

Waarom is commandline beter?

Er zijn meer mogelijkheden voor het beheer dan een GUI, toetsen zijn sneller dan een muis
het gebruikt ook minder systeembronnen, servers draaien geen GUI en Ubuntu hebben geen GUI
Het kan scripts uitvoeren om herhalende taken voor onder andere systeembeheer uit te voeren bv. backups

Hoe ontstond Linux?

Unix is ontstaan in 1970 door At&T Bell Labs met de filosofie van minimalisme en modulariteit:

Elk programma doet maar 1 ding, maar is er wel uiterst goed in

Verschillende programma's werken samen om complexe taken uit te voeren

Unix splitst op in 2 grote families en andere versies

- BSD familie (⇒ MacOS X)
- System III en V familie
- XenixOS (deels BSD en Unix)
- GNU en Minix → Linux

GNU/Linux OS is een combinatie van de Linux kernel en GNU software

Kenmerken van Linux

- Multi-user en -tasking
- processor-onafhankelijk
- Open Source

Free as in free speech, not as in free beer ~ Freedom to:

- Run the program for any purpose
- Study how the program works, and adapt it to you needs
- Redistribute copies so you can help your neighbourhood
- Improve the program, and release your improvements to the public so the whole community benefits

!! De grootste, belangrijkste en indrukwekkendste apparaten werken op een aangepaste Linux-distributie

!! Linux wordt voor ongeveer 80% van de webhosting gebruikt

Open Source maakt veel keuzes en combinaties van GNU / Linux mogelijk

Onderdelen van een Linux Distributie

- Kernel: hardware-ondersteuning en multitasking (!! meest belangrijke onderdeel !!)
- Shell: meestal bash
- GNU Utilities: Open Software Tools
- X-window systeem: X-Server voor grafisch interface
- Window managers en desktop environments zoals Gnome, Xfce, KDE, ...
- Netwerkfunctionaliteit

Command prompt in OS	
Voor windows druk op vergrootglas en type "cmd" of "powershell" DOS commando's: <i>DIR</i> , <i>DIR /P</i> , <i>CD</i> , ...	Voor MacOS X en Linux zoek in menu naar terminal !! gebruikt geen drive letters zoals windows ("c:\\") /=root /home=home directories /media=externe HD /tmp=ruimte die bij reboot wordt gewist /dev=hardware /etc=configuratie systeem /bin, /usr, /lib=software

Commando's voor het navigeren door directories

- pwd : "print work directory" print het absolute pad af waar je momenteel in zit
- ls : bekijkt wat er binnen een directory staat
- cd : navigeert naar een bepaalde directory (. .. ~ - / [niks] /tmp)

relatief pad (cd ../temp) vs absoluut pad (cd /temp)

gebruikers in Linux

- Root of privileged user (userid=0 + heeft shell en homedir)
- Service account (op Ubuntu userid tussen 1 en 1000 + geen shell login)
- gewone gebruiker of unprivileged user (userid ≥ 1000 + shell en homedir)



commando's:

- `sudo` : voert een actie uit als de root
- `sudo su -` : voert een actie uit als de root
- `sudo su - [user]`: voert een actie uit als een anderegebruiker
- `whoami` : geeft weer welke gebruiker je nu bent
- `touch` : maakt een bestand aan of refreshed de modified-tijd

Een commando bestaat uit 3 delen: de naam, de opties, de argumenten

Commandogeschiedenis

Te gebruiken met Up en Down en is een lijst van de voorgegane opdrachten

`!nummer` voert de opdracht uit die op deze regelnummer is uitgevoerd

`!string` voert de laatste opdracht die deze string bevat uit

`!!` voert vorige opdracht uit

`CTRL + R` zoekt in geschiedenis

`"history -c"` wist de hele lijst

Hulp zoeken

man nummer string (-k)

Standard Streams

kan lezen van Standard Input (STDIN)

kan schrijven naar Standard Output (STDOUT)

kan schrijven naar Standard Error (STDERROR)

Standard Stream Redirection

van terminal:

