezet worden. De meeste mensen willen **testing** als basis gebruiken. Elke tak (**stable**, **testing** en **unstable**) bestaat weer uit verschillende onderdelen: **main**, **non-free**, **non-us** (software die in de US niet gebruikt mag worden, zoals software met sterke encryptie) en **contrib** (contributed software).

Om iets van de pools te installeren gebruik je **apt-get install pakketnaam** met i.p.v. pakketnaam de naam van een pakket. Pakketten die ook nodig zijn voor het te installeren pakket (alle afhankelijkheden waar niet aan is voldaan) worden ook meteen meegenomen. Dat is precies waardoor het veel handiger is dan alleen dpkg los. Je hoeft zelf niet alle afhankelijkheden op te lossen. De apt-tools bieden deze mogelijkheid en de reden waarom het werkt is dat de vrijwilligers die aan Debian meewerken veel pakketten beschikbaar stellen met goede dependency-informatie.

Weet je de exacte naam van een pakket niet, dan kun je die zoeken met **apt-cache search zoekterm**. Met **apt-get** kun je ook gemakkelijk je hele systeem bijwerken met alle updates die beschikbaar zijn. Wanneer je **apt-get update** intikt, zal de lijst met beschikbare pakketjes geüpdatet worden en vervolgens kun je met **apt-get upgrade** alles laten bijwerken.

Meer informatie over pakketbeheer in Debian kun je vinden in de apt HOWTO (<http://www.debian.org/doc/manuals/apt-howto/>). Je kunt veel informatie opvragen over de pakketten die je hebt en die beschikbaar zijn. Het is mogelijk apt-bronnen voor **stable**, **testing** en **unstable** tegelijk bij te houden en verschillende software uit verschillende takken te halen. Verder kun je pakketten on hold zetten, zodat het niet geüpdatet zal worden. Het voert te ver dat hier allemaal uit te leggen.

Een goede grafische frontend voor apt is Synaptic. Bij Ubuntu wordt deze standaard gebruikt.

12.5 RPM

RPM is een pakkeetsysteem dat oorspronkelijk is ontwikkeld voor Red Hat, maar nu door vele distributies zoals SUSE en Mandriva wordt gebruikt. RPM stond oorspronkelijk voor Red Hat Package Manager, maar de uitleg is veranderd in RPM Package Manager, een recursief acroniem (net zoals GNU).

Het basisprogramma om RPM-pakketten te installeren is **rpm**. Je kunt een pakket installeren met **rpm -ivh foo.rpm** en upgraden met **rpm -Uvh foo.rpm**. (Met de opties **v** en **h** zie je de voortgang.) Zie voor meer opties de manual page.

In de begindagen van RPM kon je alleen rpm direct gebruiken op de hiervoor uitgelegde manier. Rpm is vergelijkbaar met dpkg bij Debian. Er zijn inmiddels lagen bovenop rpm gemaakt die vergelijkbaar zijn met apt bij Debian: een tool die rpm gebruikt, maar de afhankelijkheden voor je uitzoekt. Niet elke distributie gebruikt hier dezelfde tool voor. Mandriva gebruikt URPMI, Fedora YUM en SUSE heeft vergelijkbare functionaliteit in zijn algemene configuratietool YaST.

12.6 Het systeem van Gentoo

Gentoo heeft een ander pakkeetsysteem dan de meeste andere distributies. Deb- en RPM-pakketten bevatten voorgecompileerde software. Bij Gentoo compileer je alle software zelf, maar gecontroleerd via het pakkeetsysteem. Het pakkeetsysteem heet Portage en is geïnspireerd door BSD. Portage biedt vooral grote flexibiliteit.

Open-sourcesoftware heeft vaak mogelijkheden die je wel of niet kunt meecompileeren. De pakketten die je bij pakketgebaseerde distributies installeert, hebben vaak ondersteuning voor alles of voor wat de meeste gebruikers nodig zullen hebben. Bij Gentoo kun je volledig zelf bepalen wat je wel en niet wilt hebben. Dit doe je door middel van de variabele USE. Hiermee kun je bijv. aangeven dat je ondersteuning voor Samba en GTK+ wilt, maar niet voor Qt. De USE-variabele en andere opties voor tijdens het compileren van de software zet je in `/etc/make.conf`.

Portage gebruik je voornamelijk via het commando **emerge**. Net als voor de meeste andere distributies zijn er voor Gentoo pools met pakketten. Daar wordt de broncode van gedownload. Je kunt een gewenste mirror opgeven in `make.conf`. Gebruik **emerge --sync** om Portage informatie over beschikbare pakketten op te laten halen. Je kunt dit om de zoveel tijd uitvoeren om nieuwe versies te laten bekijken.

Een pakket installeren is erg eenvoudig: **emerge pakketnaam**. Ook alle afhankelijkheden worden daarbij uitgezocht. Het kan daardoor zijn dat er meer pakketten worden opgehaald. Wil je eerst zien wat emerge gaat doen, gebruik dan **emerge -pv pakketnaam**. De **p** staat voor pretend en de **v** voor verbose. Weet je niet hoe het pakket heet, dan kun je zoeken met **emerge --search zoekterm**.

Zie de documentatie van Gentoo voor meer uitleg over Portage.

13 X windowing system

13.1 Introductie

Het X Windowing System werd in 1983 door MIT ontwikkeld. Vanaf versie 10 uit 1986 begon het de standaard te worden voor Unix. De laatste versie is versie 11 en wordt ook wel X11 genoemd. Je kunt het X, X11 of het X Windowing System noemen, maar X Windows is geen officiële naam.

X is de basis voor een grafische desktop onder Unices. Het is netwerktransparant en vormt een hardwareonafhankelijk platform om GUI's op te bouwen, maar dicteert niet hoe de UI eruit moet zien. De "look and feel" van grafische software wordt grotendeels bepaald door de gebruikte toolkit. X heeft geen ingebouwde mogelijkheden om windows te managen. Daar is een apart proces voor nodig, genaamd een window manager. Over beide verderop meer.

Op een gegeven moment is X11 closed source geworden, maar lange tijd was het dat niet. XFree86 was een fork voor x86-machines. Het is gebaseerd op de originele code van X11 van voor de licentieverandering. Lange tijd werd XFree86 gebruikt door Linux-distributies, maar inmiddels is iedereen overgestapt naar X.org, wat weer een fork is van XFree86. De directe reden voor de overstap was een verandering in licentie (ja, geschiedenis herhaalt zich) waardoor het incompatibel werd met GPL, maar eigenlijk was dat meer de laatste druppel. Er heerste al langer ongenoegen over de voortgang en ontwikkeling. Veel ontwikkelaars en gebruikers waren van mening dat het ontwikkelproces niet voldoende open en transparant was. Keith Packerd, een van de belangrijkste ontwikkelaars, is samen met andere leden van het core team van XFree86 overgestapt op X.org.

Een recente ontwikkeling is die om 3D-effecten op de desktop te kunnen gebruiken. Het idee hierachter is dat elke computer tegenwoordig wel is uitgerust met een videokaart met 3D-acceleratie. Die mogelijkheden kunnen naast voor spellen ook op de desktop gebruikt worden. Dat stelt je in staat interessante effecten zoals transparantie toe te passen op je desktop, maar zorgt er ook voor dat je efficiënt gebruik kunt maken van de mogelijkheden van je videokaart.

13.2 Drivers

X heeft een driver nodig voor je videokaart. Dit is een van de weinige stukken hardware waar geen device driver voor in de kernel zit. Voor kaarten met een chipset van ATI (<http://www.ati.com>) en NVIDIA (<http://www.nvidia.com>) bestaan zowel binary/propriëtaire als open-sourcedrivers. Indien je de binary drivers wilt/moet gebruiken, kijk dan eerst of er een eenvoudig mechanisme is om ze te installeren onder je distributie. Is dat niet het geval, dan kun je ze van de respectieve websites downloaden. Installatie-instructies staan ook op beide sites.

Voor NVIDIA heb je de binary driver nodig voor 3D-acceleratie. Voor ATI is dat niet het geval, aangezien de open-sourcedriver daar wel ondersteuning voor heeft. Dit komt doordat de benodigde hardwarespecificaties zijn vrijgegeven sinds ATI is overgenomen door AMD. De open-sourcedrivers hebben echter geen ondersteuning voor de nieuwste videokaarten, wat een reden kan zijn om toch de binary driver nodig te hebben. Aangezien het landschap van drivers en ondersteuning van videokaarten vrij snel verandert, verwijs ik voor meer informatie naar Google.

13.3 Configuratie

Het configuratiebestand van X.org is `/etc/X11/xorg.conf`. Het bestaat uit een aantal blokken die beginnen met Section. In de meeste distributies zal X tijdens de installatie waarschijnlijk al geconfigureerd zijn. De volgende punten kunnen interessant zijn.

Voor je muis is er een sectie `InputDevice`:

```
Section "InputDevice"
  Identifier "Mijn muis"
  Driver "mouse"
  Option "Protocol" "ImPS/2"
  Option "Device" "/dev/input/mice"
  Option "ZAxisMapping" "4 5"
EndSection
```

Sinds kernel 2.6 is `/dev/input/mice` de device voor alle aangesloten muizen samen. Het protocol `ImPS/2` is een uitbreiding van het oude `PS/2`-protocol voor scrollwielletjes en zal werken voor de meeste muizen. De regel met `ZAxisMapping` zorgt ervoor dat het scrollwielletje input genereert (als knoppen 4 en 5).

Ook voor je toetsenbord is er zo'n sectie:

```
Section "InputDevice"
  Identifier "Mijn toetsenbord"
  Driver "kbd"
  Option "CoreKeyboard"
  Option "XkbRules" "xorg"
  Option "XkbModel" "pc104"
  Option "XkbLayout" "us"
EndSection
```

Het meest gangbare model toetsenbord is er een met 104 toetsen, waarvoor `pc104` goed is. Er zijn vaak verschillende varianten van hetzelfde soort model. Zo zijn er bijvoorbeeld een Duitse variant (neem daarvoor `de`) en `Dvorak` (neem daarvoor `dvorak`). In Nederland komt de Amerikaanse indeling het meeste voor. Daarvoor is `us` goed. Een interessante indeling is `us_intl`. Als je die gebruikt, kun je letters met accenten intikken door eerst een bepaalde toets in te tikken voor dat accent en vervolgens de letter zelf. De tilde geeft bijvoorbeeld een golfje boven de letter en de apostrof een streepje naar rechts (accent `aigue`). Alle beschikbare indelingen gedefinieerd in bestanden onder `/etc/X11/xkb/symbols`. Je kunt de toetsenbordinstellingen ook tijdens het werken met X veranderen met `setxkbmap`, bijv. `setxkbmap us`. Bij GNOME als KDE zitten ook tooltjes om de layout in te stellen.

Voor je monitor is er een sectie `Monitor`:

```
Section "Monitor"
  Identifier "Mijn monitor"
  HorizSync 30.0 - 81.0
  VertRefresh 56.0 - 75.0
EndSection
```

Van moderne LCD-schermen kan hardwarematig informatie worden opgevraagd. Bij oude CRT-schermen was dat niet het geval, zodat er databases werden gebruikt met o.a. de horizontale sync rate en verticale refresh rate die je hierboven ziet. Mocht het nodig zijn deze waarden met de hand te zetten, dan kun je het terugvinden in de technische specificaties van je monitor. Iets gerelateerd wat je in deze sectie terug kunt vinden zijn modelines, regels met timinginformatie per mogelijke resolutie. Voornamelijk voor CRT-schermen kon het nodig zijn deze aan te passen om het beeld gecentreerd te krijgen of een betere refresh rate te halen. Over het algemeen zul je ze echter niet meer nodig hebben.

Voor je videokaart is er een sectie `Device`:

```
Section "Device"
  Identifier "Mijn videokaart"
  Driver "nvidia"
EndSection
```

Hier is de driver te vinden die zal worden gebruikt. Voor NVIDIA staat hier ofwel `nv` voor de open-sourcedriver of `nvidia` voor de binary driver.

In een sectie `Screen` worden je videokaart en monitor aan elkaar geknoopt:

```
Section "Screen"
    Identifier      "Mijn scherm"
    Device          "Mijn videokaart"
    Monitor         "Mijn monitor"
    DefaultDepth    24
    SubSection      "Display"
        Depth       24
        Modes        "1280x1024" "800x600"
    EndSubSection
EndSection
```

Er wordt hier een standaard kleurendiepte van 24 bits ingesteld. Dat is de hoogst mogelijke en gelijk aan wat enigszins misleidend 32 bits wordt genoemd onder Windows. Per kleurendiepte kunnen er subsecties `Display` worden opgegeven, met daarin een rij mogelijk resoluties. De waarde die daarbij vooraan staat, wordt standaard genomen.

Vroeger was het niet echt mogelijk de resolutie van je desktop te veranderen zonder X te herstarten. Als er meerdere resoluties zijn opgegeven in de configuratie van X, dan kun je daartussen wisselen met `ctrl, alt` en `+/-` (tegelijk indrukken). Daarbij blijft de originele resolutie wel behouden als "virtuele" resolutie. Ga je over naar een lagere resolutie, dan zal de desktop de originele grootte behouden en daarmee groter zijn dan wat er op het scherm past. Je kunt over deze grotere desktop schuiven door met de muis bij de randen van het scherm te komen.

Nieuwere versies van X kunnen de resolutie wel echt veranderen terwijl X bezig is. Dit kan via de `RandR`-extensie. Via tooltjes van GNOME en KDE kun je de resolutie instellen, maar het kan ook met `xrandr`. Gebruik `xrandr --query` voor een overzicht met beschikbare modes. Met `xrandr --size 12801024` kun je de resolutie bijv. veranderen in 12801024.

13.4 Window managers en desktopomgevingen

Een window manager decoreert de windows, d.w.z. tekent de randen en maakt het mogelijk windows te verplaatsen of van grootte te vanderen. De exacte mogelijkheden hangen af van de window manager die je gebruikt. Van oudsher zit bij X TWM (Tab Window Manager) meegeleverd, maar ik weet vrij zeker dat je die niet wilt gebruiken. Er zijn sindsdien talloze andere gemaakt. Op <http://xwinman.org> is een overzicht te vinden van een aantal bekende.

Een concept dat Windows-gebruikers wellicht vreemd is, maar onder de meeste window managers voorkomt is dat van virtuele desktops: meerdere desktops waartussen je kunt wisselen.

