PROCEEDINGS OF
THE 2007 INTERNATIONAL CONFERENCE ON
SECURITY & MANAGEMENT

SAM'07

Editors

Selim Aissi
Hamid R. Arabnia

Associate Editors

Kevin Daimi
Danilo Gligoroski, George Markowsky
Ashu M. G. Solo

WORLDCOMP'07
June 25-28, 2007
Las Vegas Nevada, USA
www.world-academy-of-science.org

Contents

SESSION: INTRUSION DETECTION

SESSION: BIOMETRIC, AUTHENTICATION, STEGANOGRAPHY

SESSION: CRYPTOGRAPHY

SESSION: SECURITY AND PRIVACY

SESSION: SYSTEMS AND ALGORITHMS

Intrusion Detection in Wireless Sensor Networks

Hong N. Nguyen
School of Electrical Engineering and Computer Science
University of Central Florida
Orlando, Florida, USA
Ravi Palaniappan
Institute for Simulation and Training
University of Central Florida
Orlando, Florida, USA
Nevin Aydin
Department of International Logistics and Transportation
Beykent University
Istanbul, Turkey
Shiyuan Jin
School of Electrical Engineering and Computer Science
University of Central Florida
Orlando, Florida, USA
Damla Turgut
School of Electrical Engineering and Computer Science
University of Central Florida
Orlando, Florida, USA

Abstract

There are several applications that use sensor motes and researchers continue to explore additional applications. For the intruder detection application, we use a set of Berkley mica2 motes on TinyOS operating system. Different types of sensors such as pressure, light, and so on can be used to identify the presence of an intruder in the field. In our case, we choose light sensors for the detection. When an intruder crosses the monitored environment, the system detects the changes on the light values. Any change greater than a pre-defined threshold indicates the presence of an intruder. An integrated webcam is used to take snapshot of the intruder and transmit the pictures through the network to a remote station. The basic motivation of this paper is that a sensor-based web system can be used to detect any intruder in a specific area from a remote location.

Keywords: intrusion detection, sensor networks

1 Intruction

Over the years, many people have seen the use of alarm systems and video cameras in combination to detect and prevent intruders. A complete security system requires large numbers of cameras with alert operators who are actively looking for intruders or suspicious activity, which is not an effective way to detect intruders. Therefore, this paper is concentrated on looking for a technology that is easy to deploy and non-intrusively locate targets and intruders. A system should alert the operator to look at a specific area only where intrusion is detected by the sensor motes.

There are several sensor motes which collect various types of data, such as light level, pressure, and so on. The sensor nodes will use different threshold values for different types of applications. Each sensor node communicates with each other and transmits the data to the central control station which is the stargate computer. More details of this stargate computer will be discussed later on. Sensor motes operating at 900 MHz frequency in ad hoc mode

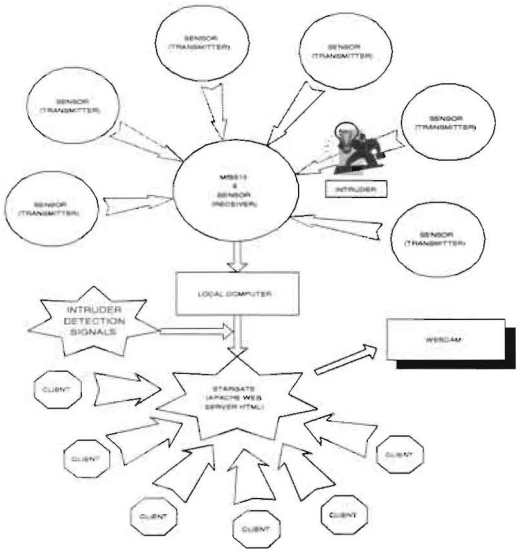

Figure 1: System overview

will be used to detect intruders in a monitored field. The real-time sensor data are used to check for the presence of an intruder. First, these motes are programmed to send the sensor data at a preset frequency. then, this data is collected and monitored from the motes using a stargate computer in a wireless mode. Finally, a webcam takes pictures of the intruder and transmits them to the Apache web server where such pictures can be accessed by security personnels through the Internet. A baseline data and background noise calibrations are needed in order to carry out the measurements. For example, when someone crosses close to the motes located on the ground, there would be a change in pressure or light sensitivity, which in turn might indicate an intruder. Figure 1 illustrates the system overview.

