

Het Slackware Handboek

Voor Slackware Linux 9.1

Daniël de Kok

Paul Beurskens

Het Slackware Handboek: Voor Slackware Linux 9.1

door Daniël de Kok en Paul Beurskens

Copyright © 2002, 2003 door Daniël de Kok en Paul Beurskens

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

Linux is a registered trademark of Linus Torvalds. Slackware is a registered trademark of BSDi and Patrick Volkerding.

Inhoudsopgave

1. Over dit boek.....	1
Introductie	1
Conventies	1
2. Een introductie tot Slackware Linux	2
Wat is Linux?	2
Wat is GNU/Linux?.....	2
Wat is Slackware Linux?.....	2
Slackware Linux op CD-ROM.....	3
3. Hulpbronnen	4
Op je systeem	4
Op het internet.....	4
4. De installatie van Slackware Linux.....	6
De installatie CD-ROM starten	6
Partitionering van de harde schijf.....	6
Slackware installeren en configureren	7
5. Netwerkconfiguratie	9
Netwerk hardware	9
Configuratie van interfaces.....	9
PPP	10
Resolving	11
IPv4 Forwarding.....	12
6. Netwerkdiensten	13
De internet super server	13
Caching nameserver	13
De Apache webserver	14
7. Printerconfiguratie	15
Inleiding	15
Vorbereiding	15
Configuratie.....	15
8. Pkgtools.....	17
Introductie	17
pkgtool	17
installpkg.....	17
upgradepkg.....	17
removepkg.....	18
9. De Shell	19
Inleiding	19
De Shell omgeving	19
Commando's	20
Werken met de Shell	21
Redirection en Piping.....	22
Tab completion.....	23

Virtuele terminals	23
10. Bestanden en directories	25
De basis	25
Permissies.....	26
11. Processen	29
Inleiding	29
Voorgond processen	29
Achtergrond processen.....	30
ps	30
Signalen.....	31
Top.....	32
12. Netwerktols.....	33
De basis	33
Secure Shell (OpenSSH).....	34
13. XFree86.....	36
X Configuratie.....	36
Window manager	36
14. Vi	38
Opmerking vooraf	38
Wat is Vi	38
De eerste stap	38
Kopiëren, verwijderen, knippen en plakken.....	39
Tekst zoeken en vervangen.....	40
Vi configureren.....	41
Overzicht vi commando's	42
A. GNU Free Documentation License.....	43
PREAMBLE	43
APPLICABILITY AND DEFINITIONS	43
VERBATIM COPYING.....	44
COPYING IN QUANTITY	45
MODIFICATIONS.....	45
COMBINING DOCUMENTS.....	47
COLLECTIONS OF DOCUMENTS	47
AGGREGATION WITH INDEPENDENT WORKS.....	47
TRANSLATION	47
TERMINATION.....	48
FUTURE REVISIONS OF THIS LICENSE.....	48
ADDENDUM: How to use this License for your documents.....	48
B. Dankbetuigingen	50
Personen	50
Organisaties	50

Hoofdstuk 1. Over dit boek

Introductie

Oorspronkelijk is dit boek begonnen als een project dat het doel had het Engelstalige boek "Slackware Essentials" te vertalen naar het Nederlands. Naarmate de vertaling vorderde kwamen we tot de conclusie dat het om verschillende redenen verstandiger was helemaal opnieuw te beginnen en zelf een boek te schrijven, "Het Slackware Handboek".

Het Slackware Handboek is geschreven door Daniël de Kok en Paul Beurskens en is vrij beschikbaar onder de GNU Free Documentation License. Het boek is continu in ontwikkeling, niet alleen om bij te blijven met de nieuwste Slackware versies, maar ook om de documentatie te verfijnen en, waar nodig, uit te breiden. De nieuwste versie is te vinden op: <http://www.slackfiles.org/>

We wensen iedereen veel plezier met Slackware Linux en dit boek!

Conventies

Bestandsnamen

Bestandsnamen worden als volgt weergegeven: `/pad/naar/bestand`. Bijvoorbeeld: `/etc/fstab`

Commando's

Commando's worden vetgedrukt. Bijvoorbeeld: **ls -l**

Schermafdrucken

Schermafdrucken worden als volgt weergegeven:

Afdruk van een scherm.

Als er commando's uitgevoerd worden in een schermafdruk, zullen deze vet gedrukt worden:

```
$ commando  
Uitvoer
```

Als een commando als root uitgevoerd wordt, zal de shell in het voorbeeld een `"#"` tonen. Wordt een commando als normale gebruiker uitgevoerd, dan zal de shell een `"$"` tonen.

Hoofdstuk 2. Een introductie tot Slackware Linux

Wat is Linux?

Linux is een Unix-achtige kernel die geschreven is door Linus Torvalds en andere ontwikkelaars die verbonden zijn via het internet. Linux werkt op veel verschillende architecturen, waaronder de IA32, IA64, Alpha, m68k, SPARC en PowerPC platformen. Meer informatie en de nieuwste kernel is te vinden op: <http://www.kernel.org/>

Linux wordt vaak verward met het GNU/Linux systeem. Linux is alleen een kernel, geen compleet besturingssysteem. GNU/Linux is het GNU besturingssysteem met de Linux kernel zoals in de volgende sectie te lezen is.

Wat is GNU/Linux?

Richard M. Stallman (RMS) is begin jaren '80 begonnen met het ambitieuze project een vrij Unix-achtig besturingssysteem te schrijven. De naam van dit besturingssysteem is GNU (GNU is Not Unix). Begin jaren '90 waren alle belangrijke onderdelen van dit systeem geschreven behalve de kernel die HURD zou gaan heten. "The Hurd" is een POSIX compatibel multi-server systeem, dat bovenop de GNU Mach microkernel draait. Begin jaren '90 moest men eerst wachten tot de broncodes van Mach vrijgegeven werden. In de tussentijd heeft Linus het gat in het GNU systeem opgevuld met de Linux kernel. GNU/Linux verwijst dus naar het GNU systeem met de Linux kernel. Inmiddels is HURD ook een werkende kernel en bestaat er ook het GNU/Hurd systeem, het Debian (<http://www.debian.org/>) project heeft zelfs een versie van het GNU besturingssysteem gemaakt dat met de NetBSD (<http://www.netbsd.org/>) kernel werkt. Omdat GNU een belangrijk deel vormt van de meeste distributies zouden we eigenlijk over "GNU/Linux distributies" moeten spreken in plaats van "Linux distributies".

Wat is Slackware Linux?

Slackware Linux is een GNU/Linux distributie die hoofdzakelijk onderhouden en ontwikkeld wordt door Patrick Volkerding. Een verschil met andere distributies is dat Slackware het zogenaamde KISS (keep it simple stupid) principe volgt. Dat betekent dat grafische tools om het systeem te configureren ontbreken. In het begin kan de configuratie van het systeem daarom wat moeilijk zijn voor beginners, maar gevorderde gebruikers houden van de transparantie en flexibiliteit die deze benadering biedt. Bovendien leer je met Slackware Linux het GNU/Linux systeem tot op de bodem kennen.

Een ander opvallend aspect van Slackware Linux sluit aan op het KISS principe: Slackware Linux gebruikt geen package manager als RPM. De packages in de distributie zelf zijn normale tgz (tar/gzip) bestanden met soms een extra installatiescript. Tgz is voor ervaren gebruikers veel krachtiger dan RPM en vermijdt onzinnige dependencyproblemen die met RPM soms de orde van de dag zijn. Een eigenschap die ook vaak genoemd wordt, zijn de BSD-stijl initialisatiescripts van Slackware Linux. Dit betekent dat

er voor elk runlevel/taak één initialisatiescript is in plaats van een script voor elke daemon. Hierdoor is het systeem erg snel en flexibel in te stellen zonder dat je eerst zelf init scripts moet gaan schrijven.

De software in Slackware Linux is met zo weinig mogelijk veranderingen gecompileerd. Hierdoor kun je in Slackware veel algemene documentatie gebruiken en hoef je meestal niet op zoek naar distributiespecifieke informatie.

Slackware Linux op CD-ROM

Slackware Linux is op een aantal plaatsen op CD-ROM te krijgen. Hierbij is het belangrijk onderscheid te maken tussen de officiële CD-ROM set, waarvan een deel van de opbrengst ten goede komt aan de ontwikkeling van Slackware Linux, en goedkope kopieën. Voor veel studenten en mensen met weinig geld is een Slackware CD-ROM voor een paar Euro erg aantrekkelijk, maar in andere gevallen is het aan te raden de officiële CD-ROM set te kopen.

NedSlack houdt een lijst bij van verschillende Slackware leveranciers, inclusief een overzicht van de prijzen. Kijk voor meer informatie op <http://www.nedslack.org/>. De Slackware Linux Store is te vinden op <http://store.slackware.com/>.

Hoofdstuk 3. Hulpbronnen

Op je systeem

Linux HOWTO's

De befaamde Linux HOWTO's zijn een collectie van documenten die allemaal een bepaald facet van de configuratie van Linux behandelen. De meeste HOWTO's zijn min of meer distributieonafhankelijk en daarom erg geschikt voor Slackware Linux. Als je de linux-howtos package uit de "f" diskset hebt geïnstalleerd zijn de HOWTO's te vinden in `/usr/doc/Linux-HOWTOs/`. Vroeger was er ook een aparte collectie van Mini-HOWTO's die korter waren en specifiekere onderwerpen behandelden. Deze Mini-HOWTO's zijn nu opgenomen in de gewone HOWTO collectie.

Op het internet

Op verschillende plekken op het internet is hulp te krijgen als je problemen hebt met Slackware Linux. Veel van deze hulpbronnen worden mogelijk gemaakt door vrijwilligers die veel vrije tijd hierin investeren.

SlackFiles

SlackFiles is een internationale Slackware Linux site, die Nederlandstalige en Engelstalige documentatie biedt. Daarnaast is er veel andere informatie te vinden, zoals een overzicht van beschikbare Slackware georiënteerde IRC kanalen, links naar Slackware gerelateerde sites en andere informatie.

Website: <http://www.slackfiles.org/>

NedSlack

NedSlack is een Nederlandstalige Slackware Linux site met nieuws, informatie en links naar online documentatie. Ook zijn er rondom NedSlack een aantal Slackware gerelateerde projecten.

Website: <http://www.nedslack.org/>

NedSlack Forum

Het NedSlack forum is een Nederlandstalig Slackware forum. Hier kun je terecht met Slackware problemen of voor Slackware gerelateerde discussies.

Website: <http://forum.nedslack.org/>

alt.os.linux.slackware

alt.os.linux.slackware is een Slackware Linux nieuwsgroep. U kunt nieuwsgroepen lezen met een

newsreader als tin of knode. Let op: op deze nieuwsgroep (en de meeste andere nieuwsgroepen) wordt verwacht dat je alle documentatie goed hebt gelezen alvorens een probleem voor te dragen, als je dat niet doet is de kans om "geflamed" te worden vrij groot.

Hoofdstuk 4. De installatie van Slackware Linux

De installatie CD-ROM starten

De eenvoudigste methode om de installatie te starten is via de installatie CD-ROM. De Slackware installatie CD-ROM is een bootable CD, dat wil zeggen dat de BIOS de CD kan starten net als het bijvoorbeeld een diskette start. De meeste moderne systemen hebben een BIOS die vanaf CD-ROM kan starten.

Als de CD niet start wanneer je het in de CD-ROM speler van je computer hebt tijdens het starten van de computer staat de boot volgorde in de BIOS waarschijnlijk fout. Ga naar het BIOS menu (dit gebeurt meestal door de "del" of "esc" toets in te drukken bij het verschijnen van de BIOS startup tijdens het starten van het systeem) en controleer of CD-ROM vooraan staat in de bootvolgorde. Als de CD-ROM een SCSI CD-ROM is kan het nodig zijn de bootvolgorde in de SCSI BIOS in te stellen in plaats van de systeem-BIOS. Lees de handleiding van de SCSI kaart voor meer informatie.

