Requirements for Wireless Sensor Networks in Order to Achieve Digital Forensic Readiness

F. Mouton and H.S. Venter

Information and Computer Security Architecture, University of Pretoria
e-mail: moutonf@gmail.com

Abstract

The field of wireless sensor networking is a new and upcoming one and unfortunately still lacking as far as digital forensics is concerned. All communications between different nodes (also known as *motes*) are sent out in a broadcast fashion. These broadcasts make it quite difficult to capture data packets forensically whilst retaining their integrity and authenticity. This paper examines the differences between IEEE 802.15.4 wireless sensor networks and IEEE 802.11x wireless networks when it comes to implementing digital forensic readiness within the network environment. It focuses on the differences in the communication protocol, proof of authenticity and integrity, time stamping, modification of the network after deployment and other differences between IEEE 802.15.4 wireless sensor networks and IEEE 802.11x wireless networks. Each of these elements is discussed, after which a table is provided that shows the specific requirements to be taken into account when proposing digital forensic readiness in a wireless sensor network environment.

Keywords

Forensic readiness, digital forensics, wireless sensor networks

1. Introduction

Our pursuit of a better lifestyle has led to a vast improvement in the technology to which we have access in today’s world. The concept of a wireless sensor network (WSN) is just another technology developed to improve our ability to better accomplish our daily tasks. The implementation of security protocols on WSNs has not received much attention to date, and, even more so, very little consideration has been given to digital forensics within a WSN environment.

The problem is that currently there is no formal set of requirements for achieving digital forensic readiness in wireless sensor networks. The purpose of this paper is to determine how IEEE 802.15.4 wireless sensor networks differ from IEEE 802.11x wireless networks when it comes to implementing digital forensic readiness.

The remainder of the paper is structured as follows: The second section provides some background information about WSNs and digital forensic readiness. Section 3 discusses the differences between IEEE 802.11x wireless networks and IEEE 802.15.4 wireless sensor networks with regard to digital forensic readiness. Section 4 proposes a set of requirements that need to be adhered to when implementing digital forensic readiness for wireless sensor networks. Finally, a summary is provided of the forensic readiness requirements that are proposed and of future work to be done in this field.

2. Background

Wireless sensor networks still constitute a relatively new area of research in computer science and the first papers on WSNs were only published in the last decade (Chong & Kumar, 2003; Mouton & Venter, 2009). Much of the research on WSNs has been dedicated to new areas of application aimed at supporting our modern lifestyle. Some background information for a better understanding of WSNs is provided next, before strategies for the achievement of digital forensic readiness for WSNs are suggested.

2.1. Wireless Sensor Networks

WSNs belong to the general family of sensor networks that use multiple distributed sensors to retrieve data from various environments of interest. Chong and Kumar (2003) provide a history of previous accomplishments of WSNs and show how they have evolved in terms of sensing, communication and computing. WSNs consist of wireless nodes with embedded processors and ad hoc networks (Estrin et al., 2001), and involve wireless communication (Ye, Heidemann & Estrin, 2002). Mouton and Venter (2009) define a WSN as an ad hoc network that consists of tiny and resilient computing nodes known as *motes* or sensors. These *motes* are extremely efficient with regard to power consumption and can collaborate effectively with other *motes* in their vicinity. A graphical representation of a wireless sensor network is provided in Figure 1, while in Table 1 the functions of each of the components are subsequently summarised briefly (Mouton & Venter, 2009; Heinzelman, Kulik & Balakrishnan, 1999; Sohrabi et al., 2000).

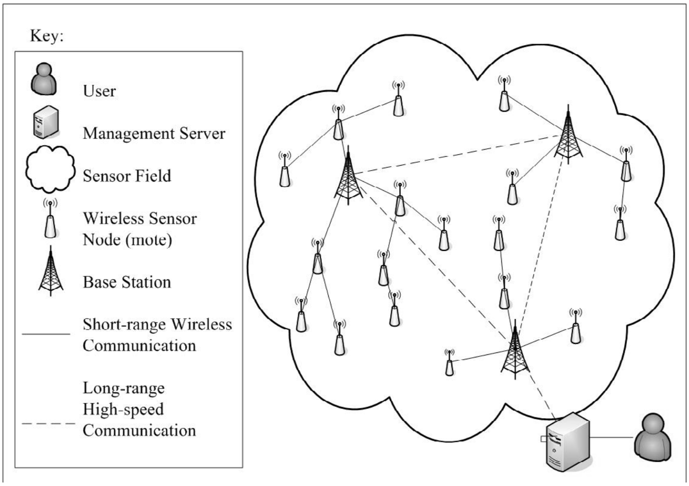
Key:
User
Management Server
Sensor Field
Wireless Sensor Node (mote)
Base Station
Short-range Wireless Communication
Long-range High-speed Communication

