Table of Contents

- 1 Introduction to Unix, Part 2 - Connecting to a remote server
- 2 How many computers are we going to use in this course?
  - 2.1 Our course - computers infrastructure
- 3 Connecting to a remote server
  - 3.1 Connecting to evop2018login.imp.fu-berlin.de - RSA key fingerprint
  - 3.2 Connecting to evop2018Server - Password
  - 3.3 Are you in evop2018 Server?
  - 3.4 Connection error?
  - 3.5 If the connection doesn't work at all - downloading the materials on your computer
- 4 Executing graphical applications from a remote server
- 5 Which folder am I?
- 6 Where can I go?
  - 6.1 The **'cd'** man page
  - 6.2 If you get lost: type "cd" without arguments
  - 6.3 How to refer to the current (.) and parent (..) folders
- 7 Structure of the Unix file system
  - 7.1 The Unix file tree
  - 7.2 Contents of the /root directory
  - 7.3 The course materials
- 8 Copy the course materials in your home
- 9 The exercises files
- 10 Accessing the contents of a file: head, cat, less
  - 10.1 The -n option in head
  - 10.2 Browsing the contents of a file with less
- 11 Editing files with nano
  - 11.1 The nano text editor
  - 11.2 How to use nano
  - 11.3 Nano exercise
- 12 Other useful Unix commands (moving and copying files)
  - 12.1 Other Unix commands
- 13 Time for lunch!

Introduction to Unix, Part 2 - Connecting to a remote server

Welcome to the Programming for Evolutionary Biology workshop!!

Giovanni M. Dall'Olio and Alvaro Perdomo-Sabogal, 03/03/2019. All materials available here: https://github.com/dalloliogm/evop2019/archive/master.zip (https://github.com/dalloliogm/evop2019/archive/master.zip)

How to use these slides: Press Space to get to the next slide. Use arrows to navigate the subsections.

How many computers are we going to use in this course?

The **classroom computers** are in front of you.

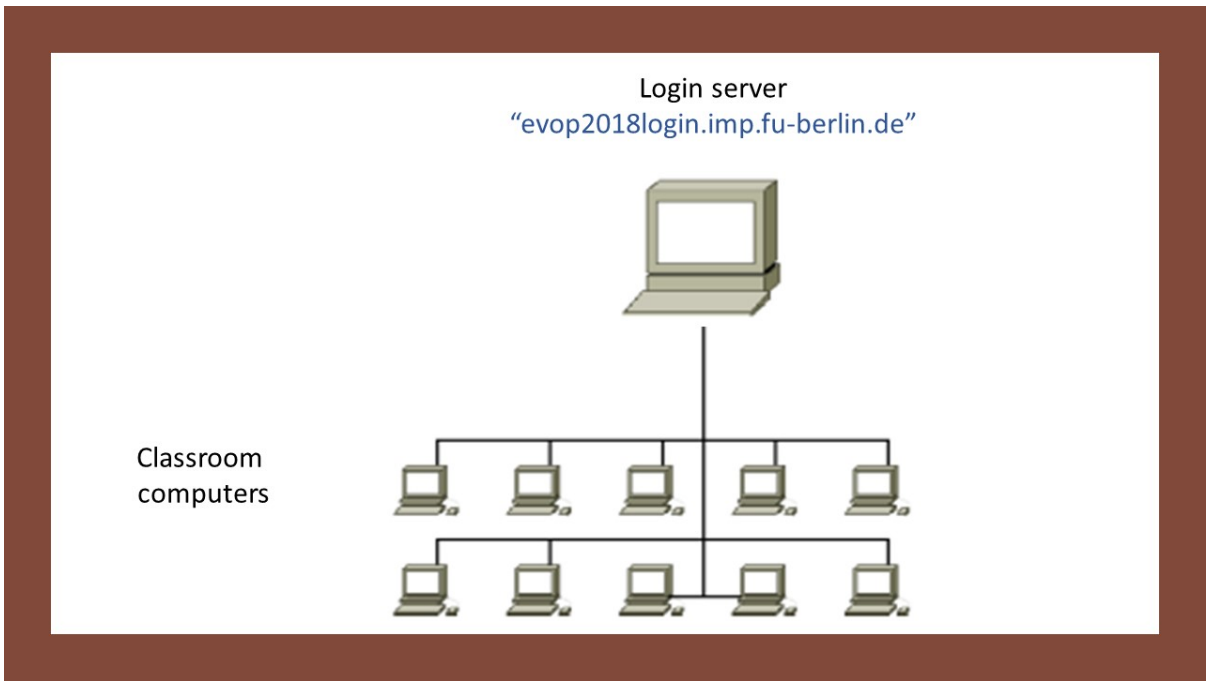
- Use them to follow the slides, ~~check emails~~, and connect to the server/ big computer/ cluster....

