

Introduction to the Linux Commandline

Release 1.4

Holger Dinkel, Frank Thommen & Thomas Zichner

February 14, 2014

Contents

1	Introduction to the Linux Commandline	1
1.1	Why Use the Commandline	1
1.2	General Remarks Regarding Using UNIX/Linux Systems	1
1.3	General Structure of Linux Commands	2
1.4	A Journey Through the Commands	3
1.4.1	Useful Terminal Tools & Keyboard Shortcuts	4
1.4.2	Getting Help	5
1.4.3	Who am I, where am I	6
1.4.4	Moving Around	6
1.4.5	See What's Around	7
1.4.6	Organize Files and Folders	9
1.4.7	View Files	11
1.4.8	Extracting Informations from Files	13
1.4.9	Useful Filetools	15
1.4.10	Permissions	16
1.4.11	Remote access	17
1.4.12	IO and Redirections	18
1.4.13	Environment Variables	19
2	Exercises	21
2.1	Misc. file tools	21
2.2	Searching	21
2.3	Misc. terminal	21
2.4	Permissions	21
2.5	Remote access	22
2.6	IO and Redirections	22
3	Links and Further Information	25
3.1	Links	25
3.2	Command Line Mystery Game	26
3.3	Real printed paper books	26
3.4	Live - CDs	26
3.4.1	Fedora Live CD	26
3.4.2	Knoppix	26
3.4.3	BioKnoppix	27
3.4.4	Vigyaan	27
3.4.5	BioSlax	27
4	About Bio-IT	29
4.1	Resources	29
4.2	Training and Outreach	29

4.3	Networking	29
4.4	Biocomputing expertise at EMBL	29
5	Acknowledgements	31
6	Solutions to the Exercises	33
6.1	Misc. file tools	33
6.2	Searching	33
6.3	Misc. terminal	34
6.4	Permissions	34
6.5	Remote access	35
6.6	IO and Redirections	36
	Index	39

Chapter 1

Introduction to the Linux Commandline

1.1 Why Use the Commandline

- It's **fast**. Productivity is a word that gets tossed around a lot by so-called power users, but the command line can really streamline your computer use, assuming you learn to use it right.
- It's **easier to get help**. The command line may not be the easiest thing to use, but it makes life a whole lot easier for people trying to help you and for yourself when looking for help, especially over the internet. Many times it's as simple as the helper posting a few commands and some instructions and the recipient copying and pasting those commands. Anyone who has spent hours listening to someone from tech support say something like, "OK, now click this, then this, then select this menu command" knows how frustrating the GUI alternative can be.
- It's nearly **universal**. There are hundreds of Linux distros out there, each with a slightly different graphical environment. Thankfully, the various distros do have one common element: the command line. There are distro-specific commands, but the bulk of commands will work on any Linux system.
- It's **powerful**. The companies behind those other operating systems try their best to stop a user from accidentally screwing up their computer. Doing this involves hiding a lot of the components and tools that could harm a computer away from novices. Linux is more of an open book, which is due in part to its prominent use of the command line.
- Many 'modern' bioinformatics tools (samtools, bamtools, ...) are written for the commandline in order to be run on clusters and to be incorporated in scripts.

1.2 General Remarks Regarding Using UNIX/Linux Systems

- **Test before run**. Anything written here has to be taken with a grain of salt. On another system - be it a different Linux distribution or another UNIXoid operating system - you might find the same command but without the support of some of the options taught here. It is even possible, that the same option has a different meaning on another system. With this in mind always make sure to test your commands (specially the "dangerous" ones which remove or modify files) when switching from one system to the other.

- **The Linux/UNIX environment.** The behaviour of many commands is influenced or controlled by the so-called “environment”. This environment is the sum of all your environment variables. Some of these environment variables will be shown towards the end of this course.
- **UPPERCASE, lowercase.** Don’t forget that everything is case-sensitive.
- **The Filesystem.** Linux filesystems start on top at the root directory (sic!) “/” which hierarchically broadens towards the ground. The separator between directories or directories and files in Linux is the slash (“/”).

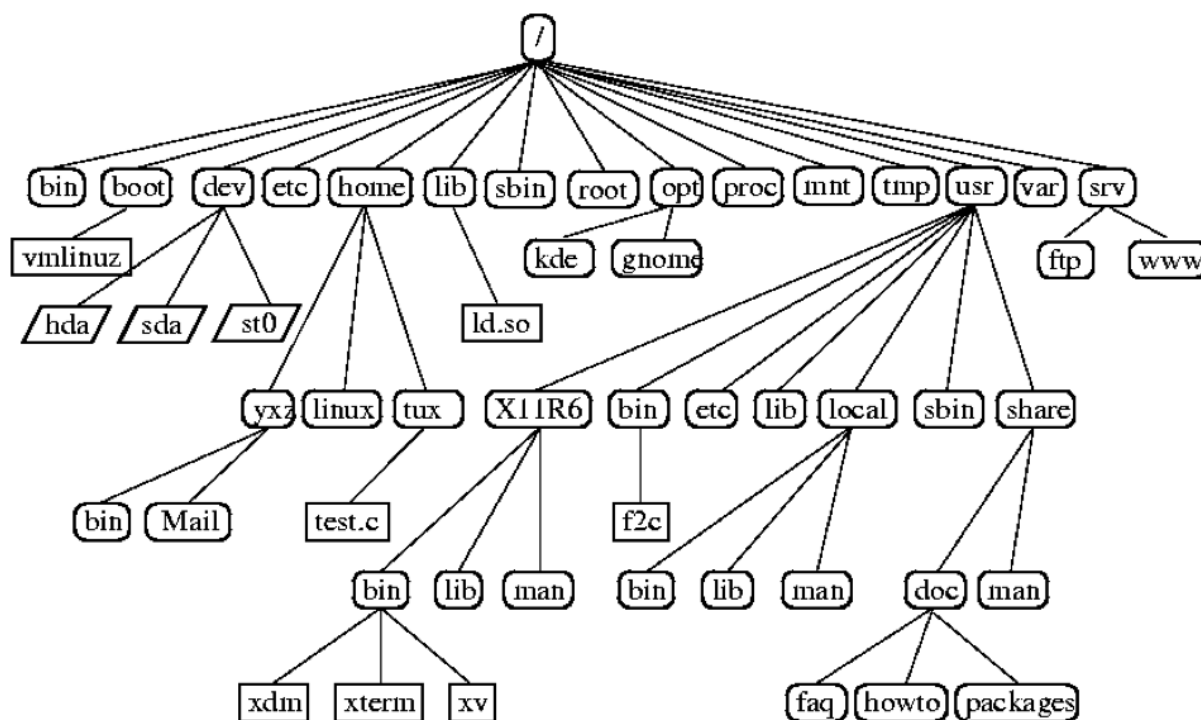


Figure 1.1: Depending on the Linux distribution you might or might not find all of above directories. Most important directories for you are `/bin` and `/usr/bin` (sometimes also `/usr/local/bin`) which contain the user software, `/home` which usually contains the users’ homedirectories and `/tmp` which can be used to store temporary data (beware: Its content is regularly removed!).

Note: The terms “directory” and “folder” are used interchangeably in this document.

1.3 General Structure of Linux Commands

Many linux commands have options and accept arguments. Options are a set of switch-like parameters while arguments are usually free text input (such as a filename).

For example, in the commandline `ls -l /usr/bin`, `ls` is the command, `-l` is an option and `/usr/bin` qualifies as an argument.

Commandline options (sometimes called comandline switches) commonly have one of the two following forms: The short form `-s` (just a single character) or the long form `--string`. E.g.