

Kennismaking Linux Deel-1

Inhoudstabel :

1	Inleiding :.....	2
2.	Wat is Linux :	3
3	De eerste stap :	7
4	De kernel :	8
5	Het bestandssysteem of directorystructuur:	8
6	De Shell:	13
7	De opdrachtregel:	15
8	Een handleiding gebruiken:.....	18
9	Handige zoekopdrachten:.....	21
10	Werken met directory's en bestanden:	24
11	Teksten bewerken met de teksteditor <i>vi</i> :	30
12	Gebruikersaccounts beheren :	38
13	Groepsaccounts instellen :.....	41
14	Machtigingen instellen voor bestanden :	43
15	Bestanden Comprimeren en archiveren :	48
15	Processen :	50
16	Het geheugengebruik weergeven :	52
17	Schijfruimte beheren:	53
18	Werken met diskette- en cd-romstations:	55

1 Inleiding :

In dit deel volgt er een korte introductie van Linux Redhat 9.0. Nadat je deze tekst hebt gelezen zou je genoeg moeten weten om met Linux van start te kunnen. We veronderstellen wel dat je al een beetje van computers kent. Je weet waar het scherm staat, het verschil tussen de muis en het toetsenbord. En dat je de nodige knopjes na enige tijd wel zal terugvinden.

Zo, dan kunnen we verder...

By the way → Linux Redhat 9.0 kan je van het internet downloaden.

http://www.redhat.com/download/howto_download.html

Kijk ook eens op de volgende websites :

<http://www.sjoerd-visser.demon.nl/linux/index.html>

<http://www.gnodde.com/linux/linux.php?nr=1>

<http://users.pandora.be/tux/progs.html>

2. Wat is Linux :

Linux is een besturingssysteem, zoals Windows dat ook is.

Linux is een volwaardige implementatie van Unix voor PC's met een 386 processor of hoger. Het is een robuust en echt multi-tasking operating systeem, met ingebouwde netwerk ondersteuning.

Een besturingssysteem is een "programma", dat o.a. de communicatie regelt met de hardware van de computer en met de randapparatuur, zoals printers, modems, diskdrives, enz. Het besturingssysteem zorgt voor het op de juiste wijze opslaan en benaderen van gegevens op harde schijven en floppy's en verdeelt de beschikbare procestijd over de draaiende programma's.

Al met al is een besturingssysteem nodig om verder ook maar iets met een computer te kunnen doen.

Linux is (nog) lang niet zo bekend als Windows.

Dat komt voor een deel, omdat Linux tot voor kort een "moeilijk" besturingssysteem was, alleen geschikt voor computer-nerds. Dat is echter al enige tijd niet meer zo!

Linux kent slechts één nadeel t.o.v. Windows: de meest bekende software wordt alleen voor Windows gemaakt. Dat wil echter niet zeggen, dat er voor Linux geen goede software bestaat; integendeel zelfs...

Linux komt in meerdere 'smaken', **distributies** genaamd (Mandrake, Red Hat, SUSE, Debian, Slackware, ...). Elke distributie wordt beheerd door een afzonderlijke organisatie en wordt naar eigen wens en smaak ingedeeld.

Bij elke distributie komt afzonderlijke software voor die bij een andere distributie kan ontbreken. Niettemin komt 90% van de software voor Linux bij alle distributies voor. Zeker de softwarepakketten die uit de kinderschoenen zijn, zullen bij alle distributies voorkomen. Denk dan bijvoorbeeld aan Gnome, KDE, KMail, LICQ, VI(M), Xmms, GPhoto, Xine etc.

Linux kent vele voordelen t.o.v. Windows:

1. Linux is gratis.
2. Linux is stabiel.
3. Linux is qua uiterlijk en gedrag aan elke behoefte aan te passen.
4. Bijna alle software voor linux is gratis.
5. Er is over linux massa's informatie te vinden op het internet.
6. Linux is veiliger dan Windows.

- 1) In veel computerwinkels en boekhandels staan versies van Linux te koop voor enkele tientallen euro's. Linux kan echter ook gewoon van het internet gedownload worden. Dat is legaal. Het is zelfs legaal om Linux zelf op een CD te branden en te verkopen. Dat is wat je ziet bij die computer- en boekwinkels.
Die gekochte versies hebben als voordeel, dat er veel extra's bij zitten, zoals een gedrukte installatie-handleiding en extra software. Vaak kun je dan ook gebruik maken van een helpdesk als er problemen zijn bij de installatie.
- 2) Een goed geïnstalleerde Linux is niet plat te krijgen. Dus geen "blauwe schermen", zoals bij Windows. Vandaar dat Linux ook veel gebruikt wordt op netwerk-servers (veel websites staan op Linux servers).
- 3) Linux kent diverse "desktop" omgevingen. Je kunt kiezen, welke je het best bevalt. En zelfs dan is het uiterlijk nog, o.a. d.m.v. skins en themes, volledig naar je eigen wens aan te passen.
- 4) Bijna alle software voor Linux is, net als Linux zelf, gratis. Heel veel software wordt geschreven onder een zgn. [GPL-licentie](#) (GNU General Public License). Kort (en niet geheel volledig) gezegd betekent dat, dat deze software vrij gebruikt, gekopieerd en verspreid mag worden.
- 5) Linux is "geboren" en groot geworden op het internet. Linux wordt niet gemaakt door één persoon of firma, maar door vele mensen. Deze hebben contact met elkaar via het internet. Ook vragen, problemen en oplossingen worden veelal op het internet gepubliceerd. Voor bijna ieder probleem of iedere vraag m.b.t. Linux is wel een oplossing of antwoord te vinden op internet. Gewoon wat sleutelwoorden plaatsen bij een zoekpagina...
En is het antwoord zo toch niet te vinden, dan kun je nog terecht bij één van de vele nieuwsgroepen.
- 6) Door o.a. de opbouw van het Linux bestandssysteem is het voor een hacker moeilijker in een Linux-systeem door te dringen dan in een Windows-systeem. Ook bestaan er minder virussen voor Linux dan voor Windows en is de uitwerking van de meeste virussen slechts beperkt.
Door de openheid van de software-ontwikkeling van/voor Linux (Open Source) worden evt. veiligheidslekken in programma's snel bekend en worden die lekken ook snel verholpen.
- 7) Linux kan draaien op oudere computers, vanaf een 386. Waar Windows niet kan werken zonder grafische interface kan Linux dat dus wel. En dat is ook een voordeel: de grafische interface vraagt nogal wat van een computer, zodat er voor de eigenlijke programma's minder processortijd, geheugen, e.d. overblijft. Voor bepaalde toepassingen is zo'n grafische interface helemaal niet nodig, bijv. bij het gebruik van de computer als server. Maar: voor een "workstation" is zo'n grafische interface toch wel erg handig, zeker voor mensen, die gewend zijn aan het werken met Windows.

Linux kent verschillende "desktop-systemen": KDE, GNOME, WindowMaker, ICEWM, enz. Elk met z'n eigen voor- en nadelen. KDE zal de Windowsgebruiker zeker aanspreken: bij het omgaan ermee zijn er veel overeenkomsten met het werken met Windows.

Wat betreft de software voor Linux: alles wat met Windows kan, kan ook met Linux.

Even wat software op een rijtje:

- Office: als vervanger voor MS-Office is OpenOffice bijvoorbeeld heel geschikt. Maar de keuze is groter: KOffice, CorelOffice (WP), StarOffice, enz.
- Internet: waar men zich bij Windows meestal beperkt tot I.E. als browser, is de keuze bij Linux groter: Netscape, Opera, Mozilla, Konquerer, enz. Voor email is Kmail een aanrader en als FTP-cliënt is er ook keuze genoeg: gFTP, NCFTP, Igloo-FTP PRO, enz. ICQ, chatprogramma's, telnet, enz: er is van alles te vinden onder Linux.
- Multimedia: tal van programma's om MP3's af te spelen, op te nemen en te bewerken, maar ook allerlei andere soorten geluidsbewerkingsprogramma's tot en met complete (software) geluidsstudio's. The Gimp is een fotobewerkingsprogramma, dat zich kan meten met programma's als Photoshop. Videobewerking is ook onder Linux prima mogelijk.
- Overige: Er is een grote keuze aan database-software. Ook voor server-toepassingen is er genoeg te vinden: de webserver Apache draait onder Linux beter dan onder Windows. Er zijn zelfs mogelijkheden om pure Windowsprogramma's onder Linux te laten draaien, bijv. d.m.v. Wine, Bochs, Win4Lin, VMware, enz.

Meer informatie is o.a. te vinden op de volgende websites:



linux.be



www.linux-mandrake.com/nl/



www.Linux-mag.com

<http://www.linuxdocs.org/>

Dit is de website van het Linux documentatieproject. De HOWTO's (info over de manieren waarop u Linux kunt gebruiken) en FAQ (antwoorden op veel gestelde vragen) zijn zeer interessant.

<http://www.firstlinux.com/>

Tips voor gebruikers die Linux voor het eerst gaan installeren.

<http://www.linuxlinks.com>

Uitgebreide verzameling Linux bronnen.

3 De eerste stap :

We gaan nu voor het eerst aanloggen. Dit is al direct een eerste bewijs van de multi-user capaciteiten van Linux. Je moet eerst het Linux-systeem laten weten wie je bent voordat je ermee kan gaan werken.

We wensen niet in de grafische console maar in een tekst-console te werken. Switch naar de tekst-console *tty1* door gelijktijdig *Ctrl+Alt+F1* in te drukken. Je hebt nu een scherm voor je met de volgende prompt :

Examen login :

Hier vul je normaal je gebruikersnaam in. Helaas is je persoonlijk account door de netwerkbeheerder nog niet aangemaakt. Deze heeft al wel een algemeen gebruikersaccount *jo* voorzien. Gebruik dit account om in te loggen. Dan vraagt het Linux-systeem een paswoord. Dit is voor het gebruikersaccount *jo* ingesteld op *testuserjo*. Als je nu beide gegevens juist hebt ingevoerd dan zal je verbonden worden met het systeem en kan je beginnen met werken.

Nu staat de prompt op jou te wachten en deze zou er als volgt moeten uitzien :

```
[jo@Examen jo]$
```

Wat wil dit nu zeggen :

Het eerste deel van deze prompt, het deel voor het @-teken, geeft de gebruikersnaam weer van diegene die ingelogd is.

Achter het @-teken, hier *Examen*, staat de hostnaam van de computer.

Het laatste deel van de prompt is de naam van de directory waarin de gebruiker zich momenteel bevindt. Wanneer u aanlogt op het systeem, gaat u automatisch naar de homedirectory van uw eigen gebruikers-account. Je hebt van zodra je gebruikersaccount werd gemaakt een eigen stuk schijf op het systeem gekregen dat jouw login-naam heeft. In deze directory mag je alles doen wat je wilt.

Bij een gebruikers-account wordt de prompt afgesloten door een \$-teken, en bij het superuser-account (administrator of beheerder) door een #-teken.

Als u straks uw systeem volledig wilt afsluiten, typ dan *shutdown -h now*.

4 De kernel :

Het centrale punt van het Linux-systeem is de kernel. De kernel is software die de interface vormt tussen de gebruiker, de hardware van de computer en de aangesloten randapparatuur. De kernel zorgt voor het onderhoud van het bestandssysteem, de uitvoering van opdrachten, het starten van programma's, het timen van systeemactiviteiten en het beheer van het systeemgeheugen en andere bronnen. De kernel gebruikt systeemaanroepen om het systeem instructies te geven. Bv. Systeemaanroepen om shellopdrachten uit te voeren. Meer informatie over de shell, die de interface vormt tussen de gebruiker en het besturingssysteem, vindt u later.

5 Het bestandssysteem of directorystructuur:

In Linux wordt alles beschouwd als een bestand. Het is daarom belangrijk te weten wat een bestand precies is en hoe bestanden binnen het Linux-systeem worden ingedeeld. Allerlei soorten bestanden zijn aanwezig, bijvoorbeeld : systeembestanden, gegevensbestanden, toepassingsbestanden, hulpprogramma bestanden, stuurprogrammabestanden, configuratiebestanden, enz... Al deze bestanden hebben een naam en zijn opgeslagen in een bepaalde directory.

Het Linux-bestandssysteem is ingedeeld op basis van een boomstructuur. Er is één hoofddirectory (de root, aangegeven met het teken /). Alle andere directory's en subdirectory's (directory die opgenomen is in een andere directory) zijn onderdelen van de root. Bestanden worden opgeslagen in directory's en subdirectory's. De namen van directory's bestaan uit getallen, symbolen en letters. Een directorynaam geeft het pad naar de directory vanaf de root aan.

Opgepast : De schuine streep / wordt ook gebruikt om het onderscheid aan te geven tussen directory's, subdirectory's en bestanden. Een losstaande schuine streep / duidt de rootdirectory aan !

Een van de vertakkingen van de rootdirectory is de subdirectory met alle homedirectory's van de gebruikers (/home). Voor elke gebruiker wordt een eigen subdirectory toegevoegd aan de subdirectory /home.

Voorbeeld:

De homedirectory van de gebruiker *jo* is */home/jo*. U ziet dat er voor de eerste schuine streep helemaal niets staat. Hiermee wordt aangegeven dat */home* een subdirectory is van de rootdirectory. Omdat er na */jo* geen / volgt, kan er verwarring ontstaan. */jo* kan namelijk een subdirectory van de directory */home* zijn, maar ook een bestand dat in deze directory is opgeslagen. Voor de duidelijkheid kunt u */jo* beter laten volgen door een /, zodat iedereen begrijpt dat het hier om een directory gaat.

De bestandsnaam van een Linux-bestand bestaat uit de locatie van het bestand ten opzichte van de rootdirectory (de hoofddirectory), gevolgd door de naam die aan het bestand is toegewezen.

Een Linux systeem heeft een vaste (toch meestal) directorystructuur. M.a.w. als u op eender welk Linux-systeem werkt zal je de meeste directory's terugvinden op dezelfde plaats.

In de root staan normaal geen bestanden behalve enkele systeembestanden van Linux zelf. Voor de rest staan in de root alleen subdirectory's. Deze hebben elk een eigen functie die aangeduid wordt door de naam van die subdirectory. Nu geven we de lijst met standaard directory's die bij de installatie van Linux gemaakt worden.

