CMPSC 381
Data Communications and Networks
Fall 2012
Bob Roos

Lab 1
30 August 2012
Due Thursday, 6 September, 1:30 pm,
in your Sakai dropbox

Preliminaries: Do you know how to create directories in the terminal window and navigate between directories? If not, please ask me at the beginning of lab!

In your Sakai dropbox, create a new folder named “lab1”. For this assignment you will be uploading two files into this folder—a text file with answers to some questions and a Python program.

PLEASE DO NOT RENAME YOUR FILES WHEN YOU UPLOAD THEM! Leave the “display name” as the name of the file.

Create a document named “`lab1-`*yourname*`.txt`” (or `.odt` if you are using LibreOffice) with your answers to the questions below that are marked “[**Answer**]”. Please number the answers to make it easier for me to find them (e.g., the first answer would be numbered “2”).

1 Some Network Tools

1. Read about the `netstat` command at `http://www.linuxhowtos.org/Network/netstat.htm`
2. [**Answer**] What does ICMP stand for? Briefly, what is it for? (You’ll need to research this in your textbook or on the Web.)
3. [**Answer**] How many ICMP packets are listed as being received; how many are listed as being sent? Paste the relevant output lines (including the command) from the `netstat` command into your answer document. To save on paper, delete any output lines that are not relevant to the question.
4. [**Answer**] Estimate how many total IP packets are received each second at your machine (of course this number can vary greatly depending on network traffic, applications running, etc.). Do this by using the `netstat -sw` command (first line of output shows total IP packets), the `date` command, and some basic arithmetic. Typing the command:

   ```
   date; netstat -sw
   ```

   will print out the date and time, followed by the output from `netstat`.

   Show me your work—paste the commands and the relevant output lines from `netstat` into your document and show me how you computed the packets per second. Again, delete irrelevant output lines to save paper.

5. [**Answer**] How many UDP ports are "listening" at your machine? Show me the output from the appropriate `netstat` command (Note: UDP is just another protocol, like TCP, so you should be able to deduce the correct `netstat` command from the examples.)

6. Read about the `dig` command at: `http://linux.about.com/od/commands/l/blcmdl1_dig.htm`. Try typing the commands:

   ```
   dig cs.allegheny.edu
   dig microsoft.com
   dig oracle.com
   ```

7. [**Answer**]
   (a) What IP address(es) are given for `cs.allegheny.edu`?
   (b) What IP address(es) are given for `microsoft.com`?
   (c) What IP address(es) are given for `oracle.com`?
   (d) What happens if, in your Web browser, you type in `http://` followed by any of the IP addresses you just listed? (i.e., "`http://xxx.xxx.xxx.xxx`" where "`xxx`..." is an IP address)

8. Read about the `whois` command at `http://www.cyberciti.biz/faq/linux-unix-command-to-find-who-owns-domainname/`

9. [**Answer**] Show the command and the relevant lines of output from a `whois` command to determine the names of the Administrative and Technical Contacts for `allegheny.edu`.

   Same question, but for `coveros.com`.

   Same question, but for `instapaper.com`.

10. [**Bonus Question—optional!**] Why did I choose `coveros.com` and `instapaper.com` as examples in the previous question?

2 A Simple Network Application

11. Create two Python programs named "`UDPClient.py`" and "`UDPServer.py`", using the code from pages 159 and 161 in your textbook. (If you forgot your book, the code is also in the Powerpoint slides for chapter 2, slides 98 and 99.)

    CHANGE THE NUMBERS FOR `serverPort` **FROM 12000 to 12345** in both programs.

    CHANGE THE `hostname` **TO THE NAME OF YOUR MACHINE** (e.g., "`aldenv100.allegheny.edu`")

    There is an error in the code for UDPClient.py! In the fourth line, replace "`socket.AF_INET`" with just "`AF_INET`" and replace "`socket.SOCK_DGRAM`" with just "`SOCK_DGRAM`".

    Open a second terminal window and navigate to the directory containing your `UDPServer.py` code. Run the command "`python UDPServer.py`". You should see a message telling you the server is ready.

In the first terminal window, make sure you are in the directory containing `UDPClient.py`. Run the command “`python UDPClient.py`”. You should see a prompt asking you to enter a sentence. When you enter one, you should see the same sentence converted into upper case.

Run the client program several more times to make sure things are working, then go to the server window and use CTRL-C to kill the program.

12. **[HAND IN]** Modify the program `UDPServer.py` so that the “`modifiedMessage`” includes your name (use the “+” operator to concatenate a string to the message). Test it to see that it works, i.e., the client program receives its original message back, with your name appended to it.

    Put a comment at the beginning with your name and “lab 1”.

    Then upload your modified `UDPServer.py` file to your drop box in the “lab1” folder.