Figure 1: A graphical representation of a wireless sensor network (Mouton & Venter, 2009).

WSN component	Functions of each component
User	The user can interact with the WSN through the management server.
Management Server	The management server serves as an interface console for the WSN.
Sensor Field	The sensor field denotes the physical boundaries of the WSN.
Wireless Sensor Node (*mote*)	Each *mote* contains a small subset of the various sensors. *Motes* in the network can also act as repeaters for packets that need to reach the base station.
Base Station	A base station serves as a gateway node through which the information of the *motes* has to travel to reach the management server.
Short-range Wireless Communication	Short-range wireless communication links are established between neighbouring *motes* and the neighbouring base stations.
Long-range High-speed Communication	Long-range high-speed communication links are established between further-ranged base stations and the management server.

Table 1: Brief summary of functions of the components of a wireless sensor network (Mouton & Venter, 2009).

WSNs can be used in many environments. Their *motes* may consist of many different types of sensors, such as thermal, visual, infrared, radar or acoustic. These motes can monitor a wide variety of ambient conditions, including humidity, pressure, sound, noise levels, temperature, lightning conditions and objects moving through a designated area (Elson & Estrin, 2001; Kahn, Katz & Pister, 1999).

Some applications of WSNs include military applications such as the tracking of moving objects and battlefield surveillance (Zhao, Shin & Reich, 2002); environmental applications such as habitat monitoring, forest fire detection and flood detection (Mainwaring et al., 2002); and health applications such as the tracking and monitoring of doctors and patients in hospitals, as well as of drug administration in hospitals (Lu et al., 2002). Finally, WSNs can also be used for home and building automation applications.

The next subsection focuses on providing the reader with a workable definition of digital forensic readiness in a WSN context.

2.2. Digital Forensic Readiness

To achieve digital forensic readiness in any type of environment, it is essential to first establish an acceptable definition for it. However, since it is a fairly new concept and the subject of divergent opinions, consensus must still be reached in this regard.

In defining digital forensic readiness, Tan (2001) identifies two objectives that have to be balanced carefully: maximising the ability to collect credible digital evidence, and minimising the cost of performing a digital forensic investigation. Tan also argues that several steps need to be taken to ensure that an environment is ready as far as digital forensics is concerned. Rowlingson (2004), on the other hand, suggests ten steps that describe the key activities in implementing a digital forensic readiness programme. Because Rowlingson's steps have actually been designed to create a business process model for digital forensic readiness, this paper gives preference to Tan's objectives for meeting the requirements of digital forensic readiness in a WSN environment.

Even though Tan's two objectives provide a very good definition of digital forensic readiness, it is important to refine them somewhat to make the definition more specific to a WSN environment. For the purpose of this paper, digital forensic readiness is defined as the notion to perform a digital forensic investigation in the shortest amount of time with the least amount of cost and without having to disrupt the original network that has to perform mission-critical tasks. This definition is set as the main goal for achieving digital forensic readiness on WSNs.

The next section discusses the differences between IEEE 802.15.4 wireless sensor networks and IEEE 802.11x wireless networks when it comes to implementing digital forensic readiness.

3. Differences between WSNs and WLANs

WSNs have special needs compared to IEEE 802.11x wireless networks and hence have more specialised requirements than would apply to wireless networks (also known as wireless local area networks or WLANs). There are many important factors that make a WSN unique and distinguish it from a WLAN. The factors that are addressed in this paper are the following:

- Communication protocol
- Proof of authenticity and integrity
- Time stamping
- Modification of the network after deployment
- Protocol data packets
- Radio frequencies
- Power supply
- Network overhead
- Data integrity

The factors listed above are the main ones that differentiate WSN environments from WLAN environments. The reasoning behind the choice of these factors will become apparent in the coming subsections, where each factor is addressed individually. It is, however, important to remember that the core of the argument about the importance of these factors concerns the manner in which they influence the design decision of how to implement a digital forensic readiness application for WSNs.

