1-1-2012

On the Effectiveness of Intrusions into ZigBee-based Wireless Sensor Networks

Michael Johnstone

Jeremy Jarvis
Edith Cowan University

Johnstone, M. N., & Jarvis, J. A. (2012). On the Effectiveness of Intrusions into ZigBee-based Wireless Sensor Networks. Journal of Network Forensics, 4(1), 27-39.

ON THE EFFECTIVENESS OF INTRUSIONS INTO ZIGBEE-BASED WIRELESS SENSOR NETWORKS

Michael N. Johnstone
Jererny A. Jarvis
School of Computer and Security Science
Edith Cowan University
Perth, Western Australia
Email: m.johnstone@ecu.edu.au
Email: j.jarvis@our.ecu.edu.au

Abstract

Wireless Sensor Networks are becoming popular as a means of collecting data by military organisations, public utilities, motor vehicle manufacturers and security firms. Unfortunately, the devices on such networks are often insecure by default, which creates problems in terms of the confidentiality and integrity of data transmitted across such networks. This paper discusses attacks that were successful on a simple network consisting of nodes using the ZigBee protocol stack and proposes defences to thwart these attacks, thus leading to increased user confidence in the ability of organisations to provide secure and effective services. The outcomes were that it was possible to add false nodes to a test network and have these nodes accepted by the network. This was because the packet encryption available for the devices was disabled by default.

Keywords

Wireless Sensor Network, Vulnerability, ZigBee, 802.15.4 Standard.

INTRODUCTION

The purpose of this paper is to test threats by initiating attacks on vulnerabilities in a wireless sensor network (WSN) system, specifically ChipCon CC24XX series sensors produced by Texas Instruments (TI). WSNs are of interest here because they have a wide range of applications from tracking wildlife to in-combat military uses (as described by Römer and Mattern, 2004) and many of these applications require secure communications between the network nodes.

The prime tenets of information security are confidentiality, integrity and availability. Confidentiality means that only authorised users have correct access to assets (assets in this case means the data transmitted over a wireless network). Integrity can be described as ensuring that only authorised users can alter data in defined ways. Availability is a guarantee that authorised users are able to access data in a timely fashion.

There are well-known classes of attack for each of these tenets. For example, brute force code-breaking could be an attack on confidentiality; the classic “man in the middle” attack is an attack on integrity; and denial-of-service is an attack on availability. This paper tests the first two classes of attack by experiment. Attacks based on denial-of-service are not covered here because they can usually be detected more easily than the other classes of attack. Further, depending on the devices attached to the network nodes, denial-of-service attacks may be less problematic. For example, a denial-of-service attack on a wireless car tyre sensor may result in the sensor being rendered inoperative, which is perhaps less of an issue than a false reading that causes a driver to respond incorrectly, which could potentially cause an accident.

The genesis of this work was not initially a desire to evaluate the security of WSNs. A TI CC243X Zigbee development kit was used to establish location information for specific nodes on a test WSN. This information (with accompanying direction data) was intended to be used for locating people within a room as part of a ‘smart-room’ design project, the rationale being that relatively small amounts of this information transmitted across a WSN could effectively and efficiently convey almost the same semantics as large volumes of video stream data captured via multiple cameras, provided that all that was required was the location of a person within an environment, coupled with the direction (compass heading) that the person was facing.

Several locator tests were conducted with a network consisting of between 2-10 nodes. The results were inconclusive and outside the range of experimental error (± 0.25m for the TI devices according to their specification). At this point, assuming the devices were not faulty (to test this assumption, a second TI development kit set up identically in the same environment showed similar results), it was speculated that the

nodes were either suffering from interference due to the physical aspects of the room in which the tests were being conducted or the network was possibly suffering from an intrusion.

This paper introduces wireless sensor networks as a means to collect information. Next, the basic principle of the operation of the TI devices with respect to location sensing is explained. Subsequently the experimental methods and tools used to craft the attacks are described. Following this is an analysis of the results. The paper then concludes by providing a mitigation strategy for those specific attacks that were successful.

