COMP 3500 Introduction to Operating Systems
Project 1 (Option 1): How to Remotely Connect to Linux Machines

Important! If you plan to carry out this project by accessing a remote Linux server rather than your local Linux machine, you may choose this option (i.e., option 1). To improve your Linux programming skills, you are strongly encouraged to go through option 2 after the completion of option 1.

1. Goals

- To setup putty to access in a remote Linux machine through a terminal.
- To learn how to use winscp to transfer files from between your local PC and a remote server.
- To learn how to compile and run your first c program.
- To learn a few basic and important Linux commands.

2. Getting Started with PuTTY

Note: Read Section 3 if you are a Mac user.

2.1 About PUTTY. PuTTY is a client program for the SSH network protocol, allowing you to access a remote Linux machine over a network. You run PuTTY on a local Windows machine, and tell it to connect to a remote Linux machine. PuTTY opens a window, in which you type commands to use the Linux machine that sends information back to your window. In other words, you can work on the Linux machine as if you were sitting at its console, while actually sitting somewhere else.

2.2 Download PuTTY. PuTTY can be downloaded at:
http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html

Information regarding PuTTY can also be found at:
http://www.chiark.greenend.org.uk/~sgtatham/putty/faq.html

2.3 Connect to a remote Linux machine using PuTTY.
You can start a remote session with any of the computers in Shelby 2119 and 2122. You can follow the 7 steps below to start a session on a Linux machine.

- Step 1: Launch PuTTY,
- Step 2: Fill out the the hostname (see Fig. below)
- Step 3: Click "Open". The first time you connect to a new host, PuTTY will alert you that the host key was not found. Press yes to continue.
- Step 4: A prompt will open. For "login as:", type the username you use to login at a remote Linux machine. Do the same for the password.
- Step 5: Press “Enter” to randomly choose an available Linux machine to login.
- Step 6: Type your password again.

- Step 7: Now you have a terminal session at the host you chose to log in to.