Na het starten zal gevraagd worden of je een nationaal toetsenbord hebt. De meeste computers hebben gewoon een US/International toetsenbord, en volstaat het <enter> in te drukken bij deze vraag. Vervolgens verschijnt een login prompt. Log hier in als 'root', er zal niet om een wachtwoord gevraagd worden. Na het inloggen verschijnt de shell en kun je aan de slag met het installeren van het systeem. De installatie zal hierna kort uitgelegd worden.

Partitionering van de harde schijf

Voor het installeren van Slackware Linux heb je minimaal één partitie nodig en bij voorkeur ook een swap partitie, vooral als je weinig geheugen hebt. Om partities aan te kunnen maken moet er vrije ruimte op de schijf zijn (let op: dit is iets anders dan vrije ruimte op een andere partitie). Er zijn verschillende programma's die bestaande partities kunnen verkleinen, FIPS kan alleen FAT partities verkleinen. Partition Magic kan ook andere types partities verkleinen, maar is niet gratis. Mensen die meer expertise hebben zullen DFSee (<http://www.dfsee.com>) waarderen wegens via commando-gestuurde interface en het ongekend grote aantal mogelijkheden. DFSee is shareware.

Nadat de Slackware Linux CD-ROM gestart is en je bij de shell aanbeld bent heb je twee partitioneringsprogramma's tot je beschikking, **fdisk** en **cdisk**. **cdisk** is door de menu-interface het eenvoudigst en zal hier besproken worden. Als je de eerste hardeschijf gaat partitioneren kun je **cdisk** simpelweg starten met **cdisk**, wanneer je een andere hardeschijf wilt partitioneren of een SCSI schijf dan moet je de device die je wilt partitioneren meegeven aan **cdisk** (**cdisk /dev/device**). Een ATA hardeschijf wordt in Linux aangegeven met **/dev/hd**n waarin "n" vervangen wordt met een letter, zo is **/dev/hda** de "primary master" en **/dev/hdd** de "secondary slave". SCSI schijven worden aangegeven met **/dev/sd**n waarin "n" vervangen wordt met de letter van de schijf (1e SCSI schijf = a, 4e SCSI schijf = d).

Na het starten van **cdisk** verschijnt er een lijst van partities op je schijf en de hoeveelheid vrije ruimte. Je kunt een partitie of vrije ruimte selecteren met de omhoog/beneden cursortoetsen. Onderin het scherm staat een reeks commando's die je kunt selecteren met de links/rechts cursortoetsen. Een commando is uit te voeren met de <enter> toets.

U kunt een Linux partitie aanmaken door "Free Space" te selecteren en vervolgens "New" te kiezen uit de opties onderin het scherm. Daarna kun je kiezen of je een primaire of logische partitie wilt aanmaken. Het aantal primaire partities is beperkt tot vier. Linux kan zowel op een primaire als logische partitie geïnstalleerd worden. Als je andere besturingssystemen naast Slackware Linux gaat installeren die per sé op een primaire partitie moeten staan, dan is het verstandig Slackware Linux op een logische partitie te installeren. De aangemaakte partitie krijgt automatisch het typenummer van een Linux partitie meer, dus je hoeft het type partitie niet te veranderen.

Het maken van een swap partitie gaat op dezelfde wijze als een normale partitie, maar daarna moet het type van de partitie bijgesteld worden. De grootte van een swap partitie is afhankelijk van je eigen behoeften, de swap partitie wordt als geheugen gebruikt voor programma's als het RAM geheugen vol is. Als je een grote harde schijf hebt kan het geen kwaad om een partitie van 256MB of 512MB te maken, dan weet je zeker dat je (vrijwel) nooit te weinig geheugen zult hebben voor het laden van programma's. Na het aanmaken van een partitie kun je het type veranderen in "Linux Swap" door "Type" te selecteren in het menu onderin het scherm. Vervolgens kun je het typenummer van de partitie intikken, voor Linux Swap is dit 82. Normaliter zal er al 82 staan en voldoet het drukken op de <enter> toets.

Als je tevreden bent met de partitionering kun je het definitief opslaan op de schijf met de "Write" optie. Deze operatie moet bevestigd worden door "yes" te tikken. Vervolgens kun je cfdisk verlaten met behulp van de "Quit" optie. Het is verstandig de computer te herstarten voordat je aan de installatie begint, zodat de partitieveranderingen goed doorgevoerd worden. Druk op <ctrl> + <alt> + om Linux af te sluiten en de computer te herstarten.

Slackware installeren en configureren

De Slackware Linux installatie kan gestart worden door **setup** uit te voeren in de shell die verschijnt na het inloggen op de installatie CD-ROM. Setup presenteert een menu met een aantal opties. In principe moeten al deze onderdelen doorlopen worden voor een volledige installatie, maar na het afsluiten van een onderdeel zal setup vragen of je verder wilt gaan met het volgende onderdeel.

Het eerste installatieonderdeel is "ADDSWAP". Setup zal kijken welke partitie van het type "Linux Swap" is en vervolgens vragen of je deze partitie wilt activeren.

Daarna volgt "TARGET", waarmee je de partities waarop je Slackware wilt installeren kunt initialiseren en mounten voor de installatie. Setup toont in dit onderdeel alle partities die het type "Linux native" hebben. Wanneer je een partitie selecteert zal setup vragen of je de partitie wilt formatteren. Als je "yes" selecteert zal een lijst van typen bestandssystemen volgen. Normaliter zijn dit ext2, ext3 en reiserfs. Ext2 is al vele jaren het standaard bestandssysteem voor Linux, het nadeel is dat het geen journaling mogelijkheden heeft. Een journal is een bestand of een gebied op de hardeschijf waarin alle schijfbewerkingen geregistreerd worden zodat na een crash het bestandssysteem snel weer consistent gemaakt kan worden. Ext3 is Ext2 met een journaling extensie die journaling van zowel metadata als data ondersteunt. Reiserfs is een nieuw bestandssysteem, het biedt naast metadata journaling ook balanced trees en is daardoor sneller bij veel schijfoperaties, met name bij het werken met grote hoeveelheden kleine bestanden. Een nadeel is dat het, omdat het jonger is, iets minder betrouwbaar is dan Ext2 en Ext3. Als je stabiliteit en snelle herstelling na een crash wilt is Ext3 waarschijnlijk de beste keuze. Als er veel schijfbewerkingen plaats gaan vinden op het systeem (bijvoorbeeld als het als databaseserver gaat functioneren) is reiserfs een goede keuze.

Na het initialiseren van een partitie vraagt setup op welke lokatie (in de uiteindelijke directorystructuur)

de partitie gemount moet worden. De hoofdpartitie moet op de root directory (/) gemount worden. Als je aparte partities voor bepaalde directories heeft (bijvoorbeeld /home, /usr, /tmp en /var) kun je ze op dit moment mounten. Nadat u alle partities geïntialiseerd en gemount heeft kun je verder gaan door "Continue" te selecteren.

Het volgende onderdeel is "SOURCE". Hiermee kun je instellen vanaf welk medium je Slackware wilt installeren. Keuzes zijn onder andere vanaf CD-ROM, via NFS en via een van tevoren gemounte directory. Het meest frequent wordt er vanaf CD-ROM geïnstalleerd, dus dat zal hier behandeld worden. Wanneer je CD-ROM hebt geselecteerd zal gevraagd worden of je automatisch (auto) wilt laten detecteren in welk CD-ROM station de Slackware CD-ROM geplaatst is of handmatig. "Auto" zal alle CD-ROM drives doorzoeken en kijken of de directory /slackware op de CD staat. Als je voor "Manual" hebt gekozen dan toont setup een lijst van devices. Selecteer de device met de Slackware Install CD-ROM.

In de "SELECT" optie kun je aangeven uit welke disksets je packages wilt gaan installeren. Bij elke diskset staat in een korte beschrijving aangegeven wat je in elke diskset kunt vinden.

Daarna volgt het "INSTALL" onderdeel. Het zal eerst vragen op welke manier je wilt installeren. Voor de meeste installaties komen "full", "menu" of "expert" in aanmerking. "Full" zal alle packages in de geselecteerde disksets installeren. Dit is een eenvoudige manier van installeren. Het nadeel is dat het erg veel schijfruimte kan kosten en dat er misschien packages op het systeem geïnstalleerd worden die je toch niet gaat gebruiken. De "menu" optie vraagt per diskset welke packages je wilt installeren, "required" packages worden niet getoond omdat deze sowieso geïnstalleerd worden, "recommended" packages staan standaard aangevinkt, "optional" packages niet. De "expert" methode is hetzelfde als "menu", maar je kunt "required" packages ook selecteren/deselecteren. De "newbie" optie installeert alle "required" packages, maar vraagt bij elke andere package of je het wilt installeren. Deze optie kan interessant zijn voor beginners omdat het voor elke package een lange beschrijving toont, maar het maakt de installatie een lang proces en dat kan het heel erg vervelend maken. De "custom" en "tagpath" methodes zijn voor installaties waarbij aangepaste tagfiles gebruikt worden (wordt over het algemeen alleen gebruikt door gevorderde Slackware gebruikers).

Het laatste onderdeel, "CONFIGURE", loopt een aantal configuratieprogramma's door om onder andere LILO, het netwerk en muisondersteuning voor de console in te stellen. Deze programma's zijn goed van commentaar voorzien en vormen meestal geen obstakel bij de installatie.

Hoofdstuk 5. Netwerkconfiguratie

Netwerk hardware

Netwerkkkaarten

De drivers voor netwerkkkaarten worden tijdens de installatie geïnstalleerd als kernel modules. De module voor je netwerkkkaart dient geladen te worden tijdens het starten van het systeem. De kans is groot dat de netwerkkkaart tijdens de installatie als geconfigureerd is door de netwerk configuratie. Je kunt je netwerkkkaart opnieuw configureren met het **netconfig** commando. **netconfig** zorgt ervoor dat de module voor de gedetecteerde netwerkkkaart geladen wordt in `/etc/rc.d/rc.netdevice`.

Je kunt natuurlijk ook zelf instellen welke modules geladen moeten worden tijdens het starten van het systeem. Dit kunt je doen door een **modprobe** toe te voegen aan `/etc/rc.d/rc.modules`. Stel je wilt de module voor 3Com 59x kaarten laden (3c59c.o), dan voeg je de volgende regel toe aan

```
/etc/rc.d/rc.modules:
```

```
/sbin/modprobe 3c59x
```

PCMCIA kaarten

Ondersteunde PCMCIA netwerkkkaarten worden automatisch door de PCMCIA software gedetecteerd. De pcmcia-cs package uit de "a" diskset biedt PCMCIA functionaliteit voor Slackware Linux.

Configuratie van interfaces

Netwerkkkaarten zijn zichtbaar in Linux als zogenaamde "interfaces". Met het **ifconfig(8)** commando kunt je zien welke interfaces er zijn:

```
# ifconfig -a
eth0      Link encap:Ethernet  HWaddr 00:20:AF:F6:D4:AD
          inet addr:192.168.1.1  Bcast:192.168.1.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:1301 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1529 errors:0 dropped:0 overruns:0 carrier:0
          collisions:1 txqueuelen:100
          RX bytes:472116 (461.0 Kb)  TX bytes:280355 (273.7 Kb)
          Interrupt:10 Base address:0xdc00

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:77 errors:0 dropped:0 overruns:0 frame:0
          TX packets:77 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:8482 (8.2 Kb)  TX bytes:8482 (8.2 Kb)
```

Netwerkkarten krijgen de naam `ethn`, waarin `n` vervangen wordt met een nummer beginnend met 0. In het bovenstaande voorbeeld heeft de eerste netwerkkart, interface `eth0`, al een IP adres. Maar als de interfaces nog niet geconfigureerd zijn zul je in de **ifconfig** uitvoer zien dat ze geen IP adres hebben. Je kunt interfaces configureren in het bestand `/etc/rc.d/rc.inet1.conf`. De gevraagde waarden wijzen voor zichzelf, bijvoorbeeld:

```
# Config information for eth0:
IPADDR[0]="192.168.1.1"
NETMASK[0]="255.255.255.0"
USE_DHCP[0]=" "
DHCP_HOSTNAME[0]=" "
```

In dit voorbeeld krijgt de eerste interface (`eth0`) het IP adres `192.168.1.1` toegekend met de netmask `255.255.255.0`. Als je een DHCP server gebruikt (ofterwijl je IP adres toegekend wordt) dan kun je de regel `USE_DHP[n]=" "` vervangen met `USE_DHP[n]="yes"` (waarin "`n`" vervangen dient te worden met het interfacenummer). De andere waarden dan, op `DHCP_HOSTNAME` na genegeerd. Bijvoorbeeld:

```
IPADDR[1]=" "
NETMASK[1]=" "
USE_DHCP[1]="yes"
DHCP_HOSTNAME[1]=" "
```

Hetzelfde is van toepassing op andere interfaces. Je kunt de instellingen actief maken door het systeem opnieuw te starten of door **/etc/rc.d/rc.inet1** uit te voeren.