- `stdin`: keyboard
- `stdout`: scherm (filedescriptor 1)
- `stderr`: scherm (filedescriptor 2)

naar een bestand: `< > << >>`

- `>` = nieuw bestand (overschrijven)
- `>>` = toevoegen aan bestand
- `2>` = de stderr ergens anders heen leiden bv. `2> /dev/null`
- `1>` = de stdout ergens anders heen leiden bv. `1> output_bestandje.txt`

Piping met "|"

koppelt de stdout naar de stdin

Variabelen

commando's die een overzicht van alle variabelen geven: env en echo \$PATH

definiëren:

- NAAM=waarde ⇒ echo \$NAAM
- NAAM=\home\dries ⇒ echo \$NAAM

Editors

handig voor scripting en omdat alle configuratie- en logbestanden tekstbestanden zijn
favorieten linux editor: Vim

Vi: niet echt gebruiksvriendelijk, maar wel snel, altijd beschikbaar, en werkt in de terminal

Vi

Command mode:

gebruikt voor bewegen, verlaten, copy past, ...

ESC: brengt je naar de command mode

bestand sluiten

:q! ⇒ quit without saving	:wq! ⇒ save + quit
---------------------------	--------------------

tekst verwijderen

x: verwijder karakter	dd : verwijder lijn	D: verwijder tot einde van lijn	d: verwijder geselecteerde sectie
-----------------------	---------------------	---------------------------------	-----------------------------------

cursor verplaatsen

h: ga naar links	l: ga naar rechts	k: ga naar boven	j: ga naar beneden
G: ga naar laatste lijn	xG: ga naar xde lijn	gg: ga naar eerste lijn	/string: ga naar eerste lijn waar string voorkomt
?string: ga naar laatste lijn waar string voorkomt	n: herhaal laatste search	N: vorige laatste search	/RegEx: regular expression zoeken

Copy-paste

v: start van blokselectie	y: kopieer selectie	p: paste selectie
d: verwijder selectie	yy: kopieer lijn	u: undo
CTRL + R: redo	:%s/oud/nieuw/g: zoek "oud" en vervang dit met "nieuw"	

Insert mode:

tekst invoegen

i: insert voor de cursor	I: insert voor de lijn	a: append na de cursor	A: append na de lijn
o: open nieuwe lijn na de lijn waar je nu op staat	O: open nieuwe lijn voor de lijn waar je nu op staat	r: vervang 1 karakter	R: vervang karakters

vi alternatieven: nano, pico, gedit, midnight, geany

Echo

- echo "tekst" : print tekst
- echo -e "tekst" : print tekst, rekening houdend met escape karakters
- echo -n "tekst" : print tekst en maakt geen 'new line' aan

Bekijken van (een deel van) een bestand

- `cat` : print de tekst in een bestand af
- `tac` : print de tekst in een bestand in de omgekeerde volgorde af
- `head` : geeft eerste 10 regels van een tekst
- `head -nx` : geeft eerste x regels van een tekst
- `tail` : geeft laatste 10 regels van een tekst
- `tail -nx` : geeft laatste x regels van een tekst
- `tail -f` : laat regels zien die aan een bestand toegevoegd worden

tellen en nummeren

- `wc:` "lijnen", woorden, chars"
- `wc -l:` aantal lijnen
- `nl:` voegt lijnnummering toe

filteren en sorteren

- `cut:` selecteert een kolom uit csv-bestand, of een andere delimiter
 - `-d; -f1` : geeft ; door als delimiter en vraagt de eerste selectie op
 - `-c 1-5` : selecteert de eerste 5 karakters
- `sort:` sorteert een selectie
 - `-f` : case insensitive
 - `-n` : numeriek ipv alfabetische
 - `-h` : sorteren op grootte
- `uniq:` elimineert identieke regels (!! enkel na sort te gebruiken)
 - `--repeated:` toont enkel de herhalende regels
- `tr:` translate ⇒ vervangt karakters
 - `-d` : verwijder karakters
 - `-s` : vervangt dezelfde karakters achter elkaar met maar 1 keer dat karakter (aaa ⇒ a)