WIRELESS SENSOR NETWORKS

Wireless networks are different from their wired counterparts in that wireless systems often have dynamic topologies, are unprotected from other signals sharing the medium and communicate over a medium that is significantly less reliable than wired networks (IEEE, 2007). A wireless sensor node consists of a microprocessor, a radio frequency transmitter/receiver, a power supply (a battery or sometimes a solar cell) and a sensor of some type.

Sensors can be microphones, still cameras, video cameras, pressure, temperature, chemical, biological, radiological or movement (passive infrared) sensors. Such sensors could also be coupled together in that a movement sensor might trigger a dormant camera. This would have benefits in terms of power consumption as the lowest power device (the movement sensor) is on continuously, but the high power consumption/high bandwidth device is only activated when there is something of interest to detect.

Two important properties of a wireless network are the data transfer rate and the maximum distance between transmit/receive nodes. IEEE 802.11n has a transfer rate of approximately 248Mbps and a range of 250m under ideal conditions. This is perhaps no surprise as 802.11n was designed to address the speed limitations of prior 802.11 standards. By comparison, 802.15.4 has a transfer rate of between 40-250Kbps and a range of 75m, again under ideal conditions. It would appear that 802.15.4 is at a significant disadvantage compared to 802.11.x, but this is not necessarily the case as the shorter range requires less power.

Following the development of the 802.15.4 standard, several bodies such as the ZigBee Alliance, were formed to promote the development of low-power networks in various application domains. The ZigBee standard is not the same as 802.15.4 (with which it is often confused), but is a protocol stack built on top of the IEEE standard (see figure 1). Strictly speaking, ZigBee now uses the 802.14.5 standard, but the distinction is not significant for this discussion. ZigBee-based devices offer a potential solution where many sensors will be deployed in sub-optimal conditions, however they suffer from two drawbacks. First, power consumption is an issue as the devices are often small and powered by batteries and second, security is a potential issue.

Most wireless sensors operating in idle mode consume power at a rate approximately equal to the power consumed in receive mode due to increased communication with a base station therefore power consumption is a significant issue for these low-power devices.

Shutting down a sensor may optimise power use compared to leaving it in idle mode when it is not transmitting or receiving. Unfortunately, once a sensor is turned off, it cannot receive any messages from its neighbour sensors (because it is effectively disconnected from the network). Further, continually turning the transceiver on and off also has an energy overhead. It might be more efficient for a sensor to be able to adjust its sampling rate and communications frequency to save power.

The 802.15.4 standard handles security at the MAC layer (see figure 1), however this security must be enabled and is disabled by default (at least on the TI devices). Figure 2 shows that two bytes are reserved for a CRC (Cyclic Redundancy Check) which is computed on the data bytes. This suggests that the datagrams are safe from tampering because the CRC would not match when recomputed at the endpoint. This is not necessarily so. Sastry and Wagner (2004) point out that 802.15.4 networks can suffer from security breaches in three areas, viz: initialisation vector management, key management and integrity protection. Regarding the latter, Sastry and Wagner claim that CRCs are insufficient for integrity protection and favour strong encryption instead (which is available on 802.15.4 devices). The main issue, according to Sastry and Wagner (2004) is that an attacker can modify the CRC such that the receiver accepts the datagram, which clearly negates the purpose of a CRC (integrity checking, rather than guaranteeing secure communications). This is still a problem as noted by Aggélou (2009) and Boudhir et al. (2010).

