

Introduction Linux

Dr. M. F. Somers, January 2016
Theoretical Chemistry group, Leiden University.
m.somers@chem.leidenuniv.nl

This compulsory module of the course 'Scientific Computing and Programming' is based on the 'Linux Fundamentals' written by Paul Cobbaut. You can find the latest .pdf of the book at <http://linux-training.be/linuxfun.pdf>. You may want to keep this page bookmarked!

The following chapters and timetable will be used during the course:

Day 1: Chapters 1 and 7 to 11.6. Chapter 5 is non-compulsory.

Day 2: Chapters 12 to 17.

Day 3: Chapters 18 to 21.2.

Day 4: Chapters 23 to 25.4 and chapter 26.

Day 5: Chapters 30, 32 and 35.

During the period you are encouraged to use a Linux live CD for a workstation (<http://getfedora.org>) and do all the exercises of the chapters. The final test will be given to you afterwards. The exam will be a practical test in which your actual skills are tested!

Examples of exam tests:

- 1) Boot the latest Ubuntu Live CD and write a small shell-script to rename several files at once; i.e. '*.o' to '*.o_s'.
- 2) Boot the latest Ubuntu Live CD and write a small shell-script that automatically cleans a users '~/tmp' directory from files older than 7 days. Let this script run through crontab.
- 3) Boot the latest Ubuntu Live CD and:
 1. Add a new user named 'god'. Make sure 'god' has a safe password.
 2. Let 'god' be part of the 'sudoers'.
 3. Add a new user named 'devil'. Make sure that 'devil' has no login shell.
- 4) Boot the latest Ubuntu Live CD and create an RSA public/private key pair. Make sure you add the public key to the '~/.ssh/authorized_keys' file and test to see if you are able to 'ssh' into your account without testing a password.
- 5) Boot the latest Ubuntu Live CD, download and install the OpenMPI library using a 'tar-ball'. Check the install and the library by running the example 'hello_c' code supplied by OpenMPI. Compile the code using the 'make' command and the 'Makefile'. Run the code by using 'mpirun'. Read the manual pages of 'mpirun' for assistance.
- 6) Boot the latest Ubuntu Live CD, download and install the FFTW 3.1 library. Use a 'tar-ball' for the installation. Check the install and the library by running the 'bench' example program supplied by FFTW. Compile the code using the 'make' command and the 'Makefile'. Read the documentation of 'bench' for assistance.
- 7) Boot the latest Ubuntu Live CD, download the EuroBen 5.0 benchmark code, compile the code and run the benchmarks.

Students will be handed out two or three questions similar to the examples above and are allowed to use the internet during the live exam. The exam will take approximately three hours.