Reguliere expressie

- `grep:` zoekt voor specifieke term
 - `-i` : case insensitive
 - `-l` : lijst van alle files
 - `-s` : silent, geen foutmeldingen
 - `-F` : string ipv regex zoeken, en diens regel afdrukken
 - `-E` : regex tussen " zoeken, en diens regel afdrukken
- `sed:` zoals grep, maar kan regel per regel een opdracht automatiseren
 - `s[directory] g[directory]` : find & replace
 - `-i` : find, replace en save

regEx karakters

. = willekeurig karakter	[a-d] = a,b,c of d [^0-9] = geen nummer	[aeiou] = letter tussen haken [^aqw] = geen van deze letters	gr(a e)y = grey of gray
\. = expliciet teken	^x = regel dat start met x x\$ = regel dat stopt met x	\<k = woord dat met k begint k\> = woord dat met k eindigt	

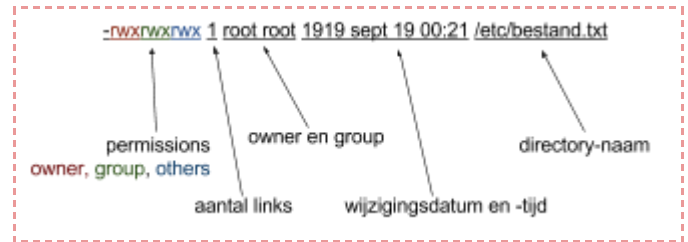
Herhalingen

+ = 1 of meer herhalingen	? = 0 of 1 herhaling	* = 0 of meer herhalingen	{5} = exact 5 herhalingen
---------------------------	----------------------	---------------------------	---------------------------

Bestandencommando's

ls

- -a : toont ook de hidden files (starten met .)
- -l : long list toont extra informatie
- -S : sort by size
- -t : sort by modifytime
- -r : reverse sort
- -R : recursief
- -d : directory info only
 - r (read): openen en lezen van bestand + lijst van files en mappen in dirs zien
 - w (write): wijzigen en deleten van bestand + mappen en files aan kunnen maken
 - x (execute): uitvoeren van programma's of scripts + toegang tot dir met "cd"



Directory maken / verwijderen

- `mkdir dir1` : maak een nieuwe directory aan
- `mkdir -p /dir1/dir2` : maak een nieuwe directory-path aan
- `rmdir dir` : verwijdert lege directory
- `rm` : verwijdert directory
 - -r : verwijdert directory en alle subdirectories recursief
 - -rf : forced het recursief verwijderen indien er geen write-permission is

Directory / bestand kopiëren of verplaatsen

- `cp dir1 dir2` : kopieert directory en plaatst deze in andere directory
 - -r : recursief kopiëren
 - -p : behouden van alle attributen, geen modifyupdates, ...
 - -u : bestanden niet overschrijven
- `mv dir1 dir2` : verplaatst dir1 naar dir2
- `mv bestand Bestand` : hernoemt bestand

file globbing

werkt zoals een regular expression, maar met minder regels

* = ongekende hoeveelheid willekeurige tekens	? = willekeurig teken	[aeiou] = een teken binnen haken
"" zal indien het alleen staat alle bestanden geven ⇒ *.dD][oO][cC] zal alle bestanden die eindigen met		

`dd` : voor kopiëren van devices

`dd if=/dev/... of=[file] bs=[#bytes] K/M/G/T Count=1`

(zie de devices bij /dev/)

`touch`: wijzigt de access time of maakt een bestand aan als deze niet bestaat

`file`: geeft info over het bestand

tar of tape archiver

maken/uitpakken van backups of compressiemappen

- `tar cvf naam.tar directory` : maakt tar bestand aan
- `tar tvf naam.tar` : laat contents van tar bestand zien
- `tar xvf naam.tar` : extract tar bestand
- `tar xvf naam.tar -C directory` : verandert directory
- `tar cvzf naam.tgz directory` : maakt gzip bestand aan
- `tar cvjf naam.tar.bz2 directory` : maakt bzip2 bestand aan

gzip	bzip
~windows zip	hogere compressie
gunzip decompressie	bunzip2 decompressie

Links

verwijzingen naar een ander bestand (shortcuts en mklink-command in windows)

In original link : maakt hardlink die direct naar locatie op de schijf verwijst \Rightarrow 2 verwijzingen naar bestand

- `-s` : maakt symbolic link die verwijst naar verwijzing van locatie \Rightarrow 1 verwijzing naar bestand