PPP

pppconfig

Voor het configureren van PPP heeft Slackware een speciale tool, **pppsetup**. Met **pppsetup** is het mogelijk je modem comfortabel via een menu-interface te configureren. Zowel voor het gebruik van PPP als **pppsetup** onder Slackware moet de `ppp` package uit de `a/` diskset geïnstalleerd zijn. Na het openingsscherm zal je gevraagd worden het telefoonnummer van je provider in te toetsen. Toets hier "ATDT" met daaraan het nummer van je provider en eventueel het nummer dat je moet gebruiken om op de buitenlijn te komen. In het volgende scherm kunt u selecteren welk serieel apparaat je wilt gebruiken voor de PPP verbinding. Voor externe modems is dit meestal `ttys0` of `ttys1`. Voor PCI modems kan dit variëren, bekijk de kernel uitvoer (met bijvoorbeeld **dmesg(8)**) om de juiste apparaatnaam te vinden. Vervolgens kunt je de baud rate van de modem instellen. Als je niet weet welke baud rate je modem heeft kan het over het algemeen geen kwaad om "115200" te kiezen. In het volgende scherm kun je opgeven of je internet provider je terugbelt. Bij vrijwel geen enkele internetprovider is dit het geval en kunt je hier "No" kiezen. Daarna kunt je een extra string met commando's voor de modem meegeven. Meestal is dit niet nodig en kunt je hier gewoon op enter drukken. Vervolgens zal je gevraagd worden het domeinnaam en de nameservers van je provider op te geven. Hiermee wordt `/etc/resolv.conf` geupdated, zodat namen om te zetten zijn in IP adressen. In de volgende schermen kun je de gegevens voor de authenticatie invullen. Eerst wordt gevraagd welke protocol je voor de authenticatie wilt gebruiken; de meeste providers gebruiken PAP. Daarna kun je de gebruikersnaam en wachtwoord invullen.

Nu PPP geconfigureerd is kun je verbinding maken met je internetprovider. De PPP verbinding wordt geregeld met een speciale daemon die op de achtergrond gestart wordt. Je kunt als volgt de daemon starten en een verbinding maken met jouw provider:

```
$ ppp-go
```

De verbinding is weer te verbreken met een ander simpel commando:

```
$ ppp-stop
```

Resolving

Hostnaam

Elke computer op het internet heeft een zogenaamde hostnaam. Ook als je geen vaste hostnaam heeft die via DNS beschikbaar is, is het toch handig om een hostname in te stellen omdat sommige software er gebruik van maakt. Je kunt je hostnaam instellen in `/etc/HOSTNAME`. Eén regel met je hostnaam voldoet. Een hostnaam heeft normaliter de volgende vorm: `host.domein.tld`, bijvoorbeeld `darkstar.slackfans.org`. Let op dat de hostnaam "resolvable" moet zijn, dat wil zeggen dat Linux in staat moet zijn het in een IP adres om te zetten. Dit kun je doen door de hostnaam aan `/etc/hosts` toe te voegen. Lees de volgende sectie voor meer informatie over dit bestand.

`/etc/hosts`

`/etc/hosts` is een tabel van IP adressen met bijbehorende namen. Dit bestand is ideaal om in een klein netwerk verschillende computers van een naam te voorzien. Dit is een voorbeeld van `/etc/hosts`:

```
127.0.0.1      localhost
192.168.1.1    tazzy.blowgish.org tazzy
192.168.1.2    gideon.blowgish.org
```

De localhost regel moet altijd aanwezig zijn. Het koppelt de naam "localhost" aan een speciale interface, de loopback. In dit voorbeeld is te zien dat de namen "tazzy.blowgish.org" en "tazzy" aan het IP adres 192.168.1.1 gekoppeld worden en de naam "gideon.blowgish.org" aan IP adres 192.168.1.2. Op het systeem met dit hosts bestand zijn beide computers met de genoemde namen.

`/etc/resolv.conf`

In het bestand `/etc/resolv.conf` kun je instellen welke nameservers het systeem moet gebruiken. Een nameserver zet hostnamen om in IP adressen. Hoogstwaarschijnlijk heeft je provider je ten minste twee nameservers (DNS servers) toegekend. Je kunt deze nameservers in dit bestand opgeven door per nameserver de regel "nameserver ipadres" toe te voegen. Bijvoorbeeld:

```
nameserver 192.168.1.1
nameserver 192.168.1.69
```

Je kunt controleren of hostnamen goed omgezet worden met behulp van het ping commando, door **ping** **hostnaam** uit te voeren. Hostnaam kunt je vervangen met bijvoorbeeld de website van je internetprovider (www.provider.nl).

IPv4 Forwarding

IPv4 forwarding maakt een verbinding tussen twee of meer netwerken door de pakketjes die op één interface naar binnen komen naar een andere interface te sturen. Dit maakt het mogelijk om een Linux computer te laten functioneren als een router tussen verschillende netwerken. Zo kun je bijvoorbeeld je thuisnetwerk koppelen met het internet. IPv4 forwarding kan ingeschakeld worden onder Slackware Linux door `/etc/rc.d/rc.ip_forward` uitvoerbaar te maken. Dit kun je doen met het **chmod** commando:

```
# chmod a+x /etc/rc.d/rc.ip_forward
```

IPv4 forwarding is weer uit te schakelen door het bestand onuitvoerbaar te maken:

```
# chmod a-x /etc/rc.d/rc.ip_forward
```

De instellingen kunnen actief gemaakt worden door de computer opnieuw starten. Ook is het mogelijk IPv4 forwarding op een lopend systeem uit te schakelen (of te activeren door 0 in het volgende voorbeeld in 1 te veranderen) met:

```
# echo 0 > /proc/sys/net/ipv4/ip_forward
```

Let op! Er zijn standaard geen filters actief. Dit betekent dat iedereen op het netwerk kan komen! Om binnenkomend en uitgaand verkeer te filteren en te controleren kun je onder Linux 2.4 kernels gebruik maken van **iptables**. Ook NAT (Network Address Translation) is een onderdeel van **iptables**. NAT maakt het mogelijk een netwerk te "verbergen" achter een IP adres, zodat u met slechts één IP adres toch een heel netwerk internettoegang kunt geven.

Hoofdstuk 6. Netwerkdiensten

De internet super server

Er zijn twee manieren om TCP/IP diensten aan te bieden: door de serversoftware als standalone daemon te draaien of via de internet super server, **inetd**(8). **inetd** is een daemon die een reeks poorten in de gaten houdt. Als er op een poort een connectie binnenkomt kijkt **inetd** of de dienst voor de desbetreffende poort ingesteld is. Is dat het geval dan wordt de connectie doorgesluisd aan de desbetreffende serversoftware, anders wordt de verbinding geweigerd. Het voordeel van dit mechanisme is dat het vrij goed te beveiligen en loggen is. Het nadeel is dat het niet optimaal werkt bij diensten die veel gebruikt worden door veel verschillende clients. Een veel gebruikte FTP server kan bijvoorbeeld beter een standalone FTP daemon draaien.

De configuratie van **inetd** wordt centraal geregeld via `/etc/inetd.conf`. Laten we eens naar een voorbeeldregel uit `inetd.conf` kijken:

```
# File Transfer Protocol (FTP) server:
ftp      stream  tcp      nowait  root    /usr/sbin/tcpd  proftpd
```

Deze regel specificeert dat **inetd** ftp connecties moet accepteren en deze kan laten verwerken door **tcpd**. Logischerwijs zou je kunnen denken dat de connecties door **proftpd** verwerkt zouden moeten worden. Dit is ook mogelijk, Slackware laat ze echter standaard door **tcpd**(8) verwerken. Dit programma geeft vervolgens de connectie door aan **proftpd**, zoals boven aangegeven is. Deze omleiding wordt standaard gedaan omdat **tcpd** de services in de gaten kan houden, daarnaast kan het gebruikt worden om toegang te beperken, etc.

Een service kan uitgeschakeld worden door het commentaarteken (#) voor de desbetreffende regel te zetten. Het is verstandig alle services eerst uit te zetten en vervolgens services die je gaat gebruiken aan te zetten. Na het wijzigen van `/etc/inetd.conf` moet **inetd** opnieuw gestart worden om de wijzigingen te activeren. Dit kun je doen door je systeem te herstarten of het HUP signaal naar het proces te sturen:

```
# ps ax | grep 'inetd'
64 ?        S          0:00 /usr/sbin/inetd
# kill -HUP 64
```

Caching nameserver

Een caching nameserver biedt DNS diensten aan een computer of netwerk, maar is zelf geen nameserver voor hosts of domeinen. Dat wil zeggen dat het alleen gebruikt wordt om hostnamen in IP adressen om te zetten. Een caching nameserver opzetten onder Slackware Linux is heel eenvoudig omdat BIND (de DNS software) standaard ingesteld staat als caching nameserver. Zelf hoeft je slechts 2 dingen te doen: BIND installeren en de initialatiescripts bijstellen. BIND kan geïnstalleerd worden door de `bind` package uit de "n" diskset te installeren. Vervolgens kun je BIND starten door het commando **named**(8) uit te voeren. Als je de caching nameserver altijd wilt starten moeten het `/etc/rc.d/rc.bind` bestand uitvoerbaar gemaakt worden. Dit kunt u doen met het **chmod** commando:

```
# chmod a+x /etc/rc.d/rc.bind
```

Indien je zelf ook gebruik wilt maken van de caching nameserver moet `/etc/resolv.conf` ook aangepast worden.

De Apache webserver

Apache is de meest gebruikte webserver op het internet en is natuurlijk ook onderdeel van Slackware Linux. Je kunt Apache installeren door de `apache` package uit de "n" diskset te installeren. Wil je PHP gebruiken, dan moet je ook de package `php` uit de "n" diskset installeren en de `mysql` package uit de "ap" diskset. MySQL is vereist omdat PHP shared libraries van PHP gebruikt. Je hoeft de MySQL server niet te draaien. Na de installatie kun je zorgen dat Apache `apache` altijd bij het opstarten gestart wordt door het bestand `/etc/rc.d/rc.httpd` uitvoerbaar te maken. Dit kun je doen met het volgende commando:

```
# chmod a+x /etc/rc.d/rc.httpd
```

De Apache configuratie is aan te passen in `/etc/apache/httpd.conf`. Apache kan op elk moment gestopt/gestart/herstart worden met het **`apachectl`** commando en de `stop`, `start` en `restart` parameters. Om Apache te herstarten voer je dus uit:

```
# apachectl restart
/usr/sbin/apachectl restart: httpd restarted
```

Hoofdstuk 7. Printerconfiguratie

Inleiding

GNU/Linux ondersteund op dit moment al een behoorlijk scala aan USB, parallele en netwerkprinters. Slackware Linux ondersteunt op dit moment twee printersystemen, CUPS en LPRNG. In dit hoofdstuk zal het CUPS systeem behandeld worden.

Onafhankelijk van welk printersysteem u gaat gebruiken kunt u een aantal collecties van printerfilters installeren. Deze zijn te vinden in de “ap” diskset. Als u goede printersupport wilt hebben is het in ieder geval aan te raden de volgende packages geïnstalleerd te hebben:

```
a2ps
enscript
espgs
gimp-print
gnu-gs-fonts
hpijs
ifhp
```

Beide printersystemen hebben hun eigen voor- en nadelen. Als u nog niet zo thuis bent in de configuratie van printers onder Linux, is het verstandig met CUPS aan de slag te gaan, omdat CUPS een comfortabele webinterface heeft die met een browser toegankelijk is.