Een heel nieuwe window manager is Compiz Fusion (<http://www.compiz-fusion.org>). Deze zal je 3D-effecten geven op je desktop. Je kunt er o.a. alle windows op een gewenste transparantie mee zetten en tussen je virtuele desktops wisselen met de desktop cube. De naam komt van het feit dat er een tijd twee takken bestaan hebben: het oorspronkelijke Compiz en het daarvan geforkte Beryl. Beide zijn nu weer samengebracht in Compiz Fusion.

Er zijn twee grote projecten met als doel een desktopomgeving aanbieden: GNOME (<http://www.gnome.org>, GNU Network Object Model Environment) en KDE (<http://www.kde.org>, K Desktop Environment). Beide omvatten veel meer dan alleen een window manager. De software van beide projecten verzorgt min of meer hetzelfde: een file manager, menu's en alle gangbare desktopsoftware.

GNOME is in 1997 gestart door Miguel de Icaza. Samen met Nat Friedman begon hij in 1999 Helix Code, wat later werd hernoemd tot Ximian. In 2003 werd Ximian opgekocht door Novell. GNOME wordt gesponsord door o.a. Sun, IBM, HP en Novell. De file manager van GNOME heet Nautilus. Alle GNOME-software is gebaseerd op GTK+, de Gimp ToolKit, wat oorspronkelijk ontwikkeld was voor GIMP (een tekenprogramma). Een overzicht van extra software voor GNOME is te vinden op <http://gnomefiles.org>.

KDE is in 1996 gestart door Matthias Ettrich, evenals Miguel uit onvrede met de situatie van de desktop onder Unices op dat moment. KDE wordt gesponsord door o.a. Trolltech, Mandriva en SUSE (ook van Novell). De file manager van KDE is Konqueror. Naast file manager is Konqueror ook meteen een webbrowswer. De KDE-software is gebaseerd op Qt, een toolkit ontwikkeld door Trolltech. Een overzicht van extra software voor KDE is te vinden op <http://www.kde-apps.org>.

X.org, KDE, GNOME participeren in <http://www.freedesktop.org>, een discussieplatform waar standaarden en ideeën worden besproken om de interoperabiliteit en techniek van de Unix-desktop te verbeteren. Wat hier o.a. mee is vastgesteld is een standaard voor drag-and-drop, zodat dat tussen de verschillende desktopomgevingen werkt.

GNOME en KDE zijn geen lichtgewicht desktopomgevingen. Beide werken goed op een moderne pc, maar minder op oudere hardware. Erg lichtgewicht is alleen een window manager gebruiken. Er zijn enkele interessante projecten die ergens tussen beide in staan: Xfce (<http://www.xfce.org>), wat ook is gebaseerd op GTK+, en ROX desktop (<http://rox.sf.net>), wat is geïnspireerd door RISC OS.

Er wordt soms gedacht dat software die bij GNOME of KDE hoort alleen gebruikt kan worden als je de desbetreffende desktopomgeving gebruikt. Dat is niet het geval. De software heeft alleen de onderliggende library's nodig. Je kunt ze prima starten zonder de gehele desktopomgeving.

Veel window managers en toolkits hebben ondersteuning voor themes. Die zijn o.a. te vinden op:

- <http://themes.freshmeat.net>
- <http://art.gnome.org>
- <http://www.gnome-look.org>
- <http://www.kde-look.org>
- <http://www.xfce-look.org>

13.5 Toolkits

Toolkits of widget sets zijn library's om grafisch software te maken. De basiscomponenten daarvoor worden widgets genoemd. Voorbeelden van widgets zijn tekstbalken, knoppen en radio buttons. Programma's voor GNOME zijn geschreven m.b.v. GTK+, programma's voor KDE met Qt. Voor deze betrekkelijk nieuwe toolkits bestonden er al verschillende andere, waaronder Motif, Xaw/Athena en LessTif. Motif is hiervan de nieuwste en was bedoeld als standaard toolkit. Het wordt nog altijd het meest gebruikt voor grafische software onder andere Unices. Er is (veelal oudere) software voor Linux die een andere toolkit gebruikt dan GTK+ of Qt. Die software ziet er heel anders uit en werkt ook anders. De hele "look and feel" hangt af van de gebruikte toolkit. Zo gaat het omhoog scrollen in een programma als xterm (de standaard terminal van X) anders dan je waarschijnlijk gewend bent: je moet daarvoor met de rechter muisknop ergens op de grijze scrollbar klikken.

Oudere toolkits zoals Motif hebben geen ondersteuning voor themes. Wat wel kan is de kleuren van alle onderdelen aanpassen m.b.v. X resources (zie paragraaf 13.8). Daardoor is het wel mogelijk die software er in ieder geval qua kleurgebruik hetzelfde uit te laten zien. KDE kan dit voor je doen om alle software een meer consistente look te geven. Je kunt dit aan- en uitzetten in KDE's Control Center, onder Appearance & Themes, Colors, en dan het veld "Apply colors to non-KDE applications".

13.6 X starten

X kun je kaal starten met **xinit**. Je ziet dan een groot grijs scherm met verder helemaal niks. Er zijn twee manieren om X en de benodigde andere software te starten: **startx** of een display manager. Heb je problemen met X, dan kun je het afsluiten door tegelijk **ctrl**, **alt** en **backspace** in te drukken. Doe je dat vanuit een display manager, dan zal deze vanzelf herstart worden. (Dat is bijv. nuttig om een nieuwe configuratie te laten gebruiken.)

13.6.1 Startx

Als je X start met **startx** in de console, dan wordt een bepaalde standaard window manager of desktopomgeving gestart (afh. van je distributie). Je kunt dat echter overrulen en je kunt meer andere dingen laten starten met **startx**. Maak hiervoor in je home directory een bestand aan genaamd **.xinitrc**. Een voorbeeldinhoud:

```
# Instellingen
xset s off
xmodmap ~/.xmodmap

# Start een terminal
xterm -bg black -fg white -geometry 100x30+300+30 &

# Gebruik Window Maker
wmaker

# Gebruik Enlightenment
# enlightenment

# Gebruik KDE
# startkde

# Gebruik GNOME
# gnome-session
```

Regels die met **#** beginnen zijn commentaar en zullen worden overgeslagen. De instellingen bovenin zetten de screensaver uit en voert **xmodmap** uit voor wat aanpassingen aan de toetsenbordinstellingen. Dan start het script **xterm** (standaard terminal). Deze regel heeft een **&** achteraan om hem in de achtergrond te starten. Doe je dat niet, dan pikt **xterm** de shell in en komt **startx** nooit bij de onderste regels aan. (**Xset** en **xmodmap** doen dat niet, waardoor er geen **&** voor nodig is.) Als laatste start je een window manager. Je kunt er maar één tegelijk gebruiken. GNOME is geen window manager, maar een hele desktopomgeving. Alle benodigde software wordt gestart door **gnome-session** (o.a. een window manager). Voor KDE geldt hetzelfde.

Wat leuk is om eens te doen, is geen enkele window manager starten, maar alleen een terminal. Als je dan **startx** intikt, wordt X opgestart en zie je de terminal zonder randen. Je kunt hem dan niet van grootte veranderen of verplaatsen. Als je dan echter een window manager start vanuit die terminal (bijv. door **wmaker** in te tikken), dan verschijnen er ineens randen! Je kunt de window manager dan ook weer afsluiten met **ctrl-c** en een andere opstarten zonder dat X of de programma's die onder X gestart zijn afgesloten worden.

Wat je ook met **startx** kunt doen is X een tweede keer (of vaker) starten. Dat kan met **startx -- :1**. Hierbij staat de 1 voor displaynummer 1. Je standaard X-sessie heeft displaynummer 0. Je kunt tussen beide sessies wisselen met **ctrl-alt-F7** voor de eerste en **ctrl-alt-F8** voor de tweede.

13.6.2 Display manager

Xdm, de X daemon, laat een grafische inlogprompt zien. De variant van GNOME is gdm en die van KDE kdm. Deze software wordt ook wel display manager genoemd. Er is over het algemeen een aparte runlevel genaamd X gedefinieerd om de display manager te starten. De standaard runlevel waartoe geboot wordt, wordt bepaald in `/etc/inittab`. Zie paragraaf 8.1. De meeste distributies hebben ook een methode om te selecteren welke display manager je wilt gebruiken. Raadpleeg hiervoor de documentatie van de distributie.

13.7 Xset

Met `xset` kun je een aantal zaken van X instellen. Je kunt er o.a. mee aangeven of een screensaver gestart moet worden, of DPMS moet worden gebruikt en of er geluid uit de pc-speaker moet komen (zie ook paragraaf 20.28). Voor alle opties van `xset` kun je `xset` zonder argumenten intikken of de manual page bekijken. Voor een overzicht van alle huidige instellingen gebruik je `xset q`.

13.8 X resources

Met name oudere software voor X wordt grotendeels geconfigureerd met behulp van X resources. Deze instellingen staan voor een aantal bekende programma's in `/etc/X11/app-defaults`. In het bestand `XTerm` in deze directory staat bijvoorbeeld:

```
*saveLines: 1024
*SimpleMenu*menuLabel.font: -adobe-helvetica-bold-r-normal--*-120-*-*--*-iso8859-*
```

Dit zet het aantal regels dat je kunt terugscrollen in de terminal en het font voor het optiesmenu. Er kunnen ook nog algemene resources worden opgegeven in `/etc/X11/Xresources`. Verschil met de bestanden in `app-defaults` is dat hier de naam van het programma voor de * genoemd moet worden. In `app-defaults` moet de bestandsnaam de naam van het programma zijn.

Wil je als gebruiker eigen X resources opgeven, zet die dan in `~/Xdefaults`. Zet je daar een reeds bestaande variabele opnieuw, dan overschrijf je daarmee de oude waarde. Om `.Xdefaults` opnieuw te laten inlezen, kun je X opnieuw starten of `xrdb -load .Xdefaults` gebruiken. Om de oude instellingen niet verloren te laten gaan, kun je ook de optie `-merge` gebruiken i.p.v. `-load`. Dan worden de oude resources niet weggegooid.

Welke variabelen je kunt zetten wordt voor sommige applicaties genoemd in een `.ad`-bestand (van application defaults) in de directory met documentatie van het desbetreffende programma.

13.9 Fonts

X.org heeft ondersteuning voor een breed scala aan fontformaten: de aloude bitmapformaten `bdf` en `pcf`, Bitstream Speedo, CIDFont, OpenType, Adobe Type1 en Apple TrueType. Type1 en TrueType zijn schaalbare fonts. Je kunt ze zo groot maken als je wilt. Het zijn een soort vectortekeningen die de karakters beschrijven als vloeiende curves. Dit type font wordt ook wel "outline font" genoemd. Bij bitmapfonts moeten er plaatjes zijn voor elke puntgrootte die je ervan wilt kunnen maken. Voor de console of terminals onder X is een bitmapfont ideaal, maar voor grafische software gebruik je liever schaalbare.

Eind jaren 1980 hadden Apple, Microsoft en Adobe schaalbare fonts nodig. Adobe had Type1 ontwikkeld samen met hun PostScript-technologie voor printers. Apple en Microsoft waren echter bedachtzaam en gebruikten het liever niet vanwege het feit dat Type1 gesloten en propriëtair was. Apple heeft daarop TrueType ontwikkeld en beschikbaar gesteld aan Microsoft. Type1 is niet technisch gezien slechter dan TrueType, maar wordt veel minder gebruikt. Er zijn veel goede fonts beschikbaar in het TrueType-formaat,

waardoor ondersteuning ervan interessant is. Oorspronkelijk was het niet aanwezig, maar tegenwoordig is er van huis uit ondersteuning voor TrueType-fonts in X.

13.9.1 Core X11 font system

Deze subsectie beschrijft het oorspronkelijke fontstelsel van X. Dit is hoe van oudsher fonts werden bijgehouden en het is nog altijd nodig voor oudere software (niet voor GNOME en KDE bijv.).

Fonts worden gezocht in de directory's die zijn ingesteld in het font path. Die kun je instellen in het configuratiebestand van X of tijdelijk aanpassen met `xset`. Om fonts beschikbaar te stellen aan alle systemen in een netwerk zijn er ook nog font servers bedacht. Daarmee hoeven de fonts maar op één systeem aanwezig te zijn en kunnen alle clienten er gebruik van maken. De font server van X, Xfs, luistert naar poort 7100. Om Xfs te laten gebruiken als bron moet ook `unix/:7100` in het font path zijn opgenomen. De directory's waar Xfs fonts in zoekt, moet je instellen in het configuratiebestand van de Xfs zelf: `/etc/X11/fs/config`.

Om nieuwe fonts toe te voegen, moet een aantal stappen worden doorlopen. Zet de fonts in een bepaalde directory en voer daar `mkfontdir` uit. Dit maakt een bestand genaamd `fonts.dir` aan, een index die gebruikt wordt door X en Xfs, en `encodings.dir`, waar in staat welke fonts symbolen voor bepaalde coderingen bevatten. Als je alleen bitmapfonts (bdf en pcf) wilt installeren, is dit voldoende. Voor schaalbare fonts moet echter voordat je `mkfontdir` aanroept een bestand `fonts.scale` aanwezig zijn. De fontnamen uit dat bestand worden dan gekopieerd naar `fonts.dir`. Voor TrueType-fonts kun je daar het tooltje `ttmkfdir` voor gebruiken: `ttmkfdir -o fonts.scale`. Je kunt eventueel ook nog een bestand `fonts.alias` aanmaken met aliases voor fonts om ze onder een eenvoudigere naam beschikbaar te stellen.

Als je nieuwe fonts in een nieuwe directory hebt gezet, moet deze directory worden toegevoegd aan het font path. Tijdelijk toevoegen kan met `xset +fp /pad/naar/fonts`. Met `xset fp rehash` worden de beschikbare fonts opnieuw bekeken. Voor Xfs moet je Xfs herstarten met `/etc/init.d/xfs restart`.

Een beschrijving van de fonts onder dit systeem is van de volgende vorm:

```
-adobe-helvetica-bold-r-normal---*-120-*-*--*-iso8859-*
```

13.9.2 Xft

Wat niet in alle implementaties van X11, maar wel het nieuwere X.org zit, is Xft. Xft is van begin af aan ontworpen voor goede en efficiënte ondersteuning van schaalbare fonts. Het biedt de mogelijkheid tot anti-aliasing en geeft de applicatie volledige controle over hoe een bepaald symbool gerenderd moet worden. Om Xft te gebruiken, moet de applicatie dit wel ondersteunen. Alle software die gebruikmaakt van GTK+ of Qt (dus GNOME- en KDE-software) doet dit.