- ***/tmp*** (= temporary) : Hierin komen alle tijdelijke bestanden terecht. Alle gebruikers kunnen hier tijdelijke bestanden opslaan die ook beschikbaar moeten zijn voor andere gebruikers. Alle gegevens in deze directory worden gewist wanneer het systeem opnieuw wordt opgestart.
- ***/etc*** (= etcetera) : Hierin staan alle configuratiebestanden en initialisatiescripts voor het systeem. Later leert u hoe u de bestanden in deze directory met een teksteditor kunt wijzigen om de systeemconfiguratie aan te passen. Bv. U wijzigt het bestand */etc/passwd* om gebruikers toe te voegen of om wachtwoorden te veranderen.
- ***/bin*** (= binair) : Bevat de standaardopdrachten en programma's van het systeem. Opdrachten om de shell te starten, opdrachten voor bestanden (kopiëren, verplaatsen, ...) , opdrachten waarmee u de machtigingen voor bestanden kunt wijzigen en opdrachten voor het starten van configuratiehulpmiddelen zoals 'linuxconf'.
- ***/sbin*** : Programma's die alleen bedoeld zijn voor de superuser. Deze programma's dienen om het systeem mee te beheren en zijn voor de andere gebruikers tot geen nut. Voorbeelden : opdrachten voor het afsluiten van het systeem, het instellen van de systeemklok, het controleren van het bestandssysteem op fouten en het opgeven van netwerkinstellingen.
- ***/boot*** (= booten) : Bevat alles (bestanden en opdrachten) wat nodig is om het systeem op te starten.
- ***/lib*** (= library) : Bevat de bibliotheken voor C en andere programmeertalen. Ook als u geen C-compiler op uw systeem gebruikt, is deze directory belangrijk, omdat hier alle gedeelde library's zijn opgeslagen die door toepassingen kunnen worden aangeroepen. Een gedeelde library wordt alleen in het geheugen geladen als het programma dat deze library aanroept wordt uitgevoerd. Op deze manier wordt de grootte van executable-programma's binnen de perken gehouden. Zonder deze constructie zou in bijna ieder lopend programma deels dezelfde code worden gebruikt, waardoor de opslag van het programma veel meer schijfruimte zou kosten en de uitvoering van het programma veel meer geheugen vergt.

- **/mnt** : Deze directory wordt gebruikt om verschillende opslagapparaten te **mounten** (koppelen) aan het bestandstestem. De inhoud van een cd-rom is bijvoorbeeld beschikbaar via de directory */mnt/cdrom*. Als u wilt zien wat de inhoud is van een diskette, kijkt u in de directory */mnt/floppy*.
- **/home** : Deze directory bevat een afzonderlijke subdirectory per gebruiker (met uitzondering van de superuser). Deze directory's zijn bedoeld voor persoonlijke bestanden, die niet toegankelijk zijn voor andere gebruikers van het systeem (WWW, FTP, Mail, News, ...)
- **/root** : Dit is de home-directory van de systeembeheerder of superuser.
- **/proc** (= proces) : Hierin staan een reeks bestanden die informatie over het systeem bevatten. (Versienummer, randapparatuur,...). Bevat ook virtuele bestanden die worden gebruikt voor de controle van actieve processen. U schrikt misschien van de grootte van deze bestanden, maar denk eraan dat het hier gaat om virtuele bestanden die in werkelijkheid niet aanwezig zijn.
- **/dev** (= device) : Bevat stuurprogrammabestanden (drivers) voor alle hardware die op de computer is aangesloten. Bij het configureren van het systeem om een bepaald apparaat te gebruiken, bewerkt u in werkelijkheid een van de configuratiebestanden in deze directory. Een van de apparaten die op vrijwel elke computer is aangesloten is een diskettestation. Dit wordt aangestuurd met behulp van het bestand */dev/fd0*. Een IDE-schijf met */dev/hda* (of */dev/hdb* voor de tweede harde schijf).
- **/usr** (= user) : Bevat bestanden die geen deel uitmaken van het bestandstestem van Linux. Alle gebruikersprogramma's zijn hierin gehuisvest, zoals je mail-reader, irc-programma, ... Deze directory bevat bovendien een groot aantal Help-bestanden.
- **/var** (= variable) : Hier komt alles terecht wat in de vorige directory's niet past. Een soort van allerlei-map voor het systeem. Deze directory bevat onder andere de logbestanden die aangemaakt worden bij het opstarten van het systeem. Deze bevatten informatie over de software die bij het opstarten is geladen en over de configuratie van de systeemhardware.
- **/lost+found** : Dit is een map gebruikt door het systeem om "verdwaalde" bestanden in te deponeren. Dit gebeurt automatisch in de achtergrond door een controleprogramma. Als u dus een bepaald bestand mist, controleer dan even of het hier is opgeslagen.

Oefening : De directory */usr* verkennen.

We werken met opdrachten die we op de opdrachtregel typen, en niet in een grafische omgeving !

We verkennen de directory */usr* omdat de aanwezige bestanden beschikbaar zijn voor alle gebruikers en het bovendien gaat over alleen-lezen bestanden. Voer de onderstaande stappen uit:

- 1) Vraag de inhoud van de directory */usr* op dmv de volgende opdracht:

```
ls /usr
```

Op het scherm verschijnt een lijst :

```
[jo@Examen jo]$ ls /usr
bin  etc          include      lib          local  share  tmp
dict games       kerberos     libexec     sbin   src    X11R6
[jo@Examen jo]$
```

- 2) Bekijk de lijst met toepassingen die op het systeem zijn geïnstalleerd en die voor elke gebruiker beschikbaar zijn. Typ **ls /usr/bin** en druk op *Enter*. Dit is een erg lange lijst, die waarschijnlijk veel te snel voorbij schuift.
- 3) Bekijk nu de inhoud van de directory */usr/bin* door middel van de pager *less* (een speciale viewer). Typ **ls /usr/bin | less** en druk op *Enter*. Om een pagina verder te gaan druk je telkens op de spatiebalk. Wanneer u de pager wenst te verlaten drukt u op de toets **q** (quit).

Als u brieven, rapporten, spreadsheets, ..., wilt opslaan, moet u aangeven in welke directory en onder welke naam. Deze naamgeving is gebonden aan een aantal regels:

- 1) Een bestandsnaam mag maximaal 256 tekens tellen. De bestandsnaam bestaat niet alleen uit de naam van het bestand, maar ook uit het pad naar het bestand.

Voorbeeld:

Het bestand *afbeelding.gif* dat opgeslagen is in de directory */home/jo/figuren* heeft een lengte van 31 tekens. De werkelijke bestandsnaam is namelijk */home/jo/figuren/afbeelding.gif*.

- 2) Gebruik de juiste tekens bij het instellen van bestandsnamen. U kunt alle letters uit het alfabet gebruiken (A to Z, kleine letters en hoofdletters) en de getallen van 0 tot en met 9. U kunt ook streepjes (-), liggende strepen (_) en punten (.) toevoegen.
- 3) Bestandsnamen zijn in Linux hoofdlettergevoelig. Het bestand *rapport.txt* is dus niet gelijk aan het bestand *Rapport.txt*.

- 4) De volgende tekens mogen niet gebruikt worden : *spatie*, < > ' " * {} [] () ^ ! \ | & \$? ~
- 5) Als u een verborgen bestand wilt maken, laat u de bestandsnaam vooraf gaan door een punt (.).

Oefening : een klein bestand aanmaken met de opdracht *cat* (= concatenate)

- 1) Typ *cd* (Change Directory) gevolgd door *Enter*. Hiermee ga je steeds naar je homedirectory.
- 2) Om een bestand met de naam *oefening1.txt* te maken typ je *cat > oefening1.txt* gevolgd door *Enter*. De cursor sprint naar de volgende regel. Typ enkele woorden en druk op *Enter* om naar de volgende regel te gaan. Typ nog enkele woorden en druk op *Enter*. Druk dan op **Ctrl+D** om het bestand op te slaan en te sluiten.
- 3) Ga na of het bestand is aangemaakt. Typ *ls* gevolgd door *Enter*.
- 4) Lees het bestand dat u zojuist hebt gemaakt. Typ *cat oefening1.txt* en druk op *Enter*.

Oefening : het bestand met opdrachtberichten van de kernel bekijken

In het bestand */var/log/dmesg* bevat alle berichten die tijdens het opstarten door de kernel zijn verzonden. Alleen de superuser heeft toegang tot dit bestand.

- 1) Meld u aan als superuser. Typ *su* (= Substitute User : allows a user to temporarily become another user) en druk op *Enter*. U wordt gevraagd het wachtwoord voor de rootaccount op te geven. Dit wachtwoord is *Networking!*. De prompt is nu *[root@Examen jo]#*

Merk op dat de prompt nu begint met *root*, je nog steeds in de homedirectory van gebruiker *jo* aan het werken bent, en dat de prompt wordt afgesloten met het #-teken.

- 2) Bekijk de berichten die altijd op het scherm voorbijkomen wanneer u Linux start. Typ *less /var/log/dmesg* gevolgd door *Enter*.

- 3) Lees de kernelberichten en gebruik de volgende navigatie-toetsen:

J → één regel naar beneden
Ctrl-B → één scherm terug gaan

- 4) Zoek de informatie over het type cpu, floppy-drive, harde schijven en eventueel cd-rom.

Cpu =
Floppy-drive =
Hda =
Hdc =
Hdd =

- 5) Wanneer u klaar bent drukt u *q* om terug te gaan naar de prompt.
- 6) Schakel terug naar het gebruikers-account *jo* d.m.v. het commando *exit* gevolgd door *Enter*.

De prompt is opnieuw : *[jo@Examen jo]\$*

6 De Shell:

In de komende ogenblikken leert u welke opdrachten u kunt typen op de opdrachtregel van Linux om bewerkingen uit te voeren. Hierbij maakt u gebruik van de shell. Het shellprogramma zet uw wensen om in opdrachten die door het besturingssysteem kunnen worden uitgevoerd. Elke opdracht die u achter de shellprompt typt, wordt vertaald en doorgegeven aan de Linux-kernel. De kernel geeft vervolgens aan de rest van het systeem door wat er precies moet gebeuren.

Een van de shells die deel uitmaakt van de meeste Linux-distributies en die meestal als standaard wordt geïnstalleerd, is de shell *bash*.

Eigenlijk bestaat "DE SHELL" niet. Een opsomming van een aantal shells :

sh : bourne SHell.
bash : Bourne Again SHell.
ksh : Korn SHell.
csh : C SHell.
tcsh : uitbreiding van de C SHell.
ash : A SHell.
sash : Stand-Alone SHell.

We vertellen iets meer over *bash*.

Oefening :

- 1) Informatie opvragen over de actieve shell. Deze informatie kunt u opvragen met behulp van de omgevingsvariabele *SHELL*. Typ *echo \$SHELL* en druk op *Enter*. Als u de shell *bash* gebruikt, verschijnt de volgende informatie:

```
[jo@Examen jo]$ echo $SHELL
/bin/bash
[jo@Examen jo]$
```

- 2) Nu vraag je help informatie op over *bash*. Typ *help* en druk op *Enter*.

Hiermee worden alle ingebouwde opdrachten van de shell weergegeven.

Indien de informatie te snel voorbyschuift kan je de informatie weergeven in de pager *less* (tekstviewer). Typ *help | less* en druk op *Enter*.

Elke shell is voorzien van een aantal **ingebouwde opdrachten** die deel uitmaken van de shell. Wanneer u een dergelijke opdracht gebruikt, hoeft niet in het Linux-bestandssysteem te worden gezocht naar het uitvoerbare bestand waarmee de bijbehorende toepassing kan worden uitgevoerd. Bovendien hoeft de shell de vereiste informatie niet op te vragen bij de kernel. De shell weet immers al hoe er moet worden gereageerd. Als een opdracht niet in de shell is ingebouwd, moet op basis van de directorypaden worden gezocht naar de gewenste opdracht. De directory's waarin de shell mag zoeken naar uitvoerbare programma's worden gedefinieerd met de omgevingsvariabele *\$PATH*.

Als deze omgevingsvariabele niet verwijst naar het pad van het programma dat u wilt uitvoeren, moet u de absolute bestandsnaam van het programma opgeven, omdat *bash* het programma anders niet kan vinden.

Oefening :

Typ *echo \$PATH* en druk op *Enter*, om de directory's waarin de shell mag zoeken zichtbaar te maken.

7 De opdrachtregel:

Misschien vraagt u zich af waarom u moet leren hoe u tekstopdrachten achter de shellprompt typt terwijl er zoveel handige grafische programma's beschikbaar zijn waarmee u deze taken ook kunt uitvoeren.

Er zijn verschillende goede redenen, maar het gaat er vooral om dat het bij tekstopdrachten niet uitmaakt welke Linux-distributie u gebruikt of hoe het systeem is ingesteld. Met de juiste tekstopdrachten kunt u elke taak altijd uitvoeren. U kan ook niet iedere taak vanuit een grafische omgeving uitvoeren. Gebruikers met RSI-klachten zullen waarschijnlijk liever met de opdrachtregelomgeving werken, omdat ze dan bestanden kunnen bewerken zonder de muis te hoeven gebruiken.

Alle opdrachten hebben dezelfde syntaxis of grammatica. Als u eenmaal weet hoe opdrachten zijn opgebouwd, kunt u zonder problemen elke gewenste opdracht geven. De syntaxis van opdrachten wordt als volgt aangeduid:

Naam-van-opdracht -opties [parameter]

Opdrachtopties zijn opties die u na de opdracht typt en waarmee u de resultaten van de opdracht kunt aanpassen. De opties die u op de opdrachtregel typt, moeten altijd worden voorafgegaan door een spatie en een streepje. Plaats geen spatie tussen het streepje en de opdrachtoptie. U kunt meerdere opdrachtopties toevoegen. Plaats echter geen spaties tussen de verschillende opties.

Met een **opdrachtparameter** geeft u aan op welk bestand of directory de opdracht betrekking heeft of binnen welke tijdslimiet de opdracht moet worden uitgevoerd. Parameters worden niet voorafgegaan door een streepje.

Een voorbeeld: `ls -la /usr | less`

Het voorbeeld heeft betrekking op de opdracht `ls` (= list), waarmee u de inhoud van een directory weergeeft. Met de optie `l` (= long) wordt aangegeven dat meer informatie (zoals machtigingen, eigenaren, bestandsgrootte en datum) over de bestanden moet weergegeven worden. Met de optie `a` (= all) worden alle bestanden weergegeven, ook de verborgen bestanden. Met de parameter `/usr` wordt aangegeven dat de inhoud van de directory `/usr` moet worden weergegeven, en dus niet van de huidige directory.

In Linux wordt de GNU-opdrachtregeditor '**readline**' gebruikt. Deze editor zorgt ervoor dat u tekst achter de shellprompt kunt typen en dat u de uitvoer op de opdrachtregel kunt lezen. Bovendien kunt u de tekst op de opdrachtregel bewerken. Typ eens iets achter de shellprompt. U ziet dat de cursor altijd één teken achter de getypte tekst staat. Als u een fout maakt, kunt u op de Backspace-toets drukken om tekens te wissen. Met de pijltjestoetsen links en rechts kunt u door de tekst bewegen.

