

Labo Linux 1

Haroen Viaene

Labo 2: BASISOPDRACHTEN 1

Oefening 1

1. `date`: geeft de datum
 - `wo sep 24 14:29:09 CEST 2014`
2. `pwd`: geeft de directory die actief is
 - `/home/student`
3. `ls`: geeft een lijst van alle items in die directory
 - `-a`: toont ook onzichtbare bestanden
 - `-al`: idem+l: long-listing format
 - `-R`: toont alle subdirectories recursief (toont ook de bestanden in de subdirectories)
4. `cat .bashrc`: drukt invoer uit in uitvoer
 - toont de inhoud van het bestand `.bashrc` (een config-bestand)
5. `mkdir werk`: maakt een map `werk` aan in `student` (de huidige directory)
6. `cd werk`: navigeer naar de subdirectory `/werk/`
7. `cd ..`: navigeer naar de onderliggende (sub-)directory
8. `rm`: verwijder bestanden
 - `-r`: en hun submappen ook (recursief)
 - `werk`: de map `werk` in de huidige directory wordt verwijderd
9. `cp`: kopiëren van een bron naar een bestemming
 - `.bashrc test.txt`: `.bashrc` wordt naar een bestand `test.txt` gekopieerd, in de huidige directory
10. `mv test.txt test1.txt`: hernoem `test.txt` naar `test1.txt`
11. `wc test1.txt`: geeft het aantal regels, woorden en bytes (110, 475, 3392 respectievelijk) en de naam van het bestand.
 - `-l`: geeft het aantal regels
12. `rm test1.txt`: `test1.txt` wordt verwijderd.

Oefening 2

1. dit kopieert de inhoud van `.bashrc` naar het nieuwe bestand `testfile`. Daarna opent het de inhoud van `testfile`.
 - Dit geeft uitgebreide informatie over `testfile` (`-rw-r--r-- 1 student student 3392 okt 1 14:00 testfile`), waarna het verwijderd wordt zodat erna diezelfde informatie niet meer verkrijgbaar is.
2. met het commando `cal october 1995` werd bevestigd dat ik op een maandag geboren ben.
3. Dit doe je door het commando `cd /etc`
4. `ls *`
5. `ls ????`

Oefening 3

1. :
 - `ls`: geeft informatie over de bestanden in de huidige directory
 - `-a`: toon ook de bestanden die verborgen zijn (beginnen met `.`)
 - `-b`: toont newlines als `\n` etc...
 - `-B`: negeert bestanden die eindigen in `~` (meestal backups)
 - `-l`: gebruik het lange formaat voor de bestanden
 - `cp`: kopiëren van bestanden naar een ander bestand.
 - `-b`: maak een backup van de destination-bestanden
 - `-f`: als het destination-bestand niet bereikt kan worden, wordt het verwijderd en opnieuw gemaakt
 - `-n`: overschrijf geen bestanden
 - `-i`: prompt wel of niet te overschrijven
 - `rm`: verwijder een bepaald bestand of directory
 - `-i`: prompt elk van de verwijderingen
 - `-I`: vraag maar één keer voor meer dan 3 bestanden
 - `-v`: leg uit welke bestanden verwijderd worden
 - `-r`: ook subdirectories worden verwijderd
 - `man`: toont handleidingen voor commando's
 - `-d`: debug, toon als er fouten voorkomen
 - `-f`: korte beschrijving
 - `-w`: toon waar de handleiding zich bevindt
 - `-L`: verander de taal
2. `whoami`: toont de actieve gebruiker (student)
 - `who`: toont alle actieve schermen (terminals en spaces)

3. `more`: toont alles wat op één scherm past, scroll naar beneden met enter
 - `less`: toont alles op verschillende spaties, je moet op "q" drukken om een ander commando te doen.
4. gebruik voor de eerste vijf regels van dat bestand `head -n 5 bash.bashrc` en voor de laatste vijf `tail -n 5 bash.bashrc`. De optie `-f` zorgt ervoor dat als het bestand wijzigt dat het scherm ook wijzigt.
5. `mkdir dummy; cd dummy; echo > 1.txt; echo > 2.txt; echo > 3.txt; echo > 4.txt; cd ..; rm -r dummy`

Oefening 4

1. `echo > 1.txt; echo > 2.txt; echo > 3.txt; cat *.txt`
2. `ctrlu` verwijdert de input die nu gegeven wordt (voor de cursor)
3. tijdens sleep kan je geen commando's invoeren. `ctrlc` onderbreekt de slaap
4. `ctrlh` verwijdert het teken voor de cursor
5. `tab` autocomplete je input `cat .x` geeft `cat .X`, want er is geen bestand dat `.x...` heet.