Xft is de API, het systeem voor applicaties, om ondersteuning voor schaalbare fonts zoals TrueType te bieden en maakt voor het feitelijke renderen van de fonts gebruik van de X-module FreeType. Xft staat los van het core X11 font system en heeft zijn eigen configuratiemechanisme. Het maakt daarvoor gebruik van de library `fontconfig`. Je kunt wijzigingen aan de standaard configuratie het beste aanbrengen via een `~/.fonts.conf` in je home directory. Je kunt daarmee bijv. aangeven geen anti-aliasing te gebruiken voor fonts kleiner dan een bepaalde puntgrootte.

Je hoeft veel minder moeite te doen om nieuwe fonts toe te voegen voor dit systeem. De directory `.fonts` in de home directory van de gebruiker wordt standaard doorzocht. Kopieer nieuwe fonts naar `~/.fonts` en update de cache van `fontconfig` met `fc-cache -v`. Het oude systeem om fonts toe te voegen (zoals hiervoor uitgelegd) is alleen nodig om de fonts te kunnen gebruiken in applicaties die geen Xft ondersteunen.

Er is nog een issue met TrueType-fonts en patenten. TrueType-fonts bevatten een zogenaamd glyph program. Het is een serie instructies die door een bytecode interpreter (kortweg BCI) uitgevoerd moet worden om

een plaatje te maken van het gewenste symbool. In het font kunnen "hints" staan die gebruikt worden om de letters bij kleine puntgroottes (juist de groottes die je voornamelijk gebruikt op het beeldscherm) mooi te houden. Apple heeft patent op delen die een BCI moet gebruiken om de hinting-informatie te gebruiken. Door deze instructies compleet te negeren, zien sommige symbolen in fonts er anders uit dan de bedoeling is. FreeType bevat code voor een volledige BCI, dus inclusief de delen voor hinting-informatie. Dit was toegevoegd voordat de situatie met de patenten duidelijk was. Apple heeft de makers van FreeType hiervoor nooit geprobeerd aan te klagen. Volgens de website: "Au contraire, because of FreeType, Apple has already seen new patent licensees in the embedded market." Sinds versie 2 van FreeType kun je kiezen of je ondersteuning voor de hinting-informatie wilt meecompileeren of niet. Het bevat ook een alternatief: de auto-hinting module. Dat is code die probeert het font er zo goed mogelijk uit te laten zien zonder de gepatenteerde onderdelen te gebruiken. Wat er in je distributie wordt gebruikt, zul je moeten nazoeken. Ik kan niet met zekerheid zeggen of fonts er beter uitzien als de hinting-informatie wordt gebruikt. Over het algemeen zijn er maar zeer weinig fonts die echt goede hinting-informatie bevatten. Het kan daardoor zijn dat de auto-hinting module juist betere resultaten geeft. Ben je tevreden met hoe de fonts eruitzien, laat het dan lekker zoals het is.

13.9.3 Goede fonts

De fonts die bij Windows meegeleverd zitten, zijn van hoogwaardige kwaliteit. Tahoma en Verdana zien er op het beeldscherm het beste uit. Er waren ooit "core fonts for the Web" (met name Arial, Courier, Times New Roman en Verdana) te downloaden van een website van Microsoft, maar dat is inmiddels verwijderd. Je kunt deze fonts eenvoudig laten downloaden en installeren met corefonts, in de meeste distributies te vinden als msttcorefonts. De fonts zullen worden gedownload van de website van het project, waar ze staan gemirrored. Je kunt de fonts ook vinden op je Windows-partitie als je Windows hebt geïnstalleerd.

De core fonts zijn in onaangepaste vorm verspreidbaar volgens de licentie ervan. Andere fonts, met name Tahoma, vallen onder de licentievoorzwaarden van Internet Explorer. Hier staat in opgenomen dat je een licentie van Windows moet hebben om het te mogen gebruiken. M.a.w. je mag Tahoma alleen gebruiken als je Windows hebt gekocht. Het corefonts-script kan Tahoma ook downloaden, maar zal het om deze reden niet standaard doen.

Door een overeenkomst tussen de GNOME Foundation en Bitstream is Bitstream Vera vrijelijk beschikbaar. Hierdoor zijn er vrije fonts van hoge kwaliteit beschikbaar voor Linux. Het is het standaard font van GNOME geworden.

14 Op afstand werken

14.1 X over het netwerk

Als je achter een andere computer zit waar X op is geïnstalleerd, dan kun je daar op vrij eenvoudige wijze grafische programma's op gebruiken die op jouw computer staan. Het programma draait dan op jouw computer, maar de X-server van de computer waar je op dat moment achter zit wordt gebruikt om het programma te tonen. Dit is wat er met netwerktransparantie wordt bedoeld.

Je moet er eerst voor zorgen dat jouw computer toegang heeft tot de X-server met **xhost +thuis**, waarbij je thuis vervangt door jouw IP-adres. (Op te vragen met `/sbin/ifconfig` op je eigen computer.) Heb je een hostnaam die op het netwerk bekend is, dan kun je dat ook gebruiken. Log vervolgens is op je computer, bij voorkeur via SSH (zie ook paragraaf 11.8). Zet daar de DISPLAY-variabele met **export DISPLAY=remote:0**, waarbij je remote vervangt door het IP-adres of de hostnaam van de computer waar je achter zit. Het cijfer achter de dubbele punt is het display-nummer en is standaard 0. Alternatief kun je bij ssh de optie `-X` gebruiken, zodat DISPLAY vanzelf goed wordt gezet. Als het goed is kun je nu alle grafische software starten die je wilt.

Als dit nog niet gedaan is, neem dan het volgende op in `/etc/ssh/ssh_config` van je eigen computer:

```
Host *
  ForwardX11 yes
```

Dit zorgt ervoor dat X via SSH getunneld wordt, wat wel zo veilig is. Indien de verbinding via een router of firewall loopt, dan dienen de poorten voor SSH en X open te worden gezet. SSH gebruikt standaard poort 22 en X poort 6000.

X zal op de meeste Unices wel aanwezig zijn, maar onder Windows is het niet gebruikelijk. Er bestaan implementaties van X die je onder Windows kunt gebruiken, zoals Exceed (<http://connectivity.hummingbird.com/products/nc/exceed/>, commercieel).

Een andere mogelijkheid is VNC (Virtual Network Computing), ook wel RFB (Remote Frame Buffer) genoemd, naar het protocol. VNC maakt het mogelijk een complete desktop over te nemen en het werkt door gecomprimeerde screenshots over het netwerk te versturen. RFB is een open protocol, zodat er voor vele platformen VNC-software is. Hierdoor is VNC bruikbaar als cross-platformoplossing voor remote beheer. RealVNC en TightVNC zijn er bijv. voor Windows en Linux. Bij zowel GNOME als KDE zit ook VNC-software.

14.2 SSH

Gebruik om op afstand in te loggen met een shell ssh. Zie ook paragraaf 11.8.

14.3 Screen

Wat je met VNC voor X kunt bereiken, kan met screen voor tekstgebaseerde software. Je kunt er programma's mee starten op de ene computer en daarna met een andere via telnet of SSH inloggen en de programma's "overnemen". Screen vormt een tussenlaag voor de input en output. Daardoor kun je het overnemen vanaf een andere computer. Alles wat je nodig hebt om het te kunnen gebruiken staat in de manual page: **man screen**.

15 Multimedia

15.1 Cd's en dvd's branden

Er zijn verschillende grafische frontends om cd's en dvd's te branden. Persoonlijk vind ik K3b (<http://www.k3b.org>) erg goed. Hieronder worden de onderliggende tools uitgelegd.

15.1.1 SCSI-emulatie

Er is enig verschil tussen kernel 2.4.x en 2.6.x wat betreft cd/dvd-branders. Bij kernel 2.4.x had je "SCSI-emulatie" in de kernel nodig. De meeste branders zijn van het type IDE en gebruiken ATAPI. Er zullen weinig mensen een SCSI-brander hebben. Het ATAPI-protocol is in essentie het versturen van SCSI-commando's over IDE, wat nodig is om verwisselbare media te kunnen gebruiken. Niet alle SCSI-commando's worden ondersteund door ATAPI, maar slechts een subset.

Om de "SCSI-emulatie" te gebruiken moet een kernel uit de 2.4.x-reeks gestart worden met de optie `hdc=ide-scsi`, met in plaats van `hdc` de device voor de brander. (Deze optie kan worden meegenomen in de configuratie van je bootloader. Zie paragraaf 9.4.) `ide-scsi` is het onderdeel dat SCSI-emulatie genoemd wordt. Deze naam is enigszins misleidend. Wat geëmuleerd wordt, is dat een ATAPI-brander omgetoverd wordt tot een "echt" SCSI-apparaat door de ontbrekende SCSI-commando's in ATAPI toe te voegen.

Met kernel 2.6.x is dit niet meer nodig. De methode van kernel 2.4.x is onnodig lastig en bemoeilijkt onderhoud. Verder zou het slecht geïmplementeerd zijn en al geruime tijd zonder maintainer zitten. Met kernel 2.6.x is er een nettere oplossing gekomen in de vorm van `ide-cd`. Er zijn geen opties voor de kernel meer nodig.

15.1.2 Tools voor cd's

`Cdrecord` in combinatie met `mkisofs` (samen `cdrtools` genoemd) is de basis voor het branden van cd's. `Cdrecord` kan een ISO-bestand op cd branden en `mkisofs` maakt ISO-bestanden aan.

Om data op een cd te branden, maak je eerst een ISO aan met `mkisofs -J -o isofile directory`. De optie `-J` zorgt ervoor dat Joliet-extensies worden gebruikt. Dat is nodig om lange bestandsnamen onder Windows te laten zien. Om het bestandssysteem meer op een Unix-bestandssysteem te laten lijken, zijn er de Rock Ridge-extensies. Gebruik daarvoor de optie `-r`. Met Rock Ridge zijn lange bestandsnamen mogelijk en ook permissies kunnen worden gebruikt. Je kunt op dezelfde cd beide (Joliet en Rock Ridge) tegelijk gebruiken. Voor permissies is Rock Ridge zeker nodig, voor alleen lange bestandsnamen kun je ook af met alleen Joliet.

Heb je een ISO-bestand gemaakt, dan kun je hem op een cd branden met `cdrecord`. Het gebruik: `cdrecord -v speed=16 dev=/dev/hdc -data isofile`, met in plaats van 16 de gewenste brandsnelheid en in plaats van `hdc` de device voor de cd-brander. (Gebruik voor kernel 2.4.x met "SCSI-emulatie" het devicenummer dat je kunt achterhalen met `cdrecord -scanbus`.)

Het is zeker op een modern systeem niet nodig om eerst een ISO te maken en dan pas te branden. Het kan veel sneller door de output van `mkisofs` door te pipen naar `cdrecord`. Dat doe je met: `mkisofs -J directory | cdrecord -v dev=/dev/hdx speed=16 -data -`. De `-` achter `-data` zorgt ervoor dat `cdrecord` data inleest van `stdin` (wat gelijk is aan de `stdout` van `mkisofs` door het pipen, zie ook paragraaf 6.5).

Alle nieuwe cd-branders hebben `burnproof`, waarmee cd's minder snel zullen mislukken bij hoge load tijdens het branden. Het is sterk aan te raden dit te gebruiken, zeker als je `on-the-fly` (zoals hiervoor uitgelegd) gaat branden. Om `burnproof` te gebruiken, moet je de optie `driveropts=burnfree` gebruiken bij `cdrecord`. Een herschrijfbaar cd kan snel worden gewist met de optie `-bank=fast` bij `cdrecord`. Vervang `fast` door `disc` om de gehele cd te wissen.

Je kunt audio-cd's branden met `cdrecord -v speed=16 dev=/dev/hdx -audio -pad track1.wav track2.wav`. De optie `-pad` zorgt ervoor dat de cd compatible is met de meeste afspeelapparatuur. Audio en data combineren is ook mogelijk. Dan gebruik je bij `cdrecord` simpelweg data- en audio-argumenten achter elkaar (`-data` en `-audio`).

Zie voor meer informatie de manual pages van `mkisofs` en `cdrecord`.

15.1.3 Tools voor dvd's

Voor het branden van dvd's is andere software nodig. Hiervoor is `dvd+rw-tools` de standaard geworden.

Op de command line moet je `growisofs` aanroepen om een dvd te branden. `Growisofs` is een frontend voor `mkisofs`, maar het brandt ook meteen de dvd. De meeste opties van `mkisofs` kunnen ook hiervoor gebruikt worden. Een voorbeeld om een dvd te branden met Joliet-extensies: `growisofs -Z /dev/hdc -J bestanden`. De optie `-Z` is om een eerste (of enige) sessie op een dvd te branden. Vervang `hdc` door de device van de dvd-brander. Er zal altijd on-the-fly gebrand worden.

Het is niet nodig de snelheid op te geven. `Growisofs` zal automatisch de hoogste snelheid gebruiken. Buffer underruns worden ook automatisch afgehandeld. Voor het overschrijven van een `dvd+rw` of `dvd-rw` is het niet nodig de dvd eerst te wissen. Je kunt er direct overheen branden.

Zie voor meer informatie de manual page van `growisofs`.

15.2 Muziek afspelen

De min of meer klassieke clone van Winamp is `XMMS`. Wellicht mede doordat de maintainers het vertikten over te gaan op een moderne GUI-library (GTK+ versie 2 i.p.v. 1) is het een langzame dood gestorven. Een interessante Winamp-clone met een moderne GUI is `Audacious` (<http://audacious-media-player.org>). Clone wil hier zeggen dat het er hetzelfde uitziet en skins voor Winamp kan gebruiken. Enkele uitgebreidere audiospelers zijn `Amarok` (<http://amarok.kde.org>) voor KDE en `Rhythmbox` (<http://www.rhythmbox.org>) voor GNOME.

De meeste audiospelers kunnen naast de bekende MP3's ook verschillende andere audioformaten afspelen, zoals AAC en Vorbis.