Er zijn nog enkele sneltoetsen waarmee u door de tekst op de opdrachtregel kunt bewegen en waarmee u de tekst kunt bewerken. De onderstaande tabel geeft een overzicht van deze opdrachten.

Handeling	Sneltoets
Eén teken naar links gaan.	Ctrl+B
Eén teken naar rechts gaan.	Ctrl+F
Eén woord naar links gaan.	Alt+b
Eén woord naar rechts gaan.	Alt+f
Naar het begin van de regel gaan.	Ctrl+A
Naar het eind van de regel gaan.	Ctrl+E
Het teken aan de linkerkant wissen.	BackSpace
Het teken aan de rechterkant wissen.	Ctrl+D of Del
Tekst vanaf de cursor tot het eind van de regel wissen.	Ctrl+K
Hele regel wissen.	Ctrl+U

Er zijn enkele hulpmiddelen voor het werken met de opdrachtregel. Als u het aantal toetsaanslagen wilt verminderen, kunt u het volgende doen:

- Het is mogelijk dat u niet meer op de naam kunt komen van een opdracht die u onlangs nog hebt gebruikt. In dat geval kunt u op de toets met de pijl omhoog drukken om eerder gebruikte opdrachten weer te geven achter de shellprompt. Druk meerdere keren op de toets met de pijl omhoog om verder terug te gaan in de **historie**. Wanneer de gewenste opdracht wordt weergegeven, kunt u op *Enter* drukken om de opdracht uit te voeren.
- De shell *bash* is voorzien van een functie voor **automatisch aanvullen**. Wanneer u een gedeelte van een opdracht of bestandsnaam typt, wordt er een poging gedaan deze aan te vullen. U moet hiertoe wel voldoende letters van de opdracht of bestandsnaam typen. U typt enkele letters en drukt op de Tab-toets. De opdracht wordt dan aangevuld. Als er geen keuze kan worden gemaakt verschijnt er een lijst met mogelijke opdrachten. Typ in dat geval meer tekens en gebruik de functie voor automatisch aanvullen opnieuw.
- U kunt de mogelijkheden van een opdracht vergroten door **jokertekens** te gebruiken. Met jokertekens selecteert u de bestanden of directory's die aan bepaalde eisen voldoen. Het jokerteken * staat voor één of meer tekens van de naam van een bestand of een directory. Het jokerteken ? staat voor één teken van de naam van een bestand of een directory.

Wanneer u een opdracht uitvoert, wordt de invoer van het toetsenbord (de standaard invoer) gelezen en verschijnen de resultaten van de opdracht (de uitvoer) op het scherm (de standaard uitvoerlocatie). Als u wilt dat de uitvoer wordt verzonden naar een andere locatie, moet u de uitvoer omleiden. Bij het omleiden van opdrachten worden twee of meer opdrachten gescheiden door middel van het teken `>` of `|`.

U kunt de uitvoer van een opdracht bijvoorbeeld opslaan in een bestand. Op deze manier kunt u de uitvoer later op uw gemak bekijken. Gebruik het teken `>` als u de uitvoer van een opdracht wilt verzenden naar een bestand of de printer.

Het is mogelijk om twee of meer opdrachten te combineren. Als de lijst van bestanden die wordt weergegeven als resultaat van de opdracht `ls` bijvoorbeeld te lang is om te lezen, kunt u de opdracht `ls` combineren met de pager `less`. Op deze manier kunt u de lijst met bestanden rustig bekijken.

Als u opdrachten wilt combineren, moet u gebruikmaken van een sluis (aangeduid met het teken `|`). De sluis zorgt ervoor dat de uitvoer van de eerste opdracht wordt doorgesluisd en wordt gebruikt als invoer voor de tweede opdracht.

In de shell wordt een lijst bijgehouden van de laatste 500 opdrachten die u hebt getypt. Als u regelmatig met opdrachten op de opdrachtregel werkt, kan het enige tijd duren voordat u de gewenste opdracht met de pijltoetsen hebt gevonden. In dat geval kunt u beter de opdracht `history` gebruiken.

Oefening :

- 1) Typ `history` en druk op `Enter`. Er verschijnt een lijst met eerder gebruikte opdrachten. De eerste kolom duidt de volgorde aan waarin de opdrachten zijn ingevoerd. De tweede kolom geeft aan welke opdracht is uitgevoerd.
- 2) Zoek in de lijst naar de opdracht die u wilt uitvoeren. Als u de opdracht gevonden hebt, typt u `!getal` en drukt u op `Enter`. Vervang `getal` door het juiste nummer in de lijst.
- 3) Typ `history | less` als de historielijst zo lang is dat deze niet op het scherm past. Hiermee verzendt u de historielijst naar de pager `less`, zodat u de pijltjestoetsen kunt gebruiken om door de lijst te bladeren. Druk op `q` om de pager af te sluiten en terug te gaan naar de shellprompt.
- 4) Mogelijk wilt u alleen de opdrachten weergeven die u zeer recentelijk hebt gebruikt. Typ `history 10` en druk op `Enter` als u bijvoorbeeld alleen de laatste tien opdrachten wilt weergeven.

8 Een handleiding gebruiken:

U hebt nu met enkele opdrachten gewerkt en u weet hoe opdrachten worden opgebouwd. Nu is het tijd dat u leert waar u meer informatie kunt vinden over de beschikbare opdrachten en het gebruik ervan.

Informatie over een opdracht, beschrijving en mogelijke opties, vindt u in de Linux handleiding. In het Engels heet dit **manual**, vandaar de afkorting **man**. Wanneer u informatie over een opdracht opvraagt met de opdracht *man*, wordt deze getoond in een pager.

Wanneer u de opdracht *man* geeft, wordt in de titels van de handleidingen gezocht naar het gewenste onderwerp. De opdracht *man* heeft de volgende syntaxis:

man -opties [sectie] onderwerp

- Wanneer u de opdracht *man* gebruikt, moet u ten minste de naam van een onderwerp opgeven. Dit is meestal de naam van een opdracht. Met *man man* vraagt u bijvoorbeeld informatie over de opdracht *man* op. Met *man ls* vraagt u informatie op over de manier waarop u de directory-inhoud kunt weergeven.
- U kunt verschillende **opties** toevoegen om de standaarduitvoer van de opdracht *man* te wijzigen. Soms zijn er bijvoorbeeld meerdere pagina's over een bepaald onderwerp beschikbaar. In dat geval wordt de eerste pagina weergegeven die wordt aangetroffen. Als u ook de andere pagina's stuk voor stuk wilt weergeven, moet u de opdracht *man* combineren met de optie *-a* (all). Typ *man -a onderwerp* en druk op *Enter*.
- Gebruik de parameter **sectie** als u de volgorde wilt wijzigen waarin in de verschillende onderdelen van het handleidingensysteem wordt gezocht naar de pagina over het gewenste onderwerp. Deze parameter komt van pas als er meerdere pagina's over een bepaald onderwerp beschikbaar zijn.

Oefening : de opdracht *man*

- 1) Typ *man man* achter de shellprompt en druk op *Enter*.
De pagina over de opdracht *man* wordt op het scherm weergegeven. Op het scherm verschijnt slechts een gedeelte van de pagina (afhankelijk van het formaat van het scherm).
- 2) Lees de informatie op het scherm. Druk op de spatiebalk wanneer u het volgende deel van de informatie wilt weergeven.
- 3) Druk op de toets **B** als u één scherm terug wilt gaan.
- 4) Druk op *Enter* als u één regel verder wilt bladeren.
- 5) Als u in de pagina van de handleiding wilt zoeken naar een bepaald woord, typt u een schuine streep naar voren (/) gevolgd door de zoekterm en drukt u vervolgens op *Enter*. De plaatsen waar het woord voorkomt worden op het scherm gemarkeerd. Vb. */man*
- 6) Druk op de toets **q** wanneer u de pagina van de handleiding hebt gelezen. De pager wordt afgesloten en de shellprompt verschijnt opnieuw.

De pagina's van de handleiding bevatten een grote hoeveelheid informatie, maar deze is voor nieuwe Linux-gebruikers niet altijd even duidelijk. Onderstaande tabel geeft een kort overzicht van de belangrijkste onderdelen.

Naam sectie	Inhoud
Name	De naam en een korte beschrijving van de opdracht.
Synopsis	Informatie over de manier waarop u de opdracht gebruikt en een lijst met bijbehorende opdracht-opties.
Description	Informatie over de opdracht en de bijbehorende opties.
Options	Een lijst van alle opdracht-opties met een korte beschrijving van de manier waarop u elke optie kunt gebruiken.
See also	Overige pagina's van de handleiding die informatie bevatten over een vergelijkbaar onderwerp.
Diagnostics	Beschrijvingen van alle mogelijke foutmeldingen.
Bugs	Bekende problemen die kunnen optreden bij het gebruik van de opdracht of het programma.
History	Een overzicht van de ontwikkeling van het programma.
Author	Informatie over de auteur van het programma en anderen die een bijdrage hebben geleverd.

Oefening : uitvoer van de opdracht *man* aanpassen

- 1) Typ *man -a man* en druk op *Enter*.
Met deze opdracht geeft u alle pagina's van de handleiding met de titel *man* weer in de pager *less*.
- 2) De eerste pagina wordt weergegeven (zie links-boven *MAN(1)*). Gebruik de spatiebalk om het hele document te bekijken.
- 3) Druk op *q* wanneer u de pagina hebt gelezen. Nu wordt de tweede pagina van de handleiding over de opdracht *man* weergegeven (zie links-boven *MAN(7)*). Hierin vindt u informatie over macro's waarmee de pagina's van de handleiding worden opgemaakt.
- 4) Bekijk de pagina. Druk op *q* wanneer u klaar bent. De shellprompt verschijnt opnieuw.

De pagina's van de handleiding zijn ingedeeld in verschillende directory's. Als u alleen in een bepaalde directory wilt zoeken naar een pagina van de handleiding, vervangt u de parameter *sectie* door het nummer van de gewenste directory. U kunt de directory's van de handleiding ook in een specifieke volgorde doorzoeken. In onderstaande tabel ziet u hoe de pagina's van de handleiding zijn ingedeeld en welke informatie u in elke directory aantreft.

Directory	Inhoud
/usr/man/man1	Opdrachten die worden uitgevoerd vanuit een shell.
/usr/man/man2	Documentatie over systeemaanroepen.
/usr/man/man3	Informatie over de functies van <i>libc</i> .
/usr/man/man4	Informatie over bestanden in de directory <i>/dev</i> .
/usr/man/man5	Informatie over speciale bestanden zoals <i>/etc/passwd</i> .
/usr/man/man6	Spelletjes.
/usr/man/man7	Beschrijving van het Linux-bestandssysteem en de pagina's van de handleiding.
/usr/man/man8	Pagina's over hulpprogramma's voor de rootgebruiker.
/usr/man/man9	Documentatie over bronroutines van de Linux-kernel.

Elk van deze secties is voorzien van een inleiding waarin wordt aangegeven welke informatie u in de directory aantreft. Wanneer u *man intro* typt, verschijnt de introductiepagina van de directory */usr/man/man1*. Typ *man 7 intro* als u de introductiepagina wilt weergeven bij de informatie over opdrachten voor het Linux-bestandssysteem.

De directory's van de handleiding worden niet in numerieke volgorde doorzocht. De volgorde is de volgende : *man1*, *man8*, *man2* t.e.m. *man7* en tot slot *man9*. Er wordt net zolang gezocht tot er een resultaat is gevonden.

Een andere belangrijke bron van informatie is het GNU-project van Free Software Foundation. De documentatie bij GNU-software is beschikbaar in de vorm van informatiebestanden. Het programma *info* is een help-systeem op basis van hypertext dat beschikbaar is voor een groot aantal toepassingen van het GNU-project. Als u de informatiebestanden wilt lezen, doet u het volgende :

- 1) Typ *info* en druk op *Enter*. Er verschijnt een pager waarin het knooppunt *Directory* van het informatiebestand wordt weergegeven (zie linksboven op uw scherm : aanduiding *File: dir*). Deze beginpagina bevat informatie over alle onderwerpen die in de informatiebestanden aan bod komen.
- 2) Gebruik de toets met de pijl omlaag om een onderwerp, bv. de opdracht *ls*, te zoeken waarover u meer informatie wilt opvragen. Markeer het onderwerp door op de toets *M* te drukken, en druk dan op *Enter*. De informatiepagina over het geselecteerde onderwerp wordt weergegeven.
- 3) Druk terug op de toets *D* om terug naar het knooppunt *Directory* te gaan.
- 4) Wanneer u *info* wenst te verlaten drukt u op de toets *q*.

9 Handige zoekopdrachten:

Zoals u merkt, moet u heel wat weten om goed met Linux te kunnen werken. Er zijn zoveel opdrachten die u kunt gebruiken dat het niet meevalt om informatie over al deze opdrachten te verzamelen en al helemaal niet om al deze opdrachten te leren gebruiken. Gelukkig zijn er verschillende opdrachten die u kunt gebruiken om het systeem te doorzoeken en informatie te geven over de beschikbare bestanden.

De handigste opdracht voor het doorzoeken van het systeem naar een bestand is de opdracht *find*. Als u de naam van een bestand weet (als u slechts een gedeelte van de bestandsnaam weet, kunt u jokertekens gebruiken), kunt u de opdracht *find* gebruiken om in de huidige directory en alle subdirectory's naar het bestand te zoeken.

De opdracht *find* komt ook van pas bij het zoeken van bestanden die na een bepaalde datum zijn gewijzigd, of om te bepalen welke bestanden zijn gekoppeld aan welke gebruikers of groepen gebruikers (later meer hierover). Vb. Typ *Find/home*

Als u een nieuwe opdracht tegenkomt en u niet precies weet wat u met de opdracht kunt doen, gebruikt u de opdracht *whatis*. Met deze opdracht toont u een korte omschrijving van een opdracht en informatie over de directory van de handleiding waarin de pagina over de opdracht is opgenomen en dus niet de volledige pagina zoals met de opdracht *man*. Als u denkt dat de opdracht van pas komt, kunt u vervolgens de volledige pagina weergeven, zodat u kunt zien op welke manier u de opdracht gebruikt.

Oefening : de *whatis*-database doorzoeken

- 1) Typ *whatis ls* en druk op *Enter*.

```
[jo@Examen jo]$ whatis ls
ls:      nothing appropriate
```

Vraag de man pages van *whatis* op en onderzoek wat u eerst moet doen om de *whatis* database te kunnen raadplegen, en hoe u dat moet doen.

