

# Basiese Linux

Bouwe van der Eems

Uitgawe 1, 2005

**Titel:** Basiese Linux

**Outeur:** Bouwe van der Eems

**Uitgewer:** Bouwe van der Eems

**Epos:** bouwe.vandereems@gmail.com

**Uitgawe:** 1 – Januarie 2006

**ISBN:** 0-620-35976-5

**Sinopsis:** *'n Inleiding tot die basiese operasies van die Linux bedryfstelsel. Die fokus is om mense te help om net aan die gang te kom en hulle in staat te stel om hulleself verder te help. Slegs die tekskoppelvlak word behandel, aangesien dit dieselfde is vir al die verspeidings van Linux, en aangesien dit die koppelvlak is wat vir mense die moeilikste is. Die boek is gerig op skoolkinders.*

**Synopsis:** *An introduction to the basic operations of the Linux operating system. The focus is to help people to get them started and help themselves to continue further. Only the text user interface is discussed, because this is the same for all linux distributions and because this is the most difficult interface. The book is aimed at school children.*



# Inhoudsopgawe

1.	Inleiding .....	5
1.1.	Wat is Linux .....	5
1.2.	Waarom 'n boek oor Linux?.....	5
1.3.	Waarom is Linux belangrik? .....	5
1.4.	Wat is 'n bedryfstelsel? .....	6
1.5.	Installering van Linux .....	8
2.	Aantekeninge op Linux .....	9
2.1.	Hoe om aan te teken.....	9
2.2.	Die skep van gebruikers .....	10
2.3.	Die gebruik van tolkprogramme .....	11
2.4.	Die struktuur van instruksies.....	12
2.5.	Hulp oor instruksies .....	12
2.6.	Kombinasie van instruksies.....	13
2.7.	Skedulering van instruksies.....	13
3.	Montering van aandrywers.....	15
3.1.	Monteer 'n CD ROM .....	15
3.2.	Ontheg 'n CD ROM.....	15
3.3.	Monteer 'n Stiffie.....	16
3.4.	Ontheg 'n Stiffie .....	16
4.	Werk met lêers.....	18
4.1.	Leërgidse in Linux .....	18
4.2.	Manupilasie van lêers.....	19
4.3.	Regte van lêers en gidse .....	21
4.4.	Vind lêers .....	22
4.5.	Sorteer lêers .....	23

4.6.	Kompaktering van lêers .....	23
5.	Redigering en uitdruk van teksleërs .....	26
5.1.	Die vi redigeerder .....	26
5.2.	Die emacs redigeerder.....	27
5.3.	Die pico redigeerder.....	28
5.4.	Gebruik die drukker .....	28
6.	Hoe om 'n Skrip te skryf.....	30
7.	Prosesse in Linux.....	32
8.	Linux en Netwerke .....	35
8.1.	Kommunikasie tussen gebruikers .....	35
8.2.	Elektroniese pos.....	36
8.3.	'n Linux sessie oor die netwerk .....	36
8.4.	Die oordra van lêers na 'n Linux masjien .....	37
8.5.	Lees die Web vanaf Linux.....	38
8.6.	Loop 'n webwerf op die Linux bediener.....	39
9.	Programmering in Linux.....	41
9.1.	Inleiding.....	41
9.2.	Die gebruik van gcc .....	42
9.3.	Outomatisering van gcc.....	43

## 1. Inleiding

### 1.1. *Wat is Linux*

Linux is 'n bedryfstelsel. 'n Bedryfstelsel is 'n program wat die rekenaar in staat stel om te kommunikeer met al sy bybehore soos die hardeskyf, die sleutelbord en die muis. 'n Bedryfstelsel op sy eie is egter van weinig nut vir gebruikers. 'n Bedryfstelsel word meestal verskaf met 'n program wat die bedryfstelsel kan installeer en 'n verskeidenheid nuttige programme. 'n Bedryfstelsel saam met al hierdie programme word 'n verspreiding genoem. Daar is 'n groot aantal verspreidings beskikbaar op die mark soos Knoppix, Mandrake, Redhat, Suse ens.

Die eerste Linux kern is geskryf deur Linus Torvald en vir die eerste keer vrygestel op die Internet in 1991. Linux is 'n oopbron weergawe van die kommersiële bedryfstelsel genaamd UNIX.

Oopbron sagteware is programme wat deur 'n groep programmeerders oor die wêreld geskryf is, en kan gratis (meestal deur dit af te laai van die Internet) of teen 'n lae koste (deur die CD's te koop) verkry word. Gebruikers mag die sagteware gratis gebruik, modifiseer, kopieer en versprei.

### 1.2. *Waarom 'n boek oor Linux?*

Dit raak toenemend moeilik raak vir Afrikaanse jongmense om werk te kry in die formele ekonomie. Die enigste manier wat jongmense gaan werk kry, is wanneer hulle vaardighede beskikbaar wat baie skaars is.

Dit is my opinie dat Linux 'n tegnologie is wat gesog gaan word in die toekoms, vir die volgende redes:

- a) Linux is 'n baie goedkoper bedryfstelsel as die bedryfstelsels wat algemeen kommersiël beskikbaar is. In die tye van moontlike resessie waarin ons nou verkeer, gaan hierdie kostefaktor al hoe belangriker word.
- b) Die swak punte van Linux word toenemend aangespreek, wat die kloof tussen die kommersiële bedryfstelsels en Linux kleiner maak.

Om te verseker dat die mense wat hierdie boek lees die inhoud hulle eie maak, word dit in die moedertaal van die teikengroep geskryf, naamlik Afrikaans.

### 1.3. *Waarom is Linux belangrik?*

Om iets op 'n rekenaar te kan doen, moet jy 'n bedryfstelsel hê. 'n Bedryfstelsel stel mens in staat om met die rekenaar te kommunikeer, instruksies vir die rekenaar te gee en om terugvoer van die rekenaar te ontvang.

Daar is heelwat bedryfstelsels beskikbaar op die mark soos Microsoft Windows, IBM OS/2, Unix, Linux of MacOS. By verre die mees gewilde bedryfstelsel is Microsoft Windows. Indien Windows so gewild is, waarom moet mens dan kyk na Linux. Om die vraag te kan beantwoord, moet mens verstaan waarom Windows so gewild is.

## Basiese Linux

Die redes waarom Windows so gewild is, is die volgende:

- a) Windows is baie maklik om te gebruik, omdat dit 'n grafiese gebruikerskoppelvlak (Graphical User Interface – GUI) het. Aan die ander kant het Linux 'n teks gebruikerskoppelvlak.
- b) Windows is maklik om te installeer, aangesien dit outomaties agterkom watter hardeware daar in die rekenaar is, en dat outomatiese die regte drywers daarvoor installeer. (Dit word die “plug and play” fasiliteit genoem). Linux egter is heelwat moeiliker om te installeer.
- c) Daar is baie programme beskikbaar wat op Windows loop. Die belangrikste program wat deur baie mense gebruik word, is die Microsoft Office pakket. Aan die ander kant is daar heelwat minder kommersiële programme wat op Linux loop.

Dinge is egter besig om te verander. Saam met Linux kry mens deesdae ook 'n grafiese gebruikerskoppelvlak wat bekendstaan as X-Windows. Selfs meer as een. Verder is Linux heelwat makliker om te installeer, indien pakette soos Redhat, SUSE Linux of Madrake Linux gebruik word. Verder is daar nou ook 'n Office pakket beskikbaar wat op Linux loop, naamlik OpenOffice.

Die grootste rede waarom mense egter Linux oorweeg is nie omdat dit alternatiewe het vir die sterk punte van Windows nie, maar omdat dit heelwat goedkoper is. Pakette soos Redhat en SUSE kos maar 'n fraksie van die prys van die Windows bedryfstelsel, en die program OpenOffice kan mens gratis van die Internet aflaai.

### **1.4. Wat is 'n bedryfstelsel?**

'n Rekenaar bestaan uit komponente soos 'n mikroprosessor, geheue, hardeskywe, 'n muis, 'n sleutelbord en 'n skerm. Hierdie komponente noem ons die hardeware van die rekenaar. Indien mens al die hardeware inmekaar sou sit, en die rekenaar aanskakel, sou die rekenaar niks doen nie. Die mikroprosessor sou geen instruksies uitvoer nie en dit sou nie eens weet van die ander komponente wat aan die rekenaar gekoppel is nie.

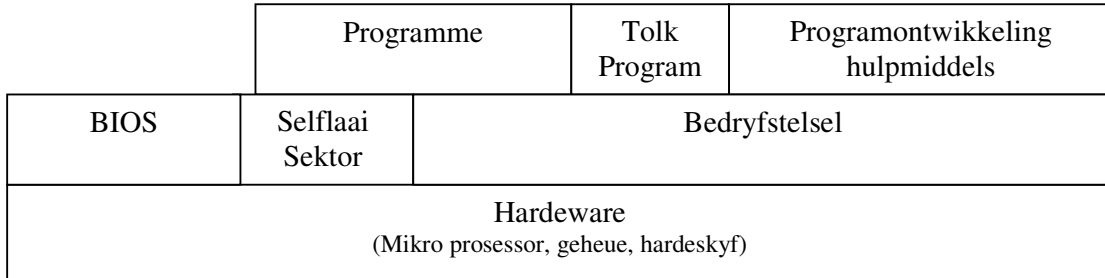
Indien mens iets met die rekenaar wil doen, moet dit in staat wees om instruksies te kan ontvang en vir jou kan sê wat die terugvoer is nadat die instruksies klaar uitgevoer is. 'n Program wat in staat is om instruksies van die gebruiker te ontvang, die instruksies uit te voer, en terugvoer aan die gebruiker te gee, word die bedryfstelsel van die rekenaar genoem. Voor mens kan begin om met 'n rekenaar te werk, moet jy daarom eers die bedryfstelsel laai.

Die bedryfstelsel weet watter hardeware komponente daar in die rekenaar is, en hoe om met hierdie komponente en met die gebruiker te kommunikeer. Indien die gebruiker instruksies aan die bedryfstelsel gee, sal die bedryfstelsel die hardeware komponente gebruik om die instruksies uit te voer. Dit sal byvoorbeeld inligting van die hardeskyf lees en dit wat hy gelees het op die skerm vertoon.

Wanneer die bedryfstelsel loop, wag dit nie slegs op instruksies nie, maar is dit konstant besig met die bestuur van die rekenaar se hardeware. Dit bestuur die geheue en maak seker dat elke program net die geheue gebruik wat aan hom toegeken is. Dit bestuur die mikroprosessor en verseker dat elke program 'n toegekende deel van die mikroprosessor se verwerkingstyd kry.

## Basiese Linux

Vir die bedryfstelsel om te weet watter hardeware komponente in die rekenaar is, moet iets dit vir hom sê. Aangesien dit omslagtig vir die gebruiker is om al daardie besonderhede te moet invoer elke keer as die rekenaar aangeskakel word, word inligting oor die hardeware in die rekenaar self gestoor. Dit word gestoor in 'n klein stukkie permanente geheue wat die BIOS (Basic InputOutput System) genoem word gestoor. Wanneer die rekenaar aangeskakel word, lees die bedryfstelsel die gegewens oor die hardeware vanaf die BIOS. Indien daar verander word aan die hardeware, moet dit in die BIOS verander word.