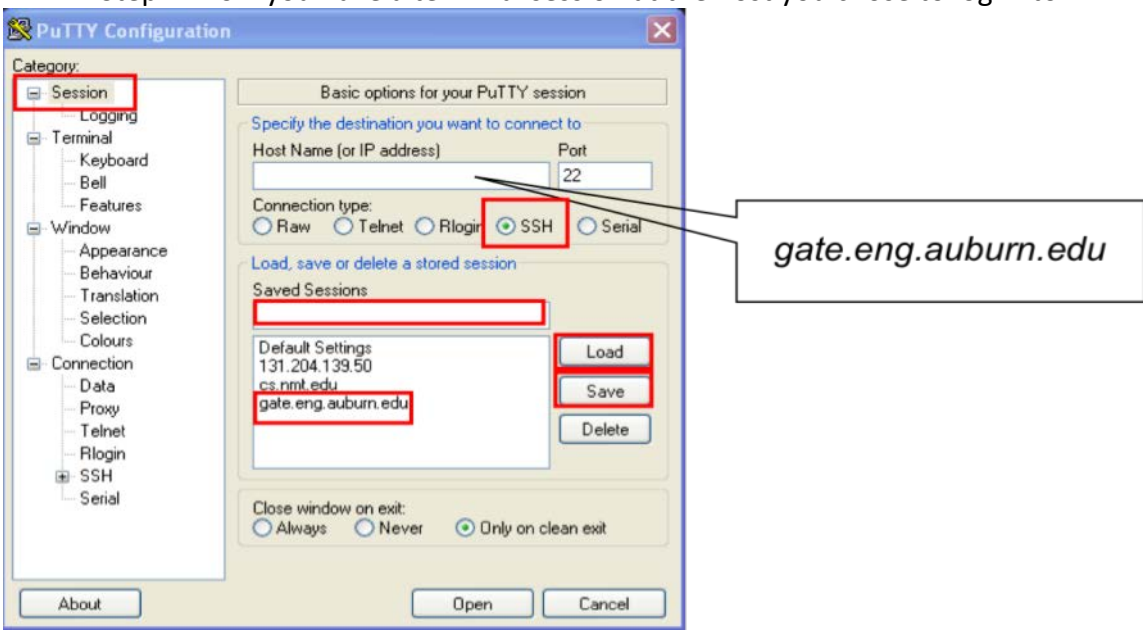

Fig. 1. Step 2 Fill out the the hostname

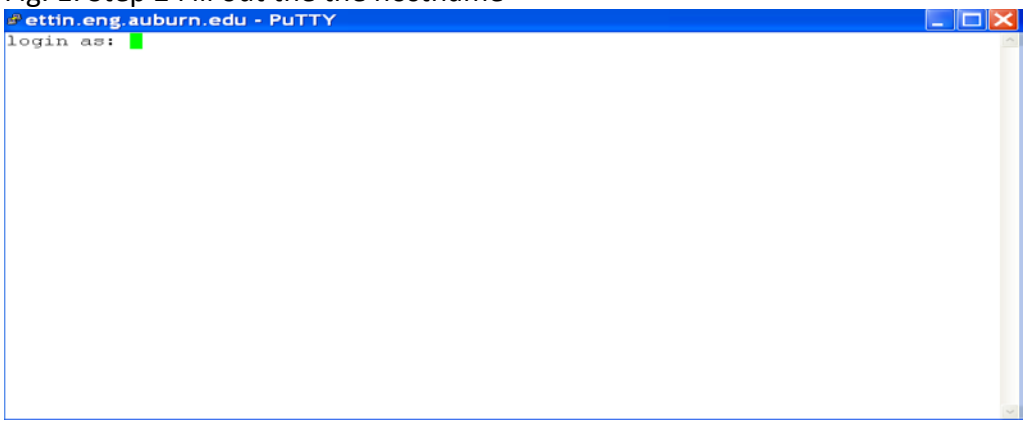

Fig. 2 Step 4 A prompt will open. For "login as:", type the username you use to login at a remote Linux machine. Do the same for the password.

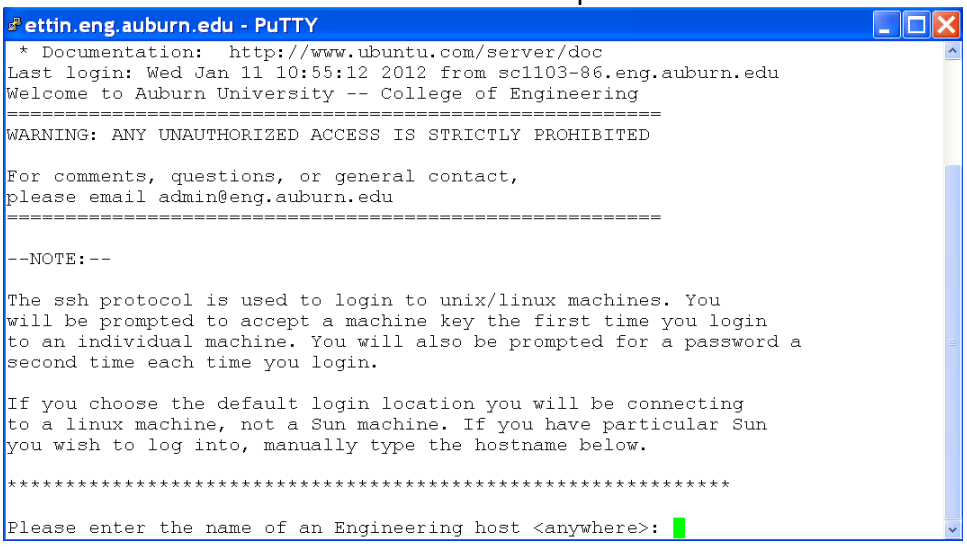

Fig. 3 Step 5: Press “Enter” to randomly choose an available Linux machine to login.

3. The “Terminal” Tool in Mac:

In Mac, you can use a built-in tool called Terminal, which is an application similar to PuTTY.

To start the Terminal tool, go to your Mac’s Applications folder, click on the Utilities folder, then click on Terminal.

To access a Linux server using the Terminal, you can type

ssh yourusername@gate.eng.auburn.edu

- when prompted for the password, enter your auburn account password.
- type "exit" to quit the session.

4. Some Linux Commands:

`exit`

This is how you leave the system.

`ls`

This lists all the files in the current directory. It is like the DOS dir command.

`mkdir`

This creates a new directory. Say mkdir dirname to create a new directory named dirname. Directories are the same thing as folders in Windows terminology. They allow you to collect related files in one place.

`cd`

Commands apply to files in the "current directory". The command cd dir changes the current directory to be dir.

`rm`

This deletes files; say rm fn to delete the file named fn. This is like DOS delete. The name stands for "remove".

5. Compile and Run your First C Program:

Please take the following steps to compile and run your first program:

- Step 1: In the Linux session in the PuTTY window, create a folder (e.g., comp3500) in your home holder. Please use “`mkdir comp3500`” to create the folder.
- Step 2 Change your directory using the following command:

```
$cd comp3500
```

- Step 3: create your first C program (e.g., helloworld.c). You can find sample source code here: https://www.programiz.com/c-programming/examples/print-sentence
- and save this source code file in "H:\comp3500" of the your Windows machine. Note that "H:" drive in your windows machine is the same as the home folder in the remote Linux machine.
- Step 4: Compile the source code using the command below:

```
$gcc helloworld.c –o helloworld
```

  Alternatively, you can compile the source code using the following command. By default, an executable file call "helloworld" will be generated.

```
$make helloworld.c
```

- Step 5: run the program:

```
$./helloworld
```

6. Using the VI editor:

Note: A detailed tutorial can be found at
http://www.eng.hawaii.edu/Tutor/vi.html

6.1 Getting Out of VI

VI has two modes and in order to get out of VI, you have to be in command mode. Hit the key labeled "Escape" or "Esc" (If your terminal does not have such a key, then try ^[, or control-[.) to get into command mode. If you were already in the command mode when you hit "Escape", don't worry. It might beep, but you will still be in the command mode.

