



# **FLOSS**

## MANUALS

# Table of Contents

<b>Linux Commando's.....</b>	<b>1</b>
<b>Linux Commando's / Opdrachten.....</b>	<b>2</b>
man.....	2
ls.....	3
cd.....	3
mkdir.....	4
pwd.....	4
mv.....	4
rm.....	5
locate / slocate.....	5
cp.....	5
ping.....	6
tab.....	6
<b>Andere Commando's.....</b>	<b>7</b>
more.....	7
<b>Linux Bestands Structuur.....</b>	<b>8</b>
<b>Tekst Bewerkers.....</b>	<b>9</b>
less.....	9
vim.....	10
nano.....	12
<b>Software installeren van de bron.....</b>	<b>13</b>
tar command.....	13
./configure.....	13
make.....	13
Dependencies /benodigdheden.....	14
<b>Verbinden met het Internet.....</b>	<b>15</b>
<b>Afkijktijstje.....</b>	<b>16</b>
<b>Credits.....</b>	<b>18</b>
Licence.....	18
Authors.....	18

# Linux Commando's

Je kunt tegenwoordig, met een linux versie als Ubuntu, zo'n beetje alles doen wat je wilt, zonder de terminal te gebruiken.

Als je echter snel wilt werken, configuraties wil veranderen, of software wilt installeren, is de commando lijn de meest efficiënte manier. Veel mensen zijn wat bang van de command line, omdat ze gewend zijn aan een grafische gebruikers interface, maar met wat oefening zal het snel makkelijker worden.

Het enige wat je echt in gedachten moet houden, is dat je het langzaam aandoet, en oefent wat je geleerd hebt. Probeer niet alles tegelijkertijd te onthouden, gebruik wat je weet, en leer meer als het nodig is...

# Linux Commando's / Opdrachten

De command line is de meest krachtige manier om met Linux te werken, maar als je er niet aan gewend bent, kan het moeilijk zijn. De beste manier om de commando's te leren, is om te beginnen met wat basis commando's te werken. Probeer niet meteen alles via je Command Line te doen. Leer een paar commando's, gebruik ze, en voeg er langzaam een paar toe. Zo kan je langzaam je vocabulaire van commando's uitbreiden. Hieronder staan wat basis commando's waarmee je kunt beginnen.

Probeer ze niet tegelijkertijd te leren. Kies er een paar, en oefen ze.

- man
- ls
- cd
- mkdir
- mv
- rm
- locate / slocate
- ping
- cp
- pwd
- tab

En nog een paar die goed zouden zijn om te kennen:

- ldconfig
- updatedb
- ./configure
- make
- make install
- tar
- more
- whereis

Laten we ze allemaal eens afzonderlijk te bekijken. Voel je vrij om ermee te experimenteren, maar pas op dat je geen bestanden of mappen aantast. Ik zal het hieronder aangeven als het een commando is waarmee je problemen kunt veroorzaken.

## man

Dit is een goed commando om ee te beginnen, omdat het de ingebouwde hulp-pagina's van Linux aanroept. man is kort voor manual of handleiding. En als je dit intypt, gevolgd door een spatie, en dan de naam van een ander commando, krijg je de hulp pagina in je terminal, met een beschrijving van het commando. Probeer eens:

```
man ls
```

Je zult een terminal venster krijgen vol informatie over het ls commando. De opzet van de helppagina kan wat verwarrend zijn, dus browse wat rond. Het gedeelte wat het meest interssant is, is de beschrijving van het commando (oftewel: wat het doet). Druk, om naar beneden te scrollen, op de handleidingpagina, en om hem af te sluiten kun je op de q toets drukken.

Probeer nog eens wat man commando's en lees dan verder.

Er is nog een hulp-systeem dat hetzelfde werkt, maar in plaats van man type je info in:

```
info ls
```

Experimenteer!

## ls

Het ls commando is het 'list/lijst' commando. Je kunt dit gebruiken om de inhoud te laten zien van de map waarin je bent. Type dit commando eens in in je terminal venster, en kijk wat je krijgt.