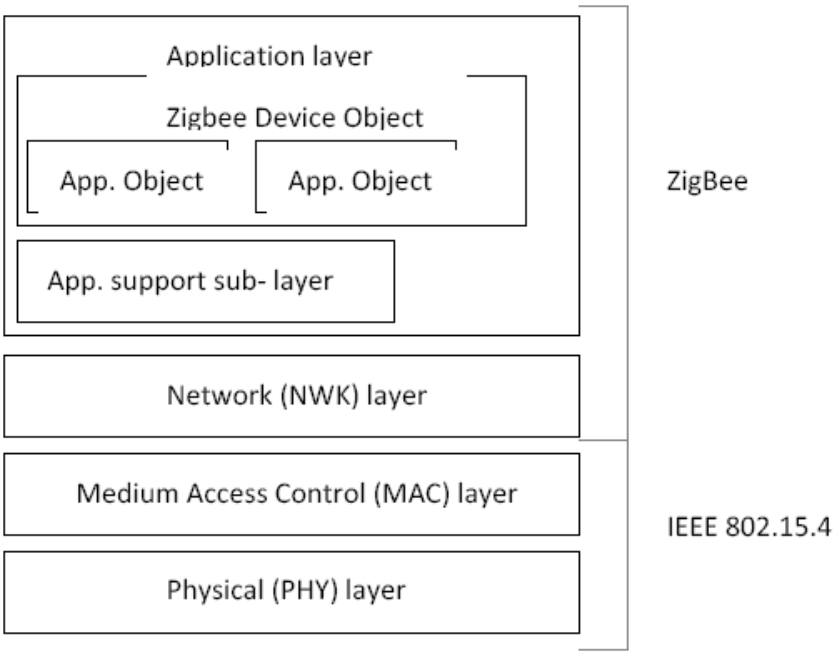

Figure 1: The Relationship between 802.15.4 and ZigBee (adapted from Akyildiz and Vuran, 2010).

Wireless sensor networks appear to show some promise but there exist security problems to be solved in terms of power consumption of the network nodes (availability) and the integrity and potentially confidentiality of the data sent across the network. The next section describes the attacks crafted to expose the vulnerabilities of a specific wireless sensor network.

1 byte	2 bytes	1 byte	0-10 bytes	0-10 bytes	variable	2 bytes
Len.	Flags	Seq. #	Dest. Address	Source Address	Data	CRC

Figure 2: The Structure of an IEEE 802.15.4 Datagram (Adapted from Sastry and Wagner, 2004).

TI IMPLEMENTATION OF ZIGBEE NODES

The ZigBee (2007) standard specifies three types of devices, viz. coordinators, routers and end devices. A coordinator starts a network sets the Personal Area Network ID and channel on which the network will operate. A router can provide services to devices to join a network, multi-hop routing and communication between end devices. An end device is neither a coordinator nor a router and has no responsibility for maintaining the network.

In the TI location engine implementation of a ZigBee protocol network, nodes are one of:

- A Location Dongle. This device acts as the coordinator and is attached to a computer running the Z-Location software provided by TI. The software displays the location of the other network nodes;
- A Reference Node: These devices are set up in known physical positions are serve to provide location data to other nodes on request; or
- A Blind Node: These devices move within the physical space of the reference nodes' layout and calculate a location estimate in real time.

As the coordinator, the location dongle starts the network and sets the operating channel. Reference nodes join the network via the location dongle. Reference nodes provide the X, Y coordinates to a blind node for its

location estimate. Also, when a blind node calculates its location the measurement is sent to the location dongle thus the blind node's position can be displayed on-screen via suitable software.

The blind node uses a combination of X, Y coordinates and a Received Signal Strength Indicator (RSSI) obtained from reference nodes within range. According to TI (Aamodt, 2006) RSSI is calculated by each reference node in range by averaging a series of packets sent by the blind node over a certain time period. Each reference node then sends a response to the blind node that contains both the RSSI and the reference node's X, Y co-ordinates. Aamodt states that RSSI is calculated by:

$$RSSI = -(10n \log_{10} d + A)$$

where

n is a signal propagation constant.

d is the distance from the sender.

A represents the received signal strength at a distance of one metre.

