Protecting Wireless Sensor Networks from Internal Attacks

Muhammad Raisuddin Ahmed

Faculty of Education, Science, Technology and Mathematics

University of Canberra, ACT 2601, Australia

Thesis submitted in partial fulfilment of the requirements for the

Degree of Doctor of Philosophy

May 2014

Abstract

Recently, technological advances in the design of processors, memory and radio communications have propelled an active interest in the area of distributed sensor networking, in which a number of independent, self-sustainable nodes collaborate to perform information gathering and processing in real time. Networks of such devices are commonly referred to as Wireless Sensor Networks (WSNs), which are envisioned as a bridge between the modern broadband packet data networks and the physical world. WSNs have made possible real-time data aggregation and analysis on an unprecedented scale. Naturally, they have attracted attention and garnered widespread appeal towards applications in diverse areas including disaster warning systems, environment monitoring, health care, safety and strategic areas such as defence reconnaissance, surveillance, and intruder detection.

Due to the distributed nature, multi-hope communications and their deployment in remote areas, WSNs are vulnerable to numerous security threats that can adversely affect performance. Therefore, to ensure the functionality of WSNs, security is the first and foremost concern in almost all wireless sensor networking scenarios. WSN mechanisms cannot at present ensure that an attack will not be launched. For example, using a compromised node an adversary could perform an attack acting as a legitimate node of the network to acquire all the information. Such attacks are known as internal attacks. Therefore, it is important to protect the wireless sensor network from internal attacks, which is the purpose of this thesis.

This thesis investigates internal security issues in wireless sensor networks (WSNs) and proposes relevant solutions. The development of multi stage mechanisms to protect WSNs from internal attacks is performed. The major contributions of this thesis to prevent internal attacks are summarised below.

Initially, this thesis developed misbehaviour identification mechanisms with multi agents through timing control, the pairwise key method and cosine similarity based on the abnormal behaviour identification method (ABIM). It is a fast, robust mechanism, and also gives good results when data sets are distinct or well separated from each other.

Secondly, this research investigated and took the advantage of the Dempster-Shafer theory (DST) to develop a novel algorithm for protecting WSNs from internal attacks. This algorithm observes neighbour nodes in a WSN and uses parameters to make judgments for the behaviour based on the DST. The DST considers the observed data as a hypothesis. If there is uncertainty about which hypothesis the data fits best, the DST makes it possible to model several single pieces of evidence within the relations of multi hypotheses. Using this method the system does not need any prior knowledge of the pre-classified training data of the nodes in a WSN.

Thirdly, this work extended the algorithm of the Markov Chain Monte Carlo (MCMC) – Metropolis-Hasting (MH) to our research to detect internal attacks on WSNs. With the MCMC method, it is possible to generate samples from an arbitrary posterior density and to use these samples to approximate expectations of quantities of interest. Moreover, it works in real time by constricting the sample chain and computes the changes together with an acceptance ratio. The new algorithm can decide the internal attacker based on the acceptance ratio.

This work used the fourth generation programming language MATLAB and Java based development J-Sim for simulations. The simulation results show that the algorithm for the detection of the internal attacks is effective. In a simulation, the accuracy of detection in one hop communication, in the three stages, is between 75% and 95% based on the percentage of the compromised node. The accuracy of detection is higher for compromised nodes less than 10% even though the system does not survive if the compromised node is more than 50%.

To my daughter Sarvia Rameen Ahmed

Acknowledgements

For the thesis support and encouragement comes from several sources in various ways. In particular, I would like to thank Professor Xu Huang for accepting me as a Ph.D. student at University of Canberra under his supervision. Prof. Xu always provided me with encouragement, support and sufficient room to think and grow. His understanding and advice have been crucial factors in the successful completion of this work. I consider myself really fortunate to have him as my guide and advisor. I would also like to express my deep-felt gratitude to my co-supervisor, Professor Dharmendra Sharma, for his advice and encouragement.

In addition I would like to thank our collaborators, Associate Professor Hongyan Cui form University of Posts and Telecommunications, China and Professor Li Shutao from Hunan University, China for their advice and support.

I am indebted to my family for their love, advice and support throughout my PhD study, to my mother for her constant encouragement and to my wife for her unlimited love support and understanding throughout the journey.

Contents

List of Figures

List of Tables

Abbreviations

ABIM	Abnormal Behaviour Identification Mechanism
ADC	Analog to Digital Converter
CPU	Central Processing Unit
CSIRO	Commonwealth Scientific and Industrial Research Organisation
DAC	Digital to Analog
DARPA	Defence Advanced Projects Research Agency
DoS	Denial of Service
DST	Dempster Shafer Theory
ECC	Elliptic Curve Cryptography
GPS	Global Positioning System
HIDS	Hybrid Intrusion Detection System
IEEE	Institute of Electrical and Electronics Engineers
IIDS	Integrated Intrusion Detection System
IHIDS	Integrated Hybrid Intrusion Detection System
JTAG	Joint Test Action Group
LEACH	Low Energy Adaptive Clustering Hierarchy
MAC	Media Access Control
MAS	Multi Agent System
MCMC	Markov Chain Monte Carlo
MH	Metropolis–Hastings
MIMO	Multiple Input Multiple Outputs
MIPS	Million Instructions Per Second
OFDMA	Orthogonal Frequency-Division Multiplexing

OSI	Open System Interconnection
RAM	Random Access Memory
RSSI	Received Signal Strength Indicator
RTT	Round Trip Time
SNR	Signal to Noise Ratio
SOSUS	Sound Surveillance System
SSL	Secure Sockets Layer
SVM	Support Vector Machine
TEEN	Threshold sensitive Energy Efficient sensor Network protocol
USB	Universal Serial Bus
WAMR	Wireless Automatic Meter Reading
WSNs	Wireless Sensor Networks
XOR	Exclusive OR

List of Publications from PhD Research

This Thesis is a monograph, which contains some unpublished material, but is mainly based on the following publications during my PhD research.

Conferences:

1. **M. R. Ahmed**, X. Huang, H. Cui, N. Srinath "A Novel Two-Stage Multi-Criteria Evaluation for Internal Attack in WSN", IEEE ISCIT 2013 Samui Island, Thailand, September 4-6, 2013. [ERA Rank B]
2. **M. R. Ahmed**, X. Huang, H. Cui, "A Novel Evidential Evaluation For Internal Attacks With Dempster-Shafer Theory in WSN", IEEE TrustCom2013 Melbourne, Australia, 16-18 July, 2013. [ERA Rank A]
3. X. Huang, **M. R. Ahmed**, H. Cui, S. Li, "Malicious Node Detection for the Future Network Security from Epistemic Uncertainties", IEEE WPMC 2013, Atlantic City, New Jersey, USA, June 24-27, 2013. [ERA Rank C]
4. X. Huang, **M. Ahmed**, D. Sharma "Protecting Wireless Sensor Networks from Internal Attacks Based on Uncertain Decisions," WCNC 2013, IEEE Wireless Communication and Networking Conference 2013. Shanghai China, 7 -10 April, 2013. [ERA Rank B]
5. X. Huang, **M. Ahmed**, D. Sharma "Novel Protection from Internal Attacks in Wireless Sensor Networks," NETCOM 2012: The Fourth International Conference on Networks & Communications, Dec. 22-24, 2012, Chennai, India. [ERA Rank C]
6. **M. R. Ahmed**, X. Huang, D. Sharma, H. Cui "Protecting WSN from Insider Attack by Misbehaviour Judgement", WCSN 2012 , Phitsanulok, Thailand, December 19-23, 2012. [ERA Rank C]
7. X. Huang, D. Sharma, **M. Ahmed**, H. Cui "Protecting an WSN from Internal Attack Based on Epistemic Uncertainties," The 18th IEEE

International Conference On Networks" ICON 2012, December (12-14), 2012, Singapore. [ERA Rank B]

8. **M. R. Ahmed**, X. Huang, D. Sharma, H. Cui "Wireless Sensor Network: Characteristics and Architectures", ICIS 2012 Penang , Malaysia, December 6-7, 2012. [ERA Rank C]

9. **M. R. Ahmed**, X. Huang, D. Sharma, H. Cui "Protecting WSN from Internal Attack with multi-criteria evaluation using Dempester-shafer Theory", ICIS 2012 Penang , Malaysia, December 6-7), 2012. [ERA Rank C]

10. **M. R. Ahmed**, X. Huang, D. Sharma, "A Novel Misbehavior Evaluation with Dempster-Shafer Theory in Wireless Sensor Networks", MobiHoc'12, June 11–14, 2012, Hilton Head, South Carolina, USA. ACM. [ERA Rank A]

11. **M. R. Ahmed**, X. Huang, D. Sharma, "Wireless Sensor Network internal attacker Identification with Multiple Evidence by Dempster-Shafer Theory", ICA3PP'2012, September 4–7, 2012, Fukuoka, Japan. [ERA Rank B]

12. X. Huang, D. Sharma, and **M. Ahmed**, "Security Computing for the Resiliency of Protecting from Internal Attacks in Distributed Wireless Sensor Networks," 12th International Conference on Algorithms and Architectures for Parallel Processing (ICA3PP- 2012), September 04-07, 2012, Fukuoka, Japan. [ERA Rank B]

13. **M. R. Ahmed**, X. Huang, D. Sharma," Dempster-Shafer Theory to Identify Insider Attacker in Wireless Sensor Network", NPC 2012, September 6 - 8, 2012, Gwangiu, Korea. [ERA Rank C]

14. **M. R. Ahmed**, X. Huang, D. Sharma, "A Novel Framework for Insider Attacker Identification and Detection for Wireless Sensor Networks", IEEE, ICIS 2012 conference to be held in Kuala Lumpur, Malaysia February (19-21), 2012, pp431-434. [ERA ranked C]

15. X. Huang, **M. Ahmed**, D. Sharma, "Timing Control for Protecting from Internal Attacks in Wireless Sensor Networks", IEEE, ICOIN 2012 conference to be held in Bali Indonesia February (1-3), 2012. pp. 7-12. [ERA Ranked B]

16. **M. R. Ahmed**, X. Huang, D. Sharma, "A Taxonomy of internal attacks in Wireless Sensor Networks", IEEE, ICIS 2012 conference to be held in Kuala Lumpur, Malaysia February (19-21), 2012. Pp427-430. [ERA Ranked C]

17. X. Huang, **M. R. Ahmed**, and D. Sharma, "A novel Protection for Wireless Sensor Networks from Internal Attacks," International MultiConference of Engineers and Computer Sciences 2012 (IMECS 2012), Hong Kong, 14-16 March, 2012. ISBN: 978-988-19251-1-4. Publisher: Newswood Limited, Organization: International Association of Engineers. Proceeding pp374-379.

18. X. Huang, **M. Ahmed**, D. Sharma, "A Novel Algorithm for Protecting from Internal Attacks of Wireless Sensor Networks", 2011 Ninth IEEE/IFIP International Conference on Embedded and Ubiquitous Computing. 24-26, Oct 2011 Melbourne, pp344-349, Australia. [ERA Ranked C]

19. X. Huang, **M. Ahmed**, and D. Sharma, "Protecting from inside attacks in wireless sensor networks," 2011 IEEE Ninth International Conference on Dependable, Autonomic and Secure Computing (DASC 2011), December 12-14, 2011, Sydney, Australia. Proc. pp186-191. [ERA Ranked C]

Journals:

1. **M. R. Ahmed**, X. Huang, H. Cui, "Mrakov Chain Monte Carlo Based Internal Attack Evaluation for Wireless Sensor Network", IJCSNS International Journal of Computer Science and Network Security, VOL.13 No.3, March 2013, page 23-31. [ERA Ranked C]

2. **M. R. Ahmed**, X. Huang, D. Sharma, "A Novel Two-Stage Algorithm Protecting Internal Attack From WSNs", International Journal of

Computer Networks & Communications (IJCNC) Vol.5, No.1, January 2013, page 97-116. [ERA Ranked C]

3. **M. R. Ahmed**, X. Huang, H. Cui, "Smart Decision Making for Internal Attacks in Wireless Sensor Network ", IJCSNS International Journal of Computer Science and Network Security, VOL.12 No.12, December 2012, page 15-23. [ERA Ranked C]

4. **M. R. Ahmed**, X. Huang, D. Sharma, "A Novel Framework for Insider Attacker Identification and Detection for Wireless Sensor Networks", International Journal of Computer and Communication Engineering, volume 6, 2012, page 148-151.

Chapter 1 Introduction

The advances in the fields of semiconductor devices and large scale transistor integration coupled with the development of high speed broadband wireless technologies such as MIMO-OFDM have led to the deployment of wireless sensor networks (WSNs). WSNs consist of spatially distributed autonomous devices to cooperatively monitor real world physical or environmental conditions, such as temperature, sound, vibration, pressure, motion, pollution and location. This technology is also widely used by military applications, such as battlefield surveillance, transportation monitoring, and sensing of nuclear, biological and chemical agents. Recently, this technology has developed and been widely used in daily life as WSNs are low cost, low power, rapid deployment, have self-organizing capability and cooperative data processing, including applications for habitat monitoring, intelligent agriculture and home automation.

The major components of a normal WSN sensor node are a microcontroller, memory, transceiver, power source and one or more sensors to detect the physical phenomena. The structure of the sensor node is generally divided into four major parts: sensing unit, processing unit, communication unit and power unit. A

sensor node sends the measurement of the physical phenomenon to the sink which has bigger memory and processing power. Depending on the application scenario, sometimes extra hardware is added in the sensor nodes and a deployment strategy is devised. Normally, in applications for WSNs the environment is unpredictable such as hostile, with remote harsh fields or disaster areas, sometimes called toxic environments. Hence, no standard deployment strategy existed. The deployment usually involves scattering or by possibly carrying out the application scenario. Despite their quick deployment and significant advantages over traditional methods, WSNs have to face various security problems because of their nature and the possibility of the presence of one or more faulty or malicious nodes in the existing network.

There are many technically interesting research discussions involving WSNs, such as development of models and tools for the design of better WSNs architecture and elaboration of standard protocols in WSN adapted to work robustly on certain scenarios. However, one of the most important issues that remains subject to debate is security. The emphasis in this thesis focuses on security in WSNs. More precisely, the work focuses on investigating models preventing internal attacks on WSNs.

1.1 Motivation

Wireless communication is the transfer of information between two or more points that are not connected by electrical conductors. Most of the wireless communication technology uses radio waves in order to transfer information between the points which are known as nodes. One application domain of wireless communication is wireless sensor networks. WSN is a distributed system, containing resource or constrained nodes that work in an ad hoc manner using multi-hope communication [1]. WSNs and Internet are integrated as a new application area called Internet of Things (IoT), covering almost every area in

current daily life [2]. IoT encourages several novel and existing applications such as environment monitoring, infrastructure management, public safety, medical and health care, home and office security, transportation, and military applications [3]. Figure 1-1 shows the complexity of wireless sensor networks [4], which translate sensing and identification activities into services using WSNs with WSN middleware and access networking. It can use: (i) different communication platforms such as WiFi, wireless LAN, 3G and 4G; (ii) different devices which are based on different processors such as various types of PDA, smart phones and laptops and (iii) all these platforms and devices being built on different architectures such as centralised, distributed or peer-to-peer.

WSNs provide unprecedented ability to identify, observe and understand large-scale, real-world phenomena at a fine spatial-temporal resolution. The applications range from military to daily life. For example, in community services WSNs can (1) provide early warnings for natural disasters such as floods, hurricanes, droughts, earthquakes, epidemics; (2) disseminate surveillance information for cities in parks, hotels, forests, to support municipality service delivery; and (3) provide enjoyment of the city by citizens and tourists through public services support such as monitoring of water quality to ensure that citizens always have clean water or providing free environmental information on the main tourist destinations. In general, the network consists of a data acquisition network and a data distribution network, monitored and controlled by a management centre.

Security is an inevitable need both in wired and wireless communication networks. The ultimate security aim in both networks is to provide confidentiality, integrity, authenticity, and availability of all messages in the presence of resourceful adversaries [5][6]. Every eligible receiver should receive all messages intended for the message recipient and be able to verify the integrity of every message as well as the identity of the sender. Adversaries should not be able to infer the contents of any message.

Figure 1-1 : The complexity of WSNs

In the wireless paradigm wireless sensor networks continue to grow because of their application scenarios and cost effectiveness. A major benefit of these systems is that they perform in network processing to reduce large streams of raw data into useful aggregated information [7]. Protecting information is critical. The traditional computer network security goal is to deliver the message to the end user in a reliable way. The leading traffic pattern in the conventional computer network is end to end communication. The message content is not important beyond the necessary header. In this process the message authenticity, integrity and confidentiality are usually achieved by an end to end security mechanism such as Secure Socket Layer (SSL) [8].

The use of wireless communication technology in WSNs introduces more challenges compared to that of fixed wired networks. The wireless medium is not only easier to eavesdrop on than guided media; it is also vulnerable to jamming

and other kinds of Denial of Service (DoS) attacks [9]. Thus, effective security mechanisms became vital for WSNs.

Wireless sensor networks pose unique security challenges. The traditional security techniques used in traditional networks cannot be efficiently applied to WSNs directly to deal with attacks, because WSNs have the following characteristics:

- The sensor networks should be economically viable as sensor devices are limited in their energy, computation, and communication capabilities.
- Unlike traditional cases wireless sensor nodes are often deployed in accessible areas, which presents the additional risk of physical attack.
- Wireless sensor networks normally have open media within the deployment environment, which increases challenges to the security.

Consequently, existing security mechanisms are inadequate, and new models are needed in order to ensure the functionality of WSNs, especially in a malicious environment or in the case of internal attacks (internal attacks are from compromised nodes, which are actually part of the network and act as a legitimate node). Internal attacks are found in almost every layer [10]. In the following, this thesis discusses only a few examples to highlight "internal attacks" which can be at different levels of a WSN.

At the physical layer a node can inject fake messages, or corrupt the messages by affecting the radio signal through compromised nodes (a compromised node is defined as the node that shares the information with an adversary by acting as a legitimate node). For example, in wormhole attacks, the attacker takes the message from one area and displays it in another area.

In the data link layer, an attacker can compromise a node to get the cryptographic information that has been implemented in the data link layer security mechanism, which is used to fulfil the security requirements of (authenticity, integrity and confidentiality) to exchange the information with the neighbour nodes. This enables the adversary for example to perform attacks such as eavesdropping, by decrypting the encrypted messages exchanged between the neighbouring node and compromised node.

The data link layer security mechanism can provide extra security to protect the message of a higher level such as network layer, transport layer. For example, if the security is broken at the data link layer an attacker can inject the bogus routing information to disrupt the routing functionality at the network layer.

At the network layer, the compromised node normally advertises the routes and creates routes to prevent new routes from being created by normal node. By attacking the routing protocols, the attacker can absorb the network traffic and control the traffic follow.

The transport layer objective is setting up the end to end connection. In this layer an attacker creates a large number of half opened Transport Control Protocol (TCP) connections with receiver but never completes the handshake to fully open the connection. In the application layer an attacker can send false messages to the sink that result in a false alarm such as false data injection [11]. Thus, from the above description an effective security mechanism is necessary to protect the WSN from internal attacks.

Internal attacks cause serious damage to WSNs [12], For functioning WSNs in the malicious (internal attacker) environment an effective security mechanism is essential. Layer based security alone is not enough to protect the whole network, as an internal attacker can completely access any message routed through any

layer, and can modify, suppress, or even discard the message. Besides, for the conventional cryptographic way, it is not possible to protect the WSNs from internal attacks because of the unpredictable wireless channel [13]. The unreliable channel makes it easy for the attacker to compromise the node and establish an untrustworthy relationship with attacker.

Therefore, to develop an efficient security mechanism to protect WSNs from internal attacks becomes a critical and challenging task. In order to do that, it is important to understand WSNs and their security strategies. This knowledge facilitates the development of new efficient methods to protect WSNs from internal attacks (compromised node). The new problems inspire new research and provide an opportunity to properly address sensor network security.

1.1.1 Motivation summary

The current challenge for the WSNs' security research is to save WSNs from internal attacks as discussed in the earlier paragraph. This problem is the main research question i.e., can internal attacks be detected to secure WSNs. Hence, the primary objective of the thesis is to develop mechanisms to protect WSNs from internal attacks.

1.2 Contribution

In this thesis, the work mainly focuses on the threats from internal attacks in wireless sensor networks (WSNs). The results of this research work reveal internal attack detection mechanism to secure wireless sensor networks. The agenda is realized in two parts: investigation of the nature of internal attacks and development of detection techniques, which are integrated into the research

contributions in this work. The research work includes the following major contributions:

- To investigate and explore the nature of internal attacks in WSNs.
- To develop a misbehaviour identification mechanism using a multi-agent system, pair wise key and cosine similarity.
- To extend the Dempster Shafer Theory method to the action of detecting an internal attack.
- To create a statistical analysis using the Metropolis Hasting most popular Markov Chain Monte Carlo for decision making about the internal attacks.

Contributions to this thesis are also shown in the list of my publications within my PhD research.

1.3 Thesis Structure

Chapter 2 provides a literature review of wireless sensor networks, investigating the gaps between the description of the literature and challenges presented in Chapter 1. Chapter 2 also introduces the main characteristics, architecture, and existing node platform and application scenarios, which motivate the work performed in the thesis. Brief discussion of wireless sensor network threats, attacks and security follows.. Then this work carefully investigates security challenges in WSNs. In that Chapter the work presents the taxonomy of internal attacks within WSNs for future discussion in the following Chapters. The shortcomings and drawbacks of the approaches employed to secure WSNs, as covered in the literature are discussed.

Chapter 3 presents an analysis of misbehaved nodes to find an internal attacker in a WSN. This chapter discusses the system model and sensing model of the

WSN. A multi-agent mechanism, pairwise key, and cosine similarity method are introduced to detect the internal attacker through misbehaviour identification.

The Dempster-Shafer theory (DST) is introduced in Chapter 4 to detect an internal attacker in a WSN. In this Chapter, DST is used to make multi criteria evaluation about the internal attacks. This work presents the DST mathematical framework as a case study to describe the whole processing of the implementation, algorithm and simulation.

Chapter 5 introduces the Markov Chain Monte Carlo (MCMC) method. MCMC techniques are often applied to solve investigation and optimization problems in large dimensional spaces. In this Chapter Bayesian interface, Monte Carlo and Markov Chain are discussed for building an algorithm to detect internal attacks. Metropolis Hastings (MH), the most popular algorithm in MCMC, is also implemented to take the decision about the internal attacker in a WSN, based on the acceptance ratio of nodes. Simulation results showed the acceptance ratio of the nodes to conform the working status of the target WSN.

Chapter 6 concludes the thesis and provides a summary of the research outcomes, and future research work involved in security of wireless sensor networks, particularly internal attacks.

Chapter 2 Literature Review

Modern technological advancements in integrated circuit fabrication made it possible for the deployment of small, inexpensive, low-power, distributed devices to be capable of local processing and wireless communication [14]. Such small devices are called sensor nodes, which are capable of only a limited amount of processing. But when they are coordinated with the information from a large number of other nodes, they have the ability to measure a given physical environment in great detail. Thus, a wireless sensor network can be described as a collection of sensor nodes which coordinate to perform some specific action. Unlike traditional networks, wireless sensor networks depend on dense deployment and coordination to carry out their tasks. In this Chapter, special care will be taken with the challenges for the current wireless sensor network. This Chapter discussed the WSNs evaluation, characteristics, architecture, protocols, applications, security and suggested mechanisms, which lead us to investigate what the gaps between the current and future challenges which lead this research direction.

2.1 Wireless Sensor Networks

During the cold war in the 1950's, the United States Navy had trouble locating soviet submarines due to the lack of underwater visibility. In that regard they developed a mesh of connected hydrophones called the Sound Surveillance System (SOSUS) to locate these submarines. SOSUS was a system that used an underwater acoustic sound microphone, hydrophone to detect the nearest submarines, which is considered one of the first large scale Wireless Sensor Networks [15].

The Defense Advanced Research Projects Agency (DARPA) started to use *Arpanet* to communicate between nodes in the 1980s. The idea was to use this form of communication to allow many low cost sensing nodes to be distributed over a larger area with each node operating autonomously using this form of communication as a central processor deciding where the information collected was best used [16].

In the early 1990s under the Cooperative Engagement Capability, the United States Navy installed a new system that used the sensed data from other nearby vessels, to produce a clearer image of the target. This communication between the vessels extended the range that the naval vessels could detect and engage from. This communication between the vessels extended the range that the naval vessels could detect and engage from.

In the early 2000s [17], DARPA developed software for networks using micro sensors, which was designed to create ad-hoc connections with the sensors. Those were low cost, small as well as disposable, which was considered the first distributed wireless self-contained sensor network. The development leads us to the current wireless sensor networks that have been used in numerous civil applications.

Each node in a wireless sensor network is a self-contained unit comprised of a power supply (generally batteries), a communication device (radio transceivers), a sensor or sensors, analog-to-digital converters (ADCs), a microprocessor, and data storage [1][18]. The nodes self-organize themselves, into wireless sensor networks and data from the nodes is relayed to neighboring nodes until it reaches the desired destination for further processing.

Recently, the WSN's technology has widely been used in our daily life [19]. A typical WSN is shown in Figure 2 -1. In Figure 2-1 an event is detected in the sensor field and the information is routed to the sinker or base station then to the user with several communication media.

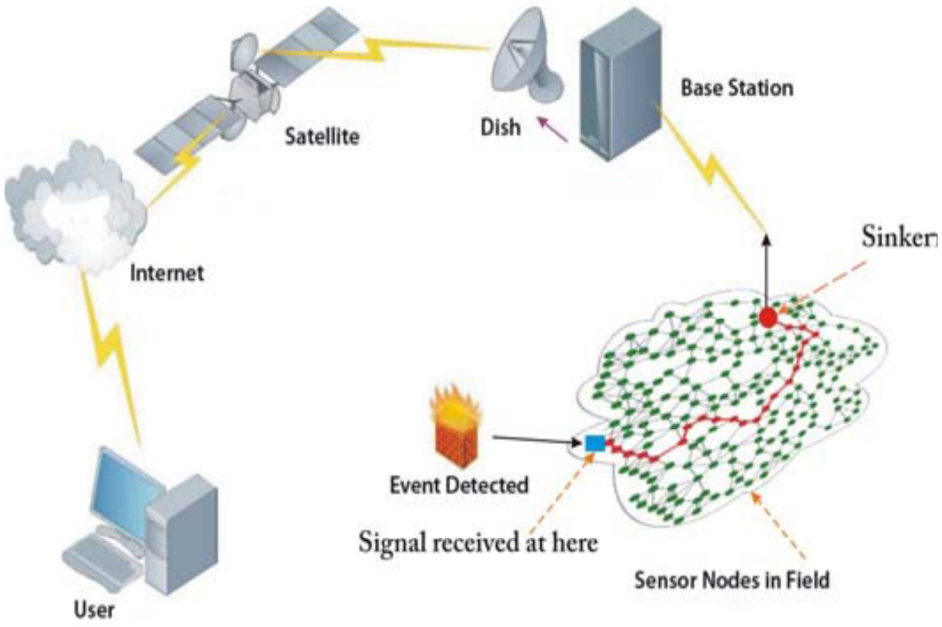

Figure 2-1 : A typical WSN

Wireless Sensor Networks have been applied to a range of applications, monitoring of space which includes environmental and habitat monitoring, indoor climate control, surveillance. Monitoring things example can be outlined

as structural monitoring, condition-based equipment maintenance. In addition, monitoring the interactions of things with each other and the surrounding space e.g., emergency response, disaster management, healthcare, energy sector [20 – 25]. The majority of these applications may be split into two classifications: data collection and event detection.

In various applications of WSNs, the node deployment always draws attention to cover the area of interest. Node deployment strategy is a fundamental issue of a WSN provisioning that is done based on the implementation scenario [24]. The types, number, and locations of devices impact on many intrinsic properties of a WSN, such as coverage, connectivity, cost and lifetime.

Deployment can normally be categorized as either a dense deployment or a sparse deployment. A dense deployment has a relatively high number of sensor nodes in a given field of interest while a sparse deployment would have fewer nodes in the same field. The dense deployment model is usually used in situations where intensive information is needed for every event or when it is important to have multiple sensors cover an area. Sparse deployments may be used when the cost of the sensors make a dense deployment prohibitive or when a WSN needs to achieve maximum coverage using the bare minimum number of sensors [26]. For example, surveillance applications require different degrees of surveillance in different locations, in highly sensitive areas, dense deployment is needed.

The limitations of wireless sensor networks are significant factors and must be addressed when designing and implementing a wireless sensor network for a specific application. Therefore, any security mechanism to extract meaningful and actionable information from WSNs becomes a challenge.

It is noted that, there is some security mechanism developed for wireless ad hoc networks but that cannot be applied for WSNs. Although wireless sensor networks share many properties with wireless ad hoc networks and may require similar techniques such as routing protocols, in certain cases WSN directly prohibit using the protocols proposed in wireless ad hoc networks. Thus, the characteristics and architecture for WSNs and wireless ad hoc networks are different concepts. To demonstrate this issue, the dissimilarities between the WSNs and wireless ad hoc networks (mobile ad hoc networks) are summarized as below: [1][27]

- The number of nodes (hundreds or thousands nodes) in WSNs can be several orders of magnitude higher than the nodes in ad hoc networks.
- WSN Nodes can be densely deployed, so multiple sensors can perform to measure the same or similar physical phenomenon.
- Even, WSNs can be stationary or moving whereas the ad hoc networks is used to moving.
- Nodes in WSNs are prone to failure because of battery exhaustion and hostile environment.
- The topology of a wireless sensor networks changes very frequently caused by so called effective nodes. For example, some nodes can fail after deployment.
- Nodes in WSNs mainly use a broadcast communication paradigm, whereas most ad hoc networks are based on point-to-point communications.
- Nodes in WSNs are limited in power, computational capacities and memory.
- Nodes in WSNs may not have global identification (ID) because of the large amount of overhead and large number of sensors.