Die BIOS stoor ook die plek op die hardeskyf waar die begin van die bedryfstelsel gestoor is. Dit is die selflaai sektor. Wanneer die rekenaar aangeskakel word, lees die rekenaar van die BIOS waar die selflaai sektor is, gaan lees die hardeskyf vanaf die selflaai sektor. Die selflaai sektor stoor die plek op die hardeskyf waarvan die bedryfstelsel gelaai moet word. Indien daar meer as een bedryfstelsel op 'n hardeskyf is, moet die selflaai sektor geprogrammeer word om te weet waar die verskillende bedryfstelsels is. Wanneer die rekenaar aangeskakel word, moet die selflaai sektor die gebruiker eers vra watter bedryfstelsel gelaai moet word, en dan sal dit aan die rekenaar die instruksie gee waarvandaan die bedryfstelsel gelaai moet word.

**NOTA :** Om die opstelling van die selflaaisektor te verander word die program `/sbin/lilo` gebruik. Die konfigurasie vir die selflaaisektor in die lêer `/etc/lilo.conf`.

In die selflaaisektor word die posisie van die bedryfstelsel gestoor. Indien daar meer as een bedryfstelsel op dieselfde masjien is, sal die rekenaar vra watter een gelaai moet word tydens die uitvoering van die selflaaisektor.

As die bedryfstelsel laai, en dit klaar in geheue gelaai is, word die bedryfstelselprogram geloop. Terwyl die Linux bedryfstelsel laai, word 'n aantal boodskappe op die skerm vertoon. Hierdie boodskappe sê watter gedeeltes van die bedryfstelsel gelaai word, en of dit suksesvol was. Hierdie boodskappe is meestal te vinnig om te lees. Om hierdie boodskappe te vertoon kan die instruksie `dmesg` gegee word. Die uitset van hierdie instruksie kan na 'n lêer geskryf word, waar dit rustig bestudeer kan word. Wanneer die bedryfstelsel loop, kan die gebruiker deur middel van die gebruikerskoppelvlak van die bedryfstelsel dinge op die rekenaar doen.

Wanneer mens programme op die rekenaar loop, beheer die bedryfstelsel die loop van die program. Dit laai die program in 'n spesifieke plek in die geheue en gee aan die mikroprosessor opdrag om die instruksies in daardie stuk geheue uit te voer.

Die manier waarop hardeware, BIOS, die bedryfstelsel en programme bymekaar inpas word geïllustreer in die skets hierbo.

In hierdie handeiding word instruksies verduidelik wat die bedryfstelsel kan uitvoer soos `useradd`, `mount`, `ls`, `rm` ens. Saam met die bedryfstel kom ook 'n groot aantal

nutsprogramme wat nodig is om die bedrystelsel te konfigureer soos **vi**, **emacs** en **pico**. Aangesien hierdie nutsprogramme onmisbaar is in die konfigurasie van Linux, word dit ook in hierdie handleiding bespreek.

### **1.5.     *Installering van Linux***

Met pakette soos Redhat en SUSE is dit glad nie meer moeilik om Linux te installeer nie. Mens selflaai maar op die CD en volg die instruksies. Om die rede gaan hierdie handleiding nie in op die installering van Linux nie.

Die belangrikste is dat mens tydens die installering van Linux die aantekennaam en aantekenwagwoord moet gee. Hierdie aantekengegewens is belangrik om Linux te gebruik wanneer dit geïnstalleer is. Sonder hierdie gegewens kan mens glad nie aanteken op Linux.



## 2. Aanteken op Linux

Afhangende van die tipe Linux sal dit selflaai met die teks gebruikerskoppelvlak of die grafiese gebruikerskoppelvlak. Indien dit selflaai met die grafiese gebruikerskoppelvlak, moet mens van daaruit die opsie kies om na die teks gebruikerskoppelvlak te gaan.

### 2.1. Hoe om aan te teken

Linux beskou hierdie verskillende gebruikerskoppelvlakke as verkillende toestande waarin Linux verkeer. Die volgende toestande word onderskei:

- Loopvlak 0 (Runlevel 0) : Rekenaar kan afgeskakel word
- Loopvlak 1 (Runlevel 1) : Enkelgebruiker modus
- Loopvlak 2 (Runlevel 2) : Multigebruiker sonder netwerk
- Loopvlak 3 (Runlevel 3) : Multigebruiker met netwerk
- Loopvlak 4 (Runlevel 4) : Dieselfde as 5.
- Loopvlak 5 (Runlevel 5) : Multigebruiker met netwerk en grafiese gebruikerskoppelvlak
- Loopvlak 6 (Runlevel 6) : Linux besig om af te sluit

**NOTA** : Om in te stel op watter loopvlak Linux moet begin, redigeer die lêer **/etc/inittab**.

Wanneer Loopvlak 3 bereik word, kan jy aanteken op die teks gebruikerskoppelvlak. Linux sal die volgende op die skerm vertoon:

```
Welcome to SuSE Linux 7.1 (i386) - Kernel 2.2.18 (0).  
linux login:
```

By die login en password velde moet die aantekennaam en wagwoord ingesleutel word.

Login : **“Tik aantekennaam in”**

Password : **“Tik wagwoord in”**

Wanneer jy suksesvol aangeteken het, sal Linux ‘n por gee, wat dikwels die woord “Linux” en die aantekennaam sal bevat.

**NOTA** : Linux sal ook ‘n verwelkomingsboodskap gee. Om hierdie verwelkomingsboodskap te verander, redigeer die lêer **/etc/motd**.

Indien mens as ‘n gewone gebruiker aanteken, het jy beperkte regte. Om sekere instruksies te gee, sal mens moet aanteken as ‘n gebruiker met meer regte. Dan moet jy aanteken as ‘n

## Basiese Linux

gebruiker met die aantekennaam van “root”. Die wagwoord van hierdie gebruiker word ook opgestel tydens die installasie. Om as “root” aan te teken, moet jy eers afteken as die gebruiker en dan weer aanteken as die “root” gebruiker.

Om af te teken, gebruik die instruksie : **logout**

Indien mens iets wil doen wat meer regte vereis, sonder om as gebruiker af te teken, kan jy aanteken as ‘n supergebruiker deur die instruksie **su** te gee. Hierna sal daar gevra word vir ‘n wagwoord. Nadat jy die korrekte wagwoord verstrek het, is jy aangeteken as supergebruiker. Indien jy jou take as supergebruiker afgehandel het, kan jy terug gaan na die sessie van die gewone gebruik deur die instruksie **exit** te gee.

Indien mens die rekenaar wil afskakel, moet jy eers die bedryfstelsel afsluit. Wanneer Linux klaar afgesluit is, en dit veilig is om die rekenaar af te sluit, is dit op loopvlak 0.

Om Linux af te sluit, moet jy aanteken as ‘n supergebruiker, aangesien ‘n gewone gebruiker nie regte het om Linux af te sluit nie. Wanneer jy aangeteken is as supergebruiker, gee die instruksie **halt** of **shutdown -h now**. Wanneer Linux aandui dat Loopvlak 0 bereik is, en die boodskap “*Runlevel 0 has been reached*” vertoon, is dit veilig om die rekenaar af te skakel.

Indien jy wil hê dat die rekenaar weer moet begin selflaai, nadat Linux afgesluit het, gebruik die instruksie **shutdown -r now**.

Om vanuit die teks gebruikerskoppelvlak die grafiese gebruikerskoppelvlak te aktiveer, gebruik die instruksie **startx**.

**NOTA** : Die loopvlak waarmee die masjien moet selflaai kan opgestel word deur die lêer **/etc/inittab** te redigeer.

## 2.2. Die skep van gebruikers

Wanneer die bedryfstelsel geïnstalleer word, kan mens gebruikers skep. Daarna kan nog gebruikers geskep word deur middel van linux instruksies. Die gebruik van hierdie instruksies word hieronder gegee.

Teken aan as gebruiker “root” en gee die volgende instruksie.

**useradd -d piet**

**passwd piet**

Nadat hierdie instruksie gegee is, sal Linux vra vir ‘n wagwoord en ‘n bevestiging vir hierdie gebruiker. Gee dan die wagwoord vir hierdie nuwe gebruiker en bevestig dit deur dit weer in te tik.

Skep ‘n tuisgids vir hierdie gerbuiker, soos uiteengesit in afdeling 4.1.

Teken af as gebruiker “root”.

Teken aan as gebruiker “root” en gee die volgende instruksie.

**userdel piet**

Skrap die tuisgids vir hierdie gerbuiker, soos uiteengesit in afdeling 4.1.

Teken af as gebruiker “root”.

Wanneer ‘n gebruiker aangeteken het, kan hy sien as wie hy aangeteken is, deur die instruksie **id** te gee. Dit sal inligting al die inligting gee oor die aantekende gebruiker.

Linux is ‘n multigebruiker bedryfstelsel. Meer as een gebruik kan gelyktydig op ‘n Linux stelsel aanteken. Om die sien wie almal aangeteken is, gebruik die instruksie **who**. Dit gee ‘n lys van wie almal aangeteken is op die stelsel.

### 2.3. Die gebruik van tolkprogramme

Indien jy aangeteken is op Linux, word daar instruksies ingetik en die bedryfstelsel stuur dan die resultate na die skerm van die rekenaar. Linux maak gebruik van ‘n tolkprogram (*Shell*) om instruksies van die gebruiker lees, en dan die resultate op die skerm te vertoon. Waar ‘n bedryfstelsel soos DOS net een tolkprogram het, kan mens in Linux kies tussen ‘n verskeidenheid van tolkprogramme.

Die verskillende tolkprogramme vertoon die inligting wat die bedryfstelsel na die skerm stuur op verskillende wyses. Die verskillende tolkprogramme vertoon die por verskillend. Die wyse waarop instruksies geredigeer kan word verskil. As mens bv. die leërs in ‘n gids vertoon, sal die verkillende tolke die leërs op verskillende wyses vertoon. Een tolkprogram sal verkillende soorte leërs en verskillende kleure vertoon, terwyl ander almal in dieselfde kleur sal vertoon.

Wanneer jy aanteken op Linux, sal die bedryfstel gebruik maak van die tolkprogram waarvoor die bedryfstelsel opgestel is om mee te begin. Om uit te vind in watter tolkprogram jy is, tik ‘n ongeldige instruksie in soos **aaa** In die bedryfstelsel se reaksie word dan getoon wat die naam van die tolkprogram is wat tans in gebruik is. Die reaksie kan bv. wees “*bash: aaa: command not found*”. Uit hierdie reaksie kan afgelei word dat die “bash” tolkprogram tans in gebruik is.

