**Designing the Protocol**

What kinds of messages will be exchanged across the control channel?

* The Control Channel lasts throughout the whole ftp session.
* It’s used to transfer all commands (ls, get, and put) from client to server and all status/error messages from server to client.

How should the other side respond to the messages?

* For each command/request sent from the client to server, the server prints out the message indicating SUCCESS/FAILURE of the command.

What sizes/formats will the messages have?

* Due to the Internet being a packet switched network, large sizes of data will be split into multiple packets.
* There will be times where packets of data may arrive faster at the receiver than the others. In that case, the remaining packets/bytes can still be received but delayed.

What message exchanges have to take place in order to setup a file transfer channel?

* The user first provides the hostname of the remote host, causing the FTP client process in the local host to establish a TCP connection with the FTP server process in the remote host.
* The user then provides the user identification and password, which are sent over the TCP connection as part of FTP commands.
* Once the server has authorized the user, the user copies one or more files stored in the local file system into the remote file system (or vice versa).

How will the receiving side know when to start/stop receiving the file?

* Simply put, the client will have to tell the server whenever they want to start/stop receiving the file.

How to avoid overflowing TCP buffers?

* For a TCP/IP socket connection, the send and receive buffer sizes define the receive window.
* The receive window specifies the amount of data that can be sent and not received before the send is interrupted.
* If too much data is sent, it overruns the buffer and interrupts the transfer. The mechanism that controls data transfer interruptions is referred to as flow control.
* If the receive window size for TCP/IP buffers is too small, the receive window buffer is frequently overrun, and the flow control mechanism stops the data transfer until the receive buffer is empty.