

Linux command line basics I: files and folders

Yanbin Yin

<http://cys.bios.niu.edu/yyin/teach/PBB/Yin-chapter1.pdf>

http://korflab.ucdavis.edu/Unix_and_Perl/current.html

Homework #6

Create a folder under your home called hw6; pwd to find out where you are

Change directory to hw6; pwd to find out where you are

Copy the entire folder Unix_and_Perl_course from my home (/home/yyin/Unix_and_Perl_course) to your working folder (the folder where you currently are). Google or man or check my slides if you do not know how to copy a folder.

Change directory back to your home

Use ls to explore the folder Unix_and_Perl_course and the sub-folders in there to locate the At_genes.gff file and the unix_and_perl_v3.1.1.pdf file. Tell me what is the size and creation date of these files.

Change the permission of the At_genes.gff file so that every user can edit this file.

Write a report (in **word or ppt**) to include all the operations and screen shots.

Due on Oct 24 (send by email, if there are 2+ files, put them in a zip file; **include your last name in the file name**)

Use Virtual private network (VPN) to connect to our Linux sever from home

Use Internet Explorer or Firefox

Go to <https://secure.niu.edu>

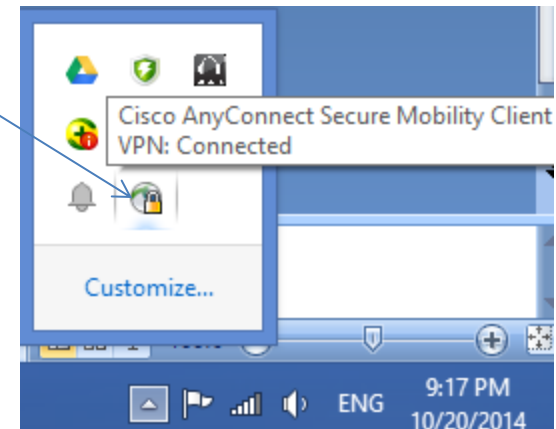
Put in your Z id and password

The next page will install CISCO AnyConnect Secure Mobility software

It will ask you to install JAVA if you don't have it yet

After that you have an icon at the right bottom corner, showing the VPN software

And you will be able to access the **10.157.217.25** Linux server (Ser)



Understand Linux/Unix and shell

File system

Basic shell commands

Commands for text processing

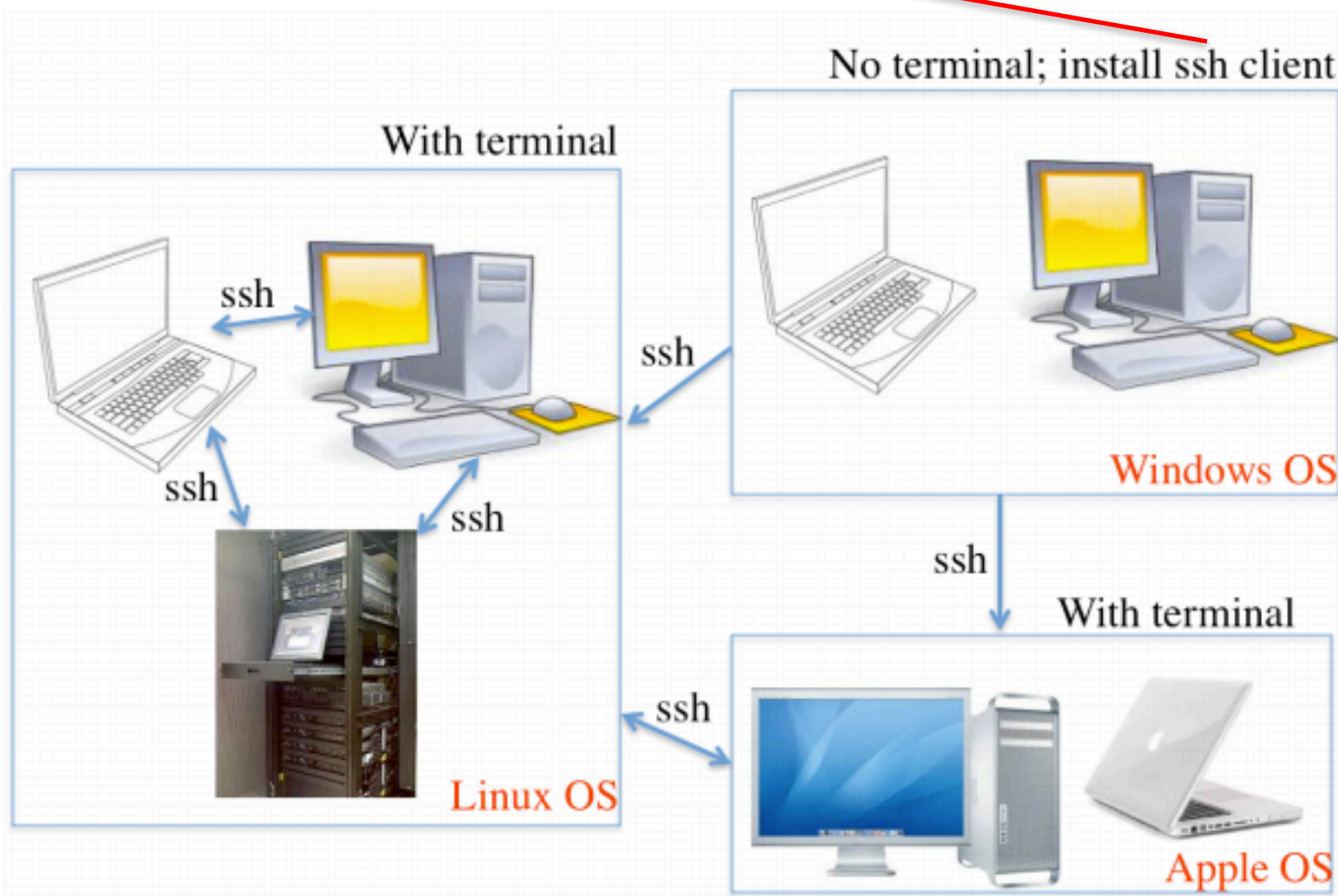
Pipe: chain multiple commands to get text processing pipeline

Shell one-liner

There's always more than one way to do it

Putty, ssh secure shell client, <https://the.earth.li/~sgtatham/putty/latest/w32/putty.exe>

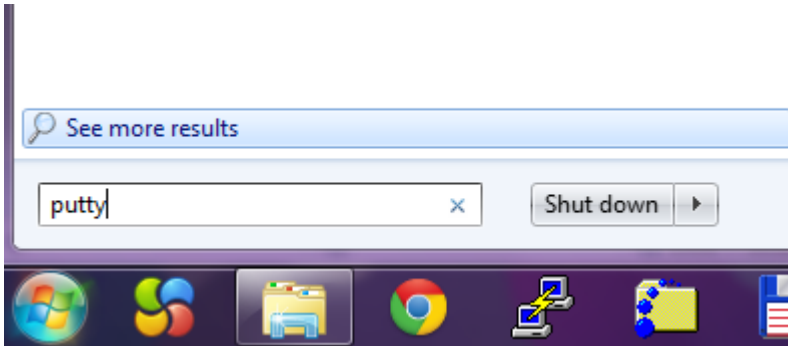
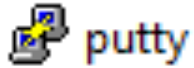
Figure 1.5 Different ways to connect to Unix terminal environment



What is ssh?