Vanuit die tolkprogram waarmee Linux geselflaai het, kan mens ander tolkprogramme aktiveer deur die naam van die tolkprogram in te tik. Voorbeelde van algemene tolkprogramme word hieronder gegee:

**Table 1 : Linux Tolkprogramme**

Naam van Tolkprogram	Besrkywing
sh	<i>Bourne Shell</i>
bash	<i>Bourne Again Shell</i>
csh	<i>C Shell</i>
tsh	<i>C Shell met uitbreidings</i>
ksh	<i>Korn Shell</i>

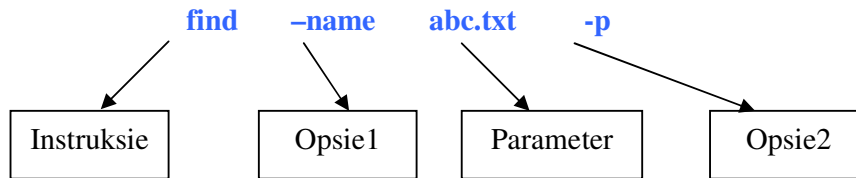
Om uit 'n tolkprogram terug te gaan na die tolkprogram waarme Linux geselflaai het, gee die instruksie **exit**.

**NOTA** : In hierdie boek word altyd gebruik gemaak van die **bash** tolkprogram, tensy anders vermeld.

**NOTA** : Net soos wat Linux meer as een tolkprogram het in die teksgebaseerde gebruikerskoppelvlak, is daar ook 'n verskeidenheid grafiese gebruikerskoppelvlakke. Hierdie grafiese gebruikerskoppelvlakke kan gekies word tydens selflaai. Name van hierdie verkillende koppelvlakke is "kde", "kde1", "gnome", "fvwm2", "windowmaker", "failsafe", ens.

## 2.4. Die struktuur van instruksie

'n Instruksie in Linux bestaan uit verskillende dele soos in die voorbeeld hieronder geïllustreer.



'n Instruksie kan gegee word saam met opsies en parameters.

Die instruksie sê vir die bedryfstelsel wat gedoen moet. In die voorbeeld word 'n instruksie gegee om 'n lêer te soek.

Die opsie gee meer inligting oor hoe die instruksie uitgevoer moet word. In die voorbeeld is daar twee opsies.

Indien daar meer inligting oor die opsie gegee moet word, moet 'n parameter saam met die opsie gegee word. In die voorbeeld word in instruksie gegee om 'n lêer te soek. In die eerste opsie word gespesifiseer dat 'n lêer met 'n spesifieke naam gesoek moet word, en saam met die opsie word die naam van die lêer gegee waarvoor gesoek moet word. Die tweede opsie spesifiseer dat die resultate van die soektog vertoon moet word. Geen parameter word saam met die opsie gegee nie.

Indien 'n instruksie saam met meer as een opsie gegee word, en geen van die opsies benodig parameters nie, kan die opsies saam gegee word. In die voorbeeld, word die instruksie **ls** gegee saam met die opsies **a** en **l**.

**ls -al**

## 2.5. Hulp oor instruksies

Om hulp te kry oor wat 'n instruksie doen, hoe hy gebruik moet word en watter opsies met die instruksie beskikbaar is, kan die instruksie man soos volg gebruik word :

**man halt**

Hierdie instruksie sal meer inligting gee oor die instruksie **halt**.

Man sal 'n beskrywing gee van wat die instruksie doen, asook al die opsies wat saam met die instruksie gebruik kan word.

'n Handleiding oor sekere instruksies of programme kan gelees word deur die instruksie **info** te gebruik. Om te leer hoe om die program **gdb** te gebruik, kan die instruksie **info gdb** gegee word.

Indien jy wil uitvind waar om inligting oor 'n sekere onderwerp te kry, kan die instruksie apropos gebruik word. Om bv. uit te vind waarom meer te leer oor Apache, kan die instruksie **apropos Apache** gebruik word. Dit sal alle verwysing na Apache gee, in die **man** databasis. Om dan meer inligting te kry, kan **man** gebruik word.

### 2.6. *Kombinasie van instruksies*

Die uitset van 'n instruksie kan na 'n leër gestuur word deur van die aanduiding **>** gebruik te maak. So kan die inhoud van 'n gids gestuur word in 'n leër met die naam "toets1" deur die instruksie **ls -l > toets1**.

Die uitset van 'n instruksie kan as die inset ander instruksie gebruik word deur van die pyp karakter gebruik te maak. In die voorbeeld word die uitset van die **ls** instruksie gebruik as inset van die **more** instruksie.

```
ls | more
```

### 2.7. *Skedulering van instruksies*

Die datum en tyd kan gevra word met die instruksie **date**. Indien mens as superbruiker aangeteken is, kan die datum en tyd ook gestel word deur die instruksie **date -s** te gebruik.

Om 'n instruksie skeduleer, kan die **at** instruksie gebruik word. Die **at** instruksie word saam met 'n datum en tyd gegee. Daarna kan 'n aantal instruksies gegee word. Indien al die instruksies gegee is, kan daar aangedui word dat dit die laaste instruksie is deur die **ctrl + D** sleutel te druk.

#### **Voorbeeld**

```
bouwe@linux:~ > at 21:20
warning: commands will be executed using /bin/sh
at> ls -al > toetsn.txt
at> <EOT>
job 7 at 2003-01-14 21:20
```

## Basiese Linux

### Opsomming van instruksies

<b>id</b>	: Wys as wie jy aangeteken het.
<b>logout</b>	: Teken af as gebruiker
<b>su</b>	: Teken aan as supergebruiker
<b>exit</b>	: Teken af as supergebruiker of verlaat tolkprogram
<b>halt</b>	: Sluit Linux af
<b>shutdown -h now</b>	: Sluit Linux af
<b>shutdown -r now</b>	: Sluit Linux af en begin weer met selflaai
<b>useradd</b>	: Skep 'n nuwe gebruiker op die bedryfstelsel
<b>passwd</b>	: Stel die wagwoord vir 'n gebruiker in.
<b>userdel</b>	: Skrap 'n gebruiker op die stelsel.
<b>who</b>	: Lys die gebruikers wat aangeteken is op die stelsel.
<b>man</b>	: Gee inligting oor 'n gegewe instruksie

### Oefeninge

Voer die volgende uit met die rekenaar.

- 1) Skakel die rekenaar aan en laat dit begin met selflaai.
- 2) Indien dit in 'n grafiese gebruikerskoppelvlak begin, kies die opsie om Linux in konsole modus te loop (Loopvlak 3).
- 3) Teken aan in Linux.
- 4) Vind uit in watter tolkprogram Linux geselflaai het deur 'n ongeldige instruksie te gee. Kyk hoe hierdie tolkprogram leërs vertoon deur die instruksie **ls** te gee.
- 5) Aktiveer 'n ander tolkprgram. Kyk hoe hierdie tolkprogram leërs vertoon deur die instruksie **ls** te gee. Verlaat die tolkprogram.
- 6) Teken aan as supergebruiker.
- 7) Teken af as supergebruiker.
- 8) Teken weer aan as supergebruiker.
- 9) Sluit Linux af.
- 10) Skakel die rekenaar af.

### 3. Montering van aandrywers

In Linux het 'n aandrywer nie 'n spesifieke letter wat na hom verwys soos in MSDOS nie, bv c: of d:. In Linux word 'n aandrywer gemonteer, en 'n spesifieke leërgids word opgestel om na die aandrywer te wys.

Wanneer 'n CDROM of stifie gemonteer word, laai Linux die gidsname daarvan in geheue. Wanneer jy die instruksie gee om gidse te vertoon, word hierdie inligting van geheue gelees.

Indien mens 'n ander CDROM of stifie in die aandrywer sit en weer vra om die gidse te vertoon, sal Linux die inligting van geheue lees, en dit gaan lyk of die ou CDROM of stifie nog in die aandrywer is, aangesien Linux nie weet dat 'n ander CDROM of stifie ingesit is nie. Om die rede moet die ou CDROM of stifie eers ontheg word, en nadat die nuwe een ingesit is, moet dit weer gemonteer word.

#### 3.1. *Monteer 'n CD ROM*

In die gebruiker se tuis leërgids kan hy bv. 'n subgids skep met die naam "cdrom". Om die cdrom te installeer om na hierdie subgids te wys, gebruik die volgende prosedure:

Teken aan as supergebruiker

```
mount -t iso9660 /dev/cdrom /home/gebruikernaam/cdrom
```

Teken af as supergebruiker

Wanneer die CD ROM gemonteer is, sal die inhoud daarvan gesien kan word in die gids "/home/gebruikernaam/cdrom".

#### 3.2. *Ontheg 'n CD ROM*

Nadat 'n CD ROM klaar gebruik is, moet dit weer ontheg word. Die bedryfstelsel sal nie toelaat dat die CD ROM verwyder word uit die aandrywer, indien die aandrywer nie ontheg word nie.

Teken aan as supergebruiker

```
umount /home/gebruikernaam/cdrom
```

Teken af as supergebruiker

Om die CD te laat uitskiet, gebruik die instruksie **eject /dev/cdrom**. Hierdie instruksie ontheg die CDROM ook.

Nadat die aandrywer ontheg is, kan 'n nuwe CD ROM in die aandrywer geplaas word. Nadat die aandrywer weer gemonteer is, kan die nuwe CD ROM in die subgids gelees word.

### 3.3. *Monteer 'n Stiffie*

In die gebruiker se tuis leërgids kan hy bv. 'n subgids skep met die naam "floppy". Om die stiffie te installeer om na hierdie subgids te wys, gebruik die volgende prosedure:

Teken aan as supergebruiker

**mount /dev/fd0**

Teken af as supergebruiker

Om hierdie stiffie te formateer vir DOS leërs, gebruik die instruksie:

**fdformat /dev/fd0 mkfs -t vfat -c /dev/fd0**

### 3.4. *Ontheg 'n Stiffie*

Nadat 'n stiffie klaar gebruik is, moet dit weer ontheg word. Indien die stiffie nie ontheg word voordat in nuwe stiffie in die aandrywer geplaas word nie, gaan die bedryfstelsel nie die nuwe stiffie lees nie.

Teken aan as supergebruiker

**umount /home/gebruikernaam/floppy**

Teken af as supergebruiker

Nadat die aandrywer ontheg is, kan 'n nuwe stiffie in die aandrywer geplaas word. Nadat die aandrywer weer gemonteer is, kan die nuwe stiffie in die subgids gelees word.

#### Opsomming van instruksies

- mount** : Monteer 'n aandrywer.
- umount** : Ontheg 'n aandrywer.
- eject** : Skiet die CD uit en ontheg die CDROM.



Oefeninge

Voer die volgende uit met die rekenaar.

- 1) Skep 'n gids wat moet wys na die CDROM deur die instruksie **mkdir cdtoets** te gee.
- 2) Monteer die CDROM en wys dit na die gids **cdtoets**.
- 3) Kyk na die leërs op die CDROM.
- 4) Ontheg die CDROM.
- 5) Skep 'n gids wat moet wys na die stoffie deur die instruksie **mkdir sttoets** te gee.
- 6) Monteer die stoffie.
- 7) Kyk na die leërs op die stoffie.
- 8) Ontheg die stoffie.
- 9) Verwyder die gidse deur die instruksies **rmdir cdtoets** en **rmdir sttoets** te gee.

## 4. Werk met leërs

Bedryfstels maak gebruik van leërs. Inligting word in leërs gestoor. Hierdie inligting kan teks, instruksies of programme wees. Op hulle beurt word leërs wat bymekaar hoort saam geplaas in gidse of subgidse. ‘n Bedryfstelsel stel ‘n mens in staat om leërs te manipuleer, te skep, te redigeer of te laat loop.