Labo 3: Basisopdrachten 2

Oefening 1

1. `ls -a`
2. de terminal. Ja, door op een bestand te schrijven met `>`
3. `ls -a > lijst.txt`
4. `vim lijst.txt`
5. `cp lijst.txt oudelijst.txt`
6. `ls -a -r > lijst.txt`
7. `cat -b lijst.txt oudelijst.txt > samengesteld.txt`

Oefening 2

1. `pico samengesteld.txt`
2. `cat lijst.txt onbestaand.txt` geeft omdat `onbestaand.txt` niet bestaat:
 - de inhoud van lijst, end dan:
 - `cat: onbestaand: Bestand of map bestaat niet`
3. `cat lijst.txt onbestaand.txt > /dev/null`
4. `cat lijst.txt onbestaand.txt &> alles.txt`
5. `nano alles.txt`
6. `rm alles.txt`

Oefening 3

1. `find / -name hosts`
2. `find *.rot -delete`
3. `find -name '*a*' -o -name '*z*'`

Oefening 4

1. `less /etc/hosts | grep localhost`
2. :

```
cat > boek.txt
David 09 3331375
Rachel 02 5454643
Emily 09 4543535
Daniel 03 7897897
Monic 014 678667
```

`ctrl+d`

Labo 4: Rechten

Oefening 1

1

```
cp -r /etc ~/etc
```

```
cd etc
```

```
pwd
```

```
ls -al
```

2

1. Eerste letter is d
2. Tweede letter is r, vierde x voor user, vijfde en zevende voor group, achtste en tiende voor other
3. Als de w staat bij user, group of other
4. Als r aan staat
5. user is het woord dat naast het getal staat, in dit geval altijd student.
6. de group is het woord dat daarna komt, in dit geval student
7. De rechten van de group worden door het 5de, 6de en 7de karakter bepaald.
8. De rechten van andere gebruikers worden gegeven door het achtste tot tiende karakter.

3

1. `-rw-r--r--` 1 student student 19398 okt 15 13:32 services
 - `-rw-r-----` 1 student student 19398 okt 15 13:32 services
 - Andere gebruikers kunnen nu niet meer lezen in die file
2. `-rwxrwx-wx` 1 student student 19398 okt 15 13:32 services
 - Iedereen kan nu schrijven en executen, enkel andere kunnen niet meer lezen, de rest wel nog.
3. `-rwxrw--wx` 1 student student 19398 okt 15 13:32 services
 - De groep waar student deel van uit maakt (die ook student heet), kan nu niet meer executen. De eigenaar van het bestand (student) kan dit wel nog.

4

1. `mkdir brieven; mkdir spelen; mkdir metalen; mkdir nonsens`
2. `echo > ./brieven/brief1; echo > ./spelen/leo; echo > ./metalen/goud`

Oefening 2

1. `cd metalen; ls -l`
 - `-rw-r--r--`: het is een file, leesbaar door iedereen, en enkel door de eigenaar beschrijfbaar.
2. `chmod g-r: controle: -rw----r--`, juist.
3. `chmod u+x`: het bestandje goud krijgt nu een groene kleur.
4. `drwxr-xr-x`: wil zeggen dat de eigenaar (ik) alle rechten heeft.
 - `chmod u-r spelen`
 - De inhoud van de directory is niet meer zichtbaar, `ls -l spelen` geeft als uitvoer:
`ls: kan de map spelen niet openen: Toegang geweigerd`
 - Het bestand leo is nog openbaar dankzij bijv. `pico /spelen/leo`
5. `chmod u+r spelen`

Oefening 3

1. `chmod u-w spelen; echo > ./spelen/poging`
 - `bash: ./spelen/poging: Toegang geweigerd`
 - Je hebt inderdaad onvoldoende rechten om dit bestand te maken.
2. `pico ./spelen/leo`
 - na het wijzigen van dit bestand, kan het zonder problemen opgeslagen worden.
3. `rm ./spelen/leo`
 - `rm: kan ./spelen/leo niet verwijderen: Toegang geweigerd`
 - er zijn ook hiervoor te weinig rechten
4. `x` voor een directory wil zeggen dat je er bestanden in kan uitvoeren en lezen. Dit wil

zeggen dat we met de volgende command een dir aanmaken die niet navigeerbaar of schrijfbaar is, maar dat de bestanden die er in zitten wel lees- en schrijfbaar zijn

- `mkdir -m u-rw,u+x test`
- We maken eventjes de dir terug lees- en schrijfbaar zodat er een paar bestanden in kunnen met
- `chmod u+rw test; echo > foo; echo > bar`
- Hierna wordt de permissie terug zoals eerder gezet:
- `chmod u-rw test`
- Nu kunnen de bestandjes foo en bar geopend en veranderd worden.