15.3 Video afspelen

Videobestanden eindigen vaak op `.avi` of `.mpg`. Dat zijn verpakkingsformaten (Engels: container formats) waar de eigenlijke video- en audiostromen in kunnen zitten. Veelvoorkomende verpakkingsformaten zijn AVI, MPEG en MOV. Je hebt ook enkele nieuwere verpakkingsformaten, zoals Ogg, Ogm en Matroska en op dvd's wordt VOB gebruikt.

De video- en audiodata kan op allerlei manieren geëncodeerd zijn. Veelvoorkomende encodings voor video zijn MPEG-1, MPEG-2, DivX en XviD, enkele encodings voor audio zijn MP3, AC3 en Vorbis. Een codec noem je wat de video- en audiodata encodeert of decodeert. Er zijn dus codecs nodig voor alle video- en audioformaten die je wilt kunnen afspelen. (Die voor audiodata kunnen ook gedeeld worden met audiospelers.)

Ondertiteling is er ook in verschillende formaten. SRT en SubRip kom je het meeste tegen. Het zijn simpelweg tekstbestanden met timinginformatie. `VobSub` (eindigt vaak ook op `.sub` in de bestandsnaam) is een heel ander formaat. Het wordt vaak vergezeld door `ifo`- en `idx`-bestanden, waar wat extra informatie in staat, zoals de positie van de ondertiteling. `VobSub`-bestanden zijn veel groter en bevatten geen tekst. Het is een kopie van de videostroom voor de ondertiteling op een dvd-films.

De eerste videospelers die echt veel moderne codecs bevatten waren MPlayer (<http://www.mplayerhq.hu>) en Xine (<http://xine.sf.net>). Een andere mooie speler is VLC (<http://www.videolan.org/vlc/>). Deze spelers zijn vrij monolithisch van aard: ze bevatten de meeste codecs intern of gebruiken rechtstreeks allerlei library's. De standaard speler van GNOME is Totem (<http://www.gnome.org/projects/totem/>). Totem maakt gebruik van GStreamer, een multimediaframework (zie ook paragraaf 8.9). Alle codecs die je nodig hebt, moet je installeren als plugins voor GStreamer. Om RealPlayer-filmpjes en -audio (.rm/.ram) af te spelen, kun je de Linux-versie van realplayer gebruiken. Deze is te vinden op <http://www.real.com/linux>.

Veel mediaspelers maken gebruik van de extensie van X genaamd Xv (X video). Dat geeft hardwareondersteuning voor videoweergave m.b.v. overlays.

15.4 Dvd-films afspelen

Bijna alle dvd's met filmmateriaal erop zijn geëncrypteerd met CSS (Content Scrambling System). In september 1999 had de Noorse tiener Jon Johansen samen met een onbekende Nederlander en Duitser (beide lid van een groep genaamd MORE, Masters of Reverse Engineering) ontdekt hoe het CSS-algoritme werkt en had DeCSS geschreven om CSS te decrypteren. DeCSS was een tooltje voor Windows. Hiervan afgeleid zijn de library's libcss en het nieuwere libdvdcss. Jon is in de tussentijd tweemaal aangeklaagd voor DeCSS, maar is beide keren vrijgesproken. Toch kom je in een grijs gebied van de wetgeving terecht als je gaat kijken naar of libdvdcss nou legaal is of niet. Het verschilt ook zeker per land. Het standpunt van Jon en vele anderen is dat het mogelijk moet zijn een door jou aangeschafte dvd te bekijken.

Distributiemakers nemen vaak het zekere voor het onzekere en bieden je libdvdcss in ieder geval niet bij de standaard installatie aan. Libdvdcss is te vinden op <http://developers.videolan.org/libdvdcss/>. Controleer of je distributie een pakket aanbiedt om het eenvoudig te installeren.

Dvd's bevatten naast CSS-encryptie ook nog een regiocode. De regio's zijn verzonden om ervoor te zorgen dat er bijv. geen dvd's uit Amerika geïmporteerd kunnen worden voordat de film hier uit is. Amerika heeft regio 1, wij zitten in regio 2. Als er regio 0 op een dvd staat, dan zal geen enkele dvd-speler moeilijk doen over het afspelen. Die zijn dus feitelijk regiovrij. Educatieve en uit sommige Aziatische landen afkomstige dvd's willen dat wel eens zijn. Eerst werden er regiovrije dvd-spelers gemaakt, RPC-1 genaamd, die altijd de sleutels geven om de CSS-data mee te decrypteren. Alleen de afspeelsoftware controleerde de regio van de dvd. Deze controle was eenvoudig te omzeilen door de afspeelsoftware te kraken of te doen geloven dat de regio goed is. Alle nieuwe dvd-spelers zijn RPC-2. Die spelers controleren hardwarematig of de regio van de dvd overeenkomt met die waar de dvd-speler op ingesteld staat. Komt het niet overeen, dan geeft hij de sleutels niet aan de software. Je kunt de regio van deze spelers (meestal) vijf keer veranderen. De laatste regio die je kiest, wordt permanent. Het is mogelijk een RPC-2-speler om te zetten in een RPC-1 door een gepatchte firmware te gebruiken. (Elke dvd-speler heeft firmware, een stukje software opgeslagen in herschrijfbaar geheugen in de speler. Het vormt de controlesoftware die bepaalt hoe de speler zich gedraagt.)

De hiervoor genoemde videospelers kunnen ook dvd's afspelen. Het is daarvoor niet nodig een gepatchte firmware te gebruiken. Door libdvdcss te gebruiken kunnen de decryptiesleutels bepaald worden zonder dat de dvd-speler ze hoeft door te geven.

15.5 Dvd's rippen

Er is een behoorlijk uitgebreid programma voor het verbouwen van videodata en het omzetten tussen verschillende formaten genaamd Transcode, te vinden op <http://www.transcoding.org>. Een van de dingen die hij kan, is een VOB-bestand (wat op een dvd staat) omzetten naar DivX, XviD of MPEG. Transcode heeft geen GUI, maar is een tekstgebaseerd programma. Het opgeven van een hele sloot argumenten (wat vaak nodig is) is niet zo heel prettig. `dvd::rip` (<http://www.exit1.org/dvdrip/>) is een makkelijke frontend voor Transcode en enkele andere tools, speciaal om dvd's te rippen.

Je kunt een dvd converteren naar verschillende formaten. Een AVI met daarin XviD voor video en MP3 voor audio wordt veel gebruikt. Kies liever XviD i.p.v. DivX. XviD is een open-sourceversie van DivX, maar

belangrijker dan dat: er is alleen een binary library te krijgen om DivX te encoderen en die kan nog wel eens problemen geven. XviD werkt altijd goed.

Met Transcode kun je ook de ondertitelingstracks rippen. Je hebt dan een VobSub-bestand van enkele MB's, afhankelijk van hoeveel ondertiteling er is. Het is mogelijk dit om te zetten naar SRT (ondertiteling in tekstformaat) door de plaatjes te OCR'en. OCR (optical character recognition) wordt vooral gebruikt om gescande tekst om te zetten in tekst, maar je kunt het ook gebruiken om veel kleinere ondertiteling te maken. Geschikte OCR-software is GOOCR (<http://jocr.sf.net>). Een script om het omzetten eenvoudig te maken is te vinden op <http://subtitledripper.sf.net> en zit ook bij Transcode meegeleverd.

15.6 Tv-kaart

Met een tv-kaart kun je televisiebeeld zien op je monitor. Het oudste en bekendste programma om je tv-kaart te gebruiken is xawtv (<http://bytesex.org/xawtv/>). Alternatieven zijn tvtime (<http://tvtime.sf.net>) en Zapping (<http://zapping.sf.net>). Sommige algemene videospelers kunnen ook de tv-kaart aansturen.

Mencoder (zit bij MPlayer) is in staat van een V4L-device (zie ook paragraaf 3.5) te lezen, dit te encoderen en in een bestand op te slaan. Daardoor kun je dus tv-beeld captureren. Analoge tv moet gedigitaliseerd worden, wat veel rekenkracht kost. Op wat voor resolutie en met wat voor compressie dat nog lukt, is afhankelijk van hoe snel je processor is en kun je het beste bepalen door wat uit te proberen. De documentatie en manual page van MPlayer/Mencoder geven voldoende informatie over hoe je dit kunt doen.

Voor het opnemen van digitale tv kun je VDR gebruiken (<http://www.cadsoft.de/people/kls/vdr/>). Veel informatie over digitale tv onder Linux is te vinden op <http://www.linuxtv.org>.

15.7 Video editen

Voor het converteren van videoformaten en toepassen van verschillende bewerkingen zijn Mencoder (zit bij MPlayer) en Transcode (<http://www.transcoding.org>) geschikt. Transcode biedt uitgebreidere mogelijkheden, maar is wel minder makkelijk in gebruik.

De meest bruikbare interactieve editor is Avidemux, te vinden op <http://avidemux.sf.net>. Het is te vergelijken met VirtualDub voor Windows. Je kunt er bewerkingen zoals knippen en plakken mee uitvoeren op AVI's en MPG. Avidemux heeft ook verschillende leuke audio- en videofilters. Voor DV (digital video, wat wordt gebruikt door digitale camera's) is Kino (<http://kino.schirmacher.de>) geschikt.

15.8 Patenten en andere wettelijke restricties

Het multimedialandschap wordt wel eens een patentenmijnenveld genoemd. Vanwege patenten en andere wettelijke restricties is er bij veel distributies na de installatie geen ondersteuning voor allerlei mediaformaten. Dit moet door de gebruiker na de installatie geïnstalleerd worden. Er zijn handleidingen voor verschillende distributies voor te vinden op internet.

Mediaformaten met problemen zijn MP3, QuickTime, DivX, RealMedia en MPEG. MP3 valt onder een patent van het Fraunhofer Institute dat niet compatibel is met GPL. Voor thuisgebruikers hoeven er geen royalty's voor betaald te worden. Ook QuickTime, DivX en RealMedia vallen onder patenten. MPEG is op zich een open standaard, maar er zijn blokkerende patenten van MPEG LA. En zoals hiervoor al duidelijk werd, is de wettelijke status van CSS-library's om geëncrypteerde dvd's te kunnen afspeelen niet geheel duidelijk.

Om overeenkomstige redenen worden Flash, Java en Acrobat reader ook vaak niet standaard geïnstalleerd. Bij Flash zijn er beperkingen op de verspreiding in de licentie opgenomen. Java mag gedownload en verder

verspreid worden van Sun, maar alleen voor persoonlijk en niet-commercieel gebruik. Acrobat reader is propriëtaire software. Verder zijn er geen problemen mee.

Er zijn enkele volledig open mediaformaten. Ze zijn gemaakt om een alternatief te hebben zonder patentproblemen. De Xiph.org Foundation (<http://www.xiph.org>) heeft een verpakkingsformaat gemaakt, genaamd Ogg (bitstream). Bijbehorende encodings zijn Ogg Theora voor video en Ogg Vorbis voor audio. Hiervan is Ogg Vorbis het populairst (als alternatief voor MP3). Een reden hiervoor is dat Theora nog niet volledig af is. Een uitbreiding op Ogg, genaamd Ogm, heeft ook ondersteuning voor enkele propriëtaire encodings. Voor films is er ook XviD, een open-sourceversie van DivX. Ten slotte is er nog een redelijk populair verpakkingsformaat genaamd Matroska (<http://www.matroska.org>). Deze bestanden eindigen op .mkv. Ogg, Ogm en Matroska zijn allemaal modernere verpakkingsformaten dan AVI. Ze hebben ondersteuning voor streamen (over internet bijv.), meerdere audio- en videostreamen en ondertiteling in het bestand zelf (ook meerdere streamen).

16 Internationalisering

16.1 Karaktersets en coderingen

Voor ons alfabet zijn acht bits meer dan voldoende om alle mogelijke letters te coderen. Daarmee heb je 256 mogelijkheden. Plaats genoeg voor alle hoofd- en kleine letters. De meest weidverspreide codering is US-ASCII. ASCII is een standaard waarin maar zeven bits gebruikt worden per karakter. Er zijn wel karaktersets waarbij alle acht bits gebruikt worden die zijn gebaseerd op ASCII.

Er zijn veel talen die niet genoeg hebben aan deze karakters (kleine- en hoofdletters en een paar speciale karakters). Duits en Frans hebben bijvoorbeeld letters met accenten nodig en Nederlands ook in mindere mate. De meest gebruikte standaard is ISO-8859-1, ook wel (ISO) Latin-1 genoemd. Is het achtste bit 0, dan komt het overeen met ASCII. Is deze bit gezet, dan krijg je nieuw toegevoegde symbolen. Deze standaard is bruikbaar voor veel Europese talen. Daarnaast zijn er nog wat meer 8859-varianten waaronder 8859-2 (Latin-2) voor Centraal- en Oost-Europa, 8859-5 voor Cyrillische talen, 8859-7 voor Grieks en 8859-15 (of Latin-9), wat bijna gelijk is aan Latin-1, met als voornaamste verandering dat het universeel valutasymbool vervangen is door een eurosymbool.

Unicode (<http://www.unicode.org>) is een heel erg leuk idee: verzin een nummer voor elk karakter dat in elke taal op deze aardbol voorkomt. Nou ja, de meest talen dan. Er zijn vele praktische problemen om dit voor elkaar te krijgen. Er wordt nog altijd gesleuteld aan Unicode. Toch is het al meer dan goed bruikbaar. Een veel gebruikte codering voor Unicode (een manier om de nummers in een bitreeks weer te geven) is UTF-8. Er zijn er nog enkele meer.

Alhoewel Unicode een erg leuk idee is, is het gebruik nog niet erg wijdverspreid. Veelal worden nog lokale oplossingen gebruikt, zodat er in de computerwereld vele coderingen door elkaar gebruikt worden. Unicode zal waarschijnlijk wel de toekomst worden. Het is dé ultieme manier om tekst in een digitale vorm te zetten. Het biedt ook de mogelijkheid om binnen een tekst karakters uit allerlei talen tegelijk op te nemen. Het hebben van vele verschillende standaarden is een crime. Maar zolang men nog allemaal eigen codering gebruikt, is er voor goede internationalisering nog altijd een uitgebreide ondersteuning voor allerlei coderingen nodig. Dat heeft Linux dan ook.

16.2 Locales

Linux gebruikt locales. De locale geeft twee dingen aan: de taal en de karakterset. Door middel van een aantal shell-variabelen kun je de gewenste taal en karakterset kiezen.

