

Introductory Course in Linux

Pedro Ojeda-May, Birgitte Brydsö, and Jerry Eriksson

HPC2N,
UmeåUniversity,



901 87, Sweden.

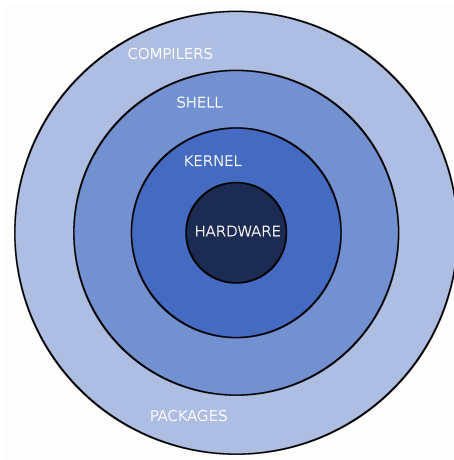
September 11, 2017



Table of contents

- 1 Linux terminal
- 2 File editing
- 3 Data handling
- 4 Environment variables
- 5 Scripting

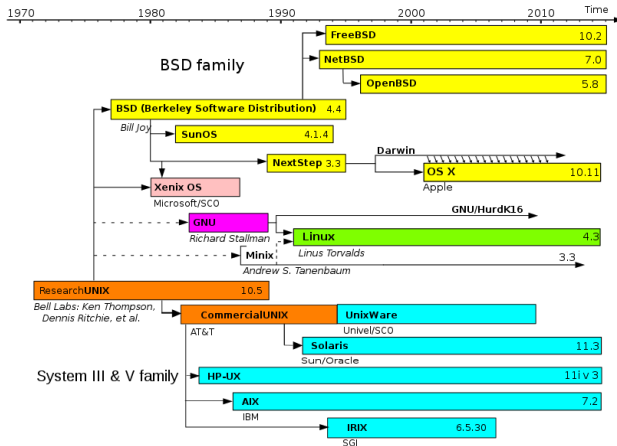
Linux OS



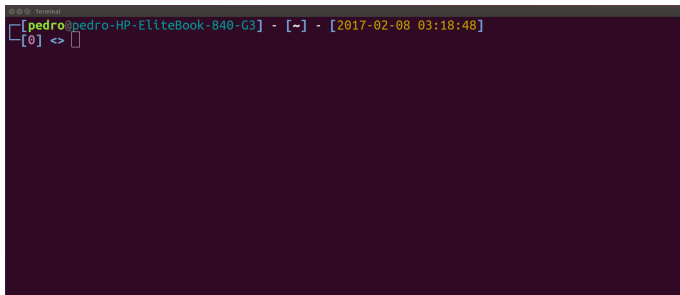
Linux OS

- UNIX-like OS
- used in modern Android smartphones
- the difference between UNIX OS is small

Linux timeline



The Linux terminal



```
terminal
[pedro@pedro-HP-EliteBook-840-G3] - [~] - [2017-02-08 03:18:48]
[0] <>
```

- on the terminal you can see the so-called Prompt
- here you can control your PC/account or even a remote server