Voorbereiding

Om CUPS te kunnen gebruiken moet de “cups” package uit de “a” diskset geïnstalleerd zijn. Na de installatie kunt u zorgen dat CUPS automatisch gestart wordt door het `/etc/rc.d/rc.cups` bestand uitvoerbaar te maken. Dit kunt u doen met het volgende commando:

```
# chmod a+x /etc/rc.d/rc.cups
```

Na het herstarten van het systeem wordt CUPS ook automatisch gestart. U kunt CUPS ook op een lopend systeem starten door het volgende commando uit te voeren:

```
# /etc/rc.d/rc.cups start
```

Configuratie

CUPS is te configureren via een webinterface. De configuratieinterface kan met elke willekeurige webbrowser bereikt worden op het volgende URL: `http://localhost:631/` Bij een aantal delen van de webinterface moet u zich eerst identificeren. Als CUPS hierom vraagt kunt u gewoon “root” als gebruikersnaam invoeren en het bijbehorende wachtwoord invullen.

Er kan een printer aan de CUPS configuratie toegevoegd worden door op “Administrate” te klikken en door vervolgens op de “Add Printer” knop te klikken. De webinterface zal vervolgens om een drietal opties vragen:

- *Name* - de naam van de printer. Gebruik een eenvoudige naam als bijvoorbeeld “epson”.
- *Location* - de fysieke plaats van de printer. Dit is niet echt een belangrijke instelling, maar handig om in te vullen binnen grotere organisaties.
- *Description* - beschrijving van de printer, bijvoorbeeld “Epson Stylus Color C42UX”.

U kunt verder gaan door op de “Continue” knop te klikken. Op de volgende pagina kunt u kiezen hoe de printer aangesloten is. Als u een USB printer heeft, en deze aanstaat, zal de webinterface de printernaam achter de desbetreffende USB poort zetten. Na dit scherm kunt u het printermerk en het printertype kiezen. Daarna is de printerconfiguratie voltooid en is de printer toegevoegd aan de CUPS configuratie.

Een overzicht van de geconfigureerde printers kan gevonden worden op de “Printers” pagina. Op deze pagina kunt u ook een aantal printerbewerkingen doen. Zo kunt u met de “Print Test Page” proberen of de printer werkt en goed geconfigureerd is.

Hoofdstuk 8. Pkgtools

Introductie

Slackware gebruikt in tegenstelling tot veel andere distributies geen ingewikkeld packagesysteem. Packages hebben de extensie `.tgz` en zijn gewoon tarballs die twee extra bestanden bevatten, namelijk een installatiescript en een beschrijving van de package. Omdat Slackware packages zo simpel zijn hebben ze geen dependency controle. Maar veel Slackware gebruikers geven hier de voorkeur aan omdat dependencies vaak meer problemen veroorzaken dan ze oplossen.

Om met packages om te gaan heeft Slackware Linux standaard een aantal tools. De belangrijkste tools zullen uitgelegd worden in dit hoofdstuk. Om de packagetools goed in de vingers te krijgen moet er kort wat uitgelegd worden over de naamgeving van packages. Stel we hebben een package met het bestandsnaam `bash-2.05b-i386-2.tgz`, dan is de naam van de package `bash-2.05b-i386-2`. Met het '-' symbool worden een aantal gegevens over de package gegeven. de betekenis is als volgt: "programmanaam-versie-architectuur-packagerevisie".

pkgtool

Het commando **pkgtool** biedt een menu-interface voor belangrijke packageoperaties. De belangrijkste opties zijn "Remove" en "Setup". De "Remove" optie presenteert de lijst van geïnstalleerde packages. Vervolgens kun je packages deïnstalleren door ze met de spatiebalk te selecteren en uiteindelijk je keuzes met de "enter" toets bevestigen. Met de spatiebalk kun je ook geselecteerde packages deselecteren.

De "Setup" optie geeft je toegang tot een aantal programma's die je helpen je systeem in te stellen, zoals **netconfig**, **pppconfig** en **xwmconfig**.

installpkg

Het **installpkg** commando wordt gebruikt om packages te installeren. **installpkg** moet altijd een packagebestand als parameter hebben. Als je bijvoorbeeld de package `bash-2.05b-i386-2.tgz` wilt installeren doe je dit als volgt:

```
# installpkg bash-2.05b-i386-2.tgz
```

upgradepkg

upgradepkg kan gebruikt worden om packages te upgraden. In tegenstelling tot **installpkg** installeert **upgradepkg** alleen packages wanneer de packages geïnstalleerd zijn en een oudere versie op het systeem aanwezig is. Om bijvoorbeeld alle packages uit een een packagedirectory op te waarderen voer je uit:

```
# upgradepkg *.tgz
```

Zoals gezegd zullen alleen de packages waarvan een oudere versie geïnstalleerd is opgewaardeerd worden.

removepkg

Het commando **removepkg** wordt gebruikt om een geïnstalleerde package te verwijderen. Stel dat we de package "bash" willen verwijderen (wat niet aan te raden is!), dan kan dit door het volgende commando uit te voeren:

```
# removepkg bash
```

Zoals je ziet wordt in de bovenstaande syntaxis alleen de naam van het programma mee te geven. Ook is het mogelijk een package te deïnstalleren door de volledige naam van de package op te geven:

```
# removepkg bash-2.05b-i386-2
```

Hoofdstuk 9. De Shell

Inleiding

Er zijn diverse manieren om werk op een computer uit te (laten) voeren. Zo kun je gebruik maken van een Grafische User Interface of programma's kunnen 'op de achtergrond' uitgevoerd worden. Een andere methode is om commando's via een zogenaamde prompt aan het systeem aan te bieden. Nu heb je op de laatste manier een programma nodig die deze opdrachten voor je aan het operating system aanbiedt of die deze opdrachten uitvoert. Op Linux systemen, en dus ook Slackware, heb je de beschikking over een zogenaamde shell. Deze shell zorgt er o.a. voor dat je werk uit kan voeren, zoals het bekijken van een bestand, het aanmaken van gebruikers etcetera.

Een shell wordt gestart op het moment dat een gebruik inlogt. Voor het inloggen zie je iets wat op het volgende lijkt:

```
slackware login:
```

Dit is de login prompt, niet te verwarren met de shell prompt. Na het inloggen en het starten van de shell wordt de volgende prompt getoond:

```
$
```

Dit is de prompt van de shell. Hierna kun je, afhankelijk van je rechten, de commando's uitvoeren die nodig zijn om je werk uit te voeren.

NB : In dit document wordt uitgegaan van de Bash shell.

De Shell omgeving

Als je met de Shell werkt, dan staan er een aantal variabelen tot je beschikking. Een voorbeeld is de PATH variabele, waarin een aantal paden naar directories staan zodat je voor bepaalde commando's alleen het commando hoeft in te voeren in plaats van het pad naar het commando en het commando zelf. Deze variabelen maken deel uit van de zogenaamde omgeving waarin je werkt (Environment). Je kunt een overzicht van je omgevingsvariabelen opvragen met het commando **env** en het commando **set**. Deze variabelen kun je ook zelf definiëren. Als je dit doet, gebruik dan voor de naam van de variabele hoofdletters. Dit is niet verplicht, maar wel een goede gewoonte. Het definiëren van een variabele doe je als volgt:

```
$ KREET=hallo
```

Hierboven is een variabele KREET gemaakt die de waarde 'hallo' gekregen heeft. Met het commando echo kun je bekijken of de opdracht inderdaad goed is gedaan:

```
$ echo $KREET
```

Let op het gebruik van het dollar(\$) teken.

Als je op deze manier Shell variabelen maakt, dan staan deze alleen ter beschikking tot de Shell. Als je ook wilt dat andere programma gebruik kunnen maken van deze variabelen, dan moet je deze variabelen "exporteren". Je doet dit op de volgende manier:

```
$ export KREET
```

Commando's

In dit deel worden een aantal commando's kort besproken.

write

Met dit commando kun je aan een andere gebruiker een bericht sturen. Het gebruik van dit commando is :

```
$ write user
```

Voor 'user' moet je de gebruiker invullen aan wie je een bericht wilt sturen.

w

Met dit commando kun je zien wie er allemaal op dit moment ingelogd zijn en wat ze aan het doen zijn. Het gebruik van dit commando is :

```
$ w
```

type

Met dit commando kun je snel bepalen waar een bepaald programma zich bevindt. Het gebruik van dit commando is :

```
$ type programma
```

Op de plaats van 'programma' vul je de naam in van het programma wat je zoekt. Voorwaarde is wel dat het programma zich bevindt in een van de directories die zich in de PATH variabele bevinden.

file

Met dit commando kun je bekijken om wat voor soort bestand het gaat. Het gebruik van dit commando is :

```
$ file bestand
```

Op de plaats van 'bestand' vul je de naam van het bestand in waarvan je wilt bekijken van voor type bestand het is.

ps

Met dit commando kun je bekijken welke processen er actief zijn op je systeem. Het gebruik van dit commando is :

```
$ ps
```

Hieraan kun je meerdere opties toevoegen (zoals a, u, x, s).

su

Met dit commando kun je tijdelijk als root gebruiker werken. Het is namelijk niet een goed idee om altijd als root gebruiker je werk. Om nu af en toe als root bepaalde commando's uit te kunnen voeren, kun je het commando su gebruiken. Het gebruik van dit commando is :

```
$ su
```

Na het invoeren van dit commando wordt er om het root password gevraagd.

locate

Met dit commando kun je op een eenvoudige manier het systeem doorzoeken voor een of meerdere bestanden. Het gebruik van dit commando is :

```
$ locate bestand
```

Op de plaats van 'bestand' vul je de naam in van het bestand (of programma of directory) wat je zoekt.

man

Met dit commando kun je informatie opvragen over een bepaald onderwerp. Het gebruik van dit commando is :

```
$ man opdracht
```

Op de plaats van 'opdracht' vul je de naam van de opdracht in waarover je meer informatie wilt weten. De documenten waar deze informatie zich bevindt worden manual pages genoemd.

Werken met de Shell

Via de Shell kun je verschillende dingen doen. Je kunt bovenstaande commando's uitvoeren, bestanden aanmaken, bestanden verwijderen en nog veel meer. Een belangrijk concept hierbij is hoe het "systeem" (lees : Shell) weet waar de commando's te vinden zijn die je intypt. Hier komt de variabele PATH (zie het deel over de Shell omgeving) om de hoek kijken. Je kunt programma's globaal op twee manieren uitvoeren via de Shell: d.m.v. het opgeven van alleen het commando of door het opgeven van het hele pad (dus waar het programma te vinden is, en het pad begint dan bij de root '/' directory).

Om programma's te kunnen uitvoeren door alleen de naam van het programma in te geven moet aan de voorwaarde voldaan zijn dat het pad naar dit programma in de PATH variabele is opgenomen. Stel dat een programma, bijvoorbeeld het programma `locate`, in de directory `/bin` staat, dan moet in de variabele PATH dit pad naar deze directory zijn opgenomen. Je kunt dit controleren met het commando `echo $PATH`. Als je nu een nieuw pad toe wilt voegen aan de PATH variabele, bijvoorbeeld omdat je een nieuw programma hebt geïnstalleerd en dit rechtstreeks wilt kunnen uitvoeren, dan kun je het op de volgende manier aan de variabele PATH toevoegen.

```
$ PATH=$PATH:/usr/bin
```

Bovenstaand voorbeeld voegt het pad `/usr/bin` toe aan de PATH variabele.

Om een programma van de huidige directory uit te voeren, dan volstaat het niet om alleen de naam van het programma op te geven, tenzij de huidige directory opgenomen is in de PATH variabele. Immers, de Shell zoekt alle paden af in de variabele PATH. Als het programma niet gevonden wordt, dan wordt deze ook niet uitgevoerd. Om nu toch een programma uit te kunnen voeren vanuit je huidige directory kun je dat als volgt opgeven:

```
$ ./programma
```

Hiermee geef je aan dat je 'programma' uit de huidige directory uit wilt voeren.