Nu, een mogelijkheid van Linux commando's is dat je verschillende parameters eraan toe kan voegen. Dit is niet zo moeilijk, en het verfijnd de manier waarop je een commando gebruikt. Over het algemeen worden deze parameters toegevoegd door een - direct na het commando te typen, en dan de parameter namen of afkortingen. Bijvoorbeeld als ik het volgende type:

```
ls -l
```

Dan voeg ik de l parameter toe aan het ls commando. De l parameter is kort voor 'long list/lange lijst' en refereert naar de manier waarop de ls informatie getoond kan worden. Deze manier geeft meer informatie dan het ls commando alleen. Probeer en vergelijk ze eens.

De parameters voor alle commando's zijn te vinden in de man pagina's van ieder commando.

Voor het ls commando zijn de volgende handig: ls zelf, ls - al en ls - l.

## cd

**cd** is het meest gebruikte commando om door het bestandssysteem op je computer heen te navigeren. cd staat voor 'change directory' of 'verander map'. Probeer het uit, door ls in te typen om een lijst te krijgen van alle bestanden en folders in de map waar je nu in zit. Type daarna eens cd gevolgd door de naam van een van de bestanden in de lijst. Als er bijvoorbeeld een bestand genaamd 'me.txt' in de lijst staat, kan ik het volgende typen:

```
cd me.txt
```

Nu krijg ik een foutmelding! Waarom? Omdat je niet naar een map kan gaan als het een bestand is. Met cd kan je verder geen schade aanrichten. Om van map te veranderen type je cd in, gevolgd door de naam van de map waar je heen wilt gaan. Als er een map genaamd src in de lijst stond, zouden we:

```
cd src
```

intypen. Als dat lukt, zal de terminal geen foutmelding geven. Probeer het eens met een echte map op je computer. Als het niet lukt is dat omdat je of geen toestemming hebt om de map te bekijken, omdat je de mapnaam verkeerd gespeld hebt, of omdat de map niet bestaat.

Nu even wat informatie over het Linux bestands-systeem. Over het algemeen, als je het systeem netjes hebt opgezet, werk je in je home map. Deze staat vaak op dezelfde plek in Linux. Om je home directory te vinden type je

```
cd /
```

in. Hierdoor kom je in de allerhoogste map van je computers bestandssysteem.

Als je nu ls intikt, zal je de lijst van mappen zien, die op het hoogste niveau van je computer staan.

Daar staan belangrijke mappen tussen, maar je wilt je richten op de map met de naam home. Om naar deze map te gaan kunnen we het cd commando gebruiken:

```
cd home
```

Als je nu ls intikt, zal je een lijst zien van meer mappen, en hopelijk heeft een ervan de je gebruikersnaam als naam. Dit is jouw home map. Nu zijn we hier aangekomen door relatieve positionering te gebruiken Dat betekent: Als ik in de hoogste map ben, en ik type cd home in, kom ik in de home map, waar alle home mappen van de afzonderlijke gebruikers in staan. Als ik in een andere map was en cd home had ingetikt, zou ik een fout krijgen. Je kunt, als je wilt ook absolute paden naar de map die je wilt bereiken gebruiken. Als ik bijvoorbeeld ergens in een vage hoek van mijn bestandssysteem ben, en ik snel naar de home map wil komen type ik:

```
cd /home
```

Als ik naar de map wil gaan die onder de home map zit, (bijvoorbeeld als ik daar een map genaamd adam heb) Type ik:

```
cd /home/adam
```

## mkdir

Dit is het commando dat je gebruikt om een map aan te maken. Het staat voor 'make directory/maak map'. Om het te gebruiken type je de naam van de map die je wilt maken na het mkdir commando:

```
mkdir bleep
```

In dit geval krijg je een map in de map waarin je bent met de naam bleep. Als er al een map is met deze naam, krijg je een foutmelding, en zal de computer niet de bestaande map overschrijven. Probeer eens wat mappen aan te maken.

## pwd

Als je de weg kwijt raakt in je bestandssysteem, kan je als pwd intikken, en het vertelt je waar je bent. Dit commando geeft je het locatie pad, of een absoluut pad naar waar je bent. Als ik bijvoorbeeld in mijn adam map ben, in mijn home map, zal ik als ik pwd intik het dit terugkrijgen:

```
/home/adam
```

Experimenteer met het veranderen van mappen met cd en dan pwd in te tikken om te kijken waar je bent.

