In-class Laboratory Exercise 6 (L06)

Part I – Objectives and Laboratory Materials

Objectives:

The objectives of this laboratory are to:

- Illustrate the set up of a Bluetooth piconet; and
- Identify the potential for interference between Bluetooth and IEEE 802.11b networks.

After completing the assignment, you should be able to:

- Set up a Bluetooth connection between two or more nodes; and
- Describe methods to mitigate the interference of Bluetooth on IEEE 802.11b networks.

Hardware to be used in this laboratory assignment:

- Xircom credit card Bluetooth adapter
- Xircom 802.11b adapter
- Dell notebook computer (fully charged)

Software to be used in this laboratory assignment:

- For groups in a Bluetooth piconet, Bluetooth connection manager and Intellisync software running under Windows XP
- For groups in an IEEE 802.11b ad-hoc network, *netperf* running on the notebooks under Linux

Part II – Pre-laboratory Assignment

Complete the following tasks ***before*** the laboratory session.

- Read the *netperf* manual installed in /root/WNMS/labs/lab_6/netperf_manual.pdf in Linux on the notebook computer. Read Sections 3, 4, 7 and 8 of the *netperf* manual. You will use *netperf* to measure the throughput. Alternatively, the *netperf* manual can also be obtained from the following location:

  http://www.netperf.org/netperf/training/Netperf.html

  If you want to learn more, see http://www.netperf.org/netperf/NetperfPage.html.

- Write a script for Linux that invokes the *netperf* client (*netperf_client*) to transfer UDP segments for 15 seconds. Ensure that the segment size is chosen to avoid any IP fragmentation. Save the script in the /root/WNMS/lab_6 folder.

- Read the article by J. del Prado and S. Choi, "Experimental study on co-existence of 802.11b with alien devices," *Proceedings of the IEEE Vehicular Technology Conference*, vol. 2, pp. 977-981, October 2001. This article is through IEEE Xplore using the URL below. (Access requires a Virginia Tech IP address or login through the Virginia Tech Library's proxy).

  http://ieeexplore.ieee.org/iel5/7588/20687/00956920.pdf

Part III – In-class Laboratory Assignment

We will study interference between Bluetooth and IEEE 802.11b. Students will form teams consisting of four groups (eight students). Two groups will establish a Bluetooth piconet and the other two groups will setup an IEEE 802.11b network in ad-hoc mode. We will study the difference in throughput for the IEEE 802.11b network with and without interference from the Bluetooth piconet. The configuration is shown in Figure 1. As suggested in Figure 1the IEEE 802.11b *dest* node and Bluetooth devices should be in close proximity to each other relative to the distance between the IEEE 802.11b *src* node and IEEE 802.11b *dest* node.

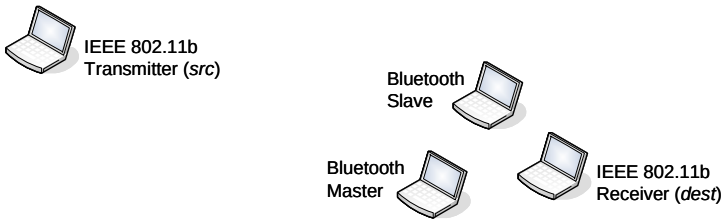

Figure 1. Configuration for the interference experiment.

Experimental Configuration

The first task is to set-up the experiment. Two groups form an IEEE 802.11b network in ad-hoc mode using notebook computers running Linux as described below. The two groups responsible for creating the ad-hoc IEEE 802.11b network will form the data source and the data destination in the ad-hoc network. One group will be the *src* and the other will be the *dest*.

- Boot the two notebooks in Linux with the IEEE 802.11b cards in the PC Card slots. Use the *iwconfig* command in Linux to set the mode of operation to Ad-Hoc, the ESSID to "wnms*groupnumber*" where *groupnumber* is the two-digit number of the group acting as *src*. Set the transmission power to 1 mW or 0 dBm. Make a note of your settings. Set the IP address of the *src* and *dest* nodes to be 169.254.1.*groupnumber* where *groupnumber* is the number associated with each group in the experiment. Set the channel to the value assigned to your team by the GTA.

Note: There is a bug in *iwconfig*. When using *iwconfig* to change the channel, you must specify a channel number one greater than the actual channel number that you want the network to use. For example, if you are to use Channel 11, you should specify the command as:

```
iwconfig eth1 mode Ad-Hoc essid sample channel 12
```

- Establish a connection between the *src* and *dest* nodes. Ping the *src* node from the *dest* node and capture a screen shot of the ping output, thus validating the ad-hoc connection.

To introduce interference, we now establish a Bluetooth piconet near the IEEE 802.11b *dest* node, as shown in Figure 1.