Redirection en Piping

Het is vaak handig om commando's te kunnen combineren, oftewel om de output van het ene commando door te geven als input voor een ander commando. Ook kan het bijvoorbeeld handig zijn om de output van een commando in een bestand op te slaan in plaats van op het scherm te tonen. Hier komen de kreten redirection en piping naar voren.

Redirecten is het veranderen van de richting van invoer en uitvoer van programma's. Stel dat je de inhoud van een bestand wilt bekijken. Dit kun je doen met het commando **cat**. Een voorbeeld:

```
$ cat file.txt
```

Bovenstaand voorbeeld laat de inhoud van een bestand op het scherm zien. Als je nu, voor wat voor reden dan ook, deze output naar een ander bestand wilt laten gaan, dan kun je dit als volgt doen:

```
$ cat file.txt > file2.txt
```

Bovenstaand voorbeeld redirect de uitvoer van het 'cat' commando naar een bestand in plaats van naar het scherm. Opmerking: je zou op kunnen merken dat dit eigenlijk een kopieerslag is en ja, daar heb je gelijk in. Echter, het gaat hier alleen maar om hoe redirection werkt

Een ander voorbeeld is het redirecten van de uitvoer van het **ps** commando naar een bestand. Voorbeeld:

```
$ ps > uitvoer.txt
```

Bovenstaand voorbeeld laat de uitvoer van het **ps** commando naar het bestand `uitvoer.txt` gaan in plaats van naar het scherm.

Bovenstaand voorbeeld stuurt de uitvoer van een commando naar een bestand. Als dat bestand nog niet bestaat wordt dat aangemaakt, maar als dat bestand wel bestaat, dan wordt de inhoud van dit bestand overschreven. Als je dat niet wilt, dan kun je besluiten om de inhoud van het nieuwe commando achter de bestaande inhoud van het bestand te plakken. Dit kun je als volgt doen:

```
$ ps > > uitvoer.txt
```

Je kunt ook i.p.v. redirecten naar bijvoorbeeld bestanden de uitvoer van een commando rechtstreeks aan een ander commando doorgeven. Dit wordt piping genoemd. Een voorbeeld van piping is:

```
$ ls -al | less
```

Bovenstaand voorbeeld geeft de output van **ls -al** door aan het commando **less**.

Tab completion

Met de tab toets kun je, tenminste met de Bash Shell, een deel van een commando of bestandsnaam of de naam van een pad invullen en de shell de naam zelf af laten maken. Scheelt je soms een hoop typewerk. Stel, je wilt een programma vanuit je huidige directory uitvoeren. Dit programma heeft de volgende naam : programmaxyz-234.hrl. Nu kun je als volgt de hele naam intypen:

```
./programmaxyz-234.hrl
```

Dat is veel typewerk en de kans dat je fouten maakt is groot, zeker met zo'n naam. Nu kun je een deel van de naam intypen, bijvoorbeeld:

```
./pro
```

Als je nu op de tab toets drukt, dan kijkt de shell of hij iets kan vinden wat begint met het deel wat je hebt ingevuld. Als dat zo is dan vult de shell de rest van de naam aan.

Nu kan het voorkomen dat je bijvoorbeeld twee programma's hebt met bijna dezelfde naam, bijvoorbeeld programmaxyz-123 en programmaxyz-124. Als je nu weer alleen 'pro' ingeeft en daarna op de tab toets drukt, dan vult de shell aan tot waar er verschillen zijn in de naam. Dus in dit voorbeeld tot :

```
./programmaxyz-12
```

Het laatste deel van de naam moet je nu zelf opgeven, in ieder geval tot en met het deel van de naam vanaf waar de shell constateert dat de naam uniek is. Vanaf het moment dat de naam weer uniek is, kun je weer de tab toets gebruiken.

NB : Tab completion geldt niet alleen voor programma's, maar ook voor bestanden, directories, complete paden en dergelijke.

Virtuele terminals

Je kunt via meerdere, zogenaamde virtuele terminals, werken. Dit kan handig zijn als je bijvoorbeeld meerdere taken uit wilt voeren en je niet steeds je huidige taak wilt stoppen en een andere taak wilt starten.

Omdat Linux een multi-user systeem is, kun je meerdere malen als dezelfde user ingelogd zijn maar je kunt ook met verschillende user-id's ingelogd zijn, bijvoorbeeld als 'gewone' user en als systeem beheerder (root).

Om met aan een andere taak te werken kun je met de toets combinatie Alt en een functie toets, wisselen tussen virtuele terminals. Alt-F2 geeft je de tweede virtuele terminal, Alt-F3 de derde enzovoort tot en met Alt-F6. De overige (Alt-F7 en verder) zijn gereserveerd voor X-sessies, dat wil zeggen een virtuele terminal voor sessies voor een X window (zeg maar grafische omgeving). Als je vanuit een grafische omgeving naar een tekst omgeving wilt, dan kun je dit doen met de toets combinatie Ctrl-Alt-F3 (voor de derde virtuele terminal). Alt-F7 stuurt je terug naar je grafische omgeving.

Hoofdstuk 10. Bestanden en directories

De basis

pwd

pwd(1) is een eenvoudige utility die de directory waarin je werkt toont. Voor het gebruik van **pwd** is het niet nodig een extra parameter te geven. Hier is een voorbeeld van de uitvoer van **pwd**:

```
$ pwd
/home/danieldk
```

ls

ls lijkt veel op het **dir** commando in DOS en Windows. Het is een commando dat de bestanden en directories in verschillende directories kan tonen. Door **ls** uit te voeren zonder extra parameters wordt de inhoud van de huidige directory directory getoond:

```
$ ls
slackware-beginselen  slackware-beginselen-20december2002.tar.gz
```

Natuurlijk kun je ook andere directories tonen. Dit kun je doen door simpelweg het pad te specificeren achter het **ls** commando:

```
$ ls /
bin  dev  home  lost+found  opt  root  tmp  var
boot  etc  lib  mnt          proc  sbin  usr
```

Het nadeel van de normale uitvoer is dat er weinig informatie gegeven wordt. Zo is het bijvoorbeeld niet mogelijk te zien of iets een bestand of directory is, wat de bestandsgrootte is en wie de eigenaar van een bestand is. Om meer informatie te tonen heeft **ls** de **"-l"** parameter:

```
$ ls -l
total 20
drwxr-xr-x  7 danieldk users      4096 Dec 21 09:24 slackware-beginselen
-rw-r--r--  1 danieldk users     14317 Dec 21 08:35 slackware-beginselen-20december2002
```

cd

Een ander onmisbaar commando is het **cd**. Met **cd** kun je de huidige werkdirectory veranderen:

```
$ cd /home/danieldk/
```

Met het **pwd** commando kun je zien dat je in de opgegeven directory directory staat:

```
$ pwd
/home/danieldk
```

mkdir

Zoals je waarschijnlijk wel kunt raden kun je met het **mkdir**(1) commando directories aanmaken. Een voorbeeld:

```
$ pwd
/home/danieldk
$ mkdir test
$ cd test
$ pwd
/home/danieldk/test
```

Het kan voorkomen dat je een directory aan wil maken waarvan de bovenliggende directory nog niet bestaat. Stel je wilt de directory `test2/hallo/` aanmaken in de huidige directory, maar de directory `test2` bestaat nog niet. In dat geval kun je beide directories op de volgende manier aanmaken:

```
$ mkdir -p test2/hallo
```

rm

rm(1) kan gebruikt worden om zowel bestanden als directories te wissen. Laten we eerst eens naar een eenvoudig voorbeeld kijken:

```
$ rm hello.c
```

Dit commando verwijdert het bestand `hello.c`. Soms vraag **rm** om bevestiging alvorens bestanden te verwijderen. Dit kun je omzeilen met de `'-f'` parameter:

```
$ rm -f *
```

Dit commando verwijdert alle bestanden in de huidige directory zonder daarbij om bevestiging te vragen. Het komt vaak voor dat we ook directories willen verwijderen, of zelfs hele directorystructuren. **rm** heeft hiervoor de `'-r'` parameter. Stel dat we de directory `ogle` met alle subdirectories willen verwijderen zonder hier bevestiging voor te geven. Dit kun je met het volgende commando doen:

```
$ rm -r -f ogle/
```

Het is bij veel commando's ook mogelijk parameterletters samen te voegen om wat tikwerk te besparen. Het volgende voorbeeld is dan ook het equivalent van het vorige:

```
$ rm -rf ogle/
```

Permissies

Een korte inleiding

In Linux heeft elk bestand zogenaamde permissies. Zoals je misschien al gezien hebt kun je de permissies van een bestand met het **ls -l** commando zien:

```
$ ls -l logo.jpg
-rw-r--r--  1 danieldk users      9253 Dec 23 19:12 logo.jpg
```

In het eerste kolom worden de permissies afgebeeld. De instelbare permissies zijn read(r), write(w) and execute(x). Deze permissies zijn in te stellen voor drie "groepen": owner(u), group(g) en others(o). Deze permissies zijn zichtbaar als de tweede tot de negende karakter in het eerste kolom. Deze negen karakters zijn in drieën verdeelt. De eerste drie karakters geven de permissies voor de eigenaar(owner) aan, de drie karakters die daarop volgen de permissies van de groep(group) en de laatste drie karakters de permissies van anderen (others). Naar bestand in het bovenstaande voorbeeld mag dus geschreven worden door de eigenaar en het bestand mag door iedereen (owner, group en others) gelezen worden.

Op het systeem zijn verschillende gebruikers aanwezig (een lijst is te zien in `/etc/passwd`) en een gebruiker kan weer lid zijn van bepaalde groepen. Op deze manier is het mogelijk per bestand vrij gedetailleerd in te stellen wie wat wel en niet mag doen. In het bovenstaande voorbeeld is te zien dat *danieldk* de eigenaar van het bestand is en dat group rechten gelden voor de groep *users*. De groepsrechten verschillen in het bovenstaande voorbeeld niet van die van andere mensen (others).

chown

chown(1) is een programma waarmee je de eigenaar van een bestand in kunt stellen en voor welke groep de groepsrechten gelden. Stel dat we *danieldk* eigenaar willen maken van het bestand `logo2.jpg`, dan kan dat als volgt:

```
$ chown danieldk logo2.jpg
```

We kunnen nu met **ls** zien dat de eigenaar *danieldk* is geworden:

```
$ ls -l logo2.jpg
-rw-r--r--  1 root      root      9253 Dec 29 11:35 logo2.jpg
$ chown danieldk logo2.jpg
$ ls -l logo2.jpg
-rw-r--r--  1 danieldk root      9253 Dec 29 11:35 logo2.jpg
```

Maar de groepspermissies gelden nog steeds voor de groep *root*. De groep waarvoor de groepspermissies gelden is te veranderen door een punt achter de eigenaar te zetten en daarachter de groep te vermelden waarvoor groepspermissies moeten gelden (in dit voorbeeld de groep *nedslackers*):

```
$ chown danieldk.nedslackers logo2.jpg
$ ls -l logo2.jpg
-rw-r--r--  1 danieldk nedslackers 9253 Dec 29 11:35 logo2.jpg
```

Natuurlijk is het ook mogelijk in één keer de rechten van een directorystructuur te veranderen dit kan met behulp van de recursive (-R) parameter:

```
$ chown -R danieldk.users oggs/
```

chmod

Permissies voor bestanden kunnen gewijzigd worden met het **chmod**(1) commando. De meest basale syntax van het chmod commando ziet er als volgt uit: **chmod [u,g,o][+/-][r,w,x] bestandsnaam**. De eerste parameter is dus een samenvoeging van: 1. voor welke groepen de operatie geldt, 2. of de

gespecificeerde permissies toegekend moeten worden (+) of afgenomen (-) moeten worden en 3. om welke permissies het gaat. Stel je wilt het bestand `notities` schrijfbaar maken voor de eigenaar van het bestand en de groep die voor het bestand ingesteld is, dan voer je het volgende commando uit:

```
$ chmod ug+w notities
```

Zoals je hieronder ziet wordt `notities` voor de eigenaar en de groep schrijfbaar:

```
$ ls -l notities
-r--r--r--    1 daniel  users          12 Mar  9 16:28 notities
bash-2.05b$ chmod ug+w notities
bash-2.05b$ ls -l notities
-rw-rw-r--    1 daniel  users          12 Mar  9 16:28 notities
```

Net als bij het `chown` commando is het met de recursive (-R) parameter mogelijk de rechten voor een directorystructuur te veranderen. Het volgende voorbeeld maakt de directory `geheim/` en alle bestanden en directories daaronder onleesbaar voor de ingestelde groep en andere gebruikers:

```
$ chmod -R go-r geheim/
```


Hoofdstuk 11. Processen

Inleiding

Elk programma wat gestart wordt is in wezen een proces. Het kan voorkomen dat een gestart programma zichzelf in meerdere taken verdeelt. In zo'n geval worden er meerdere processen gestart. Een voorbeeld van een programma dat zich in meerdere 'sub' taken verdeelt in de apache web server. Je ziet dan meerdere processen met dezelfde naam, in dit geval httpd. Onder de noemer programma kun je in dit geval zowel een systeem programma (ook wel daemon genoemd) verstaan als een programma die je als "gewone" gebruiker start, zoals je email programma. Dus elk gestart programma start als het ware een of meerdere processen op.