## mv

Dit commando is kort voor move/verplaats. Met mv kan je bestanden verplaatsen op je bestandssysteem. Het is vergelijkbaar met de knip en plak acties die je bij Mac en Windows hebt. Om v te gebruiken moet je eerst het commando intikken, gevolgd door het bestand dat je wilt verplaatsen (in absolute of relatieve paden, inclusief de bestandsnaam) en dan de plek waar je hem heen wilt verplaatsen (in absolute of relatieve paden). Als ik bijvoorbeeld het bestand 'me.txt' wil verplaatsen van mijn huidige naar de /usr/bin map. Zou ik het volgende intikken:

```
mv me.txt /usr/bin
```

Let op: Ik hoef niet de bestandsnaam in te tikken in de naam van het pad waar ik het bestand heen wil verplaatsen, behalve als ik de naam ervan wil veranderen. Als ik b.v. de naam van 'me.txt' naar 'you.txt' wil

veranderen, terwijl ik het bestand verplaats zou ik tikken:

```
mv me.txt /usr/bin/you.txt
```

Als ik alleen de naam van het bestand zou willen veranderen, maar niet een nieuwe plek geven, kan ik het mv commando gebruiken:

```
mv me.txt you.txt
```

Let op dat als je mv gebruikt, dat je het bestand verplaatst en niet kopieert. Het origineel zal daarom verplaatst worden en niet meer bestaan in de map waar je hem vandaan hebt gehaald. Je moet met deze naam wel oppassen, omdat je per ongeluk bestanden kunt overschrijven. Als je b.v. een bestand verplaatst naar een map waar al een bestand met dezelfde naam staat, wordt het oude bestand overgeschreven, en is die weg. Pas daarom op.

## rm

Een commando waarbij je heel erg op moet letten is het rm commando. Het staat voor remove/verwijder, en je gebruikt het als je een bestand of map wilt verwijderen. Type daarvoor rm gevolgd door de naam van het bestand dat je kwijt wilt. Om een map te verwijderen, kan je hetzelfde commando gebruiken, maar met de -r parameter:

```
rm -R mapnaam
```

Mapnaam in dit geval is natuurlijk de naam van de map die je weg wilt gooien. Je kunt hier ook rmdir voor gebruiken (kort voor remove directory/verwijder map). Let ontzettend goed op als je dit commando gebruikt, omdat je je besturingssysteem ermee kunt wissen.

## locate / slocate

Deze commando's helpen je om bestanden te vinden in je bestandssysteem. De locatie van alle bestanden op je computer staan in een database, die af en toe geupdate wordt met het updatedb commando. Om een bestand te vinden kan je locate of slocate intikken (hangt af welke op je computer geïnstalleerd is), gevolgd door een deel van de naam van het bestand of de map waarin je kijkt. Als je b.v. het 'icecast.conf' bestand zoekt, type je:

```
slocate icecast.conf
```

Als er geen reactie volgt, betekent dat dat het bestand niet bestaat op je computer, of dat het bestaat, maar dat de database niet weet waar het is. In het laatste geval type ik updatedb, en probeer het daarna weer.

Met locate en slocate kan je niets kapot maken, je kunt er dus zoveel mee spelen als je wilt. Soms kan updatedb even duren, als je het niet recent gebruikt hebt, of als je een langzame computer hebt. Het kan een hoop werkgeheugen gebruiken op langzame computers, dus het is slim het niet te gebruiken als je met iets heel belangrijks bezig bent.

Je kan ook met whereis en find experimenteren om bestanden te vinden.

## cp

Cp staat voor copy/kopieren. Je kunt het hetzelfde gebruiken als mv. Het enige verschil is dat het een versie van het bestand achterlaat op de originele plek.

## ping

Ping wordt normaliter niet toegevoegd aan de 10 belangrijkste commando's, maar het is handig als je wilt weten waar je online bent. Ping stuurt een verzoek naar een computer op het net, als die computer het verzoek krijgt, zal hij antwoorden. Type ping in, gevolgd door een URL die je kent. bijvoorbeeld:

```
ping www.cnn.com
```

Als die computer het verzoek krijgt, krijg je informatie terug via de terminal. Dit zal blijven scrollen totdat je het stopt door control en c tegelijkertijd in te tikken.

