Intelligent Trust Management Methodology for the Internet of Things (IoT) To Enhance Cyber Security

A Thesis Submitted for the Degree of

Doctor of Philosophy

By

Mohammad Dahman Alshehri

Faculty of Engineering and Information Technology
University of Technology Sydney

March, 2019

CERTIFICATE OF AUTHORSHIP/ORIGINALITY

I, Mohammad Dahman Alshehri declare that this thesis, is submitted in fulfilment of the requirements for the award of PhD, in the School of Software/Faculty of Engineering and Information Technology at the University of Technology Sydney.

This thesis is wholly my own work unless otherwise reference or acknowledged. In addition, I certify that all information sources and literature used are indicated in the thesis.

This document has not been submitted for qualifications at any other academic institution.

This research is supported by the Australian Government Research Training Program.

Signature: Production Note: Signature removed prior to publication.

Date: 26/03/2019


بسم الله الرحمن الرحيم

ACKNOWLEDGEMENTS

All praises are due to Allah, The Most Beneficent, The Most Merciful, Who has enabled me to accomplish my PhD study. All the goals that I have achieved are due to His mercy; while all the mistakes are mine.

I would like to express my sincere gratitude to my PhD principal supervisor, Associate Professor Farookh Hussain, for his continuous support, encouragement, precious and guidance throughout my study. Thank you for your accurate critical comments and suggestions, which have strengthened this study significantly. Your strict academic attitude, generous personality and conscientious working style have influenced me deeply, and will be of great benefit to me in my future research work and life.

I am most grateful to my father, Dahman Alshehri, who always believed in my ability to be successful. He had passed away when I just began my PhD journey. May Allah bless my father's soul and grant him the paradise.

Most of all, I would like to express my deepest appreciation to my mother. She was supporting me and always being there for me during my ups and downs. Pursuing a PhD was always a long term challenge for me and a dream of my mother. This dream would have not been achieved without the prayers, inspiration, great empathy and kind assistance of my mother. Hence, great appreciation and enormous thanks are due to my mother. Likewise, I would like to thank my entire family members.

Finally, I would also like to express my thanks to Taif University, Saudi Arabian Cultural Mission (SACM) in Australia and everyone supported me during my PhD journey.

LIST OF PUBLICATIONS

JOURNAL ARTICLES PUBLISHED

1. Alshehri, M.D., Hussain, F.K. & Hussain, O.K. 2018, 'Clustering-Driven Intelligent Trust Management Methodology for the Internet of Things (CITM IoT)*' ACM/Springer Mobile Networks and Applications Journal*, vol. 23, no. 3 pp. 419-431. **(JCR (Q1), Impact Factor (3.259)-2017)**

2. Alshehri, M.D & Hussain, F.K 2018, 'A Fuzzy Security Protocol for Trust Management in the Internet of Things (Fuzzy-IoT)', *ACM/Springer Computing Journal*. **(Accepted).** (**Excellence in Research Australia (ERA) CORE, A-Rank Journal)**

CHAPTER OF BOOK PUBLISHED

3. Alshehri, M.D., Hussain F., Elkhodr, M. & Alsinglawi, B. 2018, 'A Distributed Trust Management Model for the Internet of Things (DTM-IoT)', *Recent Trends and Advances in Wireless and IoT-enabled Networks*, Springer*,* pp. 1-9.

CONFERENCE PAPERS PUBLISHED

4. Alshehri, M, Elkhodr, M & Alsinglawi, B 2018 'Data Provenance in the Internet of Things' *32nd International Conference on Advanced Information*

Networking and Applications Workshops (WAINA-2018), Krakow, Poland, pp. 727-731.

5. Alshehri, M.D, & Hussain, F.K 2017 'A Centralized Trust Management Mechanism for the Internet of Things (CTM-IoT)' *12th International Conference on Broadband and Wireless Computing, Communication and Applications (BWCCA-2017),* Barcelona, Spain, pp. 533-543.
6. Alshehri, M.D, & Hussain, F.K 2015 'A comparative analysis of Scalable and Context-Aware Trust Management Approaches for Internet of Things', *22nd International Conference on Neural Information Processing (ICONIP2015)*, Istanbul, Turkey, pp. 596-605. **(Excellence in Research Australia (ERA) CORE, A - Rank)**

OTHER PUBLICATIONS

CHAPTERS OF BOOK PUBLISHED

7. Elkhodr, M., Alsinglawi, B. & Alshehri, M. 2019, 'A Privacy Risk Assessment for the Internet of Things in Healthcare', *Applications of Intelligent Technologies in Healthcare*, Springer, pp. 47-54

8. Alsinglawi, B., Elkhodr, M. & Alshehri, M. 2018, 'RFID Localization in the Internet of Things', ACM/ Recent Trends and Advances *in Wireless and IoT-enabled Networks*, Springer, pp. 73-81

CONFERENCE PAPER PUBLICSHED

9. Ikram, M.A., Alshehri M.D, & Hussain. F.K 2015 "Architecture of an IoT-based System for Football Supervision", *IEEE 2nd World Forum on Internet of Things (WF-IoT 2015)*, Milan, Italy, pp. 14-16.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

Nowadays, the Internet of Things (IoT) connects billions of devices (things) using the Internet. The devices could be sensors, actuators etc. The number of IoT devices growing and interacting with each other raises the issues of security and trust. Most of the existing trust and security solutions do not present a comprehensive trust management solution for IoT addressing key trust management issues for the IoT. Many of the current solutions do not consider the scalability of the IoT trust management solution. With the rapid growth of IoT nodes a significant majority of the existing techniques do not address methods (or algorithms) to detect uncompliant behaviour or attacks on the trust management approach by the IoT nodes. The uncompliant behaviour may take the form or bad-mouthing attacks, on-off attacks, contradictory attacks and bad service attacks. In the existing literature there is no trust management approach that is scalable *and* resilient against attacks by uncompliant IoT nodes.

To address the above mentioned gaps in the existing literature body, in this thesis I propose an intelligent trust management platform for IoT (TM-IoT). The TM-IoT solution is centred on trust-based clustering of the IoT nodes. IoT nodes are grouped into clusters and each cluster is managed by a Master Node (MN). MN is a responsible for all the trust management activities within each cluster. The Super Node (SN) oversees and manages the MNs. Intelligent fuzzy-logic based and non-fuzzy logic based algorithms are presented to counter untrustworthy IoT nodes from carrying out attacks such as bad-mouthing attacks, on-off attacks, contradictory attacks and bad service attacks.

To validate the proposed solutions in this thesis, simulations were conducted using Contiki network simulator for IoT environment (Cooja), which able to simulate large networks. Using the built prototype, I have evaluated and simulated our proposed solutions for the above-mentioned problems by using Cooja and C++ programming language. The obtained results demonstrate the effectiveness of the TM-IoT and also that of the algorithms in achieving their respective goals.

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

The Internet of Things (IoT) is a recent technological revolution. It delivers services over the Internet and facilitates connectivity in a number of sectors such as health, transportation, mining etc. The IoT also facilitates communication between people and nodes (things) using the Internet (Xia et al. 2012). It is an innovative and novel approach to communication that allows nodes (or things) to connect with humans or with each other without any human interaction. The nodes or things could be physical things or virtual things (Alshehri & Hussain 2017). Things are able to be integrated into a physical network and have a unique IP address that can be used to identify them. For instance, physical things, such as electrical equipment and physical products, are now able to be sensed, actuated and connected to the Internet, which means these physical things have the ability to connect to the Internet and are described as physical objects. The International Telecommunication Union (ITU) defines a node in the IoT as a piece of equipment which is able to communicate and also has the optional abilities of sensing and actuating (Peña-López 2005). A virtual node (or thing) is not essentially a physical object. There are many examples of virtual nodes that can be stored, processed and accessed via the services over the Internet, such as web services or content (Castellani et al. 2011; Di Francesco et al. 2012)

IoT nodes are heterogeneous in nature, which means they move from one place to another and carry data and information. Furthermore, communication between the IoT nodes is heterogeneous which means that the IoT nodes are able to communicate with their counterpart nodes at any time and at any place. Furthermore, IoT communications can occur between nodes without human interaction. There are, broadly speaking, three types of communication between IoT nodes: communication between physical nodes known as machine-to-machine (M2M) communication; communication between virtual nodes; and communication between physical and virtual nodes (Alshehri & Hussain 2015). There are many application domains where the IoT can be applied, such as transportation, smart (cities, buildings, homes, schools etc.) (Zanella et al. 2014), healthcare, and agriculture (Shifeng et al. 2011) and the IoT allows different domains to share information (Zorzi et al. 2010). For example, in the transportation sector, the IoT assists in the real-time sensing and sharing of traffic information with various stakeholders, such as cars, traffic departments, and drivers etc. to avoid congestion. This assists greatly in optimising traffic flow (Parwekar 2011). From the perspective of networking, the IoT is a heterogeneous network that can be connected with different types of networks. Furthermore, the IoT nodes can group or cluster themselves into complex network structures Heterogeneity in the IoT network can include various network types such as Sigfox, LoRa, 4G or 5G, Wi-Fi, ZigBee and Bluetooth technologies etc. (Abedin et al. 2015).

The IoT will change the traditional concept of network (comprising of computers exchanging information) by introducing the notion of ‘things’ that can either receive or send information, thereby having a hugeimpact on the entire society and business. Consequently, the IoT has the potential to provide a large communication paradigm comprising of a number of ‘things’ across various network types (Höller et al. 2014; Mainetti, Patrono & Vilei 2011) posited that the IoT market would be worth more than 100 billion dollars by 2020. However, in 2013, Cisco predicted that the value of the economy that IoT would generate would go beyond 14.4 trillion dollars by 2020 (Lee

& Lee 2015). Moreover, IC Insights forecast that the number of new IoT connected nodes to the Internet would increase from 1.7 billion nodes in 2015 to more than 3.1 billion in 2019 (Lund et al. 2014). However, Cisco predicts that the number of new IoT connected nodes to the Internet will be more than 50 billion by 2020 (Evans 2012). In addition, BI Intelligence (Intelligence 2015) estimates that there will be an increase of around 35% in the number of new IoT devices that will be connected over the Internet between 2014 and 2019. These predictions highlight the importance of the IoT economy and its significance in the Internet-based economy.

Despite the great opportunities presented by the IoT, it faces many challenges, such as trust modelling, trust management, scalability and the security of trust management solutions etc. (Bao & Chen 2012b; Khan & Herrmann 2017; Li et al. 2011; Zhou & Chao 2011). The existing research has addressed some elements of these problems or challenges; however they have failed to propose a trust management solution that is scalable and secure against attacks, such as bad mouthing etc.

The next section presents the main research challenges in IoT that this thesis aims to address.

1.2 Research challenges related to the Internet of Things (IoT)

The IoT is one of the main technologies for the future (Alshehri, Hussain & Hussain 2018). IoT as a technology connects devices and facilitates information sharing, exchange and connectedness. This connection with devices comes with many issues and challenges. One of the key challenges facing the IoT is cyber-attacks and other cyber issues that make the IoT vulnerable to attack.

Based on the existing literature outlined in this chapter, three main issues significantly motivate the work presented in this thesis. These three challenges were only partially discussed in the existing literature or were not discussed at all. Therefore, this section focuses on these three important issues which are: (i) challenges related to trust management in IoT, (ii) challenges related to the scalability of the trust management approaches, (iii) challenges related to security protocols (or resilience) of the proposed trust management approach against cyber-attacks. This thesis addresses these issues and proposes a comprehensive platform (TM-IoT) that addresses the aforesaid issues. In the remainder of this section, the discussion is divided into three parts. In Section 1.2.1, I discuss the challenges associated with trust management in IoT and also the role and importance of trust management in IoT. In Section 1.2.2, I discuss some of the challenges associated with the scalability of trust management approaches. In Section 1.2.3, I discuss the issues related to the security IoT trust management approaches and how this can be addressed.

1.2.1 CHALLENGES RELATED TO TRUST MANAGEMENT IN IOT

Trust management offers a fundamental approach to securing IoT networks (Alshehri, Hussain & Hussain 2018). Trust management in IoT enables IoT devices (nodes) and also people to trust the communication and the exchange of data, information and messages between nodes. Intelligent and reliable trust management approaches in IoT can assist in establishing a trustworthy communication environment which in turn can provide a secure platform for delivering IoT-based services. To deliver these IoT services, there is a need for a reliable and dynamic IoT trust management solution. A reliable and scalable IoT trust management approach can provide optimal solutions to existing IoT cyber-security issues. However, there are limitations in the existing

literature in relation to developing a comprehensive platform for IoT trust management. Generally speaking, the existing literature only partially addresses the approaches to trust management in IoT such proposing models for computing trust in IoT. No attention has been paid in the existing literature outlined in this chapter in proposing a trust management solution for the IoT that is scalable and also resilient to untrustworthy behaviour or attacks by nodes. For example, some research papers, including (Li et al. 2011; Yu et al. 2017) explored the communication framework for IoT; however no consideration has been given to the trust layer as a main requirement to establish trustworthy communication between the nodes within the clusters. Because of these limitations, there is a need for a comprehensive IoT trust management approach. Unfortunately, the existing approaches lack the key attributes associated with the development of a comprehensive IoT trust management approach such as the scalability of the IoT trust management solution; the resilience of the proposed IoT trust management approach against the uncompliant behaviour of IoT nodes and a tamper-proof messaging system so that communication is secure and reliable. The majority of the existing research on IoT trust management focuses on the techniques that are suitable for WSNs, not for IoT nodes, taking into account the heterogeneity and transportability of IoT nodes. As these issues have not been addressed in the existing literature, in Chapter 5, I propose a comprehensive framework for IoT trust management. The proposed approach is based on a clustering of the IoT nodes into groups based on their trust value for trust management.

Furthermore, most of the published papers on trust management in IoT (Alexopoulos, Habib & Mühlhäuser 2018; Bao & Chen 2012a; Bao & Chen 2012b; Chen & Helal 2011; Kotis, Athanasakis & Vouros 2018; Li et al. 2011; Ren 2011; Roman, Najera & Lopez 2011; Wang et al. 2018; Yu et al. 2017; Zhou & Chao 2011) do not propose reliable IoT trust management approaches that can identify the non-compliant behaviour of the IoT nodes in the trust management approach, such as bad-mouthing,

contradictory behaviour, on-off attack etc. This is a pressing shortcoming of the existing literature on IoT trust management that needs to be solved.

1.2.2 CHALLENGES RELATED TO THE SCALABILITY OF TRUST MANAGEMENT IN IOT

As previously mentioned, trust management can facilitate secure communication between the IoT nodes in a network. The large number of IoT devices increases the risk of security threats such as (but not limited to) viruses or cyber-attacks. The proposed solution for IoT trust management should cater for the following challenges associated with the scalability of the trust management solution:

1. Application-based grouping of IoT nodes: In the existing literature, IoT nodes are typically grouped based on the applications that they service. The billions of deployed IoT node devices form complex networks or clusters, which are patterned on human social structures and entities (Chen, Guo & Bao 2016; Ortiz et al. 2014). These structures and entities come in the form of applications such as, IoT applications for health, homes, agriculture etc. with each of them forming a cluster of IoT nodes. The vision of the IoT's capacity to link myriad things as they interact with the environment and receive information on the status of those interactions was not previously possible by simply observing such sets of things (Fenye 2012). The clusters involved in the IoT are characterized by heterogeneity with a need for the development of trust from the interaction of one device to another device as well as from a device interacting with a user (Fenye 2012; Hamadeh, Chaudhuri & Tyagi 2017). Effective trust management in the IoT needs to be able to cater for this application-based grouping of the IoT nodes (Ahmed et al. 2016; Jabeur et al. 2017).

2. Memory constraints of the IoT node: One of the important issues that faces the IoT nodes is the possible memory shortage induced by the extreme memory usage of node services during the storage and computation of trust computations. The existing literature does not focus on this critical issue, which impacts negatively on the speed, weight and trust of the IoT nodes' communication. Furthermore, the existing literature fails to address critical issues related to developing scalable trust management IoT solutions, such as the issue of memory shortage of a node taking into account trust computations (Chen, Guo & Bao 2016; Sarkar et al. 2015).

The current body of literature outlined in this chapter does not address the aforesaid two issues in achieving the scalability of the IoT nodes. Consequently, there is no existing research that ensures that the developed IoT trust solutions are scalable across billions of IoT nodes, taking into account the important issues mentioned above. In Chapter 4 and Chapter 5, I present our solution, taking into account the above mentioned issues to achieve the scalability of the IoT trust management approach.

1.2.3 CHALLENGES RELATED TO SECURITY PROTOCOLS FOR IOT

A reliable IoT trust management approach that detects the non-compliant behaviour of IoT nodes is essential for the appropriate working of the IoT network as a whole. In this regard, the existing literature fails to address or propose approaches to key non-compliant behaviour by the IoT nodes (Ahmed et al. 2016; Babar et al. 2011; Chen et al. 2011; Gubbi et al. 2013; Kotis, Athanasakis & Vouros 2018; Lin et al. 2017; Lize, Jingpei & Bin 2014; Malina et al. 2016; Naik & Maral 2017; Nguyen, Laurent &

Oualha 2015; Renubala & Dhanalakshmi 2014; Sarobin & Ganesan 2016; Sirisala & Bindu 2015; Wang et al. 2018).

When IoT nodes detect on-off attacks, they can also detect contradictory behaviour attacks by non-compliant or untrustworthy nodes. Furthermore, bad-mouthing attacks can be used by IoT nodes to spread misinformation about the other nodes in the network. IoT nodes can carry out contradictory behaviour attacks where they provide contradictory behaviour. In order to address the above gaps in the literature, I propose intelligent methods to detect non-compliant behaviour by the IoT nodes. The detailed working of the solutions to the above mentioned non-compliant behaviours by the IoT nodes is presented in Chapter 6 and Chapter 7.

Finally, to maintain the security of the communication between IoT nodes, I develop a secure messaging system that enables secure communication between nodes. This messaging system uses hexadecimal values with a structure similar to serial communication. The messaging system is presented in detail in Chapter 7.

To validate the proposed approaches, I develop a prototype of the IoT trust management platform in a real IoT scenario using a special IoT simulation tool (Cooja) and other programming languages such as Java. Several issues that may improve the performance of trust management communication in the IoT paradigm are identified in our work and solutions are developed for them.

1.3 Objectives of this Thesis

The previous sections have described the importance of trust management in the IoT and the most critical issues related to the trust and security of IoT. In view of the aforementioned research challenges, I propose TM-IoT which is a clustering-based trust management approach for IoT. Furthermore, I propose intelligent approaches that

can enable the scalability of TM-IoT. Finally, intelligent approaches are proposed to secure the TM-IoT. The main objectives of this thesis are as follows:

Objective 1: To develop an intelligent trust management platform for IoT that is intelligent and memory-efficient in assisting the IoT nodes communicate in a trusted environment.

Objective 2: To develop an intelligent scalable trust management platform for IoT.

Objective 3: To develop intelligent approaches to counter cyber security attacks on the proposed IoT trust management approach.

Objective 4: To validate the above solutions using a prototype framework and test its performance along various identified benchmarks.

1.4 SIGNIFICANCE OF THE THESIS

The focus of this thesis is on proposing an intelligent methodology for trust management in the IoT. To achieve this, this thesis proposes a platform for IoT trust management and develops approaches to ensure that the proposed platform is scalable and also proposes intelligent approaches to counter cyber-security attacks on the proposed trust management platform. I categorize the significance of the thesis into two groups - scientific significance and social significance as follows.

1.4.1 SCIENTIFIC SIGNIFICANCE

1) This is the first work of its type that proposes a clustering-based approach for trust management in IoT. In the clustering-based approach, the IoT nodes are

grouped into clusters based on their trust value. The significance of this approach is that it provides a trusted environment for communication between the IoT nodes.

2) This is the first work that focuses on memory-efficient trust-based clustering approaches for IoT trust management.
3) This is the first research that focuses on developing a scalable trust management approach for the IoT network. The significance of this contribution is that the trust management solution can scale to any number of nodes and enable trusted communication between entities in the IoT network.
4) This research focuses on ensuring the integrity of the proposed trust management approach by developing intelligent solutions and approaches to counter cyber-attacks in the proposed trust management approach to the IoT platform. For the purpose of this thesis, I focus on developing intelligent methods to detect bad-mouthing attacks, ballot stuffing attacks, bad service attacks and on-off attacks by the IoT nodes that can comprise the integrity of the trust management approaches.

1.4.2 SOCIAL SIGNIFICANCE

The social significance of this work can be discussed in light of the benefits that the IoT trust management framework brings to the IoT consumers and to the IoT providers. In the following, I discuss the benefits to both the parties:

1) From the perspective of businesses (or providers), the proposed trust management platform for IoT enables consumers to further invest in the IoT technology or platform and develop new IoT-based services. This is because the proposed trust management platform gives businesses the confidence that the IoT platform is trustworthy and provides a secure communication mechanism between IoT nodes.

2) From the perspective of the IoT consumers, a trusted IoT environment with different clusters of IoT nodes gives the consumer more confidence in a given IoT service. The IoT services are clustered into various groups, with each group hosting services in a specific trust range.

1.5 STRUCTURE OF THE THESIS

In this thesis, I provide a comprehensive methodology for IoT trust management. This includes developing intelligent approaches to enable the scalability of the IoT trust management approach and also developing intelligent approaches for countering security attacks (such as bad-mouthing attacks, on-off attacks etc.) in the IoT trust management framework.

To accomplish these objectives, this thesis is organised in eight chapters. In this section, I provide a brief summary of each chapter. This chapter outlines the role of trust management in IoT and the importance of the scalability and security of the IoT-based trust management approach. A brief overview of the challenges to achieving these objectives follows the objectives of this thesis. Subsequently, both the scientific significance and social significance of our work is briefly overviewed. The remainder of this thesis is organized as follows:

Chapter 2: This chapter provides an extensive overview of the extant literature. It discusses and analyses the existing literature from different aspects of IoT trust management, particularly the scalability of the trust management approaches and the reliability of the proposed approach against cyber-attacks. The literature is divided into five categories depending on the attributes and working of the proposed IoT trust management approaches. The features and shortcomings of each method are identified

in each section and the chapter concludes with an extensive comparison of all of the related approaches.

Chapter 3: This chapter formally defines the research problem of trust management in the IoT that I address in this thesis. Subsequently, this chapter presents the definitions of the terminologies that will be used to define the problem addressed in this thesis. Further, the research problem is divided into four research issues. Based on these research issues, the research questions and objectives are defined. Finally, the research methodology used to develop the solution to the identified research issues is presented in this chapter.

Chapter 4: This chapter overviews the proposed trust management solution for the IoT (in response to the research question that was addressed in chapter 3). An overview of the proposed trust management platform (TM-IoT) is presented in this chapter. Furthermore, this chapter describes in a methodological manner the steps in solving each of the identified research sub-questions. An overview of the intelligent processes and algorithms to address the issues related to the security and cyber-security of the proposed IoT trust management solution are presented.

Chapter 5: This chapter presents the working of the proposed trust management platform for the Internet of Things (TM-IoT). The key elements of the TM-IoT are the super node (SN), master node (MN), cluster nodes (CN), clusters and the trust communication module (TCM). This chapter describes the working of each of these components and details the inter-relationships between them.

Chapter 6: This chapter presents the clustering-driven intelligent trust management methodology for the Internet of Things (CITM-IoT). This chapter presents algorithms based on the clustering method that allows the TM-IoT platform to be scalable to a

large number of nodes. Furthermore, it presents intelligent algorithmic approaches for migrating nodes from one cluster to another based on their trust values. Finally, this chapter presents intelligent solutions to counter the non-compliant behaviour of the IoT nodes to prevent cyber-attacks, such as bad-mouthing. Extensive experiments were conducted and the results of the proposed solution are presented.

Chapter 7: This chapter describes the fuzzy-logic-based security protocol for trust management in the Internet of Things (Fuzzy-IoT). In this chapter, a messaging system is proposed for TM-IoT for secure communication between the IoT nodes. Furthermore, fuzzy-logic-based algorithms are presented to ensure the reliability and integrity of the TM-IoT such as the detection of bad-mouthing attacks, contradictory behaviour attacks, on-off attacks and bad service attacks by untrustworthy IoT nodes. Finally, the experimental process and the results obtained by the algorithms are presented.

Chapter 8: This chapter presents the conclusions and future research directions for the work presented in this thesis.

The structure of the thesis based on the eight chapters is presented in Figure 1-1.