The connection between sensors and MIB510 or between clients and stargate computer are all wireless. The MIB510 is linked to a local computer by RS-232 serial port cable. The same connection is used between this local computer and stargate computer. A webcam is connected to stargate computer by a regular cable through a USB port provided on stargate. The system has a unique feature of remote monitoring that helps the user to control the system from a distant location. This work is done as part of a large sensor web project for the Office of Naval Research to develop a long term monitoring systems for border security along the southern borders of Arizona. Our contribution is the development of this proof-of-concept sensor web system which can be deployed under different environments.

2 Related work

Much research work has been done in wireless sensor networks with varying degree of success. Most researches are focused on simulation analysis. Onat et. al., [1] introduced a novel anomaly detection based security scheme for large scale sensor networks that exploits the stability in their neighborhood information. In the simulation, each node builds a simple statistical model of its neighbors's behavior, and changes can be detected based on these statistics. Roman et. al., [2] proposed a general intrusion detection system architecture for static sensor networks, where some nodes are able to choose independently to monitor the communications in their neighborhood. Blumenthal et. al., [3], described a software architecture for mobile sensor networks. This work mainly discusses a framework to simplify the development of software for sensor network applications. Additional research work can be found on [4] and [5].

Lubrin et. al., [6] developed a mote based wireless sensor network with remote monitoring capabilities using a PDA to display patient vital information such as heart rate, body temperature, and so on. The PDA (mobile monitor) sends this data through the internet to a central database server which uses Microsoft IIS to interface with the PDA. With this type of approach, there is a possibility of data interception when transferring vital and confidential patient data over the internet. Using secure shell software can be one solution to avoid this problem.

Hamrita et. al., [7] demonstrated an event driven smart sensor and RFID reader integrated system by deploying wireless smart sensors in a controlled environment. "enventListener" program, a piece of software package in nesC, is used to write the sensor readings of the environmental parameters to the database enabling real time remote monitoring of the system. The authors also discuss setting up a database on a server and access the network by querying through a web form to be able to monitor the system. However, they were unable to find sufficient documentation on the RFID reader to integrate it with the MICA2DOT mote, thus they cannot capture the actual RFID read event.

In [8], the authors discussed the problem of tracking objects with sparsely located binary sensors. They argued that tracking with sensor net-

work localization was complicated and presented many problems such as the inaccuracy of sensors. Also, sensor network had a low detection probability of tracking and high false detection probabilities due to a limited power supply and operates in the low signal to noise ratio (SNR) regime. Thus, they developed a distributed tracking algorithm based on the formulation and over the finite state space of sensor without sensor model and sensor network localization. In summary, this system was suitable to address indoor tracking problems where both the level of tracking an object movement and sensor network self-localization were high. For outdoor tracking application, it was only helpful when the degree of the passage connectivity graph of a sensor network was small.

Demirkol et. al., [9] conducted simulations of packet traffic modeling on wireless sensor networks for intrusion detection. Simulation parameters used are the number of sensor nodes, surveillance area, sensing range, and sampling interval.

3 Hardware and software components

This work requires many different components, most of which are from Crossbow Inc., [10]. In this section, a greater detail on the components used will be explained. The hardware components include stargate computer, MICA2 motes, MIB510 serial interface board, MTS310 sensor board, and a webcam. The software components are Xlisten, cygwin, Java JDK, and apache web-server.

Stargate. Stargate is a powerful single board computer with enhanced communications and sensor signal processing capabilities. It supports applications around TinyOS based wireless sensor networks and smart dust technology. Figure 2 shows a typical stargate.

Stargate has 400 MHz RISC processor, 64MB RAM, 32MB flash with a size of 3.5 x 2.5 inches [11]. It also has one type II compact flash dot (a 802.11*b* wireless compact flash card). A 256MB SanDisk compact flash card is used to have sufficient storage space for our database. This size of card should provide a couple years worth of space to store sensor data.

One of the interesting uses for the stargate is an application server. It is a remotely deployed stargate configured with software for local management of a sensor network. There are two server software that can be installed: Apache web-server for web-based applications and a Java runtime. A

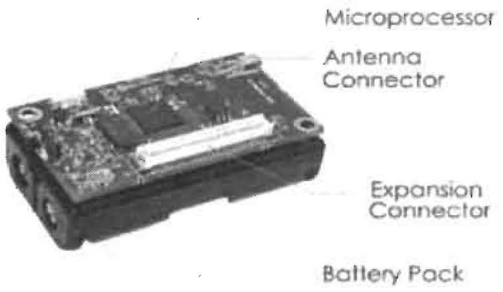

Figure 3: Crossbow MICA2 measurement system.

