# Unix Commands

## 2 oktober 2009

Computersystemen en –architectuur

2009 – 2010

# About this course

- Practicum computersystemen en computerarchitectuur

  - Computersystemen: Tools (Unix, RegExp, Bash Scripting, Vi) & MARS

    - 1B INF en 2B/3B WIS (keuzevak)

  - Computerarchitectuur: Logisim (& Verilog)

    - 1B INF, 2B/3B WIS (keuzevak) en 2B/3B INF (keuzevak)

- Blackboard: https://blackboard.ua.ac.be/ → opdrachten!

- Bart Meyers

  - Kantoor: G 3.17, tel.: 03 265 32 79 (in noodgevallen)

  - Email adres: Bart.Meyers@ua.ac.be

  - http://msdl.cs.mcgill.ca/people/bart/ → Teaching  - *Check it often!*

  - Heb je hulp nodig? Maak een afspraak via email!

Universiteit Antwerpen

# About this course (2)

- Praktisch gericht!

  - Wekelijkse opdrachten

    - (bijna) wekelijkse deadline op Blackboard

  - Permanente evaluatie

    - meer dan 50% totaal punten

  - Alles gebeurt in groepjes van twee

    - Pair programming!

# About this course (3)

- Feedback-loop
    1. Indienen oplossing oefeningen + project *(25% van de punten)*
        - je maakt de oefeningen
        - je maakt het project met een verslag
    2. Peer feedback op project *(25% van de punten)*
        - je geeft feedback op het project en het verslag (niet de oefeningen) van twee andere groepjes die ik je stuur (naar je ua email adres)
    3. Indienen verfijnde oplossing *(50% van de punten)*
        - finale oplossing met een verslag waarin de feedback verwerkt is
- Zie http://msdl.cs.mcgill.ca/people/bart/ → Teaching → CS&CA

Universiteit Antwerpen

# Overview

- Why this session on UNIX?

- Introduction on UNIX

- Basic UNIX commands

- Files
  - File system commands
  - Wildcard usage
  - Access permissions

- Processes
  - Job control

- Useful tools

- I/O
  - Standard input & output streams
  - Pipe usage

Universiteit Antwerpen

# Why this session on UNIX?

– This session (and the next two) are not related to "computer systems" or "computer architecture"

– But... you *will* be using UNIX-based systems

  – ironically not necessarily for this course...

  – introduction to the usage of UNIX

  – use this slideshow as a reference in the future

# Introduction

- UNIX

  - Multi-user & Multitasking OS

  - Environment for program development

  - Tool oriented

  - Kernel = core

- UNIX shell

  - User – kernel communication  `fenix:bmeyers> _`

    "A Program that reads in the commands you type and converts them into a form that is more readily understandable by the UNIX system"

  - Interpreted programming language

Read command

Display prompt

Interpreter loop

Interpret command

Execute command
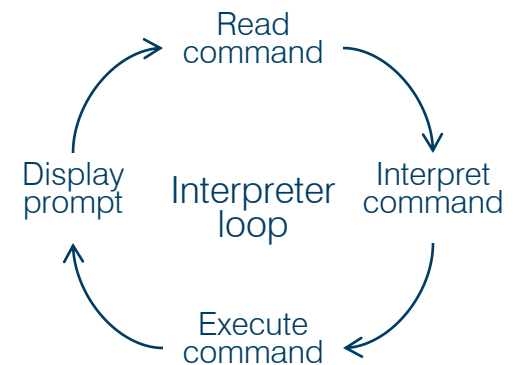
# Basic UNIX commands (1)

– Shell displays a prompt for command input

```
fenix:bmeyers>
```

– `<command_name> [-options] [arguments]`
  – literal string: surround with 'single quotes'
    – backslash \ for single character
  – string with substitutions: surround with "double quotes"
  – process substitution: surround with `backticks`

– Change your password!
  – current password is "qwerty123"
  – `passwd`

```
fenix:bmeyers> ssh p0xxxxx@fenix
Keyboard-interactive:
Password: qwerty123
Authentication successful.
...
fenix:p0xxxxx> passwd
passwd: Changing password for bmeyers
Enter existing login password:
```