evop2018login.imp.fu-berlin.de is a remote computer

- They are physically located outside of this room
- They are only accessible through the terminal
- Most exercises will be done in this server, as all the software needed is installed there

The method to connect to a remote computer is the same

Our course - computers infrastructure



Connecting to a remote server

Please refer to the login sheet **Passwords and Computer Infrastructure** for access credentials

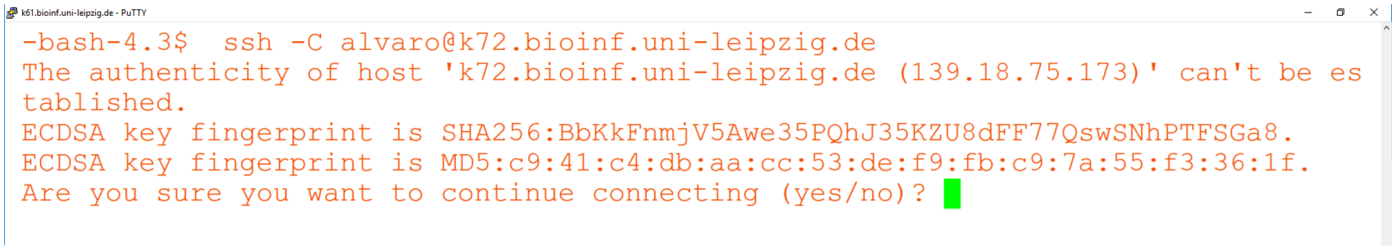
To connect to a remote server, we will use the **ssh** bash command:

```
ssh -C k00nnnn##@evop2018login.imp.fu-berlin.de
```

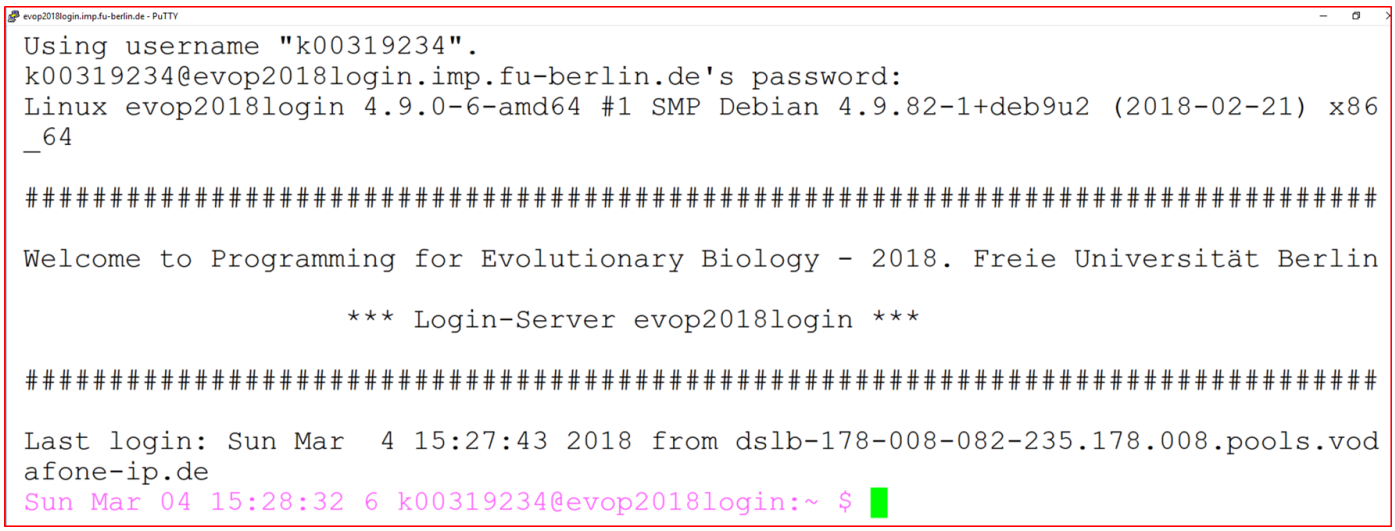
Please use the -C flag (use compression), to reduce the network load