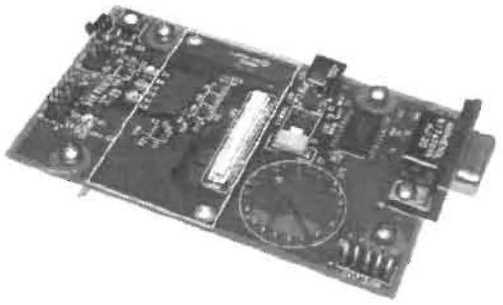

Figure 4: Serial interface board.

version of the Apache web-server is located on the stargate support CDROM. In this particular application, Apache web-server is used to display picture of the intruder.

MICA2 Motes. The MICA2 mote is a third generation mote sensors. It is designed mostly for embedded sensor networks. Its frequency is 916 MHz and can be chosen in any single frequency range from 903 MHz to 927 MHz. For outdoor application, the range can reach to 70 feet. If the MICA2 is on the ground, however, its range decrease to 40 feet. Similarly, for indoor application, its range changes between 50 and 70 feet. When there is multipath distortion, which can block the process of transmitting data, its range is 30 feet. Figure 3 shows a typical MICA2 used.

MIB510 Serial Interface Board. This serial interface board allows for the aggregation of sensor network data on any standard computer platforms. It acts as a base station for wireless sensor network with the MICA2 motes. Figure 4 shows a typical MIB510 used.

MTS310 Sensor Board. This is a flexible sensor board with a variety of sensing modalities

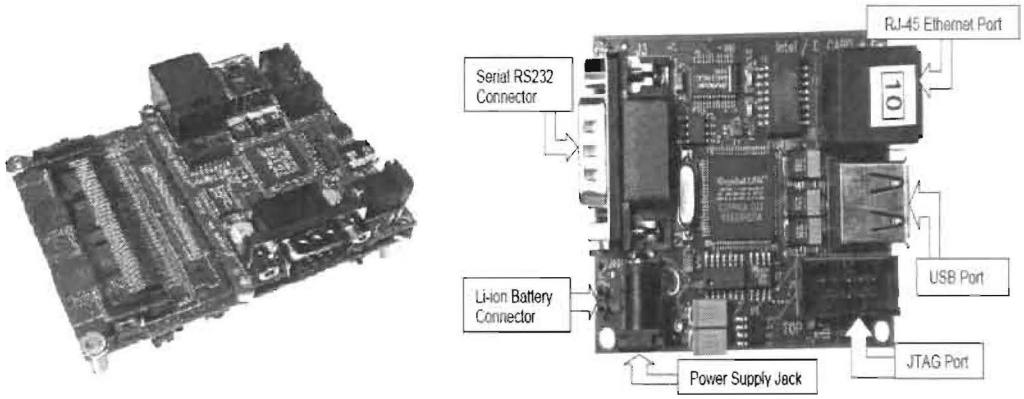

Figure 2: Stargate development platform: processor board (left) and daughter card (right).

including 2-axis accelerometer (ADXL202), 2-axis magnetometer, light, temperature, acoustic, and sounder.

Because of the fast growth in this technology, the price of motes fall and their capabilities rise along with the rest of semiconductor technology; those wireless sensors are used for boosting productivity, opening fresh avenues for scientific research, and enabling creative ways to prevent and respond to emergencies, environmental, and military applications. This type of sensor nodes run on TinyOS operating system. These sensors link up with their neighbors from the moment they are turned on. Depending on the foliage and environmental conditions, the radio range differ, lower the radio frequencies longer the ranges in an outdoor deployment. At 433 MHz, the range can be between 200 to 500 feet, and at 916 MHz on the other hand, the range varies from 100 to 300 feet. MTS310 is suitable for a wide range of applications keeping in mind that sensor units should be placed at least 1 to 3 feet above the ground to maximize the communication range. Placing units at ground, grass or other foliage are factors of decreasing the communication radius. Figure 5 illustrates the components of MTS310.

Webcam. The webcam is used to take picture of any intruder and send it to the web-server in a real-time. However, there are only two suitable units with stargate because of their special drivers. The reason is that stargate uses Linux, which provides support for Philips USB webcams and OV-cam drivers.