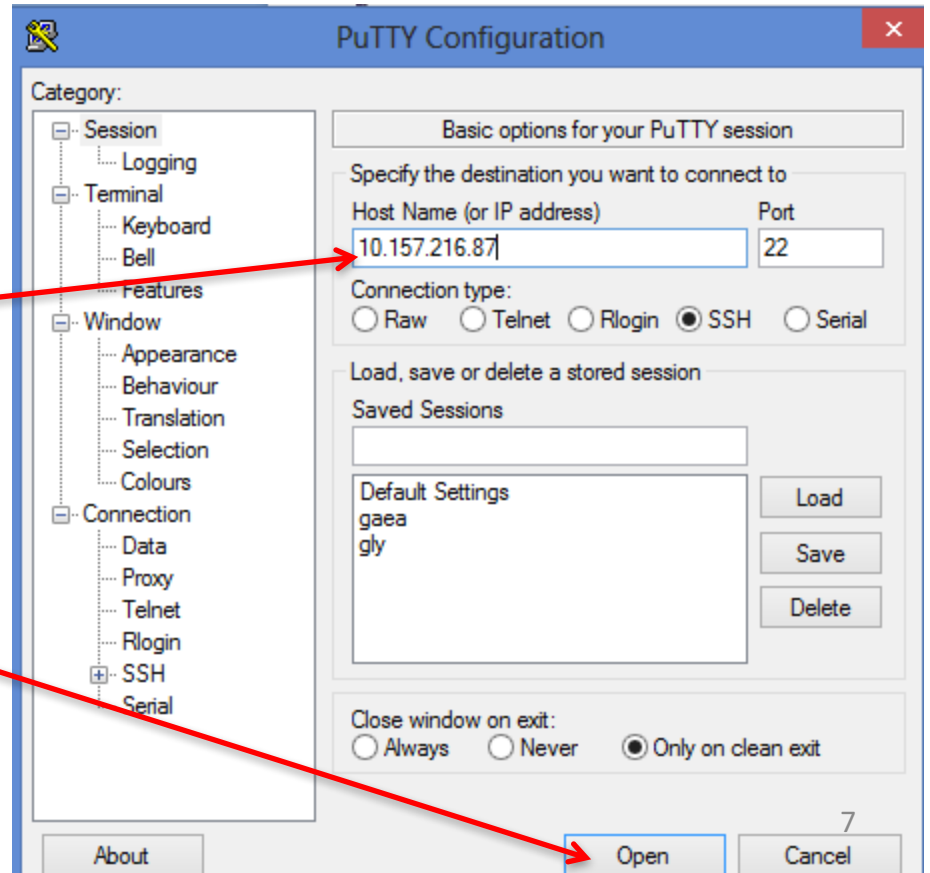
Secure Shell (SSH) is a cryptographic **network protocol** for secure data communication, **remote command-line** login, remote command execution, and other secure network services between two networked computers

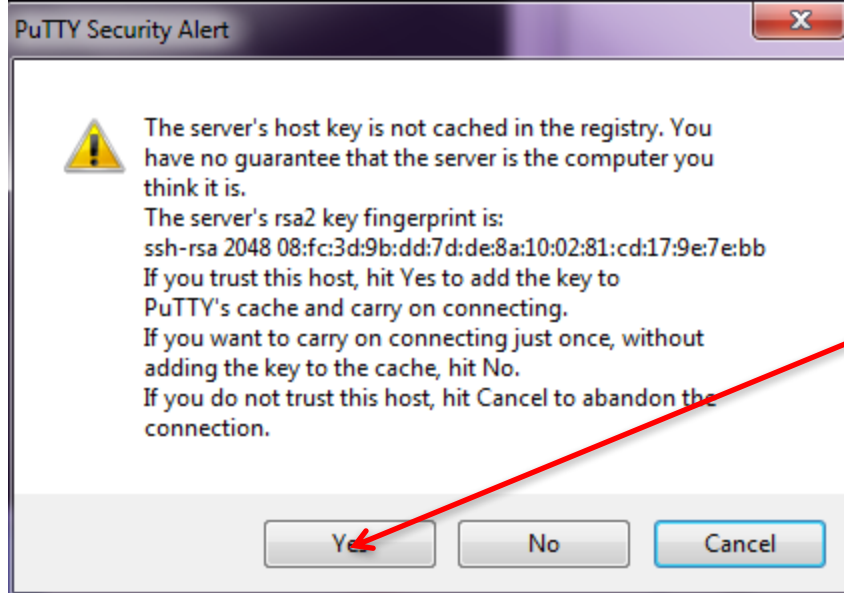
go to the left bottom, search “putty”



Put the IP address:
10.157.217.25

Hit Open





Hit Yes

Put your student ID here (e.g. **z01553886**) and hit enter

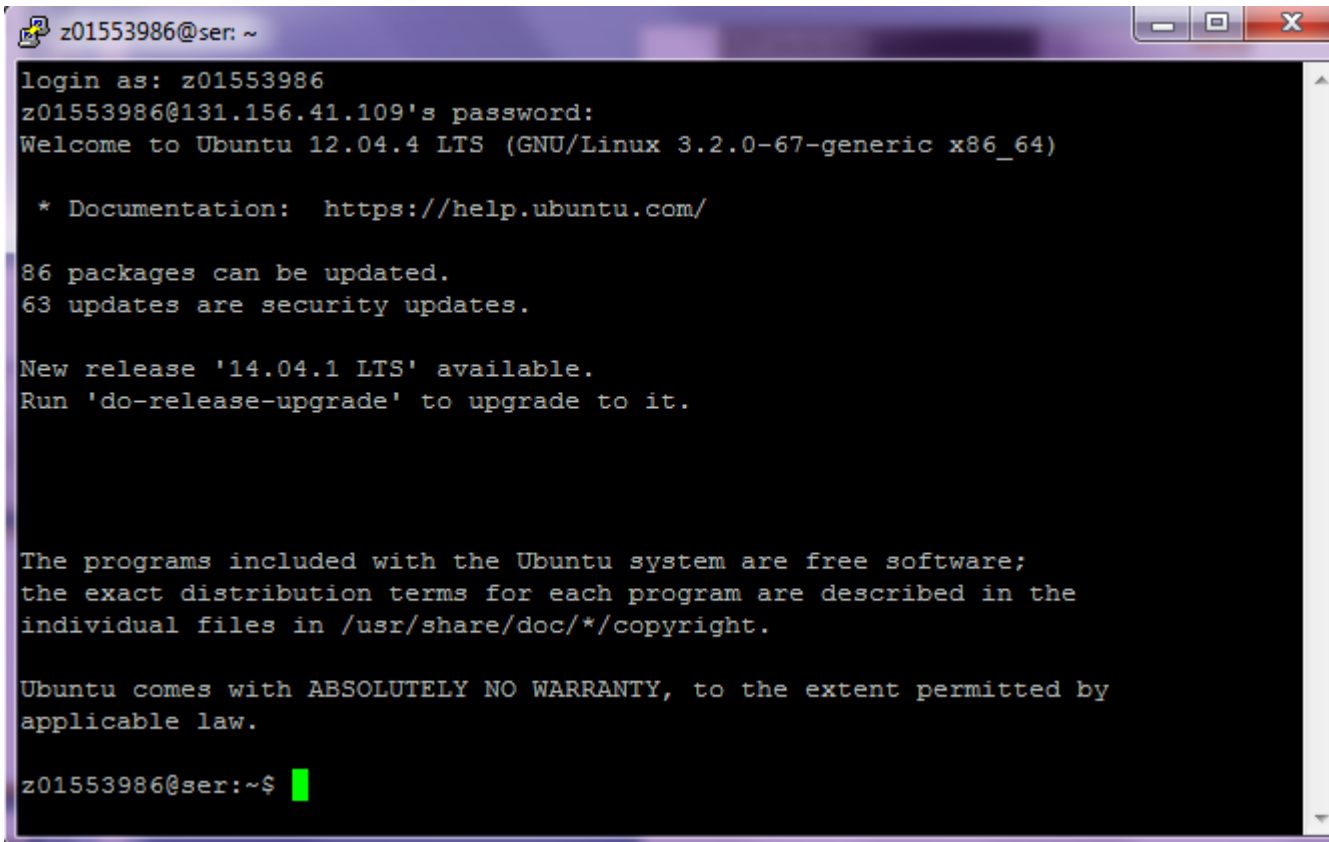


Wait until you see the following, then put your password (the same as your student ID) Your password will be invisible while you are typing. Hit enter after you are done



