Networked Systems 3: Warm-up Exercise

Dr Colin Perkins

12 January 2011

The laboratory sessions for Networked Systems 3 aim to introduce you to network programming in C on Linux systems. There are weekly labs for this course, in which you will complete several exercises that build on your knowledge of C programming and pthreads from the AP3 course last semester, and the introduction to the Berkeley Sockets API given at the start of this course. These exercises are assessed, and count for 20% of the marks for this course in total.

The purpose of this warm-up exercise is to familiarise you with network programming in C using the Berekely Sockets API. You should write two programs:

`hello_server` Your server should listen on TCP port 5000 for incoming connections. It should accept the first connection made, read all the data it can from that connection, print that data to the screen, close the connection, and exit.

`hello_client` Your client should connect to TCP port 5000, send the string "Hello, world!", then close the connection. The client should take the name of the machine on which the server is running as its single command line argument (i.e., if the server is running on machine `bo720-1-01` you should run your client using the command `hello_client bo720-1-01`.

Run your client and server, and demonstrate that you can send the text "Hello, world!" from one to the other. Try this with client and server running on the same machine, and with them running on two different machines. Once this is working, modify your client to send a much longer message, and check that works too.

You are *required* to write a simple Makefile to compile your code, rather than running the compiler by hand. You are also *strongly advised* to enable all compiler warnings (at minimum, use `gcc -W -Wall`), and to fix your code so it compiles without warnings. Compiler warnings highlight code which is legal, but almost certainly doesn't do what you think it does. Use them to help you find problems.

This exercise is worth 5% of the marks for this course. You will be assessed on the quality, correctness, and readability of your client and server code, and of your Makefile.

You are required submit an electronic copy of your source code and Makefile (do *not* submit your compiled binaries), archived as a `.tar.gz` file which expands into a directory named after your 7-digit matriculation number followed by "-lab0". For example, if your matriculation number is 0301234, your archive should expand to create a directory "0301234-lab0" with your files inside. *If you do not know how to create such an archive, ask one of the lab demonstrators*. These archives should be submitted via Moodle not later than 9:00am on Wednesday 26 January 2011. This is a hard deadline; late submissions will be awarded a mark of zero, unless a valid special circumstances form is also submitted.