Er zijn verschillende locale-onderdelen, waaronder LC_CTYPE, LC_TIME, LC_MONETARY, LC_MESSAGES en LC_ALL. LC_CTYPE geeft de te gebruiken karakterset aan, LC_TIME en LC_MONETARY geven aan welke locale gebruikt moet worden voor het weergeven van de tijd en geldbedragen. LC_MESSAGES geeft aan in welke locale berichten van programma's weergegeven moeten worden. Als je de variabele LC_MESSAGES op nl_NL zet, worden alle berichten, indien beschikbaar, in het Nederlands weergegeven. De output van ls is dan bijv. ineens Nederlands geworden. Door de locale-onderdelen apart te zetten, kun je kiezen in welk formaat je munteenheden, adressen, de tijd etc. wilt hebben. Met LC_ALL kun je alle onderdelen tegelijk zetten. Daarmee overrule je alle andere locale-onderdelen.

Tik **locale** om de huidige waarde van alle onderdelen te bekijken. Je kunt een bepaald onderdeel (in Bash) op een gewenste locale zetten met een commando als **export LC_CTYPE=nl_NL@euro**.

Verder heb je nog de variabelen LANG en LANGUAGE. Oorspronkelijk was er alleen LANG. LANGUAGE is toegevoegd bij de GNU-software. Wat de meeste software voor Linux gebruikt voor berichten in verschillende talen is de functie gettext (meer hierover verderop). Voor die software geldt de volgende volgorde van prioriteit voor de locale-variabelen: LANGUAGE, LC_ALL, LC_xxx, LANG. Enkele programma's gebruiken echter geen gettext. Een voorbeeld is OpenOffice.org. Die kijkt alleen naar LANG. Om de locale globaal voor alle software in te stellen is het daarom het beste zowel LANG als LANGUAGE op de gewenste locale

te zetten.

Met LANGUAGE kun je meer dan één locale aangeven. Het kan zijn dat er niet voor elk programma tekst beschikbaar is in de door jou gekozen locale. Maar misschien vind je naast Nederlands Frans ook acceptabel. Je kunt dan een voorkeursvolgorde opgeven: **export LANGUAGE=nl_NL:fr_FR**. Scheid de mogelijkheden steeds met een dubbele punt. Dit kun je niet doen met LANG.

Een programma draait altijd in een bepaalde omgeving van variabelen. Als je zojuist een locale-onderdeel op een andere waarde hebt gezet in een terminal in X, dan heeft dat alleen effect op programma's die in diezelfde terminal worden gestart! (Die programma's worden dan kindprocessen van de shell.) Neem export-regels op in `~/bashrc` om het elke keer te laten zetten bij het starten van Bash. Als je wilt dat het voor alle programma's die je via je window manager of desktopomgeving (zoals KDE en GNOME) start geldt, dan zal X zelf met de juiste omgevingsvariabelen zijn gestart. Log je grafisch in met een display manager, dan kun je de taal daar vaak mee kiezen. KDE heeft voor KDE-software zelf de mogelijkheid in het Control Center de gewenste taal te kiezen.

Wat zijn nou interessante locales? Voor de meeste Nederlanders deze:

- en_US (gebruikt ISO-8859-1)
- en_US.utf8 (gebruikt UTF-8)
- nl_NL (gebruikt ISO-8859-1)
- nl_NL@euro (gebruikt ISO-8859-15)

Merk op dat de definities niet onder elke distributies identiek hoeven te zijn aan de namen hierboven. Gebruik **locale -a** om te zien welke er gedefinieerd zijn.

Alle locales hebben het formaat taal_REGIO en optioneel .encoding of @modifier. Je kunt locales toevoegen met **localdef**. Om bijv. en_US.utf8 toe te voegen gebruik je **localdef -f UTF-8 -i en_US en_US.utf8**. Je distributie kan ook een mechanisme hebben om de gewenste locales uit te kiezen.

Het systeem dat onder Linux wordt gebruikt om alle tekst van een programma in verschillende talen te kunnen aanbieden (afhankelijk van de gekozen locale), is `gettext`. De tekst wordt per taal in een apart bestand gezet: een po-bestand. Over het algemeen is de tekst sowieso in het Engels aanwezig. In de po-bestanden staan eventueel vertalingen (een taal per bestand). Iemand anders naast de programmeur zelf, die vaak niet elke taal op deze aardbol beheerst, kan deze bestanden aanmaken of aanvullen. Is een bepaalde tekst nog niet helemaal vertaald, maar het meeste wel, dan worden de niet vertaalde regels in het Engels weergegeven. Dat zorgt er dus niet voor dat er tekst ontbreekt. Als de po-bestanden af zijn, maakt men er eerst nog een snel te doorzoeken mo-bestand van (een binair formaat, niet meer zomaar te bekijken in een teksteditor).

Alle modernere programma's die GTK+ (GNOME-software) of Qt (KDE-software) gebruiken ondersteunen dankzij deze library's allemaal `gettext/locales`. Indien de benodigde fonts zijn geïnstalleerd, kan deze software tekst in allerlei talen weergeven. GTK+ maakt hiervoor gebruik van Pango. Veel software van GNOME en KDE is dankzij het werk van vele vrijwilligers wereldwijd beschikbaar is zeer veel verschillende talen.

Andere software waar goede ondersteuning voor allerlei talen en encodings nodig is, zijn browsers zoals Firefox. Anders dan bij de meeste software zorgt het zetten van een locale bij een browser alleen voor de taal en encoding van de menu's. De encoding van een bepaalde website hoort aangegeven te zijn in de HTML-code en wordt ook doorgegeven door de webserver die het document aanbiedt. Als dit niet goed geregeld is, kan het zijn dat de browser de tekst niet goed weergeeft. Je moet de encoding dan handmatig kiezen in een menu van de browser.

16.3 Iconv

Er zijn conversietabellen voor alle bekende encodings. Alle software kan deze mogelijkheid gebruiken om tekst voor het programma te converteren naar de door de gebruiker gewenste encoding (aangegeven door de locale-variabelen). Er is een zeer handig toeltje om conversies handmatig uit te voeren: **iconv**. Tik **iconv --list** voor een lijst met bekende encodings. Gebruik bijvoorbeeld **iconv -f sjis -t euc-jp < foo > foo2** voor het omzetten van het bestand foo in shift-JIS naar EUC-JP. De output komt in het bestand foo2 terecht.

16.4 Xmodmap

Met xmodmap kunnen toetsencombinaties worden gedefinieerd om speciale karakters te vormen. Het werkt voor alle software onder X.

In onderstaand voorbeeld worden twee waardes toegekend aan elke toets, namelijk een symbool voor het indrukken van de toets zonder shift en een waarde voor het indrukken van de toets met shift. Elke toets heeft een nummer, een keycode. Zo is de keycode van de /-toets (op het numerieke pad) 070. Het symbool / heet ook wel "KP_Divide". Dat is het symbool dat je krijg je als je de toets zonder shift indrukt (wordt het eerste genoemd). Als je de toets met shift indrukt krijg je het symbool dat "dead_acute" heet. Er wordt dan niets getoond, maar als je erna een klinker intikt, komt daar er een accent aigu (streepje naar rechts) op te staan. (Acute accent is de Engelse benaming voor het streepje naar rechts.) In het volgende voorbeeld worden accenten toegekend aan de toetsencombinaties shift en /, shift en *, shift en - en shift en +.

Maak, als het nog niet bestaat, een bestand genaamd .xmodmap aan in je home directory en zet daar het volgende in.

```
keycode 0x70 = KP_Divide    dead_acute
keycode 0x3F = KP_Multiply  dead_diaeresis
keycode 0x52 = KP_Subtract  dead_grave
keycode 0x56 = KP_Add       dead_circumflex
```

Voer daarna **xmodmap ~/.xmodmap** uit. Tijdelijke veranderingen kun je ook sneller laten uitvoeren met **xmodmap -e "keycode xx = xxx"**.

Het is mogelijk om meer dan twee waardes aan een toets toe te kennen. Niet alleen in combinatie met shift, maar ook in combinatie met bijv. een Windows-toets (als je een toetsenbord hebt waar die op zit). In dat geval moet je zorgen dat zo'n toets een modifier is. De toetscode van de linker Windows-toets is 073. Om die als modifier te kunnen gebruiken, heb je de volgende twee regels nodig.

```
keycode 0x73 = Mode_switch
add      Mod3 = Mode_switch
```

Nu kun je bijv. de e zo gebruiken dat het normaal een e geeft, met shift een E, met de linker Windows-toets een é en met shift en de linker Windows-toets een É. Dat kan met de volgende regel voor xmodmap.

```
keycode 0x1A = e  E  eacute  Eacute
```

Nog een andere mogelijkheid is gebruikmaken van de multikey, ook wel compose key genoemd. Als je die multikey indrukt, dan een apostrof en daarna een e dan krijg je é. Multikey gevolgd door een dakje (shift en 6) en dan e geeft ê. De volgende regel zorgt ervoor dat de rechter Windows-toets gebruikt wordt als multikey.

```
keycode 0x74 = Multi_key
```

Je kunt de huidige instellingen bekijken met **xmodmap -pk**.

Deze tip is afkomstig van Bas van der Meer.

Een probleem is om alle keycodes te weten. Er is frontend voor xmodmap om grafisch alle toetsen in te kunnen stellen, te weten xkeycaps (<http://www.jwz.org/xkeycaps/>). Je kunt ook **xev** starten. Als je dan een toets intikt terwijl het venster van xev actief is, zie je ook de keycode. Alle keysyms (de namen voor de symbolen), zoals `eaacute` kunnen worden gevonden in `/usr/include/X11/keysymdef.h`. Op <http://www.chemie.fu-berlin.de/chemnet/use/xmodmap.html> is ook een overzicht te vinden.

Je kunt xmodmap gebruiken om zelf wijzigingen aan te brengen t.o.v. van de standaard toetsenbordindeling. Hoe je de toetsenbordindeling instelt voor X kun je vinden in paragraaf 13.3. Als je daarmee kiest voor `us_intl` heb je al een handige indeling om snel accenten in te kunnen tikken.

16.5 Input methods

Zeker voor de CJK-talen (Chinees, Japans, Koreaans) is de toetsen op bepaalde karakters mappen bij lange na niet toereikend. Je moet duizenden karakters in kunnen voeren met slechts enkele toetsen op het toetsenbord. Om dit voor elkaar te krijgen, heb je extra software nodig: IM's. IM staat voor Input Method. XIM is een protocol van X bedoeld voor dergelijke IM's.

Voorbeelden van IM's zijn `kinput2` voor Japans, `xcin` voor Chinees en `ami` voor Koreaans. Als de IM is gestart moet je de shell-variabele `XMODIFIERS` zetten om de IM die je wilt gebruiken aan te geven. In geval van `kinput2`: **export XMODIFIERS='@im=kinput2'**. Vervang `kinput2` door de gewenste IM. Vaak is het nodig programma's waar je de IM in wilt gebruiken te starten in een geschikte locale, voor `kinput2` bijv. een Japanse zoals `ja_JP.eucJP` (Japans in eucJP-encoding).

De IM is vaak te starten en stoppen, zodat je het toetsenbord ook nog "gewoon" kunt gebruiken. Meestal is de toetsencombinatie om de IM te activeren `shift-spatiebalk` of `ctrl-spatiebalk`. Toetsen die je nodig hebt om daarna iets met de IM te doen, verschillen per IM. Of je de toetsen kunt wijzigen en hoe verschilt ook. Toetsen die vaak iets doen zijn de `spatiebalk`, `shift-spatie`, `ctrl-spatie` en `ctrl-shift`. Probeer ook `insert`, `home` en `end` eens.

Naast de klassieke IM's die gebruikmaken van XIM, zijn er ook modernere invoermethoden voor GNOME en KDE. Zowel GNOME als KDE hebben een IM-module. Er zijn IM's die enkel hiermee werken, zoals `im-ja` (<http://im-ja.sf.net>) voor Japans in GNOME-applicaties. Een IM voor de IM-module van GNOME of KDE kan resp. gekozen worden met de shell-variabelen `GTK_IM_MODULE` en `QT_IM_MODULE`. Deze variabelen kun je op `xim` zetten om een IM die gebruikmaakt van XIM te gebruiken.

De meeste IM's zijn oplossingen voor een enkele taal. Er is een tweetal projecten om een algemene IM te maken die overal voor werkt: `SCIM` (Smart Common Input Method) en `UIM` (Universal Input Method). Beide werken met XIM of de immodule van GNOME/KDE. `SCIM` lijkt het verst gevorderd en is bruikbaar voor 30 talen.

16.6 Andere invoermethoden

Bij GNOME zit `gucharmap`. In dit programma kun je elk Unicode-symbool vinden en vervolgens kopiëren. Er zijn verschillende methodes beschikbaar om naar een bepaald symbool te zoeken.

Als je maar heel af en toe letters met accenten en andere speciale karakters uit de Latin-1-set nodig hebt en je gebruikt Vim om bestanden te editen, dan kun je een handige functie van Vim gebruiken. Druk als je in

insert mode zit op ctrl-k en daarna nog enkele toetsen. Een overzicht van mogelijke accenten (en wat andere speciale karakters) krijg je met `:digraph` in command mode. Zie ook paragraaf 6.9.1.

16.7 Het euroteken

Om het euroteken te kunnen gebruiken moet je een locale gebruiken met een karakterset waar het symbool überhaupt in kan worden weergegeven. Dat kan ISO-8859-15 of UTF-8 zijn. Localenamen die eindigen op `@euro` gebruiken ISO-8859-15. Verder moet je uiteraard fonts hebben waar het symbool in voorkomt. Je kunt controleren of je die (voor X) hebt door te kijken of er fonts geworden als je `xlsfonts | grep iso8859-15` uitvoert. Sommige software (zoals programma's voor GNOME of KDE) kiezen uit zichzelf al een goed font, maar voor andere (zoals xterm) moet je zelf een goed font aanwijzen. In het geval van wat oudere software voor X moet dat met behulp van X resources (zie paragraaf 13.8). In de Debian Euro HOWTO (<http://www.debian.org/doc/manuals/debian-euro-support/>) staan wat aanwijzingen hoe je dat kunt doen voor enkele bekende programma's. Die aanwijzingen zijn niet Debian-specifiek.