Die manipulasie van leërs word in hierdie afdeling bespreek. Die skep en redigeer van leërs word bespreek in die afdeling oor die **vi** redigeerder. Hoe om leërs te skep wat geloop kan word, word bespreek in die afdeling oor skripte en programering.

Die manipulasie van leërs gaan dus daarvan uit dat leërs klaar deur ‘n ander toepassing geskep is. Nadat hulle geskep is, kan hulle deur instruksies gemanipuleer word.

### 4.1. Leërgidse in Linux

Voordat mens sinvol met leërs kan werk, is dit belangrik om ‘n oorsig van die leërgids struktuur te verstaan.

/	bin	(Programme wat instruksies uitvoer) <sup>1</sup>
	etc	(Konfigurasie leërs)
	home /	gebruikersname
	dev	(Leërs vir aandrywers)
	lib	(Stelsel leërs)
	sbin	(Addisionele programme)
	usr	(Stelsel leërs)

Die hoogste vlak leërgids word aangedui met “/” en dit word die hoofgids genoem. Daaronder is ‘n aantal subgidse soos hierbo aangedui. Die tipe leërs wat in daardie subgidse is, word ook aangedui.

Wanneer ‘n gebruiker aanteken, is hy in sy tuisgids. Wanneer ‘n gebruiker geskep word, moet daar vir hom ‘n tuisgids geskep word met ‘n naam wat dieselfde is as sy gebruikersnaam. Hierdie tuisgids is “/home/gebruikersnaam/”. In hierdie tuisgids kan die gebruiker sy eie subgidse skep.

Om te beweeg van een gids na ‘n ander, word die instruksie **cd** (*change directory*) gebruik. Hierdie instruksie kan op verskeie wyses gebruik word.

Table 2 : Gebruike van die **cd** instruksie

Instruksie	Betekenis
<b>cd toets</b>	Gaan na die subgids met die naam “toets”
<b>cd ..</b>	Gaan na ‘n gids een vlak hoër.
<b>cd ~/</b>	Gaan na die tuisgids van die gebruiker
<b>cd /lib</b>	Gaan na die gids met die naam lib wat in die hoofgids is.

<sup>1</sup> Om te sien watter Linux instruksies beskikbaar is, kan in hierdie gids gekyk word.

Dikwels word in die por aangedui in watter leërgids mens is. In sommige tolkprogramme is dit egter nie die geval nie. Om dan uit te vind in watter leërgids jy is, gebruik die instruksie **pwd** om te sien in watter leërgids jy is.

## 4.2. Manupilasie van leërs

Eerstens moet mens in staat wees om te sien watter leërs beskikbaar is. Om die leërs te vertoon wat in die gids is waarin die gebruiker is, word die instruksie **ls** gebruik. Die instruksie kan op verskeie wyses gebruik word deur gebruik te maak van skakels. ‘n Skakel gee meer inligting aan die bedryfstelsel oor hoe ‘n instruksie uitgevoer moet word. Saam met die instruksie **ls**, kan bv. die skakel **-l** gegee word, wat die bedryfstelsel vra om die leërname met al sy besonderhede te vertoon.

**Table 3 : Gebruik van die **ls** instruksie**

Instruksie	Betekenis
<b>ls</b>	Vertoon slegs die leërname
<b>ls -F</b>	Vertoon die leërname, asook die tipe leër.
<b>ls -l</b>	Vertoon die leërname met al hulle besonderhede.
<b>ls -s</b>	Vertoon die leërs met hulle groottes.
<b>ls piet*.*</b>	Vertoon al die leërs met name wat die string “piet” aan die begin het.
<b>ls -a</b>	Vertoon al die leërs sowel as verskuilde leërs.
<b>ls -al</b>	Vertoon al die leërs sowel as verskuilde leërs met al hulle besonderhede.
<b>ls -r</b>	Vertoon die leërs in omgekeerde alfabetiese volgorde

‘n Voorbeeld van die uitset van die instruksie **ls -al** word hieronder gegee.

```

VOORBEELD

bouwe@linux:~ > ls -al

total 16

drwxrwxrwx    2 root    root          4096 Apr  4 08:04 .
drwxr-xr-x    5 root    root          4096 Feb  2 14:02 ..
-rw-r--r--    1 bouwe   users         3600 Apr  4 08:04 toets1.txt
-rw-r--r--    1 bouwe   users         3600 Apr  4 08:04 toets2.txt
    
```

Die eerste kolom vertoon die regte van die leër. Die eerste letter dui aan of dit ‘n gids of ‘n leër is. Indien dit “d” vertoon, is dit ‘n gids. Die eerste groep van drie letters toon die regte van die eienaar, die tweede die regte van die groep en die derde die regte van enigiemand anders. (r = lees, w = skryf, x = uitvoer)

Die derde kolom vertoon die eienaar van die leër.

Die vierde kolom vertoon die groep waaraan die leër behoort.

## Basiese Linux

Die vyfde kolom vertoon die grootte van die lêer in bytes.

Die sesde kolom vertoon die datum en tyd waarop die lêer geskep is.

Die laaste kolom vertoon die naam van die lêer.

Indien mens net lêers met 'n sekere tipe naam wil vertoon, kan 'n troef gebruik word. In Linux word die karakter "\*" as 'n troef gebruik. As mens bv. alle lêers wat met die letter "n" begin wil vertoon, kan die troef soos volg gebruik word : **ls n\***. Indien mens bv. al die lêers met die lêerekstensie ".txt" wil vertoon, kan die troef soos volg gebruik word : **ls \*.txt**.

**NOTA** : Linux het **verskuilde lêers** wat normaalweg nie sigbaar is nie. Hulle word onderskei van ander lêers deurdat hulle lêername begin met 'n punt. Dit is moontlik om self lêers te verander in verskuilde lêers, deur 'n punt aan die lêernaam toe te voeg. Verskuilde lêers kan gesien word deur die instruksie **ls -a** te gebruik.

Daar is verskillende tipes lêers in Linux.

- Lêers wat ASCII<sup>2</sup> teks bevat.
- Lêers wat uitgevoer kan word.
- Gidse, waarbinne daar weer ander lêers gestoor word.
- Gekompakteerde lêers.
- Koppeling na ander lêers.

Om te sien watter tipe lêer 'n lêer is, kan die die instruksie **file** gebruik word. Indien die instruksie **file toets1** gegee word, sal Linux aandui watter tipe lêer toets1 is.

Die volgende instruksies kan gebruik word om lêers te manupileer:

- rm toets** : Skrap die lêer of gids met die naam "toets"
- rm -rf toets** : Skrap die lêer met die naam "toets".
- cp toets1 toets2** : Kopieer die lêer met die naam toets1 na 'n lêer met die "toets2". Die naam kan ook 'n gidsnaam insluit.
- mv toets1 toets2** : Verskuif die lêer met die naam toets1 na 'n lêer met die "toets2". Die naam kan ook 'n gidsnaam insluit.
- cat toets1** : Vertoon die inhoud van die lêer met die naam "toets1".
- more toets1** : Vertoon die inhoud van die lêer met die naam "toets1", een bladsy op 'n slag.

---

<sup>2</sup> ASCII is 'n spesifikasie wat vir alle tekstkaraktters (bv. "a", "b", "c", ..., "A", "B", "C", ..."- " = ", ens.) 'n numeriese waarde toeken. Dit is nodig, aangesien 'n rekenaar slegs numeriese waardes kan stoor.

## Basiese Linux

- head toets1** : Vertoon die eerste 10 lyne van die lêer met die naam “toets1”.
- head -20 toets1** : Vertoon die eerste 20 lyne van die lêer met die naam “toets1”.
- tail toets1** : Vertoon die laaste 10 lyne van die lêer met die naam “toets1”.
- tail -20 toets1** : Vertoon die laaste 20 lyne van die lêer met die naam “toets1”.
- mkdir toets1** : Skep ‘n subgids met die name “toets1”.
- rmdir toets1** : Skrap lêë gids met die naam “toets1”.
- ln -s /bin/ls koppel** : Skep die lêer met die naam “koppel” en maak ‘n simboliese koppeling met die lêer **/bin/ls**, wat ‘n program is wat die inhoud van ‘n gids vertoon.

**NOTA** : ‘n Instruksie more kan in kombinasie met ander instruksies gebruik word. As die inhoud van ‘n gids bv. te veel is om op een skerm te sien, kan die instruksie **ls -l | more** gebruik om die inhoud van ‘n gids een skerm op ‘n slag te sien.

**NOTA** : Wanneer ‘n lêer oorskryf of geskrap word, sal Linux dit doen sonder om ‘n waarskuwing te gee. As mens bv. die instruksie **cp toets1 toets2** gee, terwyl toets2 reeds bestaan, sal toets2 sonder waarskuwing oorskryf word. Om dit te verhoed, kan gebruik gemaak word van die skakel **-i**. As die instruksie **cp -i toets1 toets2** gegee word, sal Linux eers vra of die lêers oorgeskryf moet word.

### 4.3. Regte van lêers en gidse

Net die gebruiker wat ‘n lêer geskep het, het regte om die lêer te lees, te verander of te skrap. Hy is daarom die eienaar van die lêer. Hy kan egter ander gebruikers ook regte gee om iets met die lêer te doen. Hierdie regte kan gegee word aan die eienaar, die groep en enigiemand anders.

Table 4 : Tipes regte

Syfer	Regte
0	Geen
1	Uitvoer alleen
2	Skryf alleen
3	Skryf en Uitvoer
4	Lees alleen
5	Lees en Uitvoer
6	Lees en Skryf
7	Lees, Skryf en Uitvoer

Die instruksie **chmod** verander die regte van 'n leër of 'n gids.

Die instruksie `chmod 660 toets1` gee bv. die volgende regte aan die leër "toets1":

- Eienaar : Lees en Skryf
- Groep : Lees en Skryf
- Enigiemand anders : Geen

Nadat die regte van 'n leër verander is, kan dit gekontroleer word deur die instruksie **ls -l** te gee en na die regte van die leër te kyk.

#### **4.4. Vind leërs**

Daar is baie leërs en gidse op 'n Linux bedryfstelsel, en daarom is dit maklik om te vergeet waar 'n spesifieke leër is. Vir die rede is daar hulpmiddels om leërs te vind. Indien mens leërs soek wat met 'n sekere karakter begin in 'n spesifieke gids, kan daar van 'n troef gebruik gemaak word, soos reeds verduidelik is in afdeling 4.2.

Indien dit onbekend is in watter gids die leër is, kan die **find** instruksie gebruik word. Hierdie instruksie het die volgende elemente:

**find** (die instruksie)

**/** (die gids waar die soektog moet begin. Om by die huidige gids te begin, gebruik "." en om die hele hardeskyf te deursoek, gebruik "/".)

**-name** (dit dui aan dat mens vir 'n spesifieke naam soek)

**toets1** (die naam van die leër wat gesoek moet word)

**-print** (dui aan dat die leërs wat gevind is vertoon moet word.)

Tussen al die elemente moet daar spasies wees.

Om bv. die leër "hosts" onder die gids "/etc" te vind, kan die volgende instruksie gebruik word :

**find /etc -name hosts -print**

Indien die hele leërnaam nie bekend is nie, kan weereens 'n troef gebruik word. Om bv. die leër wat met "ho" begin onder die gids "/etc" te vind, kan die volgende instruksie gebruik word :

**find /etc -name ho\* -print**

Die instruksie sal aanhou soek tot al die leërs gevind is. Om die soektog te stop, druk **ctrl+Z** of **ctrl+C**.

