NS3 Lab 1 – TCP Client/Server Programming in C

Dr Colin Perkins
School of Computing Science
University of Glasgow
http://csperkins.org/teaching/ns3/

9 January 2013

Introduction

The laboratory sessions for Networked Systems 3 (NS3) will introduce you to network programming in C on Unix/Linux systems. There are weekly labs for this course, during which you will complete several exercises. These exercises will build on your knowledge of C programming and pthreads from the Advanced Programming 3 course last semester, and on the material in the NS3 lectures. There are a mixture of formative and summative exercises. The formative exercises are intended to give you practice in programming networked systems in C; they are not assessed. The two summative exercises are assessed, and are worth a total of 20% of the marks for this course.

This is NS3 lab 1, an introduction to TCP client/server programming in C. It comprises one formative exercise, and should be completed during the timetabled laboratory session in week 1 of the semester. This work is not assessed, but is important preparation for the summative exercises later in the course.

Formative Exercise 1: Networked Hello World

The first formative exercise demonstrates how to build the most simple TCP-based client-server application. This exercise should be completed using the notes from the "Network Programming in C: The Berkeley Sockets API" lecture as reference. You should write two programs:

`hello_server` The 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 of a host named on the command line, send the text "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 (more than 1500 characters), 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 -Werror`), 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.