Symbolic link:

- Kan naar een ander medium (disk, partitie, ...) verwijzen
- wordt ongeldig wanneer origineel wordt verwijderd
- Kan naar een directory verwijzen
- Werkt iets langzamer omdat dit een indirecte link is

Hard link:

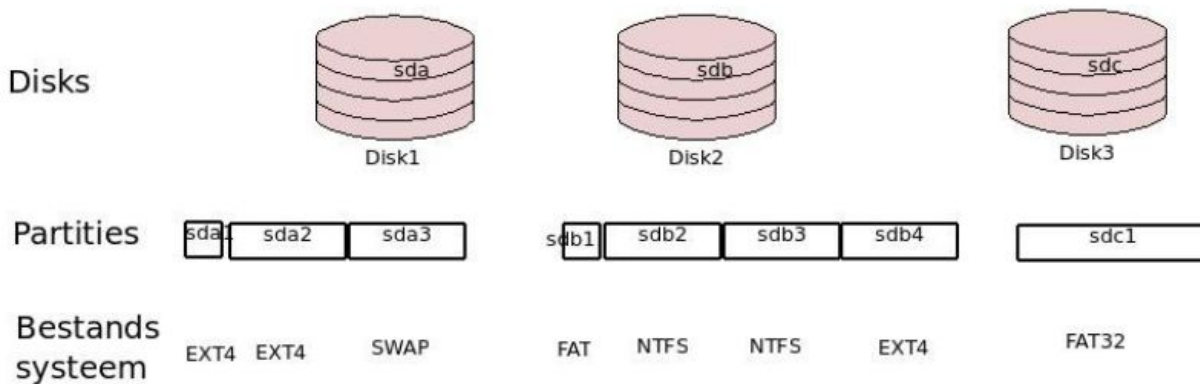
- Kan enkel naar een bestand verwijzen en is even snel als het origineel
- Link en origineel hebben geen onderscheid en werkt nog wanneer origineel wordt verwijderd/verplaatst

Bestanden zoeken

- `locate` : zoek file
- `updatedb` : db van locate updates
- `which` : zoek in pad
- `whereis` : zoek in pad + documentatie
- `type` : geeft type van commando terug
- `find` : voor het zoeken van een bestand en hier eventueel een opdracht op uitvoeren
 - `-name "name"` : zoek op naam
 - `-mmin (+-)n` : zoekt files/dirs waar modifytime (kleiner/groter is dan) n minuten (oud is)
 - `-mtime (+-)d` : zoekt files/dirs waar modifytime (groter/kleiner is dan) d dagen (oud is)
 - `-amin (+-)n` : zoekt files/dirs waar accesstime (kleiner/groter is dan) n minuten (oud is)
 - `-atime (+-)d` : zoekt files/dirs waar accesstime (groter/kleiner is dan) d dagen (oud is)
 - `-newer file_a` : zoekt files/dirs die nieuwer zijn dan file_a
 - `-group/user a` : zoekt files/dirs van group/user a
 - `-executable` : zoekt files/dirs met +x
 - `-nogroup/user` : zoekt files/dirs zonder group/user
 - `-perm -o=wx` : zoekt files/dirs op permissies (bv. hier others kunnen writen en executen)
 - `-size (+-)` : zoekt files/dirs met bepaalde/grotere/kleinere grootte ([niks]/K/M/G/T)
 - `-type (d/f)` : zoekt files/dirs die of directory of file zijn
 - `-exec ... \;` : voert commando uit op gevonden files/dirs (aan te spreken met {})
 - `\;` zorgt dat je nog andere acties kan uitvoeren, zoals bv. `2>/dev/null`
 - `-print` : stuurt de gevonden files/dirs naar stdout

Inrichten van een harde schijf

Dezelfde soort bestanden in 1 partitie/volume te steken en zo verschillende volumes te hebben
(bv. backups, dualboot, swapspace, bestanden die veel wijzigen)



Mountpoint:

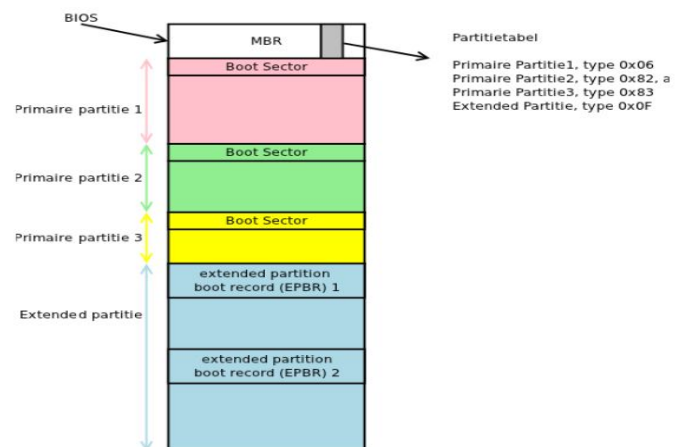
- Linux /boot / /media/WINDOWS_OS /home /media/usbstick
- Windows c:\ d:\ e:\

Lijst van partities

fdisk :dos partitiebeheer	gdisk :guid partitiebeheer	gparted (grafisch)
sdb2 → 2e sata disk, 2e partitie	sda1 → 1e sata disk, 1e partitie	hda2 → 1e ide disk, 2e partitie

“sudo fdisk/gdisk/gparted /dev/[device]” → m (⇒ menu)

- de dos partitietabel (max 2TB)
 - 4 primary of 3 primary en 1 extended
 - (in extended 11 logische partities)
- de guid partitietabel
 - 128 partities



Bestandssystemen aanmaken of formatteren

mkfs -t device

mkfs.(ntfs/ext4/vfat)

Swap Space

free: geeft informatie over het geheugen gebruik

- m : in megabytes
- h : human-readable

swap space toevoegen: mkswap(maak aan) / swapon(gebruik) / swapoff(unmount) /dev/sdXX

Onderhoud

schijfruimte monitoren

df -h : diskfree humanreadable

du : diskusage

- h : humanreadable
- s : summary
- max-depth=1: diskusage per directory

mounten van volumes

mount /dev/sdXX directory

- LABEL= : benoem met label

umount directory/LABEL

ls -l directory : list proces met open files

Commando's ivm Users

- UserAdd -C commentaar naam
- adduser naam
- usermod [optie] naam
 - -c comment : add comment field
 - -d directory : modify Home Directory of account
 - -e yyyy-mm-dd : set expiration date of account
 - -g group : change user primary group
 - -G group : add group to user, but removes user from all other groups
 - -aG : add group to user
 - -l new : change user loginname
 - -L : Lock user account
 - -U : Unlock user account
 - -p password : create unencrypted password for user
 - -s /bin/... : change the usershell
 - -u id : change userID
- passwd naam
 - -l : lock account
 - -x : aantal dagen voor expiry
 - -n : aantal dagen voor wijziging
 - -w : waarschuwing x-dagen voor het verloop v. wachtwoord

Commando's ivm groups

- groups naam : geeft alle groepen van de gebruiker terug
- id naam : hetzelfde, maar met id nummers
- groupadd naam : maak groep met naam aan
- groupdel naam : verwijder groep met naam

Permissies wijzigen

- chown (change owner): chown owner(.group) file/dir
 - -R : recursief ⇒ meestal sudo voor nodig
- chgrp (change group): chgrp group file/dir
- chmod (change mode rwx):
 - relatief: toevoegen of verwijderen
 - chmod u(+rwx),g(+rwx),o(+rwx) file/dir
 - absoluut: overschrijven
 - omzetten naar bits ⇒ indien aanwezig 1 ⇒ rwx r__w_ = 111 100 010
 - omzetten naar octaal ⇒ 111 100 010 = 742
 - chmod 742 file/dir

Speciale permissies

SUID (4) : voert non-script binary programma's uit met rechten van de owner

- chmod 4XXX programma of chmod u+s programma

SGID (2) : voert non-script binary programma's uit met rechten van de groep

alles dat binnen SGID dir wordt aangemaakt krijgt de group owner van die directory

- chmod 2XXX programma of chmod g+s programma

Sticky bit (1) : enkel de eigenaar van de bestanden of directory kunnen deze bestanden verwijderen