- Start by inserting the Xircom Bluetooth card in each of the notebook computers. Using the terminology of the Bluetooth specification, one of the groups will act as the "master" of a Bluetooth piconet and the other group will be the "slave." After inserting the Bluetooth adapter in the PC Card slot, click on the "Bluetooth places" icon in the task bar. When the manager appears on your screen, check the properties of your card. This can be done by right clicking on the icon corresponding to the name of your notebook and clicking on the "Properties" button.
- The properties of the Bluetooth device include the name of the device, the 48-bit MAC address of the Bluetooth adapter, and the device class and connection profile. Ensure that the encryption is

turned off for the connection between the communicating devices. Note the MAC address of your device.

- A Bluetooth device can be either in "non-pairable" mode or in "pairable" mode. In pairable mode, the Bluetooth device accepts paring, i.e. creation of bonds, initiated by a remote device, and in non-pairable mode it does not. Set the pairable mode to "Bondable" on your device. After confirming the settings on the notebook computers involved in the set up of the Bluetooth piconet, the group that is acting as master of the piconet should right-click on the "New" icon and click on the "Discover all" option (see Figure 2). The group acting as "slave" should right-click on the icon corresponding to the notebook name and click on the "Discoverable" icon (see Figure 3). With this, the "slave" makes itself available to be discovered by the "master." When the "slave" device is discovered, the details about it are displayed at the "master" under the "New" icon.

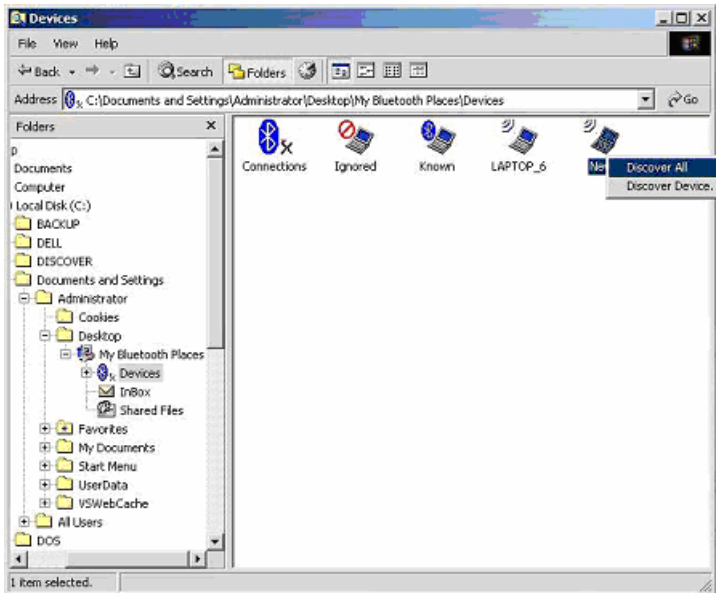

Figure 2. The master device's "Bluetooth places" window.

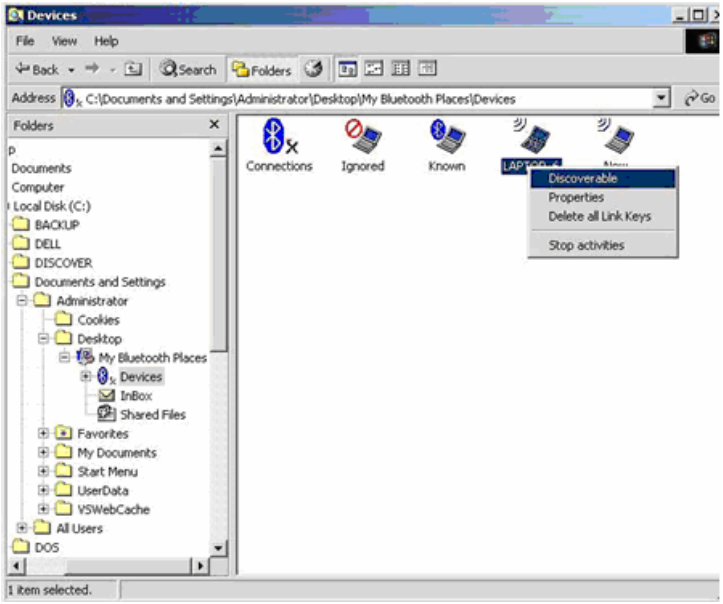

Figure 3. The slave device's "Bluetooth places" window.

