

Unix - Lab

Boris Steipe

boris.steipe@utoronto.ca

<http://biochemistry.utoronto.ca/steipe>

Department of Biochemistry
Department of Molecular and Medical Genetics
Program in Proteomics and Bioinformatics
University of Toronto

Tutorial 2.1



<http://creativecommons.org/licenses/by-sa/2.0/>



The image is a yellow rectangular box containing the Creative Commons Attribution-ShareAlike 2.0 license summary. At the top center is the Creative Commons logo (two 'C's in a circle) followed by the text 'creative commons' in a bold, sans-serif font, and 'COMMONS DEED' in a smaller, spaced-out font below it. Underneath is the title 'Attribution-ShareAlike 2.0'. The text is organized into sections: 'You are free:' followed by a bulleted list of permissions; 'Under the following conditions:' followed by two icons in circles (a 'BY' and a circular arrow) with their respective descriptions; a bulleted list of additional conditions; a statement about fair use; and a link to the full legal code with a disclaimer icon.

creative commons
COMMONS DEED

Attribution-ShareAlike 2.0

You are free:

- to copy, distribute, display, and perform the work
- to make derivative works
- to make commercial use of the work

Under the following conditions:

 **BY: Attribution.** You must give the original author credit.

 **Share Alike.** If you alter, transform, or build upon this work, you may distribute the resulting work only under a license identical to this one.

- For any reuse or distribution, you must make clear to others the license terms of this work.
- Any of these conditions can be waived if you get permission from the copyright holder.

Your fair use and other rights are in no way affected by the above.

This is a human-readable summary of the [Legal Code \(the full license\)](#).

[Disclaimer](#) 

UNIX

UNIX is an **operating system**.

UNIX is actually a number of different operating systems. Though, from a user's perspective, they are very similar.

Flavors of UNIX include:

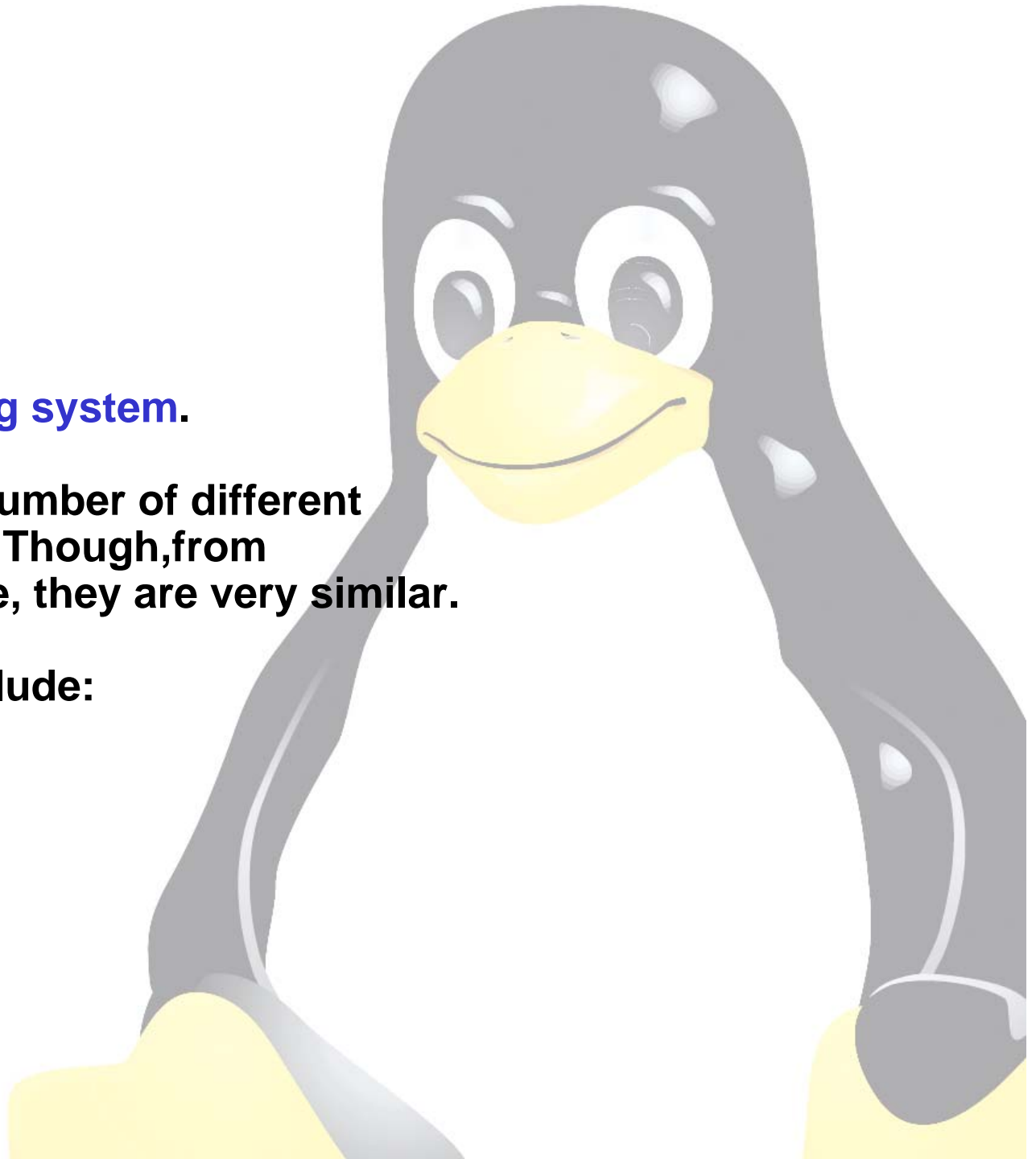
Solaris (Sun)