A comparison of WSNs and Wireless ad hoc networks is shown in Table 2-1 [28 – 32].

Table 2-1 : WSNs vs Wireless ad Hoc networks

	WSNs	Wireless ad hoc Networks
Communication pattern	Specialized to: Many-to-one One-to-many Local communications	Typically support routing between any pair of nodes
Energy and resources constrained	More	Less
Mobility	Most of the deployment is stationary	Mobile deployment is most
Node co-operation	The modes co-operate each other for different purpose (e.g. sending data, to build trust relationship)	Less cooperative compare to WSNs node
Security mechanism	Authentication and routing based on public key cryptography is too expensive and consume a lot of processing time and memory.	Both public key and asymmetric cryptography are applied.
Routing	Distance vector and source routing protocols are generally too expensive	Support different types of routing protocols.

Although many protocols and security algorithms have been proposed for traditional wireless ad hoc networks, they are not well suited to the unique features and applications requirements of sensor networks. WSN has nature that have been based on its characteristics as listed below [33][34]:

- WSNs improve sensing accuracy by providing distributed processing of vast quantities of sensing information (e.g., seismic data, acoustic data and high-resolution images). When those sensors are networked, sensors can aggregate such data to provide a rich, multi-dimensional view of the environment.
- WSNs can provide coverage of a very large area through the scattering of thousands of sensors.
- Networked sensors in WSNs can continue to function accurately in the face of fail-network self-organization: given the large number of nodes and their potential very large area through the scattering of thousands of sensors.
- Networked sensors in WSNs can continue to function accurately in the face of failure of individual sensors thus, allowing greater fault tolerance through a high level of redundancy.
- Wireless sensor networks can also improve remote access to sensor data by providing sink nodes that connect them to other networks, such as the Internet, using wide-area wireless links.
- WSNs can localize discrete phenomenon to save power consumption.
- The technology in WSN can minimize human intervention and management.
- WSNs can work in hostile and unattended environments.

2.2 Characteristics of WSNs

WSNs are currently used for real-world unattended physical environments to measure numerous parameters. Therefore, the characteristics of a WSN must be

considered for efficient deployment of a network. The discussion of the differences of WSNs with traditional wireless ad hoc networks was done in the above section and now it is necessary to summarize the characteristics of WSNs. The significant characteristics of WSNs are described as follows [35][36]:

Low cost: in a WSN normally hundreds or thousands of sensor nodes are deployed to measure the desired physical environment. In order to reduce the overall cost of the whole network the cost of the sensor node must be kept as low as possible.

Energy efficient: energy in WSNs is used for different purpose such as computation, communication and storage. Sensor nodes consume more energy compared to any other for communication. If they run out of the power they often become invalid as it does not have any option to recharge.

Computational power: normally a node in a WSN has limited computational capabilities as the cost and energy need to be considered.

Communication capabilities: a WSN typical communication uses radio waves over a wireless channel. It has the property of communicating in short range, with limited and dynamic bandwidth. The communication channel can be either bidirectional or unidirectional. With the unattended and hostile operational environment it is difficult to run a WSN smoothly.

Security and privacy: Each sensor node should have sufficient security mechanisms in order to prevent unauthorized access, attacks, and unintentional damage of the information inside of the sensor node. Furthermore, additional privacy mechanisms must also be included.

Distributed sensing and processing: the large number of sensor nodes is distributed uniformly or randomly. In WSNs, each node is capable of collecting, sorting, processing, aggregating and sending the data to the sink. Therefore the distributed sensing provides the robustness of the system.

Dynamic network topology: in general WSNs are dynamic networks. The sensor node can fail for battery exhaustion or other circumstances. Communication channel can be disrupted as well as the additional sensor node may be added to the network. All those, result in frequent changes for the network topology.

Self-organization: the sensor nodes in a network must have the capability of organizing themselves as the sensor nodes are deployed in a unknown fashion in an unattended and hostile environment. The sensor nodes have to work in collaboration to adjust themselves to the distributed algorithm and form a network automatically.

Multi-hop communication: a large number of sensor nodes are deployed in a WSN. Therefore, the feasible way to communicate with the sinker or base station is to take the help of an intermediate node through the routing path. If one needs to communicate with the other node or base station which is beyond its radio frequency, it must be through the multi-hop route by the intermediate node.

Application oriented: WSNs are different from the conventional network due to their nature. It is highly dependent on the application ranges from military, environmental as well as the health sector. The nodes are deployed randomly and spanned depending on the type of use.

Robust Operations: since the sensors in a WSN are going to be deployed over a large and sometimes hostile environment. Therefore, the sensor nodes have to be fault and error tolerant. Therefore, sensor nodes need the ability to self-test, self-calibrate, and self-repair.

Small physical size: sensor nodes are generally small in size with a restricted range. Due to size its energy is limited which makes the communication capability low

Considering the major characteristics of WSNs it is necessary to design WSN architecture. In the sext section this work shall discuss WSNs architecture.

2.3 Architecture of WSNs

The network architecture is crucial for WSNs to make them reliable and scalable. In fact, the design of architecture of WSNs enables the network to be active and workable.

2.3.1 Objectives of Architecture Design

WSNs are widely considered as the new emerging technology underpinning the different applications. Because of their characteristics, WSN proposes numerous development challenges to make the sensor nodes. However, before any of the challenges can be properly addressed the design and architecture of WSN must be considered [37]. The WSN has to be designed and implemented and it should have flexible mechanisms with means for their efficient and convenient use. In order to do that architecture design goals should be considered. Some important objectives of WSNs architecture design are as follows [35][38]:

Identifying requirements of WSNs application: based on the target application necessities, the quantitative analysis of the application needs to be able to facilitate and meet the accurate design.

Identifying relevant technological trends: technology is growing exponentially with the help of microelectronics development. A WSN is known to be a heterogeneous and complex system. In such a complex system it is essential to consider the design cost and constrains to find the best fit for a WSN with maximum power optimization based on the desired application.

Optimised design: sensor nodes are resource constrained. Therefore, it is significant to design the network in such an optimised way that maximum utilization of the sensor can be done with minimum use of resources.

Design techniques and technology: based on existing and upcoming technologies, architecture needs to be designed. Among sensor nodes components a power supply and storage existing technology is considered to be mature technology. But ultra-low power wireless communication, sensors and actuators are being upgraded almost every day and are not yet revolutionary. It is important to identify which technology can be used and which need to be developed in the design phase of architecture.

Qualitative and quantitative analysis: existing technology, components and sensors need to be surveyed to do the qualitative and quantities analysis for effective and functional architecture of WSNs.

2.3.2 WSNs Architecture

WSNs are dynamic and can consist of various types of sensor nodes. The environment is heterogeneous in terms of both hardware and software. The sensor node construction focuses on reducing cost, increasing flexibility, providing fault tolerance. Development process and conserving energy also need to be considered.

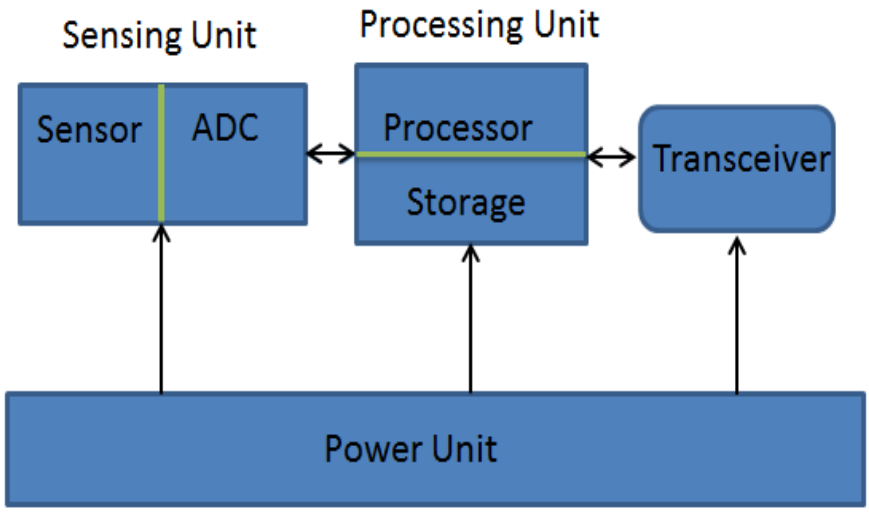

Figure 2-2 : Structure of a sensor node

The structure of sensor node consists of sensing unit (sensor and analog to digital converter (ADC)), processing unit (processor and storage), communication unit (transceiver), and power supply unit [1] [35]. The major blocks for a sensor node can be shown in Figure 2-2. Concise descriptions of different units are as follows:

Sensing unit: It is composed of a collection of different types of sensor which is needed for measurement of different phenomenon of the physical environment. Sensors are selected based on their application. Sensor's outcome is an electric signal which is normally analog. Therefore, an analog-to-digital converter (ADC) is used to transform the signal to digital to communicate with the microcontroller.

Processing unit: It consists of a processor (microcontroller) and storage (RAM). In addition, it has operating systems as well as a timer. The responsibility of the processing unit includes collecting data from various sources then processing and storing. A timer is used to do the sequencing for the processes.

Communication unit: It uses a transceiver which consists of a transmitter as well as a receiver. Communication is performed through the communication channels by using network protocols. Based on the application requirements and relevance in order to build a stable communication it normally uses a suitable method such as radio, infrared or optical communication.

Power unit: The task of the power unit is to provide the energy to the sensor node for monitoring the environment at a low cost and less time. The life of the sensor depends on the battery or power generator which is connected to the power unit. Power unit is required for an efficient use of the battery.

When the knowledge about the structure of a sensor node is acquired, it is necessary to further check and understand the communication architecture of WSNs. The communication architecture of a WSN is slightly different from the conventional computer communication and computer network. The major entities that build up the communication architecture are [35][39]:

- The sensor node objectives are to make discrete, local measurements of phenomena surrounding these sensors, forming a wireless sensor network by communicating over a multi-hop wireless medium, and collect data and rout data back to the user via a sink or a base station.
- The sink (Base Station) communicates with the user via a suitable communication method such as internet, satellite, Wimax, WiFi, 3G or 4G. It is located near the sensor field or well-equipped nodes of the sensor network. Collected data from the sensor field routed back to the sink by a hop to hop infrastructure.
- Phenomenon expressed by related physical parameters, which is an entity of interest to the user to collect measurements about specific phenomenon. This phenomenon sensed and analysed by the sensor nodes of a WSN.

The communication architecture is normally classified in different layers. In order to get the maximum efficiency with limited resources and low overhead a WSN does not adhere as closely to the layered architecture of OSI model of conventional network.

Nevertheless, the layered model is useful in WSNs for categorizing protocols, attacks and defenses. In contrast to the traditional seven layers in an OSI stack the WSN layers are reduced to the five in a TCP/IP stack, which includes the physical layer, data link layer, network layer, transport layer and application later. Figure 2-3 shows the communication protocol model of wireless sensor network [1].

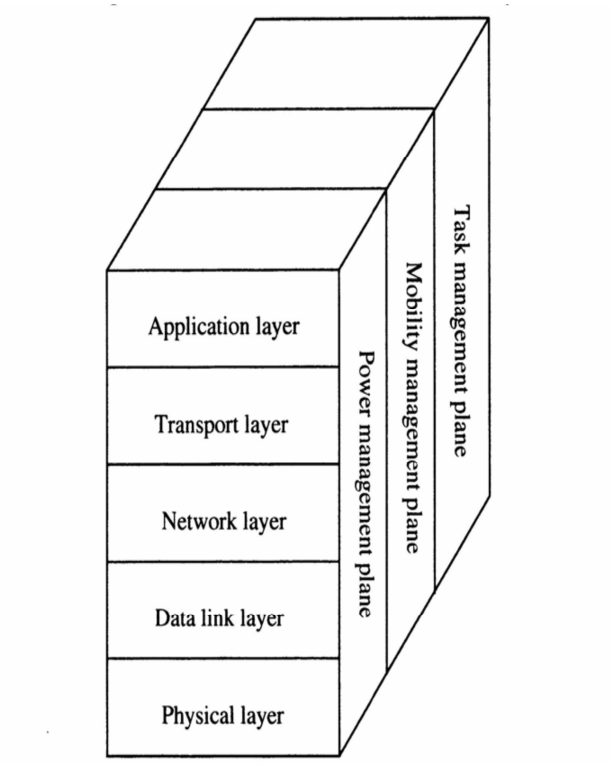

Figure 2-3 : Protocol stack of WSNs

The physical layer is responsible for frequency selection such as carrier frequency generation which corresponds to checking RFID data list to make sure the task, signal detection, modulation, and data encryption are running well. The data link

layer is concerned with the media access control (MAC) protocol. Since the wireless channel is normally affected by the noise and sensor nodes may be changing the location, the MAC protocol at the data link layer has to be power-aware and should have the capability of minimizing the collisions [40]. The network layer manages the routing data supplied by the transport layer or between the nodes. Whereas the transport layer is able to maintain the data flow if the WSN's application requires that. Various types of application can be implemented in the application layer depending on the physical environmental sensing.

Orthogonal to the five layers, Akyildiz et al. [1] defined three management plans named power, mobility and task management as shown in Figure 2-3. These plans are responsible for monitoring the power, movement and task distribution among the sensor nodes. These management plans help the sensor nodes to coordinate sensor tasks and minimize the overall power consumption.

2.4 Protocols of WSNs

WSNs are designed to carry out various tasks which are underpinned by several protocols. This section are going to discuss some major related protocols for WSNs. Routing protocols of WSNs are inspired by ad hoc networking for some similarities in their characteristics [41]. Moreover, WSNs have some specific properties such as coverage cast traffic profile, strong energy constrain, densely deployed high number of nodes [42][43]. Thus, it is necessary to take special care for WSNs. There are different ways to classify the sensor networks routing protocols. According to Ochirkhand [42], the classification of routing protocol can be divided into four categories: Flooding based routing, Probabilistic routing, Location based routing and Hierarchical routing, as shown in the Figure 2-4

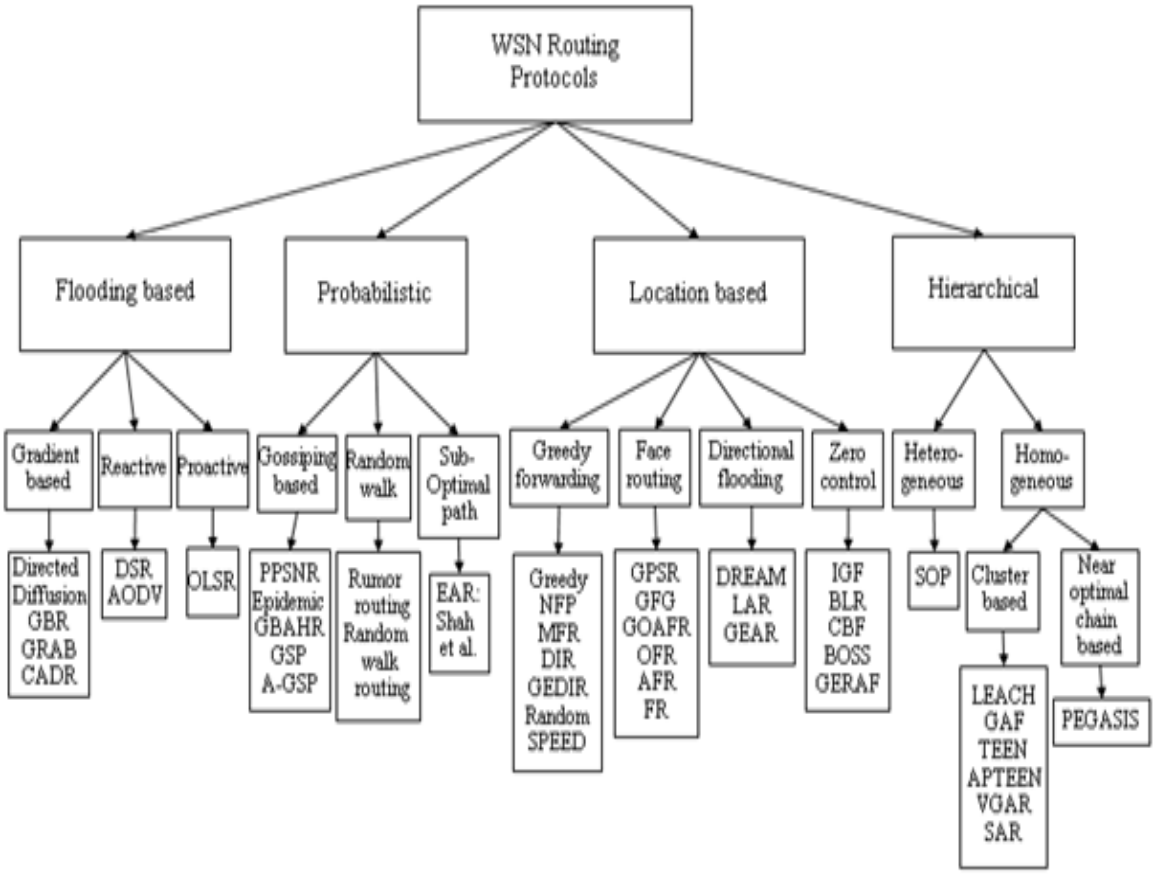

Figure 2-4 : Routing protocols of WSNs

Flooding based routing is a static algorithm which uses flooding mechanism to discover routs. In flooding based protocol every incoming packet is sent out on every outgoing line except the one it arrived on [44]. Flooding based generates infinite number of duplicate packets unless some measures are taken to damp the process. Probabilistic routing chooses the next hope using a dynamically assigned probability or random choice making their behaviour non-deterministic [42]. The location based routing protocols uses geographical location information to guide routing discovery and maintenance as well as data forwarding, enabling directional transmission of the information and avoiding information flooding in the entire network [45][46]. Each node needs to know its destination, its own location and the location of the neighbour. Hierarchical routing is based on hierarchy among the nodes [42] when a larger amount of resources is necessary to take care or a routing table becomes enormous and makes routing impossible. The idea of hierarchical routing suggests that routers should be divided into

regions, with each router knowing all the details about how to route packets within its own region, but knowing nothing about the internal structure of other regions. Most of the routing protocol is shown in Figure 2-4. The list of a few popular routing protocols for wireless sensor networks below [47].

- Direct diffusion
- GBR (Gradient Based Routing)
- AODV (Ad hoc On-Demand Distance Vector)
- GPSR (Greedy Perimeter Stateless Routing)
- LEACH (Low Energy Adaptive Clustering Hierarchy)

2.5 Applications of WSNs

The August 1999 *Business Week* has identified WSNs as one of the most important technologies for various applications in the 21st century [48]. They can be deployed on the ground, in the air, under water, on bodies, in vehicles, and inside buildings to measure different phenomenon based on the sensor nodes classifications. The existing applications can be categorised under some main general headings based on the sensor taxonomies [19] [49 -52].

- Military applications (e.g. Battlefield monitoring, Border surveillance)
- Environmental monitoring (e.g. Animal tracking, Flood detection)
- Commercial or human centric applications (e.g. Vehicle tracking, Patient monitoring)
- Robotics (e.g. Monitoring equipment and automation)

To get the maximum efficiency from any application and security sensor node selection is important. In the next section introduces the existing hardware platform for sensor nodes.

2.6 Existing Hardware Platform

There are a number of hardware platforms for WSNs. The experts from the Commonwealth Scientific and Industrial Research Organisation (CSIRO) found that the best performance of a sensor network comes from adapting the nodes and communication methods to the local environment and the application [53]. The list of a few main prototype and commercial motes/sensor nodes available in the market to choose from is bellow, which help to select the hardware of WSNs in order to develop effective security mechanisms.

- BTnode (use Atmel ATmega 128L Microcontroller)
- Mica (use ATmega 103 Microcontroller)
- T-mote (use Texas Instruments MSP430 Microcontroller)
- IMote (use ARM core 12 MHz Microcontroller)

2.7 Network Security

Recently, the world is becoming more interconnected with the advancement of semiconductor devices which drive faster Internet and new networking technology in smaller devices. Personal, commercial, military, and government information on networking infrastructures worldwide is increasing every day [54]. Hence, to secure any information in a network the security issue became a major concern both in wired and wireless networks.

Wired and wireless networks may achieve the same goal but they are not the same at the technical level. Thus, the security mechanisms are different in wireless networks because of the nature of wireless communications. Wired networks connected via Ethernet normally are reasonable secure for the

communication media by its nature as its dedicated connection. Whereas, wireless communications require security configuration to prevent anyone within the transmission range of the router, switches and bridges from connecting to the network as the transmission media is shared [55]. Thus, malicious deeds can easily happen, such as hacking of networks.

WSN is another paradigm of wireless networks. A highly distributed network indicates that it is possible to work autonomously in a harsh environment. For example, a large number of sensors are deployed to monitor specific phenomena. Considering the application for environment monitoring WSNs are organised in two structures based on underlying topology: (i) flat and (ii) hierarchical [56]. Based on the application the topology can decided. In flat structures all sensor nodes have essentially same role to perform. Hierarchical structures assign different roles to sensor nodes; it is done by crusting the network.

During the data collection and communication any significant or insignificant role performed by the low cost sensor nodes needs to transfer the meaningful information to the sink, thus security provisioning is essential. To develop the security mechanism for WSNs, it is necessary to understand threats, security requirements, challenges as well as the types of attacks that involve in WSNs.

2.7.1 Threats in WSNs

The wireless network transmission medium has a broadcast nature. Hence, it is more susceptible to security attacks compared with the traditional wired network. In wireless sensor networks, nodes can be deployed randomly in the hostile environment so an adversary can easily attack the targeted WSNs [57]. The security of WSNs can be investigated in different perspectives. This work formulate a threat model that distinguishes two major types of attacking classes [58 – 61] namely, (i) based on attacker's location, and (ii) based on attacker's

strength. In this research, the work focused on the internal attacks of a WSN. In order to clarify all those mentioned terminologies, the definitions are described below:

Attacks based on attacker's location: Based on knowledge and privileges of the attacker, attacks can be categorized as insider (internal) and outsider (external) depending on whether the attacker is a legitimate node of the network or not [62]. Attacks can also be classified as passive and active attacks.

Internal attacks: When a legitimate node of the network acts abnormally or illicitly it is considered as an internal attack. It uses the compromised node to attack the network which can destroy or disrupt the network easily. An adversary by physically capturing the node and reading its memory can obtain its key material and forge network messages. Having access to legitimate keys can give the attacker the ability to launch several kinds of attacks, such as *false data injection* and *selective reporting*, without easily being detected. Overall, insider attacks constitute the main security challenge in wireless sensor networks; that is why all of this research focusing this direction, which will be demonstrated in the following Chapters.

External attacks: This attack is defined as the attack performed by a node that does not belong to the network. Obviously, the attacker node does not have any internal information about the network such as cryptographic information.

Passive attacks: The attack does not have any direct effect on the network as it is outside the network. Passive attacks are in the nature of eavesdropping, or monitoring of packets exchanged within a WSNs when the communication takes place over a wireless channel. This type of attack does not create any interruption in communication process. An attacker can inject useless packets to drain the receiver's battery, or it can capture and physically destroy nodes.

Usually authentication and encryption techniques prevent such attackers from gaining any special access to the network.

Active attacks: This type of attack involves disruption of the normal activity of the network. It can do information interruption, modification, traffic analysis, and traffic monitoring [63]. Active attacks are jamming, impersonating, and denial of servicing and message replay.

Attack based on attacker's strength: Attackers may use different types of devices to attack the targeted network; these devices have different computation power, radio antenna and other capabilities. Two common categories have been identified by Karlof and Wagner [59] including laptop-class and mote-class attackers.

Laptop class: To launch an attack, attackers may have access to powerful devices such as faster CPU, larger battery power, bigger memory space, high-power radio transmitter or a sensitive antenna. This hardware device allows a more broad range of attacks which are more difficult to stop. Their goal may be to run some malicious code and seek to steal secrets from the sensor network or disrupt network normal functions. For example, Harting *et. al.* demonstrated how to extract cryptographic keys from a sensor node using a JTAG programmer interface in a matter of seconds [64].

Mote-class: Attackers have accessed one or more sensor nodes with the same or similar capabilities like the sensor node deployed in the network. They may try to jam a radio link, but only in the sensor node's immediate vicinity. However, these attacks are more limited since the attackers try to exploit the network's vulnerabilities using only the sensor's node capabilities.

2.7.2 Generic Security Requirements

The characteristics of WSNs lead a challenge to provide reasonable security to a network. The ultimate security requirement is to provide confidentiality, integrity, authenticity, and availability of all messages in the presence of resourceful adversaries. In the case of internal attack the compromised node uses the legitimate network parameters to attack the network [65]. In order to provide the reasonable security in WSNs all messages must maintain minimum security requirements. The standard requirements to provide security in a WSN are discussed as follows [66]:

Confidentiality: An adversary can choose any node to eavesdrop as long as it is within the radio range due to the signals are transmitted over the opened channel. It is a threat for the data confidentiality as the attacker may gain the cryptographic information and take the information away.

Authentication: To determine the legitimate node and whether the received data has come from the authorized sending node or not. Authentication is one of the key issues for a security.

Integrity: Information moving through the network could be altered or tampered by others. Infect integrity is the description to trust the received information from the network.

Freshness: To save the network from the replay packets it is needed to ensure that the received data is fresh and unused.

Secure management: It is necessary to manage the distribution of cryptographic keying material in the network

2.7.3 Security Challenges

The critical goal of WSNs security is to protect the wireless sensor networks from any types of attack. The different application scenarios presented in the earlier section point out that WSNs may have very different properties. Thus, considering the generic security requirements and application scenario the algorithm is developed to secure a WSN. The major properties that made the security mechanism challenging in WSNs are resource constraints, operational environment and unreliable communication [66], which are discussed below.

Resource constraints: it is commonly assumed that sensor nodes are highly resource constrained. For an example, the Berkeley MICA2 motes and TMote mini, are presented in Table 2-2 [67][68]. Thus, security protocols for WSNs must be executable based on the available hardware and especially must be very efficient in terms of energy consumption and execution time.

Table 2-2 : Sensor Platforms

Characteristics	Mica2	TMote mini
RAM	4(Kbytes)	10 (Kbytes)
Program Flesh Memory	128 (Kbytes)	48 (Kbytes)
Maximum data rate	76.8 (Kbps)	250 (Kbps)
Power Draw: Receive	36.81 (mW)	57 (mW)
Power Draw: Transmit	87.90 (mW)	57 (mW)
Power Draw: sleep	0.048 (mW)	0.003 (mW)

Operational environment: in most WSNs the operational environment is always assumed to be unattended or even hostile. Since sensor nodes are usually not assumed to be physically protected by some tamper resistant hardware, an adversary is able to physically attack and compromise the nodes. The attackers are not only capable of physically damaging the device, but they can also alter device characteristics and security mechanisms to send out data readings of their choice. Once a WSN is in control, the attackers can do whatever attackers wanted to the node, such as altering the node to listen to information about the network, inputting malicious data or performing a variety of attacks.

The above vulnerability can be enhanced by the absence of any fixed infrastructure. In particular, there is no central controller to monitor the operation of a network and identify attack attempts. Thus, even if security mechanisms are deployed, an adversary is able to participate in a network since it has access to all data [42], such as, cryptographic keys stored on the node can be obtained. Thus, security protocols should be able to operate when the sensor nodes are compromised, which prevents cooperating nodes from taking corrective measures against their corrupt neighbours so that they continue to rely on the fake information being fed to them.

Unreliable Communication: Certainly, the very nature of the wireless communication medium, which is inherently insecure, poses another threat to WSNs security. Unlike wired networks, where a device has to be physically connected to the medium, the wireless medium is open and accessible to anyone. Therefore, any transmission can easily be intercepted, altered, or replayed by an adversary. The wireless medium also allows an attacker to easily intercept valid packets and inject malicious ones.

Moreover, the unreliable transmission in wireless channel may result in damaged packets. If packets meet with others in the middle of transfer, conflicts will occur and the transfer itself will fail. Such a weakness can be exploited by an

attacker, with a strong transmitter, who can easily produce interference or jamming [69][70] of the network. In addition, wireless multi-hop communication can introduce great latency in a network, which makes it difficult to achieve synchronization among sensor nodes. Compromised nodes may be part of a route, enabling them to modify forwarded messages.

2.7.4 Nature and Types of Internal Attacks

Simple sensor nodes are usually not well physically protected because they are cheap and are always deployed in open or even in hostile environments where they can be easily captured and compromised. Hence, from a compromised node an adversary can extract sensitive information, control the compromised node, and let the compromised node service the attacker (adversary). The attacks are involved in corrupting network data, disconnect network communication. The compromised node has the following characteristics [71][72]:

- Compromised node is usually reprogrammed by the attacker by injecting malicious code. Thus, the compromised node seeks to steal information from the sensor network or disrupt the network normal functionality.
- Compromised node uses the same radio frequency as the other normal sensor nodes so that it appears to communicate with normal nodes.
- Deployed normal nodes are authenticated and participate in the sensor network. Since secure communication in sensor networks is encrypted and authenticated using cryptographic keys, compromised nodes with the secret keys of a legitimate node can participate in the secret and authenticated communication of the network.

The compromised nodes are dangerous in a WSN, due to the fact that an adversary can easily access information from compromised nodes such as the cryptographic information, by which a compromised node can gain trust of other

sensors. This type of attack is difficult to break or stop. That is why it has become a challenging task to secure WSNs from internal attacks.