Wat en hoe ?

Probeer dit uit.

Lukt het niet, probeer het dan te realiseren als super-user.

- 2) Zorg ervoor dat je terug als user *jo* bent ingelogd.
Typ opnieuw *whatis ls* en druk op *Enter*.

De samenvattingsregel over de opdracht *ls* wordt weergegeven.
De uitvoer van deze opdracht is :

```
ls      (1)  - list directory contents
```

De eerste kolom bevat de naam van de opdracht, de tweede kolom (het getal tussen de haakjes) bevat het nummer van de directory en de derde kolom bevat een korte omschrijving uit de handleiding.

- 3) Stel dat iemand u verteld heeft over het handige programma *mtools*, dat ervoor zorgt dat u DOS-opdrachten kunt gebruiken in Linux. Als u wilt weten of dit programma op dit systeem is geïnstalleerd, typt u *whatis mtools* en drukt u op *Enter*. De uitvoer ziet er als volgt uit:

```
mtools      (1)  - utilities to access DOS disks in Unix.
mtools.conf (5)  - mtools configuration files
```

- 4) Als u de pagina's van de handleiding over de opdracht *mtools* en het configuratiebestand *mtools.conf* wilt lezen, typt u *man -a mtools* en drukt u op *Enter*.
- 5) Druk op de toets *q* wanneer u de pagina hebt gelezen. Nu wordt de tweede pagina van de handleiding (over het bestand *mtools.conf*) in de pager weergegeven.
- 6) Druk op de toets *q* wanneer u klaar bent met lezen. De shellprompt wordt opnieuw weergegeven.

Wanneer u met de opdracht *whatis* hebt bepaald wat u met een opdracht of programma kunt doen en de bijbehorende pagina van de handleiding hebt gelezen, kunt u de opdracht *whereis* gebruiken om de locatie van het binaire bestand (het uitvoerbare bestand), de bronbestanden en de pagina's van de handleiding van een bepaalde opdracht weer te geven. Als u bijvoorbeeld de locatie van het uitvoerbare bestand van de opdracht *ls* en de bijbehorende pagina van de handleiding wilt weergeven, typt u *whereis ls* en drukt u op *Enter*. U kan de opdracht *which* gebruiken als u alleen wilt weten waar het uitvoerbare bestand van een bepaalde opdracht is opgeslagen.

Als u een bepaalde taak wilt uitvoeren maar niet weet welk programma of welke opdracht hiervoor geschikt is, kunt u de opdracht *apropos* gebruiken. Stel dat u een editor wilt gebruiken maar niet weet welke editors ter beschikking zijn, typ in dat geval *apropos editor* achter de shellprompt en druk op *Enter*. Er verschijnt informatie over verschillende programma's die u kunt gebruiken. U kunt nu met de opdracht *whatis* bepalen welk programma het meest geschikt is.

Oefening : scherm leegmaken en afmelden

- 1) Als u een tijdje hebt gewerkt in een terminal bevat het venster een grote hoeveelheid tekst en bevindt de shellprompt zich onder aan het scherm. Met de opdracht *clear* kunt u het scherm op elk moment leegmaken zodat u niet wordt afgeleid door al die informatie.
- 2) Typ *clear* en druk op *Enter*. De shellprompt wordt nu bovenaan het propere scherm weergegeven.
- 3) Als u klaar bent met het werken op het Linux-systeem is het verstandig u af te melden bij uw gebruikersaccount. Typ *logout* en druk *Enter*. Hierdoor worden uw persoonlijke bestanden beveiligd. Anderen kunnen uw bestanden niet meer lezen of wijzigen, tenzij u deze gebruikers hiervoor toestemming hebt gegeven.
- 4) Log opnieuw aan als gebruiker *jo*.

10 Werken met directory's en bestanden:

Als er accounts zijn gemaakt voor de gebruikers van het systeem, is aan iedere gebruiker ook ruimte in de directory `/home` toegewezen waarin persoonlijke bestanden kunnen worden opgeslagen. De persoonlijke bestanden van gebruiker `jo` worden bijvoorbeeld opgeslagen in de directory `/home/jo`. Deze directory die de homedirectory wordt genoemd, wordt automatisch geactiveerd wanneer de gebruiker zich bij zijn account aanmeldt. De gebruiker kan subdirectory's aan deze directory toevoegen, nieuwe bestanden maken, bestanden aanpassen en overbodige bestanden verwijderen. Alleen de gebruiker zelf, anderen die door de gebruiker zijn gemachtigd en de superuser krijgen toegang tot de bestanden in de homedirectory van een gebruiker.

Wanneer u zich bij uw gebruikersaccount hebt aangemeld, heeft u de volgende shellprompt (vb gaat uit van de gebruiker `jo`): `[jo@Examen jo]$`

De prompt bevat echter geen informatie over uw locatie in het bestandssysteem. Dit is echter geen probleem. Typ de opdracht `pwd` (Print Working Directory) om de werkdirectory informatie op het scherm weer te geven. Op het scherm verschijnt :

```
/home/jo
```

De werkdirectory, die ook de huidige directory wordt genoemd, is de directory waarin de bestanden die u maakt worden opgeslagen als u geen andere directory opgeeft. Wanneer u een bestand opvraagt, wordt in de werkdirectory naar het bestand gezocht. Als het bestand in een andere directory is opgeslagen, moet u deze directory eerst activeren of het directorypad als parameter aan de opdracht toevoegen.

Met de opdracht `ls` geeft u de inhoud van uw homedirectory weer (op dit moment is de homedirectory gelijk aan de werkdirectory). Typ `ls` en druk op `Enter`.

Als u meer informatie wilt weergeven, moet u opdrachtopties gebruiken. U kunt de opdracht `ls` bijvoorbeeld uitbreiden met de optie voor het weergeven van een uitgebreide lijst. Typ `ls -l` en druk op `Enter` (`l` = long).

Op het eerste gezicht doet de hoeveelheid informatie in de uitgebreide listing wat verwarrend aan, maar als u er even tijd voor neemt wordt alles veel duidelijker. Het onderstaande voorbeeld helpt u bij het interpreteren van de informatie die wordt gegeven.

```
-rw-r--r--      1 jo      jo  4532 Jan 23  17:01 oefening1.txt
```

Het eerste teken van de eerste kolom in de lijst geeft aan of het gaat om een directory (`d`) of een bestand (`-`).

De volgende negen tekens van de eerste kolom (de letters `r`, `w`, `x`) geven u informatie over de toegangsmachtigingen van het bestand of directory. De eerste groep van 3 tekens duidt de machtigingen van de eigenaar (owner) aan, de tweede groep van drie tekens duidt de machtigingen van de groep (group) aan en de derde groep tekens duidt de machtigingen van alle andere gebruikers (other , world) aan.

Mogelijke machtigingen zijn lezen (r = read), schrijven (w = write) en uitvoeren (x = execute). Een streepje geeft aan dat de toegang wordt geweigerd.

De derde en vierde kolom bevatten informatie over de gebruiker en groep die eigenaar zijn van het bestand.

De laatste drie kolommen bevatten informatie over de bestandsgrootte (in bytes), het tijdstip waarop het bestand voor het laatst is gewijzigd en de bestandsnaam.

Syntaxis van de opdracht *ls* : `ls -opties [directorypad]`

- Als u geen directorypad opgeeft, wordt de inhoud van de huidige directory weergegeven.
- Typ *ls -a* om alle bestanden weer te geven, ook de verborgen bestanden.
- Typ *ls -F*. Hierdoor wordt achter de bestandsnamen een teken opgenomen. Directory's worden nu aangeduid met een schuine streep (/), uitvoerbare bestanden met een sterretje (*) en symbolische koppelingen met een apenstaartje (@).

De uitvoer van bestanden omleiden :

Wanneer u de bestandenlijst opvraagt met de opdracht *ls*, maar deze te lang is, schuiven de namen zo snel voorbij dat u het grootste gedeelte van de informatie niet kunt lezen. Wanneer u de uitvoer naar een pager verzendt, kunt u de lijst op uw gemak bekijken. Hiervoor typt u *ls -l | less* en drukt u vervolgens op *Enter*. Druk *q* om de pager af te sluiten.

U wenst de bestandenlijst te bewaren in een txt-bestand. Dan moet u het resultaat van de opdracht *ls* verzenden naar een tekstbestand. U typt *ls -l > bestandslijst.txt* en drukt vervolgens op *Enter*. Controleer met de opdracht *ls* of het bestand daadwerkelijk aangemaakt is.

Wenst u een overzicht van alle opties voor de opdracht *ls*, typ dan *ls -help | less* en druk op *Enter*.

Directory's maken:

Met de opdracht *mkdir* (= make directory) kunt u een subdirectory toevoegen aan de werkdirectory. U kunt het gemakkelijkst nieuwe directory's maken door ervoor te zorgen dat de directory waaraan u de nieuwe subdirectory wilt toevoegen de werkdirectory is.

Stel dat je als gebruiker *jo* een subdirectory *verslagen* wil maken in de huidige werkdirectory */home/jo*, dan typ je *mkdir verslagen*.

Controleer met de opdracht *ls* of de subdirectory daadwerkelijk aanwezig is. Gebruik vervolgens de opdracht *cd verslagen* (= change directory) om van de homedirectory naar de nieuwe directory te gaan. Hiermee stelt u de opgegeven directory in als nieuwe werkdirectory. De shellprompt wordt aangepast aan deze nieuwe werkdirectory, zoals u hier ziet : *[jo@Examen verslagen]\$*

Typ de opdracht *pwd* om de volledige padnaam van de werkdirectory te zien. Het resultaat zou er als volgt moeten uitzien :

```
/home/jo/verslagen
```

U kunt nu snel teruggaan naar uw homedirectory door alleen *cd* op de opdrachtregel te typen en vervolgens op *Enter* te drukken.

Indien u een directory met subdirectory's wenst aan te maken kunt u dit in één keer maken met de optie *-p* (= parent).

Oefening :

- 1) Stel dat u een directory voor het vak *netwerken* wilt maken. In deze directory wilt u vervolgens een subdirectory *os* (= operating systems) toevoegen. In deze directory wilt u vervolgens een subdirectory voor het operating systeem *linux*. U kunt deze drie directory's in één keer maken door de opdracht *mkdir -p netwerken/os/linux* te typen en op *Enter* te drukken. Op deze manier hoeft u dus niet eerst de directory *netwerken* te maken, naar deze directory te gaan met de opdracht *cd*, de directory *os* te maken, enzovoort...
- 2) Om alle bestanden en directory's, inclusief het volledige directorypad, van uw gebruikersaccount weer te geven typt u *find \$HOME* en drukt u op *Enter*. Als de bestanden te snel voorbij schuiven, kunt u de uitvoer omleiden naar de pager *less* of naar een bestand.

Absolute en relatieve bestandsnamen:

U kunt naar een andere directory in het bestandssysteem gaan door gebruik te maken van de opdracht *cd*. Typ hiertoe *cd* gevolgd door het directorypad.

Als de homedirectory de werkdirectory is (bijvoorbeeld */home/jo*) en u naar de directory */home/jo/netwerken/os/linux* wilt gaan, typt u *cd netwerken/os/linux* en drukt u op *Enter*. Het is niet nodig het volledige directorypad te typen omdat de locatie van de gewenste directory relatief is ten opzichte van die van de werkdirectory. Als u echter naar de directory */usr/doc* wilt gaan, moet u *cd /usr/doc* typen en op *Enter* drukken. Nu moet u de absolute bestandsnaam gebruiken, omdat het directorypad van de directory */usr/doc* niet gelijk is aan dat van de werkdirectory.

Een **absolute bestandsnaam** begint met een schuine streep (/), die de root van het bestandssysteem aanduidt. De schuine streep wordt gevolgd door het volledige directorypad van een bestand. Elk bestand op het systeem heeft een absolute bestandsnaam en u kunt deze altijd gebruiken, op welke locatie in het bestandssysteem u zich ook bevindt.

Een **relatieve bestandsnaam** duidt de locatie van een bestand ten opzichte van de werkdirectory aan. Een relatieve bestandsnaam begint niet met een schuine streep.

Bij het aanduiden van de locatie ten opzichte van de werkdirectory kunt u in opdrachten gebruikmaken van de volgende afkortingen: één punt (.) duidt de huidige directory aan en twee punten (..) duiden de ouderdirectory van de huidige werkdirectory aan. Als u naar de ouderdirectory van de huidige directory wilt gaan, typt u dus `cd ..` en drukt u op `Enter`.

Bestanden maken en bekijken:

Gebruik de opdracht `cat` als u de inhoud van een bestand snel wilt bekijken. U kunt de opdracht `cat` ook gebruiken om een of meer bestanden tegelijk te lezen of om de inhoud van twee bestanden te combineren tot één bestand. U kunt hiermee zelfs een nieuw bestand maken.

Oefening :

- 1) Ga naar de homedirectory van de gebruiker `jo`. Typ hiervoor `cd` en druk op `Enter`.
- 2) Om een bestand met de naam `oefening2.txt` te maken typ je `cat > oefening2.txt` gevolgd door `Enter`. De cursor sprint naar de volgende regel. Typ enkele woorden en druk op `Enter` om naar de volgende regel te gaan. Typ nog enkele woorden en druk op `Enter`. Druk op `Ctrl+D` om het bestand op te slaan en te sluiten.
- 3) Maak op een gelijkaardige manier `oefening3.txt` (zie stap 2).
- 4) Vroeger hebben we al eens het bestand `oefening1.txt` in deze directory aangemaakt. Controleer dit met de opdracht `ls -l`. Kijk eventueel wat in dit bestand geschreven is, typ hiervoor `cat oefening1.txt`.
- 5) We gaan nu de bestanden `oefening1.txt` en `oefening2.txt` combineren tot een bestand met de naam `oefeningen.txt`. Typ `cat oefening1.txt oefening2.txt > oefeningen.txt` en druk vervolgens op `Enter`. Kijk eventueel wat in dit bestand geschreven is, typ hiervoor `cat oefeningen.txt`.
- 6) U kunt de inhoud van een bestand ook toevoegen aan het eind van een ander bestand. Stel dat we vergeten zijn van `oefening3.txt` op te nemen in ons bestand `oefeningen.txt`. Dit doen we door `cat oefening3.txt >> oefeningen.txt` te typen en op `Enter` te drukken.

Bestanden verwijderen:

Het komt regelmatig voor dat u een bepaald bestand niet meer nodig hebt. Overbodige bestanden kunt u verwijderen met de opdracht *rm bestandsnaam* (= remove). U kunt een bestand alleen verwijderen als u schrijftoegang tot het bestand hebt.