Xlisten. This program is supplied by Crossbow Inc. [10]. As its name indicates, its main function is to listen for incoming sensor data messages,

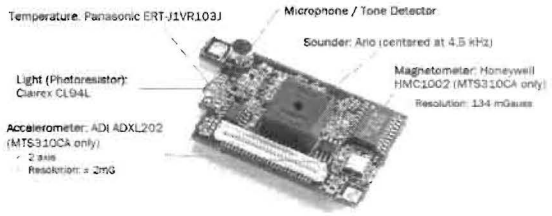

Figure 5: MTS310 sensor board.

such as temperature, humidity, and so on in a serial port. On stargate, it acts as the intermediary between the sensor readings from the wireless network of sensors and the Postgres database installed on stargate. Xlisten is able to recognize and interpret packets in a standardized format, including node ID, parent, sensor-board ID, and voltage. Data transmitted by the motes is either a raw analog or digital reading.

As mentioned earlier, final conversion to engineering units is done by Xlisten. The full C source code for conversion is available and provides a good reference for converting sensor readings for the entire line of Crossbow wireless products [10].

Cygwin. Cygwin is developed by Cygnus Solutions Inc [12]. Cygwin allows many UNIX applications to run on a windows platform. Mainly, it is used to port software that runs on POSIX systems to run on Windows. Cygwin has a GNU development toolchain which allows basic software development tasks and some application programs equivalent to common programs on the Unix system to run on windows.

Java JDK. JDK is used to compile and run

mote-test and other software applications. The communications API package, javax.comm, provides applications an access to RS-232 hardware or serial ports.

Apache web-server. The Apache web-server allows secure and reliable remote web-based client connections. For example, in our work, the files of a personal computer can be shared over the Internet. Here, Apache web-server is installed at stargate computer. Since the files are located in the Apache's document root, they can be easily shared. This software provides the designer an ability to preview and test the code while the code is being developed. Basically, this server is used to upload and update new images of any intruders collected from the stargate. The stargate runs an HTML file to check the intruder detection signal sent from the local computer. When there is a change in the signal, the html file immediately triggers the remote machine to execute the bash shell (webcam.sh) to take a picture. This web-server is refreshed every second to ensure that new pictures are updated.

When an intruder crosses the monitored field, some of the sensor motes detect a change in the intensity of the light since the intruder might obstruct the ambient light conditions in the environment in which the sensors are programmed for. When this happens, the system concludes that the light intensity crossed the threshold for triggering the alarm to indicate the presence of an intruder.

4 System Implementation

Sensor data can be accessed either from PostgreSQL database or from raw data streams. It is inefficient to read real-time data directly from the database because each time a packet is read, the JDBC record-set need to be updated. The following is an example of a raw data packet. The first three bytes in the packets are used to indicate the start of a new set of data according to Crossbow's manual for the raw data's message format.

```
FF FF 007D 1D 84010500 A801 F2011602 EF
013800780229032703000000000000000000
```

Each data packet contains several fields of data. The overall message format is as follows:

```
Destination address: 7D 1D
Message handler ID: 84
Group ID: 01
```

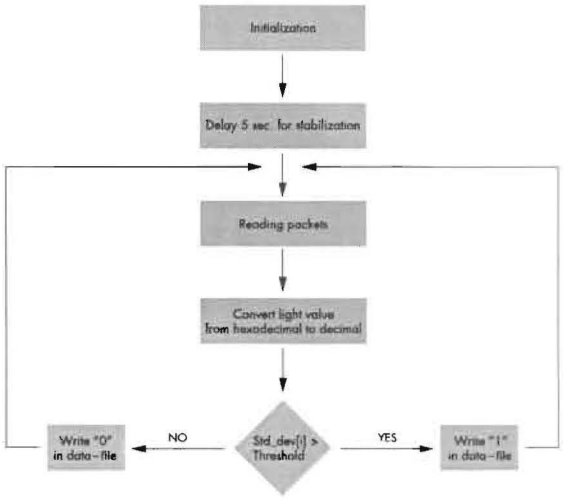

Figure 6: Calculation on the changes of the light values

```
Source address: 05 00
Temperature: F2 01
Light: 16 02
Microphone: EF 01
accelX: 38 00
accelY: 78 02
magX: 29 03
magY: 27 03
```

where $accelX$ and $accelY$ represent values of accelerometer, $magX$ and $magY$ are values of magnetometer which measures the strength of magnet field. The MTS310 MICA sensor board has five sensors: accelerometer, magnetometer, microphone, light, and temperature. In our application, we are interested in the changes of the light values which trigger an intruder detection mechanism. Figure 6 is presents our design.