Your user name and password are both your student ID (lower case z, then 0, then your ID)

If you have put your password correctly, you will see this, meaning you are logged in now:



```
z01553986@ser: ~  
login as: z01553986  
z01553986@131.156.41.109's password:  
Welcome to Ubuntu 12.04.4 LTS (GNU/Linux 3.2.0-67-generic x86_64)  
  
* Documentation:  https://help.ubuntu.com/  
  
86 packages can be updated.  
63 updates are security updates.  
  
New release '14.04.1 LTS' available.  
Run 'do-release-upgrade' to upgrade to it.  
  
The programs included with the Ubuntu system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*/copyright.  
  
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by  
applicable law.  
  
z01553986@ser:~$
```

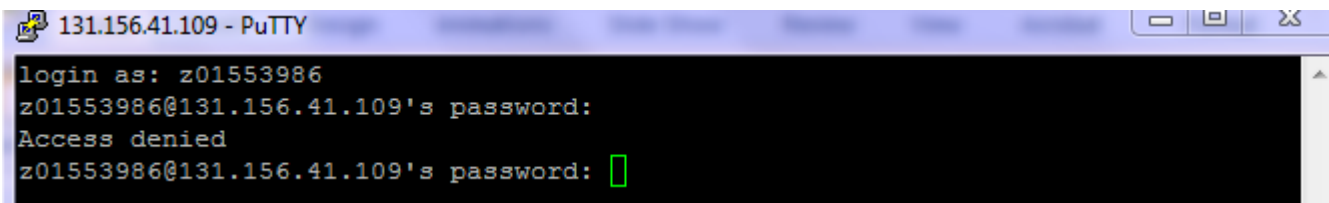
A plain text world

Everything is text,
no images

Good:
Minimal
consumption
from graphics, all
resources (CPU,
RAM, hard drive)
saved for
computation

If you did NOT put your password correctly, you will see this:

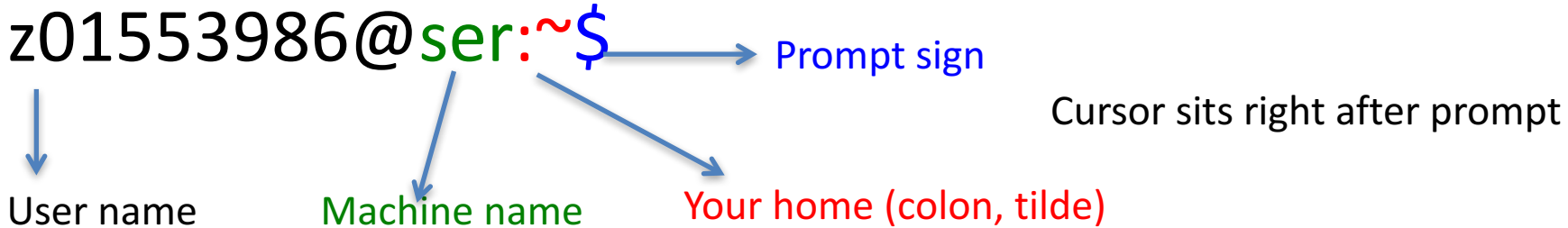
Just do it again



```
131.156.41.109 - PuTTY  
login as: z01553986  
z01553986@131.156.41.109's password:  
Access denied  
z01553986@131.156.41.109's password: 
```

