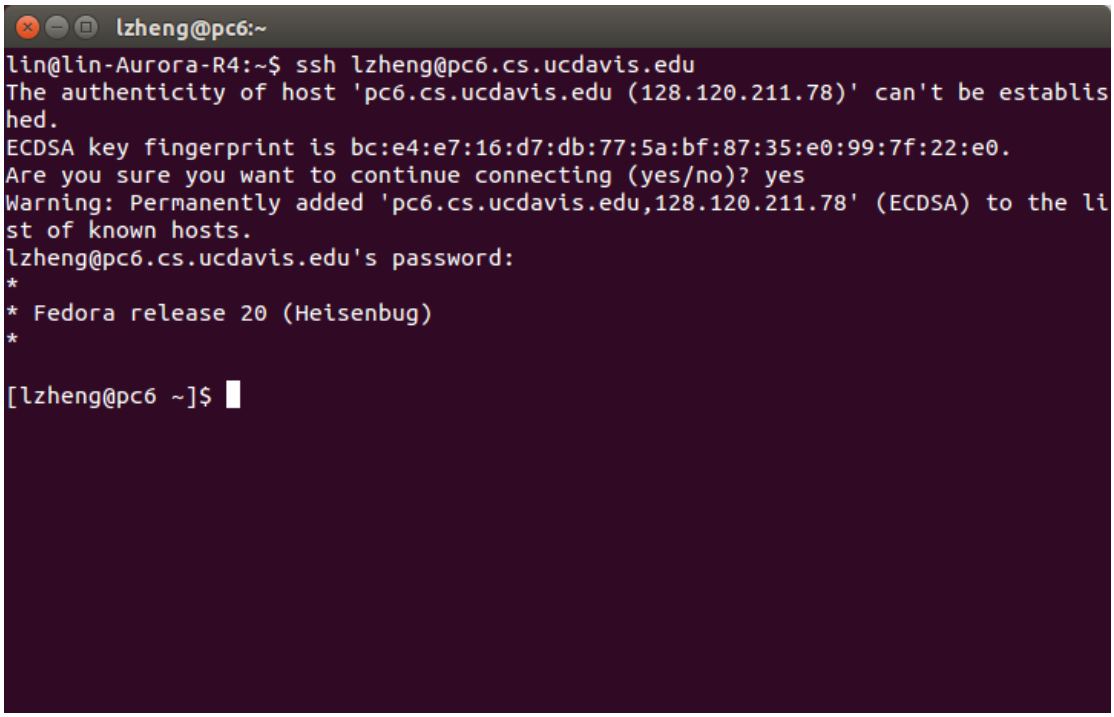
Remotely login to CSIF computers:

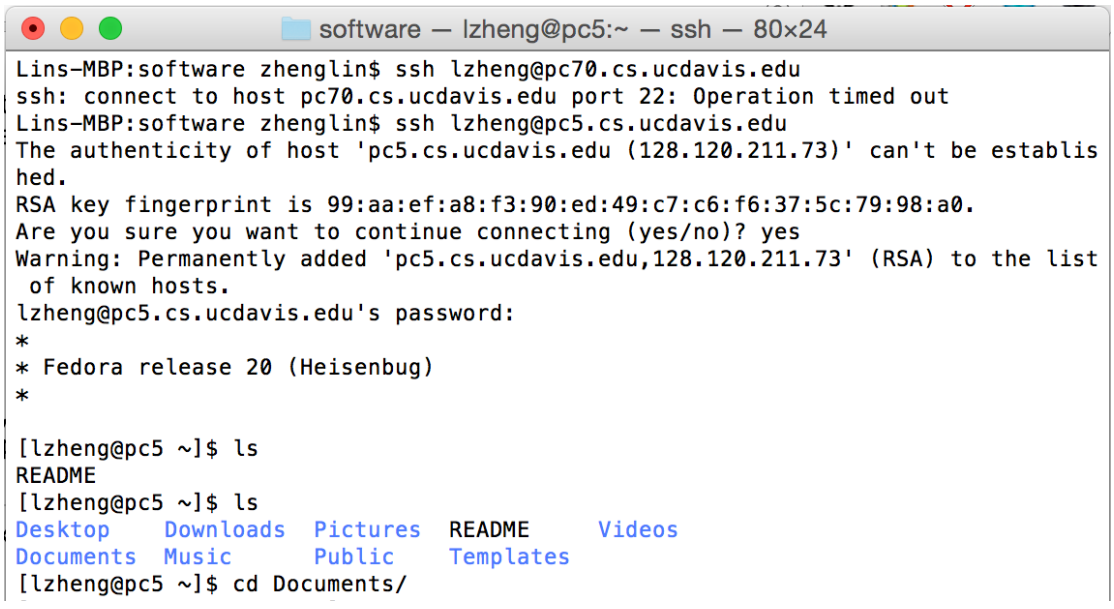
1. Remotely login to the CSIF computers with ssh:
Use your kerberos username and password

Open the terminal, input:
ssh username@pc35.cs.ucdavis.edu
Here “username” should be replaced by our own UC Davis account's name, and “pc33” can be replaced by any number between “pc33” and “pc60”.
The first time you login a host, the system will ask if you want to continue connecting, answer “yes”, and then “return”.
After that we input our kerberos password and login (when you input the password, it will not show up on the terminal).