Files organization

```
Terminal
[pedro@pedro-HP-ElliteBook-840-G3] - [~/Linux_Abisko_Kebne] - [2017-02-08 03:40:22]
[0] <=> tree
tree .
.
├── draw
│   ├── filesystem.odg
│   └── gromacs-example ─── Directories
│       ├── job.pbs
│       ├── npt.tpr
│       ├── HandsOn.aux
│       ├── HandsOn.tex
│       ├── HowToApply.aux
│       └── HowToApply.tex
├── hpc2n_intro_course_April2015.aux
├── hpc2n_intro_course_April2015.log ─── Regular files
├── hpc2n_intro_course_April2015.nav
├── hpc2n_intro_course_April2015_orig.pdf
├── hpc2n_intro_course_April2015.out
├── hpc2n_intro_course_April2015.pdf
├── hpc2n_intro_course_April2015.snm
├── hpc2n_intro_course_April2015.tex
├── hpc2n_intro_course_April2015.toc
├── hpc2n_intro_course_April2015.vrb
├── hpc2n_intro_course_Oct2016.pdf
└── Images
    ├── abisko.eps
    ├── abisko.jpg
    ├── alloktion-fatnode.eps
    ├── alloktion-fatnode-eps-converted-to.pdf
    ├── alloktion-gpu.eps
    ├── alloktion-gpu-eps-converted-to.pdf
    ├── alloktion-thinnode.eps
    ├── alloktion-thinnode-eps-converted-to.pdf
    ├── data_kebne.dat
    ├── data_kebne.eps
    ├── data_kebne-eps-converted-to.pdf
    ├── filesystem.eps
    ├── filesystem-eps-converted-to.pdf
    └── kebnekaise.eps
```



Useful commands: man

Manual pages.

- **man command: man nano**

```
NANO(1)                                General Commands Manual                                NANO(1)

NAME
    nano - Nano's ANOther editor, an enhanced free Pico clone

SYNOPSIS
    nano [options] [[+line,column] file]...

DESCRIPTION
    nano is a small, free and friendly editor which aims to replace Pico,
    the default editor included in the non-free Pine package. On top of
    copying Pico's look and feel, nano also implements some missing (or
    disabled by default) features in Pico, such as "search and replace" and
    "go to line and column number".
```

Useful commands: ls

List the content of a directory

```
$ls  
1CD9
```

```
$ls -l  
total 24843644  
drwxrwxr-x  2 pedro pedro          4096 nov  9 11:17 1CD9
```

```
$ls -la  
total 24844368  
drwxr-xr-x 44 pedro pedro          4096 feb 13 13:19 .  
drwxr-xr-x  3 root  root          4096 sep 19 11:05 ..  
drwxrwxr-x  2 pedro pedro          4096 nov  9 11:17 1CD9
```

```
$ls -lah  
total 24G  
drwxr-xr-x 44 pedro pedro 4,0K feb 13 13:25 .  
drwxr-xr-x  3 root  root  4,0K sep 19 11:05 ..  
drwxrwxr-x  2 pedro pedro 4,0K nov  9 11:17 1CD9
```

Useful commands: ls

```
$ls -laht
total 24G
drwxr-xr-x 44 pedro pedro 4,0K feb 13 13:29 .
-rw----- 1 pedro pedro 431K feb 13 13:29 .zsh_history
drwx----- 6 pedro pedro 4,0K feb 13 13:28 Linux_Abisko_Kebne
```

```
$ls -lahrt
total 24G
-rw-r--r-- 1 pedro pedro 655 sep 19 11:05 .profile
```



Useful commands: cd

Change directory.

Useful cases:

- `cd directory`
move to "directory"
- `cd`
move to `$HOME` directory
- `cd -`
move to previous visited directory
- `cd ..`
move to upper directory in the hierarchical tree



Useful commands: cp

Copy files.

Useful cases:

- `cp text.txt directory/`
copy text.txt file to "directory"
- `cp -r test/ directory/`
copy the directory test into directory/.
cp overwrites existing files!

Useful commands: touch/mkdir

Create files.

Useful cases:

- touch text.txt
creates text.txt file
- mkdir test
creates the directory test

Useful commands: rm

Remove files.

Useful cases:

- `rm text.txt`
deletes text.txt file
- `rm -rf test/`
deletes the directory test
deleted files cannot be recovered!

Wild cards

- ?
it represents a single character
- *
it represents a string of characters
- $[0 - 9]$, $[A - B]$
it represents a range of numbers or characters

Useful commands: grep

This command searches for patterns in text files.