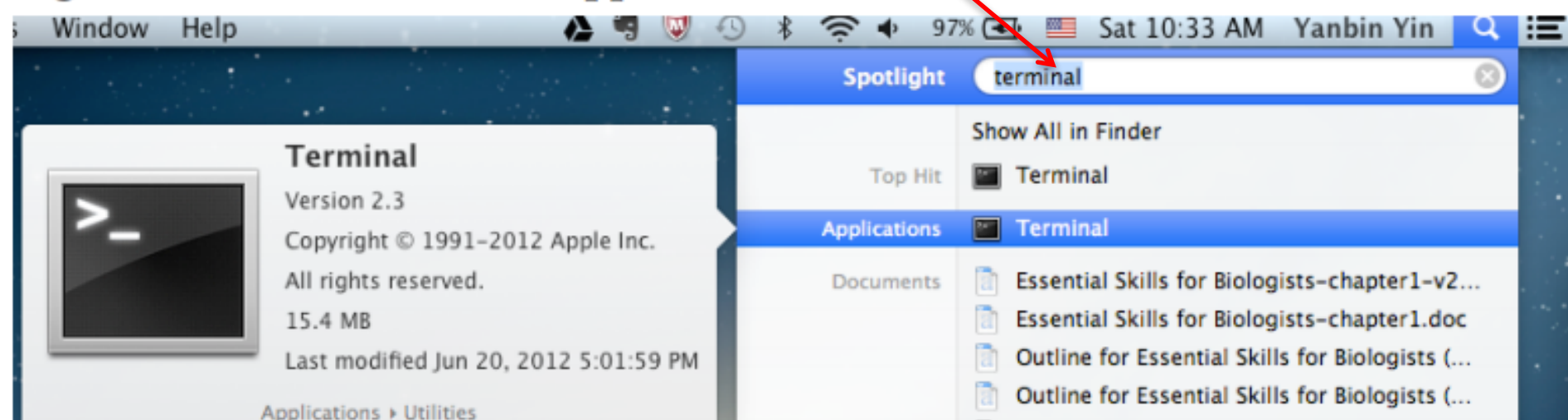
Bad:
User unfriendly

What you see after you successfully logged in



MAC has its built-in terminal. Search terminal application

Figure 1.2 Find the terminal application in MAC OS



Type in this part, hit enter and then yes, put password, and then you are connected

```
Yanbins-MacBook-Pro:~ yanbinyin$ ssh z01553986@10.157.217.87
z01553986@10.157.217.87's password:
Welcome to Ubuntu 12.04.4 LTS (GNU/Linux 3.2.0-67-generic x86_64)

 * Documentation:  https://help.ubuntu.com/

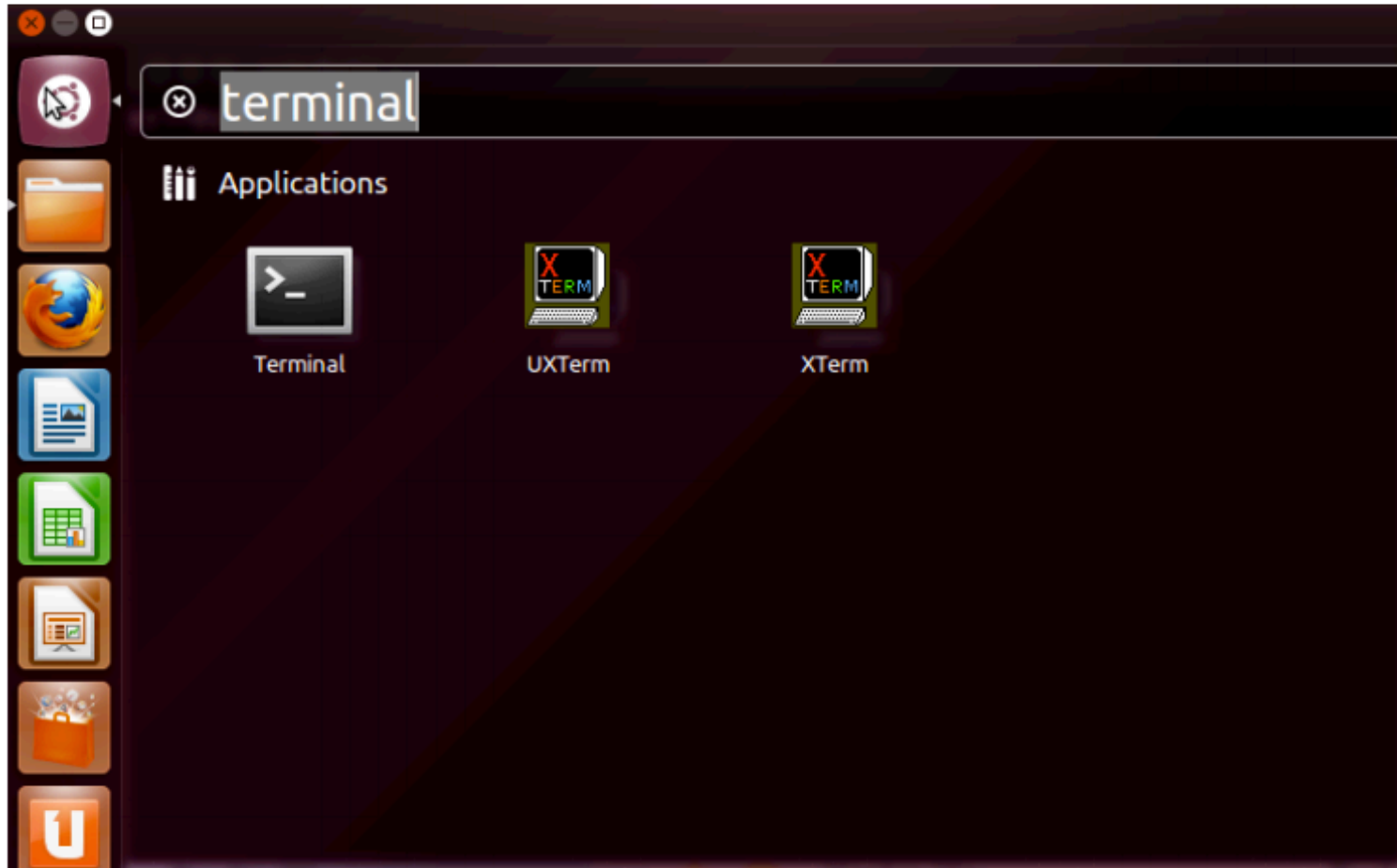
90 packages can be updated.
67 updates are security updates.

New release '14.04.1 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Mon Oct 13 05:31:35 2014 from localhost
z01553986@ser:~$
```

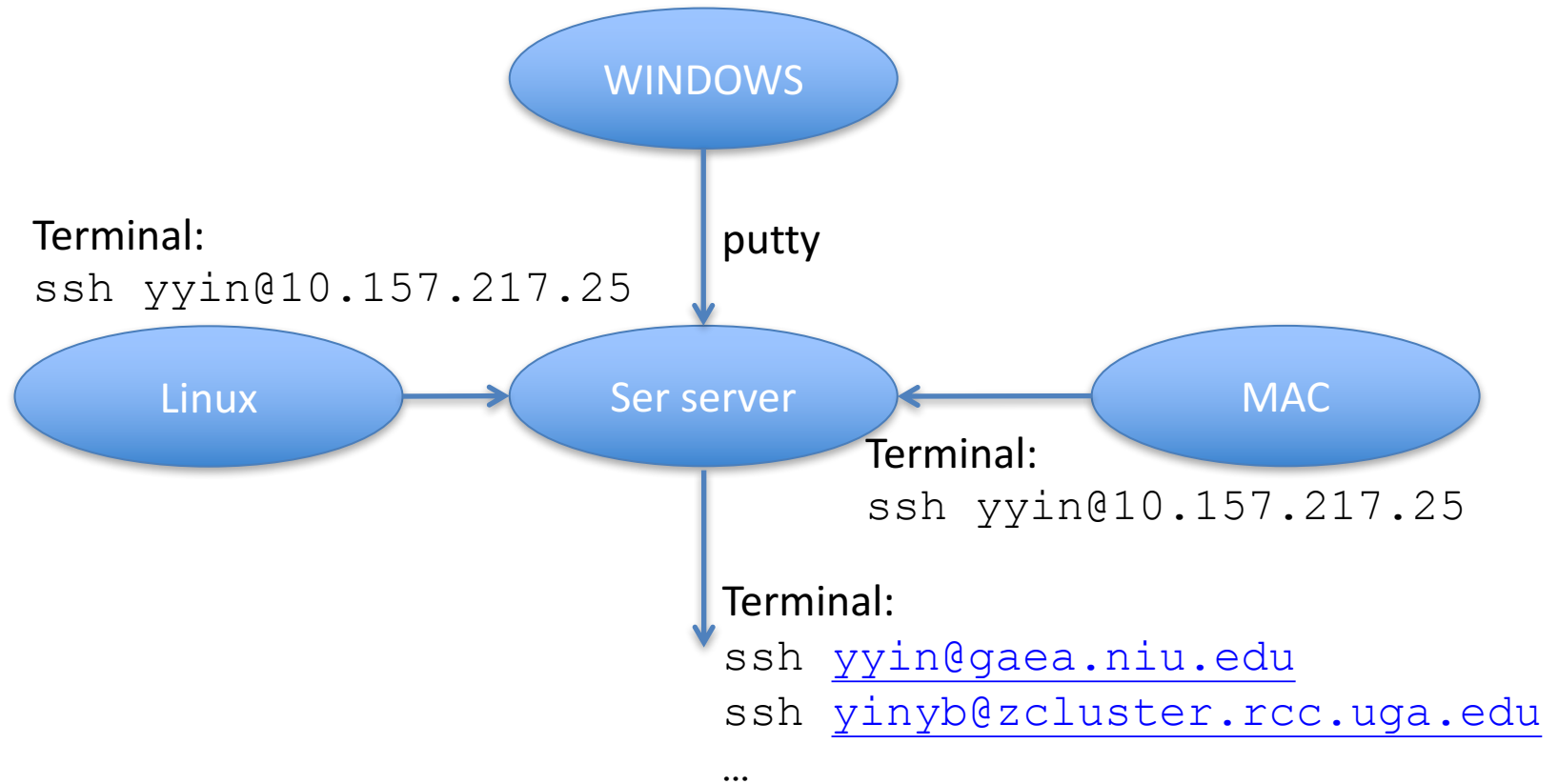
Ubuntu has its GUI and terminal and if you want to connect to another Linux server