– Need command info?
  – `man [-k] <command_name>`
  – `info <command_name>`
  – `apropos <keyword>`

Universiteit Antwerpen

# Basic UNIX commands (1)

– `man passwd`

```
Reformatting page.  Please Wait... done

User Commands                                              passwd(1)

NAME
     passwd - change login password and password attributes

SYNOPSIS
     passwd [-r files | -r ldap | -r nis | -r nisplus]  [name]

     passwd [ -r files] [-egh] [name]

     passwd [ -r files] -s [-a]

     passwd [ -r files] -s [name]

     passwd [ -r files] [-d | -l | -u |  -N]   [-f]   [-n min]   [-
--More--(1%)
```

– `q` : exit

# Basic UNIX commands (2)

– Some basic commands

- `date` : Displays date & time

- `who` : Info on all currently logged on users

- `whoami` : Info about yourself

- `echo` : Display characters on the terminal

- `cat` : Dump file contents on to terminal

- `more` : Displays a text file, one screenful at a time

- `sort` : Sorts its input

# Basic UNIX commands (2)

```
fenix:bmeyers> echo "hello world"
hello world
fenix:bmeyers>
```

```
fenix:bmeyers> who
bmeyers     pts/2          Sep 24 16:09      (143.129.75.215)
p080558     pts/3          Sep 24 15:13      (48.80-240-81.adsl-dyn.isp.belgacom.be)
bmeyers     pts/6          Sep 24 16:00      (143.129.75.215)
bmeyers     pts/7          Sep 24 16:10      (143.129.75.215)
fenix:bmeyers>
```

Universiteit Antwerpen

# Basic UNIX commands (2)

– `cat /etc/passwd`

```
p082299:x:7696:109:VanDenBogaert.Alicia,WIB2,2009,090
921:/export/home/p09/p082299:/usr/bin/bash
p080590:x:7697:109:Wessels.Rutger,WIB2,2009,090921:/e
xport/home/p09/p080590:/usr/bin/bash
p090281:x:7698:109:DeHerdt.Arne,INB1,2009,090924:/exp
ort/home/p09/p090281:/usr/bin/bash
p094227:x:7699:109:DeRoost.Gert,INB1,2009,090924:/exp
ort/home/p09/p094227:/usr/bin/bash
p094275:x:7700:109:Geerinckx.Tobias,INB1,2009,090924:
/export/home/p09/p094275:/usr/bin/bash
fenix:bmeyers>
```

– `more /etc/passwd`

– `space` : next screen

– `q` : quit

– check man pages!

```
fenix:bmeyers> more /etc/passwd
root:x:0:0:Super-User:/:/sbin/sh
daemon:x:1:1::/:
bin:x:2:2::/usr/bin:
sys:x:3:3::/:
adm:x:4:4:Admin:/var/adm:
lp:x:71:8:Line Printer Admin:/usr/spool/lp:
uucp:x:5:5:uucp Admin:/usr/lib/uucp:
nuucp:x:9:9:uucp
--More--(1%)
```