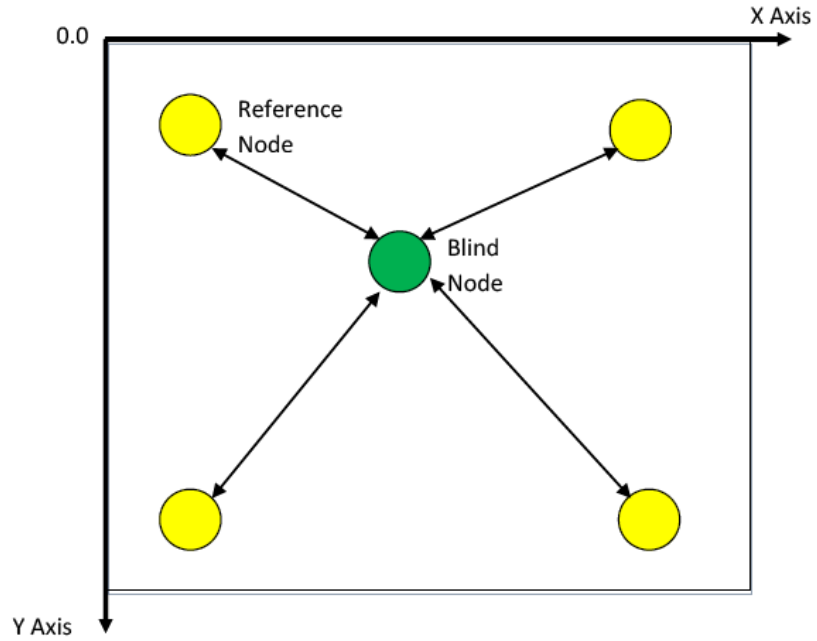

Figure 3: Co-ordinate System for Location Estimation (adapted from Aamodt, 2006).

ATTACK PREPARATION

The purpose of this research is to assess what risks are inherent in the location calculation system designed by TI which is implemented by the CC2400 series chipsets. In this section, the materials used are described, the TI nomenclature explained and the attack schemes explained.

Hardware and Software Used

The hardware used in the attacks was: a laptop (Windows 7); a SmartRF04EB Evaluation Board with a CC2430 chip flashed with the location dongle (see below) hex file; and a CC2430DB used as a capture device.

Software required was the TI SmartRF Protocol Packet Sniffer; TI SmartRF Studio 7 v1.2.3 and the SmartRF Flash Programmer, (used to flash the devices memory with the appropriate firmware).

Wireless Network Nodes

Recall that the two dimensional location system designed by TI utilises the ZigBee protocol over the 802.15.4 standard and is designed to transmit small amounts of data over a mesh-like network of nodes denoted as one of:

- A reference node: A node that is given an X, Y position and is utilised by a blind node when it (the blind node) wishes to perform a location calculation;

- A Blind Node: A node that moves around in X,Y space and performs location calculations; or
- A Location Dongle: Connected to a computer (a laptop in this case) and is used to receive location data from each node (displayed on-screen).

Specific Attacks

Initial tests will be performed to obtain packet data using various sniffing software and hardware configurations. The process will help define the later attack patterns and give initial insight into what can be obtained.

Next, the system will be attacked by supplying fraudulent packet data by introducing a false reference node and then a false blind node into the system. This will be designed as a replay attack and the success will be determined by monitoring the effect it has while the system is collecting location data.

RESULTS AND ANALYSIS

Test#1: Use TI provided hardware and software to trial basic packet-sniffing.

The objective of the test was to obtain packet data using the CC2430DB as a receiver, and the CC2431 board as a transmitter. The test used the TI packet sniffing software and SmartRF Studio 7's packet transmitter with a SmartRF04EB Evaluation Board with a CC2430 chip flashed with the location dongle hex file and a CC2430DB as a capture device.

As figure 4 shows, the TI-provided packet sniffer had no problem picking up datagrams (packets) sent across the wireless network.