In our application, we have used three sensors: two senders and one receiver. In our experimental set up, the threshold is set as 5.5. However, it can be tuned to different testing environments. Initially, our program waits for 5 seconds for signal stabilization since the initial signals are unstable. Upon activation, the local computer collects sensor data from the motes placed at random locations on the field through the serial MIB510 connector. After receiving each packet, the 14th and 15th bytes (corresponding to light data) of each packet are selected and converted into decimal data values. The variations of light data is obtained from remote sensors. Instead of having the sensor system to read a threshold level, the local computer writes a binary

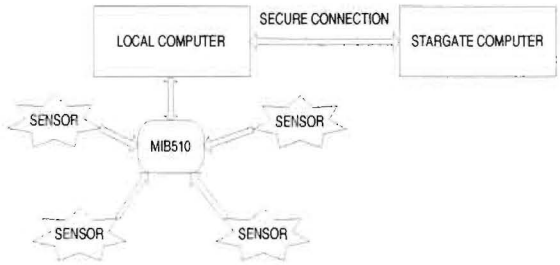

Figure 7: Secure connection.

code to a data-file. If the light_changes parameter (standard deviation) exceeds a pre-defined threshold, the Listen.java code starts to write a value of 1 to the data-file meaning that there is an intruder. Otherwise, a 0 is written to indicate an absence of an intruder. This "file writing" operation in the sensor Java code is continuously run by the local computer. After the detection is achieved, the system delays the next set of data collection for 5 seconds for the system to synchronize and convey the presence of the intruder to the stargate. This data-file is then transfered to the apache web-server.

Whenever an intruder crosses, the change of the light value triggers the remote camera and an alarm. This is achieved by sending the data file to the stargate at each instance by a network cable. This is done by establishing a "trusted" secure connection between the local computer and the stargate as demonstrated in Figure 7.

This secure connection is necessary in order to prevent data interruption and corruption. This file transfer operation is critical for the sensor web system since this will be used to locate the presence of an intruder. In order to achieve the file transfer operation, a Unix command "scp", which is embedded in the Java code is used. This operation takes place after the use of "ssh-keygen" method, which uses private and public keys to enable the stargate and the local computer to recognize each other as trusted hosts.

Once the data file is secured at the stargate, it runs continuously to detect the presence or absence of an intruder. To do this, a bash script is run on the stargate to check for the binary value written in the data file. If it finds a value of "1", it means an intruder is detected and the stargate triggers the web camera to take a picture of the intruder. This picture is then transferred to the apache web-server. The picture is embedded in HTML file which can be accessed by all the client machines. If the bash script reads a "0" from the data file, no immediate action is taken due to absence of the intruder. The webcam can be configured to take pictures at various resolutions. For example, a picture can be taken with 640 x 480 settings to have a high resolution image of the intruder. However, the system usually is set at a lower resolution to reduce system power consumption and also bandwidth required to access the pictures from the website.

Using apache web-server enables users to monitor the system from a remote location. If a user is connected to the internet, he/she would be able to access the apache web-server on the stargate to retrieve the intruder pictures from anywhere.

Even though the stargate is a small low-power computer, it has some high-end capabilities such as the ability to act as an HTTP apache web-server and run Java applications. When the stargate takes the picture of the intruder, it automatically transfers a copy of the picture to the apache server such that it can be accessed by users at remote locations. A time delay for 2 seconds is included in the stargate system such that it can synchronize with the local computer. A webpage is designed to hold the pictures from the intruders and it refreshes every second to reload the page. This ensures that no target or intruder is missed.

4.1 Raw Data

Here is an example of raw data that is collected during the testing period:

```
temp[0][0]=885
FF FF 00 7D 1D 84010100 C701 F3017503 DF
014A 00170019031603000000000000000000
temp[0][1]=841
Light_changes[0]:31.11269837220809
intruder*************!!!
FF FF 007D 1D 84010100 C701 F3014903 EC
014A 00170019031703000000000000000000
temp[0][0]=830
FF FF 007D 1D 84010100 C701 F3013E 03 EB
014A 00170019031603000000000000000000
temp[0][1]=834
Light_changes[0]:2.8284271247461903
No intruder!
FF FF 00 7D 1D 84010100 C701 F4014203
F3014A 00170019031703000000000000000000
temp[0][0]=838
FF FF 007D 1D 84010100 C701 F4014603 F2014A
00170019031603000000000000000000
temp[0][1]=806
Light_changes[0]:22.627416997969522
intruder*************!!!
```

The Listen.java is programmed to collect every two packets of the sensor data and calculate the light_changes parameter. Based on those calculated values, a message of intruder or no intruder is displayed for the users. The light value, such as temp[0][1]=806, is already converted into decimal data. The first and second numbers in the array indicate the sensor node and the packet number respectively. For example, temp[0][1]=806 indicates that light value is 806 from node ID 0 and this is the second packet.