In many applications, the data obtained from the sensing nodes needs to be kept confidential and it has to be authentic. In the absence of security a malicious node could intercept private information, or could send false messages to nodes in the network. In order to make further investigation for the attacks related to WSNs, in the corresponding sub-sections discussed and took a closer look at some popular attacks. The major attacks this work want to highlight are: Denial of Service (DoS), Worm hole attack, Sinkhole attack, Sybil attack, Selective forwarding attack, Spoofed and altered, or Replayed routing information, Hello flood attack and Flooding attack. Based on the Open System Interconnect (OSI) model the attacks can be tabulated in Table 2-3 [71] [73 - 75]:

Table 2-3 : Layer Based Security Attacks

Layer	Attacks
Physical layer	Jamming, Tampering, Sybil Attack
Data Link Layer	Collision, Sybil Attack, Spoofing and Altering Routing Attack, Replay attack
Network Layer	Internet smart attack, Sybil Attack, Blackhole Attack, Spoofing and Altering Routing Attack, wormhole attack, selective forwarding attack, Hello Flood Attack.
Transport Layer	Flooding Attack, Desynchronisation
Application	Spoofing and Altering Routing Attack, False Data Injection,

Denial of Service (DoS) attacks:

This attack is an explicit attempt to prevent the legitimate user of a service or data. The common method of attack involves overloading the target system with requests, such that it cannot respond to normal traffic. As a result, it makes the system or service unavailable for the user. The basic types of attacks are: Jamming, Tapering, Collision, Homing and Flooding.

If a sensor network encounters DoS attacks, the attack gradually reduces the functionality as well as the overall performance of the wireless sensor network. In WSNs several types of DoS can be performed in different layers which are tabulated in Table 2-4 [74] [76 - 78].

Table 2-4 : Layer Based DoS Attacks

Layer	Attacks
Physical layer	Jamming, Tampering
Data Link Layer	Collision, Exhaustion
Network Layer	Misdirection
Transport Layer	Desynchronisation
Application Layer	Path Based DoS

The discussed attacks are linking some terminologies that are defined as follows [75 - 78]:

Jamming: In this attack the attacker attempts to jam the frequencies of the radio used for communication between the nodes in the network. An adversary

may use a few nodes in strategic positions to effectively jam most of the communications inside the network. In essence, an attacker needs only a few nodes in order to disseminate a large network.

Tampering: Because of the nature of wireless sensor networks, an adversary could easily get physical access to the sensor nodes. This may enable an attacker to compromise sensor nodes in a DoS like manner.

Collision: In This attack a node induces a collision in some small part of a transmitted packet. The packet will then fail the checksum check, because of the changes brought on by the collision, and the receiver node will then ask for a retransmission of the packet.

Exhaustion: This attack is one of collision attacks which take them a bit further damage WSNs. A malicious node may conduct a collision attack repeatedly in order to exhaust the power supply of the communicating nodes.

Misdirection: In this attack a malicious node that is part of a route can, instead of dropping packets, quite simply send them on a different path which does not exist in a route to the destination. The malicious node may do this for certain packets, or all packets.

Desynchronisation: It can disrupt an existing connection between two end points. Adversary transmits a lost packet with bogus sequence numbers or control flags to degrade or prevent the exchange of data.

Path Based DoS: [11]An adversary overwhelms sensor nodes by flooding a multi-hop end to end communication path with either replayed [79] or injected false message to waste secure energy resources.

Wormhole Attack:

In this attack, a malicious attacker receives packets from one location of a network, forwards them through the tunnel and releases them into another location[80][81]. Hence, the attacker is able to send packets, routing information, ACK etc., through a link outside the network to another node somewhere else in the same network. The malicious node can achieve the faith of the neighbour node as a legitimate node [82]. This can also confuse routing mechanisms that rely on knowing distances between nodes. A wormhole attack can be used as a base for eavesdropping, not forwarding packets in a DoS like manner, and altering information in packets before forwarding them.

Sinkhole Attack:

An attacker gains attraction to surrounding nodes with respect to the routing algorithm through a compromised node [83]. It prevents the base station from obtaining complete and correct data. In this attack, a malicious node advertises a zero cost route through itself. If the routing protocol in the network is a "low cost route first" protocol, like distance vector, other nodes will chose this node as an intermediate node in routing paths. The neighbours of this node will also chose this node in their routes, and compete for the whole bandwidth. \

Sybil Attack:

The concept of Sybil (or multiple-identity) attacks was first proposed by Douceur in P2P networks [84], and it is defined as a single node has multiple identities to disrupt the accordance among the entities and physical devices in a networks. This attack poses a serious threat for damage to WSNs' integrity. A malicious node forges multiple identities to mislead the network and let the neighbour

nodes to believe that they have several trusted neighbours [85]. This attack targets fault tolerant schemes such as distributed storage, dispersity, multipath routing and topology maintenance. This attack is especially confusing to geographic routing protocols as the adversary appears to be in multiple locations at once. Hence, it is very hard to identify the position as the malicious node could appear in more than one place at the same time.

Selective Forwarding Attack:

In a simple form of selective forwarding attack, malicious nodes try to stop the packets in the networks by refusing to propagate any further [86]. Another variance of selective forwarding attacks is to delay packets passing through the nodes, creating the confused routing information between sensor nodes [87].Even though the protocol is completely resistant to the sinkholes, wormholes, and the Sybil attack. If a compromised node is strategically located near the source or a base station there is a significant probability of including the compromised node on a data flow to launch this type of attack. However, such an attacker takes the risk that neighbouring nodes will conclude that it has failed and take other route.

Spoofing Attack:

In an open nature, the characteristics of a wireless medium are easy for any malicious node to monitor the communications to find the layer-2 Media Access Control (MAC) addresses of the other entities in this network. This can have a serious negative impact on the network performance as well as facilitate many forms of security weaknesses.[88] In this attack, a malicious node is able to create routing loops, wormholes, black holes, partition the network, by spoofing, altering or replaying routing information.

Hello Flood Attack:

Hello Flood Attack is introduced in [59]. The malicious nodes broadcast *hello* messages to announce their presence to the neighbouring nodes. The node receiving the message assumes that the malicious node is within its range or a neighbour. An attacker with a high powered antenna can convince every node who receives "hello" in the same network which means this node is their neighbour. Hence, the malicious node can deceive other nodes to believe that a normal node is malicious. Nodes at a large distance from the attacker will be sending their messages to an out-of-reach malicious node that can disrupt the network by simply decreasing traffic load and make communications in a state of confusion. This form of attack is specifically designed against routing protocols that are dependent on localised information.

All of the above mentioned attacks have the common purpose that is to compromise the integrity or workability of the network that they attacked. In order to ensure the network functions as originally designed a network needs to be saved internally and externally. This research work will need to understand the internal attacks of WSNs. As mentioned in the paragraphs, this thesis highlights internal attacks and discussion about external attacks is outside the scope of this thesis even it is equally important. For meeting up security the next sub section presents related suggestions for this research focus, internal attacks.

2.8 Suggestions in the Literature to Secure WSNs from Internal Attacks

WSNs use multi-hop communication to increase network capacity. In multi-hop routing, messages may traverse many hops before reaching their destinations. However, simple sensor nodes are usually not well physically protected because they are cheap and are always deployed in open or hostile environments where

they can be easily captured and compromised. An adversary can extract sensitive information, and control the compromised nodes. Even though let those nodes service for the attackers. Therefore, when a node is compromised, an adversary gains by accessing to the network and can produce malicious activities. The attacks are involved in corrupting network data or even disconnecting a major part of the network. To address the protection from internal attacks the following paragraphs discussed some existing mechanisms.

Zhang *et al.* in [89] proposed a scheme that is the first and most cited work on intrusion detection in wireless ad hoc networks. Architecture is investigated for collaborative statistical anomaly detection which provides protection from attacks on ad hoc routing on wireless MAC protocols, or on wireless applications and services. Conceptually this architecture is divided into different modules. Firstly, *Data collection*; this module gathers streams of real time data form various sources. Secondly, using *the local detection engine* to analyze the local data traces gathered by the local data collections for evidence of anomaly and they suggested the statistical method for this stage. Detection methods need border data that requires collaboration among the nodes to be used in the cooperative detection. Intrusion responding actions are provided by both the local response and global response modules. Finally, *secure communication* module provides a high confidence communication channel to the agents. The advantage of this architecture is that they used statistical analysis. This architecture can only work on routing. For internal attack detection, it is not sufficient as it only focuses on routing protocol.

Silva *et al.* in [90] proposed the first work on the rule based intrusion detection scheme to detect many different kinds of attacks in different layers. In this scheme three main phases are involved. Phase 1: data acquisition phase, in which the messages are filtered by the monitoring node to be analyzed. Phase 2: the rule application phase, which is responsible for applying the predefined rule to the stored data from the previous phase. Phase 3: the intrusion detection

phase, which compares the case between the numbers of raised failures produced from the rule application phase with a predefined number of occasional failures. If the total number of raised failures is higher than the predefined threshold, the alarm is raised. According to Xie *et al.* [91], this scheme presents a good framework to a class of rule-based intrusion detection. But, the main drawback of this scheme is the ambiguity in determining the number of monitoring nodes and the way of choosing them, such as how to make sure that the way of selection will cover the entire network. In addition, this scheme is restricted to some types of attacks, as the decision is made based on only a simple summation of the rule.

Karlof and Wagner discussed attacks at the network layer in [59] and mentioned altered or replayed routing information and selective forwarding, node replication, Sybil attacks or black-grey-sink holes, and HELLO flooding. They suggested suitable countermeasures that can help to mitigate the attack. The solution discussed is prevention based and to secure the routing. This solution does not focus on the internal attacks or compromised node specifically.

Staddon *et al* [92] proposed a way to trace the failed nodes in wireless sensor networks at the base station assuming that all the sensor measurement will be directed along the sinker based on a routing tree. The first step of the protocol enables the base station to learn the topology of the network. During the execution of many well-known route-discovery protocols, nodes learnt the identities of their neighbours. To convey this information to the base station, each node simply attaches a little bit of information about its neighbours to each of its measurements. In a constant amount of time the base station has adjacency information for the entire network and hence can construct its topology. Once the base station knows the node topology, the failed nodes can be efficiently traced using a simple divide-and-conquer strategy based on adaptive route update messages. In this work the sinker has the global view of the network topology and can identify the failed nodes through route update message.

Watchdog like techniques were discussed in [93], [94] and [95]. The purpose of the watchdog mechanism is to identify a malicious node by overhearing the communication of the next hop. This technique can detect the packet dropping attack by letting nodes listen to the next hope nodes broadcasting transmission. From their research papers, each sensor node has its own watchdog that monitors and records its one hop neighbours' behaviours such as packet transmissions. When a sending node *S* sends a packet to its neighbour node *T*, the watchdog in *S* verifies whether *T* forwards the packet toward the Base Station (sink) or not by using the sensor's overhearing ability within its transceiver range. In this mechanism, *S* stores all recently sent packets in its buffer, and compares each packet with the overheard packet to see whether there is a match. If yes, it means that the packet is forwarded by *T* and *S* will remove the packet from the buffer. If a packet remains in the buffer for a period longer than a pre-determined time, the watchdog considers that *T* fails to forward the packet and will increase its failure tally for *T*. If a neighbour's failure tally exceeds a certain threshold, it will be considered as a misbehaving node by *S*. But, multiple watchdogs need to work collaboratively in decision making. A reputation system is necessary to provide the quality rating of the participants. This method will fail when the following matters happened, *ambiguous collision, receiver collision, limited transmission power, false misbehaviour,* and *partial dropping*.

A machine learning based approach is proposed by Huang and Lee in [96] for anomaly detection. They developed a cross feature analysis *anomaly detection* approach that explores the co-relation between each feature and all other features for the nodes. This is conducted by computing classifiers from a training set composed of normal nodes. An intrusion alarm is raised if the correlation between the features does not match those of the classifiers. The machine learning procedure assumes a large number of features being monitored from sensor behaviours, and the availability of normal sensors as the training data set, both of which are difficult to obtain considering the restrained sensor resources and dynamic networking behaviours.

Pires *et al.* in [97] presented a solution to identify malicious nodes in wireless sensor networks through detection of malicious message transmissions in a network based on the signal strength. A message transmission is considered suspicious if its signal strength is incompatible with its originator's geographical position. The geographical position is determined by the Global Positioning System (GPS). In this work they showed how to detect HELLO flood attack and the wormhole attack by comparing the energy of the received signal and the energy of the same observed signal around the network. This is work use GPS for location detection. Thus, this system can only be implemented in the *line of sight* scenario and restricted with HELLO flood attack and the wormhole attack. In addition, the signal strength can be infected by other factors such as interference from electronic devices, environmental factors for example, rain and storm.

Branch *et al.* in [98] studied the in network outlier. They developed an algorithm that has the following properties: (i) it is generic – suitable for many outliers detection heuristics; (ii) it works in networks with a communication load proportional to the outcome that is the number of outliers reported; (iii) it is robust with respect to data and network change; (iv) the outcome is revealed to all of the sensors. In other words, in this method each sensor in the network first identifies the outliers based on the neighbourhood data. Then exchange the decision with neighbours to achieve the global set of outliers. But this method does not work well for small system with limited samples. In addition, it is expensive as well as it depends on the neighbour collaboration.

Support Vector Machine (SVM), based on techniques for internal attack detection in sensor data was proposed in [99]. This technique uses one-class quarter-sphere SVM to reduce the effort of computational complexity and locally identify outliers at each node. The sensor data that lies outside the quarter sphere is considered as an outlier or internal attack. Each node communicates only summary information (the radius information of sphere) with its parent for global outlier classification. This technique identifies outliers from the data measurements

collected after a long time accumulation within a window. The technique also ignores spatial correlation of neighbour nodes, which makes the results of local outliers inaccurate. The main drawback of SVM-based techniques is their computational complexity and hard for the choice of proper kernel function.

Zhang *et al.* in [100] proposed a distance-based technique to identify *n* global outliers in snapshot and continuous query processing applications of sensor networks. This technique reduces communication overhead as it adopts the structure of aggregation tree and prevents broadcasting of each node in the network [98]. Each node in the tree transmits some useful data to its parent after collecting all the data sent from its children. The sink node then roughly figures out top *n* global outliers and floods these outliers to all the nodes in the network for verification. If any node disagrees on the global results, it will send extra data to the sink node again for outlier detection. This procedure is repeated until all the nodes in the network agree on the global results calculated by the sink node. This technique considers only one-dimensional data and the aggregation tree used may not be stable due to the dynamic changes of network topology.

Recently Game theory is commonly used to analyze wireless sensor networks with selfish/attacker nodes [101]. Reddy and Ma studied *game theory* [101][102], Reddy *et al.* presented in *zero-sum game* which may find malicious sensor nodes in the forwarding path only [101]. *Zero-sum game* method needs to maintain a certain level of energy. The proposed *game theory* method in [102] not only improves the security of WSNs, but also reduces the cost caused by monitoring sensor nodes and prolongs the lifecycle of each sensor node. However, the method does not consider the effects of the compromised entity of the sensor nodes, which can discard normal packets or not transfer normal packets in WSNs.

The fuzzy logic based intrusion detection approach has been widely used and studied such as by Chi and Moon [103][104]. In [103], node energy, transmission rate, lists of the neighbour nodes and transmission errors are taken as the

measurement parameter. Based on the four features the base station will take the decision about the denial of service (DoS) attacks. In [104], the approach is to detect sinkhole attacks in directed diffusion based sensor networks based on the radio and transmission radius. In a sinkhole attack, there will be extra message traffic in area compare to the normal traffic and the transmission radius will be smaller. The fuzzy logic system will produce detection value based on the normal traffic and transmission radius. The decision will be taken based on the predefined threshold and the fuzzy rules need to be set according to the symptoms with extensive study of sinkhole attack. The main drawback of the fuzzy logic is that it needs the manual settings of rules in this method.

Stetsko *et al.* implemented an intrusion detection system which employs the neighbour based detection technique [12]. They designed the system to work on the TinyOS operating system running the Collection Tree Protocol. They used selective forwarding, jamming and hello flood attacks to evaluate the system. In their work, the nodes collaboration among themselves is efficient as at the same time it generates the communication overhead. This method suffers from false alarm for packet dropping and sending rate. Moreover, this method does not consider the power consumption rate related to the network performance.

A collaborative and decentralized approach for an intrusion detection system was proposed by Lemos *et al.* [105] to detect node repetition attacks. In this scheme some special nodes, called monitors, will be responsible for monitoring the behaviour of neighbour nodes in turn by using predefined rules. The malicious activities evidence discovered by each monitor will be shared and correlated with the purpose of increasing the accuracy in detection of intruders. This paper also claimed that it was a robust method with two layers of protection. The drawback of this method is the monitor nodes could be compromised, which were not to be considered. It is a rule based approach that has an assumption of the parameters that need to be made. Therefore, it has inflexibility for applications.

An integrated approach is proposed by Wang *et al.* [106]. This method can provide the system to resist intrusions, and process in real-time by analysing the attacks. The Integrated Intrusion Detection System (IIDS) includes three individual Intrusion Detection Systems (IDSs): (i) *Intelligent Hybrid Intrusion Detection System (IHIDS)*; (ii) *Hybrid Intrusion Detection System (HIDS);* and (iii) *Misuse Intrusion Detection System (MIDS)*. The goal is to raise the detection rate and lower the false positive rate through misuse detection and anomaly detection. Finally, a decision-making module is used to integrate the detected results and report the types of attacks. The advantage of this method is that it is suitable for design of detection modules based on capabilities and probabilities of getting compromised. The use of back propagation method in building the detection module implies high computational complexity. In addition it has low detection accuracy and high false alarm.

Bankovic *et al.* proposed a machine learning solution for anomaly detection [107]. This combines with the feature extraction process that tries to detect temporal and spatial inconsistencies. It uses the sequences of sensed values buy nodes and the routing paths used to forward these values to the base station. The data produced in the presence of an attacker are treated as outliers and detected using clustering techniques. The techniques are coupled with a reputation system to isolate the compromised node. A drawback of this system is that the system cannot use all the information of the nodes since the nodes cannot share their bad experiences such as dropped packets. This is particularly detrimental since learning from one's own experience in this scenario comes at a very high price.

A dual-weighted trust evaluation in a hierarchical sensor network is proposed by Hyun *et al.* [108]. In this method sensor nodes report their readings to a forwarding node for aggregation. Each sensor node need to assigned two trust values. They are increased or decreased depending on its reading and the aggregation results at the forwarding node. An updating policy is developed to

keep misdetection rates low while achieving high malicious node detection rate for a wide range of fault and related probabilities. But, the performance of a malicious node detection scheme depends on the correctness of the aggregation and results at the forwarding node, since wrong decisions at the node lead to inaccurate management of trust values. The resulting false alarms might waste energy and thus shorten the network lifetime.

Znaidi *et al.* addressed the problem of nodes replication attacks [109]. They first introduced a hierarchical distributed algorithm for detecting node replication attacks using a Bloom filter mechanism and a cluster head selection. The algorithm works as soon as the network is built upon a cluster head selection mechanism generating a three-tier hierarchy. In this method, each cluster head exchanges the member nodes identifications (IDs) through a Bloom filter with the other cluster heads to detect eventual node replications. However, this method needs to employ additional clustering algorithm and the authors presented only a theoretical discussion on the boundaries.

Garofalo *et al.* in [110] proposed a new intrusion detection system architecture designed to ensure a trade-off between different requirements. It is high detection rate obtained through decision tree classification. By which the energy saving is obtained through light detection techniques on the motes. But, in this method the power consumption is high, it is not resilient to node failures as it uses a tree classification, with a long delay to send the data to the base station, data overhead is high and it is costly.

A few papers also addressed pollution attacks in internal flow coding systems employing special crafted digital signatures [111] [112] or hash functions [113] [114]. Recently some papers discussed preventing the internal attacks by related protocols [115] [116] but looking at protocol does not protect the WSN completely.

2.9 Proposed Method

The previous section discussed the suggested methods in the literature to secure WSNs from internal attacks. From the discussion it can be seen that most of the method has some serious drawbacks such as consume more energy, low accuracy raise false alarm, inflexibility for applications. In order to overcome the problems in the suggested methods in literature, this research developed a multistage mechanism to protect wireless sensor networks. Multi-agent, pairwise key, cosine similarity is discussed and implemented. To check for the misbehaving node by considering neighbour nodes output is investigated. The evidence theory, Dempester-Shafer theory, has been carefully discussed and observations of neighbour nodes parameters are considered for judgment. The benefit of this method is that it can deal with uncertainty efficiently. Markov Chain and Monte Carlo - Metropolis Hasting, has been introduced and it works in real time by constricting the sample chain and computes the changes and come up with an acceptance ratio of the node.

A detailed explanation of the above proposed methods is presented in Chapters 3, 4 and 5.

2.10 Summary

This Chapter has provided an overview of WSNs and a consideration of the importance of security in WSNs technologies, and the problems imposed by computation under such a resource limited WSNs' environment. The limited resources and sparse availability for WSN nodes frequently offer an advantage to the internal attacker, whether in regard to low-power transceivers being swamped with interference on the wireless channel, or cryptography is being restricted to small key spaces due to energy constraints. This Chapter carefully presented the most popular attacks to the WSNs and highlighted this thesis

focus in this thesis, which focuses on internal attacks and discussed some suggestions in the literature to deal with internal attacks in WSNs and outline the proposed method. The coming Chapter shall investigate misbehaviour in WSNs to build up a base for this thesis coming discussions in the rest of the Chapters.

Chapter 3 Misbehaviour Identification

In wireless sensor networks the nodes are deployed to perform specific tasks. If any node does not act or perform the task that the node was supposed to do as designed during the deployment phase, it is known as misbehaviour. Misbehaviour in WSNs can happen in different ways such as, packet dropping and packet modification, skewing of the network's topology or creating fictitious nodes [117][118]. As discussed in Chapter 2, WSN nodes are easy to compromise; thus an attacker may launch various types of attacks to disrupt the network communication via a compromised node. Misbehaved nodes interrupt the normal functionality of a WSN. In order to keep functional WSNs at design level it is necessary to identify the misbehaviour of the node, which will save the network from internal attacks. Chapter 2 summarise that suggested mechanisms to save WSNs from internal attacks have some shortcomings. This chapter and the next two chapters present the new mechanism to save WSNs from internal attacks. This Chapter introduce the system model of this research WSN. Then it introduces multi-agent, pairwise key and cosine similarity for misbehaviour or internal attacker identification. The simulation results show the mechanisms are working well to identify misbehaviour of the node.

3.1 System Model

The WSN system under consideration consists of an area of interest where region wise detection requirements are provided by the end user. The work model the area of interest as a two dimensional grid Ω of $N_x \times N_y$ points to cover the sensor field, in which $N_x \times N_y$ is representing the matrix for the grid. The sensor nodes deployment is done randomly with a single sink. Each sensor node sends the data periodically to the sink in the WSN. The sink is located within the network. The work assumes all sensor nodes and the sink are time synchronized. The sink is aware of the network topology, which can be achieved by requiring nodes to report their neighbouring nodes right after deployment. The model also assumes the network sink is trustworthy and free of compromise; this is because the sink has more powerful and bigger capacity computing, self-protection. Compromised sink discussion is out of the scope of this thesis. It is the case that the work assumes the adversary cannot successfully compromise regular sensor nodes during the short topology establishment phase after the network is deployed.

The following sections discuss the sensing model and explain how to prevent the network from internal attacks by identifying them by using multi-agent, pairwise key and cosine similarity.

3.2 Sensing Model

In WSN there are two common sensing models that are popular, namely binary detection model and the exponential detection model. Both models share the assumption that the detection capability of a sensor depends on the surrounding obstacles such as rain; background noise such as magnetic field of earth. Following the paper [119] the notations have used for the case in binary detection

model, the probability of temperature detection $P_d\ (\tau, \varepsilon)$ is given as in equation 3.1.

$$P_d(\tau, \varepsilon) = \begin{cases} 1 & if\ d(\tau, \varepsilon) \leq r_d \\ 0 & if\ d(\tau, \varepsilon) > r_d \end{cases} \quad (3.1)$$

where r_d is the detection radius and $d(\tau, \varepsilon)$ is the distance between the measurement point (at which the temperature to be measured) "τ" and the sensor location "ε" on a two dimensional space. If the distance between the sensor location and measurement point (detection point) is greater than r_d, the measurement of the temperature at measurement point is not possible by the sensor. However, if measurement point is within the detection radius, the temperature will always be measured. The exponential model is a more realistic model, where the probability of detection corresponds to as in equation 3.2

$$P_d(\tau, \varepsilon) = \begin{cases} e^{-\alpha d(\tau, \varepsilon)} & if\ d(\tau, \varepsilon) \leq r_d \\ 0 & if\ d(\tau, \varepsilon) > r_d \end{cases} \quad (3.2)$$

where α is a decay parameter that is related to the quality of a sensor or the surrounding environment. In the exponential model of equation 3.2, even if a measurement point is within the detection radius, there is a probability that the temperature will not be measured, which means it will be missed. As this model is closer to the realistic case, The thesis uses this model in following discussion.

Based on the Linear Shift – Invariant (LSI) system as described in [120], it is possible to mathematically quantify the individual sensor measurement characteristic on the grid using miss probabilities, $P_{miss} = 1 - P_d$, where P_d is the probability of detection of temperature. The collective miss probability $M(x, y)$ is the product of all sensor probabilities to miss the measurement of temperature at

the point (x, y). $M(x, y)$means the probability of a measurement point at grid point (x, y) is being missed by the all the neighbouring sensors. Considering the two points (x, y) and (i, j) on the grid, the Euclidean distance between them is $d\left((x, y), (i, j)\right)$. Hence, the collective miss probability is shown in Equation 3.3.

$$M(x, y) = \prod_{(i,j)\in\Omega} P_{miss}\left((x, y), (i, j)\right)^{u(i,j)} \tag{3.3}$$

where $u(i, j)$ is a step function that demonstrate the presence or absence of the data detection (temperature) at the location (i, j) on the grid, in this research definition when there is no detection. $u(i, j)$ can be expressed as in equation 3.4.

$$u(i, j) = \begin{cases} 1, if\ there\ is\ a\ data\ detected\ \ at\ (i, j) \\ 0, if\ there\ is\ no\ data\ detected\ at\ (i, j) \end{cases} \tag{3.4}$$

Hence, the work considers $P_{miss}\ (\ldots)^0$=1, taking the natural logarithm of the both sides in equation 3.3, it is possible to have as in equation 3.5.

$$ln\ M(x, y) = \sum_{(i,j)\in\Omega} u(i, j)\ ln\ P_{miss}\left((x, y), (i, j)\right) \tag{3.5}$$

where $M(x, y)$ is so-called the *overall logarithmic miss probability* at the point (x, y). Following the LSI model, the work define a function $b(x, y)$, which can be expressed as

$$b(x, y) = \begin{cases} \ln P_{miss}\left((x, y), (0,0)\right), & d\left((x, y), (0,0)\right) \le r_d \\ 0, & d\left((x, y), (0,0)\right) > r_d \end{cases} \tag{3.6}$$

The overall logarithmic miss probabilities for all points on the grid can be arranged in a vector M of dimension $N_x N_y \times 1$ that corresponds to equation 3.9 as shown below:

$$\overrightarrow{M} = [M(x,y), \forall (x,y)]^T \quad (3.7)$$

$$\overrightarrow{u} = [u(i,j), \forall (i,j)]^T \quad (3.8)$$

$$\text{and } \overrightarrow{M} = B\,\overrightarrow{u} \quad (3.9)$$

If there are some compromised nodes distributed in WSNs, they can be detected by miss probability count, as it will not be able to collect the actual data from the actual point. The following sections explains how to prevent the network from internal attacks and how to identify them using multi agent, pairwise key and cosine similarity.

3.3 Multi-Agent Based

A multi-agent system (MAS) is a group of agents able to interact and cooperate in order to reach a specific objective. In MAS agents are characterized by their autonomy, their ability to interact with other nodes. They can learn, plan future tasks and able to react and to change their behaviour according to the changes in their environment.

In this research WSN environment, the MAS manages a set of sensors of WSNs sensing field with agents. MAS considers a range of sensors with agents to protect WSN from compromised node. Before the work establish MAS in the next few paragraphs will first investigate about the signal transmission, the construction of sensor node, target node and sink node in WSN.

In order to communicate among agent, sensor and sink, it is necessary to check the transition among their signal to each other through wireless channel. In fact in the real world, the transmitted signal in a WSN will suffer from several noises, caused by the complex and hostile environment [121]. The transmission problem directed us to focus on signal to noise ratio (SNR) because the compromised node can take the information of SNR as an opportunity to attack the network. Following the discussion [119], a typical wireless sensor network environment can be modelled as shown in Figure 3-1. From the top view of the Figure 3-1 it can be said that, sensor nodes detect the transmitted signals generated by the target nodes over a *sensor channel* and forward the detected information to the sink nodes over a *wireless channel*.

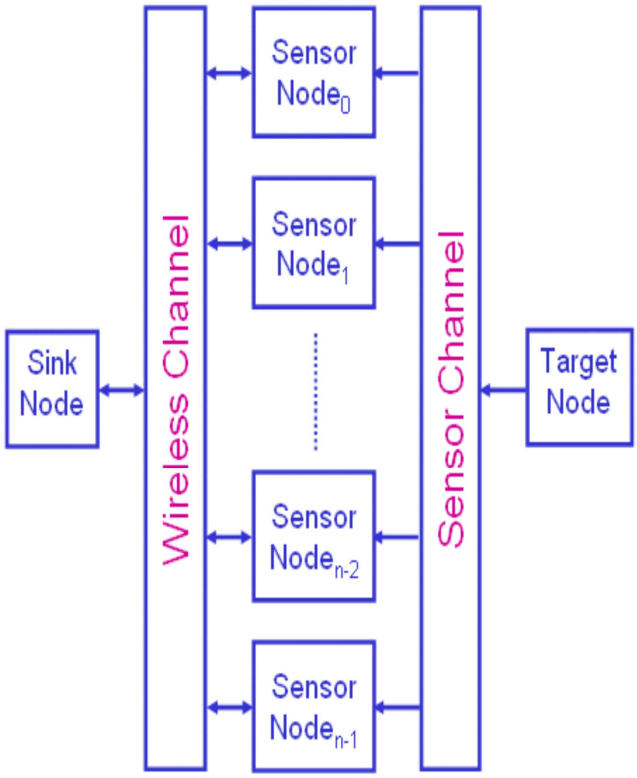

Figure 3-1 : The model of a typical wireless sensor network environment

In Figure 3-1, the operation of the nodes shown by considering a fairly simple event-to-sink transport protocol, namely a stimulus is periodically generated by a

target node and propagated over a sensor channel. It is noted that a target node can only send data packets over a sensor channel. The neighbouring sensor nodes, which are within the sensing radius of the target node, will then receive the stimulus over the sensor channel. The neighbouring sensor nodes, which are within the sensing radius of the target node as shown in equation 3.2, will then receive the stimulus over the sensor channel. To implement MAS with the channel it is necessary need to have a good understanding of the construction of target, sink and sensor nodes. In the following paragraphs, I shall discuss the construction of the nodes.