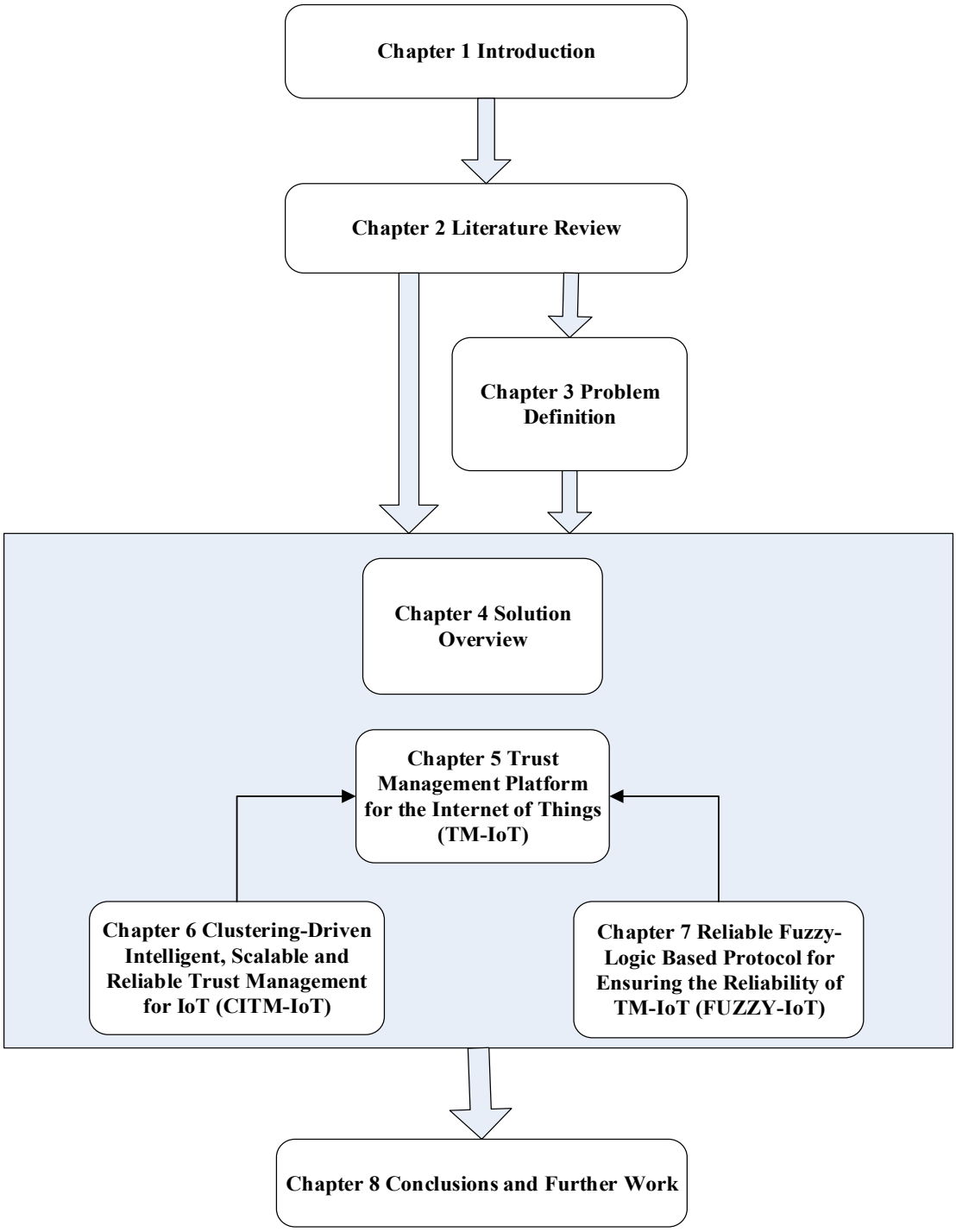

Figure 1-1 The structure of the thesis

1.6 CONCLUSION

The Internet of Things (IoT) is an emerging technology that offers easy connectivity between devices over the Internet. This connection between devices carries data, information and services. However, the risk of attack that threatens data and information on the IoT is extremely serious. Despite the risks, the adoption of IoT is growing rapidly and the number of devices that are connected to each other by the IoT is increasing day-by-day. To address the challenges that face the IoT, this thesis aims to provide a comprehensive solution to some of the key shortcomings in the existing literature on IoT trust management.

This chapter introduced the reader to the problem addressed in this thesis. It presents an overview of IoT and the future impact of IoT. It then discusses the issues facing trust management approaches for the IoT. The importance of the proposed solution of trust management for IoT is to enable IoT devices and the entire environment to communicate with each other in a secure environment and with trusted communications. The significance of this research is also discussed in this chapter. A brief description of each of the eight chapters is presented with an overview of how the chapters are linked to each other.

The next chapter outlines and discusses the existing literature on IoT trust management.

CHAPTER 2

LITERATURE REVIEW

2.1 INTRODUCTION

The main purpose of this chapter is to overview the current studies and discuss the relevant work which has been done in the field of trust management in the Internet of Things (IoT). The content of this chapter is organized as follows. In Section 2.2, I provide an overview of trust management and scalability in the IoT. This section is further divided into the following two subsections: section 2.2.1 discusses trust management in IoT and section 2.2.2 investigates the existing solutions for trust management in IoT. In Section 2.3, I discuss the context-aware assessment approaches for IoT and undertake a comparative analysis of trust management in the IoT with an overview of the shortcomings of the existing literature. Section 2.4 discusses security protocol for reliable trust management for IoT and undertake a comparative analysis of existing literature in this area with a view to identify the shortcomings of the existing literature. Clustering-based trust management for the IoT is discussed in section 2.5, followed by a comparative analysis of the existing literature body in this area. In section 2.6, fuzzy logic-based approaches for trust management in the IoT are discussed followed by a comparative analysis of the existing literature body in this area.

Section 2.7 presents a critical evaluation of the current literature, which is followed by the conclusion. The basis for classifying the existing literature into four distinct categories is the use of various approaches used for IoT trust management (fuzzy logic or clustering) and also the trust-related issues or problems that they address (such as

addressing cyber-attacks through use of trust protocols and context-aware trust assessments).

This chapter contains sections that have been earlier published by us in (Alshehri, Hussain & Hussain 2015; Alshehri & Hussain 2017; Alshehri & Hussain 2018).

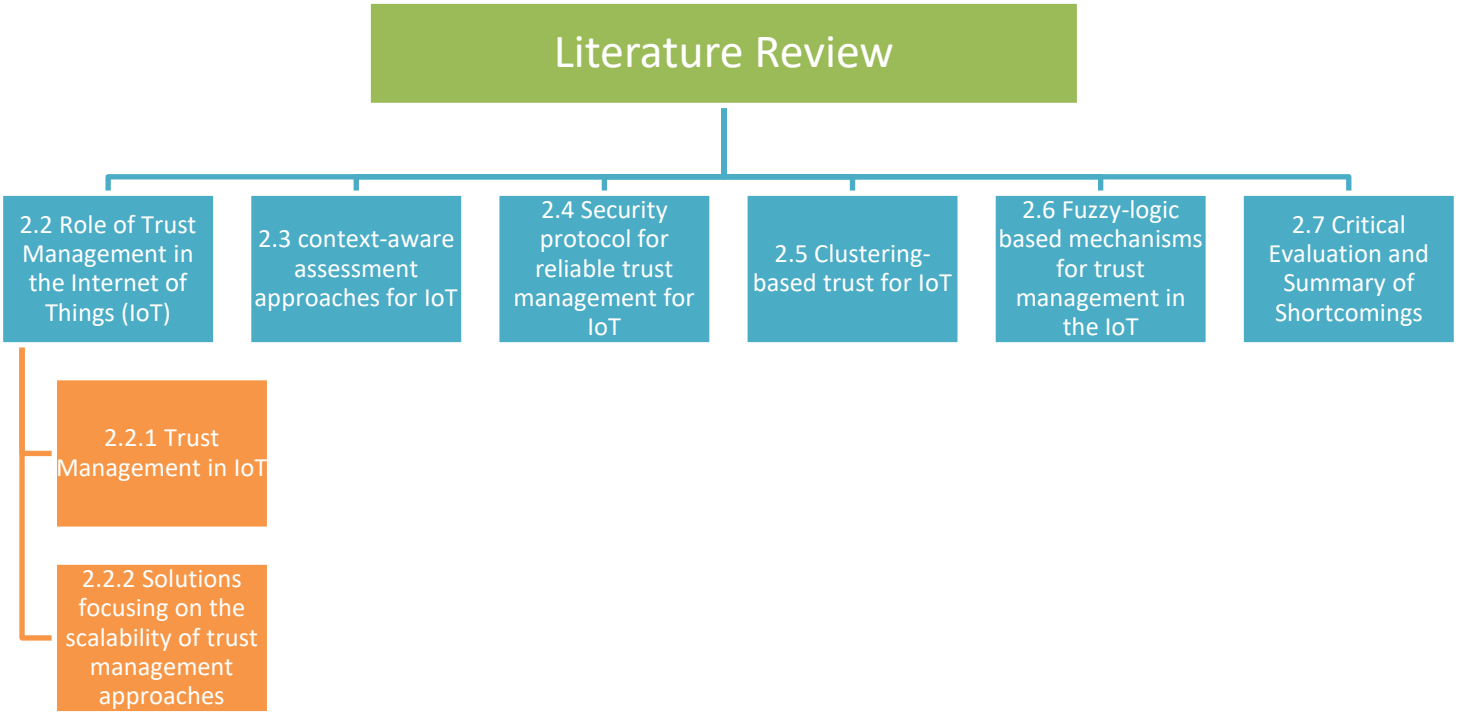

Figure 2-1 Shows an overview of the literature review sections and sub-sections in this chapter

2.2 Role of Trust Management in the Internet of Things (IoT)

In 1999 Kevin Ashton (Neisse et al. 2014) predicted that in the future, there will be devices (things) that will be linked to the Internet, paving the way for the IoT (Khan & Herrmann 2017). He is credited with coining the term in 1999, using it to refer to devices linked to other devices According to (Neisse et al. 2014), the interest accorded to IoT has seen an unprecedented increase due to projections that the number of devices could reach 50 billion by the year 2020.

The International Telecommunication Union released a report describing the mode of connectivity that allows people to be connected anytime, anyplace, for anything (Neisse et al. 2014). IoT connectivity refers to a network of wireless sensors consisting of distributed and varied components which use sensors to monitor neighbouring sensor nodes (Varghese, Chithralekha & Kharkongor 2016). Each device can be distinguished object in the IoT paradigm but is capable of engaging in interoperation within the existing system (Vögler et al. 2016).

The IoT can be described as a set of devices, considered to be smart (or intelligent), interacting collaboratively to fulfil a particular goal (Sicari et al. 2015). The IoT has ushered in a vast array of smart services, including applications used by individuals and organizations in dealing with real – life challenges that they face as they interact and connect with devices anytime, anywhere (Ortiz et al. 2014). With the increasing possibility of infusing smartness everywhere, these devices are being used to connect the physical world where field operations take place and the cyber world where data processing and decisions happen (Jabeur et al. 2017). The interaction between the devices and the physical world through the use of Internet protocols and standards make the collection of data from the environment possible (Bello & Zeadally 2016). In short,

the IoT serves as a universal networking infrastructure that deploys data acquisition devices and communication resources to connect physical and virtual objects (Jabeur et al. 2017) .

Billions of IoT nodes create networks or clusters which are patterned on human social structures and entities (Ortiz et al. 2014). These structures and entities come in the form of houses, airports and highways. The vision of the IoT is to link a myriad of things as they interact with the environment and receive information on the status of this interaction by taking a look at investigating this set of things. The clusters involved in the IoT are characterized by heterogeneity with a need for the development of trust in the interaction of one device with another device as well as the interaction between a device and a user (Hamadeh, Chaudhuri & Tyagi 2017).

However, issues of data security and authentication arise when data is transferred from a cluster to another cluster in the IoT (Miao & Wang 2012). The IoT faces many challenges, particularly in relation to trust management. The heterogeneity of IoT entities, limited storage and the existing trust management protocols do not address this critical issue (Chen, Guo & Bao 2016).

According to (Chen, Guo & Bao 2016), the IoT requires the creation of nodes which need a protocol on trust management. Another issue is that the interconnectedness of IoT networks poses a significant risk since systems could be subject to malicious attacks (Khan & Herrmann 2017).

The increase in data scattered across computing nodes requires aggregated big data, which refers to large and growing data (Jiang et al. 2015). Thousands of smart devices that allow seamless communication as a part of daily life will be prevalent in the future (Sarkar et al. 2015). IoT devices are typically considered as application dependencies engaged in providing external data or processing simple instructions (Vögler et al. 2016). As significant aspects of the smart revolution, these devices are capable of collecting data, sharing information, and initiating and executing services with the least human intervention needed. The IoT is an emerging and improved paradigm, allowing

most of these physical devices to establish a connection with each other (Sarkar et al. 2015).

However, with the rapid increase in the number of devices, the computational power of each IoT device is not sufficient for mining and analysis. It must be noted that IoT servers have computing capacity that enables a large amount of data to be gathered and stored in the existing servers. IoT servers fail to provide enough computing power, nor do they maximize bandwidth in communication, or provide enough storage capacity that is conducive for mining and analysis of data. As a measure to enhance security, keys are used to encrypt and decrypt data through algorithms and policies (Jiang et al. 2015).

The IoT is a computing paradigm where physical components or 'things' are integrated using hardware, software, programming, sensors and networks to enable it to provide value through information sharing with manufacturers, users and/or connected devices (Yan, Zhang & Vasilakos 2014). Each component can be uniquely identified, but it has the capacity to interoperate inside the underlying Internet infrastructure (Uckelmann, Harrison & Michahelles 2011). Some scholars such as (Tselentis, Domingue & Galis 2009) use the phrase ‘Web of Things’ to refer to the IoT. Typically, the IoT provides better integration and connectivity between devices, network infrastructures, systems, and services that span connectivity beyond machine-to-machine links and spreads across a mixed collection of conventions or protocols, applications, and domains (Bao & Chen 2012b). The interconnection between these embedded components coupled with the increasingly growing addition of intelligence or ‘smartness’ in devices is expected to introduce automation in almost all areas, while facilitating better applications such as Smart Grid (Chang & Chen 2012).

From 2014, the IoT has rapidly developed because of the convergence of different technological advancements, including remote connectivity via fault-tolerant networks, wireless communication, embedded systems, and micro-electromechanical systems (Zanella et al. 2014). This implies that the conventional areas of automation, remote

sensors, control systems, embedded systems, and augmented reality all empower the IoT. The idea of a system of intelligent devices has been discussed since the 1980s, when a Coke machine developed at Carnegie Mellon University was able to report on its stock and whether cans of coke which had been recently placed in the machine were cold (Boswarthick, Elloumi & Hersent 2012). Mark Weiser's fundamental paper titled 'The Computer of the 21st Century' on computing anywhere anytime in 1991 gave rise to the expression 'ubiquitous computing' and is another milestone in the IoT. Scholarly venues, for example, UbiComp, PerCom and IEEE Spectrum created the modern concept of Internet of Things (Yan, Zhang & Vasilakos 2014). The concept was further galvanized in 1994 with the conceptualization of 'moving little data packets to a huge collection of hubs', in order to incorporate and computerize everything ranging from personal, home and business appliances to complete factory operations (Sicari et al. 2015). In the period 1993-1996, organizations like Novell proposed solutions like Novell Embedded Systems Technology (NEST). In 1999, the field started gaining momentum with MIT's Auto-ID Center and related corporate sector publications (Ning 2016).

In the IoT, things include but are not limited to wearable devices such as heart monitoring tools, biochip transponders implanted on animals, cars with in-built sensors, electric clams used in coastal water areas, field operation equipment for rescue purposes, radio-frequency identification (RFID) applications, and surveillance devices. These RFID devices are used to gather valuable information. Then, the devices stream the information across other devices in their individual autonomous capacity. Current commercial IoT applications include: intelligent indoor regulator systems; health-oriented wearable devices to screen body temperature; heart rate and other wellbeing issues; spying devices; agriculture and home appliances that use Wi-Fi for remote operation and monitoring. It is anticipated that there will be a huge growth in the number of new application areas into which Internet-based automation will venture, hence, the IoT is expected to create huge volumes of information from disparate areas

that is amassed rapidly. Therefore, there is an increasing need to advance indexing, storage and processing capacity to derive value from the rapidly growing volumes of information (Pfister 2011). Scholars and information management experts have proposed diverse tools and techniques to help analyse this huge quantity of data, referred to as 'big data'. IBM, Dell, Microsoft, Google and other IT leaders have ventured into big data analytics and related technologies to help make better use of this huge amount of data through effective and efficient analysis and reporting (Perera et al. 2012).

While different technologists consider the IoT as a significant step towards a better world, researchers have questions about the guaranteed benefits of the IoT revolution as they see ubiquitous computing as a technology that will undermine confidentiality and the privacy rights of individuals. (Ma, Liu & Zhang 2015) posit that technological innovations have already impacted on our ethical decision making, and consequently our human agency, security, privacy and autonomy. Often, individuals and organizations have found themselves at the receiving end of security and privacy infringement. Organizations have been attacked, leading to millions of dollars in penalties and losses as well as damage to reputation. However, the IoT is a computing paradigm that is out to change the way we live. (Saied et al. 2013) expressed concern about the influence of collecting huge quantities of data without security considerations. (Perera et al. 2012) state that the IoT has and will continue to expose people to privacy issues, especially with the 'big data' concept. As such, the IoT may erode control over our lives. As corporations and governments try to amass huge volumes of data to gain financial advantage and control, chances are it will be difficult to ensure control over our lives (Ray, Abawajy & Chowdhury 2014).

2.2.1 TRUST MANAGEMENT IN IOT

It is clear that the entire world is heading towards a future with the IoT computing paradigm. The IoT connects the physical world into cyberspace everywhere and everything through the use of billions of smart objects (Chen, Guo & Bao 2016). The high level of heterogeneity is expected to bring security threats to the Internet brought about by the interaction of any combination of humans, machines, and robots (Sicari et al. 2015). To address security threats in IoT, there is a need to design a dynamic trust management protocol for IoT systems considering the threat of both malicious and socially uncooperative nodes (Bao & Chen 2012a; Sicari et al. 2015). However, there is little work on the management of trust in the IoT to enhance security, especially in dealing with misbehaving nodes which are currently legitimate members of an IoT community (Ahmed et al. 2016; Bao & Chen 2012a). This has paved the way for additional research pertaining to a trust management protocol.

According to (Chen, Guo & Bao 2016) one of these protocols is trust management which is both adaptive and scalable to support service composition of IoT applications in SOA-based IoT environments . This work features distributed collaborative filtering to select feedback with the use of a devised similarity rating of the relationships detected. They also developed a filtering technique which determines the best way of combining direct and indirect trust to minimize bias in the presence of malicious nodes performing opportunistic services and attacks. To ensure scalability, this work considers a framework in which a capacity-limited node only keeps the trusted information of a subset of nodes of interest and performs a minimum computation to update trust. The method uses a trust protocol with limited storage space. However, the work fails to consider attacker behaviour models that can be carried out by the IoT nodes including bad-mouthing attacks, on-off attacks, contradictory behaviour attacks, opportunistic collusion attacks, random attacks, and insidious attacks so as to make their proposed trust management approach resilient.

The study by (Sicari et al. 2015) uses middleware that involves the employment of infrastructure to deal with security threats in a dynamic environment. This work analyses the most relevant available solutions related to security, privacy, and trust in the IoT field. Their work calls for the need to design and deploy a suitable solution as far as privacy and security requirements are concerned. Issues with the security of the middleware arise. Security protection during the interaction process is also compromised. There is no guarantee of the privacy and security levels. Similar to the above work, this work has not investigated the use of trust-based clustering of IoT nodes. Additionally, the issue of intelligently detecting untrustworthy behaviour by IoT nodes such as bad-mouthing attacks, on-off attacks, and contradictory behaviour attacks etc.

Dynamic Trust Management for IoT Applications is the framework proposed by (Bao & Chen 2012a). In their proposed approach, an IoT nodes carries out and maintains its own trust assessment of the other nodes it has interacted with. Furthermore, in their proposed approach, for scalability, a node may simply restrict its trust evaluation to a limited set of nodes in which it is most interested. The trust management protocol is encounter-based as well as activity-based. Nodes exchange the results of their trust evaluation on other nodes in the form of recommendations. Honesty, cooperation and community interest form a part of the multiple trust properties. The results indicate that this protocol converges to the ground truth status in dynamic IoT environments. However, this work fails to address or evaluate key issues related to IoT trust management such as the scalability of their proposed approach and the resilience of their proposed approach against untrustworthy behaviour by the IoT nodes such as bad-mouthing attacks, on-off attacks etc.

(Ahmed et al. 2016) proposed the Trust and Energy Awareness Secure Routing Protocol (TESRP) for WSN. This protocol makes use of a distributed trust model to identify and seclude misbehaving nodes. TESRP employs a multi-faceted routing strategy that considers the trust level, residual energy, and hop-counts of neighbouring

nodes while making routing decisions. Their proposed approach for making routing decisions ensures data dissemination with trusted nodes as channels while balancing out energy consumption among trusted nodes, traversing through shorter paths. The shortcomings of TESRP are twofold as follows: (a) TESRP focuses on trust-based approach for making routing decisions and it fails to address the issue of scalability of TESRP; (b) They do not present methods by which misbehaving nodes carrying out bad-mouthing attacks, on-off attacks or contradictory behaviour attacks in TESRP can be identified and isolated from the network.

Meanwhile, a survey on trust computation models for the IoT for the purpose of service management was carried out by (Guo, Chen & Tsai 2017). This work classifies the existing trust literature in IoT along the following dimensions: trust composition, trust propagation, trust aggregation, trust update and trust formation. The survey summarizes the advantages and disadvantages of the previous trust models and stresses the effectiveness of defence mechanisms against malicious attacks. The survey concludes that there are identified gaps in IoT trust computations. However, this study not focuses on the issues of scalability of the trust management approaches and their resilience against cyber-attacks.

Another survey by (Yan, Zhang & Vasilakos 2014) involves an investigation of the properties of trust, proposes objectives for IoT trust management, and provides a review of the current literature advances towards a trustworthy IoT. After presenting the findings, the survey proposes a holistic trust management framework for IoT and suggests an extension of the state-of-the-art trust methods in IoT. The proposed approach is comprised of a number of modules for enabling intelligent and trustworthy IoT application/service based on social trust relationships. The survey calls for IoT entities based on trust relationships and privacy in social trust mining. It proposes that lightweight security and privacy solutions should be developed based on trust relationship evaluation. However, similar to TESRP, their propose approach fails to address the issue of scalability of the proposed trust management approach.

Furthermore, they do not present methods by which misbehaving nodes carrying out bad-mouthing attacks, on-off attacks or contradictory behaviour attacks can be identified and isolated from the network.

Meanwhile, a quantitative model of trust value based on multidimensional decision attributes is proposed by (Yu et al. 2017). The direct trust value of the monitored node is measured from a multitude of aspects, including but not limited to the packet forwarding capacity, repetition rate, and consistency of the packet content, delay, and integrity. Having identified the problem that the trust evaluation of nodes is not objective in the existing methods, the model employs the entropy theory. In calculating the indirect trust value, the D-S theory is adopted to deduce and synthesize the trust, and the statistics involved in the nodes' behaviour. The simulation results show that this method can effectively take into account the subjective and objective evaluation of trust, thereby avoiding malicious node. However, this survey places less emphasis on practical needs and demands such as power-efficient/memory-efficient technologies, lightweight trust management, and the ability of IoT nodes to be scalable.

(Ma, Liu & Zhang 2015) propose a reliable trust-based data aggregation protocol, called the ERTDA protocol that is energy efficient. Based on the observations of the nodal behaviour, the ERTDA protocol calculates, monitors, and evaluates the trust values of the nodes. It also detects and excludes the compromised nodes. The simulation results show that the proposed protocol effectively reduces the energy consumption and improves the reliability of the data transmission. A certain proportion of compromised nodes in the network sent the wrong data to the aggregation node. However, this study not addresses the issues of memory efficiency and how to overcome the shortages of IoT node memories.

(Bao, Chen & Guo 2013) proposed the design of a scalable, adaptive, and survivable trust management protocol in dynamic social IoT environments. In the community of interest (CoI), nodes form to communities of interest establishing social connections among entity owners. It identifies the best trust protocol settings in the presence of

changing conditions and malicious nodes that perform trust-related attacks. However, this study has not includes future research endeavours, secure routing and cyber-attacks detection such as on-off attacks, contradictory attacks and bad-mouthing attacks in that applications mentioned in the study.

The work by (Lyu et al. 2015) proposes a trust propagation method for social IoT networks. In their proposed approach, they exploit the strange nature of social networks and incorporates a landmark-based method intended to improve the efficiency of trust prediction. It involves the selection of a small number of social network landmark users. They will serve as referees in trust propagation. Landmark users provide referrals on the trust ratio between two users who are indirectly connected. In evaluating the performance of the proposed method, comprehensive experiments are conducted using a real online social network. The experiment results show that this method demonstrates more efficiency than any of the other four methods in trust prediction. An extension of this work to social networks with more general settings and using larger datasets to evaluate our proposed trust prediction method is suggested in future studies.

(Mosenia & Jha 2017) indicated that the IoT paradigm led to the development of several protocols of communication with the miniaturization of transceivers enabling the transformation of an isolated device. The advances in technology have exponentially increased the number of Internet-connected computing and sensing devices. However, these suffer the consequence of possible attacks and potential threats to privacy and security, especially when the transfer of data from a cluster to another occurs. A variety of approaches have been proposed in addressing the problem of IoT trust management even though it has been argued and acknowledged that the real challenge for trust management is based on scalability. Therefore, there is a need to consider the development of intelligent next-generation methods of trust management for IoT networks accommodating the leaving and joining of nodes, handling large-scale networks.