Indien die naam van die leër onbekend is, maar dit bekend is dat daar 'n sekere string in die leër is, kan 'n instruksie gebruik word wat binne in die leërs kyk vir 'n sekere string.

**grep** (die instruksie)

**toets1** (die string wat in die leër wat gesoek moet word)

**\*** (Die leërnaam waarin gesoek moet word. Indien die troef gebruik word, sal in alle leërs gesoek word.)

Om bv. die leër te vind waarin die string “toets1” voorkom, kan die volgende instruksie gebruik:

**grep toets1 \***

Wanneer **grep** in Linux gebruik word, word ‘n groot aantal boodskappe saam met die resultate gegee. Om hierdie boodskappe te onderdruk, kan die **-s** opsie gebruik word. Om te maak dat die soektoeg nie kassensitief is nie, kan die **-i** opsie gebruik word.

#### **4.5. Sorteer leërs**

Linux kan lyste sorteer deur middel van die **sort** instruksie. Die uitset van ‘n instruksie van bv. soos volg alfabeties gesorteer word : **ls | sort**.

Die inhoud van ‘n teksleër kan gesorteer word en na ‘n ander leër gestuur word soos volg: **sort toets1.txt > toets1sorteer.txt**.

#### **4.6. Kompaktering van leërs**

Om spasie op die hardeskyf te spaar, kan leërs gekompakteer word. Hiervoor word die instruksie **compress** gebruik.

Indien die instruksie **compress toets1.txt** gegee word, sal die leër **toets1.txt** gekompakteer word, en ‘n gekompakteerde leër met die naam **toets1.txt.Z** geskep word.

Om te sien hoeveel die leër gekompakteer is, gebruik die opsie **-v** soos volg : **compress -v toets1.txt**.

Om die gekompakteerde leër weer te herstel na sy oorspronklike vorm, kan gewoon die instruksie **uncompress toets1.txt.Z** gegee word.

Op soortgelyke wyse kan die instruksies **gzip** en **gunzip** gebruik word. Die instruksie **gzip** sal ‘n leër skep met die ekstensie van **.gz**.

#### Opsomming van instruksies

<b>cd</b>	: Gaan na ‘n ander gids.
<b>pwd</b>	: Vertoon die naam van die huidige gids.
<b>ls</b>	: Vertoon die leërs in die huidige gids.
<b>file</b>	: Vertoon watter tipe leër ‘n spesifieke leër is.
<b>rm</b>	: Skrap ‘n leër.
<b>cp</b>	: Kopieer ‘n leër.
<b>mv</b>	: Verskuif ‘n leër
<b>cat</b>	: Vertoon die inhoud van ‘n leër.
<b>more</b>	: Vertoon die inhoud van ‘n leër ‘n bladsy op ‘n slag.

## Basiese Linux

<b>head, tail</b>	: Vertoon die eerste / laaste gedeelte van die lêer.
<b>md</b>	: Skep 'n gids.
<b>chmod</b>	: Verander die regte van 'n lêer.
<b>find</b>	: Soek 'n lêer met 'n sekere naam.
<b>grep</b>	: Soek 'n lêer wat 'n sekere string bevat.
<b>sort</b>	: Sorteër 'n lêer
<b>compress</b>	: Kompakteer 'n lêer
<b>uncompress</b>	: Herstel 'n lêer na sy oorspronklike grootte.
<b>gzip</b>	: Kompakteer 'n lêer.
<b>gunzip</b>	: Herstel 'n lêer na sy oorspronklike grootte.

### Oefeninge

Voer die volgende uit met die rekenaar.

- 1) Teken aan in Linux.
- 2) Vertoon die lêers in omgekeerde alfabetiese volgorde met al hulle besonderhede, en stuur die uitset na die lêer met die naam *"toets1"*.
- 3) Sorteër die inhoud van die lêer *"toets1"* en stuur die resultaat na die lêer met die naam *"toetssorteer"*.
- 4) Bepaal watter tipe lêer *"toets1"* is.
- 5) Skep 'n subgids met die naam *"t1"*.
- 6) Kopieer die lêer *"toets1"* na 'n lêer met die naam *"toets2"* in die subgids *"t1"*.
- 7) Gaan na subgids *"t1"*.
- 8) Vertoon die inhoud van die lêer *"toets2"* op die skerm, een skerm op 'n slag.
- 9) Gaan een gidsvlak hoër tot in die tuisgids.
- 10) Skrap die subgids *"t1"* met al sy inhoud.
- 11) Vertoon die eerste 15 reëls van die lêer *"toets1"* op die skerm.
- 12) Kompakteer die lêer *"toets1"* en herstel dit weer na sy oorspronklike vorm.
- 13) Verander die regte van die lêer sodat slegs die eienaar die lêer kan leer en skryf.



## Basiese Linux

- 14) Skrap die leërs “*toets1*” en “*toetsorteer*”.
- 15) Skep ‘n subgid met die naam “*stiffie*”.
- 16) Monteer die stiffie aandrywer en koppel dit aan die subgid “*stiffie*”.
- 17) Vertoon die inhoud van die stiffie, een skerm op ‘n slag.
- 18) Ontheg die stiffie aandrywer.
- 19) Skrap die subgid “*stiffie*”.

## 5. Redigering en uitdruk van teksleêrs

Die konfigurasie van Linux is opgestel in teksleêrs. Indien mens iets daaraan sou wou verander, vereis die redigering van teksleêrs. Aangesien die redigering van teksleêrs so algemeen is, is daar 'n verskeidenheid redigereerders beskikbaar. Die belangrikste redigereerders is **vi**, **emacs** en **pico**.

### 5.1. Die vi redigeerder

Om 'n teksleêr te redigeer met **vi**, tik die instruksie **vi toets1.txt**. Indien die leêr *toets1.txt* bestaan, sal **vi** dit oopmaak vir redigering, en indien dit nog nie bestaan nie, sal **vi** die leêr skep.

Vi is 'n modale redigeerder. Dit is altyd in een van twee modusse. In opdrag modus wag vir 'n instruksie. In toevoer modus wag dit vir teks. Om na opdrag modus te gaan druk die ESC sleutel. Indien **vi** reeds in opdrag modus is en die ESC sleutel word gedruk, sal die rekenaar piep.

Wanneer **vi** in opdrag modus is, kan daar beweeg word oor die teks na enige posisie in die leêr. Die volgende sleutels kan gebruik word om die loper te beweeg deur die leêr.

Table 5 : Sleutels om in die leêr te beweeg

Instruksie	Betekenis
<b>Pyltjies</b>	Beweeg die loper een karakter links, regs, op of af.
<b>+</b>	Beweeg die loper na die begin van die volgende lyn.
<b>-</b>	Beweeg die loper na die begin van die vorige lyn.
<b>G</b>	Beweeg die loper na die einde van die leêr.
<b>1G</b>	Beweeg die loper na die begin van die leêr.

Wanneer die loper op 'n posisie is, waar teks geredigeer moet word, kan opdragte gegee word. Wanneer 'n opdrag gegee is wat benodig dat teks ingevoer moet word, is **vi** in toevoer modus.

Table 6 : Instruksies vir die **vi** redigeerder

Instruksie	Betekenis
<b>ESC</b>	Keer terug na opdrag modus
<b>A</b>	Voeg teks by vanaf die posisie van die loper. ( <b>vi</b> gaan in toevoer modus)
<b>A</b>	Voeg teks by vanaf die einde van die huidige lyn
<b>d</b>	Vee een karakter uit vanaf die loper.
<b>dd</b>	Vee die huidige lyn uit.
<b>D</b>	Vee uit vanaf die loper tot by die einde van die huidige lyn.
<b>I</b>	Voeg teks by voor die posisie van die loper. ( <b>vi</b> gaan in toevoer modus)
<b>:q!</b>	Verlaat <b>vi</b> sonder om die lêer te stoor.
<b>R</b>	Oorskryf teks by vanaf die posisie van die loper. ( <b>vi</b> gaan in toevoer modus)
<b>U</b>	Maak die vorige verandering ongedaan.
<b>U</b>	Maak die veranderinge aan die huidige lyn ongedaan.
<b>:w</b>	Stoor die lêer.
<b>ZZ</b>	Stoor die lêer en verlaat <b>vi</b> .

## 5.2. Die **emacs** redigeerder

Om 'n teksleêr te redigeer met **emacs**, tik die instruksie **emacs toets1.txt**. Indien die lêer *toets1.txt* bestaan, sal **emacs** dit oopmaak vir redigering, en indien dit nog nie bestaan nie, sal **emacs** die lêer skep.

Wanneer die lêer in **emacs** oopgemaak is, kan daar beweeg word oor die teks na enige posisie in die lêer. Die volgende sleutels kan gebruik word om die loper te beweeg deur die lêer.

Table 7 : Sleutels om in die lêer te beweeg

Instruksie	Betekenis
<b>Pyлтjies</b>	Beweeg die loper een karakter links, regs, op of af.
<b>ctrl + a</b>	Beweeg die loper na die begin van die lyn.
<b>ctrl + e</b>	Beweeg die loper na die einde van die lyn.
<b>ESC + &gt;</b>	Beweeg die loper na die einde van die lêer. (druk eers ESC en dan ">")
<b>ESC + &lt;</b>	Beweeg die loper na die begin van die lêer. (druk eers ESC en dan "<")

Op enige posisie in die lêer kan teks eenvoudig ingevoer word, sonder dat daarvoor 'n instruksie gegee moet word. Om teks andersinds te redigeer, kan 'n aantal sleutels gebruik word.

Table 8 : Instruksies vir die **emacs** redigeerder

Instruksie	Betekenis
<b>ctrl + d</b>	Vee karakter uit.
<b>ctrl + k</b>	Vee uit tot die einde van die lyn.
<b>ctrl + xc</b>	Verlaat <b>emacs</b> .
<b>ctrl + xs</b>	Stoor die lêer.
<b>ESC + D</b>	Vee uit tot die einde van die woord.

### 5.3. Die **pico** redigeerder

Om 'n teksleër te redigeer met **pico**, tik die instruksie **pico toets1.txt**. Indien die lêer *toets1.txt* bestaan, sal **pico** dit oopmaak vir redigering, en indien dit nog nie bestaan nie, sal **pico** die lêer skep.

Wanneer die lêer in **pico** oopgemaak is, kan daar beweeg word oor die teks na enige posisie in die lêer. Die volgende sleutels kan gebruik word om die loper te beweeg deur die lêer.

Table 9 : Sleutels om in die lêer te beweeg

Instruksie	Betekenis
<b>Pyltjies</b>	Beweeg die loper een karakter links, regs, op of af.
<b>ctrl + a</b>	Beweeg die loper na die begin van die lyn.
<b>ctrl + e</b>	Beweeg die loper na die einde van die lyn.
<b>ctrl + v</b>	Beweeg die loper een skerm vorentoe.
<b>ctrl + y</b>	Beweeg die loper een skerm terug.