Wees voorzichtig bij het verwijderen van bestanden. Een bestand dat u hebt verwijderd, kunt u niet meer herstellen! Door de opdracht *rm* te combineren met de optie *i* (= interactivity) wordt gevraagd of u zeker weet dat u het bestand wilt verwijderen. Typ *rm -i bestandsnaam* en druk op *Enter*. Het volgende bericht wordt weergegeven :

```
rm: 'bestandsnaam' verwijderen ?
```

Typ *y* (of *j*) en druk op *Enter* als u het bestand wilt verwijderen. Het bestand wordt nu definitief verwijderd. Als u toevallig het verkeerde bestand hebt opgegeven, typt u *N* en drukt u op *Enter*. Het bestand wordt niet gewist.

Oefening :

- 1) Verwijder het bestand *oefeningen.txt* uit de homedirectory met interactiviteit.

Directory's verwijderen:

Directory's verwijdert u met de opdracht *rmdir directorynaam* (= remove directory). Voordat u een directory kunt verwijderen, moet u de inhoud van de directory verwijderen. Dit houdt in dat u alle bestanden in de directory eerst moet wissen met de opdracht *rm*.

Als u een directory wilt verwijderen, moet u ervoor zorgen dat de werkdirectory de ouderdirectory is van de directory die u verwijdert.

Oefening :

- 1) Verwijder de directory *linux*.

Namen van bestanden wijzigen:

Er zijn verschillende redenen om de naam van een bestand te wijzigen. Het is bijvoorbeeld mogelijk dat de oorspronkelijke bestandsnaam niet duidelijk genoeg is. Namen van bestanden wijzigt u met de opdracht *mv* (= move).

De opdracht *mv* heeft de volgende syntaxis :

```
mv -opties bronbestand doelbestand
```

De parameter *bronbestand* duidt de naam aan van het oorspronkelijke bestand.

De parameter *doelbestand* duidt de nieuwe bestandsnaam aan.

Op deze manier kunt u ook de namen van directory's wijzigen.

Oefening :

- 1) Als u de naam van het bestand *oefening1.txt* wilt wijzigen in *oefening10.txt*, typt u *mv oefening1.txt oefening10.txt* en drukt u op *Enter*.
- 2) Om te voorkomen dat u een ander bestand overschrijft (dit gebeurt als er al een bestand aanwezig is met de naam van het doelbestand) kunt u de optie *-b* (= backup) toevoegen. Typ *mv -b oefening2.txt oefening10.txt* en druk op *Enter*. Het bestand *oefening10.txt* bestond al, daarom wordt de nieuwe naam *oefening10.txt~* en niet *oefening10.txt*.
- 3) U kunt ook de opdracht optie voor interactiviteit (*-i*) gebruiken om te voorkomen dat u bestanden overschrijft. Als er al een bestand met de opgegeven naam aanwezig is, verschijnt er een waarschuwing. Typ *mv -i oefening3.txt oefening10.txt* en druk op *Enter*.

Bestanden kopiëren:

U kunt een bestand kopiëren met de opdracht *cp bronbestand doelbestand*.

Deze opdracht is vergelijkbaar met de opdracht *mv*, alleen wordt het bronbestand in dit geval niet verwijderd.

Oefening :

- 1) Als u het bestand *oefening10.txt* wilt kopiëren naar een nieuw bestand *oefening1.txt*, typt u *cp oefening10.txt oefening1.txt* en drukt u op *Enter*.
- 2) Als u wilt voorkomen dat u bestanden overschrijft, gebruikt u de optie *-i*. Typ *cp -i oefening1.txt oefening10.txt* en druk op *Enter*.

11 Teksten bewerken met de teksteditor vi:

Een van de basistaken bij het werken met computers is het bewerken van tekst. Het is uiterst belangrijk dat u weet hoe u in Linux bestanden maakt en bijwerkt. De configuratiegegevens van het Linux-systeem worden bijvoorbeeld opgeslagen als bestanden met platte tekst. U kunt in deze bestanden wijzigingen aanbrengen met een teksteditor.

Een Linux-distributie is uitgerust met een aantal teksteditors. De teksteditor *vi* is echter de standaardeditor van Linux. De teksteditor *vim* is een verbeterde versie van *vi* en is in Linux-Redhat 9.0 ingesteld als de standaard teksteditor.

Een aantal andere teksteditors: *emacs*, *ed*,...

Het is belangrijk dat u leert hoe u vanaf de opdrachtregel met een teksteditor werkt. Als er problemen zijn met het systeem waardoor de grafische gebruikersinterface niet kan worden geladen en u configuratiewijzigingen moet aanbrengen om het systeem weer aan de praat te krijgen, zult u er dankbaar voor zijn dat u wat tijd heeft besteed aan uw kennismaking met *vi*.

Voor dat u een opdracht of programma gaat gebruiken, kunt u er het beste eerst iets over lezen.

Oefening :

- 1) Geef de pagina van de handleiding over het programma *vi* weer. Typ *man vi* en druk op *Enter*.
- 2) Lees de pagina. Zorg er vooral voor dat u weet hoe u het online Helpstelsysteem gebruikt. Als u bij het werken met *vi* hulp nodig hebt, drukt u op de *Esc-toets* en geeft u de opdracht *:help*.

Aan de slag met de teksteditor vi:

Met de teksteditor *vi* kunt u elk willekeurig tekstbestand in ASCII-indeling bewerken.

Wanneer u *vi* voor het eerst gebruikt, weet u misschien niet meteen wat u moet doen.

Maak een nieuw bestand door *vi oefening_vi.txt* te typen.

U ziet dat de cursor zich in de linkerbovenhoek bevindt. Wanneer u op de toets *i* (= insert) drukt, kunt u beginnen met typen. De tekst die u typt verschijnt voor de cursor. Net zoals in de meeste tekstverwerkers loopt de tekst automatisch op de volgende regel door.

Tijdens het typen kunt u op de toets *Backspace* drukken om eventuele fouten te verbeteren.

Wanneer u het einde van een alinea hebt bereikt, drukt u op *Enter* om naar een nieuwe regel te gaan.

De teksteditor *vi* kan in twee modi werken: de **opdrachtmodus** (command mode) en de **invoegmodus** (insert mode). Als u tekst aan een bestand wilt toevoegen, moet u overschakelen naar de invoegmodus. Wanneer u een bestand wilt bekijken of tekst wilt verwijderen, moet u de opdrachtmodus activeren.

De opdrachtmodus is altijd actief als u *vi* start. In deze mode kunt u opdrachten typen om taken uit te voeren. De eerste belangrijke opdracht is *i*. Met deze opdracht schakelt u over naar de invoegmodus. In de linkerbenedenhoek van het scherm verschijnt de tekst *- INSERT -*. Nu de insertmodus is geactiveerd, kunt u tekst aan het bestand toevoegen. Druk op de toets *Esc* als u wilt terugschakelen naar de opdrachtmodus.

In de opdrachtmodus kunt u ook door een bestand bewegen. Met de pijltoetsen kunt u naar boven, beneden, rechts en links bewegen. Desgewenst kunt u hiervoor ook de toetsen *k*, *j*, *l* en *h* gebruiken.

Sla het bestand regelmatig op. Er kan immers altijd een stroomstoring optreden. Als u een bestand wilt openen of opslaan, moet u overschakelen naar de opdrachtmodus (*Esc*-toets). Dergelijke opdrachten beginnen altijd met een dubbele punt (:). Het bestand waarmee u werkt kunt u snel opslaan door de opdracht *:w* (= write) te typen en vervolgens op *Enter* te drukken. Onderaan in het scherm verschijnt nu een bericht met de melding dat het document *oefening_vi.txt* is weggeschreven (*[New] written*). Voeg nu nog wat tekst toe aan het document *oefening_vi.txt*.

Als u een ander bestand wilt openen, typt u bijvoorbeeld *:e oefening10.txt* en drukt u vervolgens op *Enter*. Er verschijnt een foutboodschap omdat u de wijzigingen aan het bestand *oefening_vi.txt*, die in het buffergebied aanwezig zijn, nog niet hebt bewaard. Sla dus eerst het bestand *oefening_vi.txt* op en open daarna het nieuwe bestand.

Wanneer u *vi* wilt afsluiten, typt u *:q* en drukt u op *Enter*. De teksteditor wordt afgesloten en de shellprompt verschijnt. Het kan echter gemakkelijker. Typ *:wq* en druk op *Enter* om de wijzigingen op te slaan en *vi* af te sluiten. Als u *vi* probeert af te sluiten zonder dat u het bestand hebt opgeslagen, verschijnt er een foutmelding, tenzij u de opdracht *:q!* gebruikt.

Oefening :

- 1) Zoek de reference manual bij *vim*. Op een Linux-Redhat 9.0-systeem heeft het bestand de naam */usr/share/vim/vim61/doc/editing.txt*. Lees dit document.
- 2) Ga naar de homedirectory van de gebruiker *jo*.
- 3) We starten *vi* en maken tegelijkertijd een bestand.
Typ *vi vi_info.txt* en druk op *Enter*.
- 4) Voeg de inhoud van het hulpbestand (dat u bij stap-1 hebt gezocht) toe aan het bestand *vi_info.txt*. Typ hiertoe *:r /usr/share/vim/vim61/doc/editing.txt* en druk op *Enter*.
- 5) Sla het bestand op. Typ *:w* en druk op *Enter*.
- 6) Maak een kopie van het bestand. Typ *:w vi_info_backup.txt* en druk op *Enter*.

Wanneer u *vi* voor het eerst gebruikt, raakt u misschien ontmoedigd door het grote aantal opdrachten dat u moet onthouden om met de teksteditor te kunnen werken. Dit is echter niet nodig. Als u het programma regelmatig gebruikt, zult u al snel gewend raken aan de opdrachten en manieren waarop u tekst kunt bewerken.

Met de cursor door een bestand bewegen:

Hieronder is een tabel opgenomen met de verschillende opdrachten voor het verplaatsen van de cursor in een bestand. Probeer ze eens uit. Vergeet echter niet om eerst op de *Esc* toets te drukken. Hoort u een pieptoon ? Als dat het geval is, is de opdrachtmodus actief en kunt u een opdracht typen.

Cursorverplaatsing	Opdracht
Eén teken naar links	h
Eén teken naar rechts	l
Eén regel omlaag	j
Eén regel omhoog	k
Naar het volgende woord	w
Naar het vorige woord	b
Naar het einde van de regel	\$
Naar het begin van de regel	0
Naar de volgende zin)
Naar de vorige zin	(
Naar de volgende alinea	}
Naar de vorige alinea	{
Eén scherm verder	Ctrl+F
Eé scherm terug	Ctrl+B
Een half scherm verder	Ctrl+D
Een half scherm terug	Ctrl+U
Naar het einde van het bestand	G
Naar regel 10	10G

Tekst invoegen :

Verplaats de cursor naar de juiste locatie. Druk op de *Esc* toets, en daarna op de toets *i* (= insert). Typ vervolgens de gewenste tekst. In de onderstaande tabel vindt u informatie over andere manieren waarop u tekst kunt invoegen. Probeer ze eens uit.

Plaats waarop tekst wordt ingevoegd	Opdracht
Vóór de cursor	i
Aan het begin van de regel	I
Aan het einde van de regel	A
Op een nieuwe regel onder de cursor	o
Op een nieuwe regel boven de cursor	O
Aanwezige tekst overschrijven	R

Tekst verwijderen :

U kunt tekens of blokken tekst verwijderen. De standaardopdracht voor het verwijderen van tekst is *d*, gevolgd door een aanduiding van de cursorverplaatsing, waarmee u aangeeft wat u precies wilt verwijderen. Onderstaande tabel bevat enkele voorbeelden voor het verwijderen van tekst. Probeer ze eens uit.

Tekst die wordt verwijderd	Opdracht
Vanaf de cursor tot het einde van het woord	dw
De drie woorden na de cursor	3dw
Vanaf de cursor tot het einde van de regel	d\$
Vanaf de cursor tot het begin van de regel	d0
Vanaf de cursor tot het einde van de alinea	d}
De regel waarin de cursor zich bevindt	dd
De huidige regel en de drie volgende regels	4dd
Het teken aan de cursor	x
Het teken links van de cursor	X
Wijziging ongedaan maken (undo)	u

Tekst kopiëren :

Het indelen van teksten gaat meestal gepaard met het verplaatsen van grote hoeveelheden tekst. In Linux gebruikt men het begrip *yanken*.

Het *yanken* van tekst houdt in dat u de tekst selecteert (hoewel de tekst mogelijk niet op het scherm wordt gemarkeerd) en de geselecteerde tekst vervolgens kopieert. De gekopieerde tekst wordt tijdelijk opgeslagen in een naamloze buffer. Deze bevat alleen de tekst die u als laatste hebt geselecteerd en gekopieerd.

Plaats de cursor aan het begin van de gewenste tekst. Druk vervolgens op de toets *y* en maak de gewenste cursorverplaatsing (zie voorgaande tabel). Met *y\$* kopieert u bijvoorbeeld de tekst vanaf de cursor tot het einde van de regel. Gebruik vervolgens de opdrachten voor het verplaatsen van de cursor om de cursor op de juiste plaats te zetten. Druk op de toets *p* om de tekst te plakken. De tekst wordt naar de regel onder de cursor gekopieerd.

Als u tekst wilt verplaatsen, moet u de gekopieerde tekst eerst verwijderen met een van de opdrachten uit de bovenstaande tabel. Breng de cursor vervolgens op de locatie waar u de tekst wilt plakken en druk op de toets *p*.

→ Test uit wat hiervoor beschreven staat.

Tekst zoeken :

Als u naar een bepaald woord wilt zoeken, drukt u op de *Esc* toets om de opdrachtmodus te activeren. Typ vervolgens een schuine streep (/) en het woord dat u wilt zoeken en druk op *Enter* (bijvoorbeeld */tutor*). In de teksteditor wordt vanaf de plaats van de cursor gezocht naar de eerste locatie waar het woord (*tutor*) voorkomt. Typ */* en druk op *Enter* om te zoeken naar de volgende plaats waarop het woord *tutor* voorkomt.

Tekst zoeken en vervangen :

Als u een woord wilt zoeken en vervangen door een ander woord, moet u een hele andere opdracht gebruiken. Wanneer u het woord *tutor* bijvoorbeeld overal in het bestand wilt vervangen door het woord *cursus*, drukt u op de toets *Esc* om over te schakelen naar de opdrachtmodus en typt u vervolgens *:%s /tutor /cursus /g*. Druk daarna op *Enter*.

Verklaring : : → u gaat een opdracht invoeren
 %*s* → het volledige bestand doorzoeken (*s* =search)
 g → *tutor* op alle plaatsen vervangen door *cursus* (*g* = global)

Als u het woord slechts één keer wilt vervangen moet u de parameter *g* weglaten.