The work by (Khan & Herrmann 2017) involves the design and evaluation of some IDS mechanisms for IoT networks that are suited to small devices. In this work, a trust management mechanism is used that allows devices to manage reputation information about their neighbours. This mechanism makes it possible to single out maliciously behaving units in a node data processing and energy friendly way. The approach is explained in the context of the healthcare domain. It is quite flexible and can easily accommodate other types of attacks. One of the three algorithms used is only for small networks only. There is a need to facilitate secure identification alternatives and deploy access models to IoT devices. However, similar to the above presented works, this work too has not investigated the use of trust-based clustering of IoT nodes. Additionally, the issue of intelligently detecting untrustworthy behaviour by IoT nodes such as bad-mouthing attacks, on-off attacks, and contradictory behaviour attacks etc.
(Ray, Abawajy & Chowdhury 2014) looked at RFID using the radio frequency identification (RFID) tags through the Internet enables the identification of tags using an appropriate authentication protocol. RFID security protocols and frameworks based on the customizability and scalability of the issues to support IoT implementation. They proposed an identification technique on a group-based and collaborative approach (hybrid approach) and security check handoff.
(Kotis, Athanasakis & Vouros 2018) focus on the trustworthiness of an entity, noting that in a distributed or open IoT environment, there are multiple generic applications and third-party devices that need to be securely deployed. Their suggestion is that the semantic interoperability approaches related to IoT need to be extended through trust semantics. In the IoT, semantics refers to the ability to extract knowledge using various machines for the required services to be provided (Al-Fuqaha et al. 2015). Trust semantics are used in describing the trust-related and quality attributes for the sources and their providers. Since the high heterogeneity level in IoT can magnify the security threats during interactions, it is important to semantically enable trust in an open and

distributed IoT to secure and ensure the deployment and selection of heterogeneous IoT entities without central authorities of trust.

(Ahmed et al. 2016) specifically designed a trust and energy-aware routing protocol (TERP) to address the challenges of trust-based routing protocols. The TERP design is centred on energy efficiency and trustworthiness with the capability of the dynamic detection and isolation of misbehaving nodes during the phase of trust evaluation with the incorporation of the energy awareness feature in the route setup phase of the routing protocol. This helps to better balance the load among trusting nodes. The design also integrates trust-based routing with the additional inclusion of mechanisms to ensure the selection of end-to-end routes with the current energy levels of intermediate nodes. Through evaluations of TERP based on simulations in NS-2, there is indication of better performance regarding the average energy consumption, throughput, and lifetime of the network. However, this study fails to evaluate the scalability of TERP and its resilience against cyber-attacks such as badmouthing, on-off attacks etc. that can be carried out by misbehaving IoT nodes.

A trust-based secure routing protocol was proposed by (Renubala & Dhanalakshmi 2014), showing the flow for fuzzy logic on the basis of the trust mechanism for secured routing. The method utilizes the bio-inspired energy efficient-cluster (BEE-C) protocol characteristics through the consideration of the distance, battery level, and node density. There is also the detection of the black region on the network and the enhancement of the network's security. The shortcomings of this work are similar to that of TERP (above).

(Chen et al. 2011) recognized the use of a trust and reputation model in defending large distributions of sensor networks in IoT/CPS against malicious attacks on nodes, especially because mechanisms of trust establishment can stimulate collaboration among the communication and computing entities. This facilitates the detection of untrustworthy entities and assists the decision-making process in relation to various protocols. The focus is on a fuzzy-theory-based trust and reputation model for the

IoT/CPS environment where the unique features of trust challenges, the concept of trust and reputation, trust evaluation metrics, global trust relationship evaluation and local trust relationship evaluation are analysed. The fuzzy-based secure routing approach provides effective protection to WSN from severe attacks through the dynamic replaying of routing information. The argument is that the reputation mechanism and behaviours based on trust can be used in resolving the problem. This work although it proposes the uses of fuzzy-logic for IoT trust management, it fails to consider or evaluate the scalability of their proposed approach and also the ability of their proposed approach to counter bad-mouthing attacks, on-off attacks etc.

Table 2-1 below presents an overview and comparative evaluation of the common approaches that have been used for trust management in IoT and describes their features and shortcomings.

Table 2-1 An overview and comparative evaluation of the common approaches used for trust management in IoT

Reference	Description of the approach	Features of the approach	Limitations of the approach
(Chen, Guo & Bao 2016)	An adaptive and scalable trust management system to support service composition applications in SOA-based IoT systems is a technique based on distributed collaborative filtering to select feedback using a similarity rating of friendship, social contact, and community of interest relationships as the filter.	Distributed collaborative, novel adaptive filtering technique, and a capacity to keep trust information of a subset of nodes of interest and performs a minimum computation to update trust.	Limited storage tested to address scalability. Fails to test the scalability of the proposed trust management approach. Furthermore, it fails to address other forms of undesirable behaviours by IoT nodes such as bad-mouthing attacks, contradictory behaviour attacks etc.
(Sicari et al. 2015)	The researchers use a middleware that involves the employment of infrastructure to deal with security threats in a dynamic environment. This	Satisfaction of privacy and security requirements is the feature of this approach.	Lake of detecting untrustworthy behaviour by IoT nodes such as bad-mouthing attacks, on-off attacks, contradictory behaviour attacks.

	work analyses the most relevant available solutions related to security, privacy, and trust in the IoT field.		
(Bao & Chen 2012a)	Dynamic Trust Management for IoT Applications This trust management protocol is characterized by a node that maintains its own trust assessment towards other nodes.	Features trust evaluation and management protocols among nodes.	Fails to test the scalability of the proposed trust management approach. Furthermore, it fails to test the protocol's resilience towards untrustworthy attacks by IoT nodes such as bad-mouthing attacks, contradictory behaviour attacks etc.
(Ahmed et al. 2016)	Trust and Energy Awareness Secure Routing Protocol (TESRP) for WSN makes use of a distributed trust model for discovering and isolating misbehaving nodes and nodes traversing shorter paths.	Employs a multi-faceted routing strategy Ensures data dissemination via trusted nodes while balancing out energy consumption among trusted nodes.	Fails to evaluate TESRP in countering untrustworthy behaviour such as bad-mouthing attacks, contradictory behaviour attacks etc. Furthermore, the scalability of TESRP has not been tested or evaluated.
(Guo, Chen & Tsai 2017)	A survey on trust computation models for the IoT for the purpose of service management	Classifies existing trust computation models for service management in the IoT	Does not address the issues of scalability of the proposed solution. Furthermore, it does not address approaches to counter untrustworthy attacks or behaviour by the IoT nodes.

(Yan, Zhang & Vasilakos 2014)	The survey investigates the properties of trust, proposes objectives for IoT trust management and provides a survey of the advances in the current literature towards trustworthy IoT.	Proposes a research model – a holistic trust management framework of IoT	The proposed trust management framework does not address related to trust and privacy. Furthermore, it does not address the issues of scalability of the proposed trust management framework. Finally, it does not address approaches to counter untrustworthy attacks or behaviour by the IoT nodes.
(Yu et al. 2017)	A quantitative model of trust value based on multidimensional decision attributes	Uses the information entropy theory and the D-S theory	No emphasis on practical needs and demands of trust management solutions for IoT network such as power-efficient mechanisms, mechanisms to counter untrustworthy behaviours by the IoT nodes and scalability of the proposed mechanisms.
(Ma, Liu & Zhang 2015)	The ERTDA protocol is an energy-efficient reliable trust-based data aggregation protocol for WSN.	The ERTDA protocol calculates monitors, evaluates the trust values of the nodes, detects, and excludes the compromised nodes in a timely manner.	Does not address issues related to untrustworthy behaviours by the IoT nodes and the scalability of the proposed solution.
(Bao, Chen & Guo 2013)	Scalable, adaptive, and survivable trust management protocol in dynamic IoT environments	Identifies the best trust protocol settings in the presence of trust-related attacks. The node only keeps trusted information of a subset of nodes meeting its interest and performs minimum computation to update trust	Focuses on secure routing and intrusion detection in communities of Robot as a Service, in a cloud computing environment, or in communities of associated smart phones Similar to the above approaches, it does not address issues related to untrustworthy behaviours by

			the IoT nodes and the scalability of the proposed solution.
(Lyu et al. 2015)	A trust propagation method which exploits the peculiar properties of social networks and incorporates a landmark-based method with preprocessing to improve the efficiency of trust prediction	Exploits the peculiar properties of social networks and incorporates a landmark-based method with preprocessing to improve the efficiency of trust prediction.	Focuses on trust propagation models for social IoT. Similar to the above approaches, it does not address issues related to untrustworthy behaviours by the IoT nodes and the scalability of the proposed solution

As can be inferred from Table 2-1, the presented trust management approaches in this table neither address the scalability of the proposed trust management (or evaluate its scalability) nor present intelligent mechanism to detect and counter cyber-attacks such as bad-mouthing attacks, on-off attacks, contradictory etc.

2.2.2 SOLUTIONS FOCUSING ON THE SCALABILITY OF TRUST MANAGEMENT APPROACHES

As an IoT network connects a huge number of devices and applications, there is an increased challenge with respect to meeting scalability, dynamic adaptability and compatibility. It is argued that trust management is beset by the greatest challenge of scalability (Bao, Chen & Guo 2013). According to (Bao, Chen & Guo 2013), scalability is a key consideration in the building of trust or having reliable protocols designed for

management and developing reliable protocols for trust management. IoT networks must evolve to adapt to nodes which are joining and leaving, as they rapidly and accurately build up trust. This work presents a solution for the management of trust in the IoT through the evaluation of two trust-related approaches. In the first model, the researchers designed a protocol where trust relationship data are stored in sets of nodes within the CoI. Nodes that come and go in the process rapidly build up trust towards the others because of the provision of convergence in the CoI framework. Storage is effectively utilized, despite the constrained storage space, to adapt to a large-scale application and increased survivability.

The second model identified in (Bao, Chen & Guo 2013) is equipped with RFID frameworks with a comprehensive security structure. This is to ensure security and scalability in operation. An effective procedure in the ID process is founded on a hybrid framework and is highly sensitive and there is a flexible security handoff for RFID monitoring. The protocol banners are scalable so it can uphold security in the RFID networks. The protocol is embedded with a tool, which has the ability to identify and recognize malware to prevent the introduction of malicious nodes. However, the primary shortcoming of this research is that it fails to propose methods to counter or detect cyber-attacks on IoT trust management approaches such as, badmouthing, on-off attacks, contradictory attacks etc.

In addition, (Bao, Chen & Guo 2013) note that the IoT enables applications such as e-health and smart product management by capturing, processing and sharing data, which necessitates effective trust management protocols to manage the trust between different IoT entities. However, (Ray, Abawajy & Chowdhury 2014) argue that trust management is constrained by the huge number of IoT entities which challenges scalability with respect to accommodating the increasing number of computational and storage entities. In addition, IoT networks should evolve to adapt to nodes that are joining and leaving while building up trust rapidly and accurately. This implies that trust management protocols for IoT networks should be highly resilient to trust-based

attacks such as badmouthing, contradictory behaviour attacks etc… to ensure security in hostile environments. According to (Bao, Chen & Guo 2013), scalability should be a key consideration in the design of trust/reliable management protocols for IoT. However, the trust management protocols proposed in (Bao & Chen 2012b; Ma, Liu & Zhang 2015; Perera et al. 2012; Saied et al. 2013; Yan, Zhang & Vasilakos 2014) do not address scalability, hindering their applicability in large-scale IoT networks. Therefore, it is important to investigate trust management protocols that have been designed to address the scalability challenge.

(Ray, Abawajy & Chowdhury 2014) proposed an IoT protocol framework for RFID-based devices called the scalable RFID security framework and protocol supporting IoT (SRSFPSI). They noted that RFID frameworks need to be installed with a comprehensive security structure for secure yet scalable operation. The proposal entails an effective ID procedure founded on a hybrid framework (group-based and collaborative technique) and a highly adaptive security monitoring handoff for RFID IoT networks. The protocol offers adaptability and scalability, while upholding secure and adaptable RFID networks. In addition to preventing the introduction of malicious nodes and facilitating scalability, the protocol is integrated with a malware recognition tool. However, this proposed SRSFPSI approach does not address the critical issues of how the protocol can be made resilient to untrustworthy behaviours by IoT nodes such as bad-mouthing, on-off attacks, contradictory attacks etc.

(Gupta & Awasthi 2010) argue that trust management and quality of service is a vital step in securing peer-to-peer environments characterized by frequent encounters with unknown agents. They propose a scalable protocol for a peer-to-peer IoT framework that is founded on existing IoT principles of trust management and reputation at semantic and data management levels. To establish tangible levels of scalability, there is no central database so as to promote global knowledge sharing as a means of evaluating earlier interactions. The approach scales well to meet trust management demands for a large set of nodes, a feat achieved due to implementation of completely

peer-to-peer decentralized IoT systems. While their proposed IoT trust management approach is scalable, it fails to address issues related to its resilience against cyber-attacks by the untrustworthy IoT nodes. These cyber-attacks could be bad-mouthing attacks, on-off attacks, contradictory attacks etc.

There is a need for data scalability in any data sharing scheme by utilizing clusters of small blocks in the data transfer (Neisse et al. 2014). This restriction ensures that no re-encryption of blocks takes place during the process (Jiang et al. 2015). There is a need to reduce the size of data to ensure adaptability in a scalable data management system (Neisse et al. 2014). The rapid growth of integrated technologies has led to growth in the use of devices, which, in turn, has led to a rapid growth in data (Sarkar et al. 2015). This paves the way for the emergence of the challenge of the influx of devices and the information they generate which is challenging. Different approaches have recently been developed to address trust scalability for the IoT. Since 2014, attempts to develop the IoT have rapidly increased because of the large number of breakthroughs involving data from device-to-device connectivity to microelectromechanical systems (Vögler et al. 2016).

Currently, IoT devices requires manual configuration, making them resistant to change, both in terms of application and the infrastructure requirements (Jiang et al. 2015). In large-scale IoT systems, for instance, efforts have been made to leverage the processing capabilities of gateways in order to tap into the possibility of using them to improve resilience, reliability, and performance (Vögler et al. 2016).

The work in by (Vögler et al. 2016) uses a framework called LEONORE, which is based on the cloud and follows the design of the micro-service architecture enabling applications that are scalable. Several strategies are proposed in this work to address the issue of the immense work load. The framework enables deployments that support a variety of IoT topology and model requirements. This research proposes efficient deployments in constrained environments to further improve scalability and reduce network traffic, which is generated by the cloud and infrastructure used. This study

although it focuses on developing a scalable solution for IoT trust management, it fails to consider how the trust management solution can be made resilient against cyber-attacks such as bad-mouthing attacks, contradictory attacks, on-off attacks and bad service attacks.

The work in by (Sarkar et al. 2015) termed as Distributed Internet-like Architecture (DIAT) uses a system that affords security and scalability for accumulating data in the IoT using a tool for multi-coefficient utilization. In this work, files are transferred into shares and given to peers who are located in the storage system of the IoT. The architecture of DIAT is layered and distributed (Sarkar et al. 2015). The architecture is also capable of tackling various technical challenges such as heterogeneity, scalability, and interoperability. The protocol accommodates objects that are heterogeneous and implements security and privacy aspects using data usage control policies. Automation, intelligence, dynamicity, and zero configuration form part of DIAT. However, DIAT does not address issues related how it can be made resilient to untrustworthy behaviours such as, bad-mouthing, on-off attacks, contradictory attacks and bad service attacks by the IoT nodes.

Another work that involves the scalability of IoT is the dynamic context-aware scalable and trust-based IoT security privacy protocol in (Neisse et al. 2015). The protocol uses SecKit enforcement components. Progress is monitored by the Policy Enforcement Point. This framework uses a number of mechanisms as a part of its solution such asenforcement rules, middleware equipped with privacy-preserving behaviour, mechanisms for context specification, context adaptation, etc. Similar to the above approaches in this section, this approach fails to develop intelligent approaches for countering attacks such as bad-mouthing attacks, contradictory attacks, on-off attacks and bad service attacks.

This work by (Li et al. 2013) focuses on the PaaS framework provides the efficient IoT services delivery and continuous extension of services on the cloud. The framework offers two types of services, event processing and data services. These services handle

real-time events and persistent data respectively. The former produces data flows and detects interesting patterns or events according to data users' specifications and the latter facilitates the storage, retrieval and manipulation of persistent data while hiding the specifics of the underlying database systems. The platform efficiently delivered new solutions by leveraging computing resources and platform services such as domain mediation, application context management, and metering on the cloud. The mediators of the domain provide a mechanism for the IoT PaaS for engagement with various domain-specific models of data and provide control applications, which is primarily dependent on physical devices. Similar to the literature presented above, this work did not address approaches to counter cyber-attacks on trust management approaches.

Whether cloud environments are scalable or not has not yet been fully determined. This issue is the concern of the design and establishment of pub/sub systems and Named Data Networking (NDN) (Han & Woo 2016). NDN features distance vector routing that enables QoS in routing, the hierarchical routing for establishing the scalable topology, and the delivery of multicast data for the efficient reduction of messages. IoT cloud workloads, such as home control and personal health monitoring are the focus of future research endeavours. The shortcomings of this work are similar to the shortcomings of the work presented in this section.

The work by (Kokoris-Kogias, Voutyras & Varvarigou 2016) introduced a model TRM-SIoT where trust is derived from what is experienced by each node. This model uses distributed and centralized T&R architectures for the creation of a hybrid one, resembling a human entity or authority. This model combines solutions proposed for peer-to-peer and mobile ad-hoc networks and applies them to the IoT environment. Nodes are used to model human behaviours and relationships during the interaction. The system experiences security breach as the metric calculation is exploited to obtain a higher trust level. However, this proposed TRM-SIoT approach does not address the critical issues of how the protocol can be made resilient to untrustworthy behaviours by IoT nodes such as bad-mouthing, on-off attacks, contradictory attacks etc.

The work by (Bellavista & Zanni 2016) presents an innovative architecture that is distributed by combining MQTT and CoAP protocols to make gateways scalable to efficiently integrate the IoT cloud. The results indicate a guaranteed scalable support for constrained devices. High message delivery rate was achieved in busy network conditions without anomalies. The model lacks usage optimisation of CPU and memory and security support through DTLS. However, this work does not consider how the proposed protocols can be secured from cyber-attacks such as, on-off attacks, bad-mouthing attacks, etc.

The work in by (Gharbieh et al. 2017) proposes a spatiotemporal mathematical model that is sensitive to traffic. This protocol is used by IoT devices to support cellular uplink connectivity. It is based on the combination of the stochastic geometry and queueing theory to monitor the traffic requirements per device as well as the different transmission strategies involved in the process and the mutual interference between the devices involved in the IoT. The analysis primarily depends on a consistent solution from solving the DTMC. This work addresses elements of IoT scalability; however, it fails to address issues related to the resilience of proposed approach against cyber-attacks such as bad mouthing etc.

The work in (Ray, Abawajy & Chowdhury 2014)proposes a hybrid approach and security check handoff (SCH) for RFID systems with mobility. This model provides security, adaptability, customizability, and scalability in RFID deployment. It features a technique for malware detection, however, this model lacks the mechanism of tag tamper resistance.

Table 2-2 presents an overview and comparative evaluation of the scalable solutions and approaches used for trust management in IoT and describes their features and shortcomings.

Table 2-2 Overview and comparative evaluation of trust management approaches in IoT focusing on scalability

Reference	Description of the approach	Features of the approach	Limitations of the approach
(Bao, Chen & Guo 2013)	Adaptive and survivable trust management for the Community of Interest (CoI)-based IoT. Also a scalable trust framework for a highly mobile RFID-based IoT network is presented	Applicable in RFID IoT networks; incorporates malware detection capacity; ensures scalable implementation of RFID nodes for a distributed IoT.	Designed for RFID IoT networks only and is not applicable for IoT networks. Furthermore, does not address issues centred on the resilience of their approach against cyber-attacks such as bad mouthing, on-off attacks, contradictory attacks and bad service attacks.
(Sarkar et al. 2015)	DIAT is a simple, scalable architecture for the IoT. It accommodates heterogeneous objects and provides support for interoperability.	Features include automation, intelligence, dynamicity, and zero configuration	The scalability of this approach is evaluated under a limited number of nodes. However, it does not address issues centred on the resilience of their approach against cyber-attacks such as bad mouthing, on-off attacks, contradictory attacks and bad service attacks.
(Neisse et al. 2014)	Dynamic context - aware scalable and trust - based IoT security and privacy framework. This framework operates on security and privacy policies	Uses SecKit enforcement components. Progress is monitored by Policy Enforcement Point	The scalability is of this approach is evaluated under a limited number of nodes. However, it does not address issues centred on the resilience of their approach against cyber-attacks such as bad mouthing, on-off attacks, contradictory attacks and bad service attacks.

(Li et al. 2013)	The PaaS framework provides the necessary platform services for IoT solution providers for efficient delivery and continuous extension of services.	Features event processing and data service.	Similar to the above approaches listed in this table, it does not address issues centred on the resilience of their approach against cyber-attacks such as bad mouthing, on-off attacks, contradictory attacks and bad service attacks.
(Han & Woo 2016)	NDN-based pub/sub system for a scalable IoT cloud. The approach concentrates on how to leverage the essential scalability of NDN for building pub/sub systems, thereby achieving scalable IoT cloud services.	Distance vector routing for enabling QoS in routing, hierarchical routing for building the scalable overlay topology, and multicast data delivery for efficient message reduction.	It does not address issues centred on the resilience of their approach against cyber-attacks such as bad mouthing, on-off attacks, contradictory attacks and bad service attacks.
(Kokoris-Kogias, Voutyras & Varvarigou 2016)	TRM-SIoT ensures that trust is derived from what is experienced by each node.	Centralized and distributed T&R architectures	This approach mentions of the scalability of IoT nodes, without discussing how scalable and cyber-attacks resilient trust management solutions can be developed.
(Bellavista & Zanni 2016)	This research provides models enhance scalability gateways in IoT with cloud interactions.	Combines machine-to-machine industry-mature by using MQTT and CoAP protocols.	Similar to the above approach, if fails to consider how scalable and cyber-attacks resilient trust management solutions can be developed.
(Gharbieh et al. 2017)	Traffic-aware spatiotemporal mathematical model intended for use by IoT	Supports cellular uplink connectivity	Similar to the above approach, if fails to consider how scalable and cyber-attacks resilient trust management solutions can be developed.

As can be inferred from Table 2-2, some of the presented trust management approaches in this table, mentioned the issue of IoT trust management scalability (and IoT scalability in general), without developing a method to achieve robust scalability. Furthermore, none of the approaches presented a comprehensive IoT trust management approach that encompasses solution for the scalability of the trust management approach and solution for its resilience against cyber-attacks such as bad-mouthing attacks, on-off attacks, contradictory attacks etc.

2.3 CONTEXT-AWARE ASSESSMENT FOR IOT

In IoT networks, context awareness is the capacity to use environmental and situational data to predict instantaneous needs and offer relevant proactive responses (Perera et al. 2012). IoT consists of the following technologies: embedded sensors, smart mobile devices, cloud computing, and big data analytics, all working collaboratively to collect, model and guide users. Modern computers, networks and the Internet are completely dependent on people for data. The greater percentage of the approximately 50 terabytes of information accessible on the Internet is a result of human efforts such as typing, recording, taking digital pictures or scanning (Ning 2016). The challenge is: humans have constrained time, accuracy, memory, and attention, all implying that they are relatively poor at capturing information about real-world things (Bao & Chen 2012a). In the IoT paradigm it is possible to leverage information about all IoT devices (things) and to track and check everything of their activities and significantly reduce waste and costs. In addition, it would be possible to identify things that require replacement, repair, and review or are obsolete (Saied et al. 2013).

(Bao & Chen 2012a; Bao, Chen & Guo 2013; Ma, Liu & Zhang 2015) proposed trust management protocols that do not address the context awareness issue. (Perera et al. 2012) argue that stakeholders in the IoT–mobile, wearable and ubiquitous computing,

have recognized the need to forego convention desktop models as an increasing number of devices become mobile. As such, all services should be extended and enhanced to adapt to constantly changing contexts, but this complicates the implementation of trust management protocols in the IoT. (Saied et al. 2013) claimed that developing context-aware enabling technologies requires a well-defined security framework for IoT networks, whereby nodes are secure despite cutting across different settings transportation, home, office and others. According to (Perera et al. 2012), network reactions in relation to user mobility and settings should be adjusted to meet different needs though real-time learning and monitoring to bolster precision.

(Perera et al. 2012) proposed the context awareness for the Internet of Things (CA4IOT) framework that is based on automated filtering, synthesis, saving and reasoning in the realm of sensor data collection and the creation of meaningful information from the raw data. It understands and maintains context data about sensors (such as location, nearby sensor, battery life and sampling rate) using appropriate annotations for quick retrieval. Relationships about different domains are learned from knowledge bases that amass information. The CA4IOT framework follows a layered architecture, consisting of: the user - device owner, application or service; user management; processing; reasoning; context discovery; data acquisition; and sensing.

(Saied et al. 2013) proposed a context-aware trust management system for the IoT (CTMS4IOT), which adds some elements of adaptability to meet the challenges of today's dynamic IoT networks. The proposed model entails the following phases: information gathering, entity selection, transaction, reward and punish, and learning. For trust management, the approach uses past behaviour and allows for fine-tuning to overcome challenges brought about by malicious nodes. It uses centralized trust management servers and prioritizes the context where evaluations are captured; therefore, appropriate trust management servers return context information with trustworthy values for each node. The server discovers the context of a node (say node a) based on the context similarity computed from interaction information received from

other nodes about node a. Learning entails dynamic reactions to apply new security strategies based on contextual (environment or location) change.

Table 2-3 presents an overview and comparative evaluation of the common approaches that have been used for context-aware assessment for IoT also describes their features and shortcomings.