Useful cases:

- `grep 'word' file`
it searches for pattern 'word' in file
- `grep -rine 'word' home`
pattern word is searched recursively in the directory */home*

Useful commands: `awk`

This command finds patterns in a file and can perform arithmetic/string operations.

Useful cases:

- `awk '/gold/ {print$1}' file`
- it searches for pattern 'gold' in file and prints out the first column

Useful commands: ssh

Command for connecting to a remote computer.

Useful cases:

- `ssh username@abisko.hpc2n.umu.se`
connecting to abisko machine
- `ssh -Xl username abisko.hpc2n.umu.se`
if you want to enable graphical display.

Useful commands: sftp

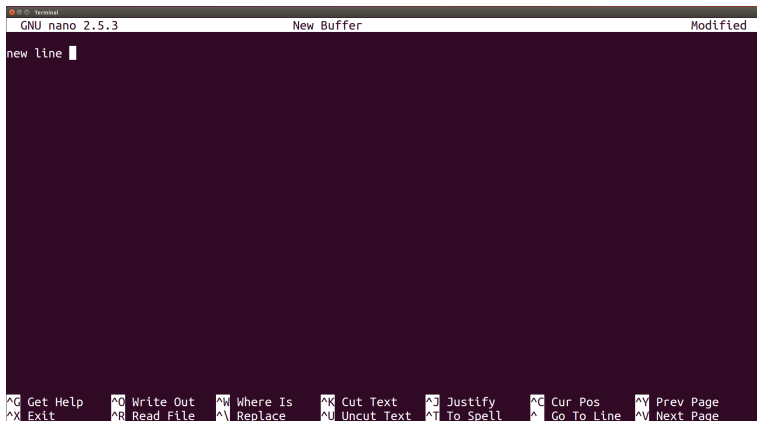
Protocol for data transfer.

```
$sftp username@abisko.hpc2n.umu.se
```

```
$get file
```

```
$put file
```

Editing files

A screenshot of a Linux terminal window running the GNU nano 2.5.3 text editor. The window title is 'Terminal'. The editor's status bar at the top shows 'GNU nano 2.5.3' on the left, 'New Buffer' in the center, and 'Modified' on the right. The main editing area is dark purple and contains the text 'new line' followed by a white cursor. At the bottom, a light-colored status bar displays various keyboard shortcuts for navigation and editing, such as '^G Get Help', '^O Write Out', '^W Where Is', '^K Cut Text', '^J Justify', '^C Cur Pos', '^Y Prev Page', '^X Exit', '^R Read File', '^_ Replace', '^U Uncut Text', '^T To Spell', '^_ Go To Line', and '^V Next Page'.



Compress/decompress files

Compressing files:

```
$gzip file      --->  file.gz
```

Decompressing files:

```
$gunzip file.gz
```

Generating archives

Generate tar-ball:

```
$tar -cvf directory.tar directory
```

Opening tar-ball:

```
$tar -xvf directory.tar
```

Exporting variables

- some programs or libraries require environment variables to work
- they allow the program to follow different schemes without being re-compiled
- some variables such as `$HOME` are intrinsic to Linux OS
- we need to export the variables for further use:

```
$export NUMBER_OF_THREADS=6
```

Scripting

- allows to perform complex tasks without user intervention
- all Linux commands can be used in a script including wild cards

Scripting

analysis.sh

```
#!/bin/bash
```

```
grep 'ABCD' file.pdb > file_filtered.pdb
```

```
program < file_filtered.pdb > output.dat
```

execute script with `./analysis.sh`

Scripting

```
$ls -lah
total 24G
drwxrwxr-x  2 pedro pedro 4,0K nov  9 11:17 1CD9
```

- permissions are set of "user", "group", or "others"
- we can change permissions with chmod command

For instance,

```
$chmod u+x analysis.sh
```

```
$execute script with ./analysis.sh
```



Announcement

Version Control Workshop on Oct. 16, 2017
<http://coderefinery.org/workshops/>