Universiteit Antwerpen

# Working with files (1)

– Paths in a filesystem tree of directories
  – relative path (according to current position in tree)
  – absolute path (full path, starting from root)
– UNIX supports a hierarchical file system
  – '.'  : current directory
  – '..' : parent directory
  – '/'  : root directory
  – '~' : home directory
– Characteristics
  – Case sensitivity!
  – Extensions are non-critical
  – Access permissions
– Types
  – Ordinary
  – Directory
  – Special (devices, queues, …)
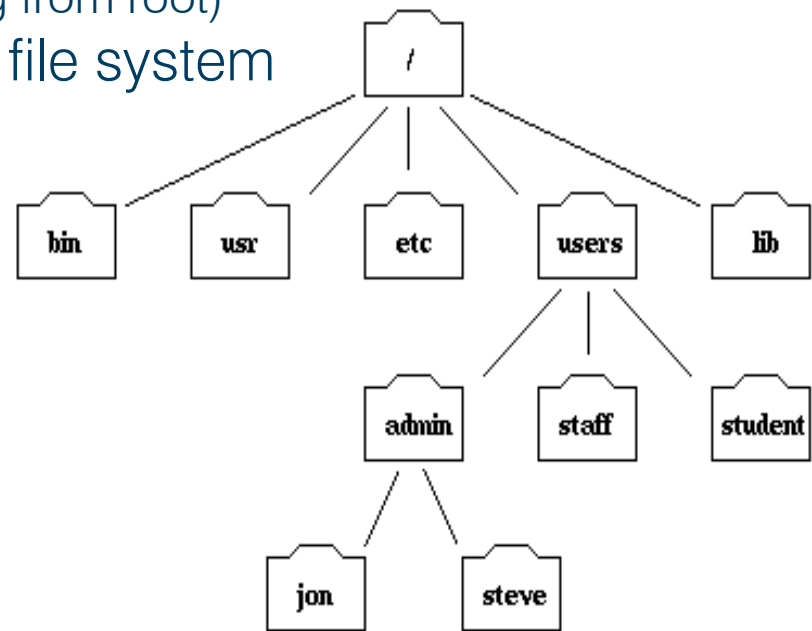
Part of the filesystem tree

# Working with files (2)

– Filesystem commands

– cd <dir>            : Change directory

```
fenix:bmeyers> mkdir test
fenix:bmeyers> cd test
/export/home/edp/bmeyers/test
fenix:test>
```

– mkdir <dir>          : Make directory

– pwd                  : Present working directory

– ls [-l] <filepath>     : Display directory contents

– rm [-r] <filepath>   : Remove file

– file                   : Determine file type

– cp [-r] <from> <to>   : copy file(s) from to

```
fenix:bmeyers> file /etc/passwd
/etc/passwd:     ascii text
fenix:bmeyers>
```

– mv <from> <to>     : move file from to

– ln –s <from> <to>    : create a soft-link from to

# Working with files (2)

```
fenix:bmeyers> mkdir test
fenix:bmeyers> cd test
/export/home/edp/bmeyers/test
fenix:test> ls -al
total 4
drwxr-xr-x   2 bmeyers  edp           512 Sep 24 17:18 .
drwxr-xr-x   5 bmeyers  edp           512 Sep 24 17:18 ..
fenix:test> cp /etc/passwd passwd
fenix:test> ln -s /etc/passwd passwd.txt
fenix:test> ls -al
total 58
drwxr-xr-x   2 bmeyers  edp           512 Sep 24 17:24 .
drwxr-xr-x   5 bmeyers  edp           512 Sep 24 17:18 ..
-rw-r--r--   1 bmeyers  edp         26258 Sep 24 17:24 passwd
lrwxrwxrwx   1 bmeyers  edp            11 Sep 24 17:23 passwd.txt -> /etc/passwd
fenix:test> ls ..
public_html  test
fenix:test>
```

# Working with files (3)

- Wildcard usage

    - '*'                 : zero or more characters

    - '?'                 : exactly one character

    - […],!,-             : range expressions

- Examples

    - [abc]               : a or b or c

    - [!abc]              : a nor b nor c

    - [!a-c]              : all but a … c

    - [a-z]*[!0-9]        : all that start with lowercase letter and don't end with a digit

    - A?p                 : all 3 letter words starting with a and ending on p

# Working with files (3)

```
fenix:test> ls /bin/*zip*
/bin/bunzip2        /bin/gzip           /bin/zipcloak
/bin/bzip2          /bin/p7zip          /bin/zipgrep
/bin/bzip2recover   /bin/unzip          /bin/zipinfo
/bin/funzip         /bin/unzipsfx       /bin/zipnote
/bin/gunzip         /bin/zip            /bin/zipsplit
fenix:test>
```

Universiteit Antwerpen

# Working with files (4)

- Access permissions

  - All files (ordinary, directory, special) have access attributes

  - Modes (for owner, group and everybody)
    - Read       r    4      file: readable      dir: files can be listed
    - Write      w    2      file: adaptable     dir: files can be adapted
    - Execute    x    1      file: execute       dir: files can be accessed

  - Change with **chmod** command (use sum of modes: "5" = "4"+ "1"