The command to quit out of VI is :q. Once in command mode, type colon, and 'q', followed by return. If your file has been modified in any way, the editor will warn you of this, and not let you quit. To ignore this message, the command to quit out of VI without saving is :q!. This lets you exit VI without saving any of the changes.

Of course, normally in an editor, you would want to save the changes you have made. The command to save the contents of the editor is :w. You can combine the above command with the quit command, or :wq. You can specify a different file name to save to by specifying the name after the :w. For example, if you wanted to save the file you were working as another filename called filename2, you would type: w filename2 and return.

Another way to save your changes and exit out of VI is the ZZ command. When in command mode, type ZZ and it will do the equivalent of :wq. If any changes were made

to the file, it will be saved. This is the easiest way to leave the editor, with only two keystrokes.

6.2 The Two Modes of VI

The first thing most users learn about the VI editor is that it has two modes: command and insert. The command mode allows the entry of commands to manipulate text. These commands are usually one or two characters long, and can be entered with few keystrokes. The insert mode puts anything typed on the keyboard into the current file.

VI starts out in command mode. There are several commands that put the VI editor into insert mode. The most commonly used commands to get into insert mode are a and i. These two commands are described below. Once you are in insert mode, you get out of it by hitting the escape key. If your terminal does not have an escape key, ^[should work (control-[). You can hit escape two times in a row and VI would definitely be in command mode. Hitting escape while you are already in command mode doesn't take the editor out of command mode. It may beep to tell you that you are already in that mode.

6.3 Simple VI Commands

Here is a simple set of commands to get a beginning VI user started. There are many other convenient commands, which will be discussed in later sections.

`a`

enter insert mode, the characters typed in will be inserted after the current cursor position. If you specify a count, all the text that had been inserted will be repeated that many times.

`h`

move the cursor to the left one character position.

`i`

enter insert mode, the characters typed in will be inserted before the current cursor position. If you specify a count, all the text that had been inserted will be repeated that many times.

`j`

move the cursor down one line.

`k`

move the cursor up one line.

`l`

move the cursor to the right one character position.

r

replace one character under the cursor. Specify count to replace a number of characters

u

undo the last change to the file. Typing u again will re-do the change.

x

delete character under the cursor. Count specifies how many characters to delete. The characters will be deleted after the cursor.

6.4 Modify your first C program using VI

Can you start editing your `helloworld.c` source code by the following command in Linux. We assume that you working directory is `$~/comp3500`

```
$vi helloworld.c
```

7. Transfer Files between your Windows PC and a remote Linux Machine

Note: (1) Read Section 8 if you are a Mac user. (2) You do not need to use WinSCP is you are using a machine in Shelby 2119 and 2122.

7.1 About WinSCP. WinSCP is a free client FTP client for Windows machines. You can use WinSCP to transfer files between your local windows machine and a remote Linux machine on campus.

7.2 Download WinSCP. WinSCP can be downloaded at:

http://winscp.net/eng/download.php

More information on WinSCP can also be found at:

http://winscp.net/eng/docs/faq

7.3 Connect to a remote Linux machine.

- Step 1: download and install winSCP.
- Step 2: set up winSCP: (see also Fig. 4)
  - click on 'Session' on the left-hand side, to bring up the Session options.
  - Fill in the 'Host name' as "scp..eng.auburn.edu"
  - 'Port number' must be 22
  - Fill in 'User name' with your Auburn username
  - In the 'Password' field, enter your password
  - Select 'SCP' in the 'Protocol' section
  - click 'Save' so you won't have to enter all that again in the future.
- Step 3: click your saved session to highlight it and click 'Login'. Note: The first time you login, you may see this message - "The server's host key is not cached in the registry, just click 'Yes'

- Step 4: drag your files into the directory you want. (see Fig. 5)
- Step 5: When you are done, just close the window, and click 'Ok'