5. `chown`: veranderen van de eigenaar van 'resolv.conf': Bewerking niet toegestaan

- `chown` is enkel als root toegestaan (superuser), en student is geen superuser.

Oefening 4

1. `touch test1.txt; vi test1.txt`
2. vul de tekst in
3. `esc::enter`
4. `escdd`

Labo 5: Shell scripts

Werken met vi

`vi .exrc`

voeg toe:

- `syntax on`
- `set number`

Oefeningen

`vi eerste.sh`

```
#!/bin/bash
pwd
date
```

`vi tweede.sh`

```
#!/bin/bash
# geeft de meegegeven argumenten op het scherm mbv een for-lus
for i in $@
do
```

```
    echo $i
done
```

vi derde.sh

```
#!/bin/bash
for i in $@
do
    echo "Je zal nu $i verwijderen. j/n"
    read ja
    if [[ $ja = "j" ]]; then
        rm $i
        echo "$i verwijderd"
    else
        echo "$i niet verwijderd"
    fi
done
```

alternatieve oplossing door Nicolas Henrotte

```
#!/bin/bash
echo "Het bestand $1 verwijderen?"
read jaofnee
case $jaofnee in
ja | Ja | JA)
    rm $1
    echo "Bestand $1 verwijderd."
    ;;
nee | Nee | NEE)
    echo "Bestand $1 niet verwijderd."
    ;;
esac
```

vi vierde.sh

```
#!/bin/bash
for i in $#@
do
    chmod 755 $i
    chown jan $i
done
```

vi getal.sh

```
#!/bin/bash
#print een getal als het tussen 0 en 9 ligt. Is het iets anders wordt het script be
ëindigd.
clear
i=1
while [ $i -eq 1 ]
```

```

do
    echo "geef een getal in."
    read getal
    if [ $getal -le 9 ]&&[ $getal -ge 0 ]
    then
        echo "Je getal $getal is tussen 0 en 9"
    else
        expr $i + 1
        exit
    fi
done

```

Labo 6: Gebruikersaccounts

Gebruikers

1. toevoegen van een account

- via /etc/passwd aan te passen
 - vi /etc/passwd
 - Test::600:600:test:/home/test:/bin/bash
- door adduser testerig
 - vul volledige naam etc in
 - er wordt ook voor kamernummer, telefoonnummer etc. gevraagd, dit kan leeg gelaten worden.
- door useradd testNogMeer
 - er kunnen opties meegegeven worden om een wachtwoord, volledige naam etc. in te vullen.
- admintool

2. Bij het automatisch aanmaken van een bestand:

- in het shadow-bestand zit een input voor
 - testerig
 - wachtwoord geëncrypteerd
 - met 0/99999 voor lifetime
 - en 7 dagen voor waarschuwing
 - testNogMeer
 - ! als wachtwoord (kan dus niet inloggen, om dit aan te passen: passwd testNogMeer, dan nieuw wachtwoord)
 - 0:99999:7 voor beveiliging
- in het passwd-bestand zit een input voor
 - testerig
 - group en user zijn 1001

- volledige naam
 - /home/testerig
 - /bin/bash
 - testNogMeer
 - 1002:1002
 - geen volledige naam
- 3.
4. Wordt aangepast in /etc/passwd/ (hij heeft nu een c-shell ipv bash)
- apt-get install csh
 - su - piet
 - %
5. pas /etc/passwd aan, of userdel, rmuser ...
- de input wordt verwijderd uit /etc/passwd
6. de shadow-entry blijft bestaan.

Groepen

1. addgroup gebruikers1; addgroup gebruikers2; addgroup gebruikers3 (dit wordt resp. 1003, 1004 en 1005)
2. groupdel gebruikers3
3. adduser piet en adduser karel
4. vi passwd het getalletje bij group bij beide veranderen naar 1003
5. usermod -G gebruikers2 karel of toevoegen aan /etc/group
6. su karel; touch test1; chmod 750 test1
7. chgrp gebruikers2 test1
8. su piet; vi test1 test1 kan niet geopend worden.
9. su karel karel heeft volledige rechten.

Nutcracker

nutcracker-1.9/nutcracker /etc/shadow /sbin/nutcracker-1.9/words