Table 2-3 Overview and comparative evaluation of context-aware assessments in IoT

Reference	Description of the Approach	Features of the approach	Limitations of the approach
(Perera et al. 2012)	A framework based on automated filtering, synthesis, saving and reasoning in sensor data collection and reasoning to derive valuable information.	Supports learning by understanding and maintaining context data in knowledge bases; uses appropriate annotations for quick retrieval; follows a layered architecture.	Relies on a dedicated server to facilitate knowledge sharing, thus is subject to the single point of failure that may challenge trust management; poor in scaling to a large number of nodes. Furthermore, does not presents approaches for detecting and countering cyber-attacks by malicious IoT nodes.
(Saied et al. 2013)	A context-aware distributed trust management system designed to address trust issues based on contextual information and learning.	Its operation is based on five phases: information gathering, entity selection, transaction, reward and punish, and learning; allows for fine tuning to meet disparate contextual constraints; modelled on a centralized server setting; support for learning.	Use of centralized trust management servers constraints scalability. Does not address issues attacks on trust management approaches such as bad-mouthing attach, on-off attacks, contradictory attacks etc.

As can be inferred from Table 2-3, the thrust of the presented approaches is on context-aware trust modelling. They fail to address the scalability issues of the trust management solution and its resilience against cyber-attacks such as bad-mouthing attacks, on-off attacks, contradictory attacks etc.

2.4 SECURITY PROTOCOL FOR RELIABLE TRUST MANAGEMENT FOR IOT

The IoT coordinates a huge amount of day-to-day devices operating in heterogeneous networks, creating a serious problem with regard to reliability and security management; notwithstanding, things under an IoT system need to agreeably interoperate (Yaqoob et al. 2017). Reliability of the trust management solution can be compromised by IoT nodes carrying out cyber-attacks such as bad-mouthing attacks, on-off attacks, contradictory attacks etc. The proposed trust management solution should be resistant to such attacks and should be able to detect such untrustworthy attacks by the IoT nodes. To counter these attacks and ensure there is a need for security mechanisms built in as a part of trust management solutions. I term such security safeguards against misbehaving IoT nodes as security protocols.

Further, (Boswarthick, Elloumi & Hersent 2012) underscore the trust management capacity established in the IoT with respect to the identification of trustworthy nodes. They fail to address issues related to the scalability and resilience of their proposed approach.

(Bao & Chen 2012a) propose a trust management protocol for IoT frameworks. The protocol has two goals: to give an exact and flexible trust evaluation on the trust levels of IoT components; and to use the proposed protocol on different IoT applications to optimize application performance. The trust management protocol models a community-oriented social IoT setting by working with many social relationships across device owners. They claim that social trust is obviously expected in such an environment. The system does not have a specialized trusted authority but spreads the role of trust evaluation across individual nodes. Similar to the above approach, they fail to address issues related to the scalability and resilience of their proposed approach.

(Chen et al. 2011) proposed a fuzzy-oriented trust management protocol for use in the IoT consisting of wireless sensors only. In addition, the protocol used Quality of Service (QoS) trust parameters, such as energy utilization and packet transfer to delivery ratio. Sensors create direct communication links among themselves using the IPv6 over Low power Wireless Personal Area Networks (6LoWPAN) protocol, a protocol used for IPv6 networking in devices with low data rates and low power radio transmission. A reputation and trust framework is a critical means to preventing malicious nodes from accessing vast sensor IoT networks, because trust creation instruments can empower coordinated effort across distributed things, support the discovery of malicious components, and facilitate the decision-making process. However, this proposed approach of fuzzy trust management protocol does not address the issue of cyber-attacks such as, on-off attacks, bad-mouthing attacks, contradictory attacks and bad service attacks.

(Ma, Liu & Zhang 2015) proposed the energy-efficient protocol of reliable trust-based data aggregation (ERTDA), which is also founded on the idea of a trust framework and reputation. It reduces the nodes' energy consumption using an effective routing and recovery approach. Path selection is also used to realize security and reliability in data segregation. The protocol ensures that security is upheld in data capturing, processing and sharing in addition to identifying mutual trust relationships between nodes and

exclude compromised components from the IoT network. Consequently, there are three dimensions of the nodes: every group of aggregated nodes should have its security guaranteed and have adequate energy to support aggregation and data sharing; link availability is ascertained based on the energy in neighbouring nodes; and the outcome of data aggregation is important to allow the selection of multiple paths. However, this proposed protocol ERTDA does not address the issue of how the protocol can be made resilient to untrustworthy behaviours by IoT nodes such as bad-mouthing, on-off attacks, contradictory attacks etc.

By looking at machine-to-machine (M2M) networks, (Tuna et al. 2017) examined security requirements such as resilience and availability against external entity attacks by increasing privacy and anonymity in the devices. The proposition is based on using a technique derived from the Information Control Theory, tagging data, and providing various properties of privacy. However, they suggest the need for an overall integrated security approach to ensure that in M2M applications, there is end-to-end security. Furthermore, this survey does not pay attention to the cyber-attacks that can be carried out by misbehaving IoT nodes such as. These attacks could take the form of on-off attacks, bad-mouthing attacks, contradictory attacks and bad service attacks.

(Lin et al. 2017) focused on fog/edge computing so that devices with computing services can be deployed at the network edge with the aim of improving the user experience and the resilience of the services when failures occur. Using the advantage of close to end-users and distributed architecture, the approach provides greater quality of service for IoT applications and a faster response. This makes it suitable for future IoT infrastructure to cover the privacy and security issues in the intelligent cyber world. However, this study does not address the issues of how the protocol can be made resilient to untrustworthy behaviours by IoT nodes such as bad-mouthing, on-off attacks, contradictory attacks etc.

In looking at the security of IoT frameworks, (Ammar, Russello & Crispo 2018) considered 8 frameworks based on their architecture, the compatible hardware, the

essentials of third-party smart app development, and the security features. Through the comparison, they showed that there are similar standards used in communication security while different methodologies are used in the provision of other properties of security. However, this proposed framework does not focus on the trust management protocol for detecting cyber-attacks of IoT nodes such as, on-off attacks, bad-mouthing attacks, contradictory attacks and bad service attacks.

(Wang et al. 2018) proposed the self-trustworthy and secure Internet protocol (T-IP) for encrypted and authenticated network layer communications. This approach is advantageous because it reserves the significant merit of IP to be stateless, with low connection latency and transmission overhead, a self-trustworthy IP address, and is compatible with the existing TCP/IP architecture. However, similar to the other approaches presented in this section, the proposed protocol T-IP does not address the issue of cyber-attacks such as, on-off attacks, bad-mouthing attacks, contradictory attacks and bad service attacks.

(Malina et al. 2016) focused on the cryptographic mechanisms that could be useful and efficient on devices. They noted that the security solutions designed for IoT environments need to deal with heterogeneous entities with different specifications of software. Such devices use the Constrained Application Protocol (CoAP) where they are required to provide authorization, authentication, confidentiality, data authenticity, freshness and integrity. By looking at the performance analysis of cryptographic primitives and memory limitations, they examine and discuss the applicability of privacy enhancing schemes and protocols. However, this research does not address the issues of how the protocol can be made resilient to untrustworthy behaviours by IoT nodes such as bad-mouthing, on-off attacks, contradictory attacks etc.

The work by (Miao & Wang 2012) uses the rapid identification authentication protocol for mobile nodes in the IoT with privacy protection. This protocol simplifies the authentication process, improves efficiency and realizes the credible transmission of mobile nodes among clusters. The proposed protocol meets both the traceability and

forward privacy properties and incurs less communication overhead. However, similar to the above presented works, this work too has not investigated the use of trust-based clustering of IoT nodes. Additionally, the issue of intelligently detecting untrustworthy behaviour by IoT nodes such as bad-mouthing attacks, on-off attacks, contradictory behaviour attacks etc. is not addressed in this work.

(Nguyen, Laurent & Oualha 2015) discussed the applicability and limitations of using IP-based Internet security protocols and other protocols of security used in WSNs and have the suitable potential of use in IoT. (Granjal, Monteiro & Silva 2015) also conducted a survey of the existing protocols and mechanisms securing IoT communications. Their analysis indicates how the approaches in place ensure the significant requirements of security and the protection of communications on the IoT. As for (Raza et al. 2014), their focus is on exploring the option of IPsec use as a security mechanism for IoT. They present a 6LoWPAN/IPsec extension and indicate the approach's viability, finding that the IPsec is feasible in securing IoT. However, this study does not focus on addressing the trust and security issues for the cyber-attacks of the IoT nodes behaviour such as, on-off attacks, bad-mouthing attacks, contradictory attacks and bad service attacks.

Table 2-4 shows an overview and comparative evaluation of the common approaches that have been used for security and trust protocols for IoT and describes their features and shortcomings.

Table 2-4 Overview and comparative evaluation of the existing work on Security protocol for reliable trust management for IoT in the literature

Reference	Description of the approach	Features of the approach	Limitations of the approach
(Boswarthick, Elloumi & Hersent 2012)	The researchers used a novel paradigm for ubiquitous computing beyond the IoT, underscore the trust management capacity established in the IoT with respect to the identification of trustworthy nodes.	This proposed work able to deal with IoT nodes in the trusted level of communication.	There is a need to address energy-efficiency as a primary design goal. There is the need for the interoperability among devices and users. Furthermore, they have not investigated approaches to ensure the scalability of the proposed approach. Finally, they are not address the issue of intelligently countering untrustworthy behaviour by IoT nodes such as bad-mouthing attacks, on-off attacks, contradictory behaviour attacks etc.
(Bao & Chen 2012a) same!	The researchers propose a trust management protocol for IoT frameworks.	The protocol has two goals: to give an exact and flexible trust evaluation on the trust levels of IoT components; and to use the proposed protocol on different IoT	Does not address the issues related to scalability of the proposed solution. Furthermore, does not proposed approaches to counter untrustworthy

		applications to optimize application performance. The trust management protocol models a community-oriented social IoT setting by working with many social relationships across device owners.	behaviour of the nodes and to eliminate the misbehaving nodes.
(Chen et al. 2011)	The researchers proposed a fuzzy-oriented trust management protocol for use in the IoT consisting of wireless sensors only.	In this work consider using the protocols of communications to apply that on the IoT nodes such as, IPv6 over (6LoWPAN) protocol.	Does not address the issues related to cyber-attacks behaviour of the IoT nodes such as, on-off attacks, bad-mouthing attacks, contradictory attacks and bad service attacks.
(Ma, Liu & Zhang 2015)	The researchers proposed the energy-efficient protocol of reliable trust-based data aggregation (ERTDA), which is also founded on the idea of a trust framework and reputation.	The protocol ensures that security is upheld in data capturing, processing and sharing in addition to identifying mutual trust relationships between nodes and exclude compromised components from the IoT network.	Primarily deals with the authentication of the IoT nodes. Does not address any issues related to trust management protocol in IoT.
(Tuna et al. 2017)	The researchers proposed protocol for M2M based on using a technique derived from the Information Control Theory, tagging data, and providing various properties of privacy.	This proposed approach examined security requirements such as resilience and availability against external entity attacks by increasing privacy and anonymity in the devices.	Similar to the above approaches in this table, they do not address how untrustworthy behaviour by the IoT nodes such as bad-mouthing attacks, on-off attacks, contradictory attacks and bad service attacks

			can be detected for the trust management approach.
(Lin et al. 2017)	This work involves using the advantage of close to end-users and distributed architecture, the approach provides greater quality of service for IoT applications and a faster response.	This research focused on fog/edge computing so that devices with computing services can be deployed at the network edge with the aim of improving the user experience and the resilience of the services when failures occur.	They focus on communication protocols for IoT, Fog and Edge computing. No focus on trust management approaches.
(Ammar, Russello & Crispo 2018)	An analysis of a systematic review of articles on the IoT, security aspects specifically at the privacy level and control access in this type of environment.	This work considered 8 frameworks based on their architecture, the compatible hardware, the essentials of third-party smart app development, and the security features.	This is a systematic review paper. No new solution for IoT trust management and its security has been presented.
(Wang et al. 2018)	This research proposed the self-trustworthy and secure Internet protocol (T-IP) for encrypted and authenticated network layer communications.	It is reserving the significant merit of IP to be stateless, with low connection latency and transmission overhead, a self-trustworthy IP address, and is compatible with the existing TCP/IP architecture.	The proposed protocol T-IP does not address the issue of cyber-attacks on the trust management solution.

(Malina et al. 2016)	This work focused on the cryptographic mechanisms that could be useful and efficient on devices.	This research noted that the security solutions designed for IoT environments need to deal with heterogeneous entities with different specifications of software by using protocol (CoAP).	They focus on communication protocols for IoT. No focus on trust management approaches.
Miao & Wang 2012	The model used by the researchers is the rapid identification authentication protocol for mobile nodes in the IoT with privacy protection. This protocol simplifies the authentication process, improves efficiency, and realizes credible transmission of mobile nodes among clusters.	The presented protocol contains a valid request message and an answer authentication message, which rapidly implements identification authentication and privacy protection.	Primarily deals with the authentication of the IoT nodes. Does not address any issues related to trust management protocols in IoT.

As can be inferred from Table 2-4, some of the presented protocols in this table addressed the issues of IoT trust management scalability. However, none of the approaches presented a comprehensive IoT trust management approach that encompasses ***both*** solutions for the scalability of the trust management approach and solution for its resilience against cyber-attacks such as bad-mouthing attacks, on-off attacks, contradictory attacks etc.

2.5 CLUSTERING-BASED TRUST FOR IOT

Since the IoT is basically an opportunistic network where not all real-world things (RWTs) are IP-enabled, an effective topology management approach is needed to allow the RWTs to optimize their communications, save their commonly limited energy resources, and increase their awareness of neighbouring peers and services (Jabeur et al. 2017). However, current research and development efforts on clustering in wireless distributed computing only focus on the extensive work done in the field of WSNs (Sarobin & Ganesan 2016) and has not been applied or carried out in IoT These efforts, however, fail to address or take into account the restrictions on power and processing capabilities of IoT (Bello & Zeadally 2016; Jabeur et al. 2017).

To overcome the limitations that are mentioned above, it is recommended that sensors, and more generally RWTs, are organised into clusters and collaborate to ultimately achieve the goals, thereby exceeding their own competencies. According to (Jabeur et al. 2017), clusters are created based on the distances between cluster-heads and base stations, the distributions and sizes of CHs, the residual energies of sensors, and the number of allowed CHs and on the spatial location of CHs, their connectivity degrees, and their semantics. Various works have been attributed to clustering-based trust in the IoT.

The work by (Jabeur et al. 2017) used a new firefly-based clustering approach for IoT applications. The researchers extended the approach to allow the IoT clusters to self-adapt by hiring and/or firing RWTs depending on ongoing events and their expected impact on the network and its current deployment area. All fireflies are unisex so that one firefly will be attracted to another firefly regardless of gender. The approach includes a micro-clustering phase during which RWTs complete and self-organize into clusters. These clusters are then polished during a macro-clustering phase where they compete to integrate small neighbouring clusters. The clustering approach used in the study comprises four steps: initialization, fetching, intimidation, and polishing. The

results indicate that the number of clusters tends to stabilize independently from the density of the network and the various communication ranges of RWTs. Some performance issues still need to be fixed such as the failure of the firefly approach to allow RWTs to attract other devices based on additional criteria, such as their energy, semantics, and quality of their services. However, they have not investigated the use of trust-based clustering of IoT nodes. Furthermore, they are not address the issue of intelligently countering untrustworthy behaviour by IoT nodes such as bad-mouthing attacks, on-off attacks, contradictory behaviour attacks etc.

The work by (Ortiz et al. 2014) uses a novel paradigm known as the Social Internet of Things (SIoT) which combines the IoT with SNs. This seamless integration is intended to bring new relationships, allowing the creation of novel services and applications that will be of great interest both to the end users and stakeholders. Many of the capabilities of SIoT will be compromised because available devices cannot be connected directly to the internet. There is a need to address energy efficiency as a primary design goal. There is a need for interoperability among devices and users. Automatic network management, autonomic data analysis, and service discovery and composition for an enhanced user experience were not specifically addressed in the development of the approach. Efficient adaptation to challenging situations and correct architectural organization supporting redundancy at several levels has not been properly addressed, affecting the reliability transmitted to the end user. However, similar to the above work by (Jabeur et al. 2017) they have not investigated the use of trust-based clustering of IoT nodes. Additionally, the issue of intelligently detecting untrustworthy behaviour by IoT nodes such as bad-mouthing attacks, on-off attacks, and contradictory behaviour attacks etc.

The work by (Varghese, Chithralekha & Kharkongor 2016) uses the self-organized cluster-based energy efficient meta-trust model for the IoT. In this model, clusters are formed to decrease energy consumption, which in turn, provides scalability and also helps to eliminate selfish nodes. A pattern matching trust model is used to ensure full

trust in the nodes while considering the identity of the node in order to punish nodes that display malicious behaviour so that they do not intrude in the network in the future. The network performance in terms of scalability and energy saving in order to prevent the selfish behaviour of the nodes and to eliminate the misbehaviour of nodes is considered a flaw of the model. The results of this work indicate that the proposed system increases scalability in the network with the clustering mechanism as well as improves trust management by providing a self-organized knowledge-based decision method for each node to examine the trust in another node. The proposed work provides a solution for scalability and the malicious and the selfish behaviour of the nodes is eliminated by the meta-trust model. It was also observed that the rate of data sent is high without any packet loss. While this approach addresses the issue of scalability of trust management solutions for IoT, it fails to address issue of intelligently detecting untrustworthy behaviour by IoT nodes such as bad-mouthing attacks, on-off attacks, contradictory behaviour attacks etc.

The work by (Hamadeh, Chaudhuri & Tyagi 2017) presents a minimal distributed trust layer based on distributed consensus-like operations. These distributed primitives are cast in the context of the APIs supported by a trusted platform module (TPM). The overall TPM functionality is distributed among several IoT devices within a cluster. Results can be hampered by the poor computing capacity and energy of IoT devices. Similar to the above presented works in this section, this work too has not investigated the use of trust-based clustering of IoT nodes. Additionally, the issue of intelligently detecting untrustworthy behaviour by IoT nodes such as bad-mouthing attacks, on-off attacks, and contradictory behaviour attacks etc.

The work (Sarobin & Ganesan 2016) proposes a novel bio-inspired cluster-based deployment algorithm for energy optimization of the WSN and to improve the network lifetime. Using the proposed algorithm, the sensor node reaches its maximum residual energy and with an optimized distance to the sink node. The algorithm optimizes CHs' energy consumption, extending the network lifetime. The shortcomings of the above

outlined approaches in this section applied to the work by (Sarobin & Ganesan 2016) as well.

The emergence of IoT has prompted work that involves not only trust scalability and trust management but also clustering-based trust. However, current research and development trends show that not much work has been done on clustering-based trust in IoT (Jabeur et al. 2017).

Table 2-5 presents an overview and comparative evaluation of the clustering approaches based on trust for IoT and describes their features and shortcomings.

Table 2-5 Overview and comparative evaluation of clustering-based trust for IoT

Reference	Description of the approach	Features of the approach	Limitations of the approach
(Jabeur et al. 2017)	The researchers used a new firefly-based clustering approach for IoT applications. The researchers extended the approach to allow the IoT clusters to self-adapt by hiring and/or firing RWTs depending on ongoing events and their expected impact on the network and its current deployment area.	The approach includes a micro-clustering phase during which real-world things (RWTs) compete and self-organize into clusters. These clusters are then polished during a macroclustering phase where they compete to integrate small neighbouring clusters. The clustering approach used in the study comprises four steps: initialization, fetching, intimidation, and polishing.	Although they discuss about clustering of IoT nodes, they fail to investigate the use of trust-based clustering of IoT nodes. Furthermore, they are not address the issue of intelligently countering untrustworthy behaviour by IoT nodes such as bad-mouthing attacks, on-off attacks, contradictory behaviour attacks etc.

(Ortiz et al. 2014)	This work uses a novel paradigm for ubiquitous computing beyond the IoT, denoted as the Social Internet of Things (SIoT). This seamless integration brings new relationships, allowing the creation of novel services and applications that will be of great interest both to end users and stakeholders.	This approach involves the union of IoTs with SNs known as SIoT. This union emerges from inheriting social networking features and values of interactivity, recommendation and filtering and services composition and suggests a universal framework to combine users, devices and services and the interactions among them.	Similar, to thee above approaches, they have not investigated the use of trust-based clustering of IoT nodes. Furthermore, they are not address the issue of intelligently countering untrustworthy behaviour by IoT nodes such as bad-mouthing attacks, on-off attacks, contradictory behaviour attacks etc.
(Varghese, Chithralekha & Kharkongor 2016)	The researchers use the self-organized cluster-based energy efficient meta trust model for Internet of Things. In this model, the energy consumption of each node is decreased by forming clusters, which in turn, provides scalability and also helps to eliminate the selfish nodes.	A pattern matching trust model is used to ensure full trust in the nodes and at the same time the identity of the node is considered to punish the malicious nodes so that they do not intrude in the network in the future.	Does not address the issues related to scalability of the proposed solution. Furthermore, does not address the selfish behaviour of the nodes and to eliminate the misbehaviour of nodes.

(Hamadeh, Chaudhuri & Tyagi 2017)	A minimal distributed trust layer based on distributed consensus-like operations	Cast in the context of the APIs supported by a trusted platform module (TPM)	Issues of scalability and resilience of the trust management approach against cyber-attacks have not been addressed.
(Sarobin & Ganesan 2016)	Using the proposed algorithm, the sensor node reaches its maximum residual energy and with an optimized distance to the sink node.	The algorithm optimizes CHs' energy consumption, extending the network lifetime.	This work has been done for WSNs and it is not applicable to large scale IoT nodes. Does not present any solutions to the existing issues related to trust management in IoT.

As can be inferred from Table 2-5 none of the approaches presented a comprehensive IoT trust management approach that encompasses ***both*** solutions for the scalability of the trust management approach and solution for its resilience against cyber-attacks such as bad-mouthing attacks, on-off attacks, contradictory attacks etc. Furthermore, none of the approaches mentioned in this table have proposed the use of trust-based clustering for IoT trust management.

2.6 FUZZY-LOGIC BASED MECHANISMS FOR TRUST MANAGEMENT IN THE IOT

The IoT is a vibrant new field of research at the intersection of electronic engineering and computer network discipline. It has transformed the Internet from interaction between humans only to that of humans and things and even between things (Mosenia & Jha 2017). This has been made possible through the capabilities of smart devices, which are able to make decisions without the intervention of humans and share information with other smart devices to achieve a particular goal. However, incorporating all these devices into the Internet leads to various challenges in security since the majority of Internet technologies and communication protocols were not originally designed for IoT support (Hossain, Fotouhi & Hasan 2015). The distributed and decentralized nature of the IoT is also a challenge in terms of access control, trust management and identity management (IdM) (Mahalle et al. 2013). To address this issue of control, the fuzzy approach to trust-based access control (FTBAC) is used. FTBAC is a scalable and flexible framework whose performance and functionality is not affected by the increased number of devices. In this approach to trust management, cryptographic protection is achieved through access control through increased levels of trust even though it creates extra overhead due to energy and time consumption. The model is easily integrated in decision-making based on utility and its flexibility allows for additional components (Mahalle et al. 2013). However, FTBAC does not address issues related to scalability of the trust management approach and on how trust management approach can be made resilient to untrustworthy behaviours (such as bad-mouthing, on-off attacks, contradictory attacks etc.) by the IoT nodes.

(Renubala & Dhanalakshmi 2014) noted that trust and security are two interdependent concepts and the key difference is the high overhead and complexity of security. They propose the use of a novel fuzzy with trust-based secure model routing protocol (FTPR) in wireless sensor networks (WSNs). The method avoids malicious nodes to reduce the

consumption of energy, reducing the number of collected recommendations from neighbours to compute indirect trust. The proposed method causes a reduction in packet loss through the recognition and rejection of malicious nodes on the basis of the trust value. The behaviour of the dynamic nodes facilitates a trust evaluation model where a two-level fuzzy system decides the intensity of trust of a node. The trust level from both levels is computed through the consideration of the dropped packet, generated replay packets and the false routing messages generated. The shortcomings of this proposed approach are that it has been proposed for WSNs and their applicability to IoT, particularly in addressing issues related to scalability of the trust management approach has not been addressed. Furthermore, (Renubala & Dhanalakshmi 2014) fail to address how trust management approach can be made resilient to untrustworthy behaviours (such as badmouthing, on-off attacks, contradictory attacks etc.) by the IoT nodes.

(Chen et al. 2011) investigated fuzzy problems to establish membership functions through the use of fuzzy set theory. They first created a mathematical model of fuzzy trust. The information of the package forwarding neighbours' behaviours is collected through a neighbouring monitoring process. (Mahalle et al. 2013) used the FTBAC trust management model where the fuzzy approach entails finding the relationship between trust and access control. The fuzzy approach for calculating trust deals with the linguistic information of devices in addressing access control, guaranteeing scalability and energy efficiency. Trust was used as a decision-making tool of access control. However, none of these researchers address how the proposed trust management approaches can be made scalable and resilient to untrustworthy behaviours (such as badmouthing, on-off attacks, contradictory attacks etc.) by the IoT nodes.

(Lize, Jingpei & Bin 2014) also looked to find a trust mechanism for IoT, establishing a formal trust management control mechanism based on the modelling architecture of IoT. They adopted a formal semantic-based method and fuzzy set theory in the

realization of the mechanism of trust and decision-making based on trust for a reasonable and coherent result. The focus is on the decomposition of IoT into three layers, each under the control of trust management for special purposes. The process indicates that the final decision-making is done by a service requester and then the use of a formal semantics-based and fuzzy set theory. Similar to the work by (Chen et al. 2011), this work although it proposes the uses of fuzzy-logic for IoT trust management, it fails to consider or evaluate the scalability of their proposed approach and also the ability of their proposed approach to counter bad-mouthing attacks, on-off attacks etc… (Sirisala & Bindu 2015) proposed the Uncertain Rule-based Fuzzy Logic QoS Trust Model in MANETs - FQTM, selecting nodes based on their cooperativeness and capability. Fuzzy logic was applied to compute the trust value of nodes by considering their reliability and quality metrics. Using FQTM, nodes with high trust values are selected to construct routes to the destination. The expert system in the fuzzification process converts the crisp values using rule-base trust into fuzzy values where all the rules are framed according to the resource status of the nodes. Unfortunately, similar to the above mentioned approaches FQTM does not address how the proposed trust management approaches can be made scalable and resilient to untrustworthy behaviours (such as badmouthing, on-off attacks, contradictory attacks etc…) by the IoT nodes.