File Help

ZigBee 2007/PR

P.nbr.	Time (us)	Length	Frame control field	Sequence number	Dest. PAN	Dest. Address	Source PAN	Source Address	Encrypted MAC payload	LQI	FCS
			Type Sec Pnd Ack.req PAN_compr								
RX 1	+0 =0	22	DATA 1 0 0 0	0x41	0xFD1B	0x4FB8	0xF668	0x7B72	14 99 CD D3 0D F0 44 3A B4	220	OK
RX 2	+101987 =101987	22	DATA 1 0 0 0	0x41	0xFD1B	0x4FB8	0xF668	0x7B72	14 99 CD D3 0D F0 44 3A B4	220	OK
RX 3	+103147 =205134	22	DATA 1 0 0 0	0x41	0xFD1B	0x4FB8	0xF668	0x7B72	14 99 CD D3 0D F0 44 3A B4	220	OK
RX 4	+102869 =308003	22	DATA 1 0 0 0	0x41	0xFD1B	0x4FB8	0xF668	0x7B72	14 99 CD D3 0D F0 44 3A B4	220	OK
RX 5	+102009 =410012	22	DATA 1 0 0 0	0x41	0xFD1B	0x4FB8	0xF668	0x7B72	14 99 CD D3 0D F0 44 3A B4	220	OK
RX 6	+101947 =511959	22	DATA 1 0 0 0	0x41	0xFD1B	0x4FB8	0xF668	0x7B72	14 99 CD D3 0D F0 44 3A B4	228	OK
RX 7	+102030 =613989	22	DATA 1 0 0 0	0x41	0xFD1B	0x4FB8	0xF668	0x7B72	14 99 CD D3 0D F0 44 3A B4	220	OK
RX	+102000										

Figure 4: TI Packet Sniffer Data Capture Test.

Test#2: Assessing packet loss with the CC2430DB.

The objective of the test was to evaluate the effect of interference from other devices and node spacing on packet-sniffing.

The test used the TI packet sniffing software, the Chipcon Z-Location Engine and the Texas Instruments Flash Programmer (used for initial flashing of devices). The hardware used was a SmartRF04EB evaluation board, several boards with CC2430 chips flashed with the reference node hex file, a board with the CC2431 chip flashed with the blind node hex file and a CC2430DB as a capture device.

The packet sniffer was used to gauge the effect of interference from other devices by conducting a series of tests inside a building (with the nodes at one, two and four metre spacing), then performing the same tests outdoors. The tests highlighted several unexpected problems, viz: the nodes would move around (on-screen) when they should have been stationary and there was some packet loss. These faults could be attributed to interference or the distances between the nodes being sub-optimal for effective transmission. As the tests progressed, a single variable (distance or site) was changed with the other held constant to attempt to identify the problem. These changes did not affect the output enough to be considered successful.

The test did, however, prove useful in that the data captured outlined the process that the system uses to perform a location calculation. The pattern that developed was:

1. The sniffer receives a few packets, then a return message from the blind node when requested to send position data with a cluster ID of 0x0014. This packet has a larger network payload than the 13 bytes expected from the blind node as a location calculation so it not a transfer of data from a reference node. This is a return message from the blind node when requested to send position data.
2. Next there are a few similar packets, then a blast message (cluster ID 0x0019) from the blind node (a packet that is transmitted to any listening devices). This is a small radius message and come before a location calculation request. It is a small range packet designed to limit the number of reference nodes that are recipients
3. Next a single packet (ID 0x0011) which is a reference node's position request. The packet is quite small and its destination address is 0xFFFF which suggests the blind node is selecting reference nodes automatically
4. There are two single packets (ID 0x0011) sent identified as an XY-RSSI Request, used to request an XY calculation response from the reference node
5. Next, an XY-RSSI response (ID 0x0012), which broadcasts the blind node's X position, its Y position and its RSSI average value
6. Finally there are some smaller packets sent; a small response message and a return message.

This information could be used by an attacker to spoof the system with packet data similar to what is expected. It might also be possible to wait and read a blast from the blind node and inject packets at that point.

Test#3: Adding False Nodes.

The objective of the test was to add a false node of each type and observe the system's behaviour towards the false node(s). This was done for each node type by:

a) Reference Node: Setting the RSSI value to the ideal signal strength and setting different XY coordinates on the GUI;

b) Blind Node: Setting a second blind node and monitoring the effect; and

c) Dongle: Setting a new dongle node both before the system is started and while it is operating.

This test (and later tests) used the same hardware/software configuration as test#2.

The result of test 3a was that the fake node showed up on the system as an existing node. It was initialised well outside of the system boundary for ease of detection on-screen (using the TI-supplied GUI). When an authentic node was moved near the fake node it didn't change the location of the blind node. This was expected behaviour as the CC2430DB doesn't have the capability to perform a location calculation. When the RSSI value is changed to 110 (the ideal RSSI response from a node) the blind node location didn't change.