Figure 1.1 Find the terminal application in Ubuntu



After the terminal is opened, type:
`ssh z01553986@10.157.217.25`

To summarize: how to log in



To logout, type
logout, exit or Ctrl+d

Use Virtual private network (VPN) to **connect to our Linux sever from home**

Use Internet Explorer or Firefox

Go to <https://secure.niu.edu>

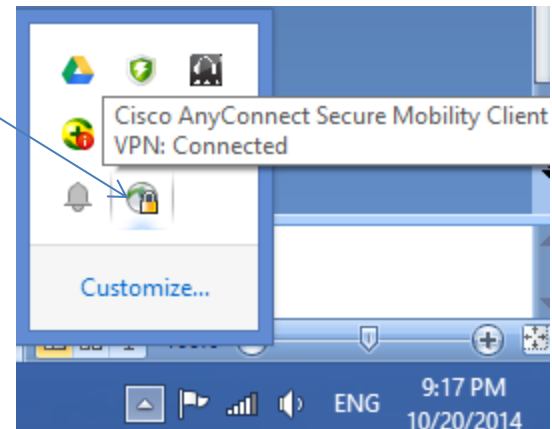
Put in your Z id and password

The next page will install CISCO AnyConnect Secure Mobility software

It will ask you to install JAVA if you don't have it yet

After that you have an icon at the right bottom corner, showing the VPN software

And you will be able to access the **10.157.217.25** Linux server (Ser)



Terminal window tricks

(may not work on all VNC or ssh clients...)

- ❑ Helpful tricks to avoid excessive command typing
 - Use **copy/paste**. Any text “mouse-selected” while holding the left mouse button is copied to clipboard. It may then be pasted, e.g., into a command, by clicking the **right** mouse button (PuTTY) or the **middle** button (when working through the console in 625 Rhodes).
 - Use **Up/Down arrow keys** – this will cycle through recently executed commands.
 - Use the **TAB key** – this will often present you with a list of choices after typing a part of a command – no need to remember everything.
 - **history** command: list all recently used commands – can copy a desired command and paste it to execute again, or refer to a command by its index

Examples:

history (*list all remembered commands*)

history | more (*list all remembered commands page by page*)

history | grep workdir (*list all remembered commands containing string “workdir”*)

Interactive command line terminal

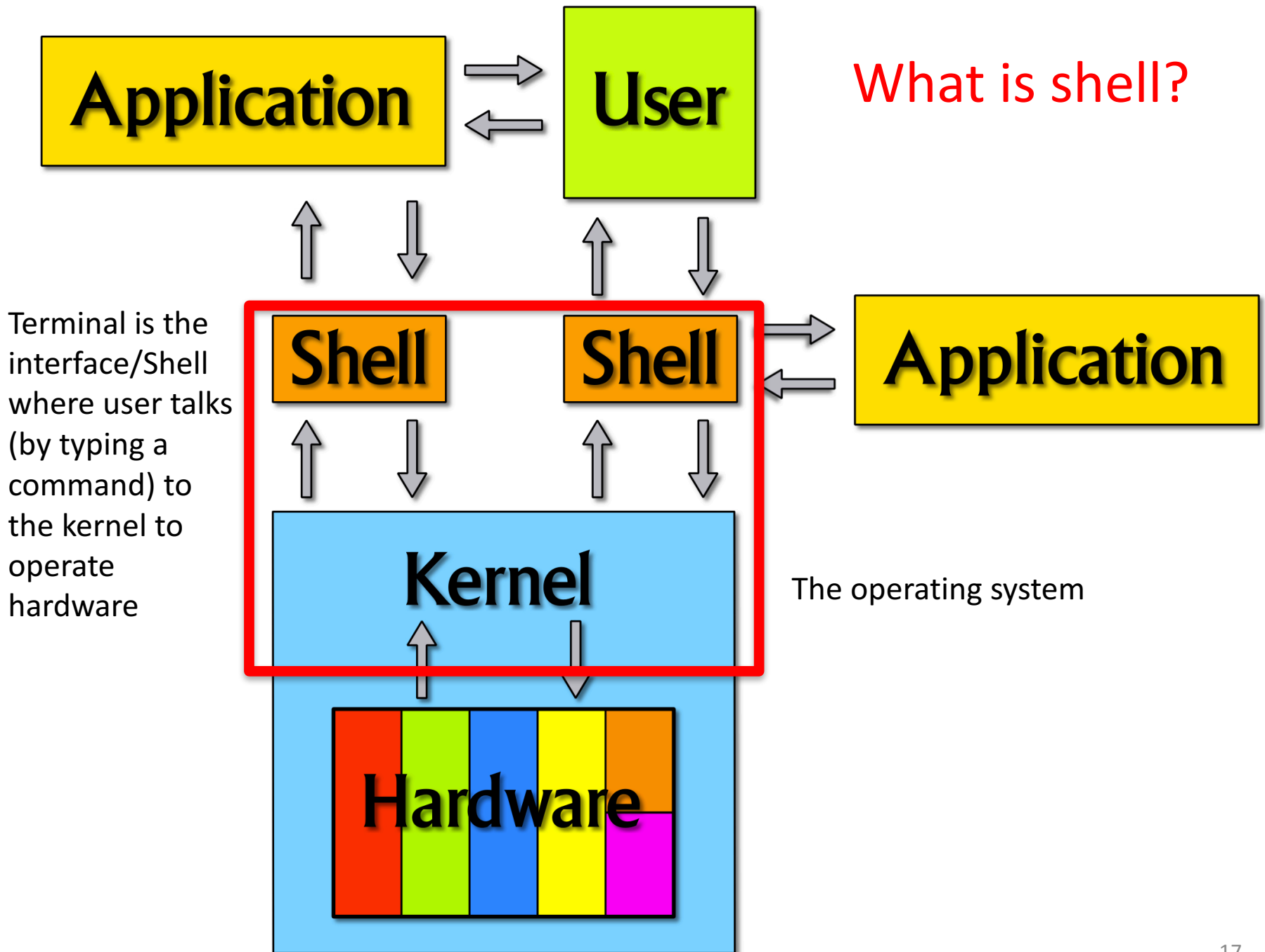
In the graphical interface of Windows, Linux and MAC, you click your **mouse** to make something happen.

However, under command line terminal (or console) interface, you always type in a command using the **keyboard** and hit Enter to let something happen.

The command you typed in is interpreted and executed by the **Shell**, the Linux software and interface to connect the users and the OS.

The output of the command is printed on the screen by default or written to a file. (together with error msg if any)

What is shell?



Commands

Commands can be a single word/letter or a few words separated by **space**; always hit Enter after you finished typing

The first word/letter is often a Shell command or an external program or your own script name

Other words can be command options, files, folders etc.

Examples:

```
yyin@ser:~$ pwd  
/home/yyin
```

```
yyin@ser:~$ ls /disk1/z01553986  
examples.desktop
```

For all of you, your home is at /disk1/yourid

e.g. /disk1/z01553986

```
yyin@ser:~$ ls /  
bin  cdrom  disk1  disk3  etc  initrd.img  lib  lost+found  mnt  proc  run  selin  
boot  dev  disk2  disk4  home  initrd.img.old  lib64  media  opt  root  sbin  srv
```

Commands you must learn

The absolute basics	File control	Viewing, creating, or editing files	Misc. useful commands	Power commands	Process-related commands
ls	mv	less	man	uniq	top
cd	cp	head	chmod	sort	ps
pwd	mkdir	tail	source	cut	kill
	rmdir	touch	wc	tr	
	rm	nano		grep	
	(pipe)			sed	
	> (write to file)				
	< (read from file)				

For people who have no experience in terminal: you are like in a dark room alone.
Where am I, what is that sound, what did I just bump into?
You can't see, you have to get used to using your other senses ...