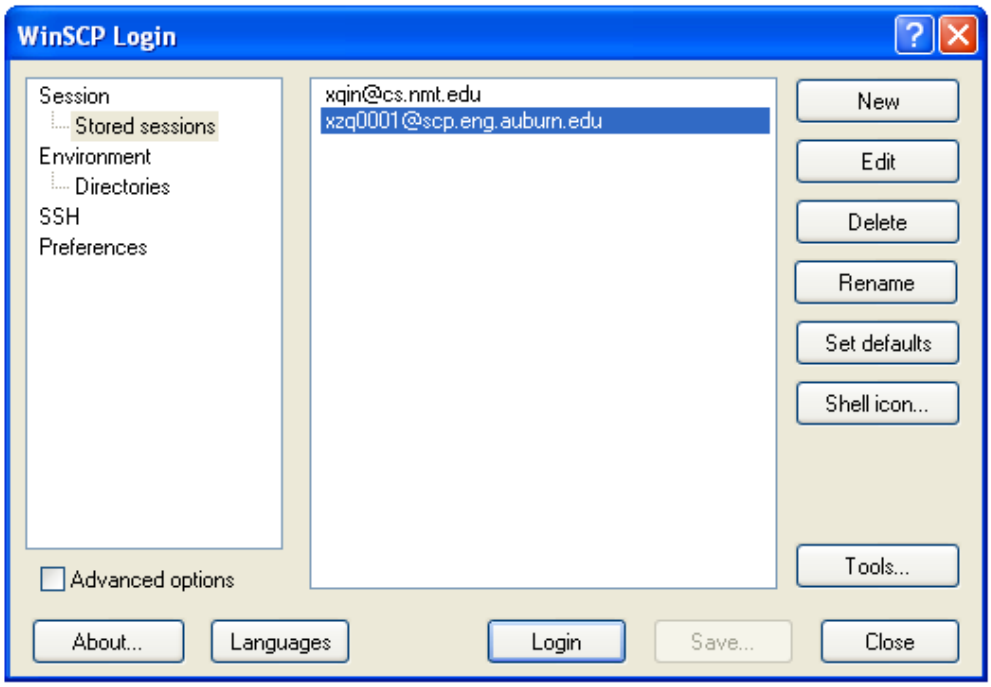

Fig. 4. Step 2: set up winSCP:

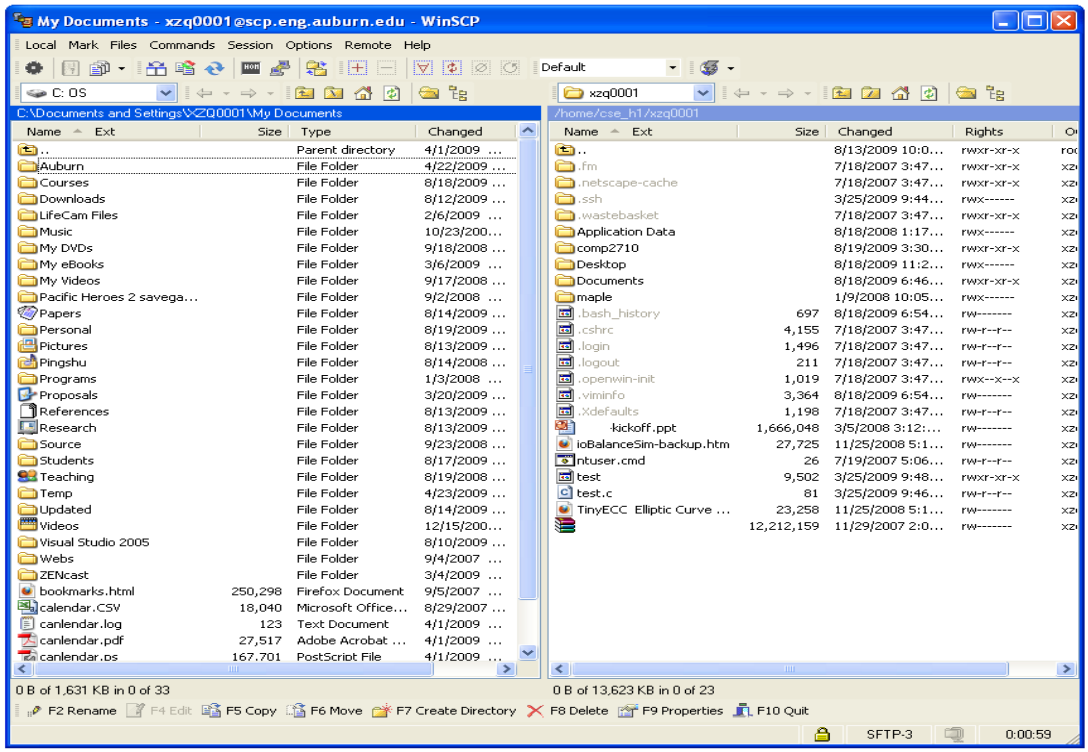

Fig. 5. Step 4: drag your files into the directory you want.

8. Transfer Files between your Mac and a remote Linux Machine

You are recommended to use FileZilla to transfer files between your Mac and a remote Linux machine. Please down FileZilla at:

http://filezilla-project.org/download.php