AIX (IBM)

FreeBSD (Free)

Linux (Free)

[...]



Characteristics of UNIX

Multitasking: The first and still the most stable

Multiuser: Down to the concurrent use of files.

Programming and customization: Programming environments and libraries available, scriptable shell, automated tasks ...

Stable, Secure, (Open and Free): But high quality GUIs are available, second to no GUI-only system.

UNIX Architecture

A **layered architecture** negotiates between hardware and software.

Applications

Text processor, Web-Browser, Programs ...

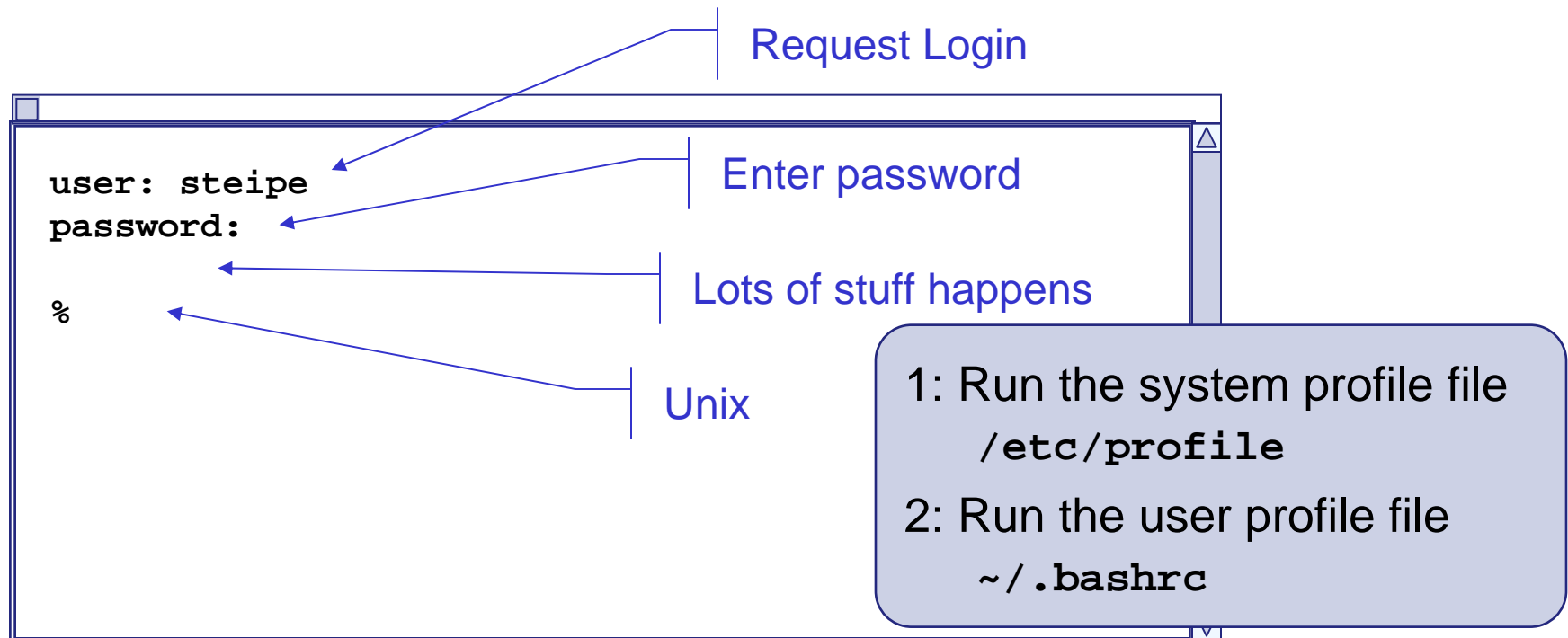
Shell

Interacts with user and kernel
...

Kernel

Interacts with hardware, schedules processes, manages memory and storage
...

What happens when you startup a shell ...



Clues to make Linux easier

- Copying and pasting text: use the mouse!
 - highlight the text you want to copy and press both mouse buttons to paste
 - This is due to the three button mouse system
 - On 2-button mouse, both buttons simultaneously act as the third button
 - This will work for all visible text
 - This works in UNIX as well
- Alternatively **Control-c** (copy) and **Control-v** (paste) may be **Alt-c** and **Alt-x** in Linux

Customizing your shell

- Special BASH files you will find in your home directory

<code>.bash_profile</code>	executed at login
<code>.bashrc</code>	executed at shell startup
<code>.bash_logout</code>	executed at logout
<code>.bash_history</code>	contains your previous commands

- Type: **ls -al** to see all your files
- Customizing your shell usually will involve editing the `.bashrc` file

Aliases: making HOME feel like home

- aliases can make your life easier
- typing **alias** will reveal 'default aliases'
