University of Maryland Baltimore County
Department of Computer Science and Electrical Engineering

CMPE 684, Wireless Sensor Networks (WSN)

Assignment #3 Due: Thursday 11/1/2018 (until Midnight)

Objective:

To experience the development environment for sensor network applications and gain hands-on experience by simulating the network operation. In this homework you are to get familiar with TinyOS and its programming language, nesC, and its simulation environment TOSSIM.

Statement:

The TA has prepared a simulation for a wireless sensor network consisting of 27 nodes using TOSSIM. You should download this simulation environment by following the instruction provided below. In the simulation, the node with ID 0 (zero) is set to be the data sink (Base-station; BS). Each sensor periodically generates a data packet that gets relayed over a multi-hop route towards the BS. The BS reports statistics about the network; e.g., number of received packets, and average delay of packet delivery. Ideally speaking, nodes should work normally and disseminate data according to the routing table in a timely fashion. However, the provided code incorporates some attacks that can be detrimental to the network operation. Some of the nodes in the simulated network are programmed to act maliciously by launching one or multiple of the incorporated attacks. You are requested to run the program, examine its output, read the code and understand it before attempting to solve the homework assignment. Please refer to "TinyOS and TOSSIM Development Environment" document that has been provided by the TA for setting up the TinyOS development environment. Also further information about TinyOS, nesC and TOSSIM is available via http://www.tinyos.net/.

How to get and run TA provided code:

The development environment, tutorial and HW code are available from following links:

Please download:

1. "TinyOS and TOSSIM Development Environment" document (the file: "A_Guide_on_TinyOS_and_TOSSIM_Development_Environment.pdf" from the class blackboard)
2. Fedora21_TinyOS_v3.ova (https://drive.google.com/drive/folders/1VDiO7h3_OBALF4mEomxngWVtVkZjPRzP?usp=sharing)
3. VirtualBox (https://www.virtualbox.org/wiki/Downloads)
4. HW3_WSN_Provided.zip (the file: "HW3_WSN_2018.zip" from the class blackboard)

Follow the instructions in "TinyOS and TOSSIM Development Environment" document to get yourself familiar with the basics and setup the development environment. Pay special attention to TOSSIM section. Run the code by issuing following two commands from within the development environment (Fedora21):

Command 1: This command will compile your code

```
make micaz sim
```

Command 2: The following command will run your code and save the output displayed on the screen to a file "`HW3Output.txt`".

```
./simulate.py > HW3Output.txt
```

The following files are included in the "HW3_WSN_Provided.zip" file:

1. HW3C.nc: main file containing HW3C module and its implementation. (You will do your changes here).
2. HW3APPC.nc: configuration module to wire HW3C.nc to TinyOS components.
3. Makefile: to compile main modules and required packet components for the project.
4. simulate.py: a python program to run the TOSSIM simulation.
    a. You might add, remove or comment dbg (print) statements defined in this file.
    b. You might need to increase/decrease simulation time for your own tests, but the HW results and answers should be based on the time that was provided "99999".
    c. You might want to increase/decrease the number of simulated nodes (number_of_nodes) to values that can help you understand the code but HW results and answers should be based on having 27 nodes.
5. topology.txt: defines communication links among sensor nodes. For simplicity we have defined same gain value for all connections. Note: Routing can only use a valid connection.
6. noise_short.txt: defines a noise model for radio layer.

Compile the code for simulation and run it on the same virtual machine (The Fedora21 VM), it should run without any problem, if you have any issue with getting it to run then read the provided document again. Also search the internet for information that is not in the document.

Homework Questions:

1. Based on the provided code, draw a simple diagram that shows the routing paths of this network. (20 points)
2. After the network stabilizes, i.e., all nodes are activated and have sent at least one message, provide the following details: (15 points)
    a. The total number of received packets at the bases station.
    b. The average delivery delay of packets at the base station.
    c. The simulation time at which the last node in the network got activated, i.e. sent its first message.
3. What is the maximum transmission rate for a node in the simulation? [*hint: TOSSIM is an event based simulator where the event triggering rate is set as a parameter; events include transmission of a packet. In other words, the answer is in the code and not in the simulation output*]. (10 points)
4. Based on the provided code, identify nodes with malicious behavior. Use the following table to help you consolidate your answer. (15 points)

No.	Node ID	Malicious behavior type
1		
2		
…		

5. The adversary falls short in realizing the intended attacks and some of the implemented malicious behaviors ended up having no negative effect on the network. Please analyze the code and identify which of the malicious behaviors are detrimental verse others that are not and explain the effect of malicious ones on the network traffic. (20 points)
6. Based on the provided code, what might cause a packet to be dropped? List all possible cases and explain each. (20 points)

Submission Instruction

Step 1: Consolidate your answers as a single PDF file with your name written inside the file and on each page. The PDF file should be clear, readable resolution, cropped to the page size (i.e. if you take pictures of the answer sheet and convert to .pdf, make sure only the page is displayed) with more than 90% of the page covering your answer and a total PDF file size less than 2MB.

Step 2: Save the output of your simulator in a file named "`HW3Output.txt`".

Step 2: Generate a compressed file with the extension .zip in which (i) add your PDF file that you generated in Step 1 above and (ii) the output file of your simulator in Step 2. Use first and last name to name the zip file.

Step 3: submit the .zip file on blackboard. Do not submit code files or any VM software.

Note: Selected students will be asked to demonstrate their understanding to the TA/Grader by answering questions about the provided code and how they obtained the answers. Failure in demonstrating a good understanding of the code may lead to a penalty up to 90% of the total points obtained.