→ Test uit wat hiervoor beschreven staat.

Vi configureren :

U kunt het programma *vi* op verschillende manieren aanpassen. Dit doet u met behulp van de opties die u kunt in- of uitschakelen via de opdracht *:set*. U kunt bijvoorbeeld regelnummers weergeven aan de linkerkant van het scherm. Onderstaande tabel bevat enkele opties voor het wijzigen van de configuratie.

Optie	Toelichting
<i>:set backup</i>	Een kopie opslaan van elk bestand dat wordt overschreven.
<i>:set incsearch</i>	Zoekopdrachten automatisch aanvullen.
<i>:set lines=X</i>	Hiermee wijzigt u het aantal regels dat tegelijkertijd op het scherm wordt weergegeven (<i>X</i> duidt het aantal regels aan).
<i>:set number</i>	Regelnummers weergeven aan de linkerkant van het scherm.
<i>:set ruler</i>	Cursorpositie weergeven op de statusregel.
<i>:set scroll=X</i>	Hiermee wijzigt u het aantal regels voor het naar boven of beneden schuiven in het venster (<i>X</i> duidt het aantal regels aan).
<i>:set visualbell</i>	Hiermee stelt u een visueel signaal in (op de statusregel verschijnt <code>^I</code>) in plaats van een piepton.

Als u een van de opties wilt inschakelen, drukt u op *Esc* om de opdrachtmodus te activeren, typt u de juiste optie en drukt u vervolgens op *Enter* (vb. *:set number*). Als u een optie wilt uitschakelen, laat u de optie voorafgaan door de tekst *no* (vb. *:set nonumber*).

→ Experimenteer met bovenstaande opties.

Wijzigingen die u aanbrengt door middel van opties en de opdracht *:set* zijn slechts tijdelijk van kracht. Wanneer u *vi* afsluit, worden de standaardinstellingen hersteld. Als u de standaardinstellingen wilt aanpassen, moet u het bestand *.vimrc* bewerken of aanmaken.

Oefening :

- 1) Zoek het *vi*-configuratiebestand in de homedirectory.
Typ *ls -al* achter de shellprompt en druk op *Enter*. Het bestand *.vimrc* of *.exrc* moet in de lijst voorkomen. Indien dit niet het geval is dien je dit bestand aan te maken.
- 2) Open het configuratiebestand, typ *vi .vimrc* en druk op *Enter*.
- 3) De configuratie van *vi* zodanig aanpassen dat er automatisch kopieën worden gemaakt van bestanden die u overschrijft.

Voeg de optie *:set backup* toe.
- 4) Wijzig de configuratie zodat elk bestand wordt voorzien van regelnummers.
Typ het volgende op een lege regel:

"Regelnummers weergeven links van de tekst
set number
- 5) Sla het bestand op en sluit *vi* af. Typ *:wq* en druk op *Enter*.
- 6) Open een bestand in *vi* zodat u kunt zien wat er precies is veranderd.

12 Gebruikersaccounts beheren :

Op dit ogenblik is er een account voor de superuser *root* en een gebruikersaccount *jo* zonder privileges. Het superuseraccount gebruikt u alleen voor beheer- en configuratietaken.

U zal dadelijk een eigen gebruikersaccount aanmaken. U moet hiervoor een unieke aanmeldingsnaam toegewezen krijgen van de superuser (in dit geval uzelf) waarmee u toegang krijgt tot het systeem. Gebruikersnamen hebben een maximale lengte van acht tekens. Verder moet u ook nog een wachtwoord kiezen. Een goed wachtwoord bestaat uit een combinatie van hoofdletters, kleine letters en cijfers, en moet een lengte hebben van ten minste acht tekens.

Oefening :

- 1) Meld u aan als superuser op een vrije terminal (tty). Typ *root* achter de aanmeldingsprompt en druk op *Enter*. U wordt gevraagd om het wachtwoord op te geven. Typ *Networking!* en druk op *Enter*. De shellprompt ziet er als volgt uit:

```
[root@Examen root]#
```

- 2) Maak uw gebruikersaccount. Typ *adduser GROEP_naam* en druk op *Enter*.

Vervang hier, en in het vervolg van deze tekst, GROEP door uw groepsletter a, b, c, ..., en naam door uw naam.

- 3) Stel een wachtwoord voor uw gebruikersaccount in. Typ *passwd GROEP_naam* en druk op *Enter*. U ziet het volgende :

```
[root@Examen root]#passwd GROEP_naam  
Changing password for user GROEP_naam  
New UNIX password:
```

Typ het *password* en druk op *Enter*. Soms wordt er een foutboodschap weergegeven. Negeer deze foutboodschap.

Noteer hier het ingegeven password :

U wordt gevraagd het wachtwoord nogmaals te typen. Doe dit en druk op *Enter*. U ziet nu het volgende:

```
Passwd: all authentication tokens updated successfully  
[root@Examen root]#
```

- 4) Schakel naar een vrije terminal (*Ctrl+Alt+Fn*) maar **meldt u niet af bij het superuseraccount**. Test uit of u, met uw zopas aangemaakt account, kan aanloggen op het systeem.

- 5) Maak nog een aantal gebruikersaccounts aan : *GROEP_test1* met wachtwoord *test1*, *GROEP_test2* met wachtwoord *test2* en *GROEP_test3* met wachtwoord *test3*.

Opgepast :

Wees uiterst voorzichtig bij het bekijken en eventueel aanpassen van de bestanden die we zo dadelijk zullen behandelen. Eén typefout kan veel ellende veroorzaken. Als u problemen wilt voorkomen, kunt u het beste eerst een reservekopie maken (voeg bij het kopiëren aan de originele bestandsnaam de extensie *.org* toe) van elk bestand dat u gaat wijzigen. Als u dan een fout maakt, kunt u het oorspronkelijke bestand altijd herstellen. Als u deze bestanden liever niet zelf aanpast, kunt u ook gebruikmaken van het hulpprogramma *Linuxconf*, dat later wordt besproken.

Als u als administrator wilt weten welke systeemgebruikers zijn toegevoegd, bekijkt u het bestand */etc/passwd* . Dit bestand ziet er ongeveer als volgt uit:

```
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:./bin:
.
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
.
jo:x:501:501:./home/jo:/bin/bash
GROEP_naam:x:502:502:./home/GROEP_naam:/bin/bash
GROEP_test1:x:503:503:./home/GROEP_test1:/bin/bash
GROEP_test2:x:504:504:./home/GROEP_test2:/bin/bash
GROEP_test3:x:505:505:./home/GROEP_test3:/bin/bash
```

Er zijn drie type gebruikersaccounts:

- 1) De **superuseraccount** : *root*. Dit account, ook wel rootaccount genoemd, is bestemd voor de superuser. Het is alleen bedoeld voor systeemonderhoud en configuratietaken.
- 2) **Afzonderlijke gebruikersaccounts** : *jo*, *GROEP_naam*, Voor gewone gebruikers van het systeem worden gebruikersaccounts zonder privileges ingesteld. Elke gebruiker heeft een eigen gebruikersnaam en wachtwoord.
- 3) **Fictieve gebruikersaccounts** : *bin*, *shutdown*, Deze accounts voor systeemdaemons worden door het besturingssysteem ingesteld en zijn niet bestemd voor normale gebruikers. Ze worden gebruikt voor speciale taken waarbij toegang moet worden verkregen tot bestanden op het systeem (zoals bijvoorbeeld het verzenden van bestanden naar de printer).

Voor ieder account wordt een ingang toegevoegd aan het bestand `/etc/passwd` die bestaat uit zeven velden. De velden worden van elkaar gescheiden door een dubbele punt (:). Deze worden hier in de juiste volgorde besproken:

- 1) De **gebruikersnaam** waarmee de gebruiker zich aanmeldt.
- 2) Het **wachtwoord** wordt aangeduid met de letter *x*. Alle wachtwoorden zijn namelijk gecodeerd, zodat onbevoegden het bestand `/etc/passwd` niet kunnen misbruiken om de wachtwoorden van anderen op te zoeken.
- 3) De **gebruikers-ID** is een uniek getal dat aan elk account wordt toegewezen. Deze getallen worden door het systeem gebruikt bij het verwerken van machtigingen voor bestanden.
- 4) De **groeps-ID** geeft aan van welke primaire groep een gebruiker deel uitmaakt. Een gebruiker kan deel uitmaken van meerdere groepen, maar alleen de primaire groep wordt opgenomen in dit bestand. Later leest u meer over groepsaccounts.
- 5) In dit bestand kan ook **allerlei informatie** over de gebruiker worden opgenomen, bijvoorbeeld de volledige naam, het adres, enz...
- 6) De **homedirectory** van de gebruiker. Een directory voor de persoonlijke bestanden van de gebruiker. Deze directory wordt bij de aanmelding ingesteld als werkdirectory.
- 7) Het laatste veld duidt de **shell** aan die wordt gestart wanneer de gebruiker zich aanmeldt. In de meeste Linux-distributies is dit `/bin/bash`.

Oefening :

- 1) Wanneer duidelijk wordt dat een bepaalde persoon het systeem niet meer zal gebruiken, kan de superuser de bijbehorende gebruikersaccount wissen. Dit kan op twee manieren :
 - a) De gebruikersaccount en de bijbehorende homedirectory (inclusief alle bestanden) verwijderen. Typ `userdel -r GROEP_test1`.
Ga na of de homedirectory `/home/GROEP_test1` nog bestaat.
 - b) Alleen de account verwijderen en de homedirectory intact laten.
Typ `userdel GROEP_test2`.
Ga na of de homedirectory `/home/GROEP_test2` nog bestaat.
- 2) Bekijk het bestand `/etc/passwd` en ga na of de accounts `GROEP_test1` en `GROEP_test2` zijn verwijderd.

- 3) Het is belangrijk dat een gebruiker zijn wachtwoord regelmatig wijzigt. Als u dat wilt doen, doet u het volgende:
- Ga naar de terminal waar u bent aangelogd met uw persoonlijk gebruikersaccount.
 - Typ *passwd* en druk op *Enter*.
 - U wordt gevraagd het huidige wachtwoord op te geven. Typ het wachtwoord en druk op *Enter*.
 - U wordt gevraagd het nieuwe wachtwoord op te geven. Typ het nieuwe wachtwoord en druk op *Enter*.
 - U wordt gevraagd het nieuwe wachtwoord nog eens te typen. Typ het nieuwe wachtwoord opnieuw en druk op *Enter*.

13 Groepsaccounts instellen :

Groepsaccounts komen van pas als verschillende gebruikers moeten werken met dezelfde bestanden. U kunt zoveel groepsaccounts instellen als u nodig hebt, en een groep kan bestaan uit elke gewenste combinatie van gebruikers. Een bepaalde gebruiker kan zelfs deel uitmaken van verschillende groepen.

Als u wilt weten welke groepen op het systeem zijn ingesteld, bekijkt u het bestand */etc/group*. Dit bestand ziet er ongeveer als volgt uit:

```
root:x:0:root
bin::x:1:root,bin,daemon
daemon:x:2:root,bin,daemon
sys:x:3:root,bin,adm
.
jo:x:501:
.
GROEP_naam:x:506:
GROEP_test3:x:509:
```

Het bestand `/etc/group` bevat voor elke groep een aparte ingang. Elke ingang bestaat uit vier velden. De velden worden van elkaar gescheiden door een dubbele punt (:). Deze velden worden hier in de juiste volgorde besproken:

- 1) De **groepsnaam** die voor de groep is ingesteld.
- 2) Het **wachtwoord** (dit is optioneel). Dit veld bevat meestal het teken *x*. Wanneer u een wachtwoord voor een groep instelt, zijn de bestanden van de groep alleen toegankelijk voor gebruikers die het wachtwoord weten.
- 3) De **groeps-ID**.
- 4) De **lijst met extra gebruikers** die aan de groep zijn toegewezen.

Wanneer u in Linux een gebruikersaccount instelt, wordt de gebruiker toegewezen aan een **primaire groep** waarvan de naam overeenkomt met de gebruikersnaam.

U voegt het gemakkelijkst nieuwe groepen toe door het bestand `/etc/group` te wijzigen. U kunt alleen groepsaccounts instellen als u bent aangemeld bij de superuseraccount.

Oefening :

- 1) Maak een reservekopie van het bestand `/etc/group`.
- 2) Open het bestand `/etc/group` in een teksteditor.
- 3) Blader naar het einde van het bestand en begin een nieuwe regel. Op deze regel voegt u de informatie over de nieuwe groep toe. Denk eraan dat u de volgende indeling moet gebruiken : `groepsnaam:x:groep-ID:groepsleden`
- 4) Typ de groepsnaam `ims_GROEP` (vervang `GROEP` door uw groepsletter *a, b, c, ...*) gevolgd door een dubbele punt (:).
- 5) Laat het wachtwoord leeg, typ *x*:
- 6) Geef een groeps-ID op (neem hiervoor het nummer volgend op de laatste groepsnummer in `/etc/group`) gevolgd door een dubbele punt.
- 7) Wijs de volgende leden toe. Typ `GROEP_test3, GROEP_naam`
- 8) Sla het bestand op en sluit het.
- 9) Ga naar de terminal waar je bent aangemeld met uw gebruikersaccount `GROEP_naam`.
- 10) Maak de directory `/tmp/map_GROEP` (vervang `GROEP` door uw groepsletter) waarin de groepsleden bestanden kunnen opslaan.
- 11) Maak een tekstbestand `welkom.txt` aan in de directory `/tmp/map_GROEP`. Sluit het bestand af.

- 12) Vraag een long listing op van de directory `/tmp/map_GROEP`.
U moet het volgende zien:

```
-rw-rw-r-- 1 GROEP_naam GROEP_naam 4532 Jan 23 17:05 welkom.txt
```

- 13) Een gebruiker kan met de opdracht **groups** opvragen tot welke groepen hij behoort. Test dit uit bij verschillende gebruikers van het Linux-systeem.

Opmerking : Als het gebruikersaccount actief is, wordt de nieuwe groepsaccount mogelijk niet getoond. Wanneer de gebruiker zich afmeldt en opnieuw aanmeldt, is de nieuwe account wel zichtbaar.

- 14) Een gebruiker kan ook de opdracht **id** gebruiken om de ID van zijn gebruikersaccount en de bijbehorende groepen weer te geven. Test dit uit bij verschillende gebruikers van het Linux-systeem.

Opmerking : U kunt een groepsaccount wissen door de ingang voor de groep te verwijderen uit het bestand `/etc/group`. Doe dit nu nog niet !