- `chmod 1XXX map` of `chmod +t map`

Shell scripting

de combinatie van shell-commando's in een tekstbestand

Zeer handig en kan gebruikt worden voor automatisering van taken in systeemadministratie

Linux gebruikt bash shell scripting (perl, php, python)

Variabelen aanmaken

`naam=waarde` ⇒ `echo $naam`

!! geen spaties in de naam, liefst in lowercase en bovenaan de script

kan enkel gebruikt worden in eigen shell, tenzij het wordt weggestuurd in subshell met "export naam"

Basis van een script

`#!/bin/bash` → zorgt voor "taal" waar in wordt geschreven

`# commentaar`

`var1=waarde1` → maakt een variabele aan en geeft deze een waarde

`clear` → doet een clearscreen

`echo tekst $var1` → doet een echo van tekst en de waarde van `var1` naar de stdout

`read naam` → maakt een variabele "naam" aan en geeft deze de waarde van de input

Kunnen uitvoeren

- opslaan van gemaakt bestand
- `chmod +x bestand` ⇒ maakt bestand executable
- uitvoeren:
 - `./bestand` of `/volledig/absoluut/pad/bestand`
 - bash bestand
 - `source bestand` of `. bestand` ⇒ geen subshell
 - wordt in huidige shell uitgevoerd en variabelen blijven nadien beschikbaar
- \$PATH aanpassen: `PATH=$PATH:directory1:directory2`
`export PATH`

Commandline parameters

kunnen na de scriptnaam meegegeven worden

`$0` = scriptnaam `$1` → `$9` = parameters `${}` = parameters vanaf 10, maar kleiner ook mogelijk

`$#` = aantal meegegeven parameters `$*` = alle parameter `$?` = laatste exit status

Quotes en backquotes

'...' of single/hard quote : alle tekst hierbinnen wordt letterlijk weergegeven

\... : zorgt dat volgende teken niet als een stuk code wordt geïnterpreteerd

"..." of double/soft quote : \$ en ` worden geïnterpreteerd

`...' of \$(...) : wordt gebruikt voor output van een commando

Rekenen

2 notaties (!! geen spaties !!)		bewerkingen: • + - * / %
<code>getal1=\$((1+2*3))</code> <code>getal2=\$((getal1%7))</code>	<code>let getal1=1+2*3</code> <code>let getal2=getal1%7</code>	

for-loop

#!/bin/bash

IFS=\$'\n'

→ IFS heeft standaard whitespaces die gebruikt worden voor scheiding

→ "IFS=:" zorgt dat de resultaten worden gescheiden bij elke ":"

for variablenaam in `commando`

do

commandos

done

Exit status

Elk programma heeft dit: als er niks fout is gegaan zal dit 0 zijn, anders <>0

Op te roepen met echo \$?

true geeft exit status 0

false geeft exit status 1

test

wordt gebruikt om alle mogelijke tests uit te voeren, geeft true of false als exit status

File/String-testing (unair)			
-f ⇒ isFile	-d ⇒ isDirectory	-e ⇒ isFile/Directory	-r ⇒ readrechten
-w ⇒ writerechten	-x ⇒ executerechten	-z ⇒ stringlengte = 0	-n ⇒ stringlengte != 0
File/String-testing (binair)			
-nt ⇒ newer than	-ot ⇒ older than	= ⇒ string equal	!= ⇒ string not equal
Number/boolean-testing (binair)			
-eq/== ⇒ equal	-ge/≥ ⇒ greater than or equal	-gt/> ⇒ greater than	-ne/!= ⇒ not equal
-lt/< ⇒ lesser than	-le/≤ ⇒ lesser than or equal	-a/&& ⇒ and-expression	-o/ ⇒ or-expression

2 notaties

test getal1 != getal2	[getal1 -ne getal2] ⇒ [! getal1 -ne getal2] mogelijk voor omgekeerde exit status
-----------------------	--------------------------------------------------------------------------------------

!! echo \$? kan meteen achter test geplaatst worden door ; tussen beiden te plaatsen