Considering an example, a sensor node may either forward data packets as soon as they detect the stimuli, or process them first, which is computing the average values measured within a period of time say a few minutes, and then forward processed data to the sink node. Any networking processing mechanism can be implemented in the sensor application layer [122]. As the sink node may not be in the vicinity of a sensor node, communication over the wireless channel is usually multi-hop as well as one hop. This implies sensor nodes are capable both send and receive data packets over the wireless channel. In WSNs nodes are normally divided into three different nodes, namely target node, sink node and sensor node. Their constructions can be shown in Figures 3-2, 3-3 and 3- 4 respectively.

The information received at the sink node over the wireless channel can be further analysed by a control server and/or a human operator. The sink node has to send commands or queries to the sensor nodes, based on the content of the information sink node received. In addition, sink nodes should be capable of both sending and receiving data packets over the wireless channel. Sensor node includes both the energy-producing components such as battery and the energy-consuming components such as CPU and radio.

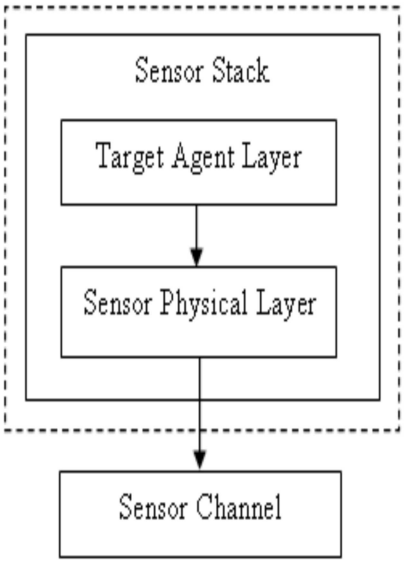

Figure 3-2: Construction of a target node

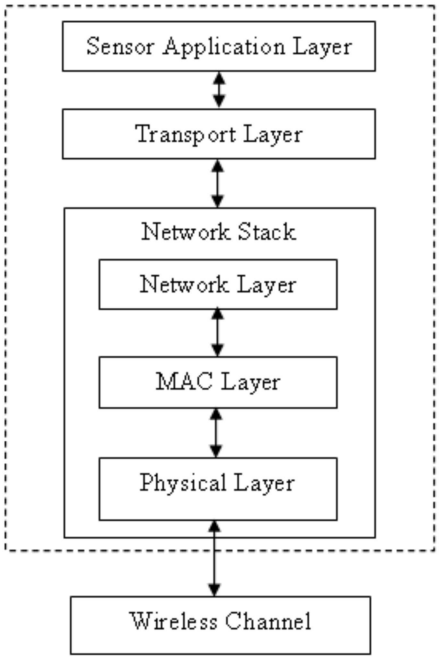

Figure 3-3: Construction of a sink node

The sensor function is subject to the energy efficiency of the sensor. For example, the energy incurred in handling a received data packet is dictated by the CPU,

and the energy incurred in sending and/or receiving data packets is dictated by the radio. Both the CPU and radio can be in one of several different operation modes [123]. For example, the radio can be in one of the following operation modes: *idle*, *sleep*, *off*, *transmit* or *receive*. The amount of energy consumed by an energy consumer (sensor) depends on the operation mode.

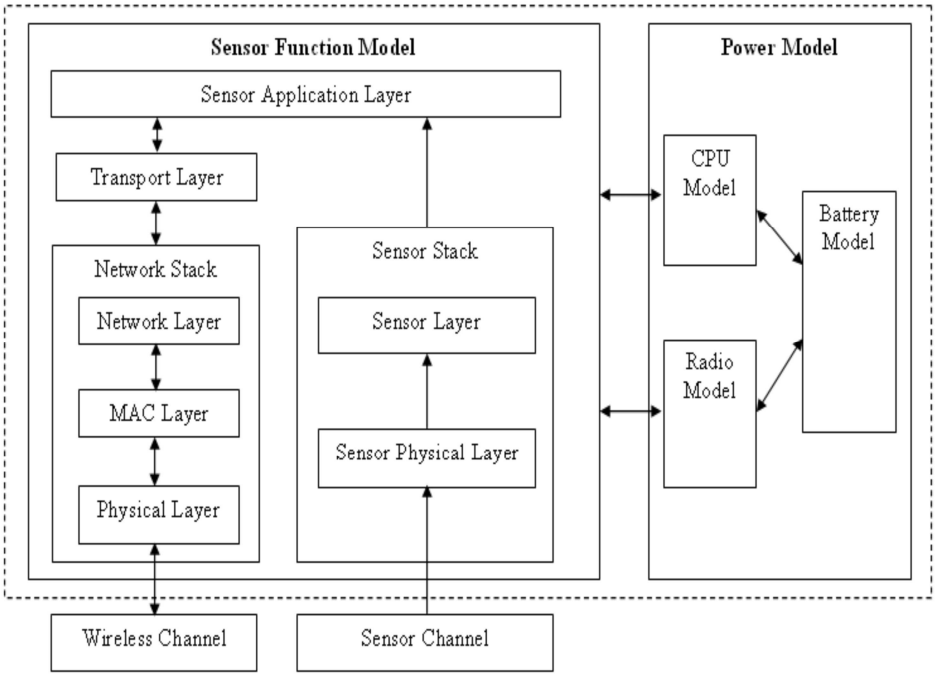

Figure 3-4: Construction of a sensor node (dashed line)

In order to save energy and increase efficiency in the system set the collaborations among the sensor nodes and sink node by their MAC layer, which makes the timing control to realize the sink node status, i.e. sleeping or active. Therefore, the sink node only opens at a special time period; the other time is in the sleeping state and ignoring any incoming signals such that sink node can not only save energy but also protect the network from the internal attacks. Hence, the possibilities of attacking the node will significantly decrease due to the "closed state" within the sleeping time period. The implementation procedure can be described as below, Algorithm 3-1.

Algorithm 3-1: Multi-agent Implementation

1. Input: p_d^{req} (detection requirement), K (number of available sensors), α(decay parameter), r_d (detection radius), $d(t,s)$ (the distance between the target's position) and B.

2. Output: u (deployment vector) and highest SNR time and location.

3. initialization: $k = 0, u = 0$

4. while $k \leq K$ do

5. find set of grid points with unsatisfied detection requirements $\{i: p_d^k(i) \geq p_d^{req}(i)\}$

6. Find the SNR and index j_{max}, where $j_{max} = max_{index}\,(u_k)$.

7. Update the deployment vector (i.e. $u(j_{max}) = 1$)

8. Calculate $M_k = Bu$

9. Calculate time t_k

10. Increment number of sensors in the grid: $k = k + 1$

11. End while

For the fixed parameters in a network, the simulations operation will give the highest SNR with the time and location information. Thus, the highest SNR can enhance the decision for the sink node status, sleeping or opening time period.

Figure 3-5 shows the multi-agent system for WSN. Based on the Figure 3-1 the work established the multi-agent system to protect a WSN from internal attacks with agents, which are sensor node agent, time and location calculation agent, target agent, sensor channel agent, wireless channel agent and sink node sleeping and opening agent. In the Figure 3.5 the dashed arrow is receiving data and solid arrow is receiving data. Each agent owns a set of rules that allow it to decide on its action. For example, the sink node sleeping and opening agent

controls the sinks sleeping and opening time. The work use MAS to control highest SNR occurring time and location to control the receiver (sink). Hence, it can protect the WSN by minimizing the risk of receiving the data from an attacker; in particular it can protect WSN from internal attack.

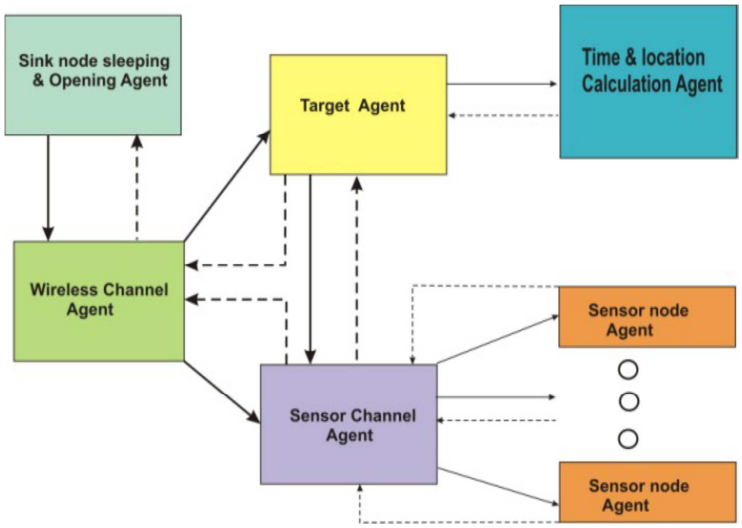

Figure 3-5: Multi-agent system to control sink node sleeping and opening time

This model in fact implements the new bisect algorithm based on the constructions of target sink and sensor nodes. The opening time of the sink node, opening window time period, will be the time t_0, the highest time plus 2RTT time, which can be expressed as in equation 3.10:

$$W_{open} = t_0 + 2RTT \qquad (3.10)$$

where RTT is defined as round traveling time for the network, which is the time it takes for a data packet to travel from node to the sink. 2RTT is used for connection establishment and request between the sink and node. The t_0 is normally sitting on the middle position of the window size, in which the sink starts to share the timing information, to ensure the received signal arrived at the sink node.

J-Sim is used to do this simulation as provides an object-oriented definition of (i) target, sensor and sink nodes, (ii) sensor and wireless communication channels, and (iii) physical media such as seismic channels, mobility models and power models (both energy-producing and energy-consuming components). The simulation used some fixed parameters, the sensing radius is 200 m, attenuation factor α = 2, moving $rate\ \beta = 30\ m/s$ and transmission power = 0.2818 W (for a 260 m transmission range). the simulation was done for 1000 seconds. One unit transmission rate is 10 bits per second. Figure 3-6 shows the simulation result with 2 target nodes with one transmission unit. The 3 target nodes with one unit transmission rate result is in Figure 3-7. In the Figure 3-8 simulation result, the work used two target nodes with three unit transmission rate. This simulation result clearly showed the highest SNR occurring with different transmission rates, target nodes numbers. Sink node operating window is shown in Figure 3-9

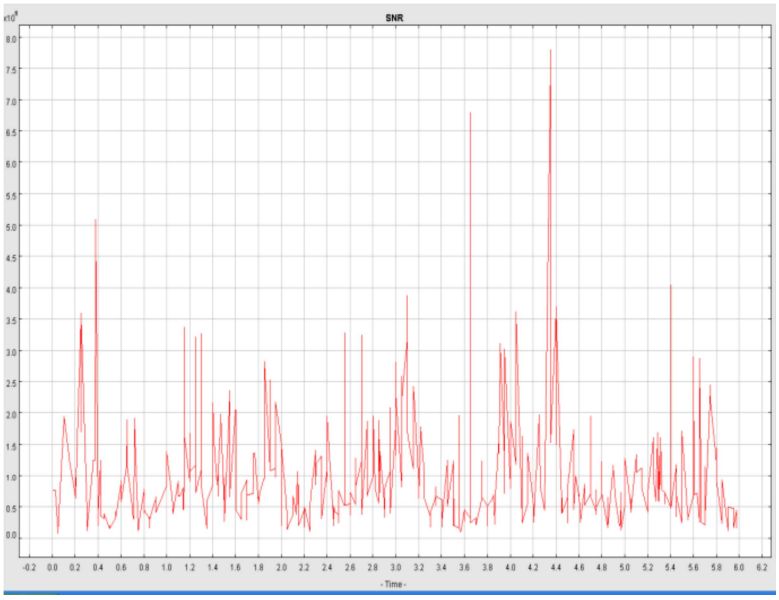

Figure 3-6: Simulation result with two target nodes and the transmission rate is one unit (normalized)

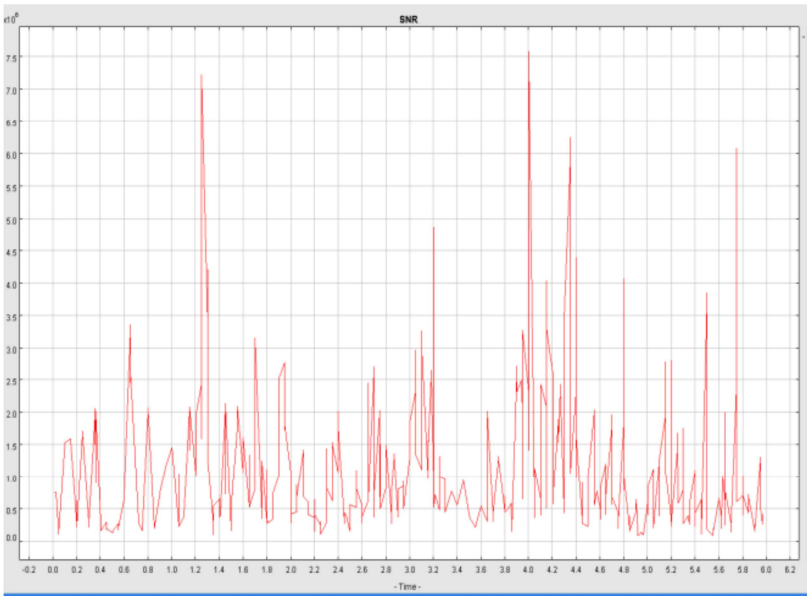

Figure 3-7: Simulation result with three target nodes and the transmission rate is one unit (normalized)

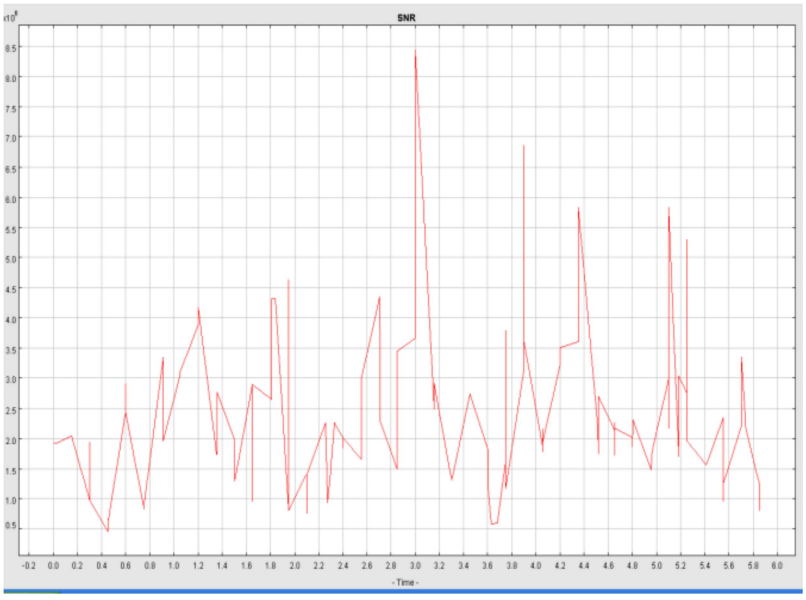

Figure 3-8: Simulation result with two target nodes and the transmission rate is three units (normalized)

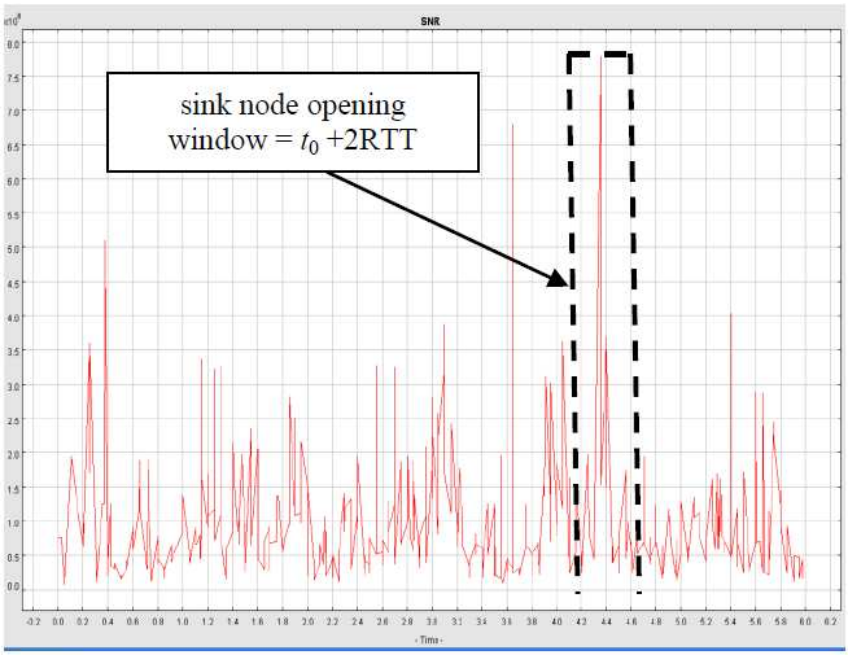

Figure 3-9: Sink node opening window

The Work has used different situations to check the algorithm listed in the Table 3-1. From the Table 3-1, it can be seen that if the network parameters are fixed, it is possible to control the occurring time and location of the highest SNR of the target node via control the transmission rate. Therefore, the sink node sleeping time and opening time can be controlled. This research have run the network with different transmission rates with design time period (with high confidential condition), which makes it difficult for the attacker to know the opening time period of the sink node window.

Table 3-1: The highest SNR with different cases

Case		Time Taken (Normalized)		
	Rate 1	Rate 2	Rate 3	Rate 4
Case 1	4.3	3.1	4.25	2.81
Case 2	4.04	3.91	4.02	2.87
Case 3	5.3	2.71	2.81	2.93
Case 4	4.0	3.9	4.25	3.84
Case 5	4.0	5.7	1.27	4.2

Now let us have a closer look at the number of the target nodes. In fact, if the target nodes increase, the fixed network will face a window size problem, as the target nodes increase, the window will have to keep opening to make sure no useful information is lost, with no sleeping time for the sink. This algorithm will fail. This case at the current stage is only for if (i) rate was fixed with one sink or (ii) there is a limited time for the whole network.

3.4 Pair Wise Key Based

Random key pre distribution is one of the approaches proposed in the literature for addressing security challenges in resource constrained wireless sensor networks (WSNs). The idea was first introduced by Eschenauer and Gligor [124], in which sensor nodes are assigned a random subset of keys from a large key pool before deployment of the network . In tested WSN system every node will have a pairwise key with its immediate neighbours respectively. This will be used to secure distribution of the cluster keys to its direct neighbour nodes and secure the transmission of data. After deployment, two neighbouring nodes can establish a pairwise key between them as long as they have at least one common key (any key which is same for both nodes) in the key ring.

In o system assume nodes within communication range establish pairwise keys with its one-hop neighbour just after deployment. This is known as initial key. Since at start up no nodes are compromised, adversary cannot learn any initial pairwise keys. When the nodes are deployed, each node, is pre-distributed an initial key, K_I. A node, u with $u \in \Omega$, can use K_I and one way hash function H_f to generate its master key, K_u as in equation 3.11:

$$K_u = H_f(ID_u, K_I) \qquad (3.11)$$

During the neighbour discovery stage, node u broadcast a HELLO message within its identification (ID) and waits for a response from its nearest neighbour v. The response message from the neighbour node v consists ID of node v and a message authentication code (MAC) to verify the node v's identity. Then, node u is able to authenticate node v since it can compute MAC value with its master key K_v, which is derived as in equation 3.12.

$$K_v = H_f(ID_v, K_I) \tag{3.12}$$

Here the work highlight the identification of v is ID_v in the above equation 3.12. Then node u broadcasts (for broadcast the work use the notation $*$) an advertisement message (ID_u, $Nonce_u$) which contains a nonce (a bit string used only once), and waits for other neighbor v (here $u \neq v$) to respond with its identity. Following the previous research [125], the process will be as follows:

$$u => *: ID_u, Nonce_u \tag{3.13}$$

$$v => u : ID_v, \mathrm{MAC}_{K_v}(Nonce_u | ID_v) \tag{3.14}$$

Therefore, after authentication both node u and node v generate pairwise keys as $K_{uv} = H_f(ID_u, K_v)$. Hence, each node can use these nodes' ID to calculate its one hop-neighbour's key, i.e. $\forall u \in N_1$, where N_1 is the space of one-hop for a fixed node in a targeted WSN. If there is any stranger node, such as the adversaries' node, it will be distinguished by the pair-wise keys from nodes u and v. In the case of packet loss due to the narrow bandwidth or bad channel condition, pairwise key may take longer time to establish the key.

Following the paper [126], it is known that wireless sensor networking ZigBee®/IEEE 802.15.4 solutions, CC2431 includes hardware 'Location Engine'

that can calculate the nodes position via the given RSSI (Radio Signal Strength Indication) and position data of the reference nodes within the network. As described in [126] another factor that affects received signal strength is antenna polarization. A designed small simple antenna produces a linearly polarized radiation. For the linear polarized case, the electrical magnetic (EM) field remains in the same plane as the axis of the antenna. Hence, it is easy to have the bad channel condition and narrow bandwidth. Therefore, it is necessary to consider transmission attributes of the sensor nodes with signal to noise ratio.

Assume that $S_u/N_u = R_u$ for a reference node $u \in N - N_1$, when $N \subset N_{one-hop}$ and $N_1 \subset N$. According to the Z-score method (Z-score method described in appendix I), the Z-score transforms an attribute value based on the mean and standard deviation of the attribute. The Z-score value indicates how far and which direction the value deviates from the mean value of the attributes. The work used the Z-score value with signal to noise ratio, which have the parameters mean $\bar{\mu}_v$, standard deviation $\bar{\sigma}_v$, of the neighbor value and $\emptyset_v$ is the deviation from the normal value:

$$\bar{\mu}_v = \frac{1}{n_1} \sum_{u=1}^{n_1} R_u \tag{3.15}$$

$$\bar{\sigma}_v = \sqrt{\frac{1}{n_1 - 1} \sum_{u=1}^{n_1} (R_u - \bar{\mu}_v)^2} \tag{3.16}$$

$$\emptyset_v = \left| \frac{R_u - \bar{\mu}_v}{\bar{\sigma}_v} \right| \tag{3.17}$$

If $\emptyset_v$ is smaller than the designed threshold, it would be taken as normal case otherwise it would be assumed as an attacker and require checking of the node $u \notin N_1$ and $u \in N$. Following the paper [127], the transmission rate is defined as, T_u as the u −th node in a targeted WSN. Its transmission attribute can be expressed as in equation 3.18.

$$T_u = \frac{T_u^{out}}{T_u^{in}} \tag{3.18}$$

Here, the transmission attribute of node T_u^{out} and T_u^{in} denoted the signal sending and receiving with one hop neighbour for the node. If the transmission attribute does not match the pre defined threshold within a time period it is considered as a compromised node or an internal attacker.

When a compromised node is detected it is important to consider the resiliency of the network. The definition of resiliency [128] is the ability of a network to continue to operate at the designed level in the presence of k compromised nodes, therefore the work assume the threshold for currently targeted WSN being designed as 30% of the total nodes becoming compromised nodes. The work assumes that if 30% of the node is compromised, the network still works as designed in the deployment phase. It may describe those compromised nodes becoming a group denoted as S_q, where q is the $q-$th sick group. The operation for resilience of the WSN is as in equation 3.19.

$$\sum_q S_q \geq 0.3\, N \tag{3.19}$$

Here, N is the number of the sensor node. With this condition, there is an operation needed to disable or isolate those compromised nodes by their locations in the targeted WSN.

The system considers a homogenous WSN with 1024 sensors uniformly distributed in a network area, which is forming the network region $b \times b$ squared field located in the normalized resiliency-degrees against the normalized time units. In order to investigate the interference effects to a WSN, The work take two cases, namely, 32×32 (low density case) and 16×16 (high density case)

squared fields with the fixed sensors in the network. The simulations were running 50 times with the final averaging the data as shown in Figure 3-10, which is the case that "normalized average delivery rate" vs. "percentage compromised nodes."

It is noted that the "scenario 1" in the Figure 3-10 is the chart about the average forward rate ≅55% and the "scenario 2" is the case average forward rate ≅ 32%. At the same compromised node rate the latter case will be more serious than the former. There are two charts: the "scenario 1" is the sensors deployed in the smaller area (16×16) and the same sensors were distributed in the larger area (32×32) in the case of "scenario 2". Due to the crowding sensors will impact each other by the interferences so detection accuracy is affected.

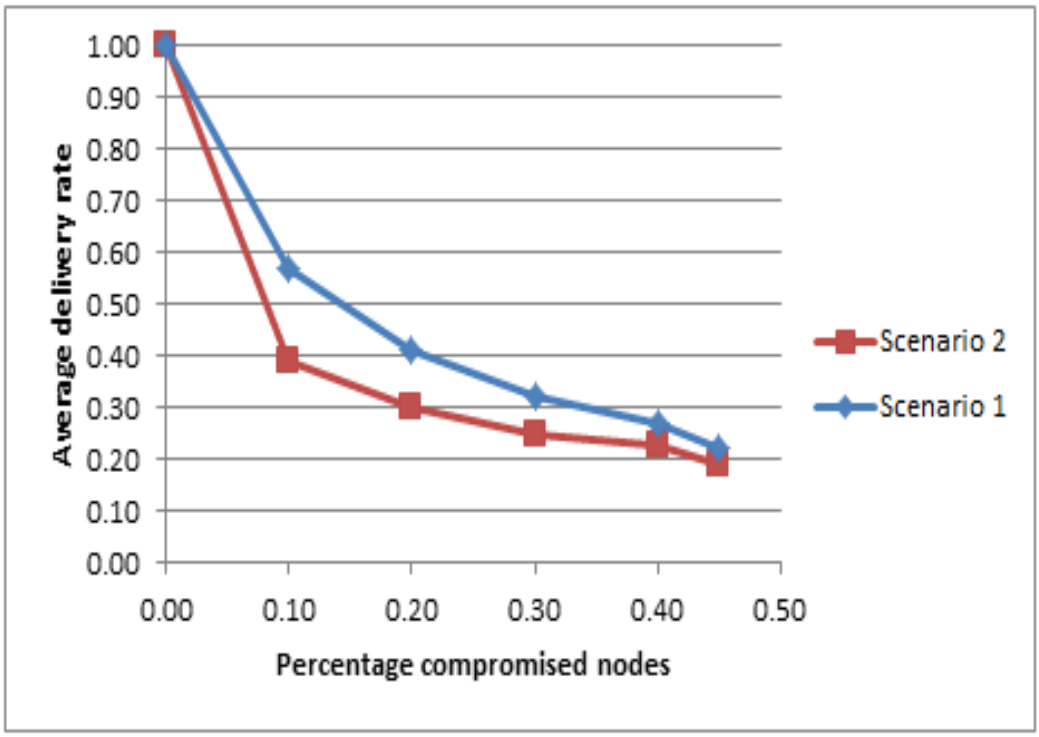

Figure 3-10: Chart of the "normalized average delivery rate" vs. "percentage compromised nodes."

Figure 3-11 shows the situations about "normalized resiliency ration" vs. "the simulation period time"

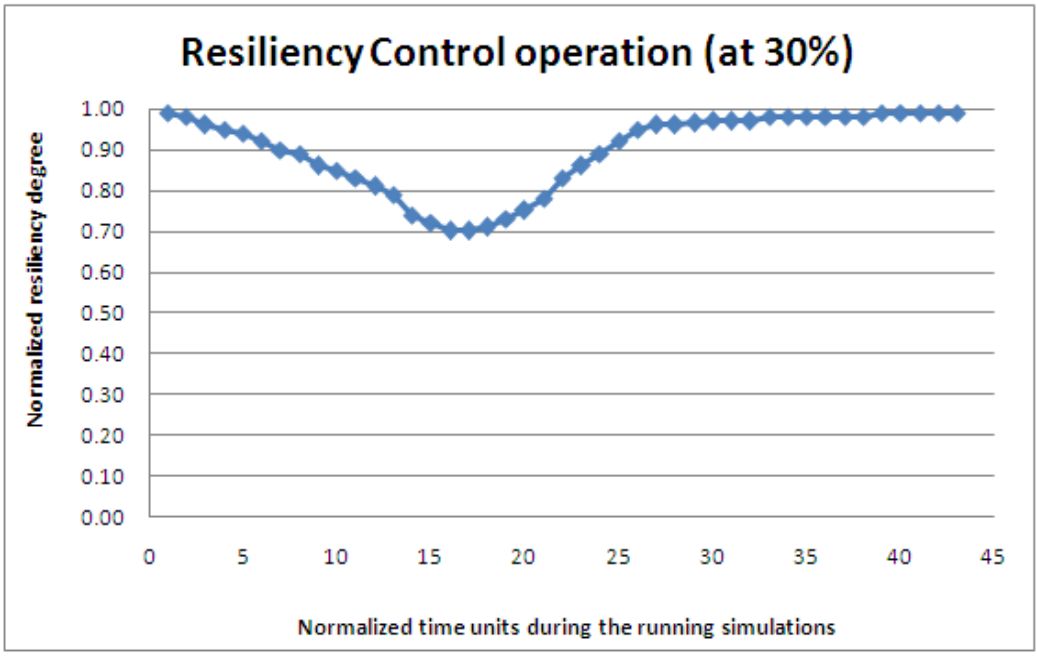

Figure 3-11: Normalized resiliency degree

In the experiments [125], if decreased by 30% the operation is taken to identify the compromised nodes and disable them with their locations. The experiments divided the whole areas into four regions. For each region the experiment designed three beacons (locations known) by which with RSSI get the locations for compromised nodes and then disable them when the "operation is running." From the Figure 3-11 it can be seen that the resiliency is under a reasonable level to be controllable.