Connecting to evop2018login.imp.fu-berlin.de - RSA key fingerprint

The first time you connect to a server, you should see a message like the following:

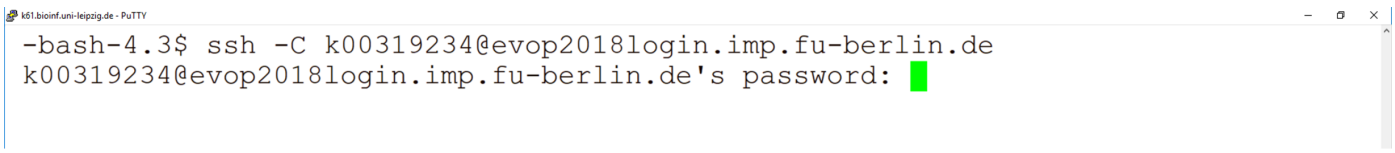


Type ** "yes + Enter"** to continue. Note that typing "y" will not be enough.



Connecting to evop2018Server - Password

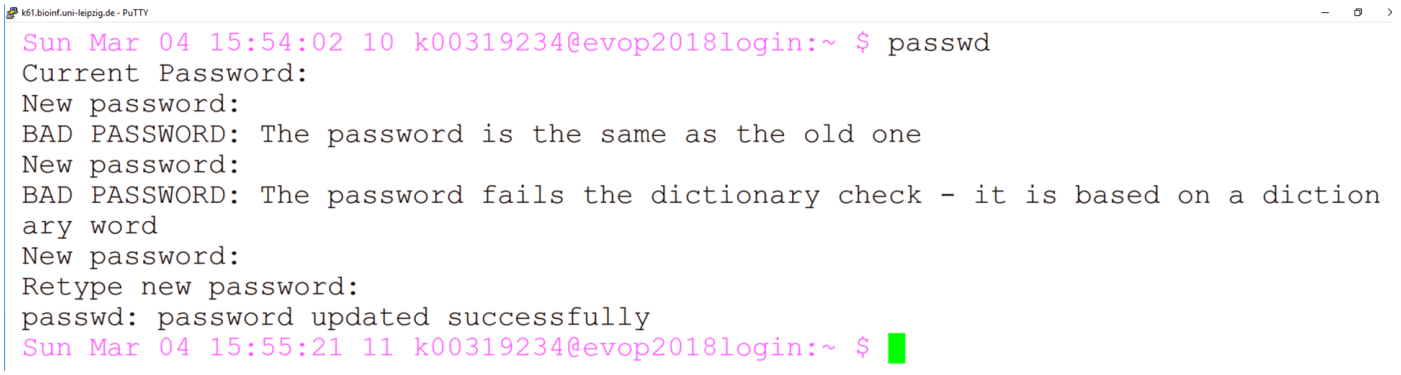
If the connection is working, you should now get a message asking you for the password.



Type the password and press Enter to continue.

** Is it is possible to change the password? Yes:**

use:



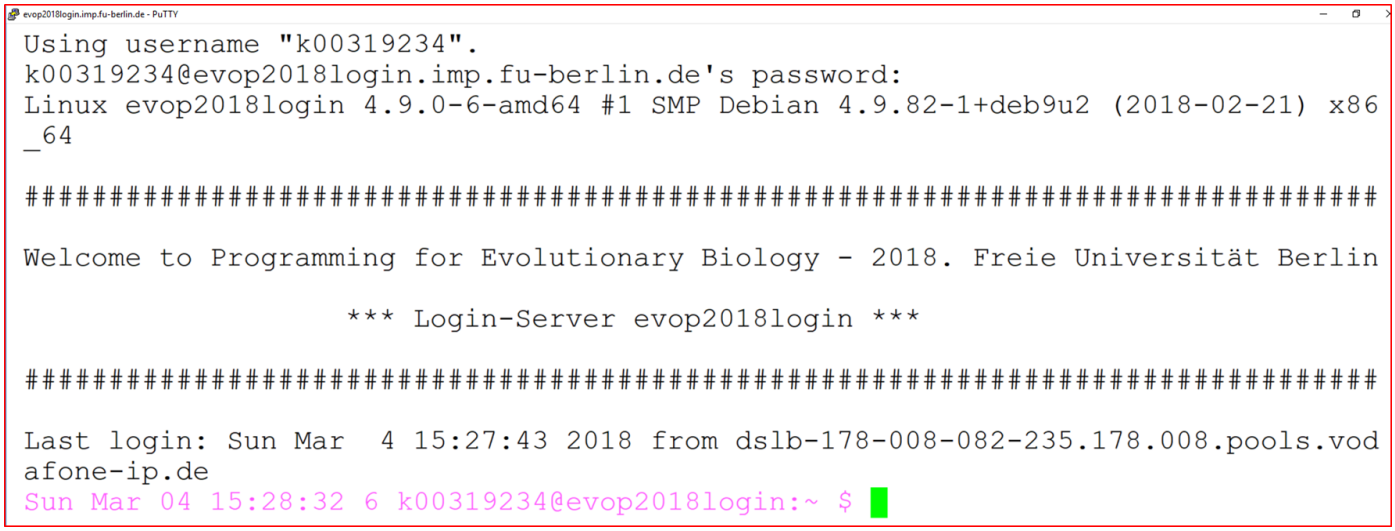
Special considerations for the new passwords:

*1. If you change the password of your **k00nnnn##**'s account, please do it wisely, choose a **strong** one.*

*2. The workshop server and the 30 desktop PCs are accessible from "the world" via SSH, so chances are that attackers will try out easily guessable passwords on these computers. If you **choose a new and weak** password for your account, **it might get compromised!***

Are you in evop2018 Server?

If the connection worked correctly, you should see something like:



You can also type **uname -n** to confirm that you are in the correct machine.

Connection error?

If you get a connection error, try the following:

- make sure you are using the -C option
- Wait 1-2 minutes and try again
- Ask for help (TAs)

If the connection doesn't work at all - downloading the materials on your computer

If the connection to the server doesn't work at all, you can download the exercises file to your computer directly:

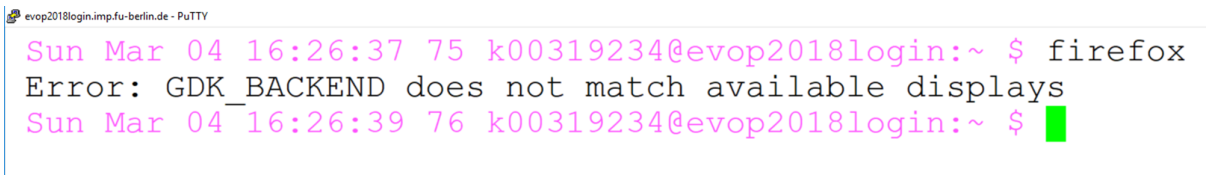
```
$: wget  https://github.com/dalloliogm/evop2019/archive/master.zip

$: unzip master.zip
```

** NOTE**: There is a directory exported via NFS from the login server to the **desktop PCs** (**/opt/evop/public/**) and can be accessed as **read-only** on each machine.

Executing graphical applications from a remote server

Notice that graphical applications like **firefox** or **kate** will not work from remote:



Executing graphical applications remotely is very demanding for the network, so by default we will use only the command line interface.

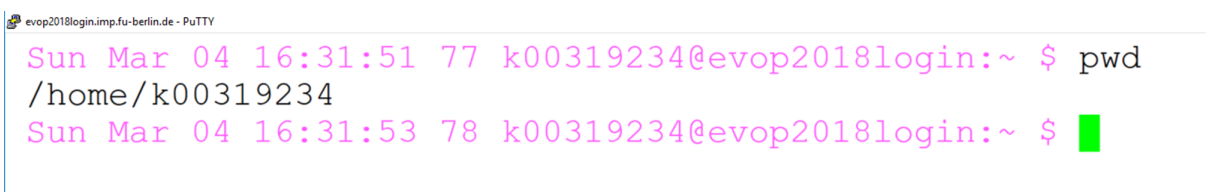
To run graphical application remotely, you would need to re-login on the server, using the **-X** option in ssh:

```
ssh -X k00nnnn##@evop2018login.imp.fu-berlin.de
```

Which folder am I?