Het laatste probleem is het mogelijk maken het eurosymbool zelf in te voeren. Je kunt de mogelijkheid zelf toevoegen met Xmodmap met de volgende twee regels:

```
keycode 113 = Mode_switch
keycode 26 = e E EuroSign
```

Je kunt nu het euroteken invoeren met de rechter alt en e. In een internationale toetsenbordinstelling is dit echter standaard al gedefinieerd. (Zie ook paragraaf 13.3.)

17 Windows-gerelateerd

17.1 Windows-applicaties draaien onder Linux

Linux is een besturingssysteem, net als Windows. Beide besturingssystemen werken heel anders, hebben een andere API (Application Programming Interface) en ook de binary's, de uitvoerbare bestanden, zitten anders in elkaar. Het is absoluut onmogelijk om zomaar een Windows-programma onder Linux te starten als ware het een normaal Linux-programma en omgekeerd geldt hetzelfde. Er is echter wel software die het mogelijk maakt dit te doen.

Wine De eerste is Wine (<http://www.winehq.com>). Wine is zeker geen emulator! De naam staat ook voor Wine Is Not an Emulator (weer een recursief acroniem). Een emulator werkt fundamenteel anders. Een emulator maakt het mogelijk om software voor een bepaald systeem (bijv. een Commodore 64) op een ander systeem te draaien (bijv. een IBM-compatible pc). Een emulator moet in de regel instructies voor de ene processor vertalen in instructies voor de andere processor. Wine is een open-source-implementatie van de gehele Windows-API. Dat betekent dat elke functieaanroep die in een Windows-programma gemaakt wordt ook in Wine aanwezig is. Wine streeft ernaar om volledig zonder een Windows-installatie Windows-programma's te kunnen uitvoeren. Optioneel kunnen DLL-bestanden worden gebruikt als daar functies in voorkomen die niet in Wine zelf zijn ingebouwd. Anders dan bij een emulator werken programma's niet noodzakelijk trager als je ze start met Wine. Het is mogelijk dat programma's even snel of zelfs sneller werken. Nog niet alles werkt perfect, maar hoe langer hoe meer applicaties werken goed. Op de website is een overzicht te vinden van geteste applicaties.

CodeWeavers (<http://www.codeweavers.com>) verkopen een versie van Wine genaamd CrossOver Office. Deze versie heeft goede ondersteuning voor Microsoft Office (Word, Excel e.d.), Photoshop en nog enkele andere kantoorapplicaties. In het verleden hadden ze ook CrossOver Plugin, wat ondersteuning had voor verschillende plugins voor webbrowsers, zoals QuickTime en Shockwave. Dit zit nu ook bij CrossOver Office. CodeWeavers werkt goed samen met Wine. Hun verbeteringen komen uiteindelijk terug in de hoofdstroom van Wine en andersom maken ze gebruik van de verbeteringen die worden aangebracht in de hoofdstroom. Op die manier worden beide wel min of meer gelijk gehouden en ontstaat er geen echte fork, maar ze kunnen er toch wat mee verdienen.

Een andere belangrijke versie van Wine is die van TransGaming (<http://www.transgaming.com>): Cedega, voorheen WineX. Cedega is gericht op het ondersteunen van spellen. Ondersteuning voor DirectX is beter in hun versie. Het heeft ook ondersteuning voor kopieerbeveiligingen. Net als CrossOver Office is dit een commercieel product. Het is ontstaan als fork toen de licentie van Wine veranderde van X11 naar LGPL. Je kunt een abonnement nemen voor een klein bedrag per maand. TransGaming biedt enkele verbeteringen in hun versie aan aan de hoofdstroom van Wine, maar helaas lang niet alles.

Virtuele machine VMware (<http://www.vmware.com>) simuleert een computer. Daarbij wordt een deel van de CPU-tijd, geheugen en schijfruimte voor de virtuele machine gebruikt. Anders dan bij Wine start je hier niet een enkel Windows-programma, maar je start gewoon Windows compleet op! (Naast Windows kun je ook andere OS'en installeren onder VMware.) Het voordeel van deze methode is dat echt zo goed als alles werkt. Nadelen zijn dat VMware niet gratis te verkrijgen is, het veel resources gebruikt en je een versie van Windows nodig hebt (Wine kan zonder).

Andere virtuele machines zijn:

- VirtualBox (<http://www.virtualbox.org>)
- QEMU (<http://fabrice.bellard.free.fr/qemu/>)
- Plex86 (<http://plex86.sf.net>)
- Bochs (<http://bochs.sf.net>)
- Win4Lin (<http://win4lin.net>)

17.2 DOS-applicaties draaien onder Linux

Hier is een tweetal goede emulators voor:

- DOSBox (<http://dosbox.sf.net>)
- DOSEmu (<http://dosemu.sf.net>)

Het voordeel van DOSBox is zijn eenvoud van configuratie. Je gebruikt geen echte versie van DOS binnen deze emulator. Hij is goed te gebruiken voor oude spellen (ook onder Windows). Binnen DOSEmu draai je wel een echte versie van DOS, zoals MS-DOS, FreeDOS of DR-DOS. Je moet alles op de originele DOS-manier configureren (via config.sys en met drivers voor DOS dus).

17.3 Windows-partities benaderen vanuit Linux

Hiertoe moet je de Windows-partitie mounten. Hoe je dat moet doen, wordt uitgelegd in paragraaf 6.12.

17.4 Linux-partities benaderen vanuit Windows

Standaard is dat niet mogelijk, maar het is wel mogelijk met:

- Explore2fs (<http://www.chrysocome.net/explore2fs>) - voor alle versies van Windows, alleen lezen
- Ext2fsd (<http://ext2fsd.sf.net>) - voor Windows NT, 2000 en XP, alleen lezen
- Ext2 Installable File System (<http://www.fs-driver.org>) - voor Windows NT, 2000 en XP, ook schrijven

Deze software kan overweg met het standaard bestandssysteem voor Linux, ext2. Ext3 zal ook werken, omdat het bestandssysteem zelf identiek is, maar Reiser zal niet lukken. Voor Reiser is er ook een tooltje: rfstool (<http://www.p-nand-q.com/e/reiserfs.html>). Het is een command-lineprogramma met enkele GUI's (links op de website).

17.5 Defragmenteren

Mensen die Windows 95 en 98 met FAT-bestandssystemen gewend zijn, kunnen denken dat het nodig is om de zoveel tijd je bestandssysteem te defragmenteren. Bij NTFS en alle bestandssystemen voor Linux (ext2, ext3 of Reiser) is het niet nodig te defragmenteren. Er is zelfs geen software voor.

18 Bash scripting

Er bestaan verschillende shells voor Linux. De meest gebruikt is Bash, de Bourn Again SHell. Elke shell kent een eigen scripttaal. Een script is een gewoon tekstbestand dat begin met de regel `#!/bin/bash` in het geval van Bash. Deze regel is in de voorbeelden die volgen steeds weggelaten. In plaats van die regel zou je ook `bash file` kunnen gebruiken om het script uit te voeren. Maak je het bestand uitvoerbaar met `chmod +x file`, dan kun je het starten alsof het een binary executable is (met `./file`).

Als je achter de prompt een regel als `for i in $(ls); do` tikt, dan krijg je een `>`-teken te zien. Het commando is nl. nog niet af en Bash verwacht meer. Je kunt dan de rest van het commando intikken of `ctrl-c` gebruiken om af te breken.

Er volgt nu een aantal voorbeelden, van simpel naar iets moeilijker, met steeds wat uitleg.

```
STRING="Hello World!"
echo $STRING
```

Maak de variabele `STRING` aan en zet daar "Hello World!" in. Laat daarna de variabele zien. Gebruik een `$`-teken om aan te geven dat het een variabele is. Anders zie je gewoon de tekst `STRING` verschijnen.

```
OF=/mnt/backup/backup-$(date +%Y-%m-%d).tar.gz
tar zcvf $OF /home/username
```

In de variabele `OF` (Output File) wordt de bestandsnaam ingevuld. Die zal altijd beginnen met `backup-` en eindigen op `.tar.gz`. In het midden wordt de datum ingevuld in de volgorde jaar, maand, dag. Je kunt de output van elk willekeurig commando gebruiken door er een `$`-teken voor te zetten en het tussen ronde haakjes te plaatsen. De tweede regel pakt een hele home directory in en noemt het zoals de variabele `OF` aangeeft.

```
cat `which startx`
```

Van een commando dat tussen backquotes staat (de komma links naast de 1 op het toetsenbord) wordt de output letterlijk ingevuld. Als `which startx` bijv. `/usr/bin/X11/startx` oplevert, dan wordt dat achter `cat` gezet.

```
STR1="string"
STR2="sting"

if [ $STR1 = STR2 ]; then
    echo waar
else
    echo niet waar
fi
```

Met `if` kun je een conditie opgeven. Als het waar is wat er tussen de vierkante haken staat, dan wordt uitgevoerd wat er achter `then` staat. Zo niet, dan wordt uitgevoerd wat er achter `else` staat. Een `else`-regel is niet noodzakelijk. Let op de `;` achter de `if`-regel en de spatie tussen de vierkante haakjes en wat er tussen staat. Geef met `fi` aan waar het `if`-blok stopt.

```
for i in $(ls); do
    echo $i
done
```

Dit is een van de drie soorten loops. De eerste regel zorgt ervoor dat de variabele `i` de waarden aanneemt die `ls` teruggeeft. De tweede regel laat de variabele `i` zien. De laatste regel geeft het einde van de loop aan. De loop wordt herhaald totdat `i` geen nieuwe waarden meer aanneemt.

```
COUNTER=0
while [ $COUNTER -lt 10 ]; do
    echo $COUNTER
    let COUNTER+=1
done
```

Dit is de tweede soort loop. De loop wordt doorlopen zolang (`while`) wat er tussen de vierkante haken waar is. `-lt` staat voor less than. Dus de loop gaat door zolang `COUNTER` kleiner dan 10 is. (Als hij 10 is, wordt de loop niet meer doorlopen.) In de loop wordt de inhoud van `COUNTER` getoond en wordt `COUNTER` 1 hoger gemaakt. Je ziet hier een verkorte notatie staan. Je kunt ook `let COUNTER=COUNTER+1` gebruiken. Je mag geen spaties rond de `=` en `+` hebben staan.

```
COUNTER=20
until [ $COUNTER -lt 10 ]; do
    echo $COUNTER
    let COUNTER-=1
done
```

De laatste soort loop: een `until`-loop. De loop wordt doorlopen totdat (`until`) wat tussen de vierkante haken staat niet meer waar is. Dus de loop gaat door totdat `COUNTER` kleiner dan 10 wordt. (Als hij 9 is, wordt de loop niet meer doorlopen.) In de loop wordt de inhoud van `COUNTER` getoond en van `COUNTER` 1 afgehaald.

```
function quit {
    exit
}
function hello {
    echo Hello!
}
hello
quit
echo test
```

Bash kent functies. Een functie staat tussen accolades en kan worden aangeroepen door de naam van de functie te gebruiken. Het is handig om vaak terugkerende handelingen in een functie te zetten. Eerst wordt de functie `hello` aangeroepen die "Hello!" laat zien. Vervolgens de functie `quit` die het script doet stoppen met het `exit`-commando. De laatste regel wordt daardoor nooit bereikt.

```
USER='whoami'

case "$USER" in
    "root") echo You're root;;
    *) echo You're not root;;
esac
```

In de variabele `USER` wordt de output van het `whoami`-commando gezet. Dat commando geeft de gebruikersnaam van de huidige gebruiker. Als (in the case) het `root` is, dan wordt "You're root" getoond. Bij elke andere inhoud (aangegeven door een `*`) van `USER` wordt "You're not root" getoond. Let op de ronde haakjes achter de mogelijkheden en de dubbele puntkomma's op het eind van elk onderdeel. Net zoals bij een `fi`-blok wordt ook het einde van een `case`-blok aangegeven door het woord `om te keren`.

```
OPTIONS="Hello Quit"
select opt in $OPTIONS; do
  if [ $opt = "Quit" ]; then
    echo done
    exit
  elif [ $opt = "Hello" ]; then
    echo Hello World
  else
    clear
    echo bad option
  fi
done
```

Select zorgt ervoor dat er een keuzemenu te voorschijn komt. Als opties in het menu worden de woorden in de variabele `OPTIONS` gebruikt. Zet tussen de opties een spatie. De optie die de gebruiker kiest wordt in de variabele `opt` gezet. Wordt `Quit` gekozen, dan wordt "done" getoond en het script gestopt. Wordt `Hello` gekozen, dan wordt "Hello World" getoond. Het bericht "bad option" hoor je niet te kunnen krijgen.

Zie <http://www.gnu.org/software/bash/manual/bash.html> voor de manual van Bash en <http://www.tldp.org/LDP/abs/html/> voor de Advanced Bash-Scripting Guide.

19 Errors

19.1 Bash: naam: command not found

De shell zoekt binary's en scripts alleen in de directory's die in de PATH-variabele staan. Dat zijn meestal o.a. /bin, /usr/bin en /usr/local/bin. Je kunt de PATH-variabele opvragen met **echo \$PATH**. Anders dan in DOS/Windows wordt de huidige directory niet doorzocht, tenzij die ook in PATH staat. De naam van een binary of script die in de huidige directory staat intikken, zal daarom zorgen voor de foutmelding "Bash: naam: command not found". De oplossing is om het complete pad of ./ (de huidige directory) voor de naam te zetten. Om de huidige directory aan de PATH-variabele toe te voegen, kun je **export PATH=\$PATH:.** gebruiken. Neem dat op in ~/.bash_profile als je dat voortaan altijd wilt.

19.2 Segmentation fault en bus error

Beide zijn een melding van een geheugenfout. Segmentation faults komen het meest voor. Een geheugenfout kan inhouden dat een programma buiten het toegestane geheugengebied wil schrijven. Er zijn twee mogelijke oorzaken van deze fout: het programma is gecompileerd voor andere versies van de benodigde library's of er zit een bug (programmeerfout) in. In het eerste geval wil het programma vaak überhaupt niet starten. Als er een bug in een programma zit, is het vaak zo dat het programma spontaan afgesloten wordt terwijl dat niet hoort. De meeste bugs worden in C gemaakt door het verkeerd gebruiken van pointers en arrays. Als het een bug is, zal die er in een volgende versie van het programma wellicht uit zijn. Ligt het aan je library's, kijk dan met **ldd prognaam** welke library's hij nodig heeft en check of je die wel hebt en of ze in orde zijn.