When protection of WSN with cryptography protection is done, the compromised node will try to launch he attacks via tampering message, which causes the abnormal traffic. The next section introduces a cosine similarity based method to mitigate this issue.

3.5 Cosine Similarity Based

Although, there are several similarity functions that can be used for feature set comparison of neighbour nodes to find the misbehaved node, such as edit distance, Euclidean's distance, cosine distance, Jaccard's similarity, and generalized edit distance, this research take the belief raised by the researchers in [129], where they showed that the Euclidean distance and cosine distance are two appropriate similarity functions for WSNs to find the compromised node. This is because those distances can easily find the data differences between the nodes.

In order to identify the misbehaviour or abnormal behaviour of a node in a WSN in this research have designed abnormal behaviour identification mechanism (ABIM) based on cosine similarity method that is sensitive to the abnormal event. In the conventional cryptographic way it is not possible to detect the internal attacker because of the unpredictable wireless communication channel [130]. The unreliable channel makes it easy to compromise a node in a WSN and establish an untrustworthy relationship [131] [117]. This research will formulate the cosine similarity based approach and how it works in the system. The fundamental of cosine similarity is dot product; to facilitate the understanding of cosine similarity the next sub-section describes the dot product.

3.5.1 Dot Product

A dot product matches up elements or features in corresponding dimensions of two different vectors. If there is two vectors $\vec{x} = (x_1, x_2, x_3 \dots\dots\dots\dots)$ and $\vec{y} = (y_1, y_2, y_3 \dots\dots\dots\dots)$, where x_i and y_j are corresponding to the vector (feature of the data) and n is the dimension of the vectors. Hence, based on the geometric defilation dot product as in equation 3.20

$$\vec{x}.\vec{y} = \|\vec{x}\|.\|\vec{y}\| \cos\theta \qquad (3.20)$$

Where, $\|\vec{x}\|$ and $\|\vec{y}\|$ is the magnitude of a vector $\vec{x}$ and $\vec{y}$, direction is the direction of the arrow points for x and y . θ is the angle between $\vec{x}$ and $\vec{y}$.

3.5.2 Cosine Similarity

Cosine similarity is a normalized metric, because its values fall in [0,1]. The cosine similarity between two vectors (features) is a measure that calculates the cosine value of the angle between them. In Figure 3-12, the projection of the vectors $\vec{x}$ and $\vec{y}$ is x_1 and y_1 for axis 1 and x_2 and y_2 for axis 2. θ is the similarity angle, α_1 and α_2 is the projection angle for axis 1 and 2 respectively, this research use the shortest distance between the vector and axis for the projection.

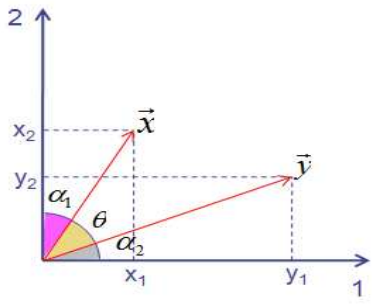

Figure 3-12: The projection of the vectors

The trigonometric and Pythagorean theorems are given in the equations 3.21 to 3.24, to work out the corresponding values.

$$\sin(\alpha + \beta) = \sin\alpha\cos\beta + \cos\alpha\sin\beta \qquad (3.21)$$

$$\cos(\alpha - \beta) = \cos\alpha\cos\beta + \sin\alpha\sin\beta \qquad (3.22)$$

$$\vec{x} = \sqrt{x_1^2 + x_2^2} \tag{3.23}$$

and

$$\vec{y} = \sqrt{y_1^2 + y_2^2} \tag{3.24}$$

From the Figure 3.12 it has,

$\cos(\vec{x}, \vec{y}) = \cos(\theta)$ and $\theta = {}^{\pi}/_{2} - (\alpha_1 + \alpha_2)$

By using equation 3.22 it is possible to obtain

$\cos\theta = \cos({}^{\pi}/_{2})\cos(\alpha_1 + \alpha_2) + \sin({}^{\pi}/_{2})\sin(\alpha_1 + \alpha_2)$

It is noted that,

$\cos({}^{\pi}/_{2}) = 0$ and $\sin({}^{\pi}/_{2}) = 1$

After using equation 3.12 it is possible to have

$\cos\theta = \sin(\alpha_1 + \alpha_2) = \sin\alpha_1 \cos\alpha_2 + \cos\alpha_1 \sin\alpha_2$

By using the sine and cosine definition formulation the equation 3.25 is done

$$\cos\theta = \frac{x_1}{|\vec{x}|} \cdot \frac{y_1}{|\vec{y}|} + \frac{x_2}{|\vec{x}|} \cdot \frac{y_2}{|\vec{y}|} = \frac{\sum_{i=1}^{2} x_i\, y_i}{|\vec{x}||\vec{y}|} \tag{3.25}$$

This metric is a measurement of orientation and not magnitude; it can be seen as a comparison between data on a normalized space because the research is not taking into consideration only the magnitude of each date feature, but also the angle between the data features. The cosine similarity equation is to solve the equation of the dot product for the $\cos\theta$ for the two vectors $\vec{x}$ and $\vec{y}$ as in equation 3.26.

$$\cos\theta = \frac{x_1 \cdot y_1}{|\vec{x}||\vec{y}|} = \frac{\sum_{i=1}^{n} x_i y_i}{\sqrt{\sum_{i=1}^{n} x_i{}^2}\sqrt{\sum_{i=1}^{n} y_i{}^2}} \tag{3.26}$$

So the similarity formula becomes as in equation 3.27

$$\cos(\vec{x},\vec{y}) = \frac{x_1 \cdot y_1}{|\vec{x}||\vec{y}|} = \frac{\sum_{i=1}^{n} x_i y_i}{|\vec{x}||\vec{y}|} = \sum_{i=1}^{n} \frac{x_1}{|\vec{x}|} \cdot \frac{y_1}{|\vec{y}|} \quad (3.27)$$

The next section will explain how the cosine similarity theorem works in the system to find the internal attack.

3.5.3 WSNs Implementation

Assuming a WSN is densely deployed and continuously observed to the phenomenon, the characteristics driving WSN nodes normally encounter the spatio-temporal correlation as discussed in Chapter 2. In the research [66] The research considered the messages generated from the nodes are similar for a defined period with the sampling rate of 0.1Hz (1 message per 10 second). The feature of the sensor node and the expected feature (based on threshold) will be checked with index terms. If the feature of the data and feature query match each other than the node is normal otherwise it is an abnormal node. The concept of implementation is shown in Figure 3-13.

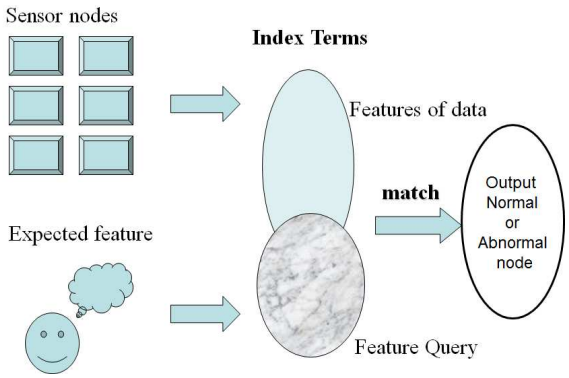

Figure 3-13: Concept of implementation

Considering the limited storage of the sensor it stores minimum information of the message in S with ring architecture. It keeps the record of the latest history of the message from the node data. The ring architecture stores the data circularly to implement easily. In this research consideration, the message m_i consists of the content of the representative message (∂) and the frequency of the messages (f). Therefore, following the previous research [132], the message consists of $m_i = \langle \partial, f \rangle$. The set of the message is shown in the equation 3.28.

$$S = \{m_i | m_1, m_2, m_3 \dots m_{|S_z|}\} \tag{3.28}$$

Here, S is the set that will store the latest message that is sent to the network recently. Suppose if there are 6 messages sent, S will store 6 representative messages. Thus S will become, $S = \{m_1, m_2, m_3, m_4, m_5, m_6\}$. When a new message, m_{new} is sent by the node it arrives at the cluster head, then m_{new} can be authenticated by the similarity function with S. In this research, system temperature of an area is measured by nodes. The research considers the message of the node is temperature. The difference between the detected and average temperature is divergence. If $D^i(m_{new})$ denoted as the divergence between the new and the normal message, it is possible have the set for different cluster can be expressed as equation 3.29 [133].

$$\begin{aligned} D^i(m_{new}) &= \{D^1(m_{new}), D^2(m_{new}), \dots, D^S(m_{new})\} \\ &= \{|m_1 - m_{new}|, |m_1 - m_{new}|, \dots, |m_{|S|} - m_{new}|\} \end{aligned} \tag{3.29}$$

According to the above equation 3.29 the distance measurement is taken, in the cases when messages are defined as “different” from others on the content (Temperature). The difference can be done based on the pre-defined maximum and minimum threshold. The minimum and maximum threshold is set using

Gaussian distribution of the temperature measured; if the data are different from the threshold then it is considered as a new message. This can be done using equation 3.29 with finding the differences from the original messages from the node in a given time. The threshold is defined as the mean of the data set. As Chapter 1 discussed, the attribute of internal attacker or misbehaved nodes can be considered abnormal transmission frequency and data differences. It could be too high or the other way around. In order to identify the differences between the nodes data, this research used k means algorithm within the Euclidian distance [66]. The Euclidean distance measures the dissimilarity between a compromised node or misbehaved node and a neighbour's node. If $D^i(m_{new})$ is not within the threshold, it is considered as new message which is a fake message. For further authentication this research will use the cosine similarity method.

Since the nodes in WSNs are recourse constrained, so a temporal buffer has to be used to accumulate incoming messages from the nodes as a frame of reference. If the fresh messages or the last message is denoted as, l_i and the frequency of fresh message denoted as ω_i. The temporal buffer B can be shown as in equation 3.30, which accumulate several coming messages for the buffer maximum size $|B_s|$.

$$B = \{(l_i, \omega_i)|(l_1, \omega_1), (l_2, \omega_2), (l_3, \omega_3), \dots, (l_{|B_s|}, \omega_{|B_s|})\} \tag{3.30}$$

Based on the cosine similarity method this research computes the similarity between m_{new} and l_i and identifies m_{new} attribute. The cosine similarity can efficiently distinguish m_{new} and l_i based on spatiotemporal correlation. The cosine similarity is shown in equation 3.31.

$$COSIM = \frac{m_{new} \,.\, B}{|m_{new}| \,.\, |B|} \tag{3.31}$$

Where, $|m_{new}|$ and $|B|$ is the magnitude of the message vector $|m_{new}|$ and $|B|$.

If the two messages are similar that came from different nodes then it is considered a normal message, and will be added into S. Otherwise it is considered a false message and the node will be considered as an abnormal node or misbehaved node.

In this method the computation is simpler with smaller latency [134]. The considered parameter for this process is supported by the resources constrained sensor nodes. The algorithm is shown in Algorithm 3-2.

Algorithm 3-2: Cosine similarity implementation

I. Get m_{new}

For $i = 1$ *to* $|S|$

If $MinTh \leq D^{i}(m_{new}) \geq MaxTh$

printf "Good Node"

else go to II

II. for $i = 1$ to B

Execute the equation 3.31

If COSIM $i \leq 0.6$

printf "the node is an internal attacker"

else

Go to step I

end

The simulation result is shown in Figure 3-14. In this simulation this research set the sampling rate 0.1Hz from the 6 minutes observed empirical data and in the case study this work have the calculation for the consign similarity for a one hop neighbour with the abnormally behaved node.

m_{new} = {6 5 6 5 6 4}

B = {1 0 3 2 1 5}

Cosine Similarity (COSIM) = 36 / (13.19) *(6.32)

= 36 / 83.42

= 0.43

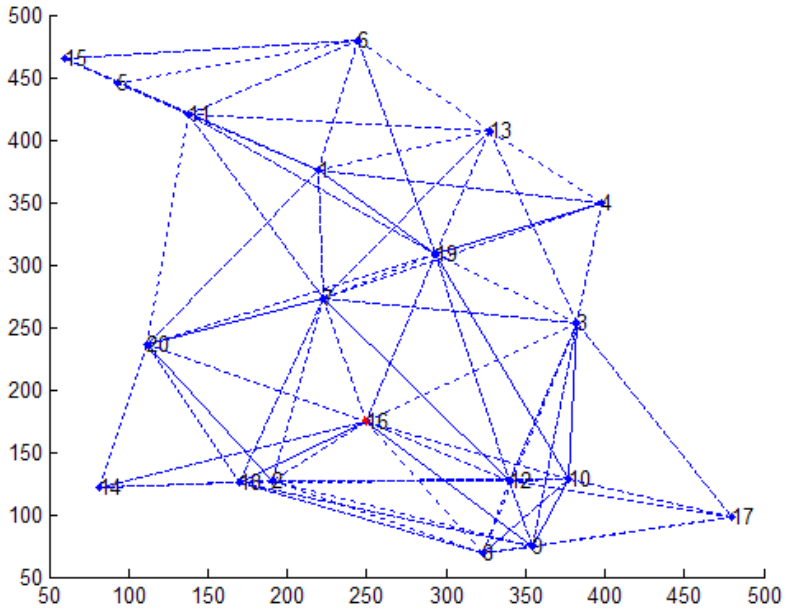

Figure 3-14: Sensor field with abnormal node detection

This research used the case study empirical data to convert into simulation. The simulation was done in a small area 500 m × 500m with 20 sensors. As in Figure 3-14 it can be seen that mode number 6 is compromised in this case.

3.4 Summary

In this Chapter, this research has identified the misbehaviour of a node to be the internal attacker in a WSN. A multi-agent is used to control the timing of the sending and receiving of data by sink. Based on the highest SNR occurring in time and location this work can control the sink, so that can prevent the network receiving the data from internal attacks. The pair wire key method identifying the misbehaviour of the node based on the designed threshold was investigated. Finally, in this Chapter this work used cosine similarity theory to identify misbehaviour. The next Chapter extends the discussions based on the uncertainties of making decisions about the internal attack.

Chapter 4 Epistemic Uncertainties Decision

Evidence, which is the basic representation of knowledge, enables the analysts and decision makers to determine the degree of belief of a proposition, to draw a conclusion and make a judgment about a complex system. Evidence is presented in several forms, such as data, information and knowledge. In this research, the terms of evidence, information and knowledge are used interchangeably.

Uncertainty is closely related to quality and quantity of knowledge or evidence. Types of uncertainty are based on patterns of evidence leading to a set of outcomes. The term evidence theory is used interchangeably in the literature with Dempster-Shafer theory (DST). Dempster-Shafer theory was originally introduced by Arthur Dempster in the middle of the 1960s. About ten years later the work of Dempster was extended, refined, recast, and published by Glenn Shafer in the 1970s. The Dempster-Shafer theory generalizes Bayesian theory to apply to distributing support not only to a single hypothesis but also to the union of hypotheses [135]. By including the distributing support in the hypothesis, the

DST easily includes uncertainty in the likelihood function and acknowledgement and even quantification of ignorance (Lack of evidence - if any belief cannot be further subdivide among the subsets of hypotheses that is reflected as ignorance) [136]. The Dempster-Shafer and Bayesian methods produce identical results when all the hypotheses are singletons (not nested) and mutually exclusive. A WSN is the most unpredictable and uncertain network as discussed in Chapter 1. Thus, to deal with uncertain events in WSNs a strong algorithm that can deal with the uncertainty is necessary.

Uncertainty can be broadly classified into objective (aleatory) uncertainty and subjective (epistemic) uncertainty. Some events or variables are inherently random and nondeterministic in nature [137]. This type of uncertainty cannot be reduced by increasing the knowledge and is called aleatory uncertainty. On the other hand, epistemic uncertainty stems from a lack of complete knowledge. Epistemic uncertainty can be reduced at the cost of increased resources, and this is the most common type of uncertainty in WSNs. In this thesis is dealing with epistemic uncertainties.

The DST has the feature of dealing with epistemic uncertainty. It considers the observed data as hypothesis. In an observation, data might be uncertain and system may not know in which hypothesis the data fits best [138]. Therefore, the DST makes it possible to model several pieces of evidence within multi hypotheses relations.

The following sections first introduce the concept of the DST to understand how DST works and how to implement in this research case then, with a case study, implement the DST in the WSN. Finally, this chapter presents the algorithm and simulation results.

4.1 Concepts of Dempster-Shafer Theory

The Bayesian theory is the canonical method for statistical inference problems. The Dempster-Shafer decision theory can be considered as a generalized Bayesian theory. The DST allows distributing support for proposition, not only to a proposition itself but also to the union of propositions that includes data [139]. In this research discussion on the Dempster-Shafer Theory (DST), a node can hold an uncertain opinion toward an event. The DST addresses the solution by representing the uncertainty in the form of *belief* functions [140]. The implementation ideas in the system, observer nodes, can obtain a degree of *belief* about the proposition from the related proposition's subjective probabilities. As the DST allows specifying a degree of ignorance in a situation instead of being forced to supply prior probabilities [141][142]. The ability of specifying the degree of ignorance explicitly models the degree of ignorance making the theory very appealing to WSNs, because of unreliable sensor and distributed infrastructure. To explain the concept of the DST, the following subsections discuss Bayesian interface and evidence methods of the DST that includes important functions of the DST.

4.1.1 Bayesian Interface

In order to understand the Dempster-Shafer Theory (DST), a study of Bayesian inference is helpful. Bayesian inference derives a posterior probability distribution as a consequence of two antecedents, a prior probability and likelihood, a probability model for the data to be observed [143][144]. Bayesian inference computes the posterior probability by conditioning, according to the rule of Bayes (Bayes rule is also discussed in appendix II) for the proposition of H and evidence E [145]. Bayes rule tells us how to perform inference about hypotheses from data (evidence). Thus, the interface as in equation 4.1:

$$P(H) = \frac{P(E|H)P(H)}{P(E)} \tag{4.1}$$

According to Bayesians interpretation, $P(H)$, the priori probability, for the proposition, H. $P(H)$ reflects the initial degree of belief in H in the absence of evidence E [146]. $P(H|E)$, the posteriori probability as a measure of belief about a hypothesis or proposition H that updates in response to evidence.

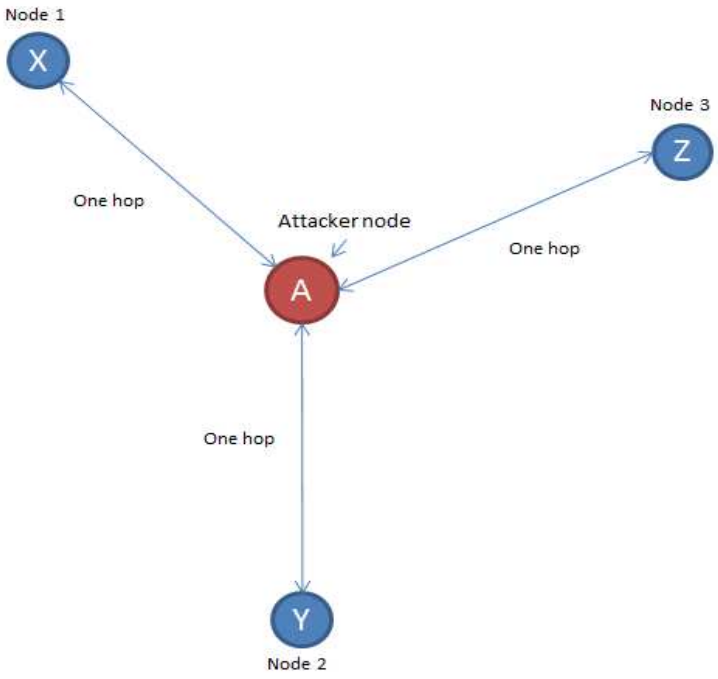

Figure 4-1: Three neighbours observing the attacker with one hop distance

In Figure 4-1 this research considers three nodes denoted as X, Y and Z. Each node has the representative pieces of evidence e_X, e_Y and e_Z as the evidence for X, Y and Z respectively, to support the hypothesis H. Hence, following the notations [146] the posteriori probability can be shown in equation 4.2:

$$P(H|e_X, e_Y, e_Z) = \frac{P(e_X, e_Y, e_Z|H)P(H)}{P(e_X, e_Y, e_Z|H)P(H) + P(e_X, e_Y, e_Z|{\sim}H)(1 - P(H))} \quad (4.2)$$

Where " $\sim H$ "is "not H" means the data is not in hypothesis. Thus, node A is an attacker. The neighbor nodes observe the attacker independently, hence the computation of the equation 4.2 can be simplified as in equation 4.3 by factorization process. The factorization process of joint probability density function is explained in appendix II.

$$P(H|e_X, e_Y, e_Z) = P(e_X H)P(e_Y H)P(e_Z H) \quad (4.3)$$

In the Bayesian interface approach, complete knowledge of the conditional and prior probabilities (The definition of conditional prior probabilities is in Appendix II) is a substantial requirement, which is difficult to have in practice.

In this approach, estimation of the prior probabilities is done from the empirical data. Hence, the limitations of this method include [146]:

- Difficulty in defining a priori probabilities;
- Complexities when there are multiple potential hypotheses and multiple conditionally dependent events;
- Mutual exclusivity required for competing hypotheses; and
- Inability to account for general uncertainty.

In order to tackle those limitations Dempster-Shafer theory is to be introduced in the research in the next section.

4.1.2 Dempster-Shafer Theory of Evidence Method

The Dempster-Shafer theory is a generalization of Bayesian theory [147] to allow for distributing support not only to a single hypothesis but also to the union of hypotheses. This way, the DST easily includes uncertainty in the likelihood function and acknowledgement [31]. The key features of the DST are as follows [148].

- The DST has the ability to specifically quantify and preserve ignorance,
- The DST has a facility for assigning evidence to combinations of choices - such as user in "'attacker OR normal" as well as singletons (unlike probability theory which must allocate probability to singletons), and
- The DST use of domain knowledge as a method for belief distribution.

Hence, the DST is suitable for the wireless sensor networks as sensor poses tend to be unreliable based on characteristics and application environment as discussed in Chapter 2.

Using the DST each evidence source (sensor nodes) has a total available amount of belief to be allocated, normalizing to a value of unity. The mass function for each evidence source allocates a source's belief across a set of choices. These choices are collectively called the *Frame of Discernment*. There are three important functions in the Dempster-Shafer theory [149]:

- The basic probability assignment function (bpa denoted by m), which is also called mass function,
- The Belief function (Bel), and
- The Plausibility function (Pl).

In the next subsections these are carefully discussed before this work implement the DST to the WSN to protect it from internal attacks.

4.1.2.1 Frame of Discernment and Mass Functions

A complete (exhaustive) set describing all of the sets in the hypothesis space is called frame of discernment (FoD) or simply called frame. FoD is a set of primitive hypothesis denoted by, θ . It must be exhaustive, in the sense of all possible primitive elements. FoD must be mutually exclusive (two events cannot occur at the same time) primitive elements [150]. It represents the set of choices $\theta = \{h_1, h_2, h_3, h_4, h_5, \dots, h_n\}$, where sources (such as sensors) assign belief or evidence across the hypotheses in the frame. For example, a weather sensor doing cloud presence prediction, where θ will represent $\{Cloud, Sunshine\}$ if the assumption is there are only two states. The possible mutually exclusive hypothesis (or events) of the same kind are enumerated in the frame of discernment also known as a universal discloser.

Formally, 2^{θ} denotes the set of all subsets of θ to which a source of evidence can apply its belief. The function $m : 2^{\theta} \rightarrow [0,1]$ is called a *mass function* that defines how belief is distributed across the frame. For example, if the function satisfies the following two conditions, for hypotheses A.

$$m(\emptyset) = 0$$

$$\sum_{A \in \theta} m(A_j) = 1$$

In which $\emptyset$ is the empty or null hypothesis, based on these conditions, belief from an evidence source cannot be assigned to an empty or null hypothesis, and belief from the evidence source across the possible hypotheses (including combinations

of hypotheses) must sum to 1, similar to the case of a probability theory. The least informative evidence (uncertainty) is the assignment of mass to a hypothesis containing all the elements $\{h_1, h_2, h_3, h_4, h_5, \dots, h_n\}$, because this evidence does not commit to any particular hypothesis.

4.1.2.2 Belief and Plausibility

Mathematically, the degree of belief is given by a single belief function, which can be related to lower bounds on probabilities, but conceptually belief and plausibility must be sharply distinguished from such lower and upper bounds. Hence, belief is the lower bound of the interval and represents supporting evidence. Plausibility is the upper bound of the interval and represents the non-refuting evidence [121]. With a frame of discernment and a body of empirical evidence, the belief committed to $A \in \theta$. The basic probability number can be translated as $m(A)$ because the portion of total belief assigned to hypothesis A, which reflects the evidences strength of support. The assignment of belief function maps each hypothesis B to a belief value $Bel\ (B)$ between 0 and 1. This is defined in equation 4.4.

$$Bel(B) = \sum_{j:A_j \epsilon \mathrm{B}} m(A_j) \tag{4.4}$$

The upper bound of the confidence interval is the plausibility function, which accounts for all the observations that do not rule out the given proposition. Plausibility maps each hypothesis B to a plausibility value $Pl(B)$ between 0 and 1, and can be defined as follows in equation 4.5.

$$Pl(B) = \sum_{j:A_i \cap B \neq \emptyset} m(A_j) \quad (4.5)$$

The plausibility function is a weight of evidence which is non-refuting to B. Equation 4.6 shows the relation between belief and plausibility.

$$Pl(B) = 1 - Bel(\sim B) \quad (4.6)$$

The hypothesis "not B" is represented by $\sim B$. The function's basic probability numbers, belief and plausibility are in one-to-one correspondence and by knowing one of them; the other two functions could be derived. Figure 4-2 shows the graphical representation of the above definition of belief and plausibility [143].

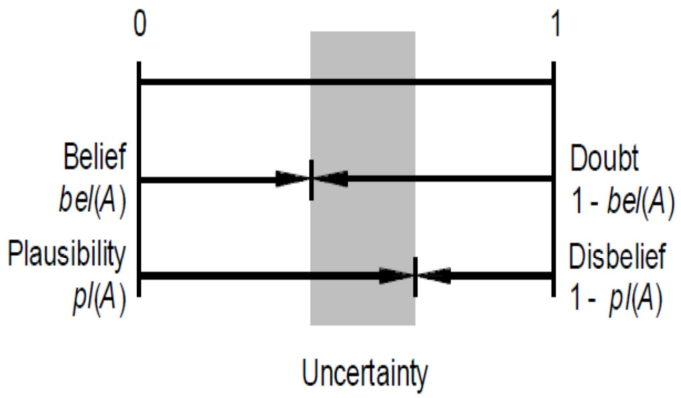

Figure 4-2: Measure of belief and plausibility

4.1.2.3 Combining Evidence

A crucial part of the process of assessing evidence is the ability to fuse evidence from multiple sources. Combining evidence is critical to the original conception of

the Dempster-Shafer theory. The measures of Belief and Plausibility are derived from the combined basic assignments. It combines multiple belief functions through their basic probability assignments (m). Specifically, the combination (called the joint $m_{1,2}$) is calculated from the aggregation of two bpa's m_1 and m_2.

Assuming $m_1(A)$ and $m_2(A)$ are two basic probability assignments by two independent items of evidence means two independent neighbour nodes which act as observers in the same frame of discernment. The observations (the pieces of evidence) can be combined using Dempster's rule of combination (known as orthogonal sum denoted by, $\oplus$) as in equation 4.7.

$$(m_1 \oplus m_2)(B) = \frac{\sum_{i,j:A_i \cap A_j = B} m_1(A_i)\, m_2(A_j)}{1 - \sum_{i,j:A_i \cap A_j = \phi} m_1(A_i)\, m_2(A_j)} \tag{4.7}$$

Here, "Dempster's combination" combines two basic probability assignments or basic belief assignments (BBA) into a third which is an unknown BBA [149]. To normalize the equation 4.7, consider G as basic probability mass associated with conflict. This is determined by the summing the products of the BPAs of all sets where the intersection is null. This research consider L is a normalization constant, which has the effect of *completely* ignoring conflict and attributing any probability mass associated with conflict to the null set, defined in equation 4.8, more than two belief functions can be combined pairwise in any order.

$$L = \frac{1}{G} \tag{4.8}$$

where ,

$$G = 1 - \sum_{i,j:A_i \cap A_j = \phi} m_1(A_i)\, m_2(A_j) \tag{4.9}$$

The Dempster's combination rule assigns the *belief* according to the degree of conflict between the evidences and assigns the remaining *belief* to the environment and not to common hypothesis. Combining evidence makes possible to combine with most of their *belief* assigned to the disjoint hypothesis [113]. The conflict between two *belief* functions bel_1 and bel_2, denoted by the $Con\,(bel_1, bel_2)$ is given by the logarithm of normalization constant shown in equation 4.10.

$$Con\,(bel_1, bel_2) = \log(L) \tag{4.10}$$

If there is no conflict between the bel_1 and bel_2, then $Con\,(bel_1, bel_2) = 0$ (or $L = 1$). The DST automatically incorporates the uncertainty coming from the evidences. It is possible to come up with a Dempster-Shafer combination, which can be given as in equation 4.11

$$m(B) = (m_1 \oplus m_2)(B) = \frac{L \sum_{i,j:A_i \cap A_j = B} m_1(A_i)\, m_2(A_j)}{1 + \log(L)} \tag{4.11}$$

Dempster's combination rule can be considered as a strict logic "AND" operation of the evidence sources because Dempster's combination rules are the special types of aggregation methods for data obtained from *multiple* sources. These multiple sources provide different assessments for the same frame of discernment and the Dempster-Shafer theory is based on the assumption that these sources are *independent*. From a set theoretic standpoint, these rules can potentially occupy a continuum between conjunction (AND-based on set intersection) [134][139]. An alternative will be required to cater for where sources are combined as logic "OR" scenarios. The next subsection explains the implementation of a temperature measurement WSN to protect from internal attack.