Op enige posisie in die lêer kan teks eenvoudig ingevoer word, sonder dat daarvoor 'n instruksie gegee moet word. Om teks andersinds te redigeer, kan 'n aantal sleutels gebruik word.

Table 10 : Instruksies vir die **pico** redigeerder

Instruksie	Betekenis
<b>ctrl + d</b>	Vee karakter uit.
<b>ctrl + k</b>	Vee lyn uit.
<b>ctrl + U</b>	Skryf lyn wat uitgegee is.
<b>ctrl + X</b>	Verlaat <b>pico</b> .
<b>ctrl + O</b>	Stoor die lêer.

### 5.4. Gebruik die drukker

Drukkers in Linux is toestelle wat gedeel word deur al die gebruikers. Om toe te sien dat alle gebruikers die drukker kan gebruik, word 'n lêer nie direk na die drukker gestuur nie, maar na 'n program, wat as 'n daemon bekend staan gestuur. Die daemon ontvang lêers van verskillende gebruikers, en sal toesien dat die lêers een vir een na die drukker gestuur word.

Om 'n lêer na die daemon te stuur, word die instruksie **lpr toets1.txt** gebruik.

Daar kan meer as een drukker opgestel wees op die rekenaar. Om te sien watter drukkers opgestel is, gebruik die instruksie `lpc status` (sommige weergawes van Linux gebruik **lpstat -a** al of **lpq -a**). Om die lêer na 'n spesifieke drukker te stuur, kan die `lpr` instruksie soos volg

## Basiese Linux

gebruik word **lpr -Pdruk1 toets1.txt**, waar “druk1” die naam van die drukker is soos in Linux opgestel.

Terwyl die drukker besig is om te druk, word die druktake gestoor in ‘n tou. Om die druktake in die tou te sien, gebruik die instruksie **lpq**. Aan elke druktaak word ‘n nommer toegeken. Om ‘n druktaak van ‘n tou te verwyder, gebruik die instruksie **lprm N** waar N die nommer is van die druktaak.

### Oefeninge

Voer die volgende uit met die rekenaar.

- 1) Teken aan in Linux.
- 2) Voer die stappe 4 tot 9 uit met die **vi**, **emacs** en **pico** redigeerders.
- 3) Skep ‘n nuwe lêer met die redigeerder. (“toetsvi.txt” in **vi**, “toetsem.txt” in **emacs** en “toetspi.txt” in **pico**)
- 4) Tik die volgende lyne teks in die lêer:  
*“Daar het ‘n doringboompie  
vlak by die pad gestaan,  
waar lange ossespanne  
met sware vragte gaan.”*
- 5) Vee die woord “n” uit in die eerste lyn.
- 6) Vee die tweede lyn uit.
- 7) Stoor die lêer uit.
- 8) Verlaat die redigeerder.
- 9) Druk die lêer uit op die drukker.

## 6. Hoe om 'n Skrip te skryf

Soms gebeur dit dat mens 'n hele reeks van instruksies herhaaldelik moet uitvoer. Om hierdie taak meer effektief te kan verrig, word gebruik gemaak van 'n skrip.

'n Skrip word geskep deur instruksies aan die bedryfstelsel in 'n teksleër te skryf, deur gebruik te maak van 'n teksredigeerder.

Die volgende skrip kan bv. geskryf word :

**VOORBEELD** (Leërnaam is “*skrvoorbeeld*”)

```
ls -al > dir.txt  
tail dir.txt  
rm dir.txt
```

Om hierdie skrip uitvoerbaar te maak, moet regte om die leër uit te voer aan die leër gegee word.

Indien alle gebruikers die skrip mag gebruik, moet die volgende regte gegee word:

**chmod 755 skrvoorbeeld**

Indien slegs die gebruiker wat die skrip geskryf het die skrip mag gebruik, moet die volgende regte gegee word:

**chmod 700 skrvoorbeeld**

Die skrip kan uitgevoer word deur dit te loop in 'n tolkprogram soos volg :

**sh skrvoorbeeld**

Al die instruksies wat Linux kan uitvoer word in die gids “*/bin*”. Om die skrip deel te maak van die bedryfstelsel, kan die skrip na hierdie gids gekopieer word. Om die skrip dan uit te voer kan dan gedoen word deur slegs die naam van die skrip te tik. Aangesien die “*/bin*” gids slegs aan die bedryfstelsel behoort, is supergebruiker regte nodig om dit te doen. Soms is dit nodig om die bedryfstelsel bewus te maak van hierdie addisionele instruksie, deur die instruksie **rehash** te gee.

**NOTA** : Die instruksies **echo** en **sleep** is baie nuttig in skripte.

## Basiese Linux

### Oefeninge

Voer die volgende uit met die rekenaar.

- 1) Teken aan in Linux.
- 2) Gebruik 'n teksredigeerder om 'n skrip te skryf wat die lêername van die lêers in die gids stoor in 'n teks lêer (Naam = "*dir.txt*") en die eerste 10 lyne van hierdie lêer op die skerm te vertoon.
- 3) Maak hierdie lêer uitvoerbaar vir die skepper van die skrip.
- 4) Voer hierdie skrip uit.
- 5) Maak hierdie skrip deel van die instuksiestel van die bedryfstelsel en voer dit as sulks uit.

## 7. Prosesse in Linux

Enige instruksie wat gegee word, of enige program wat geloop word, is 'n proses in Linux. Ook die aantekenprogram en die tolkprogram is prosesse.

Om te sien watter prosesse besig is, word die instruksie **ps** gegee. Die uitset van hierdie proses lyk soos volg:

```

VOORBEELD

bouwe@linux:~ > ps

  PID TTY          TIME CMD
  779 pts/0    00:00:01 bash
  900 pts/0    00:00:00 ps

```

Die “PID” kolom vertoon die proses-identifikasie-nommer wat aan die proses toegeken is. Hierdie is 'n sekwensiële nommer wat aan elke proses toegeken word.

Die “TTY” kolom vertoon die terminaal vanwaar die proses geloop is. Die terminaal pts/0 is 'n telnet sessie wat oor die netwerk opgestel is. Wanneer daar op die masjien self gewerk word sal terminaalname soos tty1 gebruik word.

Die “TIME” kolom vertoon die hoeveelheid tyd wat die rekenaar aan die proses gespandeer het.

Die “CMD” kolom vertoon die instruksie wat gegee is, wat die proses aan die gang gesit het. In die voorbeeld word die prosesse van die tolkprogram en die **ps** instruksie gewys.

Die **ps** instruksie het 'n aantal nuttige opsies:

- e**        Vertoon alle prosesse in die stelsel
- f**        Vertoon alle velde van die prosesse
- u**        Vertoon die prosesse van 'n spesifieke gebruiker  
(bv. ps -u bouwe)
- t**        Vertoon die prosesse van 'n spesifieke terminaal  
(bv. ps -t tty1)

As 'n instruksie gegee word of 'n program geloop word, dan sal die tolkprogram nie beskikbaar wees om enige instruksies uit te voer totdat die instruksie of program wat gegee is klaargemaak het nie. Om 'n instruksie of program uit te voer, en tegerlykertyd nog instruksies te kan gee aan die tolkprogram, kan 'n instruksie of program in die agtergrond geloop word. Dit word gedoen deur die ampersand karakter aan die instruksie agter aan toe te voeg. 'n Agtergrondproses wat deur die gebruiker afgeskop word, word 'n taak (E = job) genoem.

Hieronder word 'n voorbeeld gegee van 'n instruksie wat in die agtergrond geloop word:



## Basiese Linux

```
VOORBEELD

bouwe@linux:~ > ping 192.168.0.4 > ping.txt &

[1] 1273

bouwe@linux:~ > ps

  PID TTY          TIME CMD
  779 pts/0    00:00:01 bash
 1273 pts/0    00:00:00 ping
 1274 pts/0    00:00:00 ps

bouwe@linux:~ > jobs

[1]+  Running                  ping 192.168.0.4 >ping.txt &

bouwe@linux:~ > kill 1273

bouwe@linux:~ > ps

  PID TTY          TIME CMD
  779 pts/0    00:00:01 bash
 1305 pts/0    00:00:00 ps

[1]+  Terminated              ping 192.168.0.4 >ping.txt
```

Wanneer is instruksie gegee is, gegee Linux die proses ID terug wat aan die proses toegeken is. In die voorbeeld is die **ps** instruksie gegee, nadat die agtergrondproses begin is, en hier kan die proses gesien word. Ook die instruksie **jobs** is gegee, wat dan die status van al die agtergrondprosesse aandui.

Om die proses te beëindig, moet die instruksie **kill** saam met die proses ID (of taaknommer voorafgegaan deur die **%** karakter) gegee word. In die voorbeeld kan gesien word hoe die instruksie gegee is, en dat die proses nie meer daar is as die **ps** instruksie gegee word nie.

Indien vergeet is om 'n proses in die agtergrond te laat loop, kan die proses gestop word en dan weer in die agtergrond voortgesit word. 'n Taak kan gestop word deur die sleutel **ctrl + Z** te druk. Daarna kan die proses in die agtergrond voortgesit word deur die instruksie **bg**, saam met die taaknommer (voorafgegaan deur die **%** karakter) te gee, soos wat die **jobs** instruksie dit gee. In die voorbeeld hieronder word getoon hoe 'n taak van die voorgrond na die agtergrond geskuif word.

## Basiese Linux

### VOORBEELD

```
bouwe@linux:~ > ping 192.168.0.4 > ping.txt
[1]+  Stopped                  ping 192.168.0.4 >ping.txt
bouwe@linux:~ > jobs
[1]+  Stopped                  ping 192.168.0.4 >ping.txt
bouwe@linux:~ > bg %1
[1]+ ping 192.168.0.4 >ping.txt &
bouwe@linux:~ > ps
  PID TTY          TIME CMD
  779 pts/0    00:00:02 bash
 1316 pts/0    00:00:00 ping
 1317 pts/0    00:00:00 ps
```

Op dieselfde wyse wat 'n proses na die agtergrond geskuif word, kan die instruksie `fg` gebruik word om die proses na die voorgrond te skuif. Die instruksie `fg` gevolg deur die `%` karakter en die taaknommer word gegee.

### Opsomming van instruksies

- ps** : Vertoon prosesse.
- jobs** : Vertoon take.
- kill** : Beëindig proses of taak.
- bg** : Loop proses in die agtergrond.
- fg** : Loop proses in voorgrond.

## 8. Linux en Netwerke

Linux is 'n bedryfstelsel wat meer as een gebruiker gelyktydig kan hanteer. In die verlede het vele gebruikers na een rekenaar verbind deur middel van terminale. Nou kan meer as een gebruiker werk om 'n Linux masjien deur middel van 'n netwerk. Wanneer 'n Linux masjien op 'n netwerk, is daar 'n hele aantal instruksie waarmee daar oor die netwerk gekommunikeer kan word.

NOTA : Die opstel van die netwerkkaart word nie in hierdie handleiding beskryf nie. Om egter uit te vind hoe die netwerkkaart opgestel is, gebruik die instruksie **ifconfig**. Hierdie instruksie moet as 'n supergebruiker gegee word.

Om via 'n ander masjien aan die Internet te verbind, moet 'n standaard deurgangspoort en 'n Domein Naam Stelsel (DNS) opgestel word. Die opstelling van hierdie twee dinge word gedoen in die lêer **/etc/route.conf**. Daar moet die IP adresse van die masjien wat die deurgangspoort was opgestel word.