No shortcuts, just learn by using

```
yyin@ser:~$ s
```

```
s: command not found
```

If you type some not-existing command

```
yyin@ser:~$ '
```

```
>
```

```
>
```

```
> ^C
```

If you mistakenly type in a quote and hit enter,
you get stuck

Ctrl+c will get you out of there

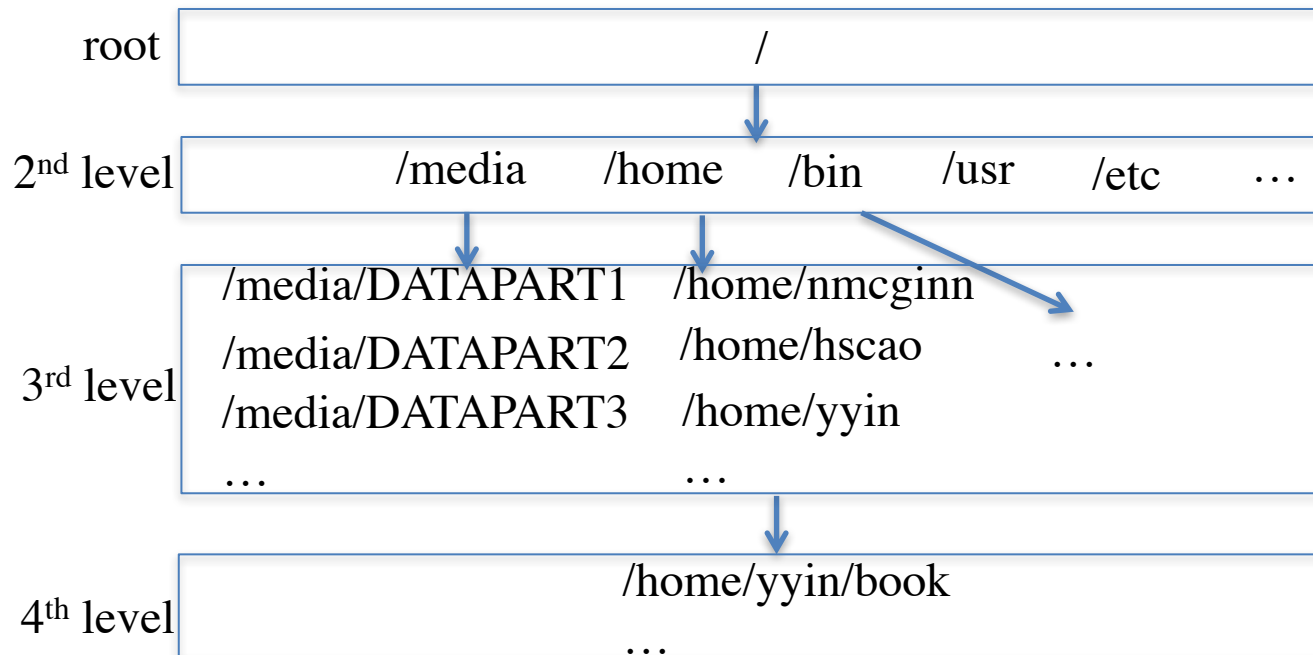
Ctrl+d will log you out

Everything is a file or a folder/directory in Linux

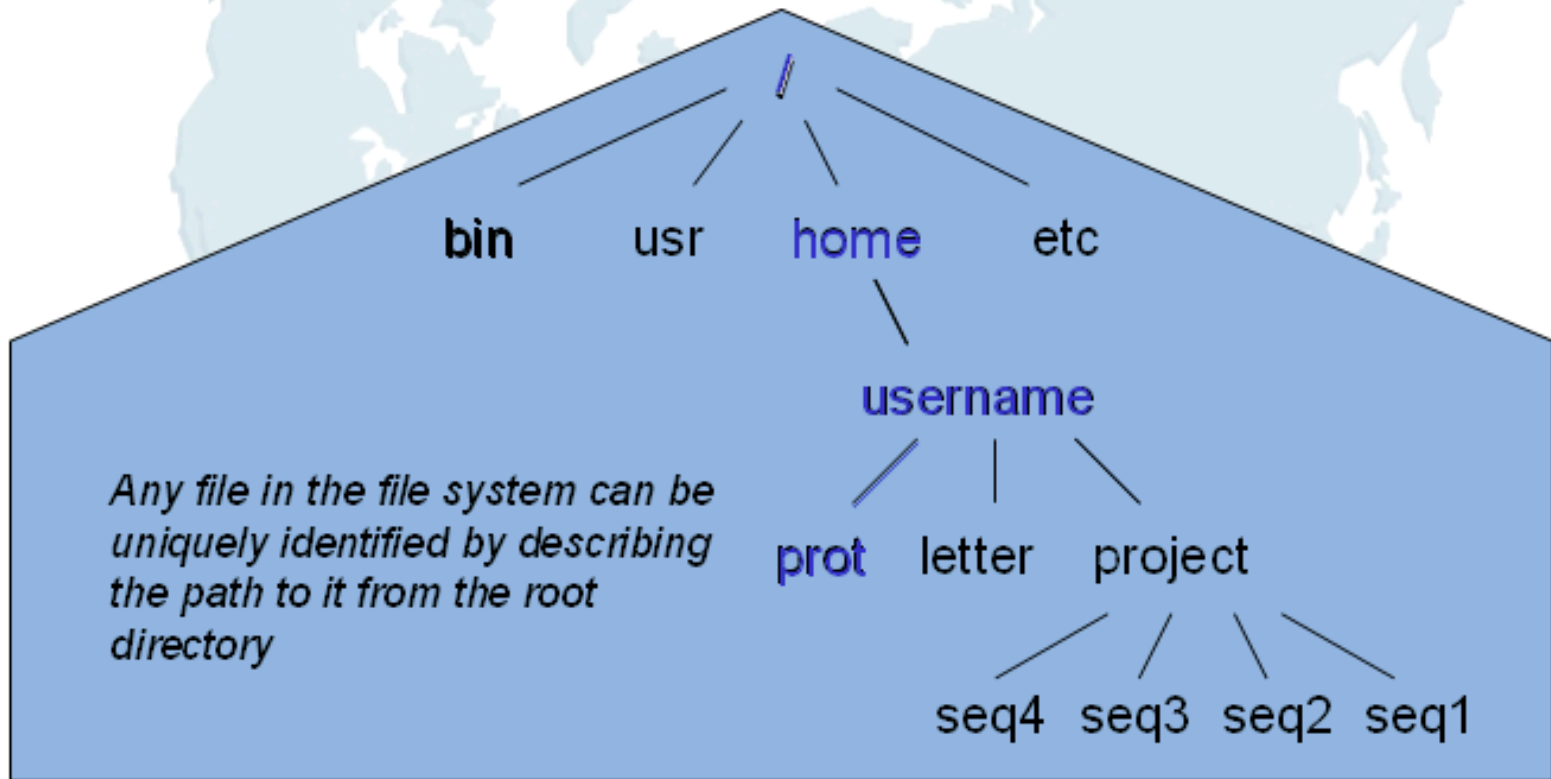
A folder contains files or/and other folders (subdirectories)

A subdirectory can contain other files or folders

So the file system is a tree-like structure (multi-furcation)



Organisation of the file system



`/home/username/prot`

Since you are in a file system, knowing where you are is very important

Remember **pwd** is the command to find where you are (**working directory**)

If you issue a command which expect a file/folder name, you need to tell Shell the location of the file/folder, which is called **path**

Without specifying the path, Shell will think the file/folder is in the **current folder**. If it can't find the file, it reports an error

```
yyin@ser:~$ pwd
/home/yyin
```

```
yyin@ser:~$ ls /disk4/z01553986
examples.desktop
```

Blue color - Directory

Green color - Executable or recognized data file

Sky Blue Color - Linked file

yellow with black background - device

Pink colour - graphic image file

Red - Archive file

```
yyin@ser:~$ ls /
bin  boot  cdrom  dev  disk1  disk2  disk3  disk4  e
tmp  usr  var  vmlinuz  vmlinuz.old
yyin@ser:~$ ls /disk1/
lost+found  z01422002  z01603076  z01625768  z01653641  z0172591
yyin@ser:~$ ls /disk2
lost+found  z01558913  z01624781  z01676006  z01690083  z0170388
yyin@ser:~$ ls /home/
elfitzek  yyin
yyin@ser:~$ █
```

File/folder/program names

Case sensitive!!!