4.2 Case Study and Implementation

In this research WSN there is a number of sensors, for which the observations are assumed independent of each other. The Dempster-Shafer evidence combination rule provides a means to combine these observations. In the following description, this research takes a case study for temperature monitoring in a wireless sensor network. Designed temperature measurement of the WSN system is based on a single sinker. This research assumes the neighbor nodes with one hop will observe the data of the suspected internal attacker. In our observation, without loss generality, the physical parameter (temperature) and transmission behaviour (packet dropping rate) for each sensor are considered as independent events. The observation of the events becomes the pieces of evidences. In the decision making process, with Dempster-Shafer Theory, this work will combine the independent pieces of evidence.

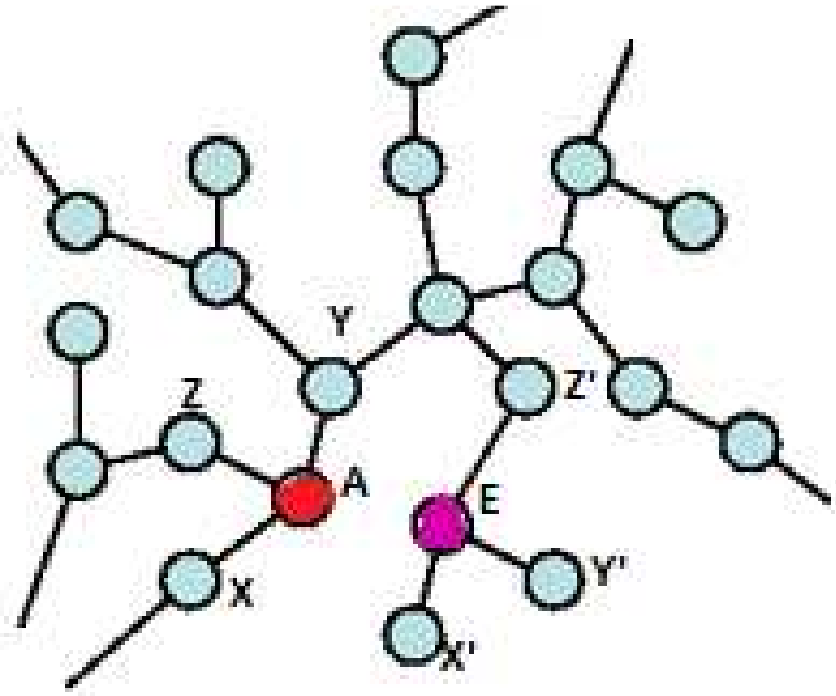

Figure 4-3 : Three neighbours observing the attacker with one hop

To take a specific scenario, which is described in Figure 4-3, the neighbour nodes X, Y and Z will observe the suspected internal attacker node A for its temperature (T) and packet drop rate (PDR). X', Y' and Z' will observe for the node E for its T' and PDR. The earlier section discussed the Dempster-Shafer theory. Now, this work applies the designed algorithm to Figure 4-3 as a case study for designed initiative.

In order to demonstrate the algorithm, the following paragraph will focus on the case study described in Figure 4-3. The main concept behind the internal attacks in a WSN is evidence or belief function. The evidence allows one to represent and fuse information evaluation provided by more or less reliable and conflicting sources on the same hypothesis. Designed case, the universal discloser or the set of local elements can be observed by the one hop neighbour that is $\theta = \{T, PDR\}$. Hence the power set becomes

$$2^{\theta} = \{\emptyset, \{T\}, \{PDR\}, \{unknown\}\}$$

Where

$$\{unknown\} = \{T\} \cup \{PDR\}$$

In specific case study for the simulation, this research has used the empirical data which was obtained from 20 runs of averages. The observation of node A by nodes X, Y and Z the basic probability assignments with T and PDR are,

$$m_T(X) = 0.3;\ m_T(Y) = 0.4; m_T(Z) = 0.2; m_T(U) = 0.1$$

$$m_{PDR}(X) = 0.4;\ m_{PDR}(Y) = 0.4; m_{PDR}(Z) = 0.2$$

Using the equations 4.8 and 4.9, the value of L and G can be obtained as shown below,

$$\begin{aligned} G = 1 - \big(& m_T(X)m_{PDR}(X) + m_{PDR}(X)m_T(U) \\ & + m_T(X)m_{PDR}(U) + m_T(Y)m_{PDR}(Y) \\ & + m_{PDR}(Y)m_T(U) + m_T(Y)m_{PDR}(U) \\ & + m_T(Z)m_{PDR}(Z) + m_{PDR}(Z)m_T(U) \\ & + m_T(Z)m_{PDR}(U)\big) \end{aligned}$$

$$G = 1 - (0.12 + 0.04 + 0.16 + 0.04 + 0.04 + 0.02)$$

$$= 1 - 0.42$$

$$= 0.58$$

Hence

$$L = \frac{1}{G} = \frac{1}{0.58} = 1.72$$

With the DST implementation as in equation 4.11, it is possible find the individual nodes observation about the suspected node A, based on the independent pieces of information or evidence.

$$m_{T,PDR}(X) = m_T(X) \oplus m_{PDR}(X)$$

$$m_{T,PDR}(Y) = m_T(Y) \oplus m_{PDR}(Y)$$

$$m_{T,PDR}(Z) = m_T(Z) \oplus m_{PDR}(Z)$$

Hence,

$$m_{T,PDR}(X) = \frac{L(m_T(X)m_{PDR}(X) + m_{PDR}(X)m_T(U) + m_T(X)m_{PDR}(U))}{1 + \log(L)}$$

$$= \frac{1.72 \times 0.16}{1 + \log(1.72)}$$

$$= 0.22$$

In the same way it is possible to calculate $m_{T,PDR}(Y)$ and $m_{T,PDR}(Z)$ and the result is 0.27 and 0.08 respectively. In the second case for the attacker E with different basic probability assignment by the nodes X', Y' and Z' are,

$$m_T(X') = 0.7;\ m_T(Y') = 0.75; m_T(Z') = 0.65; m_T(U) = 0.1$$

$$m_{PDR}(X') = 0.75;\ m_{PDR}(Y') = 0.7; m_{PDR}(Z') = 0.75$$

Using the equation 4.12 the observation by X', Y' and Z' for the node E the combination can be done as above.

From the above observations the calculation suggests that node $A\ is$ a normal node because the neighbour node considers it is compromised by 22%, 27%, and 8%. The average is about 20%. Hence, it is considered as a normal node. On the other hand for the node E, the neighbour node considers it is compromised by 65%, 70%, and 78% in Figure 4-3. The average is more than 70%. Therefore, node E is a compromised node or internal attacker.

4.2.1 Algorithm and Simulation

In order to find the internal attacks for this research case it can execute the above framework with equation 4.11. The algorithm used to do the simulation shown in Algorithm 4-1. The temperature threshold ∂_T and ∂_{PDR} is the threshold for the packet drop rate which is set based on the training data.

Algorithm 4-1 : The DST Implementation

I. Get the view of the neighbor node view

Input: $m_T, m_{PDR}, \partial_T, \partial_{PDR}$

m_T [ ]\\ BPA assignment

m_{PDR} [ ] \\ BPA assignment

II. Execute the equation 4.11

$m_{T,PDR}$ [ ]\\

If $m(B) < 0.6$

Output result accepted

printf "the node is an internal attacker"

else

Go to step I

end

Temperature measurement for the wireless sensor network is simulated in MATLAB to find the internal attack. This simulation has implemented the Dempster-Shafer theory of combination by considering individual pieces of evidences from the nodes. In the simulation environment the parameters were set are as follows,

Parameters	Values
Quantity of Sensors	100
Initial Energy	2 J
Packet Size	500 bytes
Regional Area	(0,0) to (500,500)

In the simulations, this research has established a WSN that observed for an internal attack in a two dimensional grid with one sink. The sink is located at the control center. This work has set the sensing range of the node as 100m for the simulation purpose. The results were produced with 100 different observations by the nodes. The observation is done 6 times every minute, which makes a better statistic results.

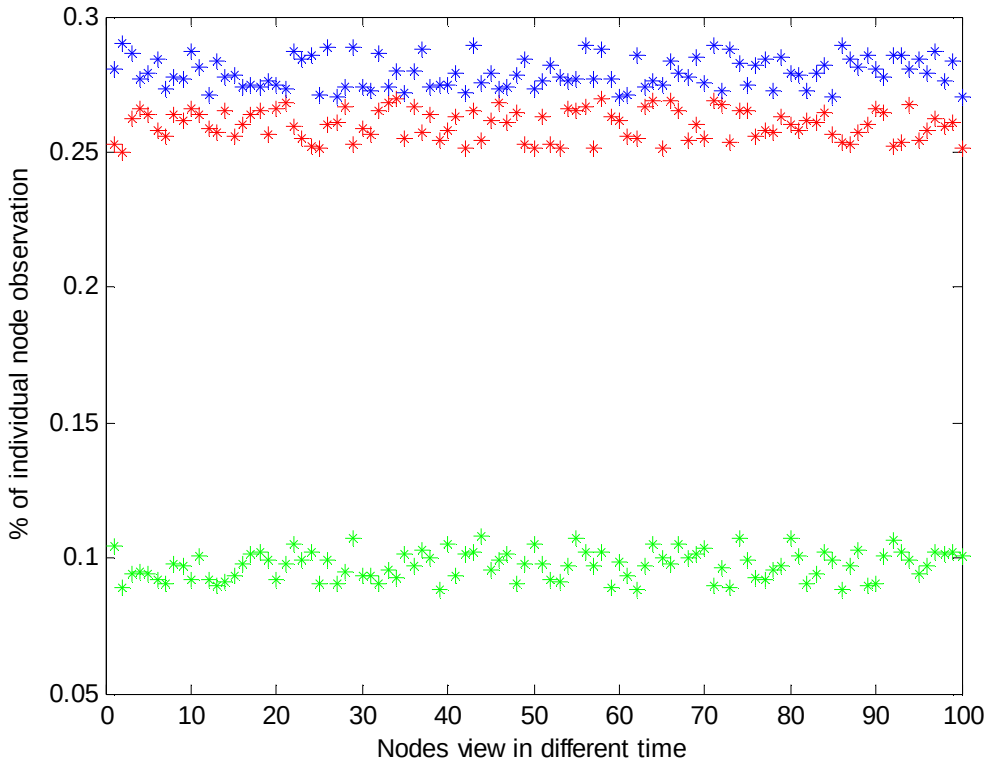

Figure 4-4 : Observation of node A by X, Y,and Z

Figure 4-4 shows that node A is compromised by observation 25% to 30% by the nodes X and Y. Red, Blue and Green is the observation by X, Y,and Z respectively. But for node Z observation says it is most likely compromised by 10%. Therefore, this work can consider that it might be a good node using the DST.

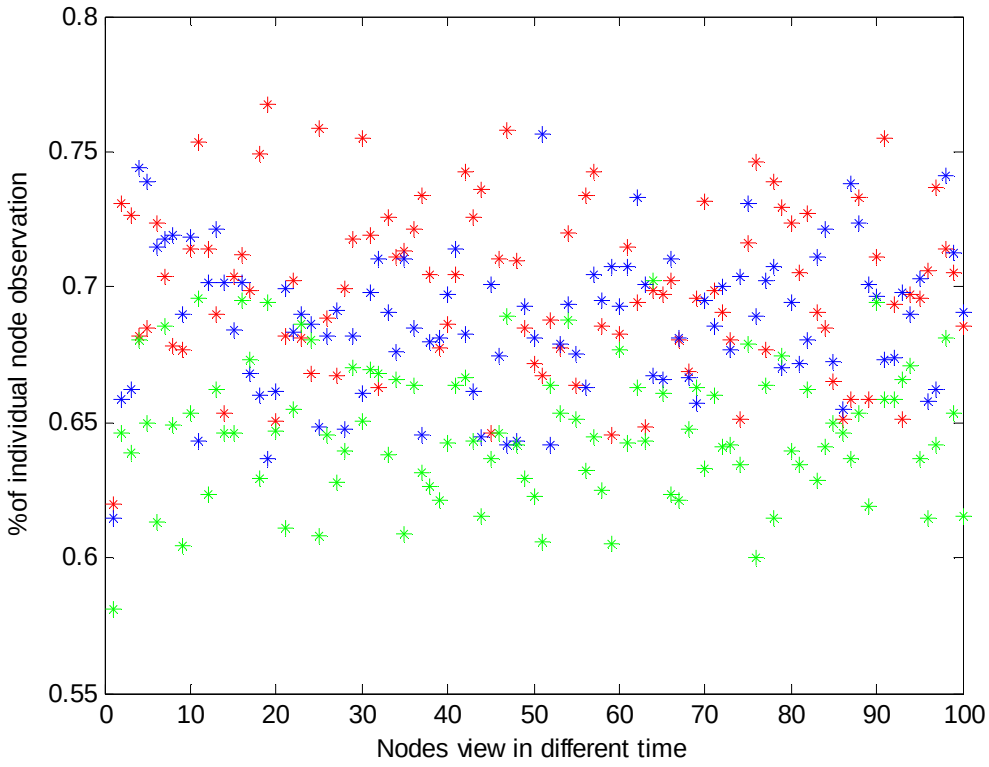

Figure 4-5 : Observation of node E by X', Y' and Z'

However, in the above second case, it showed in Figure 4-5 in the observation of the node E. Red, Blue and Green is the observation by X', Y',and Z' respectively. It showed the observation by node X', Y' and Z'. From this figure it is clearly seen that three nodes' observations give the higher percentage for the node E as an attacker. With the common result between 65 % to almost 80%, this research found that the node E is a compromised node or an internal attacker. In Figure 4-4 and Figure 4-5, red, blue and green are the observation results.

4.3 Summary

In this Chapter, a careful investigation was carried out on how to make a decision about the internal attacker in WSN based on Dempster-Shafer theory

that assigns evidence based on belief. Moreover, this Chapter discussed the concept of the DST mathematically and incorporated that into the designed application. A case study was developed to show how the DST could be applied for the protection from internal attacks in a WSN. Designed case study and simulations with empirical data showed that the algorithm works well in WSNs to find internal attacks. In the next Chapter, this research will further check for internal attacks with a novel algorithm based on Markov Chain Monte Carlo.

Chapter 5 Statistical Decision

The inception of Markov Chain Monte Carlo (MCMC) began in the late 1940s while Stan Ulam was playing solitaire [151]. He tried to compute the chances that a particular game of solitaire where 52 cards were laid out would come out successfully. After attempting exhaustive combinatorial calculations, he decided to go for the more practical approach of laying out several solitaires at random and then observing and counting the number of successful plays. This idea of selecting a statistical sample to approximate a hard combinatorial problem by a much simpler problem is the heart of modern Monte Carlo simulation. Stan Ulam soon realized that computers could be used in this fashion to answer questions of neutron diffusion and mathematical physics. He contacted John Von Neumann and they developed many Monte Carlo algorithms (importance sampling, rejection sampling) [152]. Then Nick Metropolis and Klari Von Neumann designed new controls for the state-of-the-art computer (ENIAC). Eventually Metropolis worked with Stan Ulam in 1949 and published the Monte Carlo simulation. Soon after in 1953, Nick Metropolis worked with Arianna W. Rosenbluth, Marshall N. Rosenbluth, Augusta H. Teller, and Edward Teller [153] and proposed the Metropolis algorithm. W. K. Hastings extended it to more

general cases in 1970 [154]. Hastings and his student Peskun showed that Metropolis and the more general Metropolis-Hastings algorithm are particular instances of a larger family of algorithms.

In the 1980's MCMC was implemented in the field of computer vision (image analysis) and artificial intelligence (data augmentation)[155][156]. In the 1990's MCMC made significant impacts on statistics with the work of Gelfand and Smith. In the neural network literature, the contribution of Neal in 1996 was influential for MCMC [157]. In 2000, a survey by Gelfand and Smith introduced top ten algorithms in the field of science and engineering practice and development for 20th century [158]. MCMC is among the top ten algorithms for science and engineering practice. Markov Chain Monte Carlo (MCMC) method is good to solve integration and optimization problems in large dimensional spaces, such as WSNs [159]. Although the method has been implemented in some genetic contexts, there have been relatively few published examples of its application to a WSN Security.

Markov Chain Monte Carlo sampling methods are based on construction of a Markov Chain (MC) whose equilibrium distribution is the target distribution of interest [160]. MCMC first constructed a transition kernel of an ergodic Markov Chain with the desired invariant distribution of data, and then simulated the chain for many steps to examine the data acceptance. The states sampled after the chain has converged will be distributed according to the target distribution of interest. It approximates the recursive Bayesian filtering distribution as a set of discrete samples known as a Markov Chain. In order to do this, this research follows the Monte Carlo approximation or integration, where the prior state is approximated by a set of samples.

In the following sections this work first introduces the Bayesian interface, then Monte Carlo and Markov Chain. The Metropolis-Hasting (MH) algorithm, which is the most popular method of MCMC, is discussed as method. MCMC – MH is

implemented in simulation to get the acceptance ratio of the node to make the decision about internal attackers. The simulation result shows that the algorithm is able to get the acceptance ratio of the node, based on the acceptance ratio this work decides about the internal attacks in a WSN.

5.1 Bayesian Interface

Most applications of MCMC to date, including this research application are oriented towards a Bayesian interface. Bayesian represent uncertainty about unknown parameter values by probability distributions and proceeds if the parameters are random quantities. If D represents the observed data and δ represent the model parameters, then to perform formal interface it requires setting up a joint probability distribution, $P(D, \delta)$ over all random quantities. This joint probability comprises two parts: a *prior* distribution $P(\delta)$ and a *likelihood* $P(D|\delta)$, Specifying $P(\delta)$ and $P(D|\delta)$ gives probability as in equation 5.1:

$$P(D, \delta) = P(\delta) \cdot P(D|\delta) \tag{5.1}$$

Once observation of the data D is done, Bayes theorem is used to determine the distribution of δ that is conditional on D. The *posterior* distribution of δ, as in equation 5.2, Bayes theorem is explained in appendix II.

$$P(\delta|D) = \frac{P(D|\delta) \cdot P(\delta)}{\int P(D|\delta) \cdot P(\delta)} \tag{5.2}$$

Understating and using the posterior distribution is at the heart of the Bayesian interface. Any futures of the posterior distribution are legitimate for a Bayesian interface such as moments, quantities. These quantities can be expressed as a

posterior expectation of the functions of δ. The posterior expectation of a function $f(\delta)$ is in equation 5.3.

$$E[f(\delta)|D] = \frac{\int f(\delta)P(\delta) \cdot P(D|\delta)d\delta}{\int P(\delta) \cdot P(D|\delta)\, d\delta} \quad (5.3)$$

Integrating out parameters in equation 5.3 can be time consuming; if the problem is high dimensional it would be very difficult and almost impossible to integrate as there will be many parameters. Diagnostically, performing the integration for the expectations has become a source of difficulty in application scenarios of Bayesian interface. Therefore, Monte Carlo integration using MCMC is one of the solutions.

To avoid an unnecessarily Bayesian flavor and make the discussion more general, the following discussion restates the problem in more general terms. Let X represent the vector of k random variables with the distribution denoted by$\pi(\cdot)$. In Bayesian applications, X may comprise model parameters and miss data. In frequentist applications, it may comprise data or random effects. For Bayesians, $\pi(\cdot)$ will be the posterior distribution and for frequentist it will be likelihood. Either way, the task is evaluating expectation for some function of interest $f(\cdot)$. The expectation represent as in equation 5.4

$$E\,[f(X)] = \frac{\int f(x)\,\pi(x)dx}{\int \pi(x)dx} \quad (5.4)$$

Here, this work allows for the possibility that the distribution of X is known only up to a constant normalization. That is, $\int \pi(x)dx$ is unknown.

5.2 Monte Carlo Integration

The idea for today's Monte Carlo simulation traces back to 1946, when Stan Ulam tried to figure out the chances to win a particular solitaire game laid out with 52 cards [161]. As calculations turned out to be complicated and exhausting, he had the idea to just play several times and count. This principle, approximating a complex combinatorial problem by the much easier process of drawing samples, is the basic idea of Monte Carlo simulations.

Monte Carlo integration is the fundamental part of MCMC. It uses a probabilistic interface to calculate the complex integrals or summation over a large outcome space [162]. Monte Carlo Integration evaluates expectations $E[f(X)]$ by drawing samples from $\{X_t,\ t = 1, \dots, n\}$ from $\pi(\cdot)$ and then approximating. The expectation is shown is Equation 5.5

$$E[f(X)] = \frac{1}{n}\sum_{t=1}^{n} f(X_t) \tag{5.5}$$

Hence, the population mean of $f(X)$ is estimated by a sample mean. When the samples $\{X_t\}$ are independent, based on Laws of Large Number (LLN) the approximation can be made as accurate as desired by the increasing size of n. (LLN is described in Appendix III)

In general, drawing samples $\{X_t\}$ independently from $\pi(\cdot)$ is not feasible since $\pi(\cdot)$ can be non-standard. However, $\{X_t\}$ does not necessarily have to be independent. The $\{X_t\}$ can be generated by any process which draws samples throughout the support of $\pi(\cdot)$ in the correct proposition. But Monte Carlo does not work when dependency exists among the states as the sample is drawn

independently. Thus, to deal with the independent samples a Markov Chain is introduced.

5.3 Markov Chains

A Markov Chain is a stochastic process where transition from one state to another state uses a simple sequential procedure [163]. The research starts a Markov Chain at one state and uses a transition function, to determine the next state, conditional on the last state. Then it keeps iterating to create a sequence of states, such a sequence of states being called a Markov chain.

Consider X_t is a random variable at time t and the state space referring to the possible X values. The random variable is a Markov process if the transition probabilities between different values in the state space depend only on the random variable's current state, if ϑ represents the state. Transition probabilities are shown in equation 5.6.

$$Pr(X_{t+1} = \vartheta_j \mid X_0 = \vartheta_k, \dots, X_t = \vartheta_i) = Pr(X_{t=1} = \vartheta_j \mid X_t = \vartheta_i) \quad (5.6)$$

Thus for a Markov random variable the only information about the past needed to predict the future is the current state of the random variable; knowledge of the values of earlier states does not impact on the transition probability. A Markov Chain refers to a sequence of random variables $(X_0, \dots X_n)$ generated by a Markov process. A specific chain is defined most critically by its transition probabilities, $P(i,j) = P(i \rightarrow j)$, which is the probability that a process at state space moves in a single step, as in equation 5.7.

$$P(i,j) = P(i \rightarrow j) = Pr(X_{t+1} = \vartheta_j \mid X_t = \vartheta_i) \quad (5.7)$$

This discussion will use the notation $P(i \rightarrow j)$ as a "transition probability". "$i \rightarrow j$" means moving from state i to state j. If $\pi_j(t) = \Pr(X_t = \vartheta_j)$, which means the probability that the chain is in state j at time t, and $\pi(t)$ denotes the row vector of the state space probabilities at step t. Start the chain by specifying a starting vector $\pi(0)$. Often all the elements of $\pi(0)$ are zero except for a single element of 1 unit, corresponding to the process starting in that particular state. As the chain progresses, the probability values get spread out over the possible state space.

The probability that the chain has state value ϑ_i at time (or step) $t + 1$ is given by the Chapman-Kolomogrov equation, which sums over the probability of being in a particular state at the current step and the transition probability from that state into state ϑ_i,

$$\pi_i(t+1) = \Pr(X_{t+1} = \vartheta_i)$$

$$= \sum_k \Pr(X_{t+1} = \vartheta_i | X_t = \vartheta_k) \cdot \Pr(X_t = \vartheta_k)$$

$$= \sum_k \mathrm{P}(k \rightarrow i)\pi_k(t) = \sum_k \mathrm{P}(k,i)\pi_k(t) \qquad (5.8)$$

Successive iteration of the Chapman-Kolomogrov equation describes the evolution of the chain. It is possible to compactly write the Chapman-Kolomogrov equations in matrix form as follows. Define the probability transition matrix **P** as the matrix whose $i-th$ to $j-th$ element is $P(i,j)$. The Chapman-Kolomogrov equation becomes as in equation 5.9. Details about Chapman- Kolomogrov equation is discussed in appendix III.

$$\pi(t+1) = \pi(t)\mathbf{P} \qquad (5.9)$$

Here, $t \geq 0$, using the matrix form and considering $\pi(t)$ is row vector, and considering τ is a transition, it is possible to generalise the equation as in equation 5.10.

$$\pi(t) = \pi(0)\mathbf{P}^{\boldsymbol{\tau}} \tag{5.10}$$

Defining the n-step transition probability $p_{ij}^{(n)}$ as the probability that the process is in state j given that it started in state i, n steps ago,

$$p_{ij}^{(n)} = \Pr(X_{t+n} = \vartheta_j | X_t = \vartheta_i) \tag{5.11}$$

It immediately follows that $p_{ij}^{(n)}$ is just the $i-th$ and $j-th$ element of $\mathbf{P}^{\boldsymbol{n}}$. Finally, a Markov Chain is said to be irreducible if there exists a positive integer such that $p_{ij}^{(n)} > 0$ for all i, j. That is, all states communicate with each other, as one can always go from any state to any other state (although it may take more than one step). Likewise, a chain is said to be aperiodic when the number of steps required to move between two states (say x and y) is not required to be a multiple of some integer. Put it another way, the chain is not forced into some cycle of fixed length between certain states.

As an example, consider the probabilities of WSN node conditions (modelled as either good or attacked), given the node on the preceding state, it can be represented by a transition matrix:

$$\mathbf{P} = \begin{bmatrix} 0.9 & 0.1 \\ 0.5 & 0.5 \end{bmatrix}$$

The matrix **P** represents the node condition model in which an attacked state is 90% likely to be followed by another attacked state, and a good state is 50% likely to be followed by another good state. The columns can be labelled "attacked" and "good", and the rows can be labelled in the same order. The transition matrix can be shown as a graph in Figure 5-1.

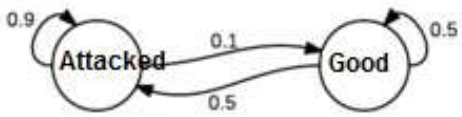

Figure 5-1 : The graph of transition matrix

$p_{ij}^{(n)}$ is the probability that, if a given state is of type i, it will be followed by a state of type j. Notice that the rows of **P** sum to 1, this is because **P** is a stochastic matrix.

Thus, the node condition at state 0 is known to be attacked. This is represented by a vector in which the "attacked" entry is 100%, and the "good" entry is 0%:

$$\pi(0) = [1 \quad 0]$$

The condition at state 1 can be predicted by:

$$\pi(1) = \pi(0)\mathbf{P} = [1 \quad 0] \begin{bmatrix} 0.9 & 0.1 \\ 0.5 & 0.5 \end{bmatrix} = [0.9 \quad 0.1]$$

Thus, there is a 90% chance that state 1 will also be attacked. The node condition at state 2 can be predicted in the same way:

$$\pi(2) = \pi(1)\mathbf{P} = \pi(0)\mathbf{P^2} = [1 \quad 0] \begin{bmatrix} 0.9 & 0.1 \\ 0.5 & 0.5 \end{bmatrix}^2 = [0.86 \quad 0.14]$$

Note that after a sufficient amount of time, the expected node condition is independent of the starting value.

A Markov Chain may reach a stationary distribution, where the vector of probabilities of being in any particular given state is independent of the initial condition. For the moment this work denotes the stationary distribution by $\varphi(\cdot)$. Thus as t increases, the sampled points $\{X_t\}$ will look increasingly dependent sample from $\varphi(\cdot)$. By using the output from a Markov Chain it is possible to estimate the expectation $E[f(X)]$, where X has stationary distribution $\varphi(\cdot)$.

The goal of MCMC is to design a Markov Chain such that the stationary distribution $\varphi(\cdot)$ is precisely the distribution of interest $\pi(\cdot)$. This is called the target distribution.

5.4 Markov Chain Monte Carlo Sampling

MCMC sampling combines the Monte Carlo principle of approximating a distribution by drawing random samples with the principle of Markov Chains. MCMC offers a mathematical framework to ensure that the derived sample has the desired properties. In this setting, the unknown parameters are the states of the Markov Chain, and a proposal function that suggests a new set of parameters based on the current one replaces the transition matrix. The main challenge is to ensure that the Markov Chain and the proposal function fulfil the required properties such that the desired posterior distribution is the invariant distribution of the chain. To this end, various methods existed. One of them is the Metropolis-Hastings algorithm which this research has implemented to protect a WSN from internal attacks. MCMC - MH allows approximating the posterior

distribution even if it is not possible to sample from it directly. The Metropolis-Hastings algorithm is simple but practical, and it can be used to obtain random samples from any arbitrarily complicated target distribution of any dimension that is known up to a normalizing constant [164]. The following sections discuss MCMC – MH and how does it works in a WSN to find the internal attacker.