Elk proces is gekoppeld aan een gebruiker. Zo kunnen er bijvoorbeeld processen zijn die aan de root gebruiker zijn gekoppeld en processen die je zelf hebt gestart en die dus gekoppeld zijn aan jezelf.

Processen kunnen op meerdere manieren actief zijn. Voorbeeld daarvan zijn processen die op de voorgrond of op de achtergrond lopen.

Voorgrond processen

Een voorgrond proces is bijvoorbeeld een programma wat een gebruiker start vanuit een shell, bijvoorbeeld een email programma of een editor. Een typerend kenmerk van een voorgrond proces is dat het toetsenbord en beeldscherm gekoppeld zijn aan dat voorgrond proces. Er wordt in dit geval ook wel gesproken van 'Controlling Terminal'.

Een proces kan in de voorgrond gestart worden door vanuit de shell prompt een opdracht of programma naam in te typen. Voorbeeld:

```
$ vi
```

Bovenstaand voorbeeld start de vi editor in de voorgrond (foreground).

Je kunt, als een voorgrond proces eenmaal gestart is, deze naar de achtergrond verbannen. Om een proces wat als voorgrond proces gestart is naar de achtergrond te verplaatsen, gebruik je de toetsen combinatie Ctrl-Z. Op deze manier zet je het proces naar de achtergrond in de wacht. Hierna gebruik je het commando 'bg' om het proces verder op de achtergrond te laten lopen. Je kunt met het commando **jobs** een overzicht krijgen van taken die vanuit de shell gestart zijn:

```
$ jobs
```

De output zou als volgt kunnen zijn:

```
[1]- Stopped vi  
[2] Running xmms &  
[3]+ Stopped top
```

In bovenstaand voorbeeld zie je drie processen die vanuit de shell gestart zijn. Het proces (lees : job) xmms loopt zoals je ziet in de achtergrond. De andere twee staan te wachten. Je kunt een wachtend

proces of een proces wat in de achtergrond draait naar de voorgrond halen. Zoals je ziet heeft elk proces een nummer. Dit zijn in dit geval geen proces id's, maar een soort van 'job nummers'. Als je nu de job `xmms` weer naar de voorgrond wil halen dan kun je dit als volgt doen:

```
$ fg 2
```

Met bovenstaand commando haalt je het achtergrond proces met 'job' nummer 2 weer naar de voorgrond (met het commando `fg`). Het - teken in de lijst van de output van het `job` commando staat voor het laatst gestart proces en het + teken voor het voorlaatst gestarte proces.

Achtergrond processen

Een achtergrond proces is een proces wat niet gekoppeld is aan het toetsenbord en beeldscherm. Dit proces is wel actief, alleen dat lopende proces is niet te "zien" via je beeldscherm.

Een proces kun je als volgt in de achtergrond starten:

```
$ xmms &
```

Het ampersand (&) teken wordt gebruikt om een proces in de achtergrond te starten. Als je nu het commando `jobs` gebruikt zie je dat een proces in de achtergrond verder loopt. Let wel : dit geldt niet voor elk proces. Er zijn programma's die in de achtergrond alleen in de wacht gezet kunnen worden. Een voorbeeld hiervan is de editor `vi`, welke alleen gesuspend (in de wacht gezet) kan worden.

```
[1]  Running  sleep 10000 &
[2]  Running  xmms &
[3]  Stopped  vi
```

Zoals je ziet is de status van twee jobs running en die van `vi` stopped terwijl ze alledrie op dezelfde manier gestart zijn (met het '&' teken).

ps

Het commando `ps`(1) gebruik je om een overzicht te krijgen van processen onafhankelijk of ze in de voorgrond of achtergrond draaien. Elk proces krijgt een nummer toegewezen, een zogenaamd proces id. Dit proces id, ook wel PID genoemd, kan erg nuttig zijn om te weten. Het commando `ps` kent verschillende opties die je mee kunt geven. Als je geen opties mee geeft, dan krijg je alleen een lijst van de processen die door je zelf gestart zijn. Voorbeeld:

```
$ ps
PID TTY TIME  CMD
12 pts/1 00:00:00 bash
15 pts/1 00:00:00 vi
```

In bovenstaand voorbeeld staat PID voor het proces ID, TTY is de terminal die bij dit proces hoort, TIME geeft de gebruikte CPU tijd aan tot op heden en CMD staat voor de naam van het gestarte programma. Als je dit commando uitbreid met een aantal opties, bijvoorbeeld de opties `a`, `u` en `x` als volgt:

```
$ ps -aux
```

dan kan de output er als volgt uitzien:

USER	PID	%CPU	%MEM	VSZ	RSS	TTY	STAT	START	TIME	COMMAND
root	1	0.0	0.0	448	64	?	S	12:21	0:03	init
root	2	0.0	0.0	0	0	?	SW	12:21	0:00	[keventd]
root	3	0.0	0.0	0	0	?	SW	12:21	0:00	[kapmd]
root	4	0.0	0.0	0	0	?	SWN	12:21	0:00	[ksoftirqd_CPU0]
root	5	0.0	0.0	0	0	?	SW	12:21	0:01	[kswapd]
root	6	0.0	0.0	0	0	?	SW	12:21	0:00	[bdflush]
root	7	0.0	0.0	0	0	?	SW	12:21	0:00	[kupdated]
root	8	0.0	0.0	0	0	?	SW	12:21	0:00	[kinoded]
root	9	0.0	0.0	0	0	?	SW<	12:21	0:00	[mdrecoveryd]

Zie de manual page van het **ps** commando voor meer uitleg (**man ps**).

Signalen

Met het commando **kill(1)** kun je processen beïnvloeden. Dit commando kun je gebruiken om processen te stoppen, maar bijvoorbeeld ook om ze te herstarten. Om processen te beïnvloeden die van een andere gebruiker zijn, moet je dit commando als root gebruiker uitvoeren. Je kunt wel als gewone gebruiker je eigen processen beïnvloeden.

Bij het commando **KILL** draait het eigenlijk om het sturen van een signaal naar een proces. Je kunt verschillende signalen naar een proces sturen en standaard wordt het **TERM** signaal naar een proces gestuurd wat in feite zegt: proces beëindig je zelf. Hieronder volgt een overzicht van een aantal verschillende signalen die je naar een proces kan sturen.

```
1) SIGHUP  2) SIGINT  3) SIGQUIT  4) SIGILL
5) SIGTRAP 6) SIGABRT 7) SIGBUS  8) SIGFPE
9) SIGKILL 10) SIGUSR1 11) SIGSEGV 12) SIGUSR2
13) SIGPIPE 14) SIGALRM 15) SIGTERM 17) SIGCHLD
```

De meest gebruikte signalen zijn **TERM** voor stoppen, **KILL** als een proces hardhandig de nek omgedraaid moet worden en **HUP** als een proces herstart moet worden. Dus als een proces het signaal **HUP** (hang-up) ontvangt, zal dit proces zich herstarten.

Niet alle signalen worden door het proces zelf verwerkt. Als een proces zelf nergens meer op reageert kun je deze met een **SIGKILL** stoppen. Dit signaal wordt door de kernel verwerkt.

Zoals je misschien opgevallen is, beginnen alle signalen met de letters **SIG**. Als je een signaal wilt versturen naar een proces, dan kun je dit als volgt doen:

```
$ kill -optie PID
```

Op de plaats van 'optie' plaats je een deel van een signaal wat je wilt versturen. Stel je wilt een **SIGTERM** versturen, dan vul je op de plaats van de optie het woord **TERM** in. Op de plaats van **PID** vul je het proces ID in, welke je op kunt zoeken met het **PS** commando.

Overigens kun je in plaats van de woorden **TERM**, **HUP** en dergelijke ook met getallen werken. Met het commando **kill -l** zie je een lijst van de signalen met de bijbehorende getallen. Het commando wordt dan:

```
$ kill -1 xcms
```

Bovenstaand commando stuurt een boodschap naar het proces dat hij zichzelf moet herstarten (HUP).

Top

Het commando **top**(1) kun je gebruiken om dynamische informatie over processen te zien. Dynamisch in de zin dat je direct veranderingen kunt zien, bijvoorbeeld t.a.v. CPU verbruik, hoeveel processen er actief zijn en dergelijke. Dit commando voer je als volgt uit:

```
$ top
```

De uitvoer van dit programma kan er als volgt uitzien:

```
9:18pm up 8:57, 7 users, load average: 0.03, 0.05, 0.06
73 processes: 71 sleeping, 1 running, 0 zombie, 1 stopped
CPU states: 3.7% user, 0.7% system, 0.0% nice, 95.4% idle
Mem: 223152K av, 190712K used, 32440K free, 0K shrd, 4920K buff
Swap: 104412K av, 13028K used, 91384K free 119232K cached
```

PID	USER	PRI	NI	SIZE	RSS	SHARE	STAT	%CPU	%MEM	TIME	COMMAND
14678	neskio	19	0	980	980	764	R	2.7	0.4	0:00	top
1	root	20	0	76	64	44	S	0.0	0.0	0:03	init
2	root	20	0	0	0	0	SW	0.0	0.0	0:00	keventd
3	root	20	0	0	0	0	SW	0.0	0.0	0:00	kapmd
4	root	20	19	0	0	0	SWN	0.0	0.0	0:00	ksoftirqd_CPU0
5	root	20	0	0	0	0	SW	0.0	0.0	0:01	kswap
6	root	20	0	0	0	0	SW	0.0	0.0	0:00	bdf flush
7	root	20	0	0	0	0	SW	0.0	0.0	0:00	kupdated
8	root	20	0	0	0	0	SW	0.0	0.0	0:00	kinoded
9	root	0	-20	0	0	0	SW<	0.0	0.0	0:00	mdrecoveryd
380	root	20	0	256	216	144	S	0.0	0.0	0:00	syslogd
383	root	20	0	672	172	116	S	0.0	0.0	0:00	klogd
419	root	20	0	0	0	0	SW	0.0	0.0	0:00	khubd
605	root	19	0	188	4	4	S	0.0	0.0	0:00	cardmgr
618	bin	20	0	224	192	136	S	0.0	0.0	0:00	portmap
665	root	18	0	252	4	4	S	0.0	0.0	0:00	sshd
696	at	20	0	152	100	64	S	0.0	0.0	0:00	atd
719	lp	20	0	548	520	424	S	0.0	0.2	0:00	lpd
761	root	20	0	172	112	76	S	0.0	0.0	0:00	cron
779	mail	20	0	816	308	216	S	0.0	0.1	0:00	sendmail
785	root	20	0	972	364	232	S	0.0	0.1	0:00	sendmail
863	root	20	0	448	416	296	S	0.0	0.1	0:00	nscd
868	root	20	0	448	416	296	S	0.0	0.1	0:00	nscd