14 Machtigingen instellen voor bestanden :

Wanneer u accounts hebt ingesteld voor alle gebruikers en groepen die met het systeem werken, kunnen de gebruikers beginnen met het opslaan van bestanden op het systeem. Het groepslid dat een bestand (of een directory) maakt, wordt beschouwd als eigenaar van het bestand. Deze gebruiker kan vervolgens eigendomsrechten en machtigingen toewijzen aan andere groepsleden en andere systeemgebruikers. Dit systeem van eigendomsrechten en machtigingen vormt de basis voor het toegangssysteem.

Machtigingen voor bestanden bepalen welke systeemgebruikers toegang krijgen tot bestanden en welk type toegang wordt toegestaan.

Er zijn drie soorten eigendomsrechten voor bestanden:

- 1) De eigenaar is de gebruiker die het bestand heeft gemaakt.
- 2) De groep is de standaardgroep waarvan de eigenaar deel uitmaakt. Dit is de **primaire groep** die is opgenomen in het bestand `/etc/passwd`.
- 3) Er kunnen overige eigendomsrechten worden toegewezen aan systeemgebruikers die geen deel uitmaken van de groep.

Voor een bestand of directory kunnen drie machtigingsniveaus worden ingesteld, en deze kunnen verschillend zijn voor de verschillende soorten eigenaren.

- 1) Een gebruiker of groep met leestoegang (*read*) tot een :
 - bestand : heeft toestemming de inhoud van het bestand te bekijken.
 - directory : kan de inhoud van de directory weergeven.
- 2) Een gebruiker of groep met schrijftoegang (*write*) tot een :
 - bestand : kan dit bestand wijzigen of verwijderen
 - directory : kan bestanden aan de directory toevoegen en verwijderen.
- 3) Een gebruiker of groep met uitvoertoegang (*execute*) tot een :
 - bestand : kan dit bestand uitvoeren als opdracht of programma.
 - directory : krijgt toegang tot de directory en kan informatie weergeven over de bestanden die in de directory zijn opgeslagen.

Zoals we reeds vroeger zagen, kunt u met de opdracht *ls -l* informatie over de eigendomsrechten van gebruikers en groepen voor bestanden en directories weergeven.

De eigendomsrechten van bestanden wijzigen :

U kunt de eigendomsrechten van een bestand (dus de gebruiker en groep die worden beschouwd als eigenaar van het bestand) aanpassen met de opdrachten *chown* (= change owner) en *chgrp* (= change group).

Oefening :

- 1) Log aan met uw gebruikersaccount of ga naar de terminal waar u met uw gebruikersaccount bent aangemeld.
- 2) Ga naar de directory */tmp/map_GROEP*, typ *ls -l* en druk op *Enter*.
U moet ongeveer het volgende zien:

```
-rw-rw-r-- 1 GROEP_naam GROEP_naam 4532 Jan 23 17:01 welkom.txt
```

- 3) We gaan voor dit bestand de groep *ims_GROEP* instellen als eigenaar.
Typ *chgrp ims_GROEP welkom.txt* en druk op *Enter*.
- 4) Typ opnieuw *ls -l* en druk op *Enter* om na te kijken of de wijziging is doorgevoerd. U moet het volgende zien :

```
-rw-rw-r-- 1 GROEP_naam ims_GROEP 4532 Jan 23 17:01 welkom.txt
```

Bestanden beveiligen :

Nu de eigendomsrechten van de bestanden zijn ingesteld, kunt u het type toegang van de verschillende soorten eigenaren wijzigen. Het instellen van machtigingen voor bestanden is iets ingewikkelder dan het wijzigen van de eigendomsrechten.

U wijzigt de machtigingen voor een bestand met de opdracht *chmod* (= change mode). Deze opdracht heeft de volgende syntaxis:

chmod toegangsreeks bestandsnaam

De parameter *toegangsreeks* heeft de volgende syntaxis :

Eigenaar+operator+bestandsmachtiging

Eigenaar duidt een van de volgende eigenaren van het bestand aan:

- Het teken *u* (= user) → machtiging voor de gebruiker die eigenaar is van het bestand.
- Het teken *g* (= group) → machtiging voor de groep die eigenaar is van het bestand.
- Het teken *o* (= others) → machtiging voor alle andere gebruikers.
- Het teken *a* (= all) → machtiging voor alle drie soorten eigenaren.

Operator kan de volgende waarden hebben:

- Teken *+* → machtiging toevoegen.
- Teken *-* → verwijder machtiging.
- Teken *=* → exacte bestandsmachtiging ?????

Bestandsmachtiging kan de volgende waarden hebben:

- Teken *r* (= read) → instellen leestoegang.
- Teken *w* (= write) → instellen schrijftoegang.
- Teken *x* (= execute) → instellen uitvoertoegang.

Het volgende bestand wordt gebruikt als voorbeeld:

```
-rw-rw-r-- 1 GROEP_naam ims_GROEP 4532 Jan 23 17:01 welkom.txt
```

Oefening :

- 1) Een gebruiker (vb. *GROEP_test3*) die lid is van de groep *ims_GROEP* kan het bestand *welkom.txt* op dit moment lezen en wijzigen. Test dit uit.
- 2) Als gebruiker *GROEP_naam* ga je de groep *ims_GROEP* alleen leestoegang geven voor het bestand *welkom.txt* . Dit is mogelijk omdat u de eigenaar bent van dit bestand. Typ aan de prompt:

```
[GROEP_naam@Examen map_GROEP]$ chmod g-w welkom.txt
```

- 3) Toon de nieuwe machtigingen voor het bestand . Typ *ls -l* en druk op *Enter*. De uitvoer zou er als volgt moeten uitzien:

```
-rw-r--r-- 1 GROEP_naam ims_GROEP 4532 Jan 23 17:01 welkom.txt
```

- 4) Test uit of een gebruiker (vb. *GROEP_test3*) die lid is van de groep *ims_GROEP* het bestand *welkom.txt* nu nog kan wijzigen en opslaan.
- 5) Voor alle gebruikers de alleen-lezen machtiging instellen. Typ aan de prompt:

```
[GROEP_naam@Examen map_GROEP]$ chmod a=r welkom.txt
```

- 6) Toon de nieuwe machtigingen voor het bestand . Typ *ls -l* en druk op *Enter*. De uitvoer zou er als volgt moeten uitzien:

```
-r--r--r-- 1 GROEP_naam ims_GROEP 4532 Jan 23 17:01 welkom.txt
```

- 7) Test uit of een gebruiker (vb. *GROEP_test3*) die lid is van de groep *ims_GROEP* , en een gebruiker (vb. *jo*) die geen lid is van de groep *ims_GROEP* , het bestand *welkom.txt* alleen kan lezen en niet kan wijzigen.

- 8) Gebruikers die geen lid zijn van de groep *ims_GROEP* mogen het bestand *welkom.txt* niet kunnen lezen. Typ aan de prompt:

```
[GROEP_naam@Examen map_GROEP]$ chmod o-r welkom.txt
```

- 9) Toon de nieuwe machtigingen voor het bestand . Typ *ls -l* en druk op *Enter*. De uitvoer zou er als volgt moeten uitzien:

```
-r--r----- 1 GROEP_naam ims_GROEP 4532 Jan 23 17:01 welkom.txt
```

- 10) Test uit of een gebruiker (vb. *jo*) die geen lid is van de groep *ims_GROEP* het bestand *welkom.txt* nog kan lezen.

11) Indien u **uw gebruikersnaam** wilt wijzigen, maar er al bestanden opgeslagen zijn in uw homedirectory die u wilt behouden, doe dan het volgende:

→ Meld u aan met het superuseraccount.

→ Typ `usermod -l nieuwe_gebruikersnaam -d /home/nieuwe_gebruikersnaam -m gebruikersnaam`

- Met de optie `-l` stelt u de nieuwe gebruikersnaam in.
- Met de optie `-d` stelt u de nieuwe directory in.
- Met de optie `-m` geeft u aan dat u alle gebruikersbestanden naar de nieuwe directory wilt verplaatsen.

12) Test uit of u met uw gewijzigde gebruikersnaam kan aanloggen op het Linux-systeem. Ga na of uw gemaakte documenten aanwezig zijn in uw nieuwe homedirectory.

Opmerking :

Met de opdracht `chmod` kunt u de machtigingen voor bestanden ook instellen door gebruik te maken van getallen.

Bestandsmachtiging kan de volgende waarden hebben:

- 4 (= read) → instellen leestoegang.
- 2 (= write) → instellen schrijftoegang.
- 1 (= execute) → instellen uitvoertoegang.

Als u bijvoorbeeld lees- (=4) en schrijftoegang (=2) voor de gebruiker en groep met eigendomsrechten van een bestand wilt instellen, telt u de getallen 4 en 2 voor de gebruiker en voor de groep bij elkaar op. We wensen alleen leestoegang (=4) voor alle overige gebruikers. De opdracht ziet er als volgt uit:

`chmod 664 bestandsnaam`

Oefening:

- 1) Pas het bovenstaande toe op het bestand `welkom.txt`.
- 2) Toon de nieuwe machtigingen voor het bestand . Typ `ls -l` en druk op `Enter`.
Zijn de machtigingen aangepast ?
- 3) Test uit of de eigenaar, en een gebruiker van de groep, het bestand `welkom.txt` kunnen wijzigen.

15 Bestanden Comprimeren en archiveren :

Alhoewel de capaciteit van de schijven, die voor opslag van files gebruikt worden, steeds groter wordt, is het belangrijk om grote files die niet vaak gebruikt worden, te **comprimeren** en zo schijfruimte uit te sparen. Tekstfiles kunnen op deze wijze bijvoorbeeld tot ongeveer 1/3 van de oorspronkelijke grootte teruggebracht worden, zonder informatie te verliezen.

In combinatie met programma's die files of delen van de boomstructuur van het filesysteem in één file samenballen, het **archiveren**, kunnen er bijvoorbeeld efficiënt kopieën (backups) gemaakt worden.

Onder Linux wordt voor het **comprimeren** van files het programma *gzip* gebruikt. De omgekeerde operatie (het "uitpakken") gaat met *gunzip*.

Gecomprimeerde bestanden hebben de extensie **.gz**.

Voorbeeld:

```
# ls -l m*
-rw-r--r--  1 jo  jo           39123 Aug  4 13:57 mammoet.txt

# gzip mammoet.txt

# ls -l m*
-rw-r--r--  1 jo  jo           8089 Aug  4 13:57 mammoet.txt.gz

# gunzip mammoet.txt

# ls -l m*
-rw-r--r--  1 jo  jo           39123 Aug  4 13:57 mammoet.txt
```

Opmerkingen :

- **een gecomprimeerde file is een binaire file!**
- *Gzip* en *gunzip* laten het gebruik van wildcards of jokers toe.
Bijvoorbeeld :

```
# gzip *           →  Alle bestanden in de huidige directory worden
                       gecompriemd.
# gzip -r *       →  Alle bestanden in de huidige directory en de
                       onderliggende subdirectory's worden gecompriemd
                       (r = recursief).
```

Elke compressieprogramma comprimeert op z'n eigen manier, al zijn er programma's die "multi-functioneel" zijn en verschillende formaten aankunnen. Dus voordat je gecomprimeerde files naar een ander systeem overbrengt, moet je eerst controleren of de files daar uitgepakt kunnen worden.

→ Test uit wat hiervoor beschreven staat.

Voor **archiveren** wordt het programma *tar* gebruikt. Het omgekeerde, files (eventueel selectief) uit het archief halen, wordt ook met *tar* gedaan. Dit is dus een handig programma om de ruimte voor opslag van veel gegevens te beperken en om hele directories met e-mail te versturen. *Tar* wordt veel gebruikt voor de distributie van softwarepakketten.

Oorspronkelijk diende *tar* voor het schrijven op en lezen van (archieftapes, vandaar de naam "Tape ARchive".

De belangrijkste vlaggetjes zijn:

Optie		
t	Table of contents	Geeft een lijst van alle files/ directories in een archief.
x	Extract	Geeft aan dat er iets uit het archief gehaald moet worden.
c	Create	Geeft aan dat er een nieuw archief gemaakt moet worden.
z	Zip (compress)	
v	Verbose	Toon de bestandsnamen terwijl ze worden gearchiveerd.
f	Archive file	Zorgt ervoor dat de subdirectory's ook worden gearchiveerd.

Voorbeeld :

Archiveren van de directory *beestjes* in het archief *beest.tar* (een al bestaande file *beest.tar* wordt hierbij overschreven):

```
# tar -cf beest.tar beestjes
```

Een inhoud tonen van het archief:

```
# tar -tf beest.tar  
  
beestjes/  
beestjes/olifant  
beestjes/giraffe  
beestjes/poes  
beestjes/vissen/  
beestjes/vissen/snoek  
beestjes/vissen/paling
```

Weghalen van de directory *beestjes* met inhoud:

```
# rm -r beestjes
```

Terugzetten van alleen de directory *beestjes/vissen*, met inhoud:

```
# tar -xf beest.tar beestjes/vissen
```

Opmerkingen :

- **een getarde file is een binaire file!**
- De extensies *.tar* en *.gz* moeten door de gebruiker zelf worden toegevoegd.
- Een gecomprimeerd tar-bestand heeft de extensie *.tar.gz* of *.tgz*.

→ Test uit wat hiervoor beschreven staat.

15 Processen :

In de afgelopen uren hebt u verschillende opdrachten geprobeerd. Daarbij hebt u zonder het te weten diverse processen gestart. U weet dat de shell *bash* de interface vormt tussen de gebruiker en de Linux-kernel en dat de opdrachten van de gebruiker door deze shell worden verwerkt. Wanneer een gebruiker een opdracht geeft, kloont de shell zichzelf om de opdracht te kunnen uitvoeren. Een dergelijke kloon wordt een **kindproces** genoemd. De relatie tussen de shell en de kloon (*de subshell*) wordt een **ouder/kind-relatie** genoemd. Als de opdracht is overgedragen aan het kindproces (*de subshell*), herstelt het ouderproces (*de shell*) zich en wordt er gewacht op de volgende taak.

Een proces is een activiteit die in het geheugen wordt uitgevoerd om de acties uit te voeren die door middel van een opdracht zijn aangeroepen. Opdrachten kunnen door de gebruiker worden getypt op de opdrachtregel, ze kunnen worden opgenomen in een shellscript of ze kunnen het resultaat zijn van een ander proces dat in het geheugen wordt uitgevoerd (bijvoorbeeld de grafische bureaublad interface). Wanneer de taak van het proces is uitgevoerd, wordt het proces uit het geheugen verwijderd, zodat de CPU-tijd voor andere processen kan worden gebruikt. Op een normaal systeem kunnen zo'n 50 processen tegelijk worden uitgevoerd, waarbij elk proces een unieke ID wordt toegekend.