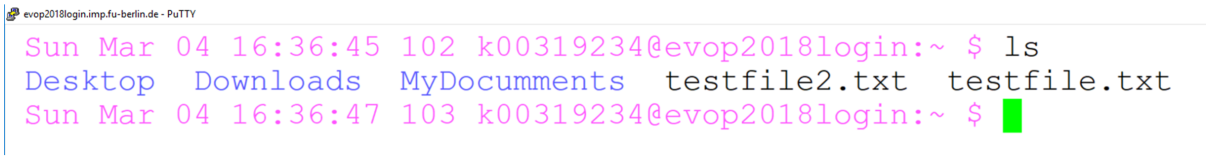
By default, when you login you are positioned in your home folder.

Use the **pwd** command to see where are you exactly:



Where can I go?

The **cd** command allows to navigate the file system, moving to a different folder.



The 'cd' man page

The **cd** command is documented inside the "bash" man page.

- Type **man bash** and then look for cd

You can also look at:

- **man dir**
- **info coreutils cd**

If you get lost: type "cd" without arguments

Typing **cd** without arguments will bring you back to your home directory.

Useful if you get lost in the file system!

How to refer to the current (.) and parent (..) folders

In Unix the current folder is also referred to as "."

- **cd .** will move you to.. the current folder again

The parent folder is referred to as ".."

- **cd ..** will bring you to the parent folder

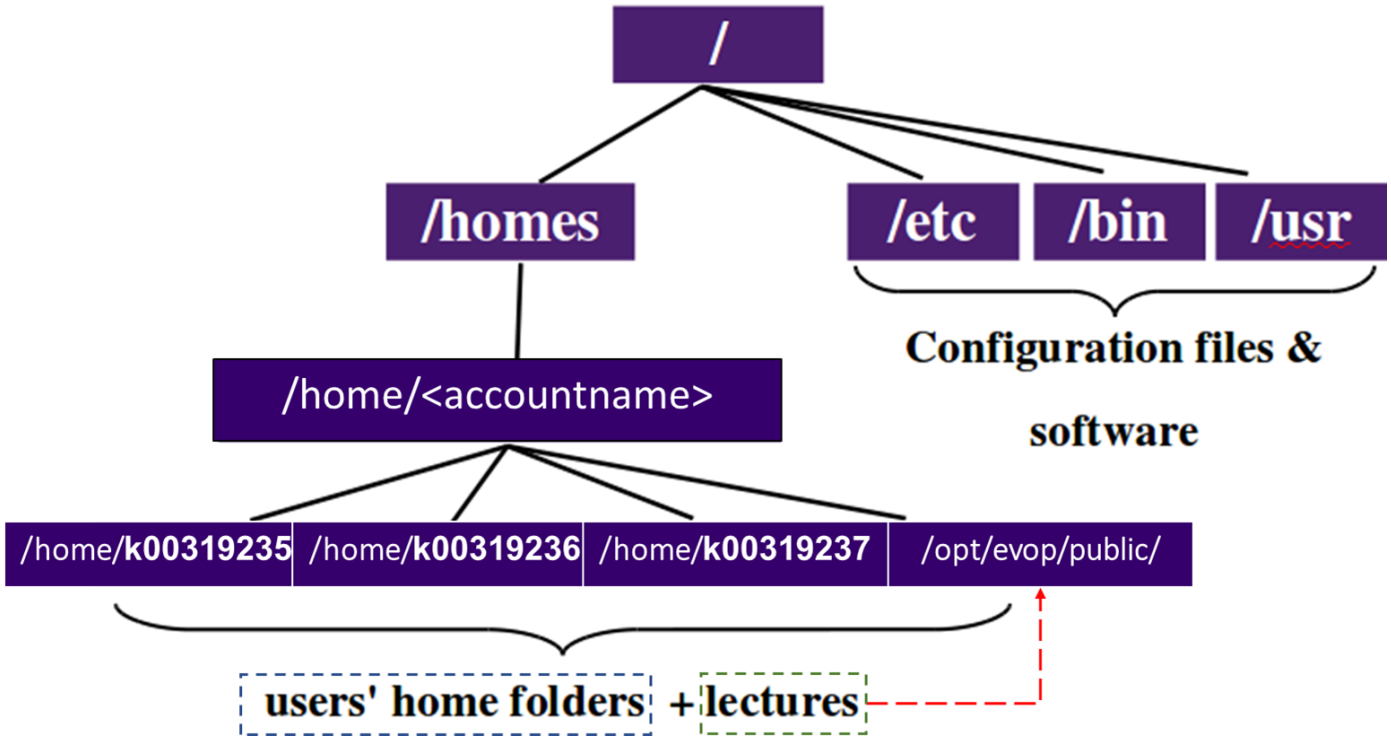
Structure of the Unix file system

Directories in a Unix file are organized as a "tree" of subdirectories.