The blind node signals the CC2430DB to perform a location calculation but when it doesn't receive a signal it simply uses the next node for its calculation. It is assumed the fake node, as it can't send a proper calculation, will send a response of "0xFFFF" for the calculation and the node will then use the other nodes for the calculation. The authentic node's position became less accurate once the fake node was in use due to it (the authentic node) having to use a node which was further away.

The result of test 3b was that the fake node showed up in the system as a new blind node but it wouldn't pick up any data. As it shows in the system as a blind node, it could possibly be populated with false location data.

The result of test 3c was that, as expected, without an application to process the nodes it picks up there isn't any point in obtaining the information.

It would be useful to expand on the code used in the reference node attack. If the code can be modified so that it sends a hard-coded response to the blind node the attack will be able to bypass the need to perform a location calculation. If this can be achieved the credibility of a location calculation would be compromised. The blind node replacement could also be expanded, if the short address of the new blind node matches one of an existing node, the system won't be able to determine from which node it should accept data.

Test#4: Improving Adding False Nodes.

The objective of this test was twofold, first to modify the reference node to provide a location response and second, to modify the blind node so that the response sent is hard-coded, thus whenever a calculation is necessary the node will send the predefined (potentially false) location.

Recall that in the previous test, the reference node was sending out data correctly and interacting with the blind node. It was displayed in the GUI as a new node but wasn't assigned a position because it can't perform a location calculation (therefore it would just give the blind node error packets).

Figure 5 shows that the system accepts the fake nodes. The test proved that the fake reference node can be convincing if it doesn't have to interact with the blind node. This issue could be addressed by examining the packet data and crafting an appropriate header/payload.

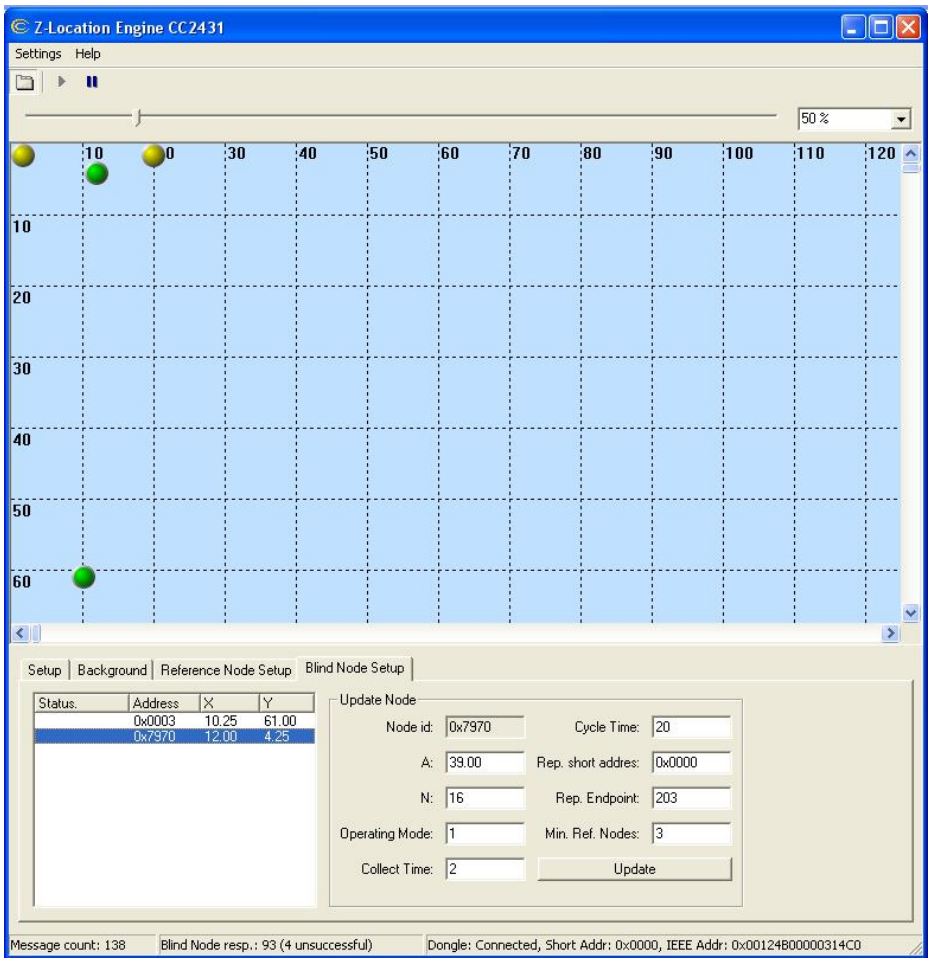

Figure 5: GUI Display of Two Authentic and Two Fake Nodes.

