Lab1 (20 points)

This assignment requires you to write a client and a server. The client will read a file (provided by instructor). The client will send the file, one line at a time. Each line will contain a set of key:value pairs. The format of the data is as follows:

version:1 destination:100 location:10 TTL:100 cmd:send msg:"hello world how are you today?" time:1228 date:12-1-1 version:1 destination:1001 location:20 TTL:300 cmd:send time:1228 date:12-1-1

Things to NOTE. Each key:value pair is separated by a space. There will be AT MOST one msg stanza per line. The message is indicated by the keyword msg:”text of message”. Note that spaces are allowed in the message and must be handled properly. No message may contain the ^ character (hint).

The client will send each line of the file, as-is, to the server. The server will receive the DGRAM and then parse the line into a set of key:value pairs. These should be stored in a structure on the server. The server MUST be able to parse the msg key and store the msg value properly. For example, if the server receives:

Msg:”hello world how are you today?” version:1 type:A

The server must store “hello world how are you today” (including the quotes) in the value associated with the key **msg**.

Your server will print out the keys/values it received. These should be printed out in fields of 20 characters each. If the server receives:

version:2 destination:100 location:10 TTL:100 cmd:send msg:"hello world how are you today?" time:1228 date:12-1-1 version:1 destination:1001 location:20 TTL:300 cmd:send time:1228 date:12-1-1

It should print:

Name Value

version 2

destination 100

location 10

TTL 100

cmd send

msg "hello world how are you today?"

time 1228

date 12-1-1

version 1

destination 1001

location 20

TTL 300

cmd send

time 1228

date 12-1-1

The order of the keys/values may be random.

Important: We will demo in class. We will go around the room. Be prepared to demo at the start of class on demo day. The submitted code will be used only to verify that you did not copy from others, to compile and re-run your program, to make sure you were indeed demonstrating your own code, and to grade for documentation of your code.

Write 2 programs called client1.c and server1.c. They will communicate via DGRAM sockets.

The server1 file will take the port number as a parameter. It will bind a datagram socket to that port number. It will then loop, waiting for client1 to send it data. It will print out the data that is sent.

The client1 file will take the ip address and port number of the server as parameters. The server should continue to run and accept messages from anywhere.

Submit well-documented and well indented code along with a README file explaining how to run

the program, and a makefile. Submit it using GitHub.

The grading rubric is as follows:

• Program correctness and robustness (what happens if I give garbage input): 80%

• Coding style (comments, indentations, README, Makefile): 20%