While examining each of these factors, it is important to note that the authors assume that no modification to the original WSN (hence forward referred to as *oWSN*) is allowed and thus a secondary independent forensic WSN (hence forward referred to as *fWSN*) would be used for the digital forensic readiness implementation of the *oWSN.*

The discussions in each subsection below briefly focus on how these factors differ from WLAN to WSN, and subsequently our focus shifts to how to address them in WSNs.

3.1. Communication Protocol

All communication within a WSN occurs in a broadcast fashion and thus a *mote* never really knows which of its neighbouring *motes* actually receives the packet (Akyildiz et al., 2002; Tseng, Ni & Shih, 2003). The default functioning of a *mote* in the sensor field is to receive all packets – upon receipt of a packet it then has to analyse if the packet was meant for it or not. This analysis requires some processing that drains the battery of the *mote,* which is an important consideration in WSN communication.

The broadcasting technique used in WSNs is very different from the communication techniques used in an IEEE 802.11x wireless network. In the WLAN environment, one can determine if a packet has arrived at its destination by monitoring the network, since acknowledgement packets are sent to confirm the receipt of packets (Xylomenos & Polyzos, 1999; Xylomenos et al., 2001). This is not the case in a WSN environment.

Due to the broadcasting fashion in which WSNs communicate, the *mote* that broadcasts packets will never be completely sure whether the packet was received by the *mote* for which the packet was intended. This uncertainty could be overcome by introducing a communication protocol that allows the receiving *mote* to reply with a receipt acknowledgement packet. However, because this would require extra transmissions that can lead to a greater battery drain, this procedure cannot simply be implemented in all WSNs. The suggested technique also has several other disadvantages. If a flooding attack is launched against the *oWSN,* it would compel the *oWSN* to reply to each flooding attempt with receipt acknowledgement messages, which would then flood the entire *oWSN.*

Considering that a protocol founded on receipt acknowledgement packets can have such a severe impact on a WSN environment, it seems quite impractical to use such a protocol in this environment. Hence the authors have agreed to accept that most WSN *motes* will be uncertain as to whether or not packets have actually arrived at their destination. This causes severe problems in terms of forensic monitoring with a secondary network. It could likely be the case that the packets received by the *oWSN* base station might differ from those received by the *fWSN* base station in the case that some of the packets are dropped in either of the two WSNs. In the case of the *fWSN,* however, this problem could be avoided by implementing a protocol that uses receipt acknowledgement packets, because it is in the nature of a forensic network to always be sure that the information received at either point of the communication line contains some degree of authenticity and integrity. In order to achieve sound digital forensic readiness, it is crucial to prove the authenticity and integrity of the data packets that have been received. The next subsection focuses on defining what the authors see as authenticity and integrity. The differences between maintaining the authenticity and integrity from a WLAN and a WSN perspective are also discussed, as well as possible ways of maintaining authenticity and integrity within a WSN environment.

3.2. Proof of Authenticity and Integrity

Authenticity and integrity first need to be defined as there could be different opinions on precisely what each of them means. In the context of this paper, authenticity is defined as the certainty that the origin and destination of the data packet are kept intact throughout its whole lifetime. The lifetime of a data packet runs from the time that it is sent from the first *mote* up to the time when it is received and processed by the base station. Next, integrity is defined as the certainty that the correctness of the data within the data packet is kept intact throughout the lifetime of the data packet.