Die opstelling van die DNS word gedoen in die lêer **/etc/resolv.conf**.

### 8.1. Kommunikasie tussen gebruikers

Om uit te vind watter gebruikers op 'n rekenaar werk, kan die instruksies **who** en **finger** gebruik word. Die uitset van die **who** instruksie lyk soos volg :

```
bouwe@linux:~ > who
diederik tty1      Apr  5 23:34
bouwe    pts/0          Apr  5 23:34 (Bouwe)
```

Die uitset vertoon die naam van die gebruik, die terminaal vanwaar hy werk, die datum en tyd wat hy aangeteken het en ook die masjiennaam van die masjien waarop hy werk. Die instruksie **finger** vertoon dieselfde inligting, plus nog addisionele inligting.

Indien daar gesoek word na die gebruikers op 'n ander Linux masjien op die netwerk, kan die instruksie **finger @naam** gebruik word, waar naam die masjiennaam is van die ander masjien is.

Om 'n boodskap te stuur aan 'n ander gebruiker wat aangeteken is, gebruik die instruksie **write**. Saam met die instruksie moet die naam van die gebruiker gegee word, en indien dieselfde gebruiker meer as een keer aangeteken is, moet die terminaal waarvan hy die boodskap stuur ook aan. Indien die naam van die terminaal nie gegee word nie, sal Linux somer een van die terminale lukraak kies. Om bv. 'n boodskap te stuur na piet wat aangeteken is op tty3 te stuur, gebruik die volgende instruksie : **write piet tty3**.

Nadat die **write** instruksie gegee is, wag die rekenaar vir insette. Dan kan daar boodskappe ingetik word, en hierdie boodskappe sal op die rekenaar verskyn waarop die ander gebruiker aangeteken is. Wanneer jy klaar die boodskappe gestuur het, druk **ctrl + D** om die sessie te beëindig. Hierdie instruksie stuur boodskappe net in een rigting.

Vir tweerigting kommunikasie kan die instruksie **talk** gebruik word. Om 'n tweerigting sessie te begin met gebruiker piet, gebruik die instruksie **talk piet**. Piet sal op sy skerm 'n versoek kry om die sessie te aanvaar deur **talk** en die gebruiker wat die sessie versoek se naam in te tik. Nadat dit gedoen is word die skerm opgedeel in twee dele, en die twee gebruikers kan elkeen in sy gedeelte van die skerm tik, en tegerlykertyd sien wat die ander gebruiker in sy gedeelte tik. Om die sessie te beëindig, tik **ctrl + C**.

## 8.2. Elektroniese pos

Gebruikers kan vir mekaar elektroniese briewe, of epos boodskappe stuur. Om te kyk of daar epos is, tik die instruksie **mail**. Indien daar geen pos is nie, sal 'n boodskap dit sê. Indien daar wel boodskappe is, sal die program 'n por vertoon, en wag vir instruksies. Indien mai die por vertoon (**?** Of **&**), kan een van die volgende instruksies gegee word.

Table 11 : Mail nstruksies

Instruksie	Betekenis
<b>ENTER</b>	Vertoon die huidige boodskap
<b>1, 2, 3, ...</b>	Vertoon die ongeleesde boodskap met die gegewe nommer.
<b>d</b>	Vee die huidige boodskap uit
<b>u</b>	Herstel die boodskap wat sopas uitgevee is.
<b>m + gebruikernaam</b>	Skryf 'n boodskap aan die gebruiker met die gegewe naam. (Beëindig met 'n punt in 'n afsonderlike lyn)
<b>r</b>	Antwoord die boodskap wat ontvang is.
<b>z</b>	Vertoon die lys van boodskappe.
<b>p</b>	Druk die huidige boodskap.
<b>q</b>	Verlaat die mail program.
<b>?</b>	Vertoon hulp oor <b>mail</b> instruksies.

Indien mens 'n boodskap wil stuur, kan mens dit direk van uit die stelselpor doen deur die instruksie **mail gebruikernaam** te gee. Die mail program vra dan vir die onderwerp en die inhoud. Indien 'n punt (".") in 'n afsonderlike lyn gegee word, sal dit beskou word as die einde van die lyn.

'n Baie gebruikervriendelike program is **pine**. Hierdie program maak gebruik van spyskaarte, wat dit onnodig maak om enige instruksies te onthou.

## 8.3. 'n Linux sessie oor die netwerk

Indien 'n Linux masjien aan 'n netwerk verbind is, kan mens oor die netwerk aanteken na die masjien toe. Dit kan gedoen word van 'n ander Linux masjien, of ook 'n DOS of Windows masjien. Die program wat hiervoor gebruik word is **telnet**.

Om 'n telnet sessie te begin, tik **telnet** en die masjien se naam of IP adres. Telnet sal dan vra vir 'n aantekennaam en wagwoord, en dan is jy aangeteken op die Linux masjien oor die netwerk.

Indien **telnet** oopgemaak word sonder om die masjien se naam te gee, sal die **telnet** program oopmaak, en wag vir verdere instruksies. Die volgende instruksies kan dan gegee word:

Table 12 : Telnet instruksies

Instruksie	Betekenis
<b>Close</b>	Beëindig die huidige sessie.
<b>Open</b>	Begin 'n sessie na 'n gegewe masjien.
<b>Quit</b>	Beëindig die Telnet program.
<b>Set</b>	Stel parameters vir die sessie.
<b>Status</b>	Vertoon die status van die Telnet program
<b>Help</b>	Vertoon hulp oor die telnet instruksies.

#### 8.4. Die oordra van leërs na 'n Linux masjien

Om leërs oor te dra na 'n Linux masjien, word die program **ftp** gebruik (E = file transfer protocol). **Ftp** is 'n protokol gebruik om leërs oor 'n netwerk oor te dra. 'n Protokol is 'n taal wat twee masjiene met mekaar praat. In die geval van ftp, is daar twee programme betrokke, naamlik die bedienerprogram wat al die instruksies uitvoer, en die kliëntprogram waardeur die gebruiker die instruksies gee. Elke Linux bedryfstelsel het 'n **ftp** bedienerprogram ingebou. Om leërs oor die netwerk oor te dra na die Linux masjien, moet mens 'n kliëntprogram loop, wat praat met die bedienerprogram op die Linux masjien. Hierdie kliëntprogram kan op 'n Linux masjien of 'n Windows masjien of enige ander bedryfstelsel loop.

Om 'n ftp kliëntprogram op 'n Windows masjien te hardloop, maak 'n DOS venter oop, en tik **ftp** in. Op 'n Linux masjien kan eenvoudig **ftp** ingetik word. Wanneer **ftp** loop, wag dit vir instruksies van die gebruiker.

Table 13 : Ftp instruksies

Instruksie	Betekenis
<b>open</b>	Maak 'n <b>ftp</b> sessie oop na 'n masjien met 'n <b>ftp</b> bedienaar. <b>Ftp</b> sal vra vir 'n masjiennaam (of IP adres) asook 'n gebruikernaam en wagwoord vir die masjien waarop die <b>ftp</b> bedienaar is.
<b>get</b>	Lees 'n lêer met 'n gegewe naam van die huidige gids op die bedienaar en kopieer dit na die huidige gids op die kliënt masjien.
<b>mget</b>	Lees maar as een lêer met gegewe name van die huidige gids op die bedienaar en kopieer dit na die huidige gids op die kliënt masjien.
<b>put</b>	Lees 'n lêer met 'n gegewe naam van die huidige gids op die kliënt en kopieer dit na die huidige gids op die bediener masjien.
<b>mput</b>	Lees maar as een lêer met gegewe name van die huidige gids op die bedienaar en kopieer dit na die huidige gids op die kliënt masjien.
<b>bin</b>	Stel die ftp sessie op om binêre leërs (bv. programme) oor te dra.
<b>ascii</b>	Stel die ftp sessie op om teks leërs oor te

## Basiese Linux

	dra.
<b>user</b>	Teken aan op die bediener as 'n gegewe gebruiker, in die <b>ftp</b> sessie.
<b>quit</b>	Verlaat die <b>ftp</b> kliëntprogram,
<b>help</b>	Vertoon hulp oor <b>ftp</b> instruksies.

Om die huidige gids van die kliënt te verander, moet mens op die kliënt masjien na die korrekte gids gaan, en vanuit daardie gids die **ftp** kliëntprogram oopmaak. Wanneer die **ftp** sessie opgestel is na die bediener, kan die **ls** en **cd** instruksies gebruik word om na die korrekte gids op die bediener te gaan.

'n Voorbeeld van 'n ftp sessie word hieronder gegee.

### VOORBEELD

```
C:\TEMP>ftp
ftp> open 192.168.0.4
Connected to 192.168.0.4.
220 linux.local FTP server (Version 6.5/OpenBSD, linux port 0.3.2)
ready.
User (192.168.0.4:(none)): bouwe
331 Password required for bouwe.
Password:
230- Have a lot of fun...
230 User bouwe logged in.
ftp> bin
200 Type set to I.
ftp> get toets1.txt
200 PORT command successful.
150 Opening BINARY mode data connection for 'toets1.txt' (3600 bytes).
226 Transfer complete.
ftp: 2896 bytes received in 60.09Seconds 0.05Kbytes/sec.
ftp> quit
221 Goodbye.
C:\TEMP>
```

## 8.5. Lees die Web vanaf Linux

Die wêreldwye web bestaan uit webwerwe, waarop allerhande inligting gestoor word. Die inhoud van hierdie webwerwe kan gelees word met weblesers. Bekende weblesers is programme soos Internet Explorer van Microsoft en Netscape Navigator. Internet Explorer is meestal deel van die Windows bedryfstelsel en Netscape Navigator is meestal ook deel van die grafiese gebruikerskoppelvlakke van Linux.

'n Webwerf bestaan uit 'n bediener waarop lêers gelaai is met inligting waarin mense daarbuite sal belangstel. Die lêers is teksleërs waarin die inhoud gespesifiseer word in 'n taal met die naam *html* (E = HyperText Markup Language). Op die bediener loop 'n web-

## Basiese Linux

bediener, wat die *html* lêrs lees en met die weblesers kommunikeer deur middel van 'n protokol met die naam van http (E = HyperText Transfer Protocol).

Die adres van 'n Webwerf word 'n webadres (E = URL = Uniform Resource Locators) genoem. Die tipiese formaat van 'n webadres is:

**http://www.yahoo.com**

http : Dui die protokol aan waarmee die webleser moet kommunikeer.

www : Dui die WêreldWye Web aan.

yahoo : Die naam van die webwerf.

com : Die tipe webwerf. (com = kommersieel, gov = staat ens.)

Wanneer die teks gebaseerde gebruikerskoppelvlak gebruik word, kan webwerwe gelees word met 'n webleser met die naam **lynx**. Hierdie program kan 'n webwerf lees, maar kan slegs die teksedeeltes vertoon.

Tik slegs die instruksie **lynx**, om die webleser oop te maak. Wanneer **lynx** loop, kan verskillende sleutels gedruk word om sekere instruksies uit te voer.