- you can add temporary aliases
- alias lf='ls -F | less'
- unalias will remove this: unalias lf
- you can add permanent aliases to your .bashrc file

history

- Scroll through your command history with ↑↓
- Look at your history file by typing **history**
- history lists your previous commands next to a number.
- you can rerun any given number by typing **!*n***

*(!**55** would re-execute command 55 in my history)*

Clues to make Linux easier

- **UNIX is case sensitive**

- certain characters in UNIX have special meaning.

<	>	'	"	*	{ }	[]	
\	!	()	\$?	~	&	#

- Avoid using these in file and directory names
- Also avoid using **spaces** in a name
- If you have files with such characters you must use **quotes** or escape the character by preceding it with \ when interacting with them (it's a hassle)
- Instead use the **underscore** and **period** to break up names

for example its better create a directory:

```
mkdir my_dna
```

instead of:

```
mkdir "my dna"
```

Clues to make Linux easier:

- Completion ...
 - by pressing **[tab]** when entering commands and filenames the UNIX shell will try to complete the word.
 - This is very useful with filenames in UNIX and emacs
- The wildcards * and ?
 - Type: **ls unix*** to list all files starting with unix
 - Type: **ls *unix** to list all files ending in unix
 - Type: **ls ?ook** to list any 4 letter files ending in ook (I.e. cook, book but not crook)

Get a PDB coordinate file

Do the following:

Open a Web browser

Navigate to <http://www.rcsb.org/pdb>

Search for PDB-ID **1JKZ**

Click on "**Download/Display File**"

Move to the table "**Download the Structure File**"

Click on the "**X**" for PDB format, Compression "none"

