CS544 -- Computer Networks
Spring 2011-2012
Professor Mike Kain

Network Protocol Implementation
Due Date: Saturday, June 9th, 2012 at 11:59pm EDT

Your final group assignment is to implement your protocol (probably a session layer) by writing a concurrent server (one that handles multiple clients simultaneously) and a client that use your stateful protocol. The goal of this assignment is to complete the process of designing a network protocol by implementing it, and showing that there is feedback in the design process through implementation. Depending on your protocol, it may be peer-to-peer or client/server.

Both the client and server can be written in (almost) any language, but C/C++ or JAVA is preferred. If other than those, please get our agreement before starting. We have to be able to run your programs when done. If you don't have a place to run it, we suggest the department's machines (Linux/Ubuntu). You can implement on other platforms as long as we can get to them to execute them ONLINE.

Requirements:

- STATEFUL: Both the client and server must implement and check & validate the statefulness of your protocol. THIS IS IMPORTANT
- CONCURRENT: The server must be able to handle multiple clients (either using a process/thread model or select) – both can be found in any programming text or see myself or Ann Marie.
- SERVICE: The server must bind to a hardcoded port number (you pick this value) and the client defaults to that port number.
- CLIENT: The client must be able to specify the hostname or IP address of the server (be able to specify either).
- UI: The user interface of the client isn't too important here – a command line UI is acceptable. It will be up to your protocol, but the protocol logic should be inside the client, not visible through the UI (only the client and server should know the actual protocol commands). What I mean here is that the user should not have to know the protocol commands to get it to work.

Please label areas of your code which address each of these requirements – it will make it easier to grade.

NOTES:

- Porting of the code for the socket algorithm(s) (the shell of the client and server) is acceptable – but every procedure MUST be accompanied by a reference in the header of where you took the routine (like a term paper). You MUST write your protocol implementation from scratch.

- You'll be graded on the implementation of your protocol and how correct and robust the implementation has become.
- Any use of open source frameworks or toolkits must be approved by the Professor before use.

Turn in (zipped together as groupxx.zip) through WebCT/BBVista:

- An updated complete proposal document (including any changes or updates we suggested and a section describing any differences since the original submission). The implementation must agree with the protocol design. Also include any performance implications that you find during your coding and testing.
- All (well-commented) source code. My basic rule is that all code should be commented in a way that others can pick up the code and instantly understand it – write comments the way you would want to see them if you picked up code and had to learn it (basically a lot of comments – each block should have some level of comments).
- A sample run from the client's point of view.
- Full pathnames to the executables. Your server should not be running when turned in. **We will verify your programs on-line!** If you develop your assignment on another machine than the main CS department machines, include the method to attach to those machines, any usercodes and passwords to logon and thereby test the assignment on-line.
- Any analysis about how robust your assignment is – do you think that it's tough to crack through fuzzing?

EXTRA CREDIT:

Have the client program dynamically find the server (e.g. just execute *client* with no parameters and the program finds the server listening on that port anywhere in the network). This is all I'll say about the extra credit -- the research is up to you. It is worth up to 10 extra points, depending on how creative you get (the more you use networking, the more points you get). Be sure to mention under what circumstances your approach will or will not work.

If you have any questions, please contact Ann Marie or myself. We will have office hours as often as possible to answer questions, and be available via e-mail and WebCT/BBVista, as well as before and after class.