Soorten processen:

Processen kunnen op basis van de gebruiker en de interactie met de gebruiker, worden verdeeld in drie categorieën:

- 1) Interactieve processen zijn processen waarbij invoer van de gebruiker vereist is. Een voorbeeld : een programma dat wordt uitgevoerd zoals een teksteditor.
- 2) Batchprocessen zijn processen die automatisch worden uitgevoerd, zonder interactie met de gebruiker. Een voorbeeld : een script waarmee op een specifiek tijdstip een reservekopie van uw homedirectory wordt gemaakt.
- 3) Daemonprocessen zijn processen die op de achtergrond worden uitgevoerd en die services leveren aan gebruikers en toepassingen. Sommige daemonprocessen worden geactiveerd bij het opstarten, zoals het proces (*lpd*) dat ervoor zorgt dat bestanden naar de printer worden verzonden of de PPP-daemon die wordt geactiveerd wanneer u een PPP-verbinding maakt met het internet.

De opdracht *ps* (= process status):

Met deze opdracht kunt u informatie opvragen over de processen die op het systeem worden uitgevoerd. Deze heeft de volgende syntaxis :

ps -opties

Oefening :

- 1) Meld u aan met **uw gebruikersaccount**.
- 2) Typ *ps* op de opdrachtregel en druk op *Enter*. De processen die momenteel actief zijn worden getoond. De uitvoer ziet er waarschijnlijk ongeveer als volgt uit:

```
[jo@Examen jo]$ ps
PID  TTY  TIME      CMD
2344  tty1 00:00:00  bash
2423  tty1 00:00:00  ps
[jo@Examen jo]$
```

- Het veld **PID** bevat de proces-ID. U hebt dit nummer nodig als u een afzonderlijk proces wilt beheren.
 - Het veld **TTY** bevat informatie over de terminal waarmee het proces is uitgevoerd. Aan elke terminal wordt een uniek TTY-getal toegewezen.
 - Het veld **TIME** geeft aan hoeveel CPU-tijd voor het proces is gebruikt.
 - Het veld **CMD** bevat informatie over de opdracht die, of het programma dat, door de gebruiker is uitgevoerd. Het voorbeeld geeft aan dat de shell *bash* actief is, en dat de uitvoer is weergegeven via de opdracht *ps*.
- 3) Start het programma *cat* op een andere terminal en vraag opnieuw de actieve processen op.

Typ *ps -a* en druk op *Enter*. Merk op dat er een proces is bijgekomen.

- 4) Indien een programma blijft hangen, of u bijvoorbeeld niet weet hoe u het moet afsluiten, gebruikt dan de opdracht ***kill***.

Typ bijvoorbeeld *kill proces-ID-cat* (vervang *proces-ID-cat* door de proces-ID van *cat* uit oef-3) en druk op *Enter*.

Ga na of *cat* werd afgesloten.

- 5) Lees de man pages van *ps* en probeer de opdracht *ps* te gebruiken met opties.

Opmerking : Soms zijn processen met *kill proces-ID* niet weg te krijgen. In die gevallen kan je het vlaggetje **-9** meegeven, dat aangeeft dat het proces gestopt **moet** worden, als het proces zelf niet wil stoppen, wordt dat door het besturingssysteem gedaan.

16 Het geheugengebruik weergeven :

Een van de grote voordelen van Linux is dat de computerbronnen optimaal worden benut. Een specifiek gedeelte van de vaste schijf (de ruimte voor *wisselbestanden*) wordt door Linux gebruikt om de beschikbare hoeveelheid RAM te vergroten.

Omdat Linux-processen worden uitgevoerd in het RAM-geheugen, is het belangrijk dat u het geheugengebruik regelmatig controleert. U geeft informatie over het fysieke geheugengebruik (RAM) en het virtuele geheugengebruik (de wisselbestanden) weer met de opdracht *free*.

Oefening :

- 1) Geef informatie weer over het geheugengebruik. Typ *free* en druk op *Enter*. De uitvoer ziet er ongeveer als volgt uit:

```
[root@Examen root]# free
              total        used         free       shared    buffers     cached
Mem:   79252      77720      1532      49972         6292      28520
-/+ buffers/cache         42908      36344
Swap: 145112           16    145096
```

- 2) De opdracht *free* geeft u het geheugen in kilobytes weer. Typ *free -m* en druk op *Enter* om deze informatie te tonen in megabytes.
- 3) Om weer te geven welke systeembronnen door de processen gebruikt worden, gebruikt u het hulpprogramma *top*. Dit programma is voorzien van een interactieve interface waarmee u processen kunt beheren. U kunt bijvoorbeeld de prioriteit van een proces aanpassen of een proces beëindigen. Het programma *top* neemt behoorlijk wat bronnen in beslag. Wees dus niet verbaasd als het programma zelf boven in de lijst verschijnt. Start het programma *top*. Typ *top* en druk op *Enter*.
- 4) Sorteert de processen op basis van de gebruikte hoeveelheid geheugen. Typ *M*. Op de opdrachtregel van *top* verschijnt het bericht *Sort by memory usage*. Het proces dat de meeste geheugenruimte gebruikt verschijnt bovenaan in de lijst.
- 5) Sorteert de processen op basis van de gebruikte hoeveelheid CPU. Typ *P*.
- 6) Log aan op een andere terminal en start het programma *top*.
- 7) Ga terug naar *tty1*. Het proces *top* op de andere terminal is bovenaan zichtbaar.

- 8) Beëindig een proces. Typ *k*. Het bericht *PID to kill* verschijnt. Typ nu de PID van het programma *top* dat draait op de andere terminal. Er wordt nu gevraagd of u zeker weet dat u het proces wilt beëindigen. Hiertoe verschijnt het bericht *Kill PID ???? with signal [15]*. Druk op *Enter* om het proces te beëindigen.

Controleer of dit is gebeurd.

- 9) Indien u een lijst wenst met mogelijk opties typt u *h* op de opdrachtregel van *top*.
- 10) Sluit *top* af. Typ *q*.

17 Schijfruimte beheren:

Wanneer u een aantal bestanden hebt gemaakt en toepassingen hebt geïnstalleerd, merkt u misschien dat de beschikbare schijfruimte snel afneemt. Daarom moet u zo nu en dan controleren of er nog voldoende vrije ruimte aanwezig is.

Het bestandssysteem is verdeeld in partities en het is belangrijk dat u de namen van alle partities weet. Met de opdracht *mount* geeft u informatie weer over de beschikbare bestandssystemen. Typ *mount* en druk *Enter*. De uitvoer ziet er ongeveer zo uit:

```
[root@Examen root]# mount
/dev/hda1 on / type ext2 (rw)
proc on /proc type proc (rw)
/dev/hda3 on /home type ext2 (rw)
```

U ziet de apparaatnaam van elk bestandssysteem dat is gekoppeld, gevolgd door het koppelpunt, het type bestandssysteem en de toegangsmachtigingen.

In bovenstaande listing bevindt het eerste bestandssysteem (waaraan de rootpartitie is gekoppeld) zich op locatie */dev/hda1*. Het gaat daarbij om het uitgebreide bestandssysteem van Linux (*ext2*) met lees- en schrijftoegang.

Met de opdracht *df* (= disk filesystem) geeft u de volgende informatie weer voor elk bestandssysteem dat is gekoppeld :

- beschikbare hoeveelheid (vrije) schijfruimte,
- de naam,
- de grootte van de partitie,
- de hoeveelheid gebruikte en beschikbare ruimte,
- het percentage gebruikte ruimte en het koppelpunt.

Oefening :

- 1) Geef informatie van elk van de gekoppelde bestandssystemen weer.
Typ *df* en druk op *Enter*. De uitvoer ziet er ongeveer als volgt uit:

<i>Filesystem</i>	<i>Size</i>	<i>Used</i>	<i>Avail</i>	<i>Use%</i>	<i>Mounted on</i>
<i>/dev/hda1</i>	<i>1961M</i>	<i>1376M</i>	<i>475M</i>	<i>74%</i>	<i>/</i>
<i>/dev/hda3</i>	<i>988M</i>	<i>4M</i>	<i>884M</i>	<i>0%</i>	<i>/home</i>

- 2) Geef informatie weer over de schijfruimte van het apparaat waaraan de rootpartitie */* is gekoppeld. Typ *df /dev/hdax* (vervang *x* → zie oef-1) en druk op *Enter*.
- 3) Geef informatie weer over de schijfruimte op het koppelpunt */home*.
Typ *df /home* en druk op *Enter*.
- 4) U kunt de opdracht *df* combineren met verschillende opties.

Typ *df -k* om de totale schijfruimte weer te geven in kilobytes (optie *-m* voor megabytes).

Met de opdracht *du* (= disk usage) geeft u de hoeveelheid schijfruimte weer die een directory binnen het bestandssysteem (of subdirectory van deze directory) in beslag neemt.

Oefening :

- 1) Typ *du /home/jo* en druk op *Enter* om te weten hoeveel schijfruimte door een bepaalde gebruiker in beslag wordt genomen.
- 2) Geef een lijst weer met alle directory's op het systeem, gesorteerd op grootte.

Typ hiervoor *du / | sort -rn* en druk op *Enter*. Als de lijst te lang is, kunt u de uitvoer bekijken in de pager *less*. Typ *du / | sort -rn | less* en druk op *Enter*.

18 Werken met diskette- en cd-romstations:

In Microsoft Windows wordt het diskettestation aangeduid met **A:**, terwijl het in Linux de naam `/mnt/floppy` heeft. Het cd-romstation heet niet **D:** of **E:**, maar `/mnt/cdrom`. Bovendien wordt in Linux een apparaatbestand aan het station gekoppeld. Het **apparaatbestand** voor de besturing van het diskettestation heet waarschijnlijk `/dev/fd0`, en dat voor het cd-romstation `/dev/cdrom`.

Normaal zijn de directory `/mnt` en de bijbehorende subdirectory's automatisch op het systeem ingesteld. Als dat niet het geval is, kunt u deze directory's met de opdracht `mkdir` aan het systeem toevoegen.

Als u een diskette met bestanden in het diskettestation plaatst, dan wordt de inhoud hiervan automatisch weergegeven in de directory `/mnt/floppy`. Indien dit niet het geval is, moet u het diskettestation zelf **koppelen (= mounten)** om deze te kunnen lezen.

Sommige Linux-distributies bieden de mogelijkheid tot een **supermount** van diskette- en cd-romstation bij het opstarten van het systeem. Hierdoor zijn deze automatisch toegankelijk.

Wanneer u werkt op een systeem dat de supermountvoorziening ondersteunt en een bestand verplaatst naar de directory `/mnt/floppy`, wordt het bestand automatisch op de diskette opgeslagen. U kunt de diskette verwijderen zodra het lampje van het diskettestation uitgaat.

Als uw systeem de supermountvoorziening **niet** ondersteunt, moet u het diskettestation koppelen nadat u een diskette hebt geplaatst. Daarna kunt u de diskette gaan gebruiken. Voordat u de diskette kunt verwijderen, moet u het station ontkoppelen.

Als u een disktestation wilt gebruiken dat **niet** automatisch wordt gekoppeld, doet u het volgende:

- 1) Bepaal het **blokapparaatadres** van het disktestation.

Typ *dmesg | less* en druk op *Enter*. Zoek naar de regel die er als volgt uitziet:

```
Floppy drive (s) : fd0 is 1.44M
```

Dit betekent dat het disktestation zich bevindt op de locatie */dev/fd0*. Zoek ook, indien aanwezig, naar de regels die het cd-romstation aanduiden, bijvoorbeeld:

```
Hdd: ATAPI 24X CD_ROM drive, 128kB Cache  
Uniform CDROM driver Revision 2.56
```

- 2) Plaats een diskette in het disktestation.
- 3) Koppel het station.

Typ *mount /dev/fd0 /mnt/floppy* (vervang */dev/fd0* indien nodig door de apparaataanduiding die u bij stap-1 hebt opgevraagd) en druk op *Enter*. Het lampje van het station gaat enkele seconden oplichten.

- 4) Kopieer een bestand naar de diskette.

Typ *cp bestandsnaam /mnt/floppy* en druk op *Enter*. Het lampje van het station licht op terwijl het bestand naar de diskette wordt gekopieerd.

- 5) Ontkoppel het disktestation.

Typ *umount /dev/fd0* en druk op *Enter*.

Opmerking : Het ontkoppelen kan alleen indien de actieve directory verschillend is van */mnt/floppy*, anders bekomt u de boodschap *device is busy*.

- 6) Verwijder de diskette uit het station. **Het is belangrijk dat u het station ontkoppelt voordat u de diskette verwijdert !**

Een cd-romstation kunt u op dezelfde manier koppelen en ontkoppelen.

Het pakket *mtools*:

Op uw systeem is misschien een pakket met de naam *mtools* geïnstalleerd. Het pakket *mtools* bevat een verzameling opdrachten die ervoor zorgen dat u de directory's en bestanden op diskettes op dezelfde manier kunt gebruiken als op een **DOS-systeem**. Bovendien hoeft u de diskette niet te benaderen via het Linux-bestandssysteem (*/mnt/floppy*), maar kunt u de oude vertrouwde aanduiding *A:* gebruiken.

Als u *mtools* gebruikt, hoeft u een diskette niet te koppelen voordat u deze kunt gebruiken. Vraag eventueel eerst de man-pages van *mtools* op.

Oefening :

- 1) Plaats de diskette in het diskettestation.
- 2) Formateer de diskette.

Typ *mformat a:* en druk op *Enter*. Mogelijk verschijnt de volgende foutmelding:

```
Mformat: Non-removable media is not supported
(You must tell the complete geometry of the disk,
either in /etc/mtools.conf or on the command line)
```

In dat geval typt u *mformat -s 18 -h 2 -t 80 a:* en druk op *Enter*.

Met deze opdracht, waarmee u een diskette van 1,44 Mb formatteert, geeft u aan dat de diskette 18 sectoren, 2 koppen en 80 sporen bevat. Deze informatie is beschikbaar in het bestand */etc/fdprm* (= floppy disk parameters)

- 3) Voorzie de diskette van het label *test*.

Typ *mlabel a: test* en druk op *Enter*.

- 4) Bekijk de inhoud van de diskette.

Typ *mdir a:* en druk op *Enter*.

- 5) Kopieer een bestand naar de diskette. Ga naar de directory met het bestand dat u wilt kopiëren. Typ *mcopy bestandsnaam a:* en druk op *Enter*.
- 6) Typ opnieuw *mdir a:* en druk op *Enter* om te controleren of het bestand naar de diskette is gekopieerd.