The root folder, equivalent to **C:** in Windows, is indicated by a single slash, **/**

All the other folders are subfolders of the root, e.g. /home/k003192**##**

The Unix file tree



Contents of the /root directory

Type **ls /** to see the contents of the root directory:

- **/bin**, **/usr**, **/local**: all the software is usually installed here
- **/etc**, **/var**: system-wide configuration
- **/homes**, **/home**: Users' private directories

The course materials

The **/home/** folder contains a folder **k003192##** for each student in this course

There is also a special folder called **/opt/evop/public/**, containing the materials, slides, and **Data sets**

```
(/opt/evop/public/   )
```

All the materials for this course are in **/opt/evop/public/unix_intro**

```
( /opt/evop/public/unix_intro)
```

** NOTE**: ****/opt/evop/public/** directory is exported via NFS from the login server to the **desktop PCs** and can be accessed as **read-only** on these machines

Copy the course materials in your home

Let's copy the exercise files to your home folder:

New path:

```
$  cp -r /opt/evop/public/unix_intro ~
```

Old path:

```
$  cp -r /homes/evop2018/lectures/unix_intro/ ~
```

Explanation:

- **cp** is the command to copy files
- the **-r** (recursive) option is mandatory when copying folders
- **/opt/evop/public/unix_intro** is the location of the folder you want to copy
- **~** is a short-cut to represent your home folder

To make sure the files are copied correctly, you can do:

```
cd
cd unix_intro
ls
```

The exercises files

Let's move to the unix_intro/exercises folder to see the exercise files.

In [15]:

```
cd exercises
ls
```

```
1_browsing_textfiles.txt  chr8.gff            multiplefiles
2_searching_patterns.txt  exercise1_grep.txt  old_files
chr8.bed                  genes               sequences.fasta
```

There is a file called 1_browsing_textfiles.txt.

Accessing the contents of a file: head, cat, less

To access the contents of a file, we can use several Unix commands:

command	description	example
head	print the first lines of the file	head start_here.txt
tail	print the last lines of the file	tail start_here.txt
cat	print the contents of the file to the screen	cat start_here.txt
less	allows to navigate contents of the file	less start_here.txt

The **head** command can be used to see the first line of a file.

Let's use it on the file 1_browsing_textfiles.txt:

In [19]:

```
head 1_browsing_textfiles.txt
```

```
 ____________________________________________________________________________
_____________________
/ the Unix command "head" allows to see the first lines of a file.
\
|
|
| You can use the -n option to specify how many lines to show. For exa
mple, try printing |
| the first 40 lines of this file.
|
\
/
 ----------------------------------------------------------------------------
-------------------
        \   ^__^
         \  (oo)\_______
            (__)\       )\/\
```

The -n option in head

The first lines of the file seemed to suggest something. Let's use the -n 40 option to print the first 40 lines of this file:

In [20]:

```
head -n 40 1_browsing_textfiles.txt
```

```
 ____________________________________________________________________________
____________________
/ the Unix command "head" allows to see the first lines of a file.
\
|
|
| You can use the -n option to specify how many lines to show. For exa
mple, try printing |
| the first 40 lines of this file.
|
\
/
 ----------------------------------------------------------------------------
-------------------
        \   ^__^
         \  (oo)\_______
            (__)\       )\/\
                ||----w |
                ||     ||


 ____________________________________________________
/                                                    \
| Well Done!                                         |
|                                                    |
| Another way to see the contents of a file is       |
| using the less command. Try typing:                |
|                                                    |
|        less start_here.txt                         |
|                                                    |
| and scroll down with the arrow keys.               |
\                                                    /
 ----------------------------------------------------
        \   ^__^
         \  (oo)\_______
            (__)\       )\/\
                ||----w |
                ||     ||



++++++++++++++++++++++++++
```

Browsing the contents of a file with less

The **less** unix command allows to browse the contents of file, read-only, in a modality similar to what we saw with the **man** pages:

```
less 1_browsing_textfiles.txt
```

Scroll down and follow the instructions in the file to see more.