    = readable and executable)
    - chmod 751 means owner:r,w,e; group r&e; everybody: e

  - **chown** & **chgrp** to change owner and group

  - **umask** to change default permissions
    - 666 (for files)/777 (for dirs) minus default permissions
    - example: **umask 022** : file permissions 644, dir permissions 755

**Universiteit Antwerpen**

# Working with files (4)

```
fenix:bmeyers> cd test
/export/home/edp/bmeyers/test
fenix:test> ls -al
total 58
drwxr-xr-x   2 bmeyers   edp              512 Sep 24 17:24 .
drwxr-xr-x   5 bmeyers   edp              512 Sep 24 17:18 ..
-rw-r--r--   1 bmeyers   edp            26258 Sep 24 17:24 passwd
lrwxrwxrwx   1 bmeyers   edp               11 Sep 24 17:23 passwd.txt -> /etc/passwd
fenix:test> umask
0022
fenix:test> chmod 755 passwd
fenix:test> ls -l passwd
-rwxr-xr-x   1 bmeyers   edp            26258 Sep 24 17:24 passwd
```

## Universiteit Antwerpen

# Working with files (5)

- Archiving
  - Tar
    - Create      : `tar -cf <tarball_path> <files>`
    - Extract     : `tar -xvf <tarball_path>`
    - View        : `tar -tvf <tarball_path>`
  - GZip
    - Create      : `gzip <archive_name> <files>`
    - Extract     : `gunzip <archive_name>`
  - Result: tar.gz archive

## Universiteit Antwerpen

# Processes (1)

- "A process is a running program, managed by the scheduler and the memory management subsystem"

- Hierarchical

- Process linked to

  - UID / GID   : user & group ID

  - PID / PPID : process & parent process ID

  - A priority

    - Range : -20 to 19

    - Use `nice` command to alter the default priority (0)

# Processes (2)

– Jobs are connected to the terminal which started them by default
  – Currently connected job runs in the foreground
  – Jobs can also be run in the background

– Job control
  – Ctrl^C : kill current job
  – `kill [-9] <pid>` : kill on process ID
  – `pkill <pname>` : kill on process name
  – Ctrl^Z : suspend current job
  – `<command>&` : run <command> in background
  – `ps [-ef]` : display process info (e = every process)
  – `top` : display process info interactively
  – `jobs` : display user's jobs
    – Refer to jobs using %1, %2; i.e fg %2
  – `fg <jobID>` : bring job into the foreground on job ID
  – `bg <jobID>` : set job into the background on job ID

Universiteit Antwerpen

# Processes (2)

- `top`

```
load averages:  0.05,  0.04,  0.04                                    18:12:42
173 processes: 171 sleeping, 1 zombie, 1 on cpu
CPU:      % idle,      % user,       % kernel,       % iowait,      % swap
Memory: 12G real, 10G free, 564M swap in use, 33G swap free

   PID USERNAME  THR PRI NICE   SIZE     RES STATE      TIME     CPU COMMAND
 14236 bmeyers     1  59     0 2736K  1672K cpu/1      0:00   0.03% top
   161 root       13  59     0 9600K  8360K sleep     10:53   0.02% picld
  1652 mysql       9  59     0   55M    20M sleep      9:18   0.02% mysqld
   976 root       28  29    10   90M    36M sleep      5:14   0.01% java
  1748 root       24  59   -10   38M    17M sleep      4:50   0.01% java
   958 root        1  59     0 3384K  2072K sleep      4:17   0.01% _upsd
  2974 root        1  59     0   15M  6008K sleep      3:56   0.01% Xsun
   357 root        4  59     0 8408K  4896K sleep     25:48   0.00% inetd
 15341 p051403      4  53     2  134M    72M sleep      2:49   0.00% soffice.bin
  1565 root        1  59     0   33M    21M sleep      2:19   0.00% Xsun
  1771 root        1   2    19 3360K  2680K sleep      2:17   0.00% tictimed
  3091 root        1  59     0 9440K  3984K sleep      2:15   0.00% dtgreet
  1715 root        1  59     0   14M  5144K sleep      2:14   0.00% dtgreet
   647 daemon      2  60   -20 2880K  1856K sleep     16:01   0.00% nfsd
   130 root       34  59     0 8600K  5384K sleep      3:34   0.00% nscd
```