4.2 HTTP Webpage

A simple webpage was created to show an image of the intruder. The HTTP address is the IP address of the stargate. As we discussed earlier, this webserver will be refreshed every second to ensure the pictures are updated. If there is no intruder, this browser does not display any image.

In conclusion, the time-line of operation of the sensor web system can be summarized as follows:

- Activate sensor web system by starting the sensor data collection by running the Listen.java code in the local machine–verify the sensors are returning data through the serial connection
- Activate the stargate bash script which checks for the binary code in the data file
- Intruder enters the environment, light threshold value is crossed, sensors indicate presence of intruder. Local machine now sends a data-file with "1" to the stargate through a secure link.
- Stargate bash script, webcam.sh, finds that there is a "1" in the data-file and triggers the camera to take a picture of the intruder. It waits 2 seconds and checks for data-file again for the binary code.
- Apache server in the stargate stores the picture in a jpeg format and waits for a client webpage to access the server. It also refreshes the webpage every second. When a user at remote location opens up a webpage, the apache webserver accepts the connection and sends the picture to the client machine through HTTP protocol. This client machine then displays the picture of the intruder.

5 Conclusions

In this paper, we implement an intrusion detection system in sensor networks using small size, low cost, low power Berkeley motes. An intruder can be successfully detected when crossing the monitored environment. Once the system detects an intruder, the webcam automatically takes pictures which, in turn, will be transferred to our stargate server.

References

[1] I. Onat and A. Miri, "An intrusion detection system for wireless sensor networks," in *Proceedings of IEEE International Conference on Wireless and Mobile Computing, Networking and Communications*, vol. 3, August 2005, pp. 253–259.

[2] R. Roman, J. Zhou, and J. Lopez, "Applying intrusion detection systems to wireless sensor networks," in *Proceedings of 3rd Consumer Communications and Networking Conference CCNC'06)*, vol. 1, January 2006, pp. 640–644.

[3] J. Blumenthal, M. Handy, F. Golatowski, M. Haase, and D. Timmermann, "Wireless sensor networks - new challenges in software engineering," in *Proceedings of IEEE Emerging Technologies and Factory Automation (ETFA) Conference*, vol. 1, September 2003, pp. 551–556.

[4] A. da Silva, M. Martins, B. Rocha, A. Loureiro, L. Ruiz, and H. Wong, "Decentralized intrusion detection in wireless sensor networks," in *Proceedings of the 1st ACM International Workshop on Quality of Service and Security in Wireless and Mobile Networks*, October 2005, pp. 16–23.

[5] P. Inverardi, L. Mostarda, and A. Navarra, "Distributed IDSs for enhancing security in mobile wireless sensor networks," in *Proceedings of 20th International Conference on Advanced Information Networking and Applications (AINA'06)*, vol. 2, April 2006, pp. 116–120.

[6] E. Lubrin, E. Lawrence, and K. Navarro, "Wireless remote healthcare monitoring with motes," in *Proceedings of International Conference on Mobile Business (ICMB)*, July 2005, pp. 235–241.

[7] T. Hamrita, N. Kaluskar, and K. Wolfe, "Advances in smart sensor technology," in *Proceedings of Industry Applications Conference*, vol. 3, October 2005, pp. 2059–2062.

[8] S. Oh and S. Sastry, "Tracking on graph," in *Proceedings of the Fourth International Symposium on Information Processing in Sensor Networks (IPSN)*, April 2005, pp. 195–202.

[9] I. Demirkol, F. Alagoz, H. Delic, and C. Ersoy, "Wireless sensor networks for intrusion detection: packet traffic modeling," *IEEE Communications Letters*, vol. 10, no. 1, pp. 22-24, January 2006.

[10] "Crossbow Technology Inc." http://www.xbow.com.

[11] "Crossbow technology manual: Stargate developer's guide, processor board (spb400cb) and daughter card (sdc400ca)," www.xbow.com, Feburary 2004.

[12] "Cygnus Solutions Inc." http://www.cygnus-solutions.com/.