Analysis

Potok et al., (2003, p13) after Abadi (1999) state that it is practically impossible to construct a truly secure information system (or in this context, a wireless sensor network). Communications are secure if transmitted messages can be neither affected nor understood by an attacker. Iinformation operations are secure if information cannot be damaged, destroyed, or acquired by an attacker or if the communication no longer has any validity or value as information in part or aggregate.

The experiments conducted show that the CC24XX wireless sensor devices are vulnerable to attack. To ensure safe and secure operation of a network, Texas Instruments recommend the following strategies:

- The use of encryption;
- More secure keys;
- A trust centre in which nodes have to be passed by the dongle before they can join the network; and

- Implementing key updates at predetermined intervals.

In fact, the original network keys utilise 128 bit AES encryption, which should have been robust, however, we detected some weaknesses with this implementation:

- The original design does not use the keys;
- The default key provided by the system is widely available;
- Keys are shared over the entire network and stored in flash memory, so if an attacker can obtain the flash data, s/he can obtain the key; and
- There is no documentation that outlines the need to change this key before the system is implemented.

It was remarkable that the plain-text security key for the devices was located in a text format file. The code is available online and if the code is not changed when the system is set up the security system is accessible to any attacker who looks for the network key.

The obvious solution is to enable encryption and use generated (non-default) keys. Following this, keys must somehow be distributed across the network, which leads to another potential vulnerability. Keys could be hard-coded onto the devices or transmitted over-the-air. Hard-coding is attractive as there is a guarantee of integrity implied in such a process. This means that there is little chance that the key can be compromised, even though all devices on the same network share the same key. Whether this is an appropriate solution depends on the application and the level of security required. For small-sized networks where the devices are available to be re-flashed physically, even if the key were to become compromised, key management is a straightforward task. For larger networks and in situations where the devices are positioned in inaccessible or unsafe locations (for example, military applications), this method is not feasible.

Over-the-air transmission of keys appears to have an advantage over hard-coding because it doesn't require physical access to situated network nodes. Such a method requires careful management as a clever attacker may join a network as an unauthenticated node for the specific purpose of forcing a key re-broadcast.

A trust centre is another TI recommendation for establishing and maintaining confidentiality of data and network integrity. A trust centre can manage a master key for establishing which devices can join a network. Yüksel, Nielson and Nielson (2008) point out that ZigBee devices use link keys between nodes, network keys across a whole network and a master key between pairs of nodes (used to establish a key). Again, as noted by Boyle and Newe (2007), the master key must be inserted out of the network to provide no opportunity for compromise. A recent improvement in the ZigBee protocol, noted by Yüksel et al., is the notion of a security procedure that recognises the difference between a node that has joined the network but is as-yet unauthenticated and one that is joined and authenticated. This multi-state model allows only authenticated nodes to request a key update, so an attacker is unable to force a re-broadcast as an attack node is not authenticated to the network.

A private or public key infrastructure is an obvious solution to the integrity problem, however issues of secure storage for the keys and over-the-air transmission of keys must still be addressed, otherwise the strength (size) of the key does not matter. The issue of key management is perhaps further complicated by the ever-decreasing cost of the hardware required to conduct a brute-force attack. Given that a 128 bit AES key has a search space of 3.4×10^{38} possible keys, this is not yet a problem.

CONCLUSIONS AND FURTHER WORK

This study sought to explore the vulnerabilities claimed to exist in wireless sensor networks. The basics of wireless sensor networks were explained and the potential vulnerabilities articulated. A series of attacks were planned and executed on a simple network. A solution was proposed for the wireless sensors (strong cryptography), but this has yet to be proven as effective as there are issues of key management still to be solved.

