CNT 5106 Computer Networks

Project I Individual Project

Handout 1/24, Due 2/14

Implementation of FTP client and server

1. **Description**

In this project, you will implement a simple version of ftp client/server software. It consists of two programs: ftpclient and ftpserver. First, ftpserver is started on a computer. It listens on a certain TCP port (such as 5106). Then, ftpclient is executed on the same computer; the server's port number are supplied in the command line, for example, "ftpclient 5106". The user can issue a command at the client side: "get <filename>", which is to retrieve a file from the server, or "upload < filename>", which is to upload a file to the server. Because the file could be arbitrarily large, you are required to split the file into chunks of 1K bytes and use a loop to send the chunks, each time one chunk.

For testing (which the TA will do), (1) the client will get downloadTestFile.pptx from the server and write the file to the local disk as newDownloadTestFile.pptx. and (2) the client will then upload uploadTestFile.pptx to the server, which will write the file to its local disk as newUploadTestFile.pptx. The reason to change the name of the file with prefix "new" is to allow you to place all files, including the ftpclient, ftpserver and the test files (downloadTestfile.pptx and uploadTestFile.pptx), in the same directory. After testing, the newly created files will not overwrite the original test files.

The test files can be found on Canvas. They are the same as chap1.pptx and chap2.pptx but under different names, downloadTestFile.pptx and uploadTestFile.pptx. These files are large. You will need to use a loop the break them into smaller segments and send each segment at a time.

The implementation does not have to conform to the ftp standard.

2. **Programming Environment**

Programming language: Java, C, C++, C#, Python
Operating System: Windows, Mac OS or Linux

Programming Tool: Eclipse, IntelliJ, Jcreator, Kawa, Netbeans, … whatever you like.

To use Eclipse, please go through the following list:

1. Download JDK from: https://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html

2. Download Eclipse from: http://www.eclipse.org/downloads/

3. Here is a link for eclipse tutorial:
http://eclipsetutorial.sourceforge.net/totalbeginner.html

4. Here is a tutorial for socket programming in Java:
http://java.sun.com/docs/books/tutorial/networking/sockets/

3. Code Submission

If you use Java, you will need to submit the following files: server.java, client.java, server.class, client.class, README.txt in a zipped directory, e.g., project1.rar. Please make sure to include server.class and client.class in the submission.

If you use C/C++/C# or Python please put all source files and executables in a zipped directory. Submit the project through Canvas:

1) Go to https://lss.at.ufl.edu/
2) Click "Login to e-Learning"
3) Login with your gator link username/password
4) Go in CNT 5106
5) Click "Assignments" and submit your project

This is an **individual** project. Students must submit their code via Canvas, by the deadline. We will run an automatic tool to catch submissions with identical or similar code. There will be no late submissions.