

# An Introduction to Linux

CS101

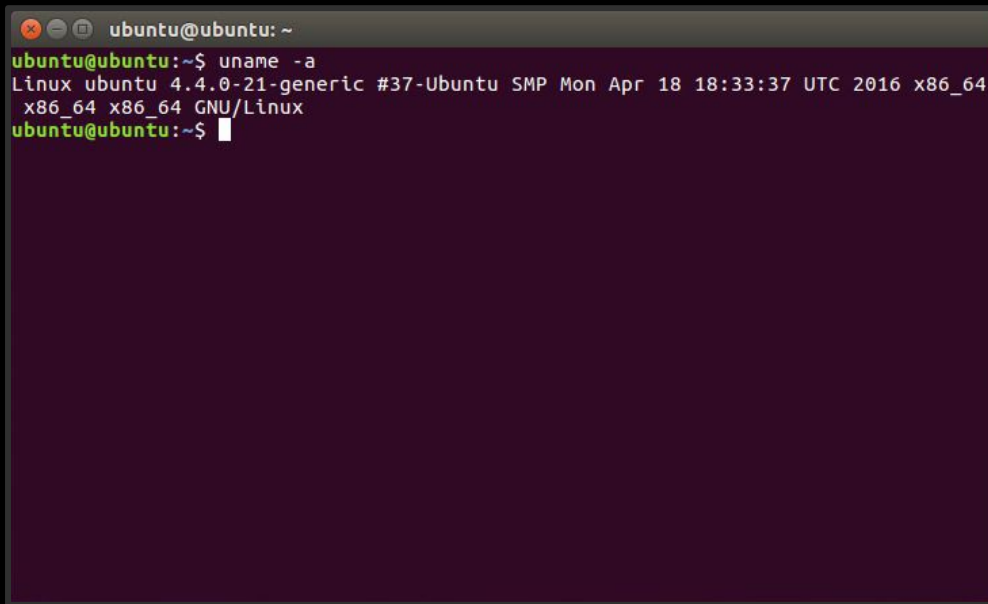
# What is Linux?

Linux is a family of operating systems like Windows or macOS, but it is free (as in food) and open source (the source code is publicly available).

All of the lab machines in UTCS run on Linux (Ubuntu with the GNOME Desktop Environment, if you want to be specific).

# The command line / shell / terminal /..

These words all technically have different definitions, but most of the time, people use it to refer to the same thing: the program where you type in commands to interact with your computer.

A terminal window with a dark purple background and a grey title bar. The title bar contains window control icons and the text 'ubuntu@ubuntu: ~'. The terminal shows the command 'uname -a' being executed, with the output: 'Linux ubuntu 4.4.0-21-generic #37-Ubuntu SMP Mon Apr 18 18:33:37 UTC 2016 x86\_64 x86\_64 x86\_64 GNU/Linux'. The prompt 'ubuntu@ubuntu:~\$' is visible at the end of each line.

```
ubuntu@ubuntu: ~  
ubuntu@ubuntu:~$ uname -a  
Linux ubuntu 4.4.0-21-generic #37-Ubuntu SMP Mon Apr 18 18:33:37 UTC 2016 x86_64  
x86_64 x86_64 GNU/Linux  
ubuntu@ubuntu:~$
```

# Anatomy of the terminal

The image shows a terminal window with a dark purple background. The window title is 'ubuntu@ubuntu: ~'. The prompt is 'ubuntu@ubuntu:~\$'. The command 'uname -a' is entered and highlighted with an orange box. The output is 'Linux ubuntu 4.4.0-21-generic #37-Ubuntu SMP Mon Apr 18 18:33:37 UTC 2016 x86\_64 x86 64 x86 64 GNU/Linux', which is also highlighted with an orange box. Below the prompt, the components of the prompt are annotated: 'ubuntu' is the username, 'ubuntu' is the name of machine/host, and '~' is the current directory (folder), in this case the home directory (~).

```
ubuntu@ubuntu: ~  
ubuntu@ubuntu:~$ uname -a  
Linux ubuntu 4.4.0-21-generic #37-Ubuntu SMP Mon Apr 18 18:33:37 UTC 2016 x86_64  
x86 64 x86 64 GNU/Linux  
ubuntu@ubuntu:~$
```

Annotations:

- command that was run
- output of that command
- current directory (folder), in this case the home directory (~)
- name of machine/host
- username

# Some vocab

Directory Folder

Home Directory The 'main' folder for your user, usually represented with '~'

Root Directory The topmost folder of your filesystem, usually represented with '/'

Host The machine you are connected to

Root/Superuser The 'admin' user, has no restrictions (on the lab machines, this is the department, on your own machine it's you!)