Save the file

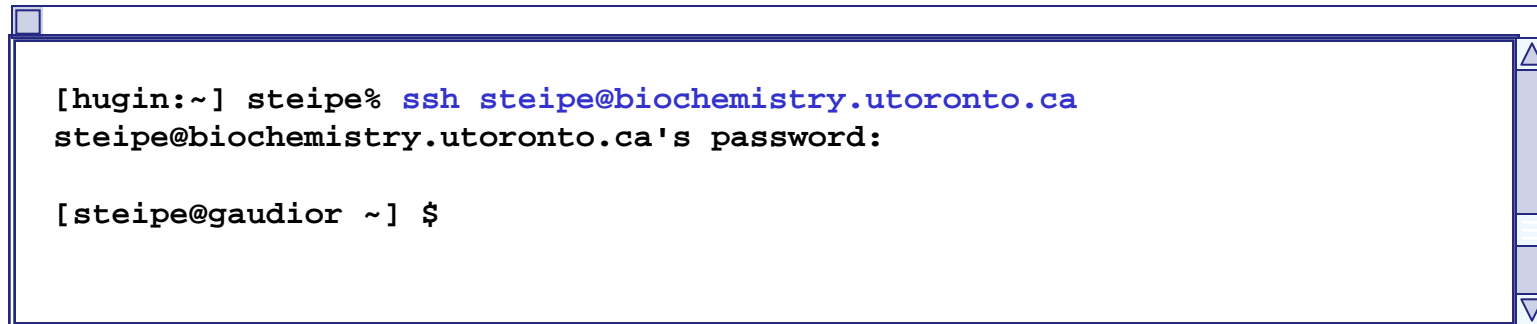
Ftpping –file transfer protocol fun

- ftp [sitename]
 - example **ftp ftp.ncbi.nih.gov**
 - asked for username and password
 - list and move into directories (use **ls** and **cd**)
 - change to binary (type **bin**)
 - type **get filename** to get a file
 - or **mget filename1 filename2 filenameN**
 - type **put filename** to put (upload) a file
 - or **mput filename1 filename2 filenameN**

ssh protocol suite

secure shell

Supersedes Telnet. IP based. Encrypts all traffic. Free. Open. Exportable.



```
[hugin:~] steipe% ssh steipe@biochemistry.utoronto.ca
steipe@biochemistry.utoronto.ca's password:

[steipe@gaudior ~] $
```

The ssh2 protocol is state-of-the-art cryptography. You need this, maybe less because you want to encrypt your files, but because sysadmins want to secure their machines and increasingly disable all other access protocols.

scp: secure version of UNIX copy command

sftp: secure implementation of ftp

Implemented in **PuTTY**, MacOS X ...

Environment variables

Environment variables are settings that control specific aspects of the system

- each program has its own variables
- the shell (a program we spend a lot of time with) has variables too
- you can look at all your shell environment variables by typing set

Some common environment variables for the shell

- HOME
 - points to the home directory
- PATH
 - places that the shell will look for your commands
- PS1
 - determines what is displayed at the command prompt

Customizing environment variables

- variables can be used in commands and filenames by preceding them with a \$ sign
 - i.e. \$HOME/unixtutorial/unix.txt
- you can use \$PATH in a command to add a new directory to your PATH
 - export PATH=\$PATH:\$HOME/unixtutorial

export makes shell variable PATH global

\$PATH refers to current path

: separates current PATH from new one

new PATH to be added

- this appends the new PATH to the existing one
- to view your PATH: echo \$PATH

Installing programs in Linux/Unix

- Unix and Linux comes with many applications preinstalled
- You will likely want to install more
- This can sometimes prove to be difficult for a few reasons some of which include:
 - what flavour of unix/linux you are running
 - what privledges you have as a user (i.e. root)
 - the nature and quality of the program you want to install
- Much softwares builds upon previous work, programs, libraries etc.
- Sometimes you need to find and install these yourself.
- Depending on the nature of the program and implementation this can become a really annoying scavenger hunt.
- **The art of installation can take patience and practice**

Making and breaking tar files

- Create a tar file of everything inside of a directory
 - Type: **tar -cvf filename.tar directoryname**
- File is now **filename.tar**
 - Tar is short for tape archive
 - It is mostly used to bundle files and directories
- To untar a tar file
 - Type: **tar -xvf filename.tar directoryname**
 - When extracting make sure the directory you choose is where you want the files (could be messy)

c	Creates (a new tar file)
x	Extracts (a tar file)
f	Files
v	Verbose (lets you see files created)
z	Use (tar -xzvf) to uncompress and untar in one step

