

Christian Ciabattoni  
Jason Alvarez-Cohen

## Report Project 2

### High level description:

#### Server:

The server works by maintaining a vector of packets as the congestion window, which is adjustable based on the performance of the network in. When it receives a packet from the client, it first checks to see what kind of packet it is to determine which action to take. If it is not a duplicate ack or we have not added it to our queue, we create the appropriate response based on the given information and send it to the client. It supports timeouts, fast retransmit, congestion avoidance, and slow start as defined in the TCP Tahoe specification.

#### Client:

The client works maintaining a fixed congestion window, which is only used when we receive out of order packets. If we receive out of order packets, they are kept in the buffer until they receive the data in order to order them properly. In the event that we receive packets in order, we simply write them to the file. Our implementation supports timeouts, duplicate acking, and cumulative acking.

### Problems:

The problems that we faced came mainly when testing the program under extreme conditions. Our project works for the most part under various packet loss, packet delay, and packet reordering conditions, however, when they are set too high the program will just hang for long periods of time since it gets so far behind that it cannot process the incoming requests from the client.

### Building the project:

Once in the directory, type “make” and that will produce the relevant binaries.

### Testing:

We tested our code by first running it without any congestion conditions on a small file (4000 bytes approximately), then on a large file (4.2MB), and then with a small file under various congestion conditions. The reason we tested it this way was so that we could incrementally deal with the issues that arose from each type of test.

### Team Members:

Christian Ciabattoni (004-331-063): Built the server and the Packet framework.

Jason Alvarez-Cohen (504-487-052): Built the client and implemented the networking components.