Linux, OS X Typical Scenario:
Linux screenshot (it is a previous version, so it is pc6, but in your case, replace it by number between “pc33” … “pc60”),



OS X screenshot(it is a previous version, so it is pc5, but in your case, replace it by number between “pc33” - “pc60”):



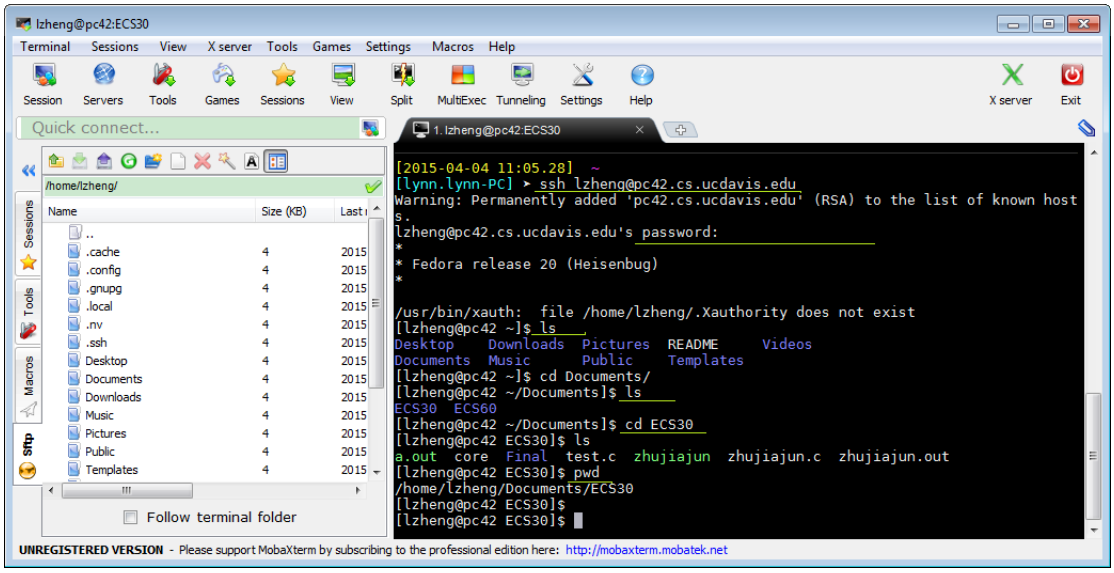
Upload files to the server:

cd /path

Enter where our files are located.

scp myfilename username@pc20.cs.ucdavis.edu:/home/username/file/

Windows Typical Scenario:
Download “MobaXterm” from
http://mobaxterm.mobatek.net/
With MobaXterm, you can drag the files you want to upload to the left area.



You can stop here.
For the Unix and Vi commands, check the files “UnixCommands.pdf”, “vi_editor_commands.pdf”, “VICommands.pdf” on Smartsite: Resources/Lab Resources/Unix/Vi

2. Unix commands
For test, you are going to input the following commands (only the words in red) to the terminal and see what happen.
pwd: print working directory
cd: change directory
$cd: no argument, return to your home directory.
$cd can fail if you do not have permission to get in that directory.
ls: list
$ls -al
drwxr-xr-x 3 barca37 users 4096 Oct 3 00:39 .local
1.types and permissions;
2.links associated with this file.
3.who owns the file.
4.groups
5.size
6.last edit
7.name
*note: ”.” and ”..” from ”ls -al” means current directory, parent directory.

chmod: change mode
-/rw-/r-x/r-: type/users/group/other Examples:
$chmod u+x 'filename'
$chmod 754 'filename'
mkdir/rmdir: make directory/remove directory
It can make a directory and it sub-directories in a single command.
$mkdir progs progs/p1 progs/p2
rmdir can remove empty only. Not empty use following:
$rm -r directory

cp: copy
$cp -r dir1 dir2
If dir2 does not exists, create dir2, same contents as dir1.
If dir2 exists, put a copy of dir1 into dir2
$cp file1 file2
If file2 does not exists, create file2, same contents as file1.
If file2 exists, over write file2 with file1's content.

mv:rename/move
$mv dir1 dir3:
If dir3 does not exist, rename
If dir3 exists, move it to dir3
=====
rm: remove
$rm dir1/* remove all files under directory "dir1", exclude hidden files. $rm dir1/*.c remove all files end with .c

There are more unix commands.

3. Some simple vi commands:
Open the terminal, go to the directory you want to create a file.
Input "vi file.c", and click key "return".

Basic commands
Insert/Append: i (for a typical scenario, you click key "i" to get into the input mode, input some words, and click key "esc" to escape the input mode, click key ":" and click key "w" then click key "return", you save the words you just input.)
go to 3rd line: ":3"
undo: "u"
save: ":w"
save and quit: ":wq"

Motion commands
left: h
right: l
up: k
down: j
start of line: 0(zero) end of line: $
start of file: gg
end of file: G