Hoofdstuk 12. Netwerktools

De basis

ping

ping(8) is een stuk gereedschap dat zogenaamde ICMP ECHO_REQUEST packets naar hosts stuurt. Dat wil zeggen dat er een verzoek wordt gedaan aan een host om te antwoorden. Dat maakt deze tool uitermate geschikt om te testen of bepaalde hosts bereikbaar zijn of om te controleren of je internet connectie goed werkt. Laten we kijken naar een voorbeeld van het gebruik van dit commando:

```
bash-2.05a$ ping tazzy.blowgish.org
PING tazzy.blowgish.org (192.168.1.1): 56 octets data
64 octets from 192.168.1.1: icmp_seq=0 ttl=64 time=0.0 ms
64 octets from 192.168.1.1: icmp_seq=1 ttl=64 time=0.0 ms
64 octets from 192.168.1.1: icmp_seq=2 ttl=64 time=0.0 ms

--- tazzy.blowgish.org ping statistics ---
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
```

Zoals te zien is de eenvoudigste syntaxis van dit commando "**ping hostnaam**". Aan dit voorbeeld is al veel informatie af te leiden. In de eerste regel wordt het IP adres van de computer getoond. Mocht dit niet gebeuren, dan zijn waarschijnlijk de nameserver instellingen niet goed of bestaat de hostnaam simpelweg dan niet. Daarna zal **ping** constant doorgaan met het sturen van beantwoordingsverzoeken. Je kunt elke keer zien wat de benodigde tijd was, wat bruikbaar is om simpele snelheidsmetingen te doen. **ping** is te stoppen door tegelijkertijd de "control" en "c" toetsen in te drukken. Hierna verschaft **ping** nog enige informatie, waaronder het percentage pakketverlies (packet loss). Om te zien of een verbinding betrouwbaar is kunt je **ping** een tijdje laten draaien en na beëindiging de packet loss bekijken.

host

Met het commando **host**(1) kun je informatie over hosts opvragen met DNS. Standaard geeft **host hostnaam** het IP adres van de desbetreffende host:

```
$ host www.nedslack.org
www.nedslack.org has address 213.239.135.210
```

Ook kun je andere records/informatie opvragen voor hosts en/of domeinen. Lees voor meer informatie de **host** man pagina.

traceroute

traceroute(8) probeert alle gateways te tonen die gepasseerd worden bij het contact maken met een bepaalde host. Zo is te zien hoeveel "hops" er nodig zijn voor een pakket om bij een bepaalde host te komen en die tijd die ervoor nodig was. Hier is een voorbeeld van **traceroute** in actie:

```
$ traceroute www.nedslack.org
traceroute to www.nedslack.org (213.239.135.210), 30 hops max, 38 byte packets
 1  192.168.1.69 (192.168.1.69)  0.726 ms  0.667 ms  0.562 ms
 2  nas.euro.net.nl (194.134.5.1)  153.248 ms  158.397 ms  194.148 ms
 3  F0-0.dr2-rt2.nl.euro.net (194.134.250.67)  312.472 ms  173.391 ms  146.758 ms
 4  P1-1-0.dr2-asd3.nl.euro.net (194.134.181.81)  343.253 ms  252.822 ms  211.670 ms
 5  P4-0.xr1-fft1.de.euro.net (194.134.181.78)  454.119 ms  219.295 ms  446.982 ms
 6  gel-2-0.de-cix.fra3.trueserver.de (80.81.192.51)  153.319 ms  156.294 ms  162.161 ms
 7  www.nedlinux.nl (213.239.135.210)  166.122 ms  167.412 ms  165.502 ms
```

Secure Shell (OpenSSH)

Vroeger werden de **telnet** en **rsh** utilities gebruikt om op andere systemen in te loggen om er vervolgens op afstand op te kunnen werken. Het nadeel van beide programma's is dat zowel de wachtwoorden als data (toetsenbordinput en schermoutput) ongecodeerd over het netwerk gestuurd wordt. Iemand die een netwerk "afluistert" kan op die manier de inloggegevens voor shell accounts achterhalen en misbruiken. Secure Shell (SSH) maakte een einde aan deze onbeveiligde gegevensoverdracht door sessies te coderen, op die manier zijn verstuurd wachtwoorden en andere data niet te achterhalen door eventuele afluisteraars.

De SSH implementatie die in de meeste Linux distributies gebruikt wordt, is ontwikkeld door het OpenBSD team en heet OpenSSH. OpenSSH is een verzameling van commando's die gebruikt kunnen worden voor beveiligde gegevensoverdracht. OpenSSH kan onder andere gebruikt worden voor het werken op andere computers (remote shell) en het versturen van bestanden.

ssh

Het **ssh(1)** commando wordt normaliter gebruikt voor het inloggen en werken op andere computers. De syntaxis is simpelweg **ssh hostnaam**, waarin *hostnaam* ook vervangen kan worden met het IP adres van het systeem waarop ingelogt moet worden. Bijvoorbeeld:

```
$ ssh 192.168.1.69
daniel@192.168.1.69's password:
```

ssh zal standaard met dezelfde gebruikersnaam proberen in te loggen als je ingelogt bent op de machine waarop **ssh** gestart wordt. Een andere gebruikersnaam kan opgegeven worden met de **-l** parameter of met de e-mailnotatie. Een voorbeeldsessie met de **-l** parameter:

```
$ ssh -l danieldk 192.168.1.69
danieldk@192.168.1.69's password:
Last login: Sat Mar  8 17:37:28 2003 from 192.168.1.1
NetBSD 1.6 (GENERIC) #0: Sun Sep  8 19:43:40 UTC 2002
Erase is backspace.
tweety: {1}
```

Met e-mailnotatie:

```
$ ssh danieldk@192.168.1.69
danieldk@192.168.1.69's password:
```

```
Last login: Sat Mar  8 17:36:51 2003 from 192.168.1.1
NetBSD 1.6 (GENERIC) #0: Sun Sep  8 19:43:40 UTC 2002
Erase is backspace.
tweety: {1}
```

sftp

sftp is een handig commando waarmee het mogelijk is met normale ftp commando's bestanden te versturen tussen verschillende systemen. Aan de serverkant is sftp een onderdeel van de **sshd** daemon. Om met **sftp** een verbinding te maken met een ander systeem moet je op dat systeem een shell account hebben. De syntaxis is vrijwel identiek aan die van het **ssh** commando:

```
sftp danieldk@192.168.1.69
```

scp

Ook is het mogelijk met OpenSSH bestanden te kopiëren op een manier die vergelijkbaar is met het **cp** commando, namelijk met **scp**. Bestanden op andere servers worden als volgt aangegeven: *gebruiker@host:bestandsnaam*. Stel dat ik het bestand `logo.jpg` naar mijn account op *192.168.1.69* wil kopiëren, dan kan dat als volgt:

```
$ scp logo.jpg danieldk@192.168.1.69:
```

Omdat ik geen directory of bestandsnaam heb gespecificeerd wordt `logo.jpg` met het bovenstaande commando naar mijn home directory gekopieerd. Stel dat ik het bestand naar de `www/` directory in mijn home directory wil kopiëren, dan volstaat het volgende commando:

```
$ scp logo.jpg danieldk@192.168.1.69:html/
```

Andersom werkt het ook. Stel dat ik het bestand `essay` wil kopiëren vanaf de server naar de huidige directory, dan kan dat met het volgende commando:

```
$ scp danieldk@192.168.1.69:essay .
```

Overigens kun je net als bij **scp** recursief bestanden kopiëren met de `-r` parameter:

```
$ scp -r danieldk@192.168.1.69:essays/ .
```

Hoofdstuk 13. XFree86

X Configuratie

De configuratie van XFree86 is opgeslagen in `/etc/X11/XF86Config`. Veel distributies leveren aparte configuratiegereedschappen voor X mee, maar Slackware Linux heeft alleen de standaardgereedschappen van XFree86. In de meeste gevallen kan X vrijwel automatisch geconfigureerd worden, soms is het nodig handmatig onderdelen in te stellen via `/etc/X11/XF86Config`.

Automatische configuratie

De XFree86 server heeft een optie om automatisch een configuratiebestand aan te maken. XFree86 zal alle modules voor grafische kaarten laden, proberen de hardware te detecteren en vervolgens een configuratiebestand te schrijven. Om XFree86 een configuratiebestand te laten schrijven voer je als root uit:

```
$ XFree86 -configure
```

Als X verder geen foutmeldingen geeft kan de gegenereerde configuratie naar de `/etc/X11/` directory gekopieerd worden en kun je X starten om te proberen of de configuratie werkt:

```
$ cp /root/XF86Config /etc/X11/  
$ startx
```

Interactieve configuratie

XFree86 bevat twee tools voor interactieve configuratie van X, **xf86cfg** en **xf86config**. **xf86cfg** zal proberen je videokaart automatisch te detecteren en start daarna de X omgeving met een configuratieprogramma waarmee je de gedetecteerde configuratie aan kan passen. Soms kan het gebeuren dat **xf86cfg** in een videomodus overschakelt die niet door je monitor ondersteund wordt. In zo'n geval kan **xf86cfg** ook in tekstmodus gebruikt worden door het te starten met **xf86cfg -textmode**.

xf86config werkt anders dan de hiervoor besproken programma's, het detecteert namelijk geen hardware en zal daarom om alle details van je hardware vragen. Als je nog geen ervaring hebt met het configureren van XFree86 is deze methode af te raden.

Window manager

De "look and feel" van XFree86 wordt bepaald door een zogenaamde window manager. Slackware bevat bijvoorbeeld de volgende veelgebruikte window managers:

- WindowMaker: Een relatief lichte WindowManager. Onderdeel van het GNUStep project.
- BlackBox: Lichte WindowManager, is van weinig bibliotheken afhankelijk.

- KDE: Een volledige desktopomgeving, inclusief browser, e-mail programma en een office suite (KOffice).
- GNOME: Ook een volledige desktopomgeving. Over het algemeen iets minder doorontwikkeld dan KDE, maar heeft toch een grote schare aanhangers.

Als je gewend bent een desktopomgeving te gebruiken, omdat je bijvoorbeeld van een andere distributie of een ander besturingssysteem overstapt naar Slackware zul je je waarschijnlijk het beste thuisvoelen in KDE of GNOME. Toch is het aan te raden één van de lichtere windowmanagers te proberen. Ze zijn veel sneller, nemen minder geheugen in en de applicaties uit een desktop omgeving (bijvoorbeeld Konqueror of KMail) zijn uitstekend te gebruiken onder een andere window manager.

Onder Slackware kunt je het volgende commando uitvoeren om een window manager te selecteren:

```
$ xwmconfig
```

Dit configuratieprogramma zal de geïnstalleerde window managers tonen, waar ja vervolgens een keuze uit kunt maken. Door **xwmconfig** als root uit te voeren kun je de standaard windowmanager voor alle gebruikers selecteren.

Hoofdstuk 14. Vi

Opmerking vooraf

Als je op slackware vi op wilt starten met het commando **vi**, dan wordt de editor elvis opgestart. Hou daar dus rekening mee. Er bestaat in de directory /usr/bin een symbolic link naar elvis. Dit kun je zien door het commando:

```
user@slackware: ls -l /usr/bin/vi
```

te geven. Om met dit commando vi op te starten, moet je dit dus even aanpassen.

Overigens, alhoewel in dit hoofdstuk over vi gesproken wordt, wordt eigenlijk vi improved (VIM) besproken. Niet alle zaken die hier genoemd worden gelden namelijk voor vi.

Wat is Vi

Vi (de afkorting VI staat voor Visual Interface) kun je beschouwen als een editor. Je kunt er tekst mee bewerken, maar ook bijvoorbeeld sources voor je programma's, bijvoorbeeld C programma's. Vi kent een aantal ingebouwde commando's die je het leven een stuk eenvoudiger maken. Een nadeel, voor zover je dat een nadeel kunt noemen, is dat je kennis moet hebben van een aantal commando's. In dit hoofdstuk vind je die commando's terug waarmee je de meeste taken kunt uitvoeren.

Met vi kun je echter niet zomaar tekst invoeren. Vi kent namelijk twee verschillende modi, te weten een invoer modus (lees : insert) en een command modus. De eerst modus verklaart zich zelf, echter de tweede verdient een nadere uitleg.

Met betrekking tot de command modus zijn er een aantal verschillende soorten commando's te onderkennen. Er zijn commando's die voorafgegaan moeten worden door het / teken, commando's die voorafgegaan moeten worden door het : teken en commando's die zonder een teken voor het commando gebruikt kunnen worden. In de volgende paragrafen worden de verschillende modi en de daarbij horende zaken behandeld.

