

Computer Networks Theory + Lab

CS252 Labs ▾ > Lab03: Introduction to Socket Programming ▾ ▾

Summary

Contents

1. [Lab03: Instructions \(Document\)](#)
2. [Socket Programming \(Video\)](#)

[« Previous \(Lab02: Internet Protocol Stack and Metrics\)](#)[Next \(Lab04: Ethernet and WiFi Exploration\) »](#)

Lab03: Instructions

Lab03: Socket Programming Intro

Objective:

1. Learn socket programming as a precursor to an upcoming project

Reference Material:

1. <https://beej.us/guide/bgnet/html/> (Latest guide, 2020)
2. Socket programming videos on Bodhitree (note these videos are based on a much earlier version of Beej's guide 2001; but the important content is the same. You can download this version via "view document" link)

Requirements and Instructions:

1. Gnu gcc compiler.
2. The code used in the Beej's guide can be compiled on a Linux/MAC machine using Gnu's gcc compiler. Download and install the same if not available.
 - i. Linux: sudo apt update; sudo apt install build-essential; check if installed via "gcc --version"
 - ii. MAC: Install command line tools from - <https://developer.apple.com/download/all/> First login with apple ID and search for "command line tools". Download the dmg (13.2 is stable) and follow along the instructions to install OR if you have homebrew package manager, you can directly install with "brew install gcc"
 - iii. Windows: Assuming WSL is installed. Open a terminal window. Type "ubuntu", you should get a shell. In the shell, you can install gcc by following instructions related to Linux above. If you want to open multiple shells, just type ubuntu again in another window's terminal.
3. To get code from Beej's guide to compile, you could just use "wget URL" within the shell, where URL is the code link from Beej's website.

Exercise 1:

Socket programming is very important to know if you want to write applications that talk with each other. While one can use Java and other frameworks that provide a better interface to managing sockets, we will however use C/C++ which is bare-bones and really reveals what is happening internally. You will get a much better idea this way.

Normally, unless you finish the entire course, you will not really appreciate some aspects of socket programming. So some aspects may not make full sense yet. But nonetheless, we cannot leave it till the end of the course, since then you won't get enough time to complete the project. So, we will introduce it now, and also give you the project details by the end of Jan. However, you will have till almost the end of sem to submit the project, plenty of time.

1. Watch the socket programming video on BodhiTree.
2. Go through the "Latest Beej's guide" (or if you want to keep it simple, go through the 2001 version.)
 - i. Latest version: You can ignore anything to do with IPv6 Cover till section 6. In section 7, cover 7.1 and 7.3, rest is optional.
 - ii. 2001 version: Do till 6.2, rest is optional
3. Attempt the quiz titled "Lab03: Socket Programming" on SAFE. This quiz tests some simple and basic aspects of socket programming. The quiz is available only for an hour. So, open it only after you finish part 1 and 2.

[View Document](#)

Discussion Forum



Threads : 0

[Recent](#) [Earlier](#) [Popularity](#) [Importance](#) [Instructor](#)[Create thread](#)