From the literature review, it is evident that fuzzy-logic based approaches have been proposed to address various challenges in the trust management literature. However, they fail to investigate the use of fuzzy-logic based approaches to enabling the scalability of the proposed trust management approaches trust for IoT. Furthermore, the existing literature fails to investigate the use of fuzzy-logic based approaches for ensuring that the proposed trust management approaches are resilient to untrustworthy behaviours by the IoT nodes such as badmouthing, on-off attacks, contradictory attacks etc.

Table 2-6 presents an overview and comparative evaluation of the existing literature on trust management in IoT that makes use of fuzzy-logic and describes their features and shortcomings.

Table 2-6 Overview and evaluation of the existing on fuzzy-logic based approaches used in IoT

Reference	Protocol description	Features of the protocol	Limitations of the protocol
(Chen et al. 2011)	The researchers investigated fuzzy problems to establish membership functions through the use of fuzzy set theory	This work created a mathematical model of fuzzy trust, and the information of the package forwarding neighbours' behaviours is collected through a neighbouring monitoring process.	This work does not focuses on how the proposed fuzzy trust model can be made scalable and resilient to untrustworthy behaviours (such as badmouthing, on-off attacks, contradictory attacks.
(Lize, Jingpei & Bin 2014)	This research focuses on trust mechanism for IoT, establishing a formal trust management control mechanism based on fuzzy modelling of IoT.	This work adopted a formal semantic-based method and fuzzy set theory in the realization of the mechanism of trust and decision-making based on trust for a reasonable and coherent result. The focus is on the decomposition of IoT into three layers, each under the control of trust management for special purposes.	This research fails to consider or evaluate the scalability of their proposed approach and also the ability of their proposed approach to counter bad-mouthing attacks, on-off attacks.
(Mahalle et al. 2013)	This research proposed a fuzzy model based trust (FTBAC) to address issue of control distributed and decentralized nature of the IoT.	FTBAC is a scalable and flexible framework whose performance and functionality is not affected by the increased number of devices. In this approach to trust management, cryptographic protection is achieved through access control through increased levels of trust even though	FTBAC does not address issues related to scalability of the trust management approach and also on how trust management approach can be made resilient to untrustworthy behaviours (such as bad-mouthing, on-off attacks, contradictory attacks by IoT nodes.

		it creates extra overhead due to energy and time consumption.	
(Renubala & Dhanalakshmi 2014)	The researchers propose the use of a novel fuzzy with trust-based secure model routing protocol (FTPR).	FTRP method avoids malicious nodes to reduce the consumption of energy, reducing the number of collected recommendations from neighbours to compute indirect trust. Also can detect some behaviour of the dynamic nodes facilitates a trust evaluation model where a two-level fuzzy system decides the intensity of trust of a node.	The proposed approach FTRP fail to investigate the use of fuzzy-logic based approaches to enabling the scalability of the proposed trust management approaches trust for IoT.
(Sirisala & Bindu 2015)	This work proposed FQTM model for selecting nodes based on their cooperativeness and capability uncertain rule-based Fuzzy Logic QoS Trust Model in MANETs –FQTM.	The approach FQTM using for nodes with high trust values are selected to construct routes to the destination. The expert system in the fuzzification process converts the crisp values using rule-base trust into fuzzy values where all the rules are framed according to the resource status of the nodes.	This work is similar to the above limitations does not address how the proposed trust management approaches can be made scalable and resilient to untrustworthy behaviours (such as badmouthing, on-off attacks, contradictory attacks etc.) by the IoT nodes.

As can be inferred from Table 2-6, fuzzy logic has been used to address various issues in IoT trust management. However, as discussed none of the approaches presented in the above table proposed the use of fuzzy logic for achieving the scalability of the proposed trust management approaches or for ensuring the resilience of the proposed approach against cyber-attacks.

2.7 CRITICAL EVALUATION AND SUMMARY OF SHORTCOMINGS

In light of the above literature review, there are various open research issues regarding IoT. The trust management protocols proposed in the literature generally lack context awareness and the device owner's subjective elements (Bao & Chen 2012b). As such, the trust assessment result is generalized and it is difficult to implement intelligence in actual IoT trust networks that require: anonymity to be upheld, issues of impersonation to be eliminated and Internet interconnectivity. Furthermore as inferred and discussed in Section 2.3, the existing trust management protocols have not addressed the issue of their resilience against cyber-attacks such as bad mouthing, on-off attacks, bad service attacks and contradictory attacks.

In addition, the concept of ubiquitous computing is yet to be entirely secure and private, and there is no deep research on a fully operational trust management framework in IoT (Tselentis, Domingue & Galis 2009). It is also evident from the discussions and critical analysis in Section 2.2 to Section 2.6 that the existing literature fails to propose a trust management framework for IoT that addresses the issues of scalability, counters attacks by misbehaving nodes (such as bad-mouthing attacks, on-off attacks, contradictory attacks etc.) and is efficient in terms of energy utilization.

The aforementioned IoT protocols and techniques in Section 2.3, have been constrained by various challenges, which include: lack of a platform to establish informed consent prior to data collection; limited capacity to accord consumers' freedom of privacy choices; context awareness (invasion of privacy); and fear of adopting intense anonymity to an extent that relevant profiling cannot be established (Saied et al. 2013). There have been concerns that the IoT is evolving quickly without proper thought to the significant security problems that may ensue and the necessary

administrative/regulatory changes. According to (Perera et al. 2012), security and privacy are the greatest concerns in implementing the IoT technology. Specifically, as the IoT spreads widely, digital crime is likely to turn into an inexorably physical threat. Numerous Internet-related appliances that spy on our activities have been recorded, with the possibility of spreading further with the incorporation of computing and Internet connectivity in almost all aspects of life. Even though much work has been done to safeguard privacy, there has been little focus on performance and adaptability concerns with the powerful usage of the IoT. Therefore, past solutions to IoT network security and/or privacy cannot comprehensively tackle trust management issues.
Furthermore, the work reviewed for trust scalability for the IoT have clearly established lack of effort in developing a trust management approach that is resilient to cyber-attacks.

The majority of the reviewed work (Bao & Chen 2012a; Bao, Chen & Guo 2013; Chen, Guo & Bao 2016; Lyu et al. 2015; Ma, Liu & Zhang 2015) addressed trust management in the IoT. Three research articles were in the form of survey paper (Guo, Chen & Tsai 2017; Sicari et al. 2015; Yan, Zhang & Vasilakos 2014). The work in (Yu et al. 2017) did not address trust management in the IoT properly. Trust management proves to be one of the most challenging aspects of the IoT. Various works that pertain to strengthening the possibility of an authentic and trustworthy data exchange among humans, things and robots is still to be developed. Privacy and security issues arise should the data exchange be filled with malicious nodes or blocks. The existing literature surveys have been instrumental in determining what is lacking in the current research that deal with trust management in the IoT.
Furthermore, none of the IoT clustering approaches (presented in Section 2.5) have investigated the use of trust as a basis for clustering the IoT nodes with a view to achieve scalability of the IoT trust management. Furthermore, none of these studies have proposed approaches to intelligently detect and counter misbehaving IoT nodes.

The fuzzy logic-based approaches presented in Section 2.6 largely focused on using fuzzy operators for trust computation. (Chen et al. 2011; Lize, Jingpei & Bin 2014; Mahalle et al. 2013; Renubala & Dhanalakshmi 2014; Sirisala & Bindu 2015) None of them have proposed the use of fuzzy logic for enabling IoT trust management scalability and for making the trust management approaches resilient to untrustworthy behaviours by the IoT nodes such as badmouthing, on-off attacks, contradictory attacks and bad service attacks by IoT nodes.

Table 2-7 presents an evaluation of the current approaches in the literature for trust management in the IoT. The dimensions used for comparison are (i) scalability of the solution; (ii) Security protocol for reliable trust management; (iii) Clustering-based trust management for IoT; and (iv) Fuzzy-logic based approaches.

Table 2-7 Critical evaluation of current IoT trust management approaches

Reference	Trust Scalable for IoT	Security protocol for reliable trust management for IoT	Clustering-based trust for IoT	Fuzzy-Logic Based Approaches
(Jabeur et al. 2017)	×	×	✓	×
(Ortiz et al. 2014)	×	×	✓	×
(Varghese, Chithralekha & Kharkongor 2016)	×	×	✓	×
(Sicari et al. 2015)	×	✓	×	×
(Miao & Wang 2012)	×	✓	×	×
(Sirisala & Bindu 2015)	✓	×	×	✓
(Bello & Zeadally 2016)	×	✓	×	×
(Khan & Herrmann 2017)	×	✓	×	×
(Hamadeh, Chaudhuri & Tyagi 2017)	×	×	✓	×
(Sarobin & Ganesan 2016)	×	×	✓	×
(Chen et al. 2011)	×	✓	×	✓
(Vögler et al. 2016)	✓	×	×	×
(Jiang et al. 2015)	✓	×	×	×
(Sarkar et al. 2015)	✓	×	×	×
(Neisse et al. 2015)	×	✓	×	×
(Lize, Jingpei & Bin 2014)	×	×	×	✓
(Li et al. 2013)	✓	×	×	×
(Han & Woo 2016)	✓	×	×	×
(Chen, Guo & Bao 2016)	✓	×	×	×
(Mahalle et al. 2013)	✓	×	×	✓
(Bao & Chen 2012a)	✓	✓	×	×
(Ahmed et al. 2016)	×	✓	×	×
(Guo, Chen & Tsai 2017)	×	✓	×	×
(Yan, Zhang & Vasilakos 2014)	×	✓	×	×
(Yu et al. 2017)	×	✓	×	×
(Ma, Liu & Zhang 2015)	×	✓	×	×
(Bao, Chen & Guo 2013)	✓	✓	×	×
(Lyu et al. 2015)	×	✓	×	×
(Renubala & Dhanalakshmi 2014)	×	×	×	✓
(Kokoris-Kogias, Voutyras & Varvarigou 2016)	✓	✓	×	×
(Bellavista & Zanni 2016)	✓	✓	×	×
(Gharbieh et al. 2017)	✓	✓	×	×
(Ray, Abawajy & Chowdhury 2014)	✓	✓	×	×

Based on the comparative analysis presented in Table 2-7, and also the analysis presented in Table 2-1 to Table 2-6, I conclude the following:

(a) There is no comprehensive IoT trust management approach in the existing literature that encompasses solutions for both scalability of the trust management approach and its resilience against the cyber-attacks.
(b) The existing literature has not investigated the issue of trust-based clustering of the IoT nodes as a means to achieve scalability of the trust management solution.
(c) The existing literature has not investigated the use of fuzzy-logic for IoT trust management scalability.
(d) The existing literature has not investigated the use of fuzzy-logic for ensuring the resilience of the developed trust management solution.

2.8 CONCLUSION

In this chapter, an extensive survey of the literature on trust management in the IoT was conducted. The literature was categorised as follows: trust management in the IoT, context-aware assessment for IoT, the scalability of the IoT, security protocol for reliable trust management for IoT, clustering-based trust in the IoT and fuzzy logic for the IoT, each study was discussed these different perspectives. Furthermore, the features and shortcomings of each approach were discussed. Finally, the current studies were evaluated based on different attributes and it was found that there are still gaps in trust management and security in the IoT.

In Chapter 3, the gaps in the existing literature are identified and the research problem associated with the current literature is defined. Chapter 4 proposes a solution overview by describing the methods to be used to solve the problems of trust management and security for the IoT and it also outlines the whole robust framework of the proposed solution.

CHAPTER 3

PROBLEM DEFINITION

3.1 INTRODUCTION

In the first chapter, the importance of the trust management in IoT was presented. Then the current literature in this area was reviewed in Chapter 2. The second chapter identified that many researchers have proposed various techniques for trust management in IoT. Most of them focused on trust management for individual IoT nodes or WSN nodes. Some of these existing studies focused on trust management solutions in IoT however, they fail to address critical issues such as the scalability of trust management solutions, the security and reliability of the trust management solutions against attacks by untrustworthy IoT nodes etc. In other words, as noted in Chapter 2, none of the existing solutions presented a comprehensive trust management solution in IoT addressing both the scalability of the proposed trust management solution and the ability of the trust management solution to detect untrustworthy or uncompliant behaviour by the IoT nodes. Hence, in the current literature, there are some limitations for developing a scalable, robust and reliable methodology for trust management in IoT. In this chapter, firstly I define the concepts and terminologies which relate to this research problem in Section 3.2. Subsequently in Section 3.3, I formally define the research problem. The research questions and the research objectives are presented respectively in Section 3.4 and Section 3.5. In Section 3.6, the

research methodology that is used to solve the defined problem is presented. Finally, the conclusion is presented in Section 3.7.

3.2 Explanation or definition of key terms and concepts

In this section, I present either the definitions or the explanations for all the key concepts and terms that are used to define the research problem in this thesis.

3.2.1 Internet of Things (IoT)

Internet of Things (IoT) is a network that allows all physical devices (things) to be connected over the Internet. IoT devices are embedded with software, actuators, and sensors which enables these devices/things to communicate each other and exchange their data without human interaction (Gubbi et al. 2013).

3.2.2 IoT Security

I define *IoT security* as the safeguard techniques or mechanisms to keep the IoT devices, communications and the entire environment protected from any dangerous threats or attacks.

3.2.3 Cyber Security

I define *cyber security* as the collection of all protection mechanisms including both hardware and software to protect data or networks from threats and attacks.

3.2.4 IoT Security Protocol

I define the *IoT security protocol* as the methods and techniques that provide robust measures to keep the IoT environment secure and trusted.

3.2.5 Trust

I define *trust* as the value assigned by an IoT node based on the services it received and actions performed by the other node.

3.2.6 Trust Management

I define *trust management* as a collection of all the processes that can be used to assess and guide the actions taken to carry out reliable trust-based automated decision-making process.

3.2.7 Trustworthy nodes

I regard *trustworthy nodes* as nodes that have been given a high trust value by other nodes and can be relied upon to deliver high quality services to other nodes in the network.

3.2.8 FUZZY LOGIC

Fuzzy logic is converting any data to be logic sense (true or false). The variables of fuzzy logic can be a trust value ranging betIen (0 and 1) (Bělohlávek & Klir 2011).

3.2.9 CLUSTER

I define a *cluster* as a group of nodes with a similar trust value or with trust value in a pre-define range in the IoT network.

3.2.10 TRUST-BASED CLUSTERING

I consider *trust-based clustering* as the grouping of IoT nodes based on their overall trust value.

3.2.11 TRUST MANAGEMENT PLATFORM

I define the *trust platform* as the overall proposed environment for IoT trust management which includes all the components and processes to deliver a reliable IoT platform. In this thesis for proposing a comprehensive IoT trust management platform I focus on the issues of scalability of the trust management framework and its resilience against IoT nodes carrying out untrustworthy behaviour.

3.2.12 SCALABILITY

I define *scalability* as ability of the trust framework to grow seamlessly to accommodate a large number of IoT nodes.

3.2.13 NODE

I define a *node* as a physical device or (thing) that is part of the IoT network and communicates with other nodes.

3.2.14 SUPER NODE (SN)

I consider the *super node (SN)* as the main and central node in the overall IoT environment that has a repository to store all the nodes' data and trust values in the IoT environment or the IoT application.

3.2.15 MASTER NODE (MN)

I consider the *master node (MN)* as the controlling node in each cluster that is responsible for managing the entry and exit of each node in the cluster (based on its trust value). The master node also stores the data and trust value of the cluster nodes under its supervision.

3.2.16 CLUSTER NODE (CN)

I consider the *cluster node (CN)* as the member node in each cluster. The cluster node (CN) communicates and interacts with other cluster nodes in this process deliver services to the other nodes. Subsequently, they are assigned a trust value based on the services provided.

3.2.17 CYBER-ATTACKS

Generally speaking, *cyber-attacks* are disruptions which aim to destroy or damage the network. This could take various forms such as overburdening the network by sending many messages so that the network cannot handle it, bad-mouthing attacks etc. In this thesis, I refer to cyber-attacks as activities or untrustworthy activities aimed at disrupting the proper functioning of the IoT trust management framework. I limit ourselves to cyber-attacks such as bad-mouthing attacks, on-off attacks, bad service attacks and contradictory attacks (Pasqualetti, Dörfler & Bullo 2011).

3.2.18 ON-OFF ATTACKS

(Chae, DiPippo & Sun 2015) defined an *On-Off attack* is a malicious node attack by behaving as a good and bad node alternatively. The On-Off attacks has represented two stats: the *On* represent is considered the attack state, the *Off* represent is a normal states and the node behaves as a good node.

3.2.19 CONTRADICTORY BEHAVIOUR ATTACKS

I consider the *contradictory behaviour attacks* as a type of malicious node attack where the node pretends that it is a good behaviour node whereas actually it is bad behaviour node.

3.2.20 BAD-MOUTHING ATTACKS

The bad mouthing attack is initiated at a certain moment and is composed of a number of malicious nodes that falsely assign low reputation to some of their neighbours in a random fashion (Banković et al. 2011).

3.2.21 BAD SERVICE ATTACKS

I consider the bad service attacks is a type of malicious node attack where this node attempt to prevent legitimate nodes from accessing the service.

3.2.22 FUZZY BANK

I consider the *fuzzy bank* is a cache of trust value stored in cluster node (CN) memory about the fuzzy status or 'node type' of its neighbours.

3.2.23 ROUTING SCORE

I consider the *routing score* as the value that is generated by evaluating the quality of service responses about a node in a different cluster.

3.3 PROBLEM OVERVIEW AND PROBLEM DEFINITION

As discussed in Chapter 2, trust and trust management approaches for the Internet of Things provide support to the IoT nodes within a network or an application by building a trusted network and enabling the IoT nodes to carry out trust-based decisions as to whether or not to interact with other IoT nodes. A lot of work has been carried out highlighting the importance of trust management in IoT. Also, trust management has been investigated in different types of networks such as WSN, mobile ad hoc networks, peer-to-peer networks etc.

As pointed out in Chapter 2, one of the critical issues in IoT is that of trusted communication or trust management. Trust management in IoT is critical as it provides the basis for creating a secure IoT environment by keeping the communication and interactions between IoT nodes under trusted and reliable environment. It acts as a counter to interacting with malicious and harmful nodes (Chang et al. 2012). Unfortunately, the existing research has failed to investigate pressing and crucial issues in IoT trust management. Rather, they have focused on how to develop IoT-based trust management approaches using the old trust management techniques that were developed for WSNs networks etc….

As discussed in Chapter 2, when a trust management approach or solution has built a trusted interaction environment between all nodes, the next crucial issue is to ensure the scalability of the developed trust framework within a given IoT application. In fact, this issue is one of the pressing issues (Miorandi et al. 2012) facing an IoT application, as there are billions of IoT nodes connecting and communicating with each other. As discussed in Chapter 2, some research has focused on this issue but there is still no solid solution which takes into account the limited memory for IoT nodes.

Another key pressing gap or issue with the existing IoT trust management approaches is to ensure the resilience of the proposed trust management approach against attacks by untrustworthy nodes. The (cyber) attacks on the IoT trust management solution could take the form of bad-mouthing attacks, on-off attacks, bad service attacks or contradictory attacks. As discussed in the previous chapter, the current studies do not focus on the development of intelligent approaches to prevent untrustworthy nodes in carrying out these cyberattacks and building them as a part of the IoT trust management approach.

In this thesis, in order to counter the above mentioned gaps in the existing literature, I focus on proposing and developing a comprehensive and intelligent methodology for trust management in IoT. The proposed trust management methodology would be both scalable and resilient against cyber-attacks, namely bad-mouthing attacks, on-off

attacks, contradictory behaviour attacks and bad service attacks carried out by untrustworthy or malicious IoT nodes. I term our approach as comprehensive because it is the first to present a combined solution for both scalability and resilience against attacks by malicious nodes. It is predicted that by 2020, more than 50 billion devices will be connected over the Internet (Evans 2012) and interacting with each other. To enable and facilitate collaboration between the IoT nodes, there is a need for such a resilient and scalable trust management system for the IoT networks.

Based on the previous discussion, the research problem that is addressed in this thesis is described as follows:

How to develop an integrated IoT trust management that is scalable and provides reliable environment for the IoT population by countering cyber-attacks on the trust management approach?

This research problem can be decomposed into four research questions. The following section presents these research questions.

3.4 RESEARCH QUESTIONS

In order to achieve the objectives discussed in Chapter 1, this section outlines the research questions that are addressed in this thesis. The aforementioned research question can be broken down into three research sub-questions are as follows:

Research Question 1:

How do I develop an intelligent, self-configuring and memory efficient trust management platform for IoT?

As the existing literature is covered in Chapter 2 has not developed an integrated trust management for the IoT platform which takes into account the memory efficiency of IoT nodes and efficient scalability of IoT nodes, there is need for a robust and reliable platform in IoT to tackle these issues.

Research Question 2:

How do I incorporate scalability into the proposed trust platform for IoT?

The existing studies failed to present a comprehensive trust management solution for IoT that is scalable. The existing literature has tested the scalability of their proposed solution under limited number of IoT nodes.

Research Question 3:

How can I ensure that the proposed IoT trust management framework is resistant to cyber security attacks on the proposed trust management framework?

For the purpose of this thesis, for cyber security attacks, I limit ourselves to bad-mouthing attacks, on-off attacks, contradictory behaviour and bad service node attacks. As discussed above and in Chapter 2 the existing literature does not propose approaches to intelligently detect and counter untrustworthy behaviour by the IoT with a view to counter cyber security attacks such as bad-mouthing attacks, on-off attacks, contradictory behaviour and bad service node attacks. These attacks are impact the working of the trust management framework and hence there is a need to develop intelligent approaches to counter them so as to increase its reliability.

Research Question 4:

How do I validate the developed solution for its scalability and reliability for the IoT framework?

I need to test our approach to ensure that it is scalable and it is validated well by using a prototype-based approach and an appropriate programming language.

3.5 RESEARCH OBJECTIVES

In order to address the above mentioned research questions, the main objectives of this thesis is as follows:

Research Objective 1:

To develop an intelligent trust management platform for IoT that is intelligent and memory-efficient in assisting the IoT nodes communicate in a trusted environment.

Research Objective 2:

To develop an intelligent scalable trust management platform for IoT.

Research Objective 3:

To develop intelligent approaches to counter cyber security attacks on the proposed IoT trust management approach.

Research Objective 4:

To validate the above solutions using a prototype framework and test its performance along various identified benchmarks.

3.6 The Research Approach to Problem-Solving

To address the above mentioned research questions and research objectives,, this thesis develops and subsequently experimentally validates a trust management methodology for the Internet of Things, taking into account all the research questions in Section 3.5. To address these research issues, it is important to follow a systematic scientific methodology in order to ensure that the developed methodology is based on a robust scientific foundation. In this section, I briefly present the existing scientific research methods and subsequently describe the method that is used to address the research issues in this thesis.

3.6.1 Existing Research Methods

There are two fundamental research categorises in information technology, namely the science and engineering approach and the social science approach. Both these approaches aim to solve research problems in a research appropriate manner.

3.6.1.1 Science and Engineering Approach

In order solve research problems, the science and engineering approach involves two methods, which are scientific and engineering (Peffers et al. 2007). The science and engineering approach uses different techniques to investigate phenomena, obtain new knowledge and combine the new or current knowledge with the previous (Goldhaber & Nieto 2010). The aim of this method is to create something that works (Galliers 1990). Overall, there are three levels for the science and engineering-based approach: conceptual, perceptual and practical. The description of these levels is as follows:

1) Conceptual Level: The first level of the science and engineering approach is the conceptual level. This level focuses on creating new concepts and perceptions through an evaluation of the process.
2) Perceptual Level: The second level of the engineering approach is the perceptual level. This level articulates a new method by designing and creating a system to solve the research problem.
3) Practical Level: The third level of the engineering approach is the practical level. This level handles the experimentation, testing and validation using and applying real-world scenarios.

3.6.1.2 Social Science Approach

The social science research approach is based on finding evidence by following a systematic plan to prove or disprove the proposed assumptions based on the data collected (Burstein & Gregor 1999). This research approach collects data using surveys or interviews. The social science research approach is divided into two types: the qualitative approach and the quantitative approach.

a) In the qualitative approach, the researcher makes direct observations and communicates through surveys and interviews, and then the data which is collected is analysed.
b) In the quantitative approach, the researcher conducts a statistical analysis of the data to validate the research claims.

The social science research approach assists researchers to understand the social and cultural problems within the domain of research. This approach can only suggest the extent the acceptability of the research method or the reason for applying this method.

However, this research method differs from engineering-based research approach as the social science approach does not propose a method for solving a research problem as in the engineering-based approach (Kaplan & Maxwell 2005).
This thesis focuses on developing an intelligent and scalable methodology for trust management in IoT. To achieve this task, the new ideas and concepts about our proposed approach trust management approach in IoT need to be defined and implemented. Furthermore, the proposed approach needs to be empirically tested and validated. Thus, this thesis follows the science and engineering research approach. The next section presents the details of the chosen research method.