5.4.1 Metropolis-Hasting (MH)

MCMC adopts the Metropolis-Hasting (MH) to generate a sample from stationary distribution. The goal is to draw samples from some distribution $\pi(\cdot)$ where, $\pi(X) = f(X)/C$, where, the normalizing constant C may not be known, and very difficult to compute. The Metropolis algorithm generates a sequence of draws from this distribution as follows:

- Start with any initial value X_0 satisfying $f(X_0) > 0$.
- Using current X value, sample a candidate point Y from some jumping distribution $Q(X_1, X_2)$ which is the probability of returning a value of X_2 given a previous value of X_1. This distribution is also referred to as the proposal or candidate-generating distribution. The only restriction on the jump density in the Metropolis algorithm is that it is symmetric, i.e., $(X_1, X_2) = Q(X_2, X_1)$.
- Given the candidate point Y, calculate the ratio of the density at the candidate Y and current (X_{t-1}) points at each time,

$$\rho = \frac{\pi(Y)}{\pi(X_{t-1})} = \frac{f(Y)}{f(X_{t-1})} \qquad (5.12)$$

Note that because this research is considering the ratio of $\pi(X)$ under two different values, the normalizing constant C cancels out.

- If the jump increases the density ($\rho \geq 1$), accept the candidate point which means ($X_t = Y$) and return to step 2. If the jump decreases the density ($\rho < 1$),, then with probability ρ, accept the candidate point,

It is possible to summarize the Metropolis sampling as first computing,

$$\rho = \min\left(\frac{f(Y)}{f(X_{t-1})}, 1\right) \tag{5.13}$$

and then accepting a candidate point with probability ρ. This generates a Markov Chain ($X_0, X_1, \ldots X_k$) as the transition probabilities from X_t to X_{t+1} depends only on X_t. Following a sufficient steps, the chain approaches its stationary distribution.

Hastings (1970) generalized the Metropolis algorithm by using an arbitrary transition probability function $Q(X_1, X_2) = \Pr(X_1 \rightarrow X_2)$, and setting the acceptance probability for the candidate or target. The target is then accepted with a probability ρ [165]. The acceptance probability for the target is shown in equation 5.14

$$\rho(X_t \rightarrow Y) = min\left(1, \frac{\pi(Y)Q(X_t|Y)}{\pi(X)Q(Y|X_t)}\right) \tag{5.14}$$

This research has the proposed target or candidate Y and calculated acceptance probability $\rho(X_t \rightarrow Y)$. Now either decide to "accept" the candidate or target (in which the next state becomes, $X_{t+1} = Y$) or decide to "reject" the target (in which, the chain does not move, $X_{t+1} = X_t$).

$$X_{t+1} = \begin{cases} Y & if \quad u \leq \rho(X_t \rightarrow Y) \\ X_t & if \quad u > \rho(X_t \rightarrow Y) \end{cases} \tag{5.15}$$

To make the decision to accept or reject the target this research depends on the uniformly distributed random number between 0 and 1, denoted by u, as shown in the equation 5.15.

From the above description its can be summarised that in order to find the internal attacks in a WSN, this research takes advantage of the Metropolis–Hastings algorithm to produce a sequence of sample values from the nodes, Therefore the next outcome of the nodes only depends on current samples of the nodes. As the process is making a Markov Chain with the sequence of samples, with some probability the algorithm produces an acceptance ratio, by which this work can make the decision about the target node. Hence, this research takes the decision if it is an internal attacker based on the acceptance ratio of the node. The next section further shows the system implementation and simulation.

5.5 System Implementation and Simulation

In the system designed for this research, based on the target of the node which is Y, and the proposal distribution of the node $Q(Y|X_t)$, this research can increase the target node acceptance probability with MCMC-MH. This research considers the time t is divided into equal length observation intervals based on 0.1 Hz and communication traffic is perceived as a sequence of states. Each observed state is descriptions of the traffic at time t the states observation. In the process, Markov Chain considers a set of states and transition matrix. This research measures a set of traffic features (packet transmissions) as a time series.

To determine the states the nodes observed the traffic feature during the implementation phase (learning phase). This work assumes at the implementation stage WSN is working perfectly with normal traffic, which is the expected traffic from the designed WSN. Hence, each node processes a time series of X of such observations. Then the MCMC - MH came into effect to find the acceptance ratio. In the system, this research considers that, if the acceptance probability is less than 60%, the node is an internal attacker. This work set the benchmark for a good node as more than 60% because of WSN characteristics such as signal noise; hostile environments affect the data collection as discussed in Chapters 2. In order to find the internal attacker this research executes a framework in the algorithm 5-1 shown below. This work has simulated the algorithm in a MATLAB environment to find the WSN node acceptance ratio. Based on the simulated output of the acceptance ratio of the WSN node this research takes the decision, whether it is an internal attacker or a good node.

Algorithm 5-1: MCMC-MH implementation

I. Initialize X_0 ; set $t = 0$

II. Iteration $t, t \geq 1$;

1. Sample a target or candidate $x \sim Q(Y|X_t)$

2. Evaluate the acceptance probability

$$\rho(X_t \to Y) = min\left(1, \frac{\pi(Y)Q(X_t|Y)}{\pi(X)Q(Y|X_t)}\right)$$

3. Sample $u \sim [0, 1]$.

III. Go to II.

end

In the simulation the acceptance probability of the node ranges from 0 to 1. The simulation was done in a small area $500m \times 500m$. The traffic feature was

chosen in a time interval to find the internal attacks because they are expected to correlate with the presence or absence of internal attacks. The simulation result is shown in Figure 5-2. The parameters this work used for the simulation in temperature measurement in the WSN are as follows:

Parameters	Values
Quantity of Sensors	100
Observation interval	10 sec
Packet Size	500 bytes
Number of samples	100

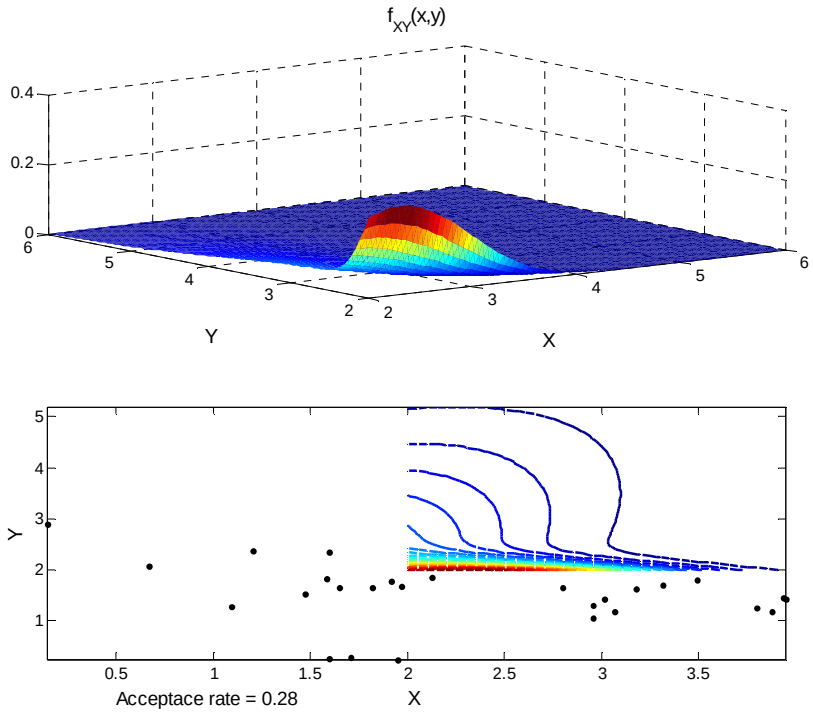

Figure 5-2 : MCMC-MH based node acceptance ratio

In the simulation result in Figure 5-2, above it can be seen that the distribution of the node data and below the acceptance rate of the target *Y* is 28%, which is shown in the bottom of Figure 5-2. As the node is accepted with the acceptance rate of 28% this work considers that node is an internal attacker.

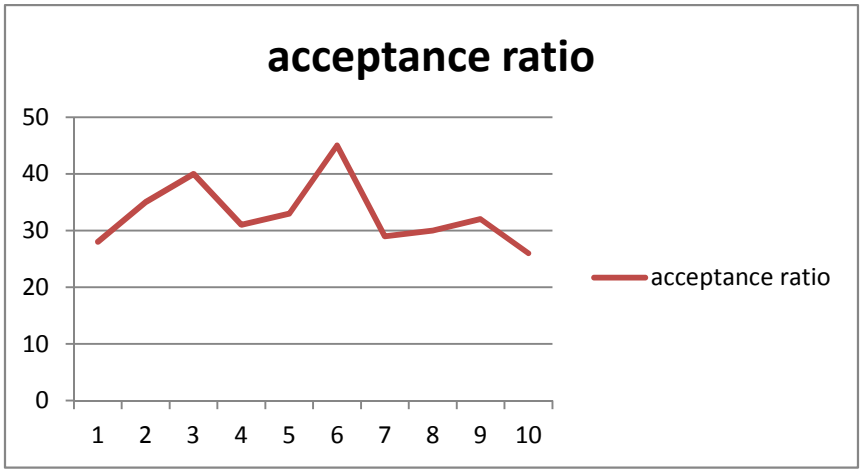

Figure 5-3: The acceptance ratio

Figure 5-3 shows the acceptance ratio based on the simulation run at different times. The simulation was run 10 times and the work found that the average acceptance ratio becomes 32.9% based on the designed case study.

5.6 Summary

This Chapter has investigated the Markov Chain Monte Carlo based Metropolis Hasting that has been implemented in WSNs to make decisions about internal attacks. MCMC provides an elegant way to access parameters of a model, even if the corresponding posterior distribution is not accessible. However, to implement this method in WSNs, this research does not need training data sets and it works in real time. The simulation results show the acceptance ratio of the internal attacks.

Chapter 6 Conclusion and Future work

Wireless sensor networks have seen extensive proliferation of applications and interest in research and industry. WSNs utilize an efficient form of technology that has no structures or rules or adher to a specific standard. Such networks are densely deployed to gather information in real time from the area of interest and send the information to the sink for further processing. Unfortunately, WSNs have several limitations in terms of security that make them vulnerable to appropriating meaningful information especially in a malicious environment.

Therefore, security becomes one of the keys to careful consideration in particular internal attacks. Detection of a compromised node (internal attack) is necessary in a WSN to ensure the functional performances. Internal attacks seriously disrupt the network functionality and almost all WSNs are susceptible to internal attacks. It is imperative to develop appropriate security mechanisms to protect WSNs from internal attacks.

In this Chapter, based on previous discussions, the next subsections highlight two issues as the final part of the thesis, namely (i) summary of mechanisms, which presents a realistic picture of this thesis for our WSN to identify internal attacks, and (ii) future work, which considers necessary and logical extensions of current research.

6.1 Contribution of the Research

This research has focused on a mechanism to protect WSNs from internal attacks. It is necessary to gain knowledge of WSNs, internal attacks and network security to develop such mechanisms. This research designed three distinct parts in the mechanism: (i) misbehaviour identification, (ii) epistemic uncertainty decisions and (iii) statistical decisions to solve the technical issues. The next paragraphs summarise the methods implemented in this thesis.

The proposed misbehaviour identification is done using multi-agent, pairwise key and cosine similarity. The multi-agent method used the highest SNR occurring time and location to control the receiver in the transmission range. The SNR established timing for the created window to open and close which is controlled by the multi-agent through a MAC layer both in sensor and sink node. Therefore, the attacker cannot transmit the signal without further information. With pair wise key model, this research establishes pair wire keys with a one hop neighbour. Prior to the network deployment, each node is pre-distributed an initial key. Then the node broadcasts an advertisement message which contains a nonce, and waits for another neighbour to respond with its identity. At the same time the neighbour node also generates a key. Then both nodes can generate the pair-wise key. Hence, each node can use its nodes' ID to calculate its one hop neighbour's key. If there is any stranger node in the WSN, such as the misbehaving node, it will be identified by those pair wise keys. This research has designed an abnormal behaviour identification mechanism (ABIM) with cosine

similarity that is sensitive to the abnormal event. It uses the dot product matchup elements or features in corresponding dimensions of two different parameter vectors. With the cosine similarity, it compares the node feature of the last message and new message to look for the similarity between them and identify the misbehaviour.

WSNs are dynamic and unpredictable networks based on particular characteristics. Hence, there is high demand for a mechanism that can deal with uncertainties to identify internal attacks. Dempster Shafer theory is one popular method which has the capability to deal with most uncertain events as discussed in Chapter 4. Dempster Shafer theory does not require an assumption regarding the probability of the individual constituents of the set or interval. Three main parts of the DST are basic probability assignment, belief and plausibility. The combination rule assigns the belief according to the degree of conflict between the evidence examples and assigns the remaining belief to the environment and not to a common hypothesis. It makes it possible to combine with most of the belief assigned to the disjoint hypothesis without the side effect of counterintuitive behaviour. Thus, this work introduced the DST based approach for detecting a compromised node (Internal attack) in WSNs. In the designed WSN for this work, the neighbour nodes with one hop will observe the data of the suspected internal attacker. In the observation, without loss of generality, the physical parameter (temperature) and transmission behaviour (packet drop rate) are considered as independent events. The observation of the events becomes the pieces of evidences. In the decision making process, with the Dempster Shafer Theory, this work combines the independent pieces of evidence to make the decision about an internal attacker.

This research proposed statistical analysis based decision making for internal attacks in WSNs with Markov Chain Monte Carlo (MCMC) using the Metropolis Hasting algorithm. MCMC provides an elegant way to access the parameters of a system. MCMC outputs a sample of parameters whose empirical distribution, for

long sequences converge the true posterior. The Metropolis Hasting (MH) generates the sample from the stationary distribution. In the designed case in this work, MCMC - MH checks for the acceptance ratio of the node in WSN; based on the acceptance ratio this research takes the decision about the internal attacker.

In summary, this thesis delivers a new approach to protect wireless sensor networks from internal attacks. WSNs are application oriented dynamic technology. Security requirements in WSNs differ based on the application scenario. It is impossible to develop a generic architecture for all WSNs at once. So, in WSN research, problems still exist based on the application scenario and characteristics.

6.2 Implication of Development

This thesis developed a solution for WSN security in particular to save WSNs from internal attack. To initiate the development of the internal security mechanism this research discussed WSN's characteristics, architecture, protocols, applications, security requirements and types of internal attacks. With the knowledge of WSNs this research has developed mechanisms to protect WSNs from internal attacks by utilizing Multi-Agent, Pair Wise Key, cosine similarity, Dempster-Shafer Theory and Markov Chain Monte Carlo algorithm. The mechanisms and approaches implemented in this thesis is an important contribution to save WSNs from internal attack.

6.3 Future Work

The previous Chapters have presented new algorithms for internal attacks in WSN. Internal attacker detection with misbehaviour identification is discussed

in Chapter 3. Chapter 4 introduced the security mechanism with Dempster Shafer theory. The detection of the internal attacker mechanism with MCMC-MH is introduced in Chapter 5. This research has identified some future work for further investigation, based on the current research, which should strengthen the proposed solutions and also some interesting research directions arising from this research:

- The combination of misbehaviour identification with the DST and the MCMC methods would make the protection more robust.
- The research would like to take a few parameters in consideration in future such as accuracy, error, reaction time and efficiency to achieve robustness.
- The benefit of recent technology development can be achieved by integration internet (such as cloud computing) and secured real time data collection platforms (secured-WSNs). In future this research would like to investigate the integration of the secure wireless sensor network with cloud computing using a content based publish / subscribe (pub/sub) broker model.
- Wireless sensor networks (WSNs) have been increasingly popular worldwide and make the complex network as shown in Figure 1-1. WSNs are one of the key enablers for the Internet of Things (IoT) where WSNs will play an important role in the future internet by collecting surrounding context and environment information. The innovations of integration of WSNs into IoT offer many interesting avenues of research for scientific communities. So this research would like to investigate the integration the IoT and WSNs.
- Big Data is a collection of large and complex data sets from a phenomenon. It is difficult to process the data using on-hand database management tools or traditional data processing applications. The challenges include capture, storage, search, sharing, analysis and visualization. This research would like to explore the potential framework of integrating IoT and WSN into the Big Data model of computing for applications in future.

6.3 Summary

This Chapter recapitulates the work undertaken in this thesis. A brief description is also provided of the further work to be undertaken to strengthen the proposed solutions.

The work undertaken in this thesis has been published extensively as a part of proceedings in peer reviewed international journals and conferences. A complete list of all the publications arising as a result of the work documented in this thesis is attached at the beginning of the thesis.

References

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *IEEE Communications Magazine*, vol. 40, no. 8, pp. 102–114, Aug. 2002.

[2] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of Things (IoT): A vision, architectural elements, and future directions," *Future Gener. Comput. Syst.*, vol. 29, no. 7, pp. 1645–1660, Sep. 2013.

[3] T. Arampatzis, J. Lygeros, and S. Manesis, "A Survey of Applications of Wireless Sensors and Wireless Sensor Networks," in *Proceedings of the 2005 IEEE International Symposium on, Mediterrean Conference on Control and Automation Intelligent Control, 2005*, June, pp. 719–724.

[4] A. Bagula, M. Zennaro, G. Inggs, S. Scott, and D. Gascon, "Ubiquitous Sensor Networking for Development (USN4D): An Application to Pollution Monitoring," *Sensors*, vol. 12, no. 1, pp. 391–414, Jan. 2012.

[5] J. Jeong and Z. J. Haas, "An integrated security framework for open wireless networking architecture," *IEEE Wireless Communications*, vol. 14, no. 2, pp. 10–18, 2007.

[6] X. Guo and J. Zhu, "Research on security issues in Wireless Sensor Networks," in *2011 International Conference on Electronic and Mechanical Engineering and Information Technology (EMEIT)*, 2011, vol. 2, pp. 636–639.

[7] A. Perrig, J. Stankovic, and D. Wagner, "Security in wireless sensor networks," *Communications of the ACM*, vol. 47, no. 6, p. 53, Jun. 2004.

[8] X. Du and H.-H. Chen, "Security in wireless sensor networks," *IEEE Wireless Communications*, vol. 15, no. 4, pp. 60–66, 2008.

[9] W. A. Arbaugh, "Wireless security is different," *Computer*, vol. 36, no. 8, pp. 99–101, 2003.

[10] W. Z. Khan, M. Y. Aalsalem, M. N. B. M. Saad, and Y. Xiang, "Detection and Mitigation of Node Replication Attacks in Wireless Sensor Networks: A Survey," *International Journal of Distributed Sensor Networks*, vol. 2013, May 2013.

[11] F. Ye, H. Luo, S. Lu, and L. Zhang, "Statistical en-route filtering of injected false data in sensor networks," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 4, pp. 839–850, April.

[12] A. Stetsko, L. Folkman, and V. Matyas, "Neighbor-Based Intrusion Detection for Wireless Sensor Networks," in *Proceedings of the 2010 6th International Conference on Wireless and Mobile Communications*, Washington, DC, USA, 2010, pp. 420–425.

[13] J. Sen, "An Efficient Security Mechanism for High-Integrity Wireless Sensor Networks," arXiv e-print 1012.2516, Dec. 2010.

[14] R. Johnstone, D. Caputo, U. Cella, A. Gandelli, C. Alippi, F. Grimaccia, N. Haritos, and R. E. Zich, "Smart Environmental Measurement and Analysis Technologies (SEMAT): Wireless sensor networks in the marine environment," presented at the Wireless Sensor and Actuator Network Research on Opposite Sides of the Globe (SENSEI), 2008.

[15] C.-Y. Chong and S. P. Kumar, "Sensor networks: evolution, opportunities, and challenges," *Proceedings of the IEEE*, vol. 91, no. 8, pp. 1247–1256, Aug.

[16] W. D. O'Neil, "The Cooperative Engagement Capability (CEC) Transforming Naval Anti-air Warfare," the Center for Technology and National Security Policy, the National Defense University,, U.S.A, Case Studies in National Security Transformation 11, Aug. 2007.

[17] S. Kumar and D. Shepherd, "SensIT: Sensor Information Technology For the Warfighter," in *Proceedings of 4th International Conference on Information Fusion*, Montréal, Canada, 2001, pp. 3–9.

[18] A. Hoskins and J. McCann, "Beasties: Simple wireless sensor nodes," in *33rd IEEE Conference on Local Computer Networks, 2008. LCN 2008*, 2008, pp. 707–714.

[19] M. El Brak and M. Essaaidi, “Wireless sensor network in home automation network and smart grid,” in *2012 International Conference on Complex Systems (ICCS)*, 2012, pp. 1–6.

[20] D. Li, K. D. Wong, Y. H. Hu, and A. M. Sayeed, “Detection, classification, and tracking of targets,” *IEEE Signal Processing Magazine*, vol. 19, no. 2, pp. 17 –29, Mar. 2002.

[21] C. Meesookho, S. Narayanan, and C. S. Raghavendra, “Collaborative classification applications in sensor networks,” in *Sensor Array and Multichannel Signal Processing Workshop Proceedings, 2002*, 2002, pp. 370 – 374.

[22] T. He, S. Krishnamurthy, J. Stankovic, T. Abdelzaher, L. Luo, R. Stoleru, T. Yan, and L. Gu, “Energy Efficient Surveillance System Using Wireless Sensor Networks,” presented at the The 2nd Annual International Conference on Mobile Systems, Applications and Services (MobiSys), 2004.

[23] B. Sinopoli, C. Sharp, L. Schenato, S. Schaffert, and S. S. Sastry, “Distributed control applications within sensor networks,” *Proceedings of the IEEE*, vol. 91, no. 8, pp. 1235 – 1246, Aug. 2003.

[24] P. Sikka, P. Corke, P. Valencia, C. Crossman, D. Swain, and G. Bishop-Hurley, “Wireless ad hoc sensor and actuator networks on the farm,” in *The Fifth International Conference on Information Processing in Sensor Networks, 2006. IPSN 2006*, 0-0, pp. 492–499.

[25] F. Zhao, “Wireless sensor networks: a new computing platform for tomorrow’s Internet,” in *Proceedings of the IEEE 6th Circuits and Systems Symposium on Emerging Technologies: Frontiers of Mobile and Wireless Communication, 2004*, 2004, vol. 1, pp. I – 27 Vol.1.

[26] R. Mulligan and H. M. Ammari, “Coverage in Wireless Sensor Networks: A Survey,” *Network Protocol and Algorithms,*, vol. 2, no. 2, pp. 27–53, 2010.

[27] M. Ahmed, X. Huang, and H. Cui, “Smart Decision Making for Internal Attacks in Wireless Sensor Network,” *International Journal of Computer Science and Network Security,*, vol. 12, no. 12, pp. 15–23, Dec. 2012.

[28] H. Wang and et al, *PDF: A Public-key based False Data Filtering Scheme in Sensor Networks*. 2007.

[29] A. Boukerche, "Performance Evaluation of Routing Protocols for Ad Hoc Wireless Networks," *Mobile Networks and Applications*, vol. 9, no. 4, pp. 333–342, Aug. 2004.

[30] S. A. K. Al-Omari and P. Sumari, "An Overview of Mobile Ad Hoc Networks for the Existing Protocols and Applications," arXiv e-print 1003.3565, Mar. 2010.

[31] L. Ertaul and N. Chavan, "Security of ad hoc networks and threshold cryptography," in *2005 International Conference on Wireless Networks, Communications and Mobile Computing*, 2005, vol. 1, pp. 69–74 vol.1.

[32] I. Mansour, G. Chalhoub, and A. Quilliot, "Security architecture for wireless sensor networks using frequency hopping and public key management," in *2011 IEEE International Conference on Networking, Sensing and Control (ICNSC)*, 2011, pp. 526–531.

[33] D. Culler, D. Estrin, and M. Srivastava, "Guest Editors' Introduction: Overview of Sensor Networks," *Computer*, vol. 37, no. 8, pp. 41–49, Aug. 2004.

[34] W. R. Heinzelman, J. Kulik, and H. Balakrishnan, "Adaptive protocols for information dissemination in wireless sensor networks," in *Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking*, New York, NY, USA, 1999, pp. 174–185.

[35] M. Ahmed, X. Huang, D. Sharma, and H. Cui, "Wireless Sensor Network: Cherecterestics and Architectures," in *World Academy of Science, Engineering and Technology*, Penang, Malaysia, 2012, vol. 72, pp. 660–663.

[36] C. Buratti, A. Conti, D. Dardari, and R. Verdone, "An Overview on Wireless Sensor Networks Technology and Evolution," *Sensors*, vol. 9, no. 9, pp. 6869–6896, Aug. 2009.

[37] K. V. Madav, C. Rajendra, and R. L. Selvaraj, "A Study of Security Challanges in Wireless Sensor Networks," *Journal of Theoretical and Applied Information Technology*, vol. 20, no. 1, pp. 39–44, May 2010.

[38] J. Feng, F. Koushanfar, and M. Potkonjak, "Sensor Network Architecture," Computer Science Department, University of California,, Los Angeles, U.S.A, Research Report for National Science Foundation, Grant No. ANI-0085773, 2005.

[39] S. Tilak, N. B. Abu-Ghazaleh, and W. Heinzelman, "A taxonomy of wireless micro-sensor network models," *SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 6, no. 2, pp. 28–36, Apr. 2002.

[40] T. He, J. A. Stankovic, C. Lu, and T. Abdelzaher, "SPEED: a stateless protocol for real-time communication in sensor networks," in *23rd International Conference on Distributed Computing Systems, 2003. Proceedings*, May, pp. 46–55.

[41] N. A. Pantazis, S. A. Nikolidakis, and D. D. Vergados, "Energy-Efficient Routing Protocols in Wireless Sensor Networks: A Survey," *IEEE Communications Surveys Tutorials*, vol. 15, no. 2, pp. 551–591, 2013.

[42] O. Erdene-Ochir, M. Minier, F. Valois, and A. Kountouris, "Resiliency of wireless sensor networks: Definitions and analyses," in *2010 IEEE 17th International Conference on Telecommunications (ICT)*, 2010, pp. 828 –835.

[43] N. Rathi, J. Saraswat, and P. P. Bhattacharya, "A review on routing protocols for application in wireless sensor networks," *International Journal of Distributed and Parallel Systems (IJDPS)*, vol. 3, no. 5, pp. 39–58, Oct. 2012.

[44] S. Misra, S. C. Misra, and I. Woungang, *Guide to wireless sensor networks*. London: Springer, 2009.

[45] M. Al-Rabayah and R. Malaney, "A New Hybrid Location-Based Ad Hoc Routing Protocol," in *2010 IEEE Global Telecommunications Conference (GLOBECOM 2010)*, 2010, pp. 1–6.

[46] Z. Kai, T. Libiao, and L. Wenjun, "Location-Based Routing Algorithms for Wireless Sensor Network," *ZTE communications*, vol. 1, no. 1, pp. 1–9, 2009.

[47] Z. Manap, B. M. Ali, C. K. Ng, N. K. Noordin, and A. Sali, "A Review on Hierarchical Routing Protocols for Wireless Sensor Networks," *Wireless Pers Commun*, vol. 72, no. 2, pp. 1077–1104, Sep. 2013.

[48] N. Gross, "21 ideas for the 21st century," *Business Week*, pp. 78–167, 30-Aug-1999.

[49] A. Pascale, M. Nicoli, F. Deflorio, B. Dalla Chiara, and U. Spagnolini, "Wireless sensor networks for traffic management and road safety," *IET Intelligent Transport Systems*, vol. 6, no. 1, pp. 67–77, 2012.

[50] A. Mainwaring, D. Culler, J. Polastre, R. Szewczyk, and J. Anderson, "Wireless Sensor Networks for Habitat Monitoring," in *Proceedings of the 1st ACM International Workshop on Wireless Sensor Networks and Applications*, New York, NY, USA, 2002, pp. 88–97.

[51] P. Juang, H. Oki, Y. Wang, M. Martonosi, L. S. Peh, and D. Rubenstein, "Energy-efficient Computing for Wildlife Tracking: Design Tradeoffs and Early Experiences with ZebraNet," in *Proceedings of the 10th International Conference on Architectural Support for Programming Languages and Operating Systems*, New York, NY, USA, 2002, pp. 96–107.

[52] C. Sharp, S. Schaffert, A. Woo, N. Sastry, C. Karlof, S. Sastry, and D. Culler, "Design and implementation of a sensor network system for vehicle tracking and autonomous interception," in *Proceeedings of the Second European Workshop on Wireless Sensor Networks, 2005*, 2005, pp. 93–107.

[53] CSIRO, "Wireless sensor networks: a new instrument for observing our world." [Online]. Available: http://www.csiro.au/Outcomes/ICT-and-Services/National-Challenges/Sensors-and-network-technologies.aspx. [Accessed: 30-Sep-2013].

[54] B. Rudra, A. P. Manu, and O. P. Vyas, "Investigating the Security Goals and Requirements for a Flexible Network Architecture," *Journal of Computer networks*, vol. 12, no. 3, pp. 10–29, Jun. 2012.

[55] S. Xiao, W. Gong, and D. Towsley, "Secure Wireless Communication with Dynamic Secrets," in *2010 Proceedings IEEE INFOCOM*, 2010, pp. 1–9.

[56] Y. Liu, J. Wang, H. Du, and L. Zhang, "Key Sharing in Hierarchical Wireless Sensor Networks," in *2010 IEEE/IFIP 8th International Conference on Embedded and Ubiquitous Computing (EUC)*, Dec., pp. 743–748.

[57] S. K. Singh, M. P. Singh, and D. K. Singh, "A Survey on Network Security and Attack Defense Mechanism for Wireless Sensor Networks," *International Journal of Computer Trends and Technology*, vol. 1, no. 2, pp. 1–9, Jun. 2011.

[58] D. G. Padmavathi and M. D. Shanmugapriya, "A Survey of Attacks, Security Mechanisms and Challenges in Wireless Sensor Networks," arXiv e-print 0909.0576, Sep. 2009.

[59] C. Karlof and D. Wagner, "Secure routing in wireless sensor networks: attacks and countermeasures," in *Sensor Network Protocols and Applications, 2003.*, Berkeley, CA, USA, 2003, pp. 113 – 127.

[60] R. Di Pietro, L. V. Mancini, and A. Mei, "Random key-assignment for secure Wireless Sensor Networks," in *Proceedings of the 1st ACM workshop on Security of ad hoc and sensor networks*, New York, NY, USA, 2003, pp. 62–71.