Un/compressing files

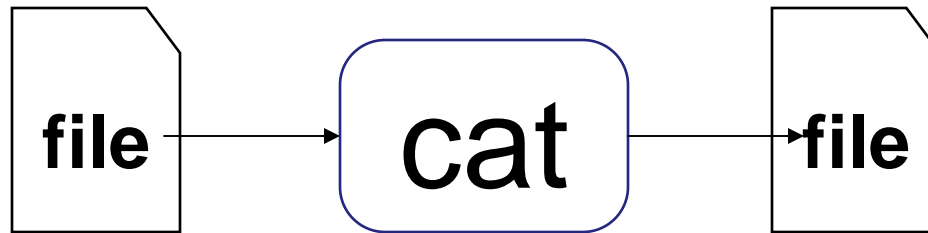
- Most UNIX software you download will be compressed and bundled.
- These files often look like: **filename.tar.gz**
 - To uncompress type: **gunzip filename.tar.gz**
 - File is now **filename.tar**
- To compress a file type: **gzip filename**
 - You then get a smaller file named: **filename.gz**

A review of the essential

- **cd** – change directory
- **ls** – list
- **cp** – copy a file
- **mv** – move or rename a file
- **rm** – remove a file (use caution)
- **mkdir** – make a directory
- **rmdir** – remove a directory (must be empty)
- **pwd** – print working directory

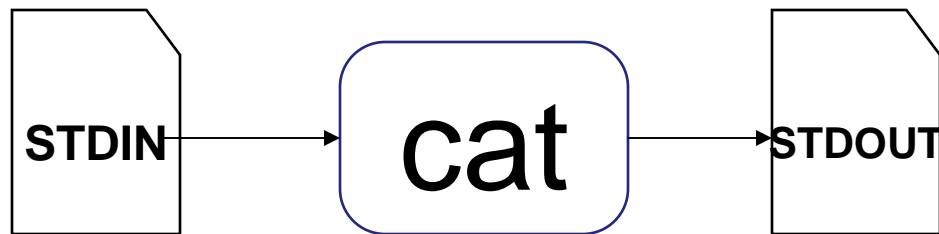
In unix, everything is a file ...

UNIX commands usually read from a file, do something and write the result to a file ...



But in UNIX actually **everything** is engineered to behave like a file. Files, disks, network connections, terminals ... and most importantly: if nothing is specified, UNIX uses default filehandles to pass data: **STDIN** and **STDOUT**

STDIN and STDOUT



Normally **STDIN** is associated with the **keyboard** and **STDOUT** is associated with the **terminal** ...



Type: wheee ...

Display: wheee ...

> >>

STDOUT can be redirected to "real" files ... with amazingly useful results ...

```
$ cat > joy.txt  
Wheee....  
<ctrl D>  
$ cat joy.txt  
Wheee....  
$ cat >> joy.txt  
YeeeHaw !  
<ctrl D>  
$ cat joy.txt  
Wheee....  
YeeeHaw !  
$
```

← | > ... create or
overwrite

← | >> ... append



... same as STDIN

```
$ cat < joy.txt  
Wheee....  
YeeeHaw !  
$ cat < joy.txt > joy2.txt
```

← and what would this do ???

The most useful thing about Unix:



... the | tutorial.



Some Linux and Unix resources

- <http://www.genome.washington.edu/uwgc/unix/unix.htm>
- <http://www.ee.surrey.ac.uk/Teaching/Unix/index.html>
- http://stein.cshl.org/genome_informatics/unix1/index.html
- <http://www.linux.org/>
- <http://www.justlinux.com/>
- <http://jgo.local.net/LinuxGuide/>
- <http://tldp.org/>
- <http://www.linuxchix.org/>
- <http://www.utexas.edu/cc/unix/index.html>
- <http://www.gnu.org/>

<http://www.google.com>

UNIX tutorials:

UNIX Tutorial for Beginners

<http://www.ee.surrey.ac.uk/Teaching/Unix/>

A Basic UNIX Tutorial

[http://www.isu.edu/
departments/comcom/unix/workshop/unixindex.html](http://www.isu.edu/departments/comcom/unix/workshop/unixindex.html)

The Linux Cookbook

<http://www.ibiblio.org/obp/cookbook/>