Universiteit Antwerpen

# Some useful tools (1)

- Search
  - `grep`    : searches file for lines matching a pattern
  - `find`    : descend directory hierarchy recursively in search for files
              matching a boolean expression
  - `finger`  : lookup info on user

- Filters
  - `tail`    : get last lines of file
  - `head`    : get starting lines of file
  - `cut`     : extract selected fields of each line of text
  - `paste`   : merge same lines of several files or subsequent lines in
              one file
  - `uniq`    : report repeated lines in a file
  - `tr`      : copies the stdin to stdout with substitution or deletion of
              specified characters.

Universiteit Antwerpen

# Some useful tools (2)

- Remote
  - `ssh`  : remote login program
    - `ssh bmeyers@fenix.cmi.ua.ac.be`
  - `scp`  : copy files over a network securely
    - `scp /etc/passwd bmeyers@fenix.cmi.ua.ac.be:passwd.txt`
  - `wget`  : download files from the web
    - `wget http://msdl.cs.mcgill.ca/people/bart/courses/materials/cs1oef`

- Misc
  - `wc`  : count words/lines/characters in input file
  - `alias`  : bind an alias to a command string
    - `alias <name>="<command>"`
  - `unalias`  : unbind an alias

Universiteit Antwerpen

# Streams (1)

– Every program run from the shell is tied to 3 channels (special files)
  – stdin  : standard input (default : keyboard)
  – stdout: standard output (default : terminal)
  – stderr : standard error (default : terminal)

– Redirection possible using $<$, $>$, $>>$
  – `ls -alrF > listing.txt`   : store ls output in listing.txt
  – `sort < listing.txt`       : feed listing.txt to sort program
  – `echo HOI >> listing.txt`  : append string HOI to listing.txt
– Redirection of stdout (1>) and stderr (2>)
  – sometimes %1> and %2>
  – redirection of stderr to stdout: `<command> 2>&1`

– Use the 'null' device to suppress output
  – `sort listing.txt 2> /dev/null`

# Streams (2)

- Connect output of command A to input of command B?

  - => Use pipes

  - redirect stdin and/or stdout

- Example: find all files that changed in september

```
fenix:bmeyers> ls -l /usr/bin | grep "Sep"
-r-xr-xr-x   1 root      bin          18984 Sep 13  2007 ctags
-r-xr-xr-x   5 root      bin         239828 Sep 13  2007 edit
-r-xr-xr-x   5 root      bin         239828 Sep 13  2007 ex
lrwxrwxrwx   1 root      other           18 Sep 14 13:29 javaws -> ../java/bin/javaws
drwxr-xr-x   2 root      bin           1536 Sep 14 13:57 sparcv9
-r-xr-xr-x   5 root      bin         239828 Sep 13  2007 vedit
-r-xr-xr-x   5 root      bin         239828 Sep 13  2007 vi
-r-xr-xr-x   5 root      bin         239828 Sep 13  2007 view
fenix:bmeyers>
```

## Universiteit Antwerpen

# Streams (2)

– Example: give a sorted list of all occurring first names of 1B INF

```
fenix:bmeyers> grep INB1 /etc/passwd | cut -f5 -d: | cut -f2 -d. |
cut -f1 -d, | uniq | sort | pr -5 | more


Sep 24 18:41 2009    Page 1


Adnan          Detlev         Jan            Lukas          Robin
Alexander      Dorsan         Jiannan        Matteo         Sam
Arne           Elliot         Joeri          Matthias       Stephane
Bert           Eric           Jonas          Michael        Timmy
Bo             Fabio          Joram          Oualid         Tobias
Bram           Gamale         Joris          Philippe       Tom
Chris          Geoffri        Katrien        Pieter         Ward
Christof       Gert           Keke           Quinten        Wolf
Christophe     Glenn          Kemal          Robbe          Yentl
Daan           Ignace         Kerstin
```

# Exercises

– http://msdl.cs.mcgill.ca/people/bart/courses/materials/cs1oef