Als je geen reactie krijgt op de ping, ben je waarschijnlijk offline. Het kan echter ook zijn dat de machine niet op pings reageert vanwege veiligheids en andere redenen. zorg daarom dat je zeker weet dat de machine waarnaar je pingt reageert op ping verzoeken

En niet alle internetverbindingen laten ping-verkeer toe, als je dus op niks een ping terugkrijgt, kan dat ook aan je internetverbinding/provider liggen.

## tab

**tab** is niet echt een commando, maar meer een 'toets' .. Ieder toetsenbord heeft een tabtoets, en het is erg handig om te hebben in Linux. Je kan deze eerder gebruikt hebben om je tekst meer naar rechts te zetten in een tekstverwerker, dit kan in Linux tekstverwerkers nogsteeds. Maar als je de tab gebruikt in de Linux terminal kan je er veel tijd mee winnen.

Wat tab doet is eigenlijk je zin automatisch afmaken. Als ik b.v. het bestand 'dlfjasalfiasjdfslsjafta.txt' ergens wil verplaatsen met het mv commando, kan ik of alle letters in gaan tikken, of ik tik mv (voor verplaats), gevolgd door de eerste paar letters van de bestandsnaam, en druk dan op tab. De rest van de bestandsnaam wordt dan automatisch toegevoegd. Als hij niet wordt aangevuld, betekent het dat er meerdere bestanden of mappen zijn die met die eerste letters beginnen. Je kunt dan meer letters intikken en weer op tab klikken, of je kunt twee keer op tab toetsen, om een lijst te krijgen van de bestanden die met die letters beginnen.



# Andere Commando's

Aan het begin van dit onderdeel gaf ik een lijstje met andere die ook handig zijn om te weten. Ze waren:

- ldconfig
- updatedb
- ./configure
- make
- make install
- tar
- more
- whereis

Twee ervan (whereis en updatedb) heb ik al genoemd. De andere kunnen handig zijn om software te installeren.

## more

more is handig als je een enorme lange output wilt kunnen bekijken. Als ik in een map ben met 1000 bestanden, en ik type ls in, past de output niet in mijn kleine terminal scherm, zodat het sneller langscrollt dan ik kan lezen. Om het langzamer te krijgen, kan ik more gebruiken:

```
ls | more
```

Ik krijg dan 1 pagina output per keer, en door op de spatiebalk te drukken ga ik naar de volgende pagina. Als je op q drukt, wordt more afgesloten.

Dat rare verticale lijntje in het bovenstaande commando is het pipe commando.

Met **Pipe** kan je verschillende commando's samen gebruiken om de output die je krijgt te bepalen. Vaak wordt het gebruikt om een commando te verfijnen. Als je deze dus goed leert kennen kan je heel efficient worden.

# Linux Bestands Structuur

Als je de terminal opent en het volgende typt (gevolgd door enter):

```
cd /
```

Wordt je in de bovenstemap van het Linux bestandssysteem geplaatst. Als je dan typt:

```
ls -al
```

Zie je zoiets:

```
sh-2.05b$ ls -al
total 80
drwxr-xr-x 19 root root 4096 Oct  9 13:57 .
drwxr-xr-x 19 root root 4096 Oct  9 13:57 ..
drwxr-xr-x  2 root root 4096 Oct  5 09:31 bin
drwxr-xr-x  3 root root 4096 Oct  9 21:47 boot
drwxr-xr-x  1 root root    0 Jan  1  1970 dev
drwxr-xr-x 71 root root 4096 Oct 15 11:35 etc
drwxr-xr-x  4 root root 4096 Oct  9 19:21 home
drwxr-xr-x  8 root root 4096 Sep 18 23:29 lib
drwx----- 2 root root 16384 Sep 18 20:06 lost+found
drwxr-xr-x  9 root root 4096 Oct  9 16:36 mnt
drwxr-xr-x  9 root root 4096 Oct  8 23:20 opt
dr-xr-xr-x 64 root root    0 Oct 15 11:35 proc
drwx----- 75 root root 8192 Oct 15 12:35 root
drwxr-xr-x  2 root root 4096 Sep 23 18:58 sbin
drwxr-xr-x  9 root root    0 Oct 15 11:35 sys
drwxrwxrwt 60 root root 4096 Oct 15 12:36 tmp
drwxr-xr-x 17 root root 4096 Oct  5 09:31 usr
drwxr-xr-x 15 root root 4096 Oct  9 13:57 var
```

De bovenste lijst is een behoorlijk standaard mapstructuur voor Linux. Elke naam aan de rechterkant staat voor een map, en elke map heeft bestanden en andere mappen die specifiek zijn voor deze map. De **lib** map, bijvoorbeeld, bevat code-bibliotheken die je software gebruikt. Nu hoef je alleen maar de **home** map te kennen. Deze map heeft verschillende mappen voor alle gebruikers van de computer. Als je inlogt als 'lotte' bijvoorbeeld, zal je ingelogd worden in de map met je usernaam (in dit geval lotte) als naam in de home map.