Numerous techniques for proving the authenticity and integrity of packets in an IEEE 802.11x wireless network have already been published (Chen, Jiang & Liu, 2005; Komori & Saito, 2004; (Guizani & Raju, 2005). Firewalls, Intrusion Detection Systems, Wireless Routers and Wireless Network Interface Cards are all examples of equipment you would find in an IEEE 802.11x wireless network and most of these devices have the ability to generate a log or some other way of showing which data packets have passed through the network. Most of these abilities are fairly simple techniques that are performed by the device itself. In most cases where a log file is generated, it is safe to assume that the information reflected in the log file is actually the true pattern of traffic that has passed through the device. However, this is only the case when it is certain that the device is not defective or that the log file has not been tampered with. This single log file can also be backed up by looking at all the other devices through which this single packet has travelled, as most devices in an IEEE 802.11x environment should have some form of logging. In a WSN environment, however, very little or no logging is done on the *motes* in the sensor field, due to various reasons. These reasons can include the limited power source and the limited storage space that these devices have. WSN equipment, by default, only does logging at the base station and if logging were to be required at every *mote,* one would have to go and implement this yourself. This obviously raises another issue, namely as to the trustworthiness of the code with which one does the logging. Tried and tested techniques for logging are generally more trustworthy than one's own attempts at implementing logging. It is easier to defend the authenticity and integrity of a well-known logging technique than that of a self-developed technique. In the case where a self-developed technique is used, it must be based on some solid theory as to why it can provide authenticity and integrity. Because WSNs differ so significantly from WLANs, the authors have decided to propose a form of logging that is based on the Casey Certainty Scale (Casey, 2002).

Fortunately, in a WSN environment, multiple *motes* tend to be able to each capture the same data packet simply because they are all in range of a particular broadcasted packet. This is a feature of WSNs, which is not the case in IEEE 802.11x networks. Most devices in WLANs will ignore packets that are not meant for them and do not even attempt to log these packets. The opposite is true for WSNs, where motes attempt to capture every data packet within range. This feature of WSNs can be successfully exploited in an attempt to prove the authenticity and integrity of packets in the WSN. All the packets captured by each independent *fWSN mote* could be forwarded to the base station, as a central point of analysis, in an effort to prove the authenticity and integrity of the data packet according to the Casey Certainty Scale (Casey, 2002).

According to Casey (2002), the integrity and authenticity of information is more certain if this information was recorded by different independent sources. Each *mote* can, in essence, be seen as an independent source. Thus, the authenticity and integrity of each packet can be determined based on the number of *motes* in the network that have received the same broadcasted packet. This paper therefore assumes that, in accordance with the Casey Certainty Scale (Casey, 2002), a packet that has been seen by a larger number of *motes* has far greater authenticity and integrity than a packet that has only been seen by a few forensic *motes* in the network.

The above technique constitutes only one of several ways to determine the authenticity and integrity of the packets in a WSN. Time stamping and the sequence of packets can also be used for this purpose. However, time stamping in a WSN is a tedious task. The next subsection is nevertheless devoted to it.

3.3. Time stamping

Time stamping in a WLAN environment is a fairly easy task, since all the devices in a WLAN would under normal conditions either have access to a time server or have been set with the correct time. Thus, time stamping in the logs for a WLAN would under most conditions be correct, provided that the device has not been tampered with or is not faulty. In the case of a WSN, however, only the management server (which is connected to the base station) has a sense of time. The *motes* in a WSN environment have no sense of physical world time and the only measurement they can use is their own sense of time, which is the time that has elapsed since they were switched on (Sundararaman, Buy & Kshemkalyani, 2005; Su & Akyildiz, 2005; Sun, Ning & Wang, 2006). Such elapsed time can be measured on WSN devices in terms of ticks, where each tick represents 100 nanoseconds (Sundararaman, Buy & Kshemkalyani, 2005). This uptime, although fairly accurate, is a poor indication of time, because each *mote* in the entire network has to be switched on simultaneously and the time should also be synchronised by transmitting their uptime along with their data packets. It is impractical to switch on *motes* simultaneously and synchronisation is not feasible due to resource restrictions.

When tests were conducted concerning the time stamping of WSNs, the authors noted that it takes at most one second to capture any data packet and transmit it to the *fWSN* base station. This nevertheless introduced a time delay between capturing a packet and receiving it at the base station. The time delay also differed according to the distance of the *fWSN mote* from the base station in terms of hops and physical distance. Thus the time stamps at the base station are not an accurate reflection of when the packet was initially captured, as the base station is the only device that can assign an accurate time stamp if it is connected to the management server. (The reason for this is that only the management server has access to a time server (Sundararaman, Buy & Kshemkalyani, 2005; Su & Akyildiz, 2005).) It is also important to note that each *fWSN mote* captures packets sequentially, in the order that the *oWSN motes* transmit their data packets. This proves to be a vital piece of information, because one would then be able to claim that even if the time stamps are altered, the sequence would still be intact. The order in which they arrive at the *fWSN* will stay intact even if the time stamps are slightly delayed. This allows one to assume that the time delay between capturing the packet and sending it to the forensic base station would not really affect the authenticity and integrity of the packets, as the sequence of packets can be used to determine their authenticity and integrity.