- Double-click the "New" icon to display the discovered "slave." Double-click on the "slave" and there will be a list of profiles available on that "slave." Double-click on the "Intellisync" profile and there will be an option of establishing a one-time or a permanent connection. Establish a permanent connection between the "master" and the "slave" such that in the event of a disconnection, the two devices will be connected again. Select the "permanent" radio button and click OK.
- One of the main applications of Bluetooth is to serve as a "cable-replacement" technology; it provides emulation of serial data transfer between two devices. The Bluetooth devices that you are using can emulate serial ports. When you use the synchronization software, Intellisync, you make use of serial port COM 12. You have now established a Bluetooth piconet. The set-up of the experiment is now complete.

Throughput Measurements

With the IEEE 802.11b and Bluetooth networks in place, we can test the throughput between the *dest* and *src* nodes in the IEEE 802.11b network.

- We will measure the throughput of the connection in the ad-hoc IEEE 802.11b network in the absence and then in the presence of Bluetooth interference using *netperf*. The *netperf* binary is located in /usr/local/netperf and is linked at /usr/local/bin, so it can be executed from any directory in Linux.
- Stop the Bluetooth radio on both, the "master" and the "slave" devices. To stop the Bluetooth radio, right-click on the  icon in the task bar and then uncheck the "Radio on" option.
- Start the *netperf* server on the *dest* node by typing *netserver* in the terminal window. netserver is configured to listen to connections on port 12865. Run the *netperf* client script (*netperf_client*) saved in the /root/WNMS/labs/lab6 folder by typing *./netperf_client* in the terminal window on the *src* node.
- Measure the throughput returned by *netperf* for a data rate of 1 Mbps. The data rate can be varied using using *iwconfig*. The signal level can be read by typing *iwconfig* in the terminal window. Take the measurement at a signal level of -60 dBm at the *src* node. Perform three such data transfers using *netperf* and record the throughput values for each of the three transfers.
- Capture a screen shot of the *netperf* terminal window for the report. The screen shot should be taken for only one of the three readings of throughput at 1 Mbps. To take a snapshot of a screen in Linux, click on the "K" icon in the taskbar and go to graphics>KSnapshot. Use KSnapshot to take a screen shot of the *netperf* window.

We now introduce interference in the form of a data transfer in the Bluetooth piconet.

- Start the Bluetooth radio on the "master" and the "slave" devices by clicking on the icon in the task bar, as done above for stopping the Bluetooth radio. Right-click on the same icon in the task bar after your radio is "on" and click on "Open my Bluetooth places." When the manager appears on your screen, on the "master" device, double-click on the "Connections" icon in the left-hand frame. You will see the permanent connection being displayed as shown in Figure 4.

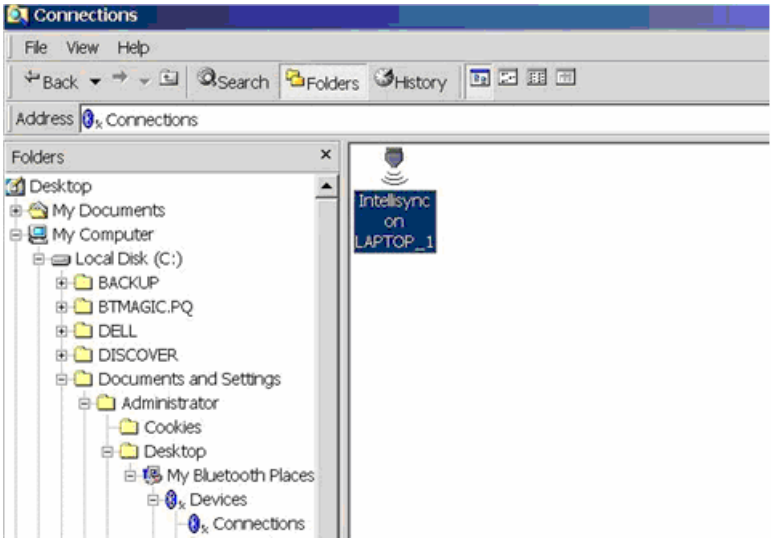

Figure 4. The "Connections" window under "My Bluetooth places."

- The "master" connects to the slave using Intellisync software. Start the Intellisync agent on both devices by going to Start>Programs>Intellisync>Intellisync agent. Using the Intellisync agent on the desktop, select the "File Transfer" icon on both the notebook computers, as shown in Figure 5.

Figure 5. The Intellisync agent.

- A connection will be established between the two Bluetooth devices with an accompanying sound. A security window will appear on the screen if the connection is being established for the first time. Under this security setting, select the "Security" tab and allow the remote user to access your inbox only. Select the "Inbox transfer only" button, as shown in Figure 6.