Het belangrijke ding waar je op moet letten is deze Linux map structuur. Het andere belangrijke is dat Linux vooral bestaat uit tekst bestanden. Je kunt daardoor bijna ieder deel van Linux veranderen (hoe het eruit ziet, of werkt) door alleen het tekst bestand aan te passen. In Windows en op de Mac doe je dit soort aanpassingen vaak via kleine programmaatjes. In Windows, bijvoorbeeld, als je de resolutie van je scherm wilt aanpassen, gebruik je het 'schermvoorkeuren' controlepaneel in je controlepanelen map. In Linux kan je dit doen door een tekstbestand aan te passen.

Dit heeft wat voordelen - een ervan is dat je een stuk meer controle krijgt. Maar het heeft ook nadelen - het kan moeilijk zijn om te leren welke bestanden je kan aanpassen, en wat je moet veranderen. Om de overgang van andere besturingssystemen naar Linux makkelijker te maken zijn er soms configuratie softwares voor Linux op je computer geïnstalleerd die een GUI (Grafische gebruikers interface) gebruiken die lijkt op die van Mac en Windows. Dit is echter niet altijd het geval, dus uiteindelijk zou je, als je dit wilt, het zelf moeten aanleren om aan te passen met een tekst-bewerker

# Tekst Bewerkers

Als je niet weet hoe je een tekstbewerker in Linux moet gebruiken, kom je niet erg ver. README en INSTALL bestanden lezen is de basis als je Linux op de commandline wilt leren.

We zullen eens kijken naar de tekstbewerkers nano en vim. En ook naar less, dit is geen tekstbewerkers, maar een commando waarmee je bestanden op je systeem kunt lezen.

## less

Laten we met less beginnen. Dit commando opent alleen tekstbestanden om te lezen. Als de map waarin je nu bent een bestand heeft met de naam "README" probeer dan dit commando eens:

```
less README
```

je zult dan zoiets zien in je terminal:

Je kunt je pijltjes gebruiken om te scrollen, en afsluiten met de q (van quit).

Less is de makkelijkste manier om bestanden te lezen, maar het is niet altijd geïnstalleerd op alle Linuxen. Probeer daarom eerst less, en als het niet werkt, probeer dan een van de anderen hieronder. Met less kan je dus alleen bestanden lezen, om ze te bewerken moet je een textbewerker hebben. Nano en Viim worden vaak gebruikt door programmeurs die aan code werken.

Check eerst of deze geïnstalleerd zijn door de naam in de terminal in te tikken. Je zult waarschijnlijk wel vim hebben. Als je er geeneen hebt, kan je ze installeren door de bron pakketten te downloaden, of via je pakjes-beheer-systeem.

Je kunt, als je niks wilt installeren ook de namen van andere tekst bewerkers invoeren:

- vi
- gvim
- emacs
- pico

- soffice
- fte
- vfte
- gedit
- glimmer

De makkelijkste om te gebruiken is soffice (ook bekend als openoffice,) en gedit.

## **vim**

Laten we eens naar vim kijken. Als je vim intikt in de terminal krijg je zoiets:

(Als je toevallig vi geïnstalleerd hebt, zie je ongeveer het zelfde ding)

Als je een bestand wilt openen met vim, is het het beste om de naam van het bestand na het vim commando te zetten, zodat vim opent met het bestand al open. Als je bijvoorbeeld het "README" bestand wilt lezen (in de map waarin je nu bent). type dan:

```
vim README
```

Vim zal openen met het readme bestand:

Scroll nu door het bestand heen met de pijltjestoetsen. Type om vim af te sluiten:

```
:q
```

Als je een bestand wilt bewerken in vim, moet je het eerst openen, en daarna i typen:

```
i
```

Now, I am imagining vim is quite a bit different to any text editor you have used before, so perhaps some explanation is needed. vim opens a file initially as a read only file. This means that when you first open the file with vim you are not allowed to change the file. vim has then a whole world of commands you can use to work on the file and most of the commands are executed by just typing a single letter, or they are in the format:

```
: command
```

command is hier de naam van het commando dat je wilt gebruiken. Hieronder staat een lijstje die handig zijn.

