CPS 472/572: Homework 3 **Dr. Clark**

UDP Applications
Due Monday February 24, 1997

For this homework you will experiment with a pair of programs that communicate using UDP. You will use the Sun systems in Anderson 131.

Part 1: Sample Programs

The first step is to compile and build the sample programs we went over in class. You can copy the programs from my account.

To do this, execute the following sequence of commands in a shell window:

1. "mkdir hwk3" - create a directory for the files
2. "cd hwk3" - change to that directory
3. "cp /home/rjc/pub/cps472/hwk3/* ." - copy the files into your new directory
4. "make" - use the make tool to compile and build the programs

The two programs are *udpclient* and *udpserver*. Before you can run them you must make two changes. First, you must choose a unique port number so you don't collide with other students in the class. The best way to do this is to add your login id number to the SERVER_UDP_PORT value defined in both *udpserver.c* and *udpclient.c* source files. You can get your unique login id number by running the *id* command and using the number given for *uid*.

The second change you must make is that you need to specify the IP address of the machine on which you will be running the server. To do this, do the following steps.

1. Select a system to run your server.
2. Determine the host name eg. sun131-33
3. "host sun131-33" - this command gives you the IP address for host sun131-33
4. Edit the *udpclient.c* file and change the SERVER_HOST_ADDR value to the IP address you found.
5. "make" - use the make tool to compile and build the programs

Now that you have the programs built you should run them and experiment with the output. Run the client on one system and the server on another. You should always start the server first. Experiment with different input values for bytes per packet and number of packets to transfer.

Answer the following questions.

1. You will probably find that the client does not always exit correctly and you sometimes have to stop it (by typing a $Ctrl - C$). Why does this happen?
2. What happens if you start two clients at the same time? Do they both exit correctly?
3. What happens if you start two servers at the same time on the same system? Why?

Part 2: Header Analysis

Now that you've experimented some with the programs you should be prepared to do some analysis of the data that is generated on the network when these programs run. To do this you will use a tool called *udpsnoop*. This tool gives a decoding of the headers in network packets. To use this tool you will need to specify the port number you selected for your client and server code. For example, if you selected port 23978 then you would type the command *udpsnoop 23978*.

Try this by running *udpsnoop* in a window while you run the client and server applications. You should see a verbose printout of the Ethernet, IP and UDP headers for the traffic generated by your application programs. Experiment with this program and **answer the following questions.**

1. What are the system names and IP addresses for the client and server systems. By client and server system I mean the system on which you run your client and the system on which you run your server.
2. What are the Ethernet addresses for the client and server systems.
3. Is the UDP checksum the same for all of the data packets? How about the IP checksum? If they are different, why?
4. What is the largest user data size you can send without having IP fragmentation in this network? In other words, what is the largest value you can give for bytes per packet to the client program without causing fragmentation?