19.3 Use "exit" to leave the shell

Je kunt een shell in plaats van door **exit** in te tikken ook afsluiten met ctrl-d. Er bestaat echter een beveiliging tegen het per ongeluk indrukken van ctrl-d. Je kunt deze beveiliging uitschakelen met **export IGNOREEOF=0**. Je kunt in plaats van 0 ook het aantal keren opgeven dat EOF (End Of File, wat je invoert als je op ctrl-d drukt) moet worden genegeerd voordat het wel wordt doorgelaten.

20 Tips & Trucs

20.1 Beeld vol vreemde tekens

Het beeld kan vol komen te staan met vreemde tekens als je binaire data in de console of terminal laat zien. Dit kan de terminal verwarren, zodat alle tekst daarna uit vreemde karakters bestaat. Je kunt het verhelpen op een van de volgende manieren.

1. **reset**
2. **echo '\033c'**
3. **echo -ne '\017'**
4. Het direct screen escape command **echo ^O**.

Optie 1 is duidelijk het eenvoudigste om in te typen. De andere opties zijn mogelijkheden om het meer met de hand te doen. Het vreemde teken achter echo bij mogelijkheid 4 maak je door eerst op ctrl-v te drukken en vervolgens op ctrl-o.

20.2 Core dumps

Als een programma vastloopt, zal in een aantal gevallen een core dump worden gemaakt. De informatie daarin is voor een programmeur van nut om de fout die ervoor zorgde dat het programma vastliep te achterhalen. Je kunt het als gebruiker gebruiken om een bug report naar de auteur te zenden.

Je kunt je hele harde schijf laten afzoeken naar core-bestanden en deze verwijderen als ze langer dan vijf dagen niet benaderd zijn met (als root):

```
find / -name core -atime +5 -exec rm -f "{}" ';' 
```

Om ervoor te zorgen dat ze niet meer worden gemaakt, kun je in `/etc/profile` de regel **ulimit -c 0** zetten. Die 0 staat voor de grootte die core-bestanden maximaal mogen hebben.

20.3 Dingen tijdens het opstarten laten uitvoeren

Sommige distributies hebben een script als `/etc/rc.d/rc.local` of `/etc/rc.local` om extra dingen bij het opstarten te laten uitvoeren. Dit script zal als laatste worden uitgevoerd tijdens het bootproces. Het is daarmee enigszins vergelijkbaar met `autoexec.bat` onder DOS. Is er geen dergelijk script, dan zou je het ook wel zelf kunnen aanmaken. Maak `rc.local` (of elke naam die je leuk vindt) aan en maak een symlink genaamd `S99local` aan in `/etc/rc3.d` die wijst naar het script. Vervang hierin 3 door de runlevel waartoe standaard geboot wordt in je distributie. Zie voor meer informatie over runlevels paragraaf 8.1.

20.4 Firefox: snel een URL openen

Als je een URL hebt geselecteerd, dan kun je door op de middelste muisknop te klikken op een leeg plekje in een Firefox-scherf die URL openen. Een snelle manier om een link in een nieuw tabblad (of venster, afhankelijk van de instellingen) te openen is door er met de middelste muisknop op te klikken.

20.5 Functie ctrl-alt-del

Normaal gesproken wordt er bij het indrukken van de toetsencombinatie ctrl-alt-del een reboot uitgevoerd (tenzij je in X zit, dan wordt X er abrupt mee afgesloten). Je kunt de uit te voeren actie echter veranderen in `/etc/inittab`. Verander de **shutdown -r now** in de regel

```
ca::ctrlaltdel:/sbin/shutdown -r now
```

in wat je wilt laten gebeuren als je op ctrl-alt-del drukt. Bijvoorbeeld **shutdown -h now**. Gebruik **init q** om de wijzigingen door te voeren. Na een reboot is de wijziging ook van kracht.

20.6 Geheugen: er wordt minder gebruikt dan beschikbaar is

Kijk eerst in je BIOS naar de optie memory hole. Schakel die uit. Wordt dan nog niet al je geheugen gebruikt onder Linux, probeer dan de kerneloptie `mem=128M` (met in plaats van 128 de hoeveelheid geheugen). Dit kun je meegeven bij de prompt van LILO of GRUB. Als het goed werkt, neem het dan op in de configuratie van de bootloader. (Zie ook paragraaf 9.4.)

20.7 GNOME: instellingen buiten de desktopomgeving om

Je zult merken dat instellingen van GNOME, zoals de fonts en het thema, niet gebruikt worden als je niet de gehele GNOME-omgeving gebruikt. bijv. als je een GNOME-applicatie vanuit KDE start. Start `gnome-settings-daemon` in de achtergrond met **gnome-settings-daemon &** en de instellingen zullen wel gebruikt worden.

20.8 ISO en andere archiefformaten mounten

Je kunt een ISO-bestand mounten met een loopback device: **mount -t iso9660 -o loop isofile testdir**. Dan staat de inhoud van het ISO-bestand onder de directory `testdir` gemount.

Naast ISO bestaan er nog een aantal andere archiefformaten. Deze kunnen niet standaard door de kernel worden gemount. Er is wel `cdemu` (<http://cdemu.sf.net>), een kernelmodule waarmee je meer soorten bereiken kunt mounten. Zie de website voor het gebruik.

Met `bchunk` (<http://he.fi/bchunk/>) of `bin2iso` (<http://users.andara.com/~doiron/bin2iso/>) kun je een BIN/CUE omzetten naar ISO. Een archief van CloneCD, bestaande uit CCD/IMG/SUB, kan naar ISO geconverteerd worden met `ccd2iso` (<http://sourceforge.net/projects/ccd2iso/>) en voor NRG (van Nero) is er `nrg2iso` (<http://gregory.kokanosky.free.fr/v4/linux/nrg2iso.en.html>).

20.9 Knippen en plakken

De service `gpm` zorgt er niet alleen voor dat je de muis kunt gebruiken in de console. Het zorgt er ook voor dat je kunt knippen en plakken in de console. In X kan dat altijd al.

Om een stuk tekst te kopiëren, moet je het eerst selecteren door eroverheen te slepen terwijl je de linker muisknop ingedrukt houdt. Plakken doe je vervolgens door met de muiscursor op de gewenste positie te gaan staan en de middelste muisknop in te drukken.

In software van GNOME en KDE kun je knippen, kopiëren en plakken met resp. `ctrl-x` en `ctrl-c` en `ctrl-v`. In andere/oudere software kunnen de toetsen verschillen.

20.10 Magic SysRq

Indien opgenomen in de kernel kun je met de SysRq-toets (dezelfde toets als waar print screen op staat) direct commando's aan de kernel geven voor noodgevallen. Mogelijkheden zijn het synchen en unmounten van alle partities. Dat kan fouten in het bestandssysteem voorkomen als het systeem om een of andere reden mocht hangen. De kernel reageert vaak nog wel op deze toetsencombinaties. Kijk voor meer informatie in de directory /Documentation/sysrq.txt onder de kernel sources (gebruikelijk in /usr/src/linux).

20.11 Maximal mount count

Er wordt een teller bijgehouden die aangeeft hoeveel keer de partities gemount zijn. Als deze teller een bepaalde waarde bereikt, dan worden de partities gecontroleerd op fouten en wordt de teller weer op nul gezet. Je kunt het aantal keren dat er zonder controle opgestart mag worden veranderen. Daar heb je **tune2fs** voor nodig. Gebruik je een bestandssysteem met journaal, zoals ext3 of Reiser, dan kan het aantal keren ook best op 0 gezet worden. De desbetreffende partitie mag niet gemount zijn. Dezelfde instructies als in paragraaf 8.3 kunnen gevolgd worden. In plaats van e2fsck voer je tune2fs uit. Met **tune2fs -c #** verander je het aantal reboots. Vervang # door het gewenste aantal.

20.12 Melding bij het inloggen veranderen

De inhoud van /etc/issue wordt weergegeven vóór het inloggen in de console, de inhoud van /etc/motd (message of the day) erna. Je kunt hier je eigen boodschappen in opnemen.

Je kunt tekst in een console of terminal in kleur laten weergeven door gebruik te maken van zg. escape characters. Een voorbeeld:

```
echo "`[[44;33;01mtest`[[m"
```

Dit laat het woord test zien in gele letters op een blauwe achtergrond. Je moet niet alle tekens zo letterlijk invoeren. Tekens als ^[zijn escape characters. Die kun je in Bash (en Vim) als volgt aanmaken. Druk op ctrl-v (tijdens de insert mode in Vim) en vervolgens op de toets waar je het escape character van wilt hebben. Bij ^[is dat ESC.

Een voorbeeldinhoud voor /etc/issue met een tekst in kleur:

```
44;33;01m  
Welkom op mijn Linux-does
```

Let op dat bij sommige distributies /etc/issue elke keer bij het opstarten wordt overschreven door bijv. /etc/rc.d/rc.local. Je kunt de code die /etc/issue overschrijft uitcommenten of weghalen en een vaste boodschap in /etc/issue zetten, maar je kunt die code ook aanpassen. Dat laatste is handig als je variabelen in het bericht wilt zetten.

Zie paragraaf 20.18 over de kleurcodes.

20.13 MPlayer: stream opnemen

Videostreams die je via internet kunt bekijken, kun je downloaden met MPlayer. Je moet daarvoor wel eerst de URL van de stream te weten komen. Meestal kan dat door de link naar een stream op te slaan (in Firefox rechtsklikken op de link en kiezen voor Save Link As). Vaak is dit een klein tekstbestandje waar de echte link in staat. De URL begint met rtsp:// of mms://. Als het filmpje in de pagina "geëmbed" zit, is de URL

terug te vinden in de HTML-code van de pagina (in Firefox op te vragen door rechts te klikken op een leeg plekje van de pagina en dan te kiezen voor View Page Source).

Je kunt de stream opslaan met **mplayer url -dumpstream -dumpfile bestandsnaam**.

20.14 Muisknoppen omdraaien in X

Met het programma **xmodmap** kun je niet alleen de toetsenbordinstellingen veranderen, maar ook de instellingen van de muisknoppen. Wil je de knoppen van een drieknopsmuis omkeren (handig als je linkshandig bent), gebruik dan:

```
xmodmap -e 'pointer = 3 2 1'
```

En als je een scrollwielje hebt:

```
xmodmap -e 'pointer = 3 2 1 4 5'
```

Zie ook paragraaf 16.4.

20.15 Norton Commander-achtige file managers

Enkele Norton Commander-achtige file managers zijn:

- Midnight Commander (<http://www.ibiblio.org/mc/>) voor de command line
- Krusader (<http://krusader.sf.net>) voor KDE
- Tux Commander (<http://tuxcmd.sf.net>) voor GNOME
- emelFM2 (<http://emelfm2.net>) voor GNOME

20.16 Num-lock in alle consoles inschakelen

Het volgende script schakelt num-lock in op tty1 t/m tty6 (de consoles).

```
for i in 1 2 3 4 5 6
do /usr/bin/setleds +num < /dev/tty$i > /dev/null
done
```

Dit zou je bij het booten kunnen uitvoeren (zie ook paragraaf 20.3).

20.17 Omhoog scrollen

Je kunt in een console of terminal omhoog scrollen door op shift en page-up te drukken. Shift en page-down scrollt weer omlaag. Bij xterm kun je ook scrollen door op de schuifbalk (standaard links) te klikken. Om omhoog te scrollen moet je op de rechter muisknop drukken!

20.18 Output van ls in kleur

Het listing commando `ls` kan zijn uitvoer in kleur weergeven. Dat is niet alleen wat minder saai om naar te kijken, maar ook functioneel, omdat je er snel directory's en verschillende typen bestanden mee van elkaar kunt onderscheiden. Gebruik `ls --color=auto` om de output van `ls` in kleur te krijgen als dat nog niet standaard zo is. Er zijn een aantal standaard kleuren. Typ `dircolors -p` om daar een overzicht van te krijgen.

Wil je persoonlijke toevoegingen of aanpassingen maken, dan zul je de variabele `LS_COLORS` moeten zetten. Je kunt de output van `dircolors -p` gebruiken om een configuratiebestand voor `dircolors` te maken. Zet deze output in een bestand en pas het aan. Gebruik `dircolors -b file` (met in plaats van `file` onder welke naam je de configuratie hebt opgeslagen) om daar een bruikbare waarde voor de variabele `LS_COLORS` van te laten maken. Je kunt in `~/.bashrc` `eval 'dircolors -b file'` zetten om elke keer dat je Bash start deze persoonlijke kleuren te laten zetten.

Hier volgt een overzicht van kleurcodes en andere dingen die je kunt gebruiken.

Attribute	Text color	Background
00 none	30 black	40 black
01 bold	31 red	41 red
04 underscore	32 green	42 green
05 blink	33 yellow	43 yellow
07 reverse	34 blue	44 blue
08 concealed	35 magenta	45 magenta
	36 cyan	46 cyan
	37 white	47 white

Je kunt meerdere dingen tegelijk gebruiken door de getallen te scheiden met een puntkomma. Gebruik bijv. voor een vette zwarte tekst met rode achtergrond `01;30;41`.

20.19 Prompt aanpassen

De prompt is het stukje tekst in de shell waarachter je de commando's intikt. Deze prompt kun je zelf veranderen. Een standaard prompt voor alle gebruikers wordt gezet in `/etc/profile`. Per gebruiker kun je een eigen prompt opgeven in `~/.bashrc`. De prompt wordt bepaald door de variabele `PS1`. Twee voorbeelden:

```
export PS1='\u@$PWD>'
# PWD staat voor Print Work Directory. De prompt is van de vorm user@directory>

export PS1="[\u@\h] [\w]\\$"
# De prompt is van de vorm [gebruiker@host][directory]$
```

Voor alle mogelijke elementen in een prompt zie `man bash`.

20.20 Screenshot maken

Je kunt een screenshot van je desktop in X maken met `import -window root screenshot.png`. Vervang `root` door de titel van een window om een screenshot van dat window te maken. Gebruik je `import screenshot.png`, dan kun je door te klikken kiezen waar een screenshot van gemaakt moet worden. `Import` is een onderdeel van `ImageMagick`.

Er zijn verder nog veel meer andere tooltjes die een screenshot kunnen maken. Het tekenprogramma GIMP kan ook.