`File` and `file` are two different files

Allow to contain letters, numbers, underscore (`_`), dot (`.`), dash (`-`), plus (`+`) but not to use other **special characters and spaces**

Example:

```
[yyin@gaea fungal]$ ls  
all.hmmlib_1.75.fungal-dockerin.domain.fa.nobac.n2+.fa.id.source.nrgi.ps.col.lab
```

For programs, better use the correct file extension (`.sh`, `.pl`, `.c`, `.py`), e.g. `run-blast.sh`

Home folder (dir)

After you login a remote Linux machine or you opened a terminal in your Ubuntu or MAC computer, you are at your home

You can create or download folders, files, bioinformatics softwares

Make it **organized and clean** by creating folders that have meaningful names, such as tools, data, work, project, scripts etc.

You can also create subfolders, e.g. blast under tools, or project1 under project

Don't put everything in your home directory; you will easily accumulate too many files/folders/programs/scripts that you will have a headache to find or remember

Directory (folder) commands

pwd find out where you are (your current directory or working directory)

cd change directory

Remember? Save you from some typing

Don' need to type everything in

Always use **TAB key** to auto-complete a word!!!!!!!!!!!!

Up arrow key to bring your previous commands

Relative path and absolute path

Find out where you are:

```
yyin@ser:~$ cd /home/yyin
```

```
yyin@ser:~$ pwd  
/home/yyin
```

Absolute path or full path




If I want to change directory to /home/

```
yyin@ser:~$ cd ..  
yyin@ser:/home$
```

If I want to go back to my home

```
yyin@ser:/home$ cd yyin  
yyin@ser:~$
```

Relative path, relative to your
current dir: /home



I can also use the absolute path to go back

```
yyin@ser:/home$ cd /home/yyin  
yyin@ser:~$
```

Special denotations:

tilde	~ or ~/	your home
	. or ./	your current directory
	.. or ../	the dir one level above
	../..	?

Try:

```
cd ~
```

```
cd ..  
pwd
```

```
cd ../..  
pwd
```

If you think you are lost in the file system and don't know where you are, always run

```
pwd
```

From anywhere to go back to you home

```
cd  
cd ~
```

More directory commands

`mkdir` create a directory

`rmdir` delete an empty directory (have no subdirectories or files)

At your home, try

`mkdir bioinfo`



Remember case sensitive and no special characters and space!!!

List command

ls list what files and directories are there in a folder

Try

These are **options**, used to enrich the functionalities of a command

<code>ls</code>	
<code>ls -l</code>	list in long listing format
<code>ls -la</code>	list all including hidden files/folders
<code>ls -lt</code>	list according to modification time
<code>ls -ltr</code>	list according to time in reverse order
<code>ls ..</code>	list one level up
<code>ls -l /home/yyin</code>	list things under a given folder
<code>ls -l less</code>	if there are too many files to display in one page, use pipe and less to show page by page (will explain shortly) have to type q to exit less

The manual command

```
man ls
```

If the manual is more than one page, hit **space** or **PgDn** key to page down, **PgUp** key or **u** to page up

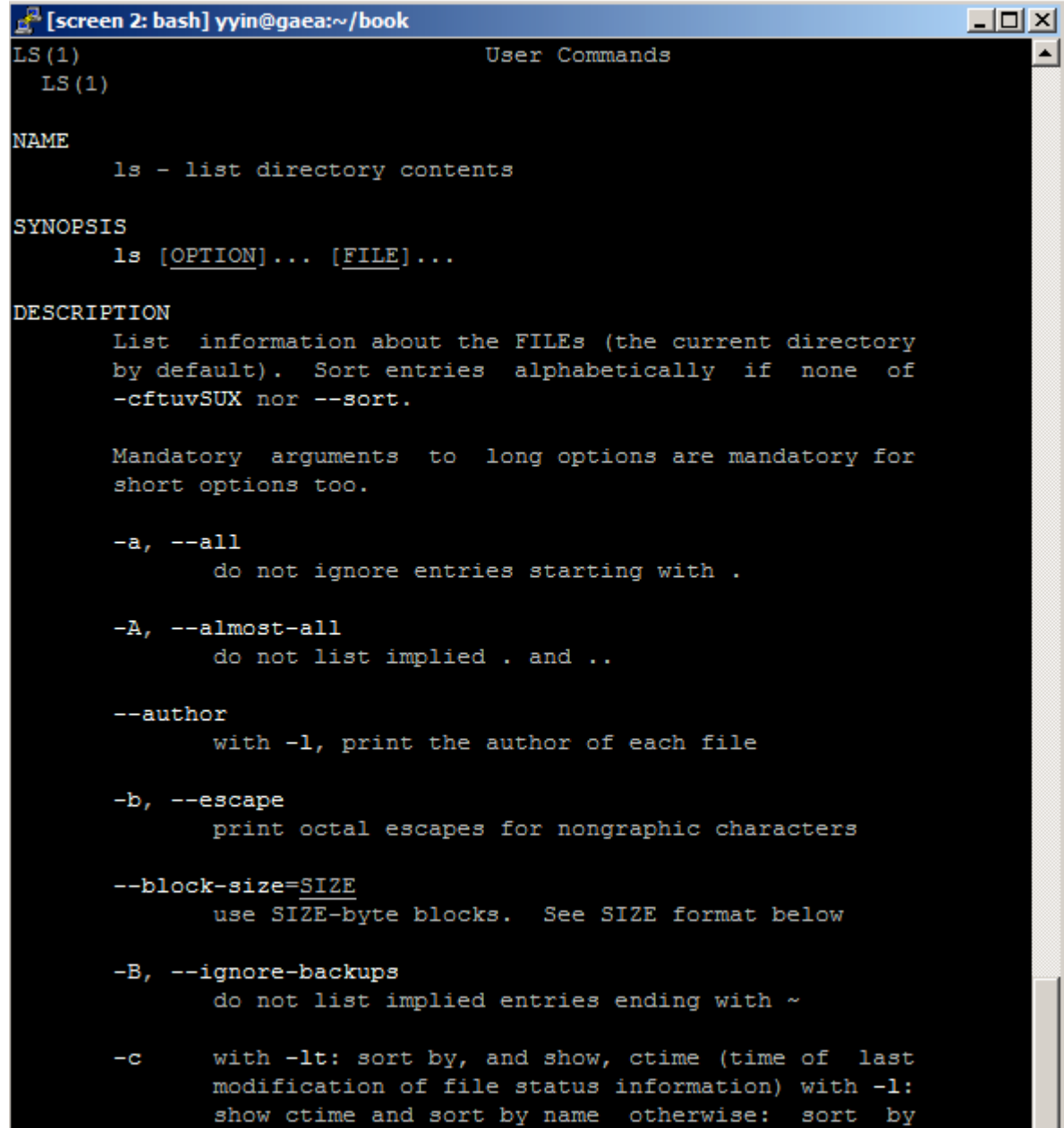
To exit the man page, hit **q**

Man followed by any Linux command to display the manual of that command

```
man pwd
```

```
man cd
```