<b>commando</b>	<b>actie</b>
i (alleen gebruikt in alleen-lezen modus)	tekst invoegen
:w (alleen gebruikt in alleen-lezen modus)	veranderingen opslaan
u (alleen gebruikt in alleen-lezen modus)	veranderingen ongedaan maken
:q (alleen gebruikt in alleen-lezen modus)	vim afsluiten

Daarnaast kan je door 'esc' in te toetsen teruggaan naar de alleen-lezen modus. Je moet zelfs eerst esc klikken voordat je een commando uitvoert in vim. Als ik bijvoorbeeld een bestand wil openen, en daarna wat tekst aanpassen doe ik het volgende: Ik typ

```
vim README
```

in de terminal. Dit opent het bestand.

Dan ga ik met mijn pijltjestoetsen naar de plek waar ik tekst wil invoegen en ik druk:

```
i
```

Nu ben ik in de invoegmodus, en alles wat ik type komt in het bestand terecht. Als ik klaar ben, druk ik op 'esc' en om de veranderingen op te slaan :

```
vim
```

:w

Dit slaat het bestand op met de nieuwe veranderingen. Daarna kan ik vim afsluiten, door op 'esc' te drukken, gevolgd door:

:q

## nano

**nano** is echt een lichtgewicht bewerk, en het is zonde dat het niet vaker gebruikt wordt. Start nano met het bestand dat je wilt lezen door te typen:

```
nano README
```

Het programma zal in de terminal openen:

Als je bestanden op een server wilt aanpassen, scroll dan door het bestand met de pijltjestoetsen.

Om tekst te bewerken kan je gelijk beginnen te typen in het terminal venster.

Opslaan: ctrl-o

Afsluiten: ctrl-x

Als je afsluit zonder eerst op te slaan, vraagt nano je of je je veranderingen wilt opslaan. Druk dan op y als je dat wilt.

# Software installeren van de bron

Software installeren op Linux is een breed onderwerp, omdat iedere Linux versie zijn eigen pakjes-beheer-systeem heeft. Maar alle Linux versies hebben gemeen dat je ook software via de broncode kunt installeren. Het is dan wel handig als je wat van Linux commando's en het Linux bestandssysteem weet. (Als je dat nog niet weet, lees dan eerst even die hoofdstukken)

## tar command

Als je een bron pakket hebt:

```
tar zxvf packagename.tar.gz
```

Waar de 'packagename' in het voorbeeld hierboven de eigenlijke naam is van het pakket dat je wilt installeren. Als het tar commando wordt gevolgd door de zxvf parameters wordt het tar.gz bestand uitgepakt, en wordt er een nieuwe map gemaakt met alle uitgepakte bronnen. Nu moet je je werkmap veranderen naar deze map via het cd commando. Meestal is de nieuwe mapnaam de naam van het pakket zonder het .tar.gz gedeelte. Als mijn pakje bijvoorbeeld "packagename.tar.gz" heet, zou ik na het tar zxvf commando een map hebben met de naam 'packagename'. en dan zou ik cd packagename intikken om deze nieuwe map in te gaan.

Als je niet zeker bent van de naam, type dan ls.

## ./configure

Als je eenmaal in de nieuwe map bent aangekomen, kunnen we het installatieproces starten. Type daarvoor:

```
./configure
```

Dit is niet echt een commando. Ieder installatiepakket heeft een script met de naam 'configure;. Door er een stip en een schuine streep voor te zetten vertel je linux dat hij dat script moet uitvoeren. Het configuratie script doet dan zijn ding, kijkt wat voor machine je hebt en wat je al geïnstalleerd hebt, welke Linux versie je hebt etc. etc.