3.6.2 THE CHOICE OF THE SCIENCE AND ENGINEERING-BASED RESEARCH METHOD

In this thesis, I use the science and engineering-based research method to achieve the research objectives outlined in Section 3.5. An overview of how the science and engineering -based methodology is used in this thesis is illustrated in Figure 3-1.

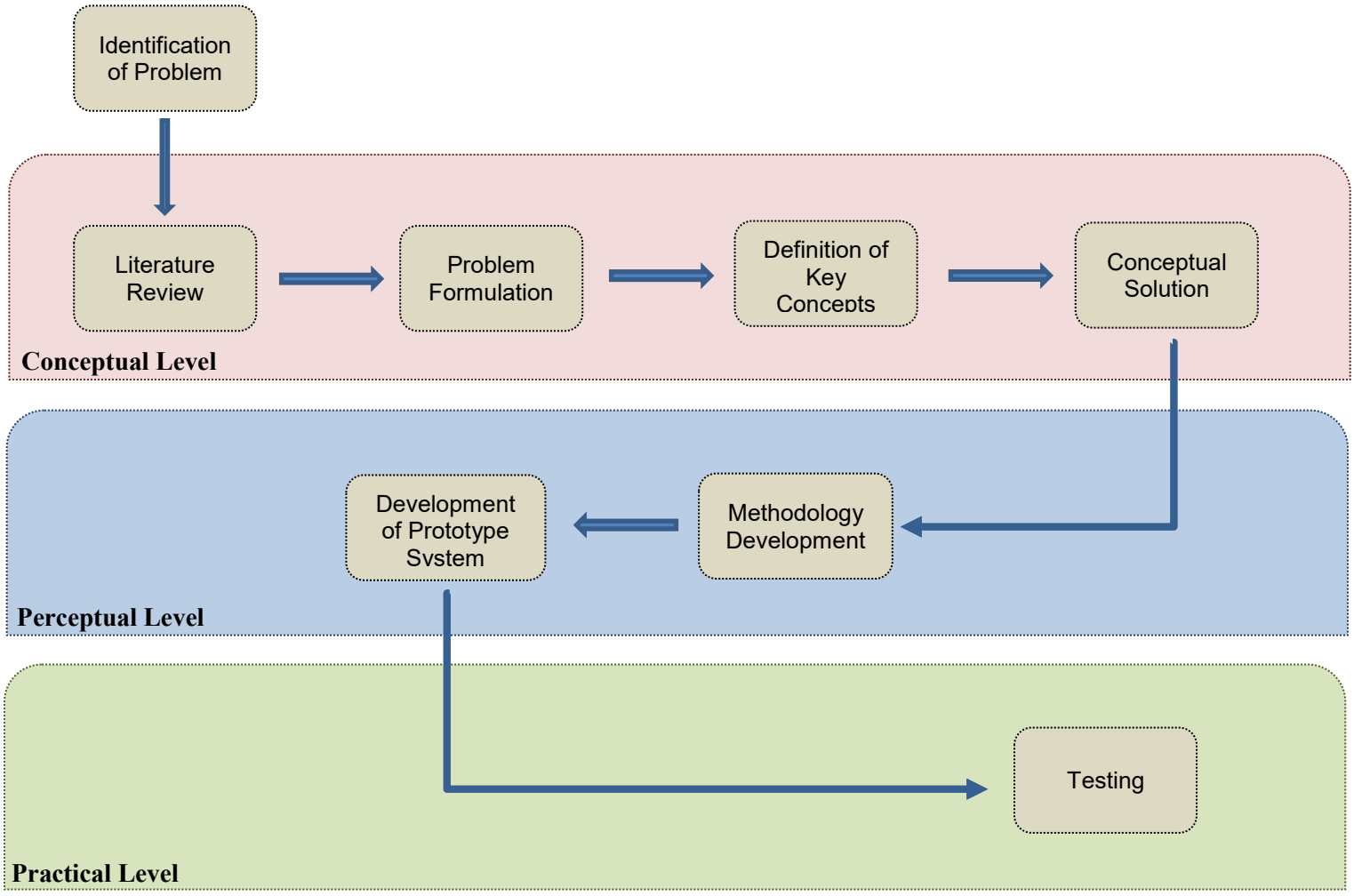

Figure 3-1 Overview of the use the science and engineering-based research methodology is used in this thesis

As stated by (Galliers 1990), there are three research levels in the science and engineering-based methodology - conceptual level, perceptual level and practical level. Firstly, the research starts by identifying the research problems in the area of trust management in IoT. Next, in the conceptual level, sub-processes are applied to identify the concepts and ideas of a novel trust management in IoT method. Then, in the perceptual level, sub-processes are applied to develop and implement the proposed method, and finally in the practical level, the proposed method is tested.

3.6.2.1 CONCEPTUAL LEVEL

This is the first level of the science and engineering research method in which new concepts and ideas are generated in the area of trust management in the IoT with a view of achieving the research objectives listed in Section 3.5. This research level includes the following four processes: a *literature review*, *problem formulation*, *definition of key concepts* and *conceptual solution*.

3.6.2.1.1 LITERATURE REVIEW

Based on a comprehensive review of the current literature covering the research area of trust management in IoT, the research gaps are identified. Based on this, the research questions that need to be addressed have been formalized. The research gaps that have been identified from the extensive literature review process are listed in Section 2.7.

3.6.2.1.2 PROBLEM FORMULATION

Based on the existing literature reviews, the research question to be addressed in this thesis is identified and formulated. In this process, research sub-questions and research objectives are identified, as presented respectively in Section 3.4 and Section 3.5.

3.6.2.1.3 DEFINITION OF KEY CONCEPTS

The next process after formulating the problem is to define a set of key concepts, which were identified in Section 3.2. The research questions and research objectives uses some the presented concepts in Section 3.2

3.6.2.1.4 CONCEPTUAL SOLUTION

This is the final process in the conceptual level. The aim of this process is to create an overall conceptual solution for the identified research problem. Chapter 4 describes the proposed solution for the identified research problem in this thesis.

3.6.2.2 PERCEPTUAL LEVEL

This is the second level of the science and engineering research method. It focuses on the actual development of the solution and its implementation. The perceptual level includes the following sub-processes namely *methodology development* and *development of the prototype*.

3.6.2.2.1 METHODOLOGY DEVELOPMENT

The detailed working of the methodology for trust management in IoT is developed based on the conceptual solution (presented in Chapter 4). The detailed working of the methodology is explained in Chapter 5, Chapter 6 and Chapter 7.

3.6.2.2.2 DEVELOPMENT OF PROTOTYPE

After the methodology has been developed fully, the proposed methodology is validated. In this thesis, the IoT simulator tool Cooja, C++ and Java are used to implement and validate various aspects of the proposed methodology. The development of the prototype and its working has been presented in Chapter 5, Chapter 6 and Chapter 7 against the research outcomes presented in each of these respective chapters.

3.6.2.3 PRACTICAL LEVEL

This is the third level of the science and engineering research method. This level focuses on testing and validating the developed method. In this thesis, the proposed method is tested by developing a prototype implementing the proposed IoT trust management platform and it is validated using the IoT simulator tool. Moreover, the research method is evaluated using the trust performance measures in the area of IoT.

I use a prototype-based approach for validation and the IoT simulator tool. I use is Cooja from Contiki and other programming languages such as C++ and Java to ensure the results are validated.

I develop the proposed platform for the first research issue by creating a comprehensive trust management platform which includes the main components of the platform to ensure reliable communication between all the IoT platform entities. Then, I execute the experiments for the second research issue using scenarios of expanding the nodes within the IoT environment by increasing the number of nodes. This will help us getting an understanding of the scalability of the proposed trust management approach. Furthermore, I implemented both fuzzy-logic based and non-fuzzy logic based algorithms to counter cyber-security attacks such as bad-mouthing attacks, on-off attacks, and contradictory behaviour attacks. Extensive experimentation was carried out and the results were obtained. The experiments are carried out in the IoT simulator tool (Cooja).

3.7 CONCLUSION

This chapter begins by identifying the research gaps related to trust management in the IoT. The key concepts and terms that are used in defining the research question and research objectives are then presented. Subsequently, this chapter presented the formal research problem definition that is addressed in this thesis. Then the identified research problem was divided into four research questions that need to be addressed in order to solve the research problem of trust management in IoT.

Furthermore, different research approaches were discussed and described. The most appropriate approach for the requirements of this research is the science and engineering method, which is the method chosen for this research. An overview of how research process carried out using the science and engineering method was presented

The next chapter presents an overview of the research solution proposed to solve the problems presented in this chapter.

CHAPTER 4

SOLUTION OVERVIEW

4.1 INTRODUCTION

The Internet of Things (IoT) is not a new technology by itself, it is a new computing paradigm that facilitates many aspects of human life. The nodes within an IoT network work collaboratively to deliver various specific services in various sectors such as agriculture, medicine, transportation etc. (Bao & Chen 2012b). The IoT devices communicate with each other without any human interaction, such as receiving and sending messages, executing commands etc… At the time of writing this thesis, there are a huge number of applications and examples of IoT applications that enhance the quality of life of human beings, such as smart cities, smart buildings, smart healthcare etc. As predicted by Cisco (Evans 2012), there will be more than 50 billion devices connected over the Internet by 2020 (Talwar et al. 2014).

However, the current generation of IoT devices and networks face a number of limitations and challenges, in area of the trust management such as managing the scale of the IoT network, ability to intelligently detect and prevent untrustworthy IoT nodes from carrying out cyberattacks (such as the ones mentioned in Chapter 3), the constraints on the memory of the IoT nodes and so on.

As discussed in Chapter 2, a large body of existing research has attempted to solve the problems of trust management in IoT. However, based on the discussion in Chapter 2 and Chapter 3, it is apparent that there are still significant research gaps concerning

trust management in IoT which need to be addressed. Chapter 3 provides the research questions that aim to solve these critical research issues.

This chapter presents an overview of the proposed solutions to the research questions. It is organized as follows. Section 4.2 proposes a step-wise overview of the proposed trust management approach (TM-IoT). It also presents and discusses the trust-based clustering of the IoT nodes with an IoT application for trust management. Section 4.4 outlines the overview of the solution to ensure that TM-IoT is scalable and resilient against bad mouthing and bad service attacks from untrustworthy IoT nodes. This section also presents intelligent approaches for trust-based clustering of IoT nodes within an application. It also outlines approaches for trust-based migration of an IoT nodes from cluster to another cluster (within an application)

Section 4.5 presents an overview of the fuzzy-logic based approaches to counter the four different types of attacks that can be carried by the untrustworthy IoT nodes. The attacks for which solutions are outlined in this section are bad-mouthing attacks, on-off attacks, contradictory attacks and bad service attacks. Section 4.6 presents an overview of the validation approaches used in this thesis. Finally, Section 4.7 concludes this chapter.

4.2 Overview of the Solution for the Trust Management Platform for IoT-based Clustering (TM-IoT)

In this section, I present an overview of the proposed approach for enabling the scalable trust management in IoT networks (TM-IoT). A stepwise overview of the solution is as follows:

Step 1: Overview of the platform of trust management for IoT (TM-IoT):

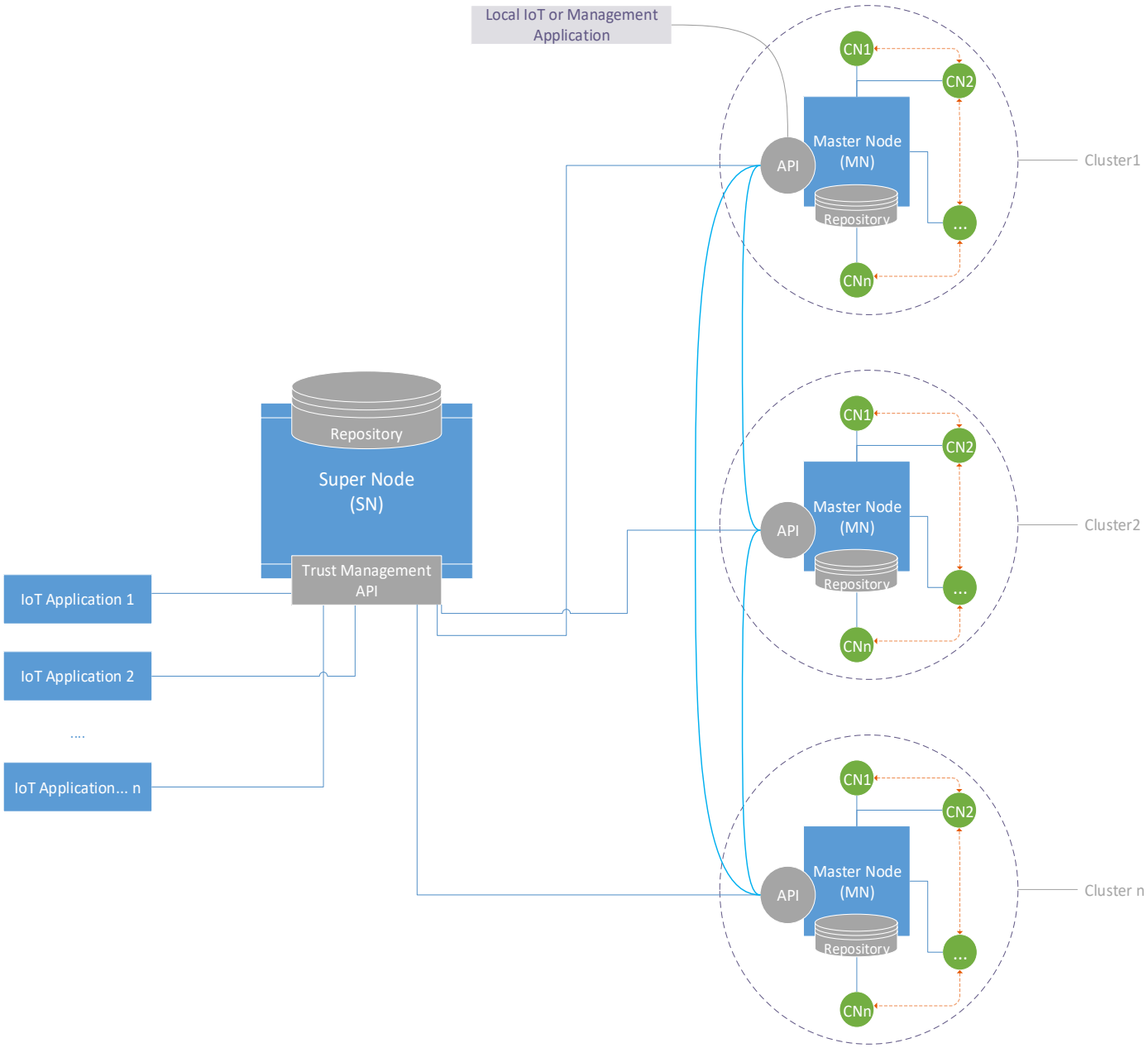

Figure 4-1 Overview of the TM-IoT platform

The TM-IoT provides a trustworthy platform for communication between all the IoT nodes or devices that communicate with other nodes in an IoT environment. In TM-IoT, the Master Node (MN) stores the trust values of all the nodes within its cluster. The Super Node (SN), in turn, stores the trust value of all the MNs. The TM-IoT platform allows heterogeneous IoT devices and applications to contact each other in trusted heterogenic-device communication.

The framework of TM-IoT shown in Figure 4-1 is a distributed architecture that consists of *Cluster Node (CN)*, *Master Node (MN)*, *Clusters* and a *Super Node (SN)*. The TM-IoT framework consists of a CN, which is an IoT node. The CN communicates with other CNs in the network or in the application and in the process delivers services to other CNs. From a trust management perspective, the MN manages many CNs in the cluster, and stores the trust values of CNs in the MN repository.

In the TM-IoT architecture, illustrated in Figure 4-1, the SN is one of the key nodes. It is responsible for building a trusted environment in IoT networks and contains an application programming interface (API), referred to as the trust management API. The API allows the SN to communicate with many MNs in the clusters. The SN also has a repository that stores the trust value and addresses of each MN and CN. The SN repository is hierarchical (tree-structured), and each entry relevant to a CN is addressed through the MN's unique ID. Thus, the SN repository does not store any data collected from the CNs directly. It only stores their trust value and address information that it gets through the MNs. An IoT application running with the SN can provide trust management services based on combined trust data collected from various CNs. Therefore, IoT applications and services are built on top of the IoT by supporting communications between nodes via the SN.

The TM-IoT architecture provides a model comprising of several clusters and an MN which allows for the central trust management of things over the internet. The TM-IoT architecture comprises of several MNs and clusters; thereby creating a distributed trust management system where CNs communicate and deliver services to each other, and

the MNs communicate with the CNs in their cluster and also with the SN in a cooperative manner. This architectural flexibility is specifically designed for the communication requirements of the IoT, given that most IoT devices may play different roles in both centralized and distributed operations setups, especially for trust management in the IoT. Additionally, it is this architectural flexibility that provides a key fundamental underpinning for TM-IoT to scale to a large number of nodes.
In Chapter 5, I discuss the complete description of the proposed scalable platform of trust management in IoT (TM-IoT), along with its components and features in details.

Step 2: Sending/receiving messages from the Super Node (SN)

The main central node in the TM-IoT platform is the SN. As defined earlier, this node controls the entire TM-IoT framework. The mechanism of sending and receiving messages is as follows: there is a message channel from MN to SN and from SN to MN. The message from MN to SN includes updates about the MN's clusters. Examples of messages sent by the MN to the SN include notifications about any new data to the respective cluster, such as updates in the cluster for example, the joining of nodes, cluster status etc.
On the other hand, there are messages from SN to MN including commands from SN delivered to MN and to the clusters, such as permission to move an IoT node from one cluster to another cluster, cutting communication off with a cluster if there is any uncompliant behaviours or if the trust value of that cluster is low etc.
I assume that the SN is fixed and is responsible for all the trust management activities in the TM-IoT platform.

Step 3: Sending/receiving messages from the Master Node (MN)

In the TM-IoT platform, the MN node is responsible for overseeing and implementing the proposed trust management approach at the cluster level. The communication between this node (MN) and the other CNs within the cluster is as follows. The trusting CN and the trusted CN send the trust values after each interaction to the MN who updates it in its repository. The MN sends and receives messages or updates from each CN within its cluster. The updates may include, trust value assigned to another CN, memory capacity etc.

Consequently, the MN of each cluster is selected based on the overall trust value of all the IoT nodes within the cluster. This process is carried out and overseen by the SN. The MN is responsible for carrying out trust-based clustering of IoT nodes and the trust value of all the CNs in a cluster are in the same range. Trust-based clustering of the IoT nodes is explained in Chapter 6.

Step 4: Sending/receiving messages from CN

The TM-IoT platform has clusters and each cluster has a number of CNs. These CNs communicate directly with the MN and indirectly with the SN (through the MN). The CN sends and receives messages from the MN. After the CNs have completed their interaction which may involve the delivery or provisioning of service by the trusting CN to the trusted CN, both of them communicate their trust value to the MN.

On receiving the trust value from the CNs, the MN sends an acknowledgement messages to confirm the receipt of the message and updates the received trust value in its repository. Also, MN gives permission to the CN if it is eligible to move to another cluster based on its trust value metrics.

Step 5: Sending/receiving messages from foreign nodes

If a node does not belong to any cluster (or is not part of the IoT application implementing TM-IoT) and if it wishes to join a given cluster, it will firstly communicate with the MN of the cluster it wishes to join. If the node desiring to join the cluster has the required trust value of the cluster, then the MN allows the appointed CN to join, otherwise its request is denied.

Figure 4-2 shows the entire steps for the solution for the trust management platform for IoT-based clustering (TM-IoT).

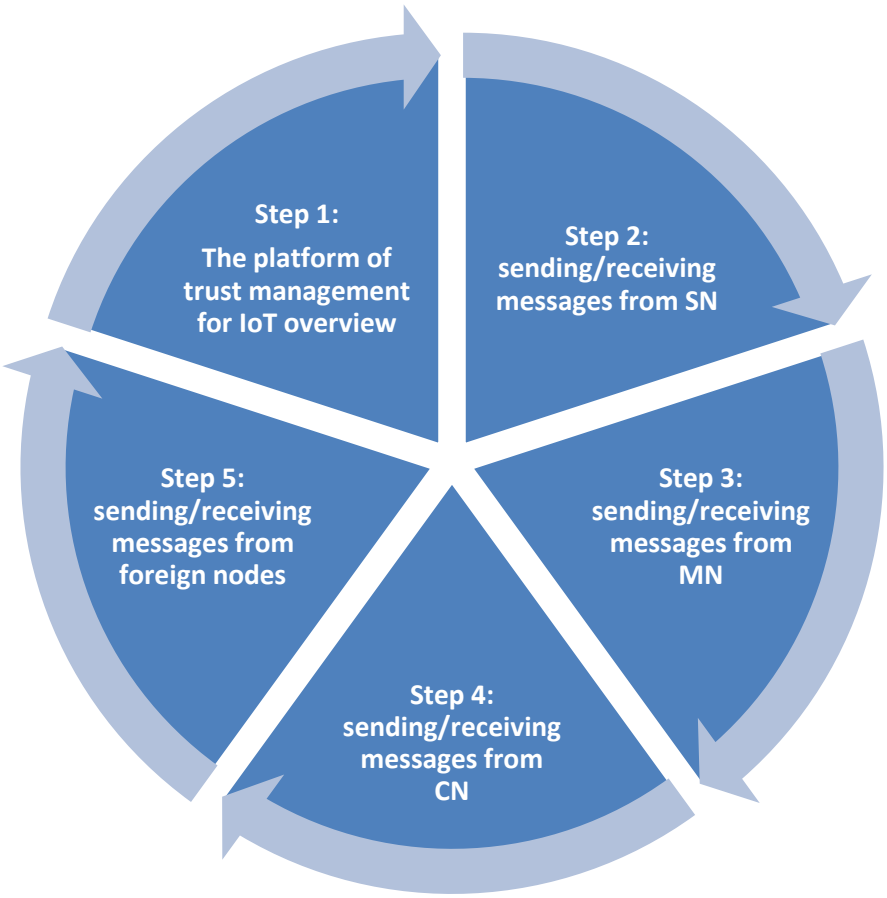

Figure 4-2 The lifecycle solution for the trust management platform for IoT-based clustering (TM-IoT)

4.3 Overview of the solution to ensure TM-IoT scalability and reliability (CITM-IoT)

This section presents an overview of the solution for trust management scalability for IoT-based clustering. I term the collection of all the approaches used to achieve scalability in TM-IoT as Clustering-driven intelligent, scalable and reliable trust management for IoT (CITM-IoT). The primary contribution of CITM-IoT is that it presents a trust-based clustering-approach for enabling IoT trust management scalability. The second contribution of CITM-IoT is that it presents approaches to counter bad service and bad-mouthing attacks within TM-IoT.

In TM-IoT, I enable the scalability of the proposed trust management framework by trust-based grouping of the IoT nodes into clusters. This framework uses a double threshold system on the MNs to limit the trust values of the CNs within the cluster and maintain efficient usage of the MN memory. Our proposed approach for scalability reveals the importance of memory management for MNs under an extreme memory condition and the effect of using statistics outliers on memory efficiency under extreme bad mouth attacks.