Je kunt ook de tekst uit de console naar een bestand schrijven om zo een soort screendump van de console te maken. Als je **script output.log** intikt, wordt alle output die daarna komt in output.log gezet. Dit stopt zodra je **exit** intikt. Met **setterm -dump -file output.dump** wordt er een screendump gemaakt van de huidige tekst in de console. Laat je de optie **-file** weg, dan wordt een bestand genaamd screen.dump aangemaakt.

20.21 Scripts in je home directory

Maak een directory bin in je home directory aan als deze er nog niet is. Deze directory staat meestal al standaard in de PATH-variabele, zodat je alle scripts die je daarin zet kunt starten door alleen hun naam in te tikken.

20.22 Sendmail: starten duurt erg lang

Waarschijnlijk heb je geen gekwalificeerde hostname (lees: je hostname komt niet op DNS-servers voor). Sendmail wacht dan op een DNS lookup. Hij stopt daarmee na een timeout. Dat kan na 1 tot 5 minuten zijn. Dit is op te lossen door je eigen hostname achter localhost te zetten in /etc/host.

20.23 Signature automatisch laten updaten

Een signature is een afsluiting voor je e-mail en berichten die je post in nieuwsgroepen. Veelal kan e-mailsoftware, zoals Thunderbird, dit uit een bestand naar keuze ophalen. Je zou het volgende scriptje kunnen gebruiken om een leuke signature te maken.

```
rm -f ~/.signature
echo "Je naam hier" >> ~/.signature
echo "Running Linux kernel $(uname -r)" >> ~/.signature
echo "$(uptime) >> ~/.signature
```

Om dit script elke vijf minuten te laten uitvoeren kun je een entry voor crontab (zie ook paragraaf 8.7) maken met een inhoud als:

```
*/5 * * * * /home/jan/bin/makesig
```

Een andere, bij niet veel mensen bekende, manier om je signature te laten updaten is een named pipe gebruiken. Dan wordt je signature alleen geüpdatet als dat nodig is. Maak eerst een named pipe met **mkfifo ~/.signature**. Laat daarna een script met een inhoud als het volgende in de achtergrond draaien.

```
while [ 1 ]; do
  ( echo "Je naam hier"
    echo "Running Linux kernel $(uname -r)"
    echo "$(uptime)"
  ) > .signature;
  sleep 1;
done
```

Deze tip is afkomstig van Maarten van Steenberg.

Opmerking: maak je signature maximaal vier regels lang. Meer wordt erg groot.

20.24 Standaard editor

Een aantal (command-line)programma's gebruikt als editor de waarde van de shell-variabele VISUAL of EDITOR. Die kun je in Bash veranderen met:

```
export VISUAL=editor
export EDITOR=editor
```

Zet op de plaats van editor je favoriete editor. Zet deze regels in ~/.bashrc om deze variabelen elke keer dat je Bash start te laten gebruiken. Is er geen editor opgegeven, dan zal Vi geprobeerd worden.

20.25 Tekstbestanden van/naar DOS-formaat omzetten

Unices maken alleen gebruik van een line feed (LF) om een nieuwe regel te laten starten, terwijl er in DOS/Windows een carriage return én een linefeed (CR + LF) gebruikt wordt. De carriage return is in een aantal editors zichtbaar als ^M. Je kunt de carriage returns weghalen op een van de volgende vier manieren.

Editor/shell	Manier
Vim	:%s/^M//g
Emacs	ESC x replace-string RET ctrl-q ctrl-m RET RET
Bash	sed 's/^M//' < dos.txt > linux.txt
Bash	cat dos.txt tr -d \r > linux.txt

De karakters die beginnen met ^ zijn "literal characters". Om deze in te voeren worden bij Vim dezelfde toetsen gebruikt als in Bash en bij Emacs dezelfde als in de C shell. Gebruik bij Vim en Bash ctrl-v met daarachter het gewenste karakter. ^M moet worden ingevoerd met ctrl-v en daarna enter. (Alternatief werkt ctrl-v ctrl-m ook.) Gebruik bij Emacs ctrl-q i.p.v. ctrl-v. Dit is in bovenstaand overzicht uitgeschreven.

Er is een klein tooltje genaamd dos2unix (<http://www.megaloman.com/~hany/software/hd2u/>) waarmee je de omzetting ook kunt doen.

20.26 Terughalen van verwijderde bestanden

Als je per ongeluk een bestand hebt verwijderd en je wilt het terughalen, dan heb je met ext2 grote kans dat dat kan als je maar niet veel nieuwe bestanden op de desbetreffende partitie hebt gezet. De inodes die van de verwijderde bestanden waren, moeten nog niet overschreven zijn.

Een methode om tekstbestanden met bekende tekst terug te vinden staat beschreven in de Linux HOWTO onder "desperate person's text file undelete". Methodes voor het terughalen van (ook binary) bestanden zijn te vinden in de Ext2fs Undelete mini-HOWTO.

Op <http://packetstormsecurity.nl/UNIX/utilities/> kun je een script genaamd unrm vinden, dat de methode die wordt beschreven in de laatstgenoemde HOWTO gebruikt. Als je verwijderde bestanden hebt teruggehaald met unrm, dan heb je misschien een hele hoop bestanden met een naam die je niet veel zegt

(ze worden `unrm.inodenummer` genoemd). Dan is de GNU-tool **file** ontzettend handig om een overzicht te krijgen van wat voor data overal in staat (gebruik **file ***).

Een andere (interactieve) tool is `e2undel` (<http://e2undel.sf.net>). Deze tool kan wel de oorspronkelijke bestandsnaam terughalen, maar alleen als je gebruikmaakt van een bijbehorende library, `libundel`. De bestandsnaam wordt normaal gesproken verwijderd, maar als deze library geladen wordt door alle processen die bestanden verwijderen, kan het in een logbestand bijhouden wat de bestandsnamen waren. Zie de documentatie voor meer details.

Zie voor het terughalen van verwijderde bestanden op een `ext3`-bestandssysteem:

- <http://projects.izzysoft.de/trac/ext3undel>
- <http://www.linux.com/feature/141074>
- http://www.xs4all.nl/~carlo17/howto/undelete_ext3.html

Lees over het terughalen van verwijderde bestanden op een Reiser-partitie http://www.antrix.net/journal/techtalk/reiserfs_data_recovery_howto.comments.

20.27 Titel van een terminal veranderen

Sommige programma's veranderen de titel van de terminal van waaruit je ze opstart, maar ze zetten het niet meer terug! Of je vindt het misschien wel leuk om de titel van de terminal te veranderen. Je kunt dat voor elkaar krijgen met: `echo -n -e "\033]2;nieuwe titel\007"`.

20.28 Uitschakelen van de pc-speaker

De pc-speaker kun je in de console uitschakelen door een van de volgende commando's te gebruiken: `setterm -blength 0` of `setterm -bfreq 0`. Ben je aan het werk onder X, dan kan dat met `xset -b`. Met mixerprogramma's zoals `aumix` kun je ook het volume van de speaker instellen.

21 Links

Hier volgen de interessantere links uit de rest van de handleiding en enkele nieuwe.

21.1 Beveiliging

- <http://www.chkrootkit.org> - chkrootkit, check op bekende rootkits
- <http://www.iptables.org> - iptables, firewall van kernel 2.4.x en 2.6.x
- <http://www.insecure.org/nmap/> - Nmap, poortscanner
- <http://www.courtesan.com/sudo/> - Sudo
- <http://www.openssh.com> - OpenSSH
- <http://www.chiark.greenend.org.uk/~sgtatham/putty/> - PuTTY, SSH-client voor Windows

21.2 Distributies

- <http://www.distrowatch.com> - DistroWatch
- <http://www.slackware.com> - Slackware
- <http://www.gentoo.org> - Gentoo
- <http://fedora.redhat.com> - Fedora Core
- <http://www.mandrivalinux.com> - Mandriva (voorheen Mandrake)
- <http://www.opensuse.org> - SUSE Linux
- <http://www.nl.debian.org> - Debian
- <http://www.ubuntulinux.org> - Ubuntu
- <http://www.knopper.net/knoppix/index-en.html> - Knoppix, live-cd met KDE
- <http://www.gnoppix.org> - Gnoppix, live-cd met GNOME

21.3 Documentatie

- <http://www.gnu.org> - de officiële site van GNU en de FSF
- <http://www.tldp.org> - Linux Documentation Project
- <http://doc.nl.linux.org/HOWTO/> - Nederlandstalige HOWTO's
- <http://www.pathname.com/fhs/> - File Hierarchy Standard

21.4 Emulators, virtuele machines e.d.

- <http://www.winehq.com> - Wine, open-source-implementatie van de Windows-API
- <http://www.codeweavers.com> - CodeWeavers CrossOver Office
- <http://www.transgaming.com> - TransGaming Cedega, een versie van Wine voor spellen
- <http://www.vmware.com> - VMware, een virtuele machine
- <http://dosbox.sf.net> - DOSBox, DOS-emulator
- <http://dosemu.sf.net> - DOSemu, DOS-emulator

21.5 Graphics

- <http://www.gimp.org> - GIMP (GNU Image Manipulation Program), tekenprogramma
- <http://www.inkscape.org> - Inkscape, vectortekenprogramma
- <http://www.gnome.org/projects/dia/> - Dia, diagramtekenprogramma

21.6 Hardware

- <http://leenooks.com> - Linux Incompatibility List
- <http://www.alsa-project.org> - ALSA, Advanced Linux Sound Architecture
- <http://www.linuxprinting.org> - informatie over printers
- <http://www.cups.org> - CUPS, het moderne printstelsel
- <http://www.sane-project.org> - SANE (Scanner Access Now Easy)
- <http://gphoto.sf.net> - digitale camera's
- <http://www.linuxtv.org> - digitale tv (met DVB-kaarten)

21.7 Internet en netwerken

- <http://www.apache.org> - Apache, webserver
- <http://www.mysql.com> - MySQL, een snelle database
- <http://www.postgresql.org> - PostgreSQL, database
- <http://www.php.net> - PHP, scripttaal
- <http://www.sendmail.org> - Sendmail, een MTA (Mail Transfer Agent)
- <http://www.postfix.org> - Postfix, idem
- <http://www.exim.org> - Exim, idem
- <http://fetchmail.berlios.de> - fetchmail, haalt mail op
- <http://www.proftpd.org> - ProFTPD, een FTP-server
- <http://www.wu-ftp.org> - WU-FTP, idem
- <http://www.samba.org> - Samba, voor bestandsdeling met Windows-machines
- <http://www.mozilla.org> - Mozilla, Firefox en Thunderbird
- <http://www.konqueror.org> - Konqueror, de file manager en tevens browser van KDE
- <http://www.mutt.org> - Mutt, tekstgebaseerd mailprogramma
- <http://www.washington.edu/pine/> - Pine, idem
- <http://www.pidgin.im> - Pidgin (voorheen Gaim), instant messenger voor MSN, ICQ en meer
- <http://kopete.kde.org> - Kopete, idem

21.8 Kernel

- <http://www.kernel.org> - de officiële site van de kernel
- <http://www.nl.kernel.org/pub/linux/> - een mirror in Nederland

21.9 Multimedia

- <http://www.k3b.org> - K3b, cd/dvd-brandprogramma
- <http://audacious-media-player.org> - Audacious, Winamp-clone (audiospeler)
- <http://amarok.kde.org> - Amarok, uitgebreidere audiospeler voor KDE
- <http://www.rhythmbox.org> - Rhythmbox, uitgebreidere audiospeler voor GNOME
- <http://www.mplayerhq.hu> - MPlayer, videospeler
- <http://xine.sf.net> - Xine, videospeler
- <http://www.gnome.org/projects/totem/> - Totem, mediaspeler van GNOME
- <http://www.real.com/linux> - RealPlayer
- <http://www.transcoding.org> - Transcode
- <http://www.exit1.org/dvdrip/> - dvd::rip
- <http://avidemux.sf.et> - avidemux, simpele video-editor
- <http://kino.schirmacher.de> - Kino, editor voor DV (digitale video)

21.10 Nieuws

- <http://slashdot.org> - Slashdot, news for nerds, stuff that matters
- <http://www.osnews.com> - OSNews
- <http://www.linuxtoday.com> - dagelijks nieuws

21.11 Office

- <http://www.openoffice.org> - OpenOffice.org
- <http://www.koffice.org> - KOffice
- <http://www.abisource.org> - AbiWord
- <http://www.gnome.org/projects/gnumeric/> - Gnumeric

21.12 Softwareoverzichten

- <http://freshmeat.net> - overzicht van open-sourcesoftware
- <http://sourceforge.net> - SourceForge, host veel open-sourceprojecten
- <http://gnomefiles.org> - software voor GNOME
- <http://kde-apps.org> - software voor KDE
- <http://www.happypenguin.org> - Game Tome, een overzicht van spellen

21.13 \TeX en \LaTeX

- <http://tug.org/teTeX/> - teTeX
- <http://www.latex-project.org> - \LaTeX
- <http://www.ctan.org> - \TeX Archive Network
- <http://tobi.oetiker.ch> - Lshort, the not so short introduction to \LaTeX
- <http://people.cs.uu.nl/piet/> - Nederlandstalige \LaTeX -handleiding

21.14 Utility's

- <http://www.gnu.org/software/parted/> - GNU parted, partitioneringstool
- <http://gparted.sf.net> - GParted, GTK+-frontend voor parted
- <http://qtparted.sf.net> - QtParted, Qt-frontend voor parted
- <http://www.megaloman.com/~hany/software/hd2u/> - dos2unix

21.15 Windows-gerelateerd

- <http://www.linux-ntfs.org> - NTFS en Linux
- <http://www.chrysocome.net/explore2fs> - Explore2fs
- <http://ext2fsd.sf.net> - Ext2fsd
- <http://www.fs-driver.org> - Ext2 Installable File System
- <http://www.p-nand-q.com/e/reiserfs.html> - rfstool

21.16 X

- <http://www.x.org> - X.org
- <http://xwinman.org> - overzicht van bekende window managers
- <http://www.gnome.org> - GNOME, desktopomgeving
- <http://www.kde.org> - KDE, idem
- <http://www.xfce.org> - Xfce, lichte desktopomgeving
- <http://rox.sf.net> - ROX desktop, idem
- <http://themes.freshmeat.net> - veel themes voor window managers en toolkits
- <http://art.gnome.org> - achtergronden, themes etc. voor GNOME
- <http://www.gnome-look.org> - idem
- <http://www.kde-look.org> - idem voor KDE