The trustworthiness of log time stamps is an issue that many digital forensics researchers have queried and investigated (Schatz, Mohay & Clark, 2006; Schneier & Kelsey, 1999). The dilemma faced by the fWSN is merely intensified. It becomes a more severe issue to trust the time stamps as the limitation as having no access to a centralised time server for WSNs might prevent them from reflecting the correct time. However, since the sequence of the data packets is not altered, this (rather than the time stamps) could be used to verify the authenticity and integrity of the data packets. This paper therefore assumes that the fixed sequence of the data packets is more important than the precise time at which they were transmitted. More information can be gathered by looking at the sequence of the data packets than by looking at their time of transmission.

It is therefore sufficient to capture the data packets and merely provide a time stamp for them as soon as they arrive at the *fWSN* base station. In the event that this is done, one would admittedly create a time stamp error. The time stamp error would nonetheless be a constant error for each *oWSN mote* respectively, as it would reflect the time the data packet was first transmitted together with the added time it took for this data packet to reach the *fWSN* base station. The *fWSN* base station, which is connected to a time server, assigns a time stamp to each data packet upon its arrival there. This allows the order of the packets to be kept intact and records a one-second error on the time stamp of each packet due to the fact that the base station assigns the time stamps and not the forensic *mote* that captured the packet initially. The time stamp error stays constant for all the packets received from a specific *mote* in the sensor field and thus it is still possible to guarantee the authenticity and integrity of a packet. This constant error could be measured, if needed, by comparing the time stamps at the *oWSN* base station and the *fWSN* base station. The time stamp, combined with the sequence of the data packets, would then be sufficient to be used in a forensic investigation.

Another issue that the authors have considered while examining the differences between WLANs and WSNs is the feasibility of modifying the network after it has been deployed. This matter is discussed in the following subsection.

3.4. Modification of the network after deployment

Being able to modify the network after deployment is the only factor that was found to be fairly similar between WLANs and WSNs, as it is always possible to modify code on a device by retracting it from the field, redeveloping it and then redeploying it back into the field. However, the practicality of altering *oWSN* devices after deployment must be taken into consideration. It is important to remember that *oWSN motes* are usually scattered in an area and to alter them, one would have to go and collect the entire network and redeploy it. Hence, it seems essential that the *oWSN* should not be modified to accommodate an *fWSN* solution. This is the very reason why the authors have opted to add an overlaying *fWSN* to the *oWSN* in order to do all the forensic monitoring. The overlaying *fWSN* would consist of a separate set of WSN *motes* that does not affect the *oWSN* and also requires no modification of the *oWSN*.

The difficulty and impracticality of modifying the *oWSN* has led the authors to believe that this should also be seen as a specific requirement when attempting to provide forensic readiness to a WSN environment. Considering that we cannot easily alter the *oWSN,* we must ensure that the *fWSN* should be able to handle any type of protocol headers and footers that could originate from the *oWSN*. Against this background, the next subsection focuses on the protocol data packets that are used by WSN devices and the reasons why it is important to take this into consideration when implementing an overlaying *fWSN*.

3.5. Protocol Data Packets

The *oWSN* can have many different types of communication protocols in its normal operation. For example, the data packets can include packets to determine the routing protocol, sensory packets, encrypted packets or even malformed packets. In order to ensure that all of the possible protocols used in WSNs are encapsulated in this approach, it has been assumed that the *oWSN* uses an address-free protocol. This protocol generates the largest amount of network overhead in WSNs, as it would cause data to be sent from a source *mote* in the network to every other *mote* in the network on each data transmission (Dunkels, Osterlind & Zhitao, 2007). The most commonly used address-free protocols are data dissemination protocols, where neither the sender *mote* nor any of the other motes in the network knows the address of the receiving *mote*. If the *fWSN* is able to successfully log this communication of an address-free protocol in a way that ensures authenticity and integrity, one could assume that the name-based WSN protocols would effortlessly be accounted for, as they have much less network overhead (Dunkels, Osterlind & Zhitao, 2007).

As is also the case in WLANs, the *motes* in the *fWSN* should listen in promiscuous mode and should be able to handle any type of packet that is transmitted or received by the *oWSN*. The authors define promiscuous mode to be a configuration of the WSN *mote* in which all traffic within the WSN *mote*'s frequency range and wireless range will be received by the WSN *mote*. Thus, if an attacker uses a foreign *mote* to inject data into the *oWSN*, the *fWSN* should also be able to listen in on this data. This requirement should be fairly simple to adhere to, because if the *fWSN* is implemented on the same type of equipment, it should be possible to intercept all communication.

Lastly, the *fWSN* should be using a name-based WSN protocol for communication between other *fWSN motes* as it is more optimal in terms of network overhead than address-free protocols. In name-based protocols the source *mote* knows the address of the receiving *mote* and the *motes* between the sender and receiver know the path to the receiving *mote* (Dunkels, Osterlind & Zhitao, 2007).

All the major differences between WSNs and WLANs have now been discussed. Due to space constraints, discussions on radio frequencies, power supply, network overhead and data integrity have been omitted. However, the following section is devoted to arranging all these factors, including the ones that have been excluded from the discussion, into a single workable list of requirements that need to be adhered to when implementing digital forensic readiness in a WSN environment.

4. Forensic Readiness Requirements for WSNs

The previous sections identified the factors that differentiate between WLANs and WSNs in terms of digital forensic readiness. These factors were simply broad overviews of issues to be considered in the WSN environment (most of which do not exist in a WLAN environment).

The authors consequently propose a broad, yet detailed set of the important requirements to be adhered to in order to successfully implement digital forensic readiness in a WSN environment. This list of requirements (see Table 2) could serve as a good starting point for anyone working on digital forensic readiness and makes it easier for an individual to implement digital forensic readiness within a WSN environment. Most other researchers focus mainly on one or two of these requirements by going into more detail on them in their research papers, but many other requirements are usually not mentioned, regardless of their importance.

Table 2 therefore gives a quick but comprehensive overview and summarises *all* the important requirements that need to be taken into account in order to achieve digital forensic readiness in an IEEE 802.15.4 WSN environment.

Factors	Detailed requirement list
Communication Protocol	1. The *fWSN* should use a receipt acknowledgement packet protocol to ensure that all data packets captured by the *motes* in the field do indeed reach the base station.
	2. The broadcasted communication from the *oWSN* should be intercepted in a manner that ensures that the data packets are not altered in any fashion.
	3. The *fWSN* should be able to capture all possible types of communication that can be sent from the *oWSN*.
Proof of Authenticity and Integrity	4. The authenticity and integrity of all the data packets should remain intact while being captured on the *fWSN*.
	5. The data packets that are captured in the *fWSN* should be stored in such a way that their authenticity and integrity are not compromised.
	6. It should be possible to verify the authenticity and integrity of all the data packets in case a digital investigation takes place.
Time Stamping	7. The data packets should have a time stamp assigned to them that does not violate their authenticity and integrity.
	8. The sequence of the packets captured should reflect the true sequence in which they were transmitted from the original network.
Modification of the network after deployment	9. It should be possible to implement the *fWSN* without any modification of the *oWSN*.
Protocol Data Packets	10. The *fWSN* should be designed in such a manner that the network topology or the routing protocol used by the *oWSN* does not influence the *fWSN*'s operation.
Radio Frequencies	11. The *fWSN* should be able to communicate on the same radio frequencies that are available to the *oWSN*.
	12. All communication within the *fWSN* should occur on a frequency not utilised in the *oWSN*.
	13. If an intruder WSN is in the area and communicates on a frequency that influences the *oWSN,* then the *fWSN* should be able to forensically capture these data packets.
Power Supply	14. The *fWSN* should not increase power consumption in the *oWSN* and the *fWSN* should have at least the same or a longer network lifetime than the *oWSN* in terms of battery power.
Network Overhead	15. While intercepting communication, there should be no extra network overhead on the *oWSN*.
Data Integrity	16. The *fWSN* should by no means be able to influence the *oWSN* or influence any sensory data transmitted within the *oWSN*.

Table 2: Requirements in order to achieve digital forensic readiness in a IEEE 802.15.4 WSN environment

The list in table 2 provides a sound basis to start from when attempting to achieve digital forensic readiness in a WSN environment. The following section concludes this paper and proposes future work.

5. Conclusion

Wireless sensor networks constitute a type of network that makes any type of digital forensic analysis very difficult due to the nature of the network. This paper therefore proposed a list of requirements that need to be taken into consideration when implementing digital forensic readiness for an IEEE 802.15.4 wireless sensor network.

The main aim of this paper was to establish the differences between IEEE 802.15.4 wireless sensor networks and IEEE 802.11x wireless networks from a digital forensic readiness point of view. The problem was that currently there is no formal set of requirements for successfully implementing digital forensic readiness in wireless sensor networks. This problem was addressed by focusing on the special needs WSNs have for digital forensic readiness and providing a list of requirements that need to be taken into account when implementing digital forensic readiness in WSNs.

In future research, the authors intend to explore this list of requirements in greater detail and develop a digital forensic readiness prototype for wireless sensor networks. The focus of the research will be to develop the prototype in such a way that it proves to be robust enough to function in most types of WSNs.

6. References

Akyildiz, I.F., Su, W., Sankarasubramaniam, Y. and Cayirci, E. (2002) 'Wireless sensor networks: a survey', *Computer Networks*, vol. 38, no. 4, pp. 393-422.

Casey, E. (2002) 'Error, Uncertainty and Loss in Digital Evidence', *Internation Journal of Digital Evidence*, vol. 1, no. 2, Summer.

Chen, J., Jiang, M. and Liu, Y. (2005) 'Wireless LAN security and IEEE 802.11i', *Wireless Communications, IEEE*, vol. 12, no. 1, February, pp. 27-36.

Chong, C. and Kumar, S.P. (2003) 'Sensor networks: evolution, opportunities, and challenges', *Proceedings of the IEEE*, vol. 91, no. 8, pp. 1247-1256.

Crossbow Technology Inc (2007) *Imote2 Hardware Reference Manual*, Revision A edition, San Jose: Crossbow Technology Inc.

Doufexi, A., Armour, S., Butler, M., Nix, A., Bull, D., McGeehan, J. and Karlsson, P. (2002) 'A comparision of the HIPERLAN/2 and IEEE 802.11a wireless LAN standards', *IEEE Communications Magazine*, vol. 40, no. 5, May, pp. 172-180.

Dunkels, A., Osterlind, F. and Zhitao, H. (2007) 'An adaptive communication architecture for wireless sensor networks', Proceedings of the 5th international conference on Embedded networked sensor systems, Sydney, Australia, 335-349.

Elson, J. and Estrin, D. (2001) 'Time synchronization for wireless sensor networks', Proceedings of the 15th International Symposium on Parallel and Distributed Processing, 1965-1970.

Estrin, D., Girod, L., Pottie, G. and Srivastava, M. (2001) 'Instrumenting the world with wireless sensor networks', Proceedings of the 2001 IEEE International Conference on Acoustics, Speech and Signal Processing, 2033-2036.

Guizani, M. and Raju, A. (2005) 'Wireless Networks and Communications Security', in Xiao, Y., Li, J. and Pan, Y. (ed.) *Security and Routing in Wireless Networks*, 3rd edition, New York: Nova Science Publishers.

Heinzelman, W.R., Kulik, J. and Balakrishnan, H. (1999) 'Adaptive protocols for information dissemination in wireless sensor networks', In Proceedings of the 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking, Seattle, 174-185.

Kahn, J.M., Katz, R.H. and Pister, K.S. (1999) 'Next century challenges: mobile networking for "Smart Dust"', In Proceedings of the 5th Annaul ACM/IEEE International Conference on Mobile Computing and Networking, New York, 271-278.

Komori, T. and Saito, T. (2004) 'A secure wireless LAN system retaining privacy', 18th International Conference on Advanced Information Networking and Applications, Kanagawa, 370-375.

Lu, C., Blum, B.M., Abdelzaher, T.F., Stankovic, J.A. and He, T. (2002) 'RAP: a real-time communication architecture for large-scale wireless sensor networks', Proceedings of the 8th IEEE International Workshop on Real-Time and Embedded Technology and Applications Symposium, 55-66.

Mainwaring, A., Culler, D., Polastre, J., Szewczyk, R. and Anderson, J. (2002) 'Wireless sensor networks for habitat monitoring', In Proceedings of the 1st ACM International Workshop on Wireless Sensor Networks and Applications, New York, 88-97.

Mouton, F. and Venter, H.S. (2009) 'A Secure Communication Protocol for Wireless Sensor Networks', Proceedings of the Annual Security Conference "Security Assurance and Privacy: organizational challenges", Las Vegas.

Polastre, J., Hill, J. and Culler, D. (2004) 'Versatile low power media access for wireless sensor networks', Proceedings of the 2nd international conference on Embedded networked sensor systems, Baltimore, 95-107.

Rowlingson, R. (2004) 'A Ten Step Process for Forensic Readiness', *International Journal of Digital Evidence*, vol. 2, no. 3.

Schatz, B., Mohay, G. and Clark, A. (2006) 'A correlation method for establishing provenance of timestamps in digital evidence', Proceedings of the 6th Annual Digital Forensics Research Workshop, 98-107.

Schneier, B. and Kelsey, J. (1999) 'Secure audit logs to support computer forensics', *ACM Transactions Information System Security*, vol. 2, no. 2, May, pp. 159-176.

Shnayder, V., Hempstead, M., Chen, B., Allen, G.W. and Welsh, M. (2004) 'Simulating the power consumption of large-scale sensor network applications', In Proceedings of the 2nd international Conference on Embedded Network Sensor Systems, Baltimore, 188-200.

Slijepcevic, S. and Potkonjak, M. (2001) 'Power efficient organization of wireless sensor networks', In IEEE International Conference on Communications, 472-476.

Sohrabi, K., Gao, J., Ailawadhi, V. and Pottie, G.J. (2000) 'Protocols for self-organization of a wireless sensor network', *Personal Communications, IEEE Wireless communications*, vol. 7, no. 5, October, pp. 16-27.

Su, W. and Akyildiz, I.F. (2005) 'Time-diffusion synchronization protocol for wireless sensor networks', *IEEE/ACM Transactions on Networking*, vol. 13, no. 2, pp. 384-397.

Sundararaman, B., Buy, U. and Kshemkalyani, A.D. (2005) 'Clock synchronization for wireless sensor networks: a survey', *Ad Hoc Networks*, vol. 3, no. 3, pp. 281-323.

Sun, K., Ning, P. and Wang, C. (2006) 'TinySerSync: secure and resilient time synchornization in wireless sensor networks', Proceedings of the 13th ACM conference on Computer and communications security, Alexandria, 264-277.

Tan, J. (2001) *Forensic Readiness*, Technical Report edition, Cambridge: @Stake.

Tseng, Y., Ni, S. and Shih, E. (2003) 'Adaptive Approaches to Relieving Broadcast Storms in a Wireless Multihop Mobile Ad Hoc Network', *IEEE Transactions on Computers*, vol. 52, no. 5, pp. 545-557.

Wander, A.S., Gura, N., Eberle, H., Gupta, V. and Shantz, S.C. (2005) 'Energy analysis of public-key cryptography for wireless sensor networks', Third IEEE International Conference on Prevasive Computing and Communications, 324-328.

Xylomenos, G. and Polyzos, G. (1999) 'TCP and UDP Performance over a Wireless LAN', In Proceeedings of the IEEE INFOCOM.

Xylomenos, G., Polyzos, G., Mahonen, P. and Saaranen, M. (2001) 'TCP performance issues over wireless links', *IEEE Communications Magazine*, vol. 39, no. 4, pp. 52-58.

Ye, W., Heidemann, J. and Estrin, D. (2002) 'An energy-efficient MAC Protocol for wireless sensor networks', Proceedings of the 21st Annual Joint Conference of the IEEE Computer and Communication Societies, 1567-1576.

Zhao, F., Shin, J. and Reich, J. (2002) 'Information-driven dynamic sensor collaboration for tracking applications', *IEEE Signal Processing Magazine*, vol. 19, pp. 61-72.