Het meest voorkomende probleem dat je op dit moment kunt krijgen is dat het configure script stopt, en je vertelt dat de software-bibliotheek die de nieuwe software moet gebruiken mist. Dit kan erg vervelend zijn, daarom zijn er pakjes-beheer-systemen uitgevonden. Als je deze foutmelding krijgt, moet je een zoekmachine gebruiken om erachter te komen welke software er bedoeld wordt door de foutmelding, en het vinden. Begin dan het installatieproces opnieuw met dit nieuwe pakket. (Dit kan dagen duren)

## make

Laten we ervanuit gaan dat je geen foutmeldingen kreeg bij het configure script.. Typ dan het volgende commando in:

```
make
```

Dit commando maakt / compileert de software voor je. Je krijgt dan een heleboel gecompileerde bestanden die samen je software vormen. Het make proces kan even duren, afhankelijk van de snelheid van je computer, en de grootte van de pakjesbronnen die je installeert.

Als make stopt, type dan het volgende:

```
make install
```

dit zal de nieuw gemaakte software installeren op de goede plek op je computer. Je hoeft daarna alleen de naam van je programma in te tikken in je terminal venster, en het zou op moeten starten. Als het niet lukt, en je krijgt een foutmelding. Probeer dan eens ldconfig in te tikken, en probeer het daarna opnieuw. Ldconfig update het systeem zodat je besturingssysteem weet dat er nieuwe bestanden zijn etc.

## Dependencies /benodigdheden

Een ander geliefd onderwerp is 'benodigdheden'. Linux ontwikkelaars schrijven niet vaak een programma vanuit het niets, ze gebruiken vaak werk dat al door andere programmeurs is gemaakt. Dit scheelt tijd, en een heleboel aardige programmeurs hebben al een hoop code-bibliotheken gemaakt die andere programmeurs kunnen gebruiken. Deze bibliotheken staan op een vaste plaats in je Linux besturingssysteem, vaak in de /lib map.

Als je nu een programma installeert dat verschillende bibliotheken nodig heeft om te werken, en je hebt deze bibliotheken nog niet, zal je eerst deze bibliotheken moeten hebben. Als het een aardige programmeur is, zal hij informatie over de benodigdheden in het "README" of de "INSTALL" bestand gezet hebben. Deze vind je in de bron-map van het programma.

Luie Linux gebruikers hebben vaak geen zin om deze bestanden te lezen, en gaan gewoon door het proces heen, totdat ze een foutmelding krijgen, waarna ze de benodigdheden zoeken en installeren.

Als je echter nieuw bent, is het slim om de readme en install bestanden wel te lezen. Het zal tijd en moeite schelen. Installeer de benodigde pakketten gewoon een voor een (op de manier die hierboven beschreven staat), tot je alles hebt om je favoriete programma te installeren en gebruiken.



# Verbinden met het Internet

Als je je linux computer opstart, en je een browser hebt geopend, en je kunt niet online, kan je een probleem hebben. Het makkelijkste om dit op te lossen is door een locale linux nerd aan te spreken, omdat er veel dingen zijn die het probleem kunnen veroorzaken. Je kunt echter wel wat doen voordat je de linux nerd gaat zoeken. Probeer:

```
dhcpcd eth0
```

of dit:

```
dhclient eth0
```

Deze commando's proberen om een verbinding te maken met het internet, via je ethernet kaart. Je ethernetkaart wordt vaak aangesproken met een alias, in dit geval (en heel vaak) is het alias eth0. Als je meer dan een manier hebt om met het internet te verbinden, zullen deze opeenvolgend genummerd zijn. Mijn draadloze verbinder heet b.v. eth1. Het commando vraagt om een IPadres voor je computer. De bovenstaande twee commando's kunnen een van de drie volgende resultaten hebben:

- Het zorgt ervoor dat je terminal er voor een lange tijd mee kapt, zonder reactie. Om een of andere reden ga je geen IPadres krijgen. Vraag daarom je systeem beheerder voor hulp. De meest voorkomende reden is dat je een 'dedicated' IP adres nodig hebt.
- Je krijgt een foutmelding dat het een onbekend commando is.

Je hebt de dhcpcd of dhclient commando's niet geïnstalleerd, of je hebt de privileges niet die je nodig hebt om het commando uit te voeren. In dit geval moet je de linux nerd spreken, of je systeembeheerder

- Je krijgt een IPadres (helaas krijg je geen feedback in text, je moet daarom je browser opnieuw proberen om te kijken of het gewerkt heeft.

