■ **ECE/CS 4984: Wireless Networking and Mobile Systems** ■

In-class Laboratory Exercise 7 (L7)

Part I – Objectives and Lab Materials

Objective:

The objective of this lab is to:

- Analyze the effect of the virtual sensing mechanism (RTS/CTS) on 802.11b performance.

After completing the assignment, you should be able to:

- Enumerate the drawbacks of the CSMA/CA protocol and how the virtual sensing mechanism helps to avoid them.
- List the trade-offs in enabling the exchange of RTS/CTS frames and discuss under which circumstances it provides the greatest benefits.

Hardware to be used in this lab assignment:

- Xircom 802.11b PC card adapter
- Dell Notebook, with a fully charged battery

Software to be used in this lab assignment:

- *iperf*: traffic generation and network performance measurement tool

Part II – Pre-lab Assignment

It is important that you complete the entire pre-lab assignment *before* you come to class. It is not fair to other students if you are not prepared for the in-class experiments

- Write two scripts to configure the *iperf* server and client. The server script (*iperf_RTS_server*) configures the server to receive UDP datagrams at ports 4984, 5000 and 5001 and at 2 Mbits/s. The server script should read the datagram length from the command prompt. The client script (*iperf_RTS_client*) should configure the client to transmit UDP datagrams at a rate of 2 Mbits/s to the server for 60 seconds. The client script should read the server's IP address, port number and the length of UDP datagrams from the command prompt. The *iperf* executable is located in /usr/local/bin. Store the scripts in /root/WMSD/labs/lab_7.

Part III – In-class Lab Assignment

You are expected to perform the following tasks.

We will study the effect of the RTS/CTS mechanism on nodes that are closely spaced and can hear each other. The experiment involves the measurement of throughput achieved by each transmitter, transferring UDP data to one receiver. We will vary the packet size and observe the effect of the virtual sensing mechanism on link throughput.

1. Teams consisting of four groups will perform the experiment. One group will serve as the receiver. The other transmitter groups must ensure that they are all within hearing range of one another.

2. Using *iwconfig*, set each of the four nodes to ad-hoc mode of operation. Set the ESSID to "wmsd*n*" where *n* is the group number corresponding to the receiver of the connection. Set the channel of operation to the one assigned to your team by the GTA and set the rate to 2 Mbits/s on each notebook. Set the IP address of your machine, using *ifconfig*, to 169.254.1.*x*, where *x* is the number corresponding to your group. Set the RTS threshold to 2300 bytes. Check whether all the notebooks are within hearing range of one another. (Each node should be able to ping the other three nodes in the team). Capture a screen shot of the ping output obtained at the receiver from each of the other three nodes and include it in your report. Check your channel of operation in the file /proc/driver/aironet/eth1/BSSList.

3. At the notebook behaving as a receiver, run the server script, *iperf_RTS_server*, in a terminal window by typing

   ./iperf_RTS_server len_of_dgram

   On the transmitters run the client script, *iperf_RTS_client*, by typing

   ./iperf_RTS_client server's_IP_address portnumber len_of_dgram

   Each of the three transmitters should transmit to a different port on the server. Start with a datagram length of 50 bytes.

4. At the *iperf* server, take note of the throughput corresponding to the data exchanges from each of the three clients. Capture a screen shot of the throughput returned by *iperf* at the server.

5. Repeat the transmission of UDP datagrams in steps of 200 bytes until the datagram length is equal to 1400 bytes. Note the throughput for each case.
6. Now we will study the effect of RTS packets as overhead on the throughput. Enable the virtual sensing mechanism by setting the RTS threshold to 20 bytes using *iwconfig*.
7. Repeat steps 2 through 5 and make a note of the link throughput returned by *iperf* at the server for each of the three transmitters.