Condities

if [...] && {actie ; actie}	→ begint if-statement	case string in	→ begin van case
then		pattern) acties;;	→ stringpatroon, geen regex
code		pattern) acties;;	
elif [...] actie	→ else if	...)	→ default indien geen matches
then		esac	→ einde van case
code			
else	→ heeft geen then nodig omdat het geen conditie stelt	while [...]	→ voorwaarde die loop laat herhalen
code		do	→ begin van loop
fi	→ eindigt if-statement	sleep 1	→ "slaapt" voor 1 seconde (s,m,h,d)
		code	→ voert code uit
		done	→ einde van loop

softwarebeheer

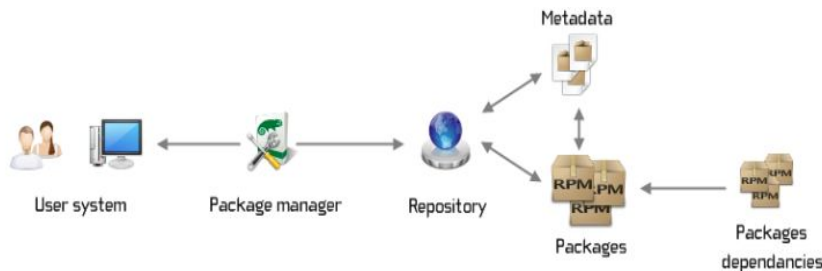
Is op linux modulair

Software, tools en libraries zitten samen in een package met metadata:

(dependencies, install/uninstall, file locations, beschrijving, licentie, man pages, architectuur, md5-key voor beveiliging)

Software repositories

werkt ongeveer zoals Google Store: installatie + auto-updates en wordt door een package manager beheerd



2 types:

- RPM of Redhat Package Manager
 - redhat, fedora, suse, centos
 - rpm en yum command
- deb of debian packages
 - ubuntu, debian, mint
 - apt-get, dpkg en aptitude
 - grafisch: synaptic, ubuntu SC

Debian Software Depositories

/etc/apt/sources.list

- ⇒ main (ondersteuning canonical), restricted (geen Open Source), universe (geen ondersteuning)
- ⇒ multiverse (geen ondersteuning, geen Open Source), backports (niet getest)

Adding debian repositories:

Ubuntu Software Centre, Synaptic, /etc/apt/source.list

- Voeg repository toe (meestal automatisch in .deb)
- Voeg key toe
- refresh de lijst pakketten op repositories
- sudo apt-get update

Commando's

- **apt**
 - -cache, -add-repository, -key, ...
 - -get
 - update, upgrade, install, remove
- **aptitude**
 - textmode, veel mogelijkheden, menu's
 - search samba: toont pakketten met samba in de beschrijving
 - show samba: toont info over het pakket "samba"
- **dpkg**
 - overzicht van alle geïnstalleerde/beschikbare software
 - -l: geeft installatiestatus
 - ii = installed
 - h = on hold ⇒ kan niet gewijzigd worden
 - p = moet verwijderd worden
 - r = alles buiten de config-bestanden verwijderen
 - u = onbepaald
 - -l "*" ⇒ -l van alle software, zelfs non-installed
 - -l map ⇒ -l van map
 - --search (pad): gaat na welk pakket dit bestand heeft
 - -i: instaleert een deb-pakket

Procesbeheer

Parent-child proces

- bv. als je gedit start in de terminal afsluit, zal het proces ook afsluiten
 - gedit & ⇒ gedit zal op de achtergrond blijven draaien
 - wanneer de focus hier niet meer op ligt zal deze nog steeds blijven draaien
 - nohup gedit ⇒ gedit zal onafhankelijk van de terminal lopen
 - als de terminal wordt gesloten zal gedit open blijven staan

Monitor processen

- ps aux → alle proces info
- ps fax → parent child
- pstree → parent child
- top → dynamische info
- htop → dynamische info + menu interface
- uptime → system uptime

processen stoppen

- kill signal procesid ⇒ stop programma met gegeven procesid
 - -SIGKILL of -9 : stop direct en werk niet af
 - -SIGTERM of -15 : stop, maar laat het proces zich afwerken
- killall signal procesnaam ⇒ stop alle programma's waar deze naam in voorkomt

Process-scheduling

crontab -e ⇒ nano ⇒ formaat: (minute hour day-of-month month day-of-week command)

!! man 5 crontab