Als je erg ambitieus wordt, kan je ook over het ifconfig commando leren. Het is een belangrijk commando voor het netwerken met Linux, maar het kan zijn dat je meer over Linux wilt weten, voordat je aan deze begint. Er is een handig help-bestand over dit commando, die je kunt bekijken door

```
man ifconfig
```

te tikken in de terminal. Je kan ook gewoon dit tikken

```
ifconfig
```

of

```
ifconfig -a
```

om informatie te krijgen over hoe je machine momenteel geconfigureerd is. Let erop dat online komen ingewikkeld kan zijn, vraag daarom iemand om hulp, of probeer online hulp te krijgen via een andere computer en gebruik zoekmachines om meer over Linux en het internet te weten te komen.

# Afkijktijstje

Soms zijn er gewoon handige dingen die je moet weten zonder alles van Linux te kennen:

## welke versie van Linux gebruik ik?

type dit in de terminal:

```
dmesg | grep Linux
```

## hoe stel ik een statisch IP in voor mijn ethernet kaart?

kijk naar de handleidingpagina's voor ifconfig. Maar je kunt deze syntax volgen als je de goede nummers weet.

```
ifconfig eth0 IP1 netmask IP2  
route add default gw IP3
```

Vervang de waardes IP1, IP2 en IP3 hierboven. IP1 is je statische IP adres, IP2 het netmask (vaak 255.255.255.0) en IP3 is het gateway IP adres.

## welke versie van deze software gebruik ik?

soms kan je een -v of --version na de programma-naam intikken in de terminal. Als ik bv wil weten welke versie van MuSE ik heb, type ik:

```
muse -v
```

## met welke gebruikersnaam ben ik ingelogd?

typ:

```
whoami
```

## welk commando gebruik ik om bestands-vergunningen aan te passen?

*chmod...* lees erover in de rest van de handleiding

## welk commando gebruik ik om mijn externe harde schijven aan te sluiten?

*mount...* zoek het op

## hoe zet ik mijn computer uit:

via een van deze twee mogelijkheden

```
halt
```

```
shutdown now
```

## mijn computer is in Linux opgestart, maar ik zie alleen een enorm terminal venster, en het ziet er nogal eng uit?

Je bent nog niet ingelogd.. log in met je gebruikersnaam en wachtwoord. Als je inlogt en geen mooie interface ziet, type dan:

```
startx
```

of

```
gdm
```

**wat moet ik doen als ik per ongeluk een belangrijk bestand verwijderd heb met 'rm'?**

beloof tegen jezelf dat je dat nooit meer doet, sla alles op, en log nooit in als root.

**als ik ingelogd ben onder een gebruikersnaam, maar ik wil iets doen als de root gebruiker, hoe kan ik dat dan doen?**

gebruik de commando's su of sudo (zie handleiding)

**hoe laat is het?**

type date in de command line

**welke maand is het?**

type cal in de terminal.

**welke modules zijn geladen?**

type lsmod

**welke systeemplicessen, software etc draaien nu op mijn computer?**

probeer het ps commando

**ik denk dat iemand mijn login wachtwoord weet. hoe verander ik het?**

probeer dit commando ( of zoek het op in de man paginas)

passwd

# Credits

## Licence

All chapters copyright of the authors (see below). Unless otherwise stated all chapters in this manual licensed with **GNU Free Documentation License 1.2**

```
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 here:
http://www.gnu.org/licenses/fdl.txt
```

## Authors

People that have contributed to this manual include (please add your name if you have made a contribution) :

### *CHEAT SHEET*

© adam hyde 2006, 2007

Modifications:

Lottes Meijer 2007

---

### *BASIC COMMANDS*

© adam hyde 2006, 2007

Modifications:

Lottes Meijer 2007

---

### *CONNECTING*

© adam hyde 2006, 2007

Modifications:

Lottes Meijer 2007

---

### *FILE STRUCTURE*

© adam hyde 2006, 2007

Modifications:

Lottes Meijer 2007

---

### *INSTALLING SOFTWARE*

© adam hyde 2006, 2007

Modifications:

Lottes Meijer 2007

---

### *TEXT EDITING*

© adam hyde 2006, 2007

Modifications:

Lottes Meijer 2007

---

### *CREDITS*

© adam hyde 2007

---

### *INTRODUCTION*

© adam hyde 2007

Modifications:  
Lottes Meijer 2007

---

Free manuals for free software