# Basic commands

<code>ls</code>	<code>ls</code>	<u>L</u> ist the files in your current directory
<code>pwd</code>	<code>pwd</code>	Show your <u>p</u> resent <u>w</u> orking <u>d</u> irectory
<code>cd [a]</code>	<code>cd [a]</code>	<u>C</u> hange <u>d</u> irectories to [a], cd by itself returns to ~ (home)
<code>touch [a]</code>	<code>touch [a]</code>	Make the file [a]
<code>cat [a]</code>	<code>cat [a]</code>	Print the contents of [a] <small>(technically it's made to concatenate files but it's mainly used to print the contents)</small>
<code>mkdir [a]</code>	<code>mkdir [a]</code>	<u>M</u> ake the <u>d</u> irectory [a]
<code>mv [a] [b]</code>	<code>mv [a] [b]</code>	<u>M</u> ove file/dir [a] to file/dir [b] (also used to rename)
<code>cp [a] [b]</code>	<code>cp [a] [b]</code>	<u>C</u> opy file/dir[a] to file/dir [b]
<code>rm [a]</code>	<code>rm [a]</code>	Remove the file [a] (to remove a folder, add -rf flags)
<code>man [a]</code>	<code>man [a]</code>	Show the <u>m</u> anual pages for the command [a]

# Anatomy of a Linux command

```
rm -rf foobar.txt /foo
```

## command

The command to be run, in this case, remove.

## flags

Optional settings you can turn on for this command. They are always preceded by a '-'. Here the 'r' means recursive (delete all sub-directories and files) and the 'f' means force (don't confirm that I want to delete these files).

## arguments

The inputs the command needs. Here, these are the files/dirs to delete. Arguments are separated by spaces

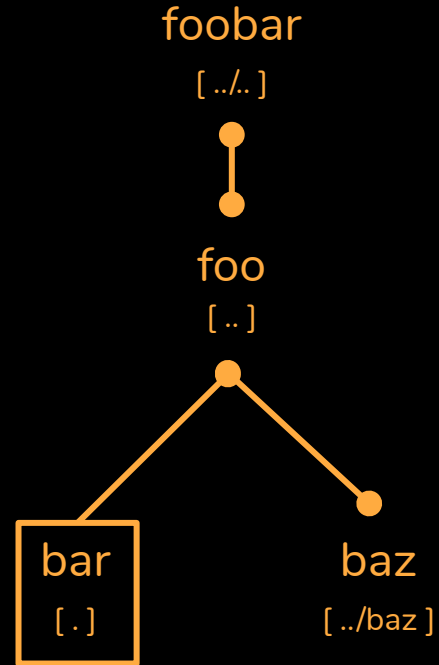
# Directories

- . Present working directory
- .. Parent directory of present working directory

`cd ..` Move up a directory to parent directory

`ls ..` List contents of parent directory

`ls ../bar` List contents of sibling directory foo





# Chaining commands

`foo && bar` Run 'foo' command, and if it finishes successfully, then run 'bar'

`foo; bar` Run 'foo' command, and then run 'bar', whether or not 'foo' finished successfully

# A challenge!

0. Log into lab machine
1. Navigate to home directory
2. Make a directory called **test**
3. Navigate into **test** (and check your present working directory)
4. Make a file called **foo**
5. Run **echo 'hello' > foo** to put 'hello' into the file **foo**
6. Copy the file **foo** to the file **bar**
7. List the files in the directory to check you have two files
8. Move the file **bar** to the file **baz** to rename it
9. Remove the file **baz**

Questions?

Thanks for coming!