Editing files with nano

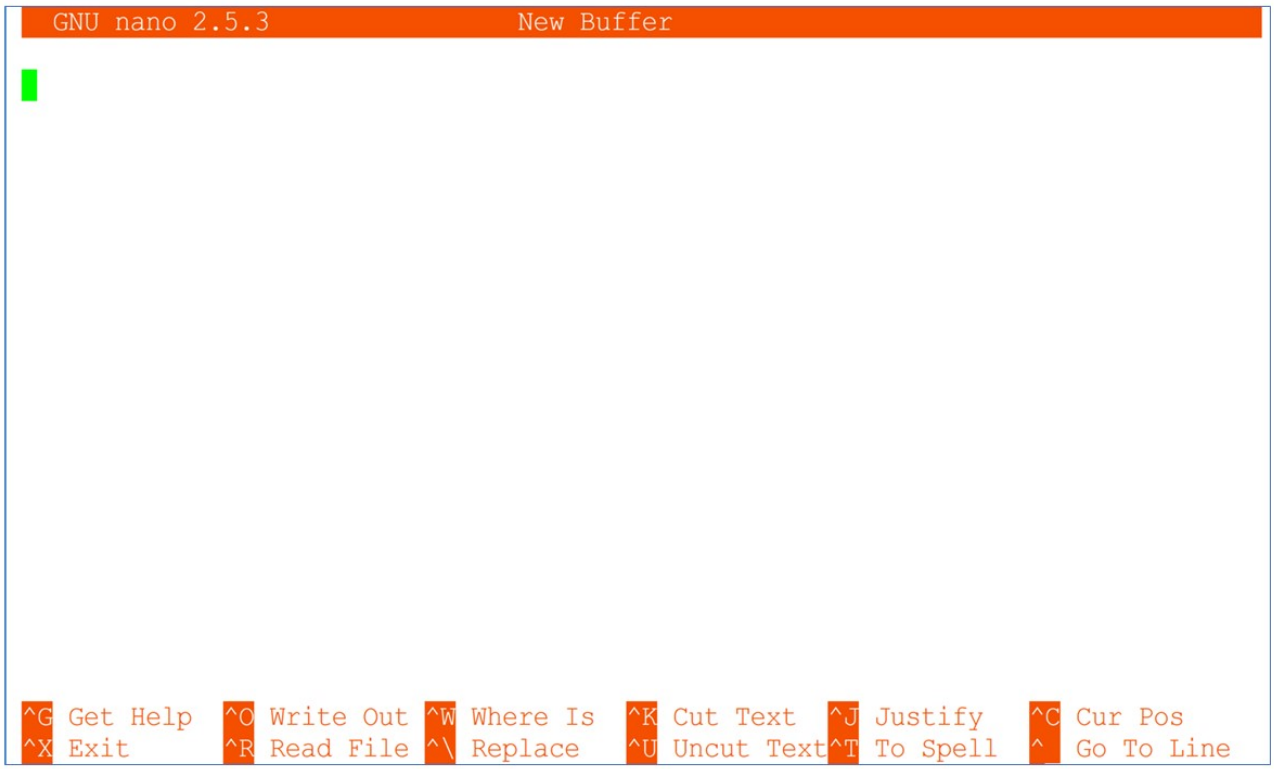
In the following days you will have to write several scripts into evop2018Server.

There are two alternatives:

- connect using the -X option, then use **gedit** -> however if too many people use it, the network will crash
- **nano** is a powerful command-line editor

The nano text editor

Type **nano** on the command line to access the text editor:



How to use nano

Type **nano** on the command line to access the text editor

```
type nano filename.txt to open a file directly
```

Keyboard shortcuts:

```
- save a file: CTRL-o
- read a file: CTRL-r
- exit: CTRL-x
- undo/redo: ALT-u and ALT-e
- Copy/paste: CTRL-SHIFT-c and CTRL-SHIFT-v
```

Nano exercise

Create a file named "myfile.txt"

Write "hello world"

- Save it using the CTRL-o shortcut
- Exit the editor using CTRL-x

Other useful Unix commands (moving and copying files)

- **mkdir**: create a folder
- **rm**: delete files (note: no trash bin!)
- **mv**: move or rename a file (note: overwritten files will be lost)

Other Unix commands

- **clear**: clear the terminal output
- **history**: see all the commands typed so far
- **dos2unix**: clean files edited in MS Windows notepad for Unix (see also unix2dos)
- **echo**: print a message (useful in scripts)

Time for lunch!

See you at 14.00!