Testing for Program #3 – Dropping Specific Packets

If you want to test your program #3 using a specific pattern of dropping packets (e.g. drop packets 20-32) here is how you can do it.

The CPE464 library we provide reads in environmental variables that you can set in your shell. If you set the variable CPE464_OVERRIDE_ERR_DROP to the correct values, this will cause our library to drop a specific set of packets.

Examples:

1) **To drop packets 14, 25, 44, and 45 on the *server* only.**

   a. In the window you are running your **server** enter the *export* command below and then run your server.

      - export CPE464_OVERRIDE_ERR_DROP=14,25,44,45
      - server 0 (this runs your server with a 0 error rate)

   b. In the window you are running rcopy do not enter any export command. Just run rcopy with a 0 error rate so that the only packets dropped are packets 14, 25, 44, and 45 sent from the server.

2) **To drop packets 20-25 on both *rcopy* and *server***:

   a. In the window you are running your **server** enter the export command and then run server

      - export CPE464_OVERRIDE_ERR_DROP=20,21,22,23,24,25
      - server 0 (assume server is assigned port number 43210

   b. In the window you are running **rcopy** enter the same export command then run rcopy:

      - export CPE464_OVERRIDE_ERR_DROP=20,21,22,23,24,25
      - rcopy file1 file2 10 1000 0 localhost 43210

Notes:

a. Do NOT add any spaces in the export command except between export and CPE464….

b. If you provide a drop override (e.g. export command) and provide an error percentage when you run the program both will take effect and you will drop the specified packets and randomly drop other packets.

c. To turn it off set the override value to -1:

   - export CPE464_OVERRIDE_ERR_DROP=-1

   (you can always just close your shell (e.g. window) and open a new window)

d. If you drop packet 1 this may cause your program to have problems. You may continue to fork() new children. This is because each child will inherit the drop pattern, and then drop packet 1, causing the server to fork() a new child which then drops packet 1. (To test this you would need to use a different way of dropping packet 1 that does not do this for every fork()ed child.)