...



```
[screen 2: bash] yyin@gaea:~/book
LS (1)                                User Commands
LS (1)
NAME
ls - list directory contents

SYNOPSIS
ls [OPTION]... [FILE]...

DESCRIPTION
List information about the FILES (the current directory
by default). Sort entries alphabetically if none of
-cftuvSUX nor --sort.

Mandatory arguments to long options are mandatory for
short options too.

-a, --all
do not ignore entries starting with .

-A, --almost-all
do not list implied . and ..

--author
with -l, print the author of each file

-b, --escape
print octal escapes for nongraphic characters

--block-size=SIZE
use SIZE-byte blocks. See SIZE format below

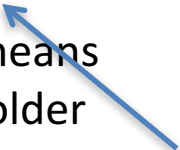
-B, --ignore-backups
do not list implied entries ending with ~

-c
with -lt: sort by, and show, ctime (time of last
modification of file status information) with -l:
show ctime and sort by name otherwise: sort by
```

ls -lt /

```
yyin@ser:~$ ls -lt /
total 116
drwxrwxrwt  8 root root  4096 Oct 18 10:17 tmp
drwxr-xr-x 133 root root 12288 Oct 17 05:36 etc
drwxr-xr-x  22 root root   820 Oct 16 23:54 run
drwxr-xr-x  15 root root  4400 Oct 16 00:20 dev
drwxr-xr-x  13 root root    0 Oct 16 00:19 sys
dr-xr-xr-x 191 root root    0 Oct 16 00:19 proc
drwxr-xr-x  2 root root  4096 Oct 13 00:50 media
drwxr-xr-x  8 root root  4096 Oct 12 11:08 disk4
drwxr-xr-x  8 root root  4096 Oct 12 11:07 disk3
drwxr-xr-x  8 root root  4096 Oct 12 11:05 disk2
drwxr-xr-x  8 root root  4096 Oct 12 11:03 disk1
drwxr-xr-x  4 root root  4096 Sep  9 00:48 home
drwxr-xr-x  14 root root  4096 Aug 17 21:49 var
drwx----- 8 root root  4096 Aug 13 00:59 root
lrwxrwxrwx  1 root root   32 Jul 31 22:37 initrd.img.old -> boot/initrd
lrwxrwxrwx  1 root root   29 Jul 31 22:37 vmlinuz.old -> boot/vmlinuz-3
drwxr-xr-x  2 root root  4096 Jul 31 22:36 cdrom
drwx----- 2 root root 16384 Jul 31 22:32 lost+found
drwxr-xr-x  3 root root  4096 Jul 31 08:11 opt
drwxr-xr-x  3 root root  4096 Jul 31 08:09 boot
lrwxrwxrwx  1 root root   29 Jul 31 08:09 vmlinuz -> boot/vmlinuz-3.2.0
lrwxrwxrwx  1 root root   33 Jul 31 08:09 initrd.img -> /boot/initrd.im
drwxr-xr-x  2 root root 12288 Jul 31 07:19 sbin
drwxr-xr-x 21 root root  4096 Jul 31 07:17 lib
drwxr-xr-x  2 root root  4096 Jul 31 07:17 bin
drwxr-xr-x  2 root root  4096 Jul 31 07:15 lib64
drwxr-xr-x  2 root root  4096 Aug 23 2012 srv
drwxr-xr-x 10 root root  4096 Aug 23 2012 usr
drwxr-xr-x  2 root root  4096 Apr 19 2012 mnt
drwxr-xr-x  2 root root  4096 Mar  5 2012 selinux
```

d
means
folder



Permission User and Group size Modification time

File and folder permission can be changed

Let's look at one of the files under my home:

```
yyin@ser:~$ ls -l /home/yyin/Unix_and_Perl_course/Data/GenBank/
total 22048
-rw-r--r-- 1 yyin yanbin 10602328 Oct  9  2008 E.coli.genbank
-rw-r--r-- 1 yyin yanbin 11969735 Jul 19  2009 Y.pestis.genbank
```

-rw-r--r--

The first one “-” means it is a file; it will be “d” if it is a folder (directory).

The following **nine columns** indicate the permission of read (r), write (w) and execute (x) granted for the **user** (first three columns), the **group** (middle three) and **others** (last three). “-” means no permission

-rwxrwxrwx

the file could be read, write, and execute by anybody

-r-----

can only be read by the owner

```
yyin@ser:~$ chmod go-r /home/yyin/Unix_and_Perl_course/Data/GenBank/E.coli.genbank
yyin@ser:~$ ls -l /home/yyin/Unix_and_Perl_course/Data/GenBank/E.coli.genbank
-rw-r--r-- 1 yyin yanbin 10602328 Oct  9  2008 /home/yyin/Unix_and_Perl_course/Data/GenBank/E.coli.genbank
```

Now you can not read that file in my home

How to view files

In order for you to read my file:

```
yyin@ser:~$ chmod go+r /home/yyin/Unix_and_Perl_course/Data/GenBank/E.coli.genbank
```

```
yyin@ser:~$ ls -l /home/yyin/Unix_and_Perl_course/Data/GenBank/E.coli.genbank  
-rw-r--r-- 1 yyin yanbin 10602328 Oct  9 2008 /home/yyin/Unix_and_Perl_course/Data/GenBank/E.coli.genbank
```

At your home, try

```
head /home/yyin/Unix_and_Perl_course/Data/GenBank/E.coli.genbank
```

Similar commands:

```
more  
less  
tail  
cat
```

Unlike in Windows,
text files of any size
can be viewed

Only text files can be viewed

There are also binary files, zipped files and tarred files that can not be viewed. For example,

- Executables (e.g., blast, samtools, bwa, bowtie)
- Data in binary format (e.g, BAM files, index files for BWA or Bowtie, formatted BLAST databases)
- Compressed files (usually *.gz, *.zip, *.bz2,..., but extensions not necessary)

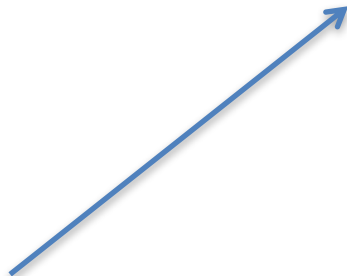
How to copy and move files/folders

At you home, try

```
cp /home/yyin/Unix_and_Perl_course/Data/GenBank/E.coli.genbank .
```

Remember what dot (.) means?

You created this folder in your home moments ago



Now try,

```
ls
mv E.coli.genbank bioinfo
ls
mv bioinfo/E.coli.genbank .
ls
```

What about?

```
mv E.coli.genbank bioinf
```

What if you want to copy a folder?

```
cp -r bioinfo bioinfo2
```

Difference?

```
mv file folder
mv file file
```

How to delete files/folders?

<code>rm file</code>	remove file
<code>rm folder</code>	remove empty folder
<code>rm -rf folder</code>	remove not empty folder

For example,

```
rm -rf bioinfo2
```

WARNING: deleted things can not be recovered in Linux!!!

To be asked before deletion,

```
yyin@ser:~$ rm -i bioinf  
rm: remove regular file `bioinf'? y
```

Next class: commands for
controlling files