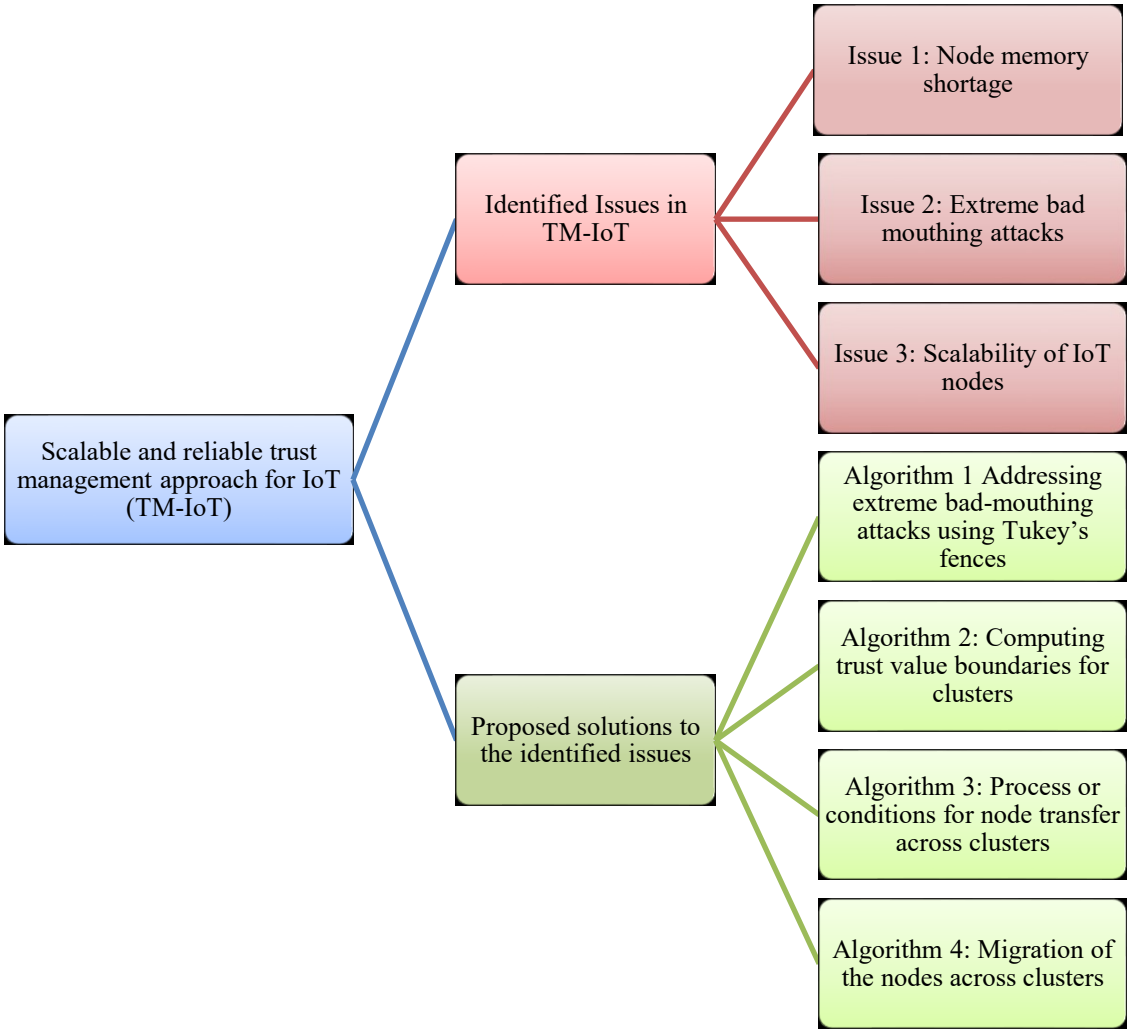

Figure 4-3 Overview of the solution for efficient trust management in TM-IoT

As shown in Figure 4-3, the proposed solution addresses three issues. The first issue is the possible memory shortages induced by the extreme memory usage of node services. This issue is addressed by Algorithm 2. The second issue is to develop intelligent approaches to counter bad-mouthing attacks. I propose the use of statistical outliers (Algorithm 1) to eliminate extreme bad-mouthing attacks on reputation systems in IoT. The last issue is to develop intelligent approaches to ensure the scalability of our proposed trust management approach. This issue is addressed by Algorithms 3 and 4. The proposed solutions follow these steps:

Step 1: Algorithm 1 – Addressing extreme bad-mouthing attacks using Tukey's fences

Algorithm 1 is used to eliminate the outliers of a set of floats using Tukey's fences. This algorithm is a proposed solution for extreme bad mouth attacks in TM-IoT.
With Algorithm 1, the first step is to break the set of inputted float values from lowest to largest into two parts. Q 1 will be the medium of the first part and Q 3 will be the medium of the second part, the interquartile range

$$Qrange = Q_3 - Q_1 \tag{4.1}$$

Then the upper threshold of the set will be computed

$$Tupper = Q_3 + 1.5 \times Qramge \tag{4.2}$$

The lower threshold of the set will be computed in a similar way

$$Tlower = Q_1 - 1.5 \times Qramge \tag{4.3}$$

Step 2: Algorithm 2 – Computing trust value boundaries for clusters

Algorithm 2 creates a new mechanism for clustering by calculating the trust value boundaries for each cluster according to the memory boundaries. Algorithm 2 is executed by the SN to allocate trust value thresholds and memory thresholds for all the MN. The trust values of all the MNs are collected from the entire interactions between the CNs in each cluster. The average of these trust values is calculated and input into a set *trustMn*. Then, total memory *Mt* is retrieved on all MNs with *Ms*, memory permanently used by service. *Ms* is calculated on the MN by multiplying *Mt* with a global predefined service rate *Rs* to simulate a heavy use of memory.

$$Ms = Mt \times Rs \tag{4.4}$$

Total memory available, *Ma*, is calculated by subtracting *Ms* from *Mt*. *Ma* is part of the set *Sma*.

$$Ma = Mt - Ms \tag{4.5}$$

Based on *Ma*, the maximum number of nodes connected to this MN *Mnm* is calculated by Equation 4.6. The result is placed into the set *Smnm*

$$Mnm = Ma \div Me \tag{4.6}$$

Me is a constant, defining one set of connection information. It consists of the IP address and trust value of a CN. Next, sort *Sma*, *Smnm* according to *trustMn* from the smallest to the largest. In the latter part of this algorithm, this sorting ensures the consistency of the trust values between the connected MNs and CNs. The next step is to compute the memory rates from *Nmc* the sum of *Smnm* elements. *Nmc* is the total

number of CNs possible in the system. Memory minimum bound rate *Rmm* and memory upper bound rate *Rum* are calculated by the following equations.

$$Rmm = \frac{numCn}{Nmc} \tag{4.7}$$

$$Rum = \frac{Rmm + (1 - Rmm)}{256} \tag{4.8}$$

where *numCn* is the total number of CNs in the system. The intention of *Rum* is to give every MN a small amount of memory allowance to allow some memory for CN movement between the MNs. The number 256 is chosen to provide a suitable memory allowance. The higher this number, the lower the memory allowance. A high memory allowance leads to an unbalanced distribution of CNs to MNs, whereas a low memory allowance results in a high restriction on CN movement. Then, the set of minimum memory boundaries and the set of upper memory boundaries are formed from equation 4.9 and 4.10.

$$Smm = Sma \times Rmm \tag{4.9}$$

$$Sum = Sma \times Rum \tag{4.10}$$

After this, all the trust values of the CNs stored on the MNs are gathered and sorted from the lowest to highest into a set *Stcn*. A temporary 2-dimensional set *tempTrust* is created. Trust values in *Stcn* are considered from the middle to the boundaries and added to *tempTrust*. Values are added to first the middle of *tempTrust* towards the boundaries. During the process of adding a temporary, memory usage is calculated.

$$Tmemory = (tempTrust[i].size + 1) \times Me \quad (4.11)$$

If *Tmemory* exceeds *Smm[i]*, it adds the value to the next set of *tempTrust*. Finally, the trust upper bound *Tupper* and trust lower bound *Tlower* are computed. When calculating *Tupper* for *tempTrust[i]*, *tempTrust[i+j]* is considered. It continues to look for the next *tempTrust* until the highest number of *tempTrust[i]* is not equal to the highest number of *tempTrust[i+j]*. Then *Tupper* is calculated using equation 4.12.

$$Tupper = \frac{[max(tempTrust[i]) + min(tempTrust[i+j])]}{2} \quad (4.12)$$

Tlower is calculated in a similar way. *TempTrust[i-j]* is considered. It continues to look for the next *tempTrust* until the lowest number of *tempTrust[i]* is not equal to the highest number of *tempTrust[i-j]*. Then *Tupper* is calculated using equation 4.13.

$$Tlower = \frac{[min(tempTrust[i]) + max(tempTrust[i+j])]}{2} \quad (4.13)$$

Finally, all *Smm*, *Sum*, *Tupper* and *Tlower* are assigned to the consistent MNs.

Step 3: Algorithm 3 – Process or conditions for node transfer across clusters

Algorithm 3 defines that the conditions for a CN are able to change to a specified new MN. Algorithm 3 is used to accept a request to transfer a CN between clusters. It gathers the trust values of a CN from the cluster peers of the current cluster and cluster peers of the target cluster into a list *trustValues*. Then, Algorithm 1 is executed for *trustValues* to eliminate outliers. The outliers are the badmouthed trust values. *Tv* will be calculated as average of *trustValues*. If *Tv* is in the trust bound of the target MN and adding CN will exceed the memory threshold of the target MN, CN and CN's trust value record is removed on the current MN and transferred to the target MN.

Step 4: Algorithm 4 - Migration of the nodes across clusters

The MN uses Algorithm 4 to monitor the CN trust values and decide to move some CNs away. A check is then made on the memory of the current MN to decide if further CNs need to be moved to different clusters for the purpose of trust management.
The purpose of Algorithm 4 is to update and check the trust values of the CNs of the MNs. Also, the memory usage of the MNs is also checked. If the trust value or memory exceeds the respective thresholds, the MN uses Algorithm 3 to determine if it can move certain CNs to other clusters. Firstly, the trust values of a CN are collected from the neighbour CNs. Then, the outliers are eliminated by Algorithm 1 and *Tv* is calculated as an average. The trust value of CN is updated to *Tv* and stored on the MN. If *Tv* is not in the range of *Tupper* and *Tlower* of the current MN, it is added to the list *trustOut*. For all CNs in *trustOut*, compute a list of MNs where the *Tv* of the CN is in range of the trust bounds of the MNs. This list of MNs is the *trustMNodes*. For all MNs in the *trustMNodes*, execute Algorithm 3 with the minimum memory bound of MN. If CN is removed, terminate the algorithm. If all MNs of the *trustMNodes* are considered in Algorithm 3 and CN is not removed from the current MN, run Algorithm 3 again for all MNs in the *trustMNodes* with the upper memory bound for MNs.

The second part of Algorithm 4 triggers when the memory of the current MN is over the upper memory bound. In this case, the number of nodes needing to be removed is computed. To calculate this value, the current memory usage is required. This value doesn't consider the service memory and *Ma* is the current memory available.

$$Mi = Mt - Ms - Ma \tag{4.14}$$

The number of CNs needing to be moved is *Num* and is calculated by the upper memory boundary *mum*.

$$Num = \frac{(Mi - mum)}{Me} \tag{4.15}$$

Then Algorithm 3 is executed for all CNs targeting all other MNs using their minimal memory bound until the *Num* of the CNs is removed. After this process, the *Num* of CNs is not removed, and Algorithm 3 is run again for all CNs targeting all other MNs using their upper memory bound until the removed number of CNs reaches *Num*.

In the next section, I present an overview of the fuzzy-logic based approaches to counter the four different types of attacks that can be carried by the untrustworthy IoT nodes on TM-IoT. The attacks for which solutions are outlined in this section are bad-mouthing attacks, on-off attacks, contradictory attacks and bad service attacks. I term the collection of all these approaches to counter untrustworthy behaviour in TM-IoT as Fuzzy-IoT.

There are significant differences between the solutions presented in Sections 4.4 and 4.5. Section 4.4 presents an outline of the solution on how to protect the TM-IoT platform from bad-mouthing attacks based on clustering techniques whereas Section 4.5 shows how to maintain the reliability of TM-IoT by proposing fuzzy-logic based approaches to the above mentioned four different types of attacks on TM-IoT.

The detailed working of the above mentioned algorithms in CITM-IoT along with experimental results and evaluation is presented in Chapter 6.

4.4 Overview of the solution to detect malicious nodes compromising TM-IoT (Fuzzy-IoT)

The purpose of this solution is to develop security protocols addressing critical security issues in TM-IoT. By 'protocol', I mean a sequence of steps to solve a particular security issue. In this approach, I propose a fuzzy-logic-based approach to detect malicious nodes in the IoT network and isolate them in the IoT network. This protocol uses a message system similar to serial communication for secure message encryption. I term the collection of all these approaches as Fuzzy-IoT.

This solution comprises five sequential algorithms as follows:

(a) Algorithm 1 is used to classify trust score values of the IoT nodes into fuzzy sets. This is used to carry out fuzzy-logic based clustering of IoT nodes

(b) Algorithm 2 classifies the CNs into three categories: trusted, semi-trusted, and non-trusted. These categories enforce restrictions on node interaction.

(c) Algorithm 3 uses a direct trust score, indirect trust score and routing score to calculate a node's overall trust value.

(d) Algorithm 4 uses the overall trust value to determine if a CN is able to change to another cluster.

(e) Algorithm 5 checks the current condition of CNs and uses all algorithms to update a new fuzzy status and new trust value. Furthermore, it uses trust values for clustering.

Step 1: A secure HEXA decimal-based messaging system for tamper detection

The structure of a message is shown in Table 4-1. Each unit/code of the message is a two-digit hexadecimal number or an unsigned char with values from 0 - 255. Message Length refers to the length of the data section. Check Code is generated by processing other hexadecimals in the message. In our simulation, the Check Code is generated by adding the Data Sections together. If the result is greater than 255, reduce 256 from the result to avoid exceeding the numerical maximum of a two-digit hexadecimal 255 until it is smaller or equal to 255. The message system provides two extra layers of security. If the head code of a received message is wrong, the current message is discarded, protecting the system from an outside source or malicious nodes. A different check code implies a wrong check code generation mechanism, which means the message is from an unsecure origin.

Table 4-1 A Description of the general structure of a Message

Code	Description	Length (bytes)
Head Code	The head code is the same for all messages within the system	1
Id Code	Determines which operation should be performed with the current message	1
Data Length	Determines the length of the data section	1
Data	Data required for the operation performed by the current message	Depends on the operation
Check Code	Calculated by the code before sending the message. It is	1

	calculated again at the receiver node and compared to verify the validity of the message	

Step 2: Algorithm 1 uses fuzzy boundaries to determine three trust scores of an IoT node - low trust score, medium trust score and high trust score. These trust scores are in the range of 0 to 1.

Algorithm 1 evaluates an input value with three fuzzy sets (low, medium and high) trustworthy and obtaining a set of three 1 or 0 results representing membership or non-membership to the sets. Each of the values is tested with a set of three equations with an upper bound and lower bound. For example, with the medium number set, a value is compared that is between the lower bound of medium set Ml and the upper bound of medium set Mu.

$$Ml < value < Mu \tag{4.16}$$

In our proposed approach, there are three trust scores as components of the trust value. A direct score is generated from the quality of a service response. When a new direct score is stored, the last one becomes the new history score. Routing scores are generated by evaluating service responses from a node in a different cluster. Putting these three scores into algorithm one generates the fuzzy membership for the three trust scores.

Algorithm 2 uses the results of algorithm one to determine the state of a given IoT node. There are three possible states for an IoT node. All IoT nodes start with the non-restricted state *trusted*. As more nodes give more trust scores, it could move down to more restricted states. A *semi-trusted* node can only provide service to nodes within the same cluster. *Semi-trusted* nodes are restricted to provide services to an outside node. A service request to a *semi-trusted* node is blocked by its MN.

Step 3: Algorithm 2 is based on fuzzy logic as follows. If the routing score (*Rr*) is low AND the direct score *Rd* is NOT low AND past score (*Rp*) is not low, this node is *semi-trusted*. If *Rd* is low OR *Rp* is low then this node is *non-trusted*. All nodes start with a trusted state. The state can only downgrade towards the *non-trusted* state.

Directly using the average of trust scores obtained from other CNs and other MNs, a trust value can be obtained. Three direct scores, past score and routing score coefficients *Cd*, *Cp* and *Cr* are values with a sum of 1. These coefficients are used to provide a weighted average of direct, past and routing scores (*Srd*, *Srp*, *Srr*). In our simulation, an average is simply provided.

$$result = Cd \times Srd + Cp \times Srp + Cr \times Srr \tag{4.17}$$

Step 4: When a node's trust value is not within the boundaries of the current MN, Algorithm 4 uses the trust value calculated from Algorithm 3 and checks if it is within the range of other MNs' trust value boundaries (*Tnl* and *Tnu*). These boundaries can be requested from a SN.

Algorithm 5 specifies the process of a MN checking the status of the CNs and decides if it needs to move them. The direct, past, routing scores (*Sdi*, *Spa*) are obtained from other CNs and the sum of *Ssdi* and *Sspa* is calculated.

$$Ssdi = \sum Sdi \tag{4.18}$$

$$Sspa = \sum Spa \tag{4.19}$$

The averages of these scores, *Asdi* and *Aspa*, are then calculated. Then, routing scores *Sro* are obtained from the other MNs to calculate *Ssro*.

$$Ssro = \sum Sro \tag{4.20}$$

Again, an average *Asro* is calculated. Algorithm 2 and 3 are used to produce a fuzzy state (*NTs*) and a trust value (*Trust*), respectively. The fuzzy state will be broadcasted to other same CNs. The trust value obtained is checked against the upper and lower trust boundaries (*Tu*, *Tl*) of the current MN. If it is not within the boundary, algorithm 4 will be run to check if this node can move to another cluster.
The detailed working of the above mentioned algorithms in Fuzzy-IoT along with experimental results and evaluation is presented in Chapter 7.

4.5 OVERVIEW OF THE VALIDATION APPROACH

I validated our approach by applying this research in the IoT simulator tool (Cooja). In addition, I used the Java programming language to measure the trust values for the components of our proposed platform (TM-IoT). For the research issues, I used the IoT simulator tool Cooja and the Java language to calculate the trust values for the TM-IoT platform. Below I provide an overview of the validation approaches that I have used in this thesis. The detailed experimental results and validation processed have been explained in Chapter 5, Chapter 6 and Chapter 7.
First, I created our TM-IoT platform including the components that provide the IoT nodes to interact with each other and subsequently assign a trust value.
Second, I engineered our proposed CITM-IoT solution using IoT simulator tool Contiki Cooja (Contiki 2018) and Java programming language. The performance of the various

algorithms in CITM-IoT was captured using the developed prototype and evaluation benchmarks.
Finally, I evaluated the performance of our proposed Fuzzy-IoT approach using the IoT simulator tool Contiki Cooja and Java programming language as well. The fuzzy-logic based algorithms in Fuzzy-IoT were validated using trust metrics.

4.6 CONCLUSION

In this chapter, a general overview of the proposed trust management solution (TM-IoT) was presented. According to the research problems identified in Chapter 3, an overview of the components and the step-wise working of the proposed TM-IoT was presented. The first section presents an overview of the TM-IoT platform. Then, an overview of the trust-based clustering approach to achieve scalability of the TM-IoT is presented. An overview of the algorithms used to achieve this is presented and discussed. Finally, an overview of the approaches to ensure the reliability of TM-IoT by presenting intelligent approaches to detect untrustworthy behaviour by the IoT node within the framework of TM-IoT were presented.

In the next chapter, the working of the trust management platform for IoT (TM-IoT) is given.

CHAPTER 5

TRUST MANAGEMENT PLATFORM FOR THE INTERNET OF THINGS (TM-IOT)

5.1 INTRODUCTION

This chapter presents the trust management platform for the Internet of Things (TM-IoT). TM-IoT aims to provide a trusted platform that can scale seamlessly using the proposed TM-IoT components. Also, the TM-IoT platform provides a platform for trusted communication between all components and enables IoT applications to support machine-to-machine (M2M) over the Internet. The TM-IoT provides trust management for various services and applications over services running on cluster nodes. The entire platform is managed by the super node (SN), and each cluster is managed by its master node (MN). The design of the proposed TM-IoT platform takes into account the intrinsic attributes of the devices that communicate using the platform, such as limited power, limited memory of the nodes. This TM-IoT platform protects all the nodes in the platform based on the trust metrics (using trust-based decision making) that are applied in it. The main components of the TM-IoT platform are the super node (SN), master node (MN), cluster node (CN) and clusters. The proposed TM-IoT platform is introduced in Section 5.2. Section 5.3 discusses the SN with its components (API

Module, Trust Management Module and Repository and Communication Module). Section 5.4 presents the MN along with its components. Section 5.5 discusses the cluster components. Section 5.6 describes the CN components with its mechanisms. Section 5.7 concludes the chapter.

This chapter contains sections that have been published in (Alshehri & Hussain 2017; Alshehri, Hussain & Hussain 2018).

5.2 TRUST MANAGEMENT FOR THE IOT PLATFORM MECHANISM (TM-IOT)

In this section, I present a novel centralized approach for trust management in the Internet of Things (IoT). I demonstrate the overall working mechanism and components of the proposed trust management mechanism for the Internet of Things (TM-IoT), which is designed to provide trustworthy communication and service delivery between the IoT nodes in an IoT application of a network

Figure 5-1 shows the overall platform of the TM-IoT which includes a Super Node (SN) as the centralized trust manager node. To achieve trustworthy communication between nodes, I propose dividing the IoT environment into clusters. Each cluster has a local trust manager called the Mater Node (MN). There are also multiple Cluster Nodes (CN) in each cluster which communicate with each other, interact with each other and where requested deliver services each other. The outcomes of their interactions are supervised by the MN. The SN has a central repository to store the trust data for all MNs and CNs for the entire IoT platform, and MNs have local repositories in which the trust values for the CNs in each cluster are stored.

TM-IoT provides trustworthy communication amongst devices that are part of an IoT application or an IoT network. The TM-IoT architecture allows various IoT devices

and applications to establish a trusted communication amongst themselves, thus paving the way towards secure communication and a secure environment. The architecture of the TM-IoT is presented in Figure 5-1. Any type of IoT device such as sensors, actuators etc. can join the TM-IoT platform.

The TM-IoT structure is inspired by the structure that is used by traditional network management approaches. The TM-IoT architecture consists of a CN, which resides on a node and a MN. The MN manages many CNs in the cluster. Each CN in a cluster sends regular updates to its corresponding MN about its status such as memory power consumer, trust value assigned to other nodes etc. The MN stores the updates received from the CN including the trust value, in its repository.

The SN is a crucial node in the TM-IoT on which the entire architecture is centred. It is responsible for ensuring trust in the IoT network. The SN contains an application programming interface (API), which I refer to as the trust management API. This API allows the SN to communicate with the MN of a given cluster. The SN also has a repository that stores the trust values and addresses of MNs and CNs. The repository of the SN is hierarchical (tree-structured). Each entry relevant to a CN is addressed through the MN's unique ID. Therefore, the repository of the SN does not store any data collected from the CN. It only stores its trust values and address information i.e. through which MN and CN can be accessed. An IoT application running makes use of the services of the SN for trust assessment for the MNs. Therefore, IoT applications and services are built on top of the network by supporting trustworthy communications amongst the CNs.

The TM-IoT architecture provides a model comprising of several clusters and a MN that allows for the central trust management of things over the internet. On the other hand, the TM-IoT platform comprises of several MNs and clusters creating a distributed trust system. The overall proposed TM-IoT platform is managed by a central node (Super Node). However, there can be multiple distributed clusters with each cluster managed by a Master Node that is responsible for implementing the trust

management framework at the cluster level. This hybrid combination is key to the scalability of TM-IoT, while the approaches built for ensuring the reliability of TM-IoT can enable a trustworthy environment, that the integrity of the trust management framework

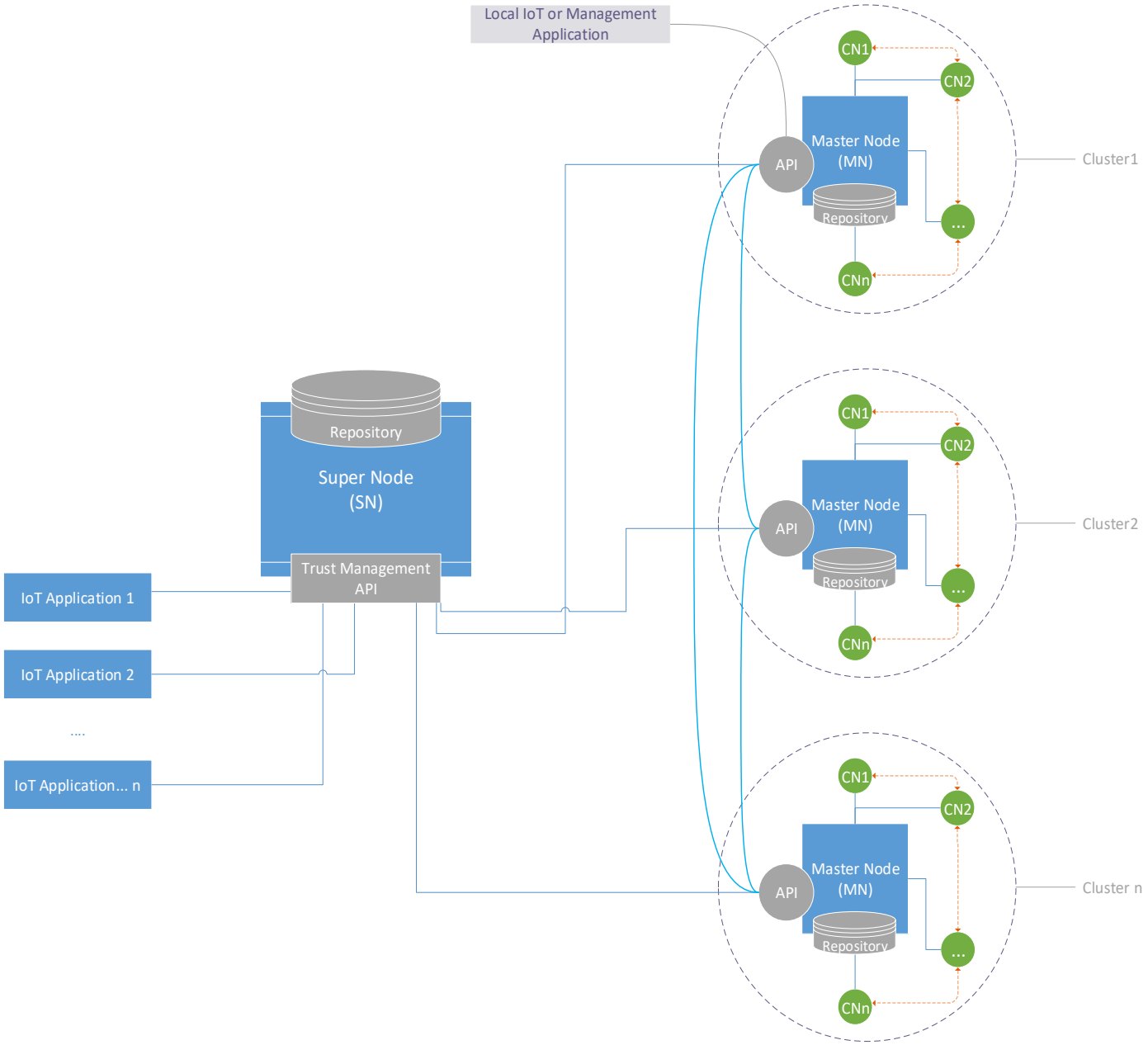

Figure 5-1 A Centralized platform for trust management of IoT mechanism (TM-IoT)

5.3 SUPER NODE (SN) MECHANISMS AND MODULES

The SN provides centralized trust monitoring and mapping services for the TM-IoT. The concept of the SN is similar to that of a router, which carries out the function of directing traffic; however, the SN performs another function by providing a trust management service for the IoT application using the TM-IoT platform. The SN is responsible for monitoring MNs and it receives regular trust management data from MNs about their respective clusters. Any IoT application can access the SN remotely over the Internet (using the SN's API). The IoT application can request trust data about any CN by sending a request to the corresponding MN of the CNs cluster.