De eerste stap

Vi kun je starten door vanaf de command line het commando vi in te geven:

```
user@slackware: vi
```

Na het invoeren van het voorgaande commando wordt VI gestart. Je kunt echter niet meteen beginnen met het invoeren van tekst aangezien je nu in de command modus zit. Om tekst in te voeren moet je aan VI aangeven dat je tekst wilt invoegen (insert). Je kunt nu op de insert toets drukken op je toetsenbord of je kunt de letter i indrukken op je toetsenbord. Hierna verschijnt links onderin het woord INSERT en kun je beginnen met het invoeren van tekst.

Om de tekst te bewaren moet je het write commando gebruiken. Bedenk wel dat als je vi op deze manier hebt opgestart, je nog geen bestandsnaam hebt opgegeven. Dit zul je dus moeten doen bij het bewaren (write) van je tekst. Om je tekst te bewaren moet je eerst weer terug naar de command modus. Dit doe je door op de Escape knop van je toetsenbord te drukken. Daarna voer je als volgt het write commando uit:

```
:w text.txt
```

De : betekent dat je dit teken op je toetsenbord moet ingeven. Hierna toets je de letter w om aan te geven dat je de tekst wilt bewaren (write) en hierna geef je de naam op van het bestand waarin je de tekst wilt bewaren.

Je kunt ook op het moment dat je vi opstart opgeven welk bestand je wilt bewerken of, als je een nieuw bestand maakt, de naam van dit nieuwe bestand meegeven als zogenaamde parameter. Vi zal, als het bestand bestaat, deze openen waarna je, natuurlijk rekening houdend met insert mode e.d., het bestand kunt bewerken. Het openen van een dergelijk bestand doe je zo:

```
user@slackware:vi test.txt
```

Een andere mogelijkheid is het opstarten van vi waarna je opgeeft, in de command mode van vi, welk bestand je wilt bewerken (of lezen voor mijn part). Om dit te doen geef je binnen vi het volgende commando:

```
:e test.txt
```

Kopiëren, verwijderen, knippen en plakken

Het meest essentiële weet je nu : hoe een stuk tekst te maken en op te slaan en hoe je een bestand moet maken c.q. openen. Er zijn natuurlijk nog veel meer dingen die je wilt doen, bijvoorbeeld een stuk tekst kopiëren of verwijderen. Voordat ik je iets vertel over kopiëren, moet je het volgende weten. Let wel : het stukje over buffers is eigenlijk bedoelt voor iemand die al weet hoe je moet kopiëren en dergelijke. Laat je dus hier niet door van de wijs brengen en sla dit deel desnoods over.

Als je tekst kopieert, dan wordt deze tekst tijdelijk ergens opgeslagen. Dit ergens wordt een buffer genoemd, een stukje geheugen. Trouwens, ook tijdens het schrijven van tekst wordt deze tekst ook in een buffer opgeslagen totdat je het geschrevene opslaat op bijvoorbeeld je schijf en je de editor verlaat. Vi kent 27 buffers, waarvan er eentje de zogenaamde onbenoemde buffer is oftewel de buffer die gebruikt wordt als jij niet opgeeft welke buffer er gebruikt moet worden. Je kunt zelf opgeven, als je dit wilt, naar welke buffer iets geplaatst moet worden. Laat je hierdoor niet van de wijs brengen; ook als je dit niet doet dan is er niks aan de hand. Vi gebruikt dan gewoon de "standaard" buffer.

Zoals ik al zei, kent vi 27 buffers. De eerste is de "default" buffer en die gebruik je als je verder niks opgeeft. Wil je nu een andere buffer gebruiken dan kun je dit doen middels de letters van het alfabet. Dus de letter "a" staat voor buffer a enzovoorts. Als je dus iets in buffer a wilt plaatsen, bijvoorbeeld iets wat je wilt kopiëren, dan doe je dit als volgt:

```
"a2yy
```

Bovenstaand voorbeeld houdt in dat je twee regels naar de buffer a kopieert. Let op het teken ". Dit betekent dat de inhoud van buffer a overschreven mag worden. Als je in plaats van het overschrijven van

de buffer een aantal regels aan het eind van de buffer wilt plaatsen, dan moet je de hoofdletter A gebruiken in plaats van de kleine letter a, dus zo:

```
"A2yy
```

Overigens : deze commando's zijn natuurlijk niet zichtbaar als je ze gebruikt. Let dus goed op wat je doet!

Goed, back to basic. Stel, je wilt een aantal regels kopiëren. De tekst die je hebt ziet er zo uit:

```
Dit is de eerste regel
Dit is een andere regel
Tja, weer een andere regel, t wordt een beetje saai zo
```

Als je nu de eerste regel wilt kopiëren en na de laatste regel wilt plaatsen, ga dan met je cursor op de eerste regel staan op de letter D van Dit en druk twee keer op de toets van je toetsenbord waar het teken y op staat. Deze y staat in dit geval voor yank, oftewel kopieer. Het commando wat je dus geeft is : yy

Ga nu naar de laatste regel, dus ga op de laatste regels staan en paste de tekst met het commando p. Bedenk hierbij dat dit commando's zijn die je in de command modus uit moet voeren waarbij geen : of / teken gebruikt hoeft te worden. Bedenk ook dat het pasten van tekst gebeurt na de regel waarop je staat. Voordat je tekst paste moet je natuurlijk wel de command modus actief hebben, denk daarom!

Als het goed is zal de tekst er nu als volgt uitzien:

```
Dit is de eerste regel
Dit is een andere regel
Tja, weer een andere regel, t wordt een beetje saai zo
Dit is de eerste regel
```

Het deleten van tekst doe je eenvoudig met met het dd commando. Ook hier weer geldt dat je meerdere regels kunt verwijderen door een getal, bijvoorbeeld 3 om drie regels te verwijderen, op te geven voor het dd commando.

Knippen en plakken van tekst werkt op dezelfde manier als kopiëren en plakken met de uitzondering van het gebruikte knip commando. Het klinkt misschien gek, maar om te knippen kun je het delete commando, dd, gebruiken. Wat er eigenlijk gebeurt bij het deleten van tekst is dat deze in een buffer opgeslagen wordt, immers je kunt verwijderen ook weer ongedaan maken dus ergens moet deze tekst nog aanwezig zijn. Dus een regel, of meerdere regels, kun je met het paste (p) commando op een andere plaats in de tekst zetten. Knippen en plakken zagezegd.

In deze paragraaf zijn de verschillende commando's behandeld om hele regels te knippen kopiëren en dergelijke. Je kunt ook bijvoorbeeld een woord kopiëren en knippen. In plaats van de dubbele aanduiding yy of dd plaats je op de eerste positie van het commando de letter w van woord (eigenlijk word want het is natuurlijk engels) dus :

```
yw
```

betekent het kopiëren van een woord en

```
dw
```

betekent dat je een woord wilt verwijderen.

Tekst zoeken en vervangen

Voor het zoeken van tekst gebruik je het `/` commando. Stel je wilt het woord `slackware` in een bestand zoeken dat ziet het zoek commando er als volgt uit:

```
/slackware
```

Bedenk dat vi zoekt vanaf de plaats waar je cursor in de tekst staat. Wil je vanaf het begin van de tekst zoeken dat moet je zorgen dat je cursor aan het begin van je tekst staat. Je kunt ook opgeven in welke richting er gezocht moet worden. Als je vanaf de cursor positie tot het begin van je tekst wilt zoeken dan gebruik je het vraagteken (`?`):

```
?slackware
```

Dus met `/` zoek je vanaf de cursor positie naar het einde van de tekst en met `?` zoek je vanaf de cursor positie naar het begin (backwards) van de tekst. Als je hetgeen gevonden hebt waar je naar zocht en je wilt kijken of de string waar je naar zocht vaker voorkomt, dan kun je dit doen met het `n` (next) commando. Met het `N` commando kun je een volgend voorkomen van de zoek string in de tegengestelde richting zoeken. Stel, je zoekt op `slackware` met het `?` commando, dus terug in de tekst. Als je deze string hebt gevonden en je wilt nu verder zoeken maar dan naar het einde van de tekst, dan gebruik je het `N` (dus hoofdletter `n`) commando.

Er kunt tekst op verschillende manieren veranderen. Stel je wilt een woord vervangen. Dan kun je twee keer de insert toets op je toetsenbord indrukken waarna je tekst kunt overtypen. Een andere manier om een woord te vervangen in de zin waar de cursor zich bevindt is het het volgende commando:

```
:s/suse/slackware
```

Om alle instanties van een woord in de zin waar de cursor zich bevindt te vervangen kun je het volgende commando gebruiken:

```
:s/suse/slackware/g
```

Bovenstaand commando verandert de string `suse` in `slackware` in de zin waar de cursor staat. Als je nu alle instanties in de tekst van `suse` in `slackware` wilt veranderen dan kun je dit op de volgende manier doen:

```
:%s/suse/slackware
```

Vi configureren

Het is heel goed mogelijk om vi aan je smaak aan te passen. Zo kun je tijdens een vi sessie met het `set` commando een aantal opties aan- of uitzetten. Helaas gelden deze aanpassingen alleen voor je huidige sessie, dus waarschijnlijk zul je je eigen aanpassingen een wat permanenter karakter willen geven. Om dit te doen moet je zorgen dat er een `.vimrc` (let op de punt voor de bestandsnaam) bestand in je home directory aanwezig is. Als deze nog niet bestaat dan moet je deze even creëren.

Je kunt erg veel opties zelf bepalen met behulp van dit configuratie bestand. Zo kun je bijvoorbeeld bepalen of je syntax highlighting aan wilt hebben tijdens het bewerken van een source programma, je kunt op deze manier macro's gebruiken om je vi leven iets eenvoudiger te maken.

Hieronder volgen een aantal opties die je in je .vimrc bestand kan zetten.

```
number : hiermee kun je regel nummering aanzetten
showmatch : hiermee worden overeenkomstige haken getoond, dus { en }
backup : hiermee wordt een kopie bijgehouden als je een tekst aan het bewerken bent
syntax on : syntax highlighting aan zetten
```

Alle opties die je in het .vimrc bestand plaatst, moet je als volgt noteren:

```
set number
set syntax on
```

Zie voor opties en dergelijke het voorbeeld vimrc bestand wat met jou kopie van vi is meegeleverd.

Overzicht vi commando's

dd	:	delete c.q. knip
yy	:	kopieer
:q	:	quit, verlaat vi
:q!	:	verlaat vi zonder de veranderingen door te voeren
p	:	paste (plakken)
u	:	maak iets ongedaan
:w	:	schrijf, bewaar
:wq	:	bewaar en einde vi
x	:	verwijderen van 1 karakter
/	:	zoek naar een string vanaf cursor positie naar einde van het bestand
?	:	zoek naar een string van cursor positie naar het begin van het bestand
n	:	herhaal actie (next)
:set	:	een optie "zetten" voor vi
:set all:	:	laat alle mogelijke parameters zien die je in kunt stellen

Aanhangsel A. GNU Free Documentation License

Copyright (C) 2000,2001,2002 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondly, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the

copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

- A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.
- B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.

- C. State on the Title page the name of the publisher of the Modified Version, as the publisher.
- D. Preserve all the copyright notices of the Document.
- E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.
- F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.
- G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.
- H. Include an unaltered copy of this License.
- I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.
- J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.
- K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.
- L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.
- M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.
- N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.
- O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties--for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements".

COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See <http://www.gnu.org/copyleft/>.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

Copyright (c) YEAR YOUR NAME. Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the

Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the "with...Texts." line with this:

with the Invariant Sections being LIST THEIR TITLES, with the Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.

Aanhangsel B. Dankbetuigingen

We willen graag in dit aanhangsel een aantal personen en organisaties bedanken voor het tot stand komen van dit boek:

Personen

- Chemtux, voor de vele correcties, commentaar en suggesties.
- Patrick Volkerding, zonder Patrick had Slackware Linux nooit bestaan.

Organisaties

- Berlios, voor het hosten van de SlackFiles CVS tree, waar Het Slackware Handboek een onderdeel van is.
- NL.Linux.org, voor de mirror van Het Slackware Handboek.