**Table 14 : Lynx instruksies**

Instruksie	Betekenis
<b>G</b>	Gaan na 'n spesifieke webadres.
<b>Q</b>	Verlaat die <b>lynx</b> program.
<b>Regs pyltjie</b>	Gaan na 'n hiperteks koppel.
<b>Links pyltjie</b>	Gaan na die vorige bladsy.
<b>Op pyltjie</b>	Beweeg op in die bladsy.
<b>Af pyltjie</b>	Beweeg af in die bladsy.
<b>M</b>	Gaan na die tuisbladsy.

### **8.6. Loop 'n webwerf op die Linux bediener**

Saam met Linux, kom meestal 'n Apache web-bediener. Hierdie web-bediener lees die html lêrs in 'n spesifieke gids, en kommunikeer met weblesers wat versoeke rig aan die web-bediener.

Om te sien op die Apache web bedienaar loop, gebruik die instruksie **ps -ef** en kyk of daar 'n proses met die naam **httpd** loop.

Om te sien in watter gids die lêrs is wat Apache lees, kyk in die lêr **/etc/httpd/httpd.conf**, en kyk na watter gids die parameter DocumentRoot gestel is. HTML lêrs kan in hierdie gids geplaas word, en oor die netwerk gelees word deur middel van 'n webleser.

Om leer inligting oor die web-bediener te kry, gebruik die instruksie **man httpd** en kyk na die konfigurasie van die web-bediener in die lêr **/etc/httpd/httpd.conf**.

## Basiese Linux

### Opsomming van instruksies

<b>who</b>	: Vertoon die gebruikers wat aangeteken is.
<b>finger</b>	: Vertoon die gebruikers wat aangeteken is.
<b>write</b>	: Skryf 'n boodskap aan 'n ander gebruiker
<b>talk</b>	: Stel 'n gesels sessie op na 'n ander gebruiker
<b>mail</b>	: Lees en skryf elektroniese pos
<b>pine</b>	: Lees en skryf elektroniese pos.
<b>telnet</b>	: Stel 'n sessie op na 'n Linux masjien oor die netwerk
<b>ftp</b>	: Program vir die oordra van lêers na ander masjiene
<b>lynx</b>	: Webleser program



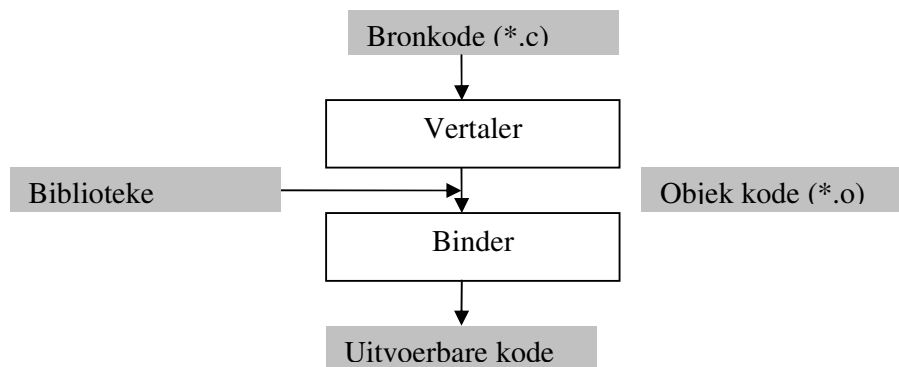
## 9. Programmering in Linux

Linux kom saam met die gereedskap om programme te ontwikkel wat op die Linux bedryfstelsel kan loop.

### 9.1. Inleiding

'n Program word in 'n spesifieke taal geskryf, wat vir mense verstaanbaar is. In die geval van Linux, word die taal met die naam "C" gebruik. In die taal kan instruksies aan die masjien gegee word, soos om 'n karakter van die sleutelbord te lees, of om 'n karakter na die skerm te skryf. 'n Aantal C instruksies bymekaar word 'n C program genoem. Die C instruksies word in 'n teksleër geskryf. So 'n teksleër met bv. C instruksies, word die bronkode van die program genoem.

Die rekenaar is nie in staat om die bronkode van 'n program uit te voer nie. Hierdie bronkode moet vertaal word na instruksies wat die rekenaar kan verstaan. Hierdie vertaling bestaan uit 'n aantal prosesse.



Die prosesse verrig die volgende funksies:

- |          |   |
|----------|---|
| Vertaler | : Vertaal bronkode objek kode vir 'n spesifieke mikro verwerker.    |
| Binder   | : Voeg die funksies uit biblioteke waarna verwys by die objek kode. |

Saam met Linux kom 'n vertaler wat bronkode in die C taal kan vertaal na objekkode. Die bronkode kan geskryf word in 'n teksleër. Hoe om 'n program in C te skryf is nie deel van hierdie boek nie. Die leër waarin die C bronkode is, moet 'n ekstensie van ".c" hê, en is die inset vir die C-vertaler. Die vertaler wat saam met Linux gegee word, is die program **gcc**.

'n Voorbeeld van 'n eenvoudige program in C word hieronder gegee.

**VOORBEELD**

```

/* first.c */

#include <stdio.h>

int main()
{
    printf("Hello world\n");
    return 0;
}

```

C is 'n baie eenvoudige taal met min funksies. Indien meer funksies benodig word, moet hulle ingesluit maak deur funksie in biblioteke te gebruik. Biblioteke kan ingesluit word in programme deur van die **#include** funksie gebruik te maak in die bronkode. Om te sien watter biblioteke almal beskikbaar is, gaan kyk in die gids **/usr/include**.

Die volgende stap is om die objek kode te bind en uitvoerbare kode te genereer. Die funksie van die binder is om die objek kode van die program te neem, en dit bind met die objek kode van die biblioteke. In die geval van die voorbeeld, moet die binder die objek kode vir die instruksie **printf** in 'n biblioteek gaan kry. Inligting oor daardie biblioteek kry die binder in die lêer **stdio.h**. Die binder wat saam met Linux gegee word is die program **ld**.

## 9.2. Die gebruik van gcc

Om die bronkode van **first.c** om te sit in uitvoerbare kode, kan die volgende instruksie gegee word:

```
gcc -o first first.c
```

Hierdie instruksie sê dat die bronkode van die lêer **first.c** vertaal moet word na objek kode en dat hierdie objek gestuur moet word aan die binder **ld**, wat dit moet omsit in uitvoerbare kode wat in die lêer **first** gestoor moet word.

Om die proses in twee afsonderlike stappe te doen, kan dit met die volgende instruksie gedoen word:

```
gcc -c first.c
```

Hierdie instruksie sal die lêer **first.c** lees, die vertaal na objek kode en die objek kode stoor in 'n lêer met die naam **first.o**. Hierna moet die objek kode omgesit word na uitvoerbare kode met die volgende instruksie.

```
gcc -o first first.o
```

Hierdie instruksie sal die objek kode in die lêer **first.o** lees, dit bind aan die biblioteke en dit dan in 'n lêer met uitvoerbare kode stoor met die naam **first**.

Om die uitvoerbare kode uit te voer, kan eenvoudig die naam van die lêer met die uitvoerbare kode gegee word. Indien die instruksie **first** nou gegee word, sal die program geloop word, en die teks "Hello world" sal op die skerm vertoon word.

Sommige weergawes van Linux weet nie in watter gids die programme loop nie. Dan moet daar 'n aanduiding gegee word in watter gids die program die instruksie “./first”.

Gestel 'n program bestaan uit meer as een leërs met bronkode.

### VOORBEELD

#### LEER 1

```
/* second.c */
#include <stdio.h>
double sinc(double);
int main()
{
    double x;
    printf("Please input x: ");
    scanf("%lf", &x);
    printf("sinc(x) = %6.4f\n", sinc(x));
    return 0;
}
```

#### LEER 2

```
/* sinc.c*/
#include <math.h>
double sinc(double x)
{
    return sin(x)/x;
}
```

Om die leër om the sit, moet die volgende instruksie gegee word:

**gcc -c second.c**

**gcc -c sinc.c**

**gcc -o second -lm second.o sinc.o**

Die **-lm** opsie gee die binder die instruksie om met die wiskundige biblioteek te bind, waarin die **sin()** funksie is. Andersinds kan die instruksie in een gekombineer word:

**gcc -o second -lm second.c sinc.c**

### 9.3. *Automatisering van gcc*

Indien mens 'n groot hoeveelheid bronleërs het wat vertaal en gebind moet word, kan dit 'n baie lang proses word om al die instruksies te gee om die leërs te vertaal en te bind. Om die proses te vergemaklik, kan van die instruksie **make** gebruik gemaak word.

Die instruksie **make**, sal al die leërs, wat gespesifiseer is in die leër met die naam **Makefile**, vertaal en bind. Die **Makefile** leër met in dieselfde gids wees as al die bronleërs.

In die voorbeeld word die Makefile gewys wat gebruik word om die uitvoerbare leër **second** te skep.

**VOORBEELD**

```
# Makefile

COMPILER      = gcc -Wall
LIBS          = -lm
EXECUTABLE    = second
OBJECT        = second.o sinc.o

$(EXECUTABLE) : $(OBJECT)
                $(COMPILER) -o $(EXECUTABLE) $(OBJECT) $(LIBS)

%.o: %.c
    $(COMPILER) -o $*.o -c $*.c
```

Die volgende dinge word in die Makefile gespesifiseer:

- COMPILER : Die **gcc** instruksie met al die opsies wat gebruik moet word.
- LIBS : Die opsies wat vir die binder vertel watter biblioteke gebruik moet word.
- EXECUTABLE : Die naam van die uitvoerbare lêer.
- OBJECT : Die name van die objekte wat gebind moet word.

Al die inkepinge in Makefile moet gedoen word met die TAB karakter, en nie met spasies nie.

Indien die instruksie make gegee word, met die Makefile in die voorbeeld, gaan Linux die volgende instruksies uitvoer:

**VOORBEELD**

```
bouwe@linux:~ > make

gcc -Wall -o second.o -c second.c

gcc -Wall -o sinc.o -c sinc.c

gcc -Wall -o second.o sinc.o -lm
```

Indien daar nou veranderings gemaak word aan een bronleër, sal Linux outmaties uiterk wat gedoen moet word, en al die nodige doen om die uitvoerbare lêer te skep.

**NOTA** : Om 'n .c bronleër om te sit in 'n netjies leesbare formaat, kan die instruksie **indent** gebruik word. Die instruksie **indent first.c** kan gegee word om die lêer **first.c** om te sit in 'n netjiese formaat.

**NOTA** : Om te help met die ontfouting van programme, kan die program **gdb** gebruik word.

## Basiese Linux

### Opsomming van instruksies

**gcc** : Program om C bronkode te vertaal.

**make** : Skep een program uit meerdere lêrs met bronkode.

### Oefeninge

Voer die volgende uit met die rekenaar.

- 1) Teken aan in Linux.
- 2) Tik die volgende program in met 'n teksredigeerder en stoor dit in 'n lêr met die naam eerste.c.

```
/* eerste.c */  
  
#include <stdio.h>  
  
int main()  
{  
    printf("Hello daar. Hierdie is jou eerste program.\n");  
    return 0;  
}
```

- 3) Vertaal die bronkode in 'n uitvoerbare program met die naam eerste.
- 4) Loop die uitvoerbare program.