As noted above, the SN is the main node in the TM-IoT platform, and it communicates with many MNs of different clusters. Communication between the SN and MNs takes place over HTTPS and via their APIs. The SN has a trust value repository for all the MNs in the TM-IoT platform, in which the trust values of each cluster's MN and the membership of the CNs in that cluster. The SN repository does not store CN functions but works as a routing table to store the trust value data and network topology and to direct which nodes should join which cluster in the TM-IoT platform.
The table of the SN repository has three fields, as follows:

- The master node ID (MNID): denoting which MN belongs to which cluster.
- The cluster node ID (CNID): lists the CNIDs of each cluster MN.
- IP addresses of MNs.

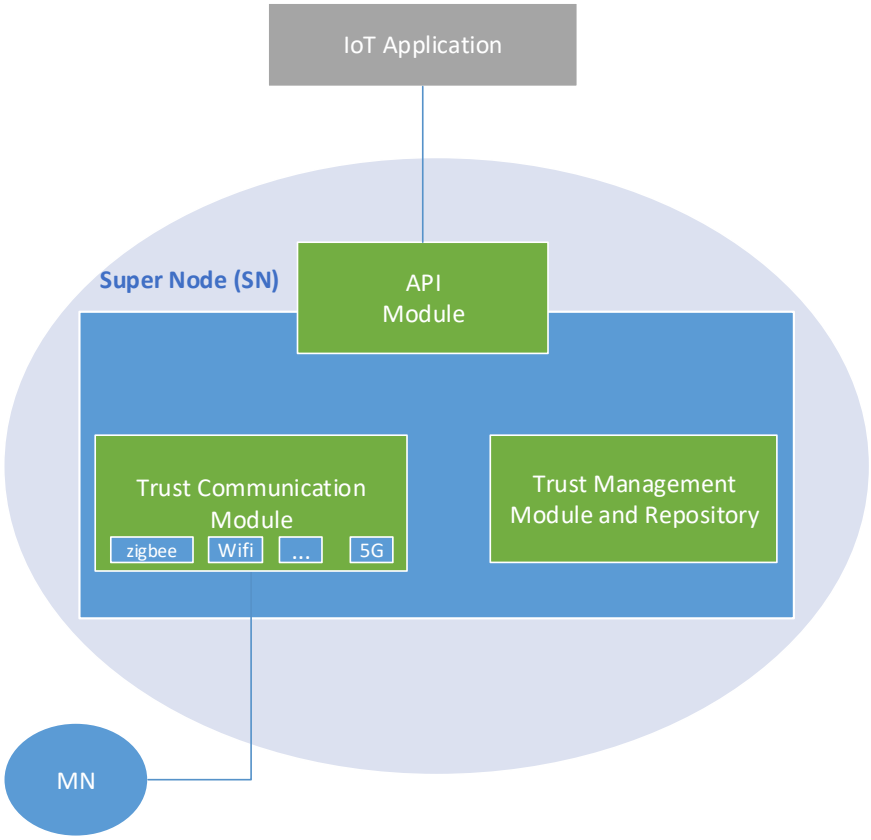

Figure 5-2 Centralized IoT super node (SN) modules

The design of the SN uses a modular format to develop the capabilities and functionalities of the SN. Figure 5-2 shows the three modules of the SN:

1. API Module
2. Trust Management Module and Repository
3. Trust Communication Module

These modules are described in the subsections that follow.

5.3.1 API MODULE OF THE SUPER NODE (SN)

The application programming interface (API) improves interface communication for IoT applications. It assists various architectural components of the proposed TM-IoT solution (such as MNs) and other IoT application to interact with each other. The API also allows IoT applications to retrieve the CN data which is stored in the repository. The MN interacts and sends trust management instructions to the relevant CNs. The design pattern for the API is Representational State Transfer (REST), which supports independent languages and platforms such as UNIX, IOS, Android, and Windows, and thus does not require the use of a specific programming language such as Python. Figure 5-3 shows the components of API module with TM-IoT.

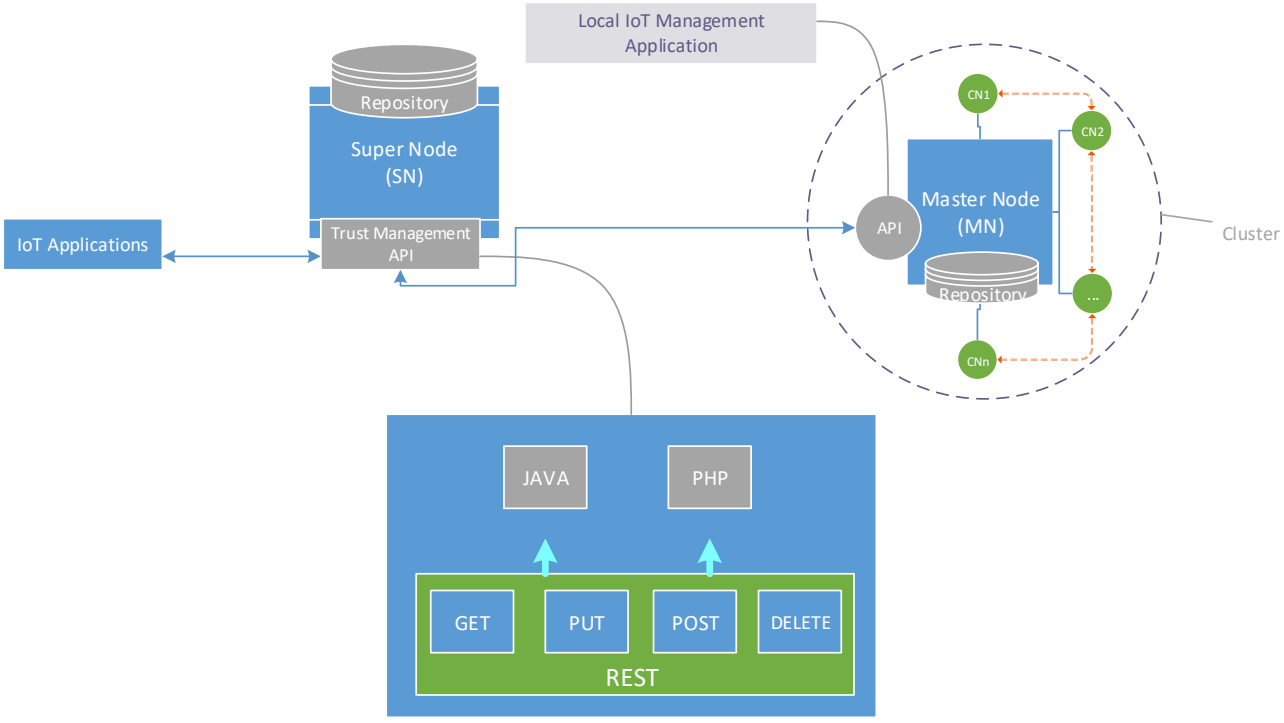

Figure 5-3 API module components

The commands available for the SN API are as follows:

a) **GET:** used by the IoT application to request the CN data which is stored in the MN's repository.
b) **PUT:** used by the MN application to send and update the data to the SN's repository.
c) **POST:** used by the SN application to send SN instructions to the MN.
d) **DELETE:** used by the MN application to delete data from the SN repository.

5.3.2 TRUST MANAGEMENT MODULE AND REPOSITORY

The trust management model has two key responsibilities: trust-related communications between the SN and MNs, trust-related communications between the MNs and CNs (including between the CNs in a cluster).

In the trust communication between the SN and MN, the MN authenticates and establishes contact with the SN by providing its main authentication data (MNID) to the SN. The SN has a list of all the registered MNs along with its MNID in its repository. Once the MN has been authenticated to supervise the trust management activities in the corresponding cluster, it is authorized to register the CNs. For trust-related communications between the MN and CNs, each CN should send its data identification (CNID) to the MN (for authentication). The MN will check whether or not the trust value for the CN falls in the range of trust values for its cluster prior to making a decision of allowing it into the cluster. If the CN is accepted by the MN (based on its trust value), the CN's trust value will be registered in the MN's repository. The CN can then subsequently interact with other CNs within the cluster (neighbor nodes) and also interact with the corresponding MN.

5.3.3 THE TRUST COMMUNICATION MODULE (TCM)

The trust communication module (TCM) is the module which manages and controls trust communication between the MN and the CN and between the cluster and the SN. The communication in TCM consists of two types of messages: *management messages* and *trust value messages*.

(a) The MN is responsible for implementing and enforcing the trust management approaches at the cluster level, i.e. between the individual IoT nodes within a cluster. On the contrary, the SN is responsible for implementing and enforcing the trust management approaches at the TM-IoT platform level.

(b) Trust value messages are used to obtain the trust value status of the CNs and send updates to the MN. These trust messages take two forms, *Receive* and *Send*. *Receive* messages are used by the SN to accept messages from the MN, and for the MN to receive new trust value information from the CNs. On the other hand, the *Send* messages are sent by the CNs to communicate trust value updates to the MN, following which the MN updates the SN. The definition of these messages is given below.

Program listings or program commands in the text are normally set in typewriter font, e.g., CMTT10 or Courier.

- **Receive (CNID, TrV Array[]):** The Receive () message receives the updated messages and other alerts from the CN to the MN, and from the MN to the SN. The Receive () message has two main parameters: the first parameter is the CNID, which is the exclusive identifier ID for the cluster node. The second parameter is the TrV Array []. This is an array format which stores the message content of the new trust value of the CN and sends it to the MN. Table 5-1 shows the trust value levels and description in TM-IoT.

Figure 5-4 Receive message format

CNID	TrV
Receive message ()	

Figure 5-4 shows the structure format and the main parameters for the receive message.

Table 5-1 Trust value levels used in TM-IoT

TrV Index	Description	Status
0	Completely Not Trusted	Extremely Harmful
0.1	Semi-Not Trusted	Very Harmful
0.2	Risk Trust	Risk
0.3	Low Trust	Medium Risk
0.4	Medium Trust	Low Risk
0.5	Semi-Trust	Semi-Safe
0.7	Trust	Safe
0.8	High Trust	Safe
0.9	Very High Trust	Safe
1	Completely Trust	Completely Safe

- **SendUpdate (CNID):** The SendUpdate() message is generated by the MN to request and send an update from the CN. The SendUpdate() message has one main parameter which is the unique ID of the cluster node to avoid any overlap or miscommunication.

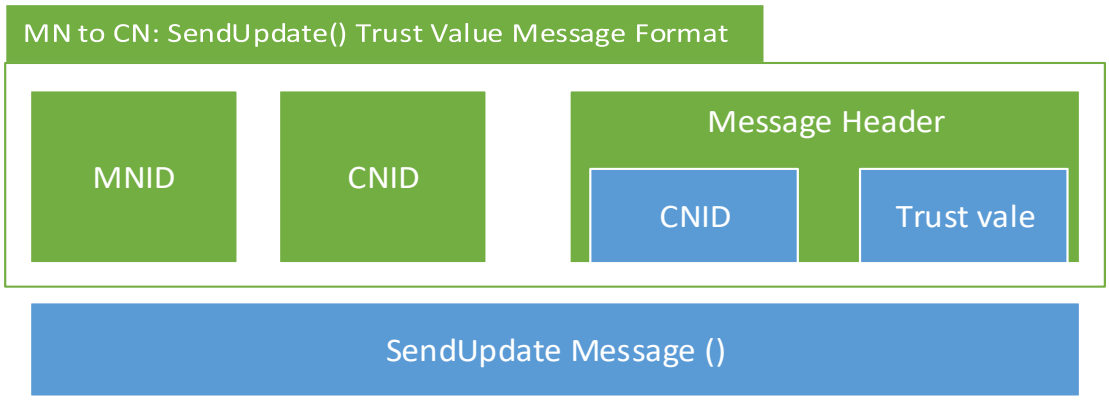

Figure 5-5 SendUpdate trust value message format MN to CN

Figure 5-5 shows the format of the SendUpdate message sent by an MN to a CN. For example, SendUpdate(trust value) is used to obtain a new update about the trust value of a particular CN in the cluster.

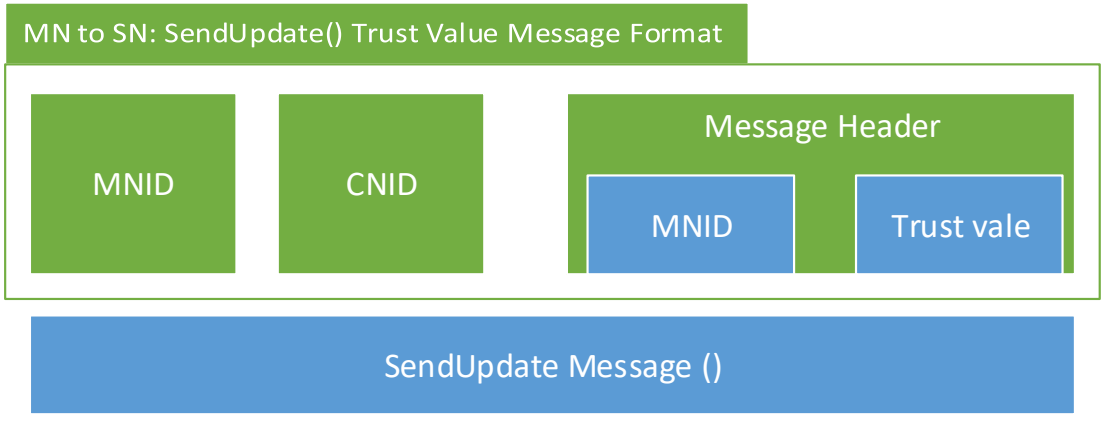

Figure 5-6 SendUpdate trust value message format MN to SN

Figure 5-6 shows the format of the SendUpdate message sent by an MN to the SN. For example, SendUpdate(trust value) is used to update the SN about the MN trust value.

- **Response (MessageID, CNID, Array[]):** this is a message from the CN in reply to the MN SendUpdate() message. The message Response() has three main parameters: the MessageID, which is handled by the MN (and matches the response to the original request sent to the CN). The second parameter is the CNID, which is the unique ID for the appointed CN. The last parameter is the content of Response() message, formatted in array[].

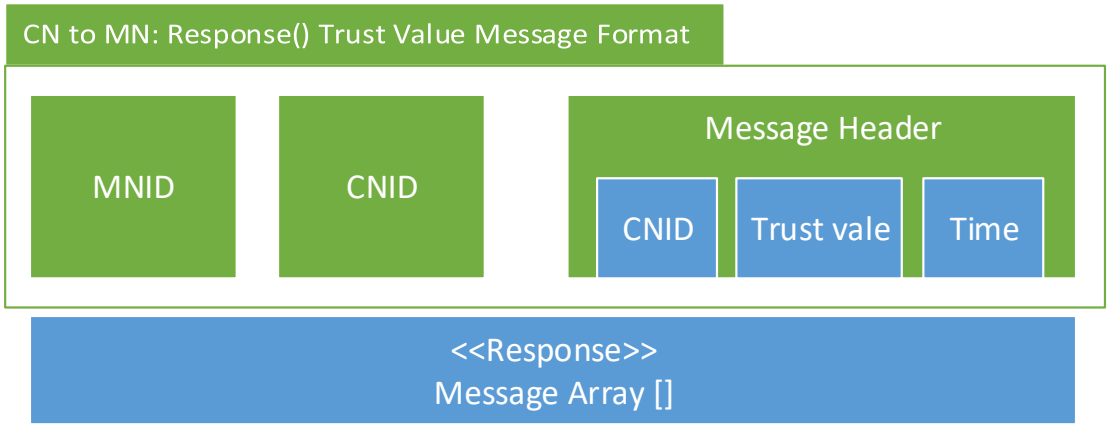

Figure 5-7 Response trust value message format CN to MN

Figure 5-7 shows the format of the SendUpdate and Response message sent by an MN to a CN. The message header includes the time of the request and the message reply.

5.4 THE MASTER NODE (MN) COMPONENTS

As I defined in Chapter 4, in TM-IoT I consider a cluster as a group of nodes with a similar trust value in the IoT network. An MN receives regular updates from the CNs such as the trust value assigned to another neighbour CN, memory being used etc. The MN controls the interactions amongst nodes in the cluster. It also manages the entire cluster so that it is secured and trusted. An MN maintains a trust value repository, which stores the trust value and other information about all CNs in the cluster. The MN accesses CNs' data that is stored in its repository. An MN also communicates with the SN to update it about its activity (such as getting trust values, interacting with other neighbour nodes etc.), and to send the trust value of the whole cluster to be stored in the SN repository.

An MN provides a trust management Application Programming Interface (API) which is based on the robust architecture of the TM-IoT. CNs' trust information, which is stored in the MN repository, can be requested over the Internet once the API allows IoT applications to communicate with a MN. Then MN evaluates whether that IoT node is sufficiently trustworthy to deliver it to the particular CN (within its cluster) or not. The trust management API is used to communicate topology information over the Internet about the clusters and IoT network to the SN. The SN is a higher IoT node that is responsible for the trust management of the entire trust IoT network. In a hierarchical manner, the SN is located above the MNs and the CNs. Under the supervision of the MN, the IoT applications run on top of the SN, who in turn sends trust requests to access the MN's trust data. Figure 5-8 shows the main components of MN. To provide trustworthy communication in the cluster, the MN includes several modules that are shown in Figure 5-8.

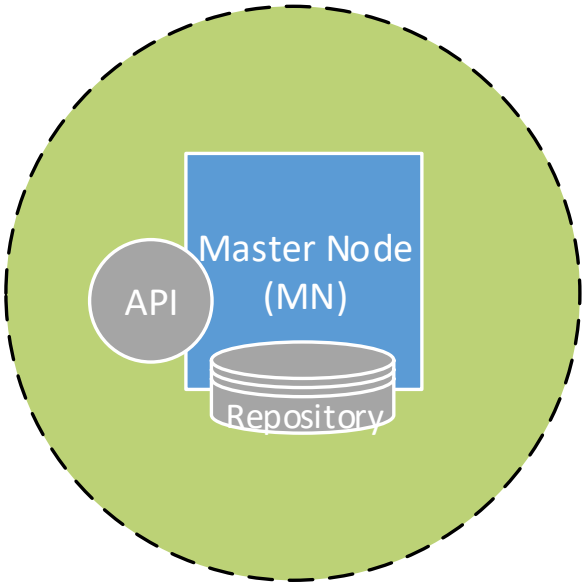

Figure 5-8 Master node (MN) components

5.5 THE CLUSTER COMPONENTS

The purpose of the trust management IoT platform is to divide IoT nodes into different clusters based on their trust value. Based on each cluster's overall trust value (and the trust ranges associated with each cluster), nodes will request to join clusters. This method provides efficient and trusted IoT communication, by determining the trust boundaries for clusters and providing the trust-based membership of IoT nodes in the clusters. This approach provides a memory-efficient, reliable and scalable approach for trust management in IoT. Nodes within the clusters contact the MNs to get authorisation to join or redirect the node to another cluster. They request joining the appropriate cluster based on its trust value.

The cluster has two main functions. The first function is to provide the basis for providing a trustworthy IoT environment centred on clusters. This allows nodes that have similar trust values to join the clusters. The cluster will then deal with other clusters rather than nodes. Any IoT nodes in a cluster has the attributes listed in Table 5-2. Figure 5-9 shows the components of the cluster.

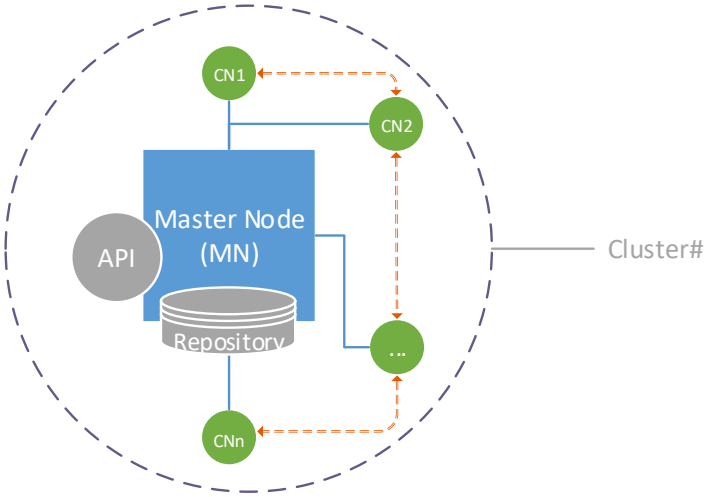

Figure 5-9 Cluster components

5.6 The Cluster Node (CN) Components

As defined in Chapter 3, I regard a cluster node as an IoT node that logically falls within a cluster. The CN communicates and interacts with other CNs and in this process and they assign a trust value based to each other based on the quality of services provided. In particular, CN is represented using the Trust Management Attributes (TMA). These attributes represent parameters such as the communication status of CN. A TMA is used to establish communication and make trust-based decisions of interaction with

other nodes. TMAs are also used to maintain the data CNs collect. They are also used to maintain information relating to CN activities. All a CN's TMA attributes are stored in its corresponding MN. Table 5-2 shows the TMA that represent a CN.

Table 5-2 IoT CN trust management attributes (TMA)

Trust Management Attributes (TMA)
• ID • PKI • Trust value • Firmware version • IPv6, MAC address, network name or others • Battery life • Location (as a heterogeneous node)

The CNs in TM-IoT have more processing capability relative sensors in networks such as WSN. Furthermore, each CN has unique identification and trust attributes. The MN is responsible for identifying the key TMA attributes for each CN. Table 5-2 shows that the TMAs are descriptions of the IoT node based on trust attributes. For instance, the ID is used to identify a unique IoT node. A public key infrastructure PKI, trust value, firmware version, and the rest of these TMAs are also used as identifiers or descriptions for IoT nodes. These trust attributes assist MNs to decide which nodes are eligible and appropriate to join their cluster, based on the similarity of the trust value between the node and the cluster. These trust attributes support the CN in establishing a trusted communication among the CNs within the cluster.

Figure 5-10 shows the CN component and demonstrates the main module. The *trust communication module* is responsible for establishing communications that are trustworthy among other CNs and communicates with the MN.

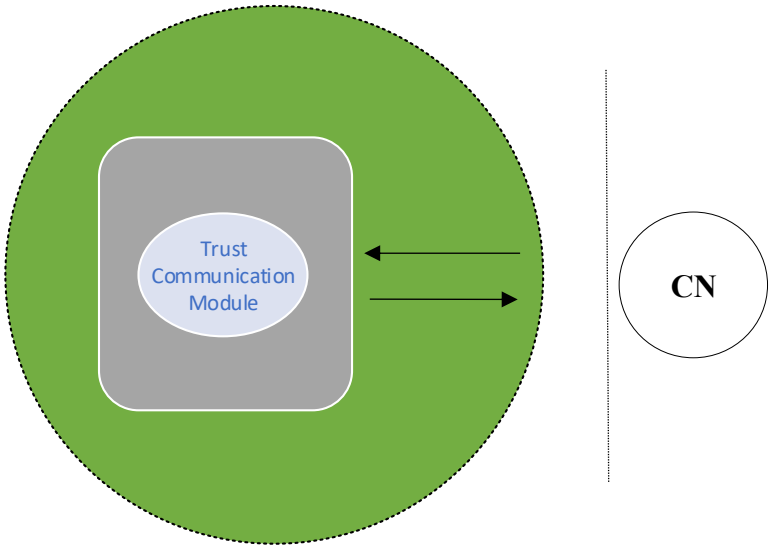

Figure 5-10 Cluster node (CN) component

5.7 CONCLUSION

This chapter introduced the trust management for the Internet of Things platform (TM-IoT). This proposed model consists of an SN, an MN, and CNs. The TM-IoT provides IoT devices with a mechanism that allows them to negotiate and establish trust among them. The model relies on a distributed architecture, in which the SN assigns trust values to the cluster components. The communication between the SN and the cluster component is facilitated by the MN. In this components the architectural design and working of the entire TM-IoT platform, including SN, MN, CN and clusters was discussed. The architectures of the SN, MN, CN and clusters to illustrate the working of TM-IoT.

The next chapter presents Clustering-driven intelligent, scalable and reliable trust management for IoT (CITM-IoT) which is based on TM-IoT. I have simulated our proposed TM-IoT trust management solution in Chapter 6 and Chapter 7 to evaluate its working under different parameters.

CHAPTER 6

CLUSTERING-DRIVEN INTELLIGENT, SCALABLE AND RELIABLE TRUST MANAGEMENT FOR IOT (CITM-IOT)

6.1 INTRODUCTION

As discussed in the previous chapters, a proposed trust management solution for the IoT environment should be scalable. Furthermore, a proposed trust management solution for the IoT environment should take into account the memory constraints of the IoT node. Finally, a trust management system should be resistant to attacks such as bad-mouthing attacks.

One possible approach to achieving IoT security is to enable a trustworthy IoT environment wherein the interactions are based on the trust value of the