[61] T.-G. Lupu, "Main types of attacks in wireless sensor networks," in *Proceedings of the 9th WSEAS international conference on signal, speech and image processing, and 9th WSEAS international conference on Multimedia, internet & video technologies*, Stevens Point, Wisconsin, USA, 2009, pp. 180–185.

[62] S. Mohammadi and H. Jadidoleslamy, "A Comparison Of Physical Attacks On Wireless Sensor Networks," *International Journal of Peer to Peer Networks*, vol. 2, no. 2, pp. 24–42, Apr. 2011.

[63] P. Goyal, S. Batra, and A. Singh, "A Literature Review of Security Attack in Mobile Ad-Hoc Networks," *International Journal of Computer Applications*, vol. 9, no. 12, pp. 11–15, Nov. 2010.

[64] C. Hartung, J. Balasalle, R. Han, C. Hartung, J. Balasalle, and R. Han, "Node compromise in sensor networks: The need for secure systems," 2005.

[65] K. Lu, Y. Qian, and J. Hu, "A framework for distributed key management schemes in heterogeneous wireless sensor networks," *IEEE Transactions on Wireless Communications*, vol. 7, no. 2, pp. 639–647, Feb. 2008.

[66] M. R. Ahmed, X. Huang, and H. Cui, "A Novel Two-Stage Algorithm Protecting Internal Attack from WSNs," *IJCNC*, vol. 5, no. 1, pp. 97 – 116, Jan. 2013.

[67] M. Corporation, "Data sheet Tmote sky." Moteiv Corporation, 13-Nov-2006.

[68] B. Cross, "MICA 2: Wireless Measurement System." Crossbow technologiey, Inc, 2009.

[69] W. Xu, K. Ma, W. Trappe, and Y. Zhang, "Jamming sensor networks: attack and defense strategies," *IEEE Network*, vol. 20, no. 3, pp. 41–47, 2006.

[70] M. Li, I. Koutsopoulos, and R. Poovendran, "Optimal Jamming Attack Strategies and Network Defense Policies in Wireless Sensor Networks," *IEEE Transactions on Mobile Computing*, vol. 9, no. 8, pp. 1119–1133, 2010.

[71] M. Ahmed, X. Huang, and D. Sharma, "A Taxonomy of Internal Attacks in Wireless Sensor Network," in *World Academy of Science, Engineering and Technology*, Kuala Lumpur, Malaysia, 2012, pp. 427–430.

[72] E. Shi and A. Perrig, "Designing secure sensor networks," *IEEE Wireless Communications*, vol. 11, no. 6, pp. 38–43, Dec.

[73] X. Chen, K. Makki, K. Yen, and N. Pissinou, "Sensor network security: a survey," *IEEE Communications Surveys Tutorials*, vol. 11, no. 2, pp. 52–73, Quarter.

[74] K. Sharma and M. K. Ghose, "Wireless Sensor Networks: An Overview on its Security Threats," *IJCA, Special Issue on "Mobile Ad-hoc Networks" MANETs; CSE Department*, vol. 1, no. 1, pp. 42–45, 2010.

[75] H. K. D. Sarma and A. Kar, "Security Threats in Wireless Sensor Networks," in *Carnahan Conferences Security Technology, Proceedings 2006 40th Annual IEEE International*, Oct., pp. 243–251.

[76] D. R. Raymond and S. F. Midkiff, "Denial-of-Service in Wireless Sensor Networks: Attacks and Defenses," *IEEE Pervasive Computing*, vol. 7, no. 1, pp. 74–81, Jan.-March.

[77] H. Ghamgin, M. S. Akhgar, and M. T. Jafari, "Attacks in Wireless Sensor Network," vol. 5, no. 7, pp. 954–960, 2011.

[78] A. D. Wood and J. A. Stankovic, "A Taxonomy for Denial-of-Service Attacks in Wireless Sensor Networks," in *Handbook of Sensor Networks: Compact Wireless and Wired Sensing Systems*, U.S.A: CRC Press, 2004, pp. 1–18.

[79] J. Deng, R. Han, and S. Mishra, "Defending against path-based DoS attacks in wireless sensor networks," in *Proceedings of the 3rd ACM workshop on Security of ad hoc and sensor networks*, New York, NY, USA, 2005, pp. 89–96.

[80] Y.-C. Hu, A. Perrig, and D. B. Johnson, "Packet leashes: a defense against wormhole attacks in wireless networks," in *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies*, March-3 April, vol. 3, pp. 1976–1986 vol.3.

[81] A. Modirkhazeni, S. Aghamahmoodi, A. Modirkhazeni, and N. Niknejad, "Distributed approach to mitigate wormhole attack in wireless sensor networks," in *2011 The 7th International Conference on Networked Computing (INC)*, Sept., pp. 122–128.

[82] A.-S. K. Pathan, H.-W. Lee, and C. S. Hong, "Security in Wireless Sensor Networks: Issues and Challenges," arXiv e-print 0712.4169, Dec. 2007.

[83] C. Tumrongwittayapak and R. Varakulsiripunth, "Detecting Sinkhole attacks in wireless sensor networks," in *ICCAS-SICE, 2009*, Aug., pp. 1966–1971.

[84] J. R. Douceur, "The Sybil Attack," in *Revised Papers from the First International Workshop on Peer-to-Peer Systems*, London, UK, UK, 2002, pp. 251–260.

[85] K.-F. Ssu, W.-T. Wang, and W.-C. Chang, "Detecting Sybil attacks in Wireless Sensor Networks using neighboring information," *Computer Networks*, vol. 53, no. 18, pp. 3042–3056, Dec. 2009.

[86] L. K. Bysani and A. K. Turuk, "A Survey on Selective Forwarding Attack in Wireless Sensor Networks," in *2011 International Conference on Devices and Communications (ICDeCom)*, Feb., pp. 1–5.

[87] T. H. Hai and E.-N. Huh, "Detecting Selective Forwarding Attacks in Wireless Sensor Networks Using Two-hops Neighbor Knowledge," in *Seventh IEEE International Symposium on Network Computing and Applications, 2008. NCA '08*, 2008, pp. 325–331.

[88] Y. Chen, W. Trappe, and R. P. Martin, "Detecting and Localizing Wireless Spoofing Attacks," in *4th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks, 2007. SECON '07*, June, pp. 193–202.

[89] Y. Zhang and W. Lee, "Intrusion Detection in Wireless AdHoc Networks," presented at the ACM MOBICOM, The Annual International Conference on Mobile Computing and Networking, Boston, Massachusesttes, USA, 2000, pp. 275–283.

[90] A. P. R. da Silva, M. H. T. Martins, B. P. S. Rocha, A. A. F. Loureiro, L. B. Ruiz, and H. C. Wong, "Decentralized Intrusion Detection in Wireless Sensor Networks," in *Proceedings Of The 1st ACM International Workshop On Quality Of Service & Security in Wireless And Mobile Networks (Q2SWINET'05)*, 2005, pp. 16–23.

[91] M. Xie, S. Han, B. Tian, and S. Pravin, "Anomaly detection in wireless sensor networks: A survey," *Journal of Network and Computer Applications*, vol. 34, no. 4, pp. 1302–1325, Jul. 2011.

[92] J. Staddon, D. Balfanz, and G. Durfee, "Efficient tracing of failed nodes in sensor networks," in *Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*, New York, NY, USA, 2002, pp. 122–130.

[93] S. Marti, T. J. Giuli, K. Lai, and M. Baker, "Mitigating routing misbehavior in mobile ad hoc networks," in *Proceedings of the 6th annual international conference on Mobile computing and networking*, New York, NY, USA, 2000, pp. 255–265.

[94] K. Paul and D. Westhoff, "Context aware detection of selfish nodes in DSR based ad-hoc networks," in *IEEE Global Telecommunications Conference, 2002. GLOBECOM '02*, 2002, vol. 1, pp. 178 – 182 vol.1.

[95] S. Bansal and M. Baker, "Observation-based Cooperation Enforcement in Ad hoc Networks," *Research Report, cs.NI/0307012*, vol. 2, no. 1, pp. 1–10, Jul. 2003.

[96] Y. Huang and W. Lee, "A cooperative intrusion detection system for ad hoc networks," in *Proceedings of the 1st ACM workshop on Security of ad hoc and sensor networks*, New York, NY, USA, 2003, pp. 135–147.

[97] J. Pires, W.R., T. H. de Paula Figueiredo, H. C. Wong, and A. A. F. Loureiro, "Malicious node detection in wireless sensor networks," in *Parallel and Distributed Processing Symposium, 2004. Proceedings. 18th International*, 2004, p. 24.

[98] J. Branch, B. Szymanski, C. Giannella, R. Wolff, and H. Kargupta, "In-Network Outlier Detection in Wireless Sensor Networks," in *26th IEEE International Conference on Distributed Computing Systems, 2006. ICDCS 2006*, 2006, p. 51.

[99] S. Rajasegarar, C. Leckie, M. Palaniswami, and J. C. Bezdek, "Quarter Sphere Based Distributed Anomaly Detection in Wireless Sensor Networks," in *IEEE International Conference on Communications, 2007. ICC '07*, 2007, pp. 3864 –3869.

[100] K. Zhang, S. Shi, H. Gao, and J. Li, "Unsupervised Outlier Detection in Sensor Networks Using Aggregation Tree," in *Proceedings of the 3rd international conference on Advanced Data Mining and Applications*, Berlin, Heidelberg, 2007, pp. 158–169.

[101] Y. B. Reddy, “A Game Theory Approach to Detect Malicious Nodes in Wireless Sensor Networks,” in *Third International Conference on Sensor Technologies and Applications, 2009. SENSORCOMM ’09*, June, pp. 462–468.

[102] Y. Ma, H. Cao, and J. Ma, “The intrusion detection method based on game theory in wireless sensor network,” in *2008 First IEEE International Conference on Ubi-Media Computing*, 2008, pp. 326–331.

[103] S. H. Chi and T. H. Cho, “Fuzzy logic anomaly detection scheme for directed diffusion based sensor networks,” in *Proceedings of the Third international conference on Fuzzy Systems and Knowledge Discovery*, Berlin, Heidelberg, 2006, pp. 725–734.

[104] S. Y. Moon and T. H. Cho, “Intrusion Detection Scheme Against Sinkhole Attacks in Directed Diffusion Based Sensor Networks| Whitepapers | TechRepublic,” *Intrusion detection scheme against sinkhole attacks in directed diffusion based sensor networks*, vol. 9, no. 7, pp. 118–122, Jul. 2009.

[105] M. V. de S. Lemos, L. B. Leal, and R. H. Filho, “A New Collaborative Approach for Intrusion Detection System on Wireless Sensor Networks,” in *Novel Algorithms and Techniques in Telecommunications and Networking*, T. Sobh, K. Elleithy, and A. Mahmood, Eds. Springer Netherlands, 2010, pp. 239–244.

[106] S.-S. Wang, K.-Q. Yan, S.-C. Wang, and C.-W. Liu, “An Integrated Intrusion Detection System for Cluster-based Wireless Sensor Networks,” *Expert Systems with Applications*, vol. 38, no. 12, pp. 15234–15243, Nov. 2011.

[107] Z. Banković, J. M. Moya, J. C. Vallejo, and D. Fraga, “Detecting Unknown Attacks in Wireless Sensor Networks Using Clustering Techniques,” in *Hybrid Artificial Intelligent Systems*, E. Corchado, M. Kurzyński, and M. Woźniak, Eds. Springer Berlin Heidelberg, 2011, pp. 214–221.

[108] S. Hyun Oh, “A Malicious and Malfunctioning Node Detection Scheme for Wireless Sensor Networks,” *Wireless Sensor Network*, vol. 04, no. 03, pp. 84–90, 2012.

[109] W. Znaidi, M. Minier, Ub&#xe9, S. Da, and phane, “Hierarchical Node Replication Attacks Detection in Wireless Sensor Networks,” *International Journal of Distributed Sensor Networks*, vol. 2013, Apr. 2013.

[110] A. Garofalo, C. D. Sarno, and V. Formicola, “Enhancing Intrusion Detection in Wireless Sensor Networks through Decision Trees,” in *Dependable Computing*, M. Vieira and J. C. Cunha, Eds. Springer Berlin Heidelberg, 2013, pp. 1–15.

[111] D. Charles, K. Jain, and K. Lauter, “Signatures for Network Coding,” in *2006 40th Annual Conference on Information Sciences and Systems*, March, pp. 857–863.

[112] Z. Yu, Y. Wei, B. Ramkumar, and Y. Guan, “An Efficient Signature-Based Scheme for Securing Network Coding Against Pollution Attacks,” in *IEEE INFOCOM 2008. The 27th Conference on Computer Communications*, April, pp. 1409–1417.

[113] M. N. Krohn, M. J. Freedman, and D. Mazieres, “On-the-fly verification of rateless erasure codes for efficient content distribution,” in *2004 IEEE Symposium on Security and Privacy, 2004. Proceedings*, May, pp. 226–240.

[114] C. Gkantsidis and P. R. Rodriguez, “Cooperative Security for Network Coding File Distribution,” in *INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings*, April, pp. 1–13.

[115] A. Ababnah and B. Natarajan, “Optimal Control-Based Strategy for Sensor Deployment,” *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, vol. 41, no. 1, pp. 97–104, Jan.

[116] A. Sobeih, J. C. Hou, L.-C. Kung, N. Li, H. Zhang, W.-P. Chen, H. Tyan, and H. Lim, “J-Sim: a simulation and emulation environment for wireless sensor networks,” *IEEE Wireless Communications*, vol. 13, no. 4, pp. 104–119, Aug.

[117] M. Ahmed, X. Huang, and D. Sharma, "A Novel Framework for Abnormal Behaviour Identification and Detection for Wireless Sensor Networks," *International Journal of Computer and Communication Engineering*, vol. 6, no. 2, pp. 148–151, 2012.

[118] M. Drozda and H. Szczerbicka, "Artificial immune systems: Survey and applications in ad hoc wireless networks," in *Proc. 2006 International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS'06*, 2006, pp. 485–492.

[119] X. Huang, M. Ahmed, and D. Sharma, "Timing control for protecting from internal attacks in wireless sensor networks," in *2012 International Conference on Information Networking (ICOIN)*, 2012, pp. 7 –12.

[120] J. Zhang, T. Yan, and S. H. Son, "Deployment Strategies for Differentiated Detection in Wireless Sensor Networks," in *2006 3rd Annual IEEE Communications Society on Sensor and Ad Hoc Communications and Networks, 2006. SECON '06*, 2006, vol. 1, pp. 316–325.

[121] X. Huang, M. R. Ahmed, H. Cui, and L. Shutao, "Malicious node detection for the future network security from epistemic uncertainties," in *2013 16th International Symposium on Wireless Personal Multimedia Communications (WPMC)*, 2013, pp. 1–6.

[122] X. Huang, M. Ahmed, and D. Sharma, "Protecting from Inside Attacks in Wireless Sensor Networks," in *2011 IEEE Ninth International Conference on Dependable, Autonomic and Secure Computing (DASC)*, 2011, pp. 186–191.

[123] X. Huang, M. Ahmed, and D. Sharma, "A Novel Algorithm for Protecting from Internal Attacks of Wireless Sensor Networks," in *2011 IFIP 9th International Conference on Embedded and Ubiquitous Computing (EUC)*, 2011, pp. 344 –349.

[124] L. Eschenauer and V. D. Gligor, "A key-management scheme for distributed sensor networks," in *Proceedings of the 9th ACM conference on Computer and communications security*, New York, NY, USA, 2002, pp. 41–47.

[125] X. Huang, M. Ahmed, and D. Sharma, "Novel Protection from Internal Attacks in Wireless Sensor Networks," in *Computer Networks & Communications (NetCom)*, N. Chaki, N. Meghanathan, and D. Nagamalai, Eds. Springer New York, 2013, pp. 105–113.

[126] X. Huang, M. Barralet, and D. Sharma, "Accuracy of Location Identification with Antenna Polarization on RSSI," in *International Multi-Conference of Engineers and Computer Sciences*, Hong Kong, 2009, pp. 542–548.

[127] X. Huang, M. R. Ahmed, D. Sharma, and H. Cui, "Protecting wireless sensor networks from internal attacks based on uncertain decisions," in *2013 IEEE Wireless Communications and Networking Conference (WCNC)*, 2013, pp. 1854–1859.

[128] M. Cinque, A. Coronato, A. Testa, and C. Di Martino, "A Survey on Resiliency Assessment Techniques for Wireless Sensor Networks," in *Proceedings of the 11th ACM International Symposium on Mobility Management and Wireless Access*, New York, NY, USA, 2013, pp. 73–80.

[129] K. T.-M. Tran, S.-H. Oh, and J.-Y. Byun, "A comparative analysis of similarity functions of data aggregation for underwater wireless sensor networks," *International Journal of Digital Content Technology and Its Applications*, vol. 7, no. 2, pp. 830–837, 2013.

[130] M. R. Ahmed, X. Huang, and D. Sharma, "Protecting WSN from Insider Attack by Misbehaviour Judgement," in *Eighth International Conference on Wireless Communication and Sensor Network*, Naresuan University, Phitsanulok, Thailand, 2012.

[131] W. T. Zhu, Y. Xiang, J. Zhou, R. H. Deng, and F. Bao, "Secure localization with attack detection in wireless sensor networks," *Int. J. Inf. Secur.*, vol. 10, no. 3, pp. 155–171, Jun. 2011.

[132] X. Huang, M. R. Ahmed, and D. Sharma, "A Novel Protection for Wireless Sensor Networks from Internal Attacks," in *Proceedings of the International*

MultiConference of Engineers and Computer Scientists 2012, Hong Kong, 2012, pp. 10–16.

[133] A. Jøsang, "A logic for uncertain probabilities," *Int. J. Uncertain. Fuzziness Knowl.-Based Syst.*, vol. 9, no. 3, pp. 279–311, Jun. 2001.

[134] M. R. Ahmed, X. Huang, H. Cui, and N. K. Srinath, "A novel two-stage Multi-crieteria evaluation for internal attack in WSN," in *2013 13th International Symposium on Communications and Information Technologies (ISCIT)*, 2013, pp. 198–203.

[135] C. Thomas and N. Balakrishnan, "Modified evidence theory for performance enhancement of Intrusion Detection Systems," in *2008 11th International Conference on Information Fusion*, 2008, pp. 1–8.

[136] H. Wu, M. Siegel, R. Stiefelhagen, and L. Yang, "Sensor fusion using Dempster-Shafer theory [for context-aware HCI]," in *Proceedings of the 19th IEEE Instrumentation and Measurement Technology Conference, 2002. IMTC/2002*, 2002, vol. 1, pp. 7–12 vol.1.

[137] F. Khalaja, M. Khalajb, and A. H. Khalaj, "Bounded Error for Robust Fault Detection under Uncertainty, Part 1: Proposed Model Using Dempster-Shafer Theory," *Journal of Basic and Applied Scientific Research*, vol. 2, no. 2, 2012.

[138] M. Ahmed, X. Huang, and D. Sharma, "Dempster-Shafer Theory to Identify Insider Attacker in Wireless Sensor Network," in *Network and Parallel Computing*, J. J. Park, A. Zomaya, S.-S. Yeo, and S. Sahni, Eds. Springer Berlin Heidelberg, 2012, pp. 94–100.

[139] K. Sentz, "Combination of Evidence in Dempster-Shafer Theory," Binghamton University, Binghamton, NY, Unlimited Release SAND 2002-0835, Apr. 2002.

[140] X. Huang, D. Sharma, M. Ahmed, and H. Cui, "Protecting an WSN from internal attack based on epistemic uncertainties," in *2012 18th IEEE International Conference on Networks (ICON)*, 2012, pp. 389–394.

[141] M. Ahmed, X. Huang, D. Sharma, and L. Shutao, "Wireless sensor network internal attacker identification with multiple evidence by dempster-shafer

theory," in *Proceedings of the 12th international conference on Algorithms and Architectures for Parallel Processing - Volume Part II*, Berlin, Heidelberg, 2012, pp. 255–263.

[142] A. Bellenger and S. Gatepaille, "Uncertainty in Ontologies: Dempster-Shafer Theory for Data Fusion Applications," *CoRR*, vol. abs/1106.3876, 2011.

[143] M. R. Ahmed, X. Huang, and H. Cui, "A Novel Evidential Evaluation For Internal Attacks With Dempster-Shafer Theory in WSN," in *P of The 12th IEEE International Conference on Trust, Security and Privacy in Computing and Communications (IEEE TrustCom-13)orceedings*, Melbourne, Australia, 2013.

[144] D. Koks, *An Introduction to Bayesian and Dempster-Shafer Data Fusion*. DSTO Systems Sciences Laboratory, 2003.

[145] D. L. Hall and S. A. H. McMullen, *Mathematical Techniques in Multisensor Data Fusion*. Artech House, 2004.

[146] L. A. Klein, *Sensor and Data Fusion Concepts and Applications*, 2nd ed. Bellingham, WA, USA: Society of Photo-Optical Instrumentation Engineers (SPIE), 1999.

[147] J. W. Guan, Z. Guan, and D. A. Bell, "Evidential Reasoning in Expert Systems: Computational Methods.," in *IEA/AIE*, 2006, pp. 657–666.

[148] M. Beynon, B. Curry, and P. Morgan, "The Dempster–Shafer theory of evidence: an alternative approach to multicriteria decision modelling," *Omega*, vol. 28, no. 1, pp. 37–50, Feb. 2000.

[149] M. R. Ahmed, X. Huang, and D. Sharma, "A Novel Misbehavior Evaluation with Dempster-shafer Theory in Wireless Sensor Networks," in *Proceedings of the Thirteenth ACM International Symposium on Mobile Ad Hoc Networking and Computing*, New York, NY, USA, 2012, pp. 259–260.

[150] M. R. Ahmed, X. Huang, D. Sharma, and H. Cui, "Protecing WSN from Internal Attack with Multi-criteria Evaluation using Dempester-shafer Theory," in *Proceedings of International Conference on Information Systems, 2012*, Penang, Malaysia, 2012, vol. 62.

[151] C. Robert and G. Casella, "A Short History of Markov Chain Monte Carlo: Subjective Recollections from Incomplete Data," arXiv e-print 0808.2902, Aug. 2008.

[152] H. L. Anderson, "Metropolis, Monte Carlo, and the MANIAC," *Los Alamos Science*, vol. 14, no. Fall, pp. 96 – 108, 1986.

[153] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, "Equation of State Calculations by Fast Computing Machines," *The Journal of Chemical Physics*, vol. 21, no. 6, pp. 1087–1092, Dec. 2004.

[154] P. H. Peskun, "Optimum Monte-Carlo Sampling Using Markov Chains," *Biometrika*, vol. 60, no. 3, pp. 607–612, Dec. 1973.

[155] S. Geman and D. Geman, "Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-6, no. 6, pp. 721–741, Nov. 1984.

[156] J. Pearl, "Evidential reasoning using stochastic simulation of causal models," *Artificial Intelligence*, vol. 32, no. 2, pp. 245–257, May 1987.

[157] A. E. Gelfand and S. K. Sahu, "On Markov Chain Monte Carlo Acceleration," *Journal of Computational and Graphical Statistics*, vol. 3, no. 3, p. 261, Sep. 1994.

[158] I. Beichl and F. Sullivan, "The Metropolis Algorithm," *Computing in Science and Engg.*, vol. 2, no. 1, pp. 65–69, Jan. 2000.

[159] J. M. Zobitz, A. R. Desai, D. J. P. Moore, and M. A. Chadwick, "A primer for data assimilation with ecological models using Markov Chain Monte Carlo (MCMC)," *Oecologia*, vol. 167, no. 3, pp. 599–611, Nov. 2011.

[160] D. J. C. MacKay, *Information theory, inference, and learning algorithms*. Cambridge, UK; New York: Cambridge University Press, 2003.

[161] R. Eckhardt, "Stan Ulam, John von Neumann, and the Monte Carlo method," *Los Alamos Science*, vol. Special Issue, no. 15, pp. 131 – 137, 1987.

[162] I. Kyriakides, *On the Use of Monte Carlo Techniques for Integrated Sensing and Processing*. ProQuest, 2008.

[163] M. Martalò, S. Busanelli, and G. Ferrari, "Markov Chain-based performance analysis of multihop IEEE 802.15.4 wireless networks," *Performance Evaluation*, vol. 66, no. 12, pp. 722–741, Dec. 2009.

[164] S. A. S. Institute and S. A. S. Publishing, *SAS/STAT 9. 3 User's Guide: Survival Analysis (Book Excerpt)*. SAS Institute, 2011.

[165] M. R. Ahmed, X. Huang, and H. Cui, "Mrakov Chain Monte Carlo Based Internal Attack Evaluation for Wireless Sensor Network," *International Journal of Computer Science and Network Security,*, vol. 13, no. 3, pp. 23–31, Mar. 2013.

Appendix I

Z-Score method:

A normal distribution that is standardized (so that it has a mean of 0 and a standard deviation of 1) is called the standard normal distribution, or the normal distribution of z-scores. If the mean ("mu") is known, and standard deviation ("sigma") of a set of scores which are normally distributed, it is possible to standardize each "raw" score, x, by converting it into a z score by using the following formula on each individual score:

$$Z = \frac{x - \mu}{\sigma}$$

A z score reflects how many standard deviations above or below the population mean a raw score is. For instance, on a scale that has a mean of 500 and a standard deviation of 100, a score of 450 would equal a z score of (450-500)/100 = -50/100 = -0.50, which indicates that the score is half a standard deviation below the mean.

Note that converting x scores to z scores does NOT change the shape of the distribution. The distribution of z scores is normal if and only if the distribution of x is normal.

Appendix II

Bayes theorem, conditional probability and prior probability:

Bayes' theorem (also known as Bayes' rule or Bayes' law) is a result in probability theory that relates conditional probabilities. If A and B denote two events, P(A|B) denotes the conditional probability of A occurring, given that B occurs. Two conditional probabilities P(A|B) and P(B|A) are in general different. Bayes theorem gives a relation between P(A|B) and P(B|A).

Bayes' theorem relates the conditional and prior probabilities of stochastic events A and B:

$$P(A|B) = \frac{P(B|A)\,P(A)}{P(B)}$$

Each term in Bayes' theorem has a conventional name:

- $P(A)$ is the prior probability or marginal probability of A. It is "prior" in the sense that it does not take into account any information about B.
- $P(A|B)$ is the conditional probability of A, given B. It is also called the posterior probability because it is derived from or depends upon the specified value of B.
- $P(B|A)$ is the conditional probability of B given A.
- $P(B)$ is the prior or marginal probability of B, and acts as a normalizing constant.

Factorization process:

Let $[X \quad Y]$ be an absolutely continuous random vector with support R_{XY} and joint probability density function $f_{XY}(x,y)$. Denote $f_{X|Y=y}(x)$ by the conditional probability density function X of given $Y = y$ and by the marginal probability density function of . Then for any x and y the factorization becomes:

$$f_{XY}(x,y) = f_{X|Y=y}(x) f_Y(y)$$

If the joint probability density function $f_{XY}(x,y)$ is known, it is necessary to factorize it into the conditional probability density function $f_{X|Y=y}(x)$ and the marginal probability density function $f_Y(y)$, usually to proceed in two steps is followed:

- Marginalize $f_{XY}(x,y)$ by integrating it with respect to x and obtain the marginal probability density function $f_Y(y)$;
- Divide $f_{XY}(x,y)$ by $f_Y(y)$ and obtain the conditional probability density function $f_{X|Y=y}(x)$.

Appendix III

Law of Large Numbers:

The statistical principle that *the larger* the sample, *the more likely* it is that the frequency of events within the sample will approximate to the event's true frequency, *or put another way: the larger* the sample observed, *the more confident* one can be that a statistic derived from it (e.g., a mean or a proportion) is closer to its true value: with small samples greater variability should be expected, and with larger samples, less variability.

For example, imagine a *fair* coin is tossed four, sixteen, one hundred, or ten thousand times. Even though the expected number of 'heads' is ½ for a *single* toss, the expected outcome for, say, sixteen tosses is not certain to be eight 'heads'. Because *each* toss is an independent event having a *50/50* probability there is a variance in the proportion of heads yielded in any set of tosses. However, as the number of tosses *increases*, this variance of proportion *decreases*. The greater the number of trials, the closer the proportion of heads gets to ½, and produces what is called a "predictable ratio".

If the sample average

$$\overline{X_n} = \frac{1}{n}(X_1 + \cdots + X_n)$$

Which converges to the expected value

$\overline{X_n} \rightarrow \mu$ for $n \rightarrow \infty$

where X_1, X_2, ... is an infinite sequence of i.i.d. integrable random variables with expected value $E(X_1) = E(X_2,) = \ldots = \mu$. Integrability of X_j means that the expected value $E(X_j)$ exists and is finite.

An assumption of finite variance $Var(X_1) = Var(X_2) = \ldots = \sigma^2 < \infty$ is not necessary. Large or infinite variance will make the convergence slower, but the LLN holds anyway. This assumption is often used because it makes the proofs easier and shorter.

Chapman–Kolmogorov equation:

In mathematics, specifically in probability theory and in particular the theory of Markovian stochastic processes, the **Chapman–Kolmogorov equation** is an identity relating the joint probability distributions of different sets of coordinates on a stochastic process. The equation was arrived at independently by both the British mathematician Sydney Chapman and the Russian mathematician Andrey Kolmogorov.

Suppose that $\{f_i\}$ is an indexed collection of random variables, that is, a stochastic process. Let

$$p_{i_1} \ldots p_{i_n}(f_i \ldots f_n)$$

be the joint probability density function of the values of the random variables f_1 to f_n. Then, the Chapman–Kolmogorov equation is

$$p_{i_1} \ldots p_{i_n}(f_i \ldots f_{n-1}) = \int_{-\infty}^{\infty} p_{i_1} \ldots p_{i_n}(f_1 \ldots f_n) df_n$$