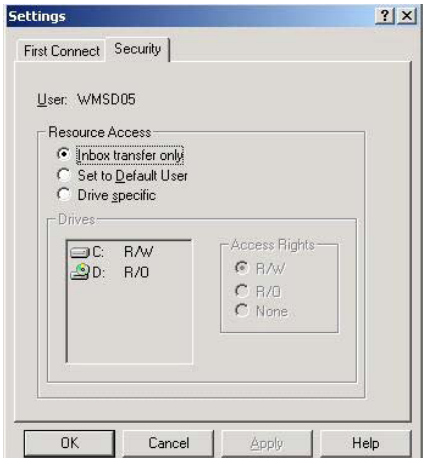

Figure 6. Security settings for Intellisync.

- To start the data transfer, on the "slave" system, select the file "pockettvsetup-0.9.4.exe" (which is interesting for this experiment only because it is large) in C:\WNMS\iPAQ and start its transfer to the "master" device. The file can be chosen from the right-hand side File Transfer explorer window. Right-click on the file and select the "master" as shown in Figure 7 ("Wmsd05" in this example).

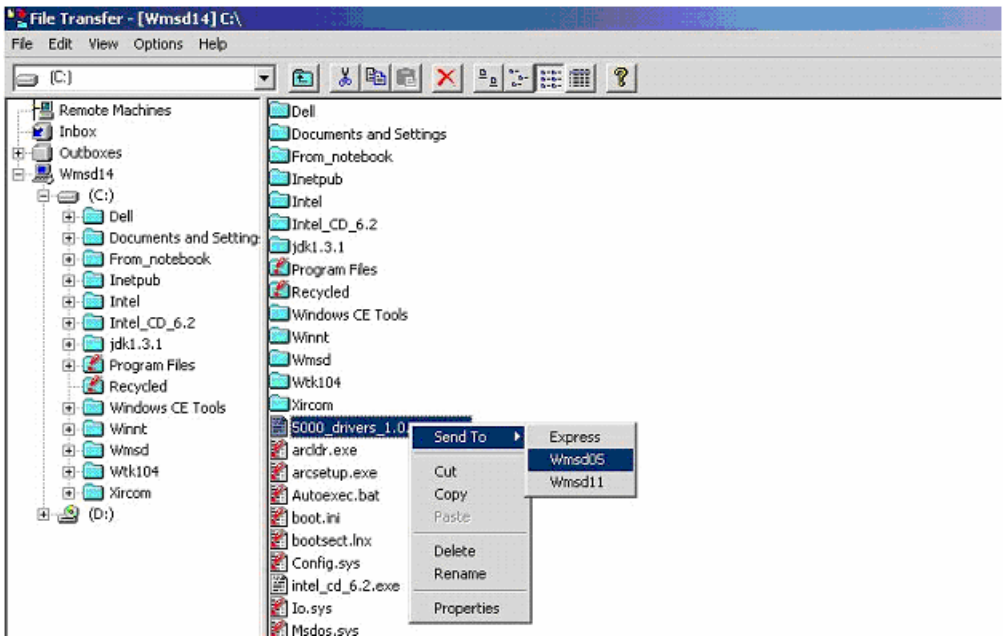

Figure 7. The File Transfer explorer window.

- At this point, the Bluetooth piconet is interfering with the IEEE 802.11b network due to the data transfer between two Bluetooth devices. Now measure the throughput of the IEEE 802.11b connection using *netperf*. Start the *netperf_client* at the *src* node and measure the throughput. While you measure the IEEE 802.11b throughput, ensure that the Bluetooth devices are transferring data. Perform three such data transfers using *netperf* and record the throughput values for each of the three transfers.
- Report the throughput for the IEEE 802.11b connection. Capture a screen shot of the *netperf* terminal window, to be included in the report for the associated at-home exercise. The screen shot should be taken for only one of the three readings of throughput at 1 Mbps with Bluetooth interference. To take a snapshot using Linux, click on the "K" icon in the taskbar and go to graphics>KSnapshot. Use KSnapshot to take a screen shot of the *netperf* window.
- Also, take a snapshot of the Bluetooth data transfer for one of the three trials.

Note: While transferring data using for different trials, delete the file being transferred after it is transferred to the "master" device and re-transfer the same file for the next trial.

- Repeat the above procedure to measure throughput for different IEEE 802.11b data rates of 2 Mbps and "auto." Do this in the absence and in the presence of an interfering Bluetooth connection as shown in Figure 1. Use *iwconfig* to change the data rate of the ad-hoc IEEE 802.11b network. Note that the "auto" data rate means that the IEEE 802.11b link will try to maintain a theoretical bandwidth of 11 Mbps, but it may reduce bandwidth to less than 11 Mbps, depending on current channel conditions. Record the throughput results for the three trials with and without Bluetooth interference for both data rate settings.