

Kennismaking



Onze eerste kennismaking met het systeem

DOELSTELLING

Aan het eind van dit hoofdstuk is de cursist bekend met:

- Multi-user, multi-tasking
- Inloggen, Passwords
- De shell
- profile bestanden
- De commandoregel en enkele commando's
- hulp inschakelen
- Command history, control keys
- Relatie commando's, shell en de kernel

Multi-user, multi-tasking

- Meerdere (multi) gebruikers (users) kunnen meerdere taken (tasks) "tegelijkertijd" uitvoeren.
- Tegenwoordig gemeengoed.

Inloggen

- Waarom inloggen?
- Combinatie van loginnaam en wachtwoord
- Case sensitive!

Wachtwoorden

Een wachtwoord moet:

- Moeilijk te raden zijn
- Relatief makkelijk te onthouden zijn

Systeem kan eisen opleggen, in de trant van:

- Gebruik minimaal 8 posities
- Gebruik tenminste 2 alfabetische karakters
- Gebruik minimaal 1 speciaal teken of numerieke waarde
- Een nieuw wachtwoord moet minstens drie karakters verschillen van het oude
- Wachtwoord "aging"

Oefeningen

Tijd voor oefening: kennismaking 1

De shell: uw commando interpreter

Werking:

- gestart na inloggen
- toont aanwezigheid middels een prompt, bv.:

```
oscar@cws001~$
```

- U tikt een commando gevolgd door <Enter>
- De shell voert uw commando uit
- Commando klaar? Prompt terug.
- Volgend commando kan gegeven worden.

profile bestanden (1/2)

- bij starten leest (*sourced*) de shell zgn. profile bestanden
- bepalen de **shellomgeving** van de gebruiker. Variabelen spelen een grote rol:
 - PS1, PATH, HOME
- verschil *system-wide* en user-specifieke profile bestanden
- welke bestanden gelezen worden verschilt per shell

profile bestanden (2/2)

- ook onderscheid login-shell, interactieve shell, niet-interactieve-shell (bv. in een *shell-script*).
- globaal voor **bash** (maar ook verschillen in distributies..):

| login-shell | interactieve shell | niet-interactieve shell |
|--|--------------------|------------------------------------|
| /etc/profile | /etc/bashrc | bestanden in de variabele BASH_ENV |
| ~/.bash_profile of anders ~/.bash_login of anders ~/.profile | ~/.bashrc | |

- andere shells gebruiken weer andere bestanden
- bestanden *sourcen* weer andere bestanden (!)
- om zeker te zijn: raadpleeg de manpage van uw shell (zie opgaven)!

Verschillende shells (1/2)

Verschillen in shells:

- te gebruiken profile bestanden
- terugroepen van commando's (*command history*)
- (programmeer) syntax
- job control
- filenaam/command *completion*

Verschillende shells (2/2)

Meest voorkomende shell (vooral op Linux): bash.

"Meest gebruiksvriendelijk"

Overige shells:

- De bourne-shell (sh). Altijd aanwezig en zeer basic (geen job-control, geen history). Let wel: op Linux vaak een verwijzing naar bash.
- C-shell (csh). C-achtige syntax, job control.
- Tcsh shell (tcsh). Afgeleid van csh maar betere command history.
- Korn shell (ksh). Mix van Bourne shell (syntax) en csh (job control).

De commandoregel (1/4)

"alles wat je invoert totdat je <Enter> geeft"

Vaak opbouw commandoregel:

commandonaam [optie]... [argument]...

Naam:

- veelal **lowercase** en geen "speciale karakters"
- meestal afgekort:
 - `ls` (list), `cp` (copy), `rm` (remove), `mv` (move), `df` (disk dfree), `du` (disk usage), `ps` (proces status), `ln` (link)
- soms niet:
 - `find`, `date`, `kill`, `more`, `less`

De commandoregel (2/4)

commandonaam [optie]... [argument]...

Opties:

- beïnvloeden het gedrag van een commando.
- beginnen meestal met een hyphen (-).
- zowel los als gegroepeerd:

- `ls -l -r -t <=> ls -lrt`

- veelal ook een long format equivalent:

- `ls -a of ls --all`

- `ls -r of ls --reverse`

vooral in shell scripts (ter bevordering van de leesbaarheid).

De commandoregel (3/4)

commandonaam [optie]... [argument]...

Argumenten:

- waarop het commando actie(s) uitvoert.
- veelal één of meerdere filenamen

Enkele voorbeelden:

- `cp /etc/passwd /etc/passwd.bak`

- `ls -l ~`

- `man man`

De commandoregel (4/4)

commandonaam [optie]... [argument]...

- commandonaam, opties en argumenten worden gescheiden door witruimte.
- De shell **leest en interpreteert altijd eerst de hele commandoregel** voordat een commando uitgevoerd wordt (later meer!)

Enkele commando's

- inhoud van een directory: `ls`
- huidige directory: `pwd`
- inhoud van files bekijken:
`vi, view, cat, less, more, pr`
(meer in de opgaven)

Help (1/2)

- documentatie m.b.t. commando's, files, functies e.d. in de online *man pages*.
 - commando: `man [sectie] "page"`
 - opgedeeld in 9 secties, sectie 1 zijn de user commando's (default)
 - beknopt, **goed overzicht syntax (synopsis) en opties**
 - syntax conventie ([] = optioneel, ... herhaling mogelijk)
 - opgedeeld in headings: NAME, SYNOPSIS, DESCRIPTION, OPTIONS, EXAMPLES, BUGS, FILES, SEE ALSO
 - niet vergeten: `man man`
- `whatis` *woord*
 - zoekt de `whatis` database (man pagina's) voor compleet *woord*. Deze database wordt bijgewerkt door `makewhatis` (soms `mandb`)
- `apropos` (ook wel `man -k`) *string*
 - zoekt de `whatis` database (man pagina's) voor *string*

Help (2/2)

- de online "info pages"
 - commando: `info cat`
 - veelal uitgebreider, navigeerbaar
- `/usr/share/doc`
- onderschat de systeembeheerder "google" niet!

Command history

- Commando's terughalen, evt. wijzigen en opnieuw uitvoeren kan veelal eenvoudig met de pijltjestoetsen.
- Erg handig in bash: zoeken op eerder gegeven *patterns* in je historie met `<Ctrl>-R`

De kernel

- kernel: de kern van het besturingssysteem
- kan als enige communiceren met de **hardware**.
- een programma, net als de shell

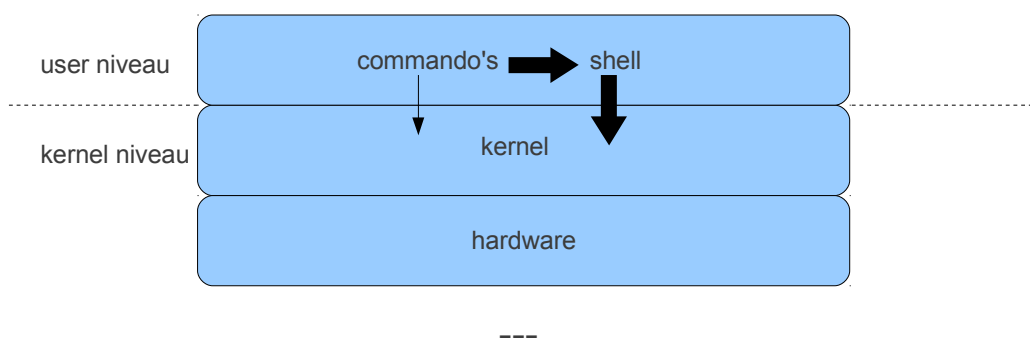
Taken:

- invoer- en uitvoerbeheer (keyboard / terminal)
- **geheugen**beheer
- het afhandelen van **interrupts** (ook foutafhandeling)
- het beheren van **filesystems** (de 'bestandsorganisatie' onder Linux).
- het verdelen van **processor**capaciteit tussen de verschillende processen.

Relatie kernel, shell en commando's

De kernel werkt nauw samen met de shell: als u een commando intypt vraagt de shell aan de kernel om dit commando uit te voeren.

In een plaatje:



Samenvatting (1/2)

- inloggen verplicht
- na inloggen krijgt u een shell
- commando's geeft u aan de shell (welke de kernel weer aan het werk zet)
- commando's hebben veelal opties en/of argumenten (komen terug in de opgaven)
- de shell inspecteert **altijd** eerste de hele commandoregel voordat een commando wordt uitgevoerd (hierover later meer).

Samenvatting (2/2)

commando's, de **shell** en de **kernel** zijn drie belangrijke aspecten van een Linux systeem.

Verder in deze cursus komen meer belangrijke aspecten aan bod waaronder:

- het linux **file-systeem**: de organisatie van bestanden
- de **bestanden** zelf
- **processen**

Oefeningen

Tijd voor oefening: kennismaking 2