The purpose of network forensics is to capture and analyse network traffic with the objective of identifying and potentially replaying malicious exchanges in order to assess any damage and provide a sound basis for evidentiary requirements. The forensic implications of this research into ZigBee communications are clear. The datagrams exchanged by nodes on a WSN of this type can be intercepted, interpreted and modified. This means that the contents of the datagrams cannot be trusted. The repercussions for the acceptance and use of these devices are serious.

Reflecting on a relatively simple use, home automation, a WSN can provide control of mains power outlets or a security system in a home. If the network can be compromised, it appears to be a low-risk, low impact problem. This is not so as one of the major benefits (the ability to control devices remotely) becomes a significant

liability. The ability to take control of a mains power outlet may become a high impact risk if the mains outlet controls a life-critical system such as a home oxygen concentrator. Similarly, assuming control of a security system would grant an attacker physical access to a home that would normally be blocked.

Due to their low cost and robustness, networks of wireless sensors have many potential uses ranging from the mundane such as tagging goods, weather reporting and home automation to more innovative uses of this technology such as home monitoring of individuals in aged care environments and automated meter reading for public/private utilities. Considering just the last application, wireless meter reading is obviously more efficient than traditional physical meter reading and thus would be an attractive solution because of its lower recurrent costs. Whilst this research has shown that these devices are vulnerable, perhaps the security fears of being able to misrepresent a meter reading are not significant for domestic consumption as it would be expected that the utilities concerned would be immediately aware of any 'outlier' readings. The issue is a larger one of public confidence in such automated systems, especially given the pervasiveness of software systems.

The next step in this research programme is to evaluate the security of later versions of the chipset, specifically the CC25XX series of sensors by forcing over-the-air key updates and seeing if the security of the keys can be compromised by a fake network node. Research will also be conducted into other families of wireless sensor currently used for home automation, security systems, motor vehicle electronic control units and industrial process control devices. The investigation of these areas is of paramount importance because of issues of public safety and the need to ensure security and resilience in critical infrastructure systems.

REFERENCES

Aamodt, K. (2006). *Application Note AN042-CC2431 Location Engine*. Dallas, TX: Texas Instruments.

Aggélou, G. (2009). *Wireless Mesh Networks*. New York, NY: McGraw-Hill.

Akyildiz, I.F. and Vuran, M. C. (2010). *Wireless Sensor Networks*. Chichester: John Wiley.

Boudhir, A.A., Bouhorma, M., and ben Ahmed, M. (2010). Multi-Agents Platform for Security in Wireless Sensor Networks. *IJCSNS International Journal of Computer Science and Network Security*, 10(10), October, pp. 198-201.

Boyle, D. and Newe, T. (2007). Security Protocols for use with Wireless Sensor Networks: A Survey of Security Architectures. *Proceedings of the Third International Conference on Wireless and Mobile Communications (ICWMC'07)*.

IEEE (2007). IEEE Standard for Information technology—Telecommunications and information exchange between systems—Local and metropolitan area networks—Specific requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. New York: IEEE Computer Society.

Potok, T., Phillips, L., Pollock, R., Loebl, A. and Sheldon, F. (2003). Suitability of Agent-Based Systems for Command and Control in Fault-tolerant, Safety-Critical Responsive Decision Networks. *Proc. 16th Int'l Conf. On Parallel and Distributed Computing Systems*, Aug. 13-15,2003 Reno NV.

Römer, K and Mattern, F. (2004). The Design Space of Wireless Sensor Networks, *IEEE Wireless Communications*, 11(6), pp. 54-61.

Sastry, N. and Wagner, D. (2004). Security Considerations for IEEE 802.15.4 Networks. *ACM Workshop on Wireless Security (WISE 04)*, pp. 32–42.

Yüksel, E., Nielson, H.R. and Nielson, F. (2008). ZigBee-2007 Security Essentials. In: *Proceedings of the 13rd Nordic Workshop on Secure IT-systems (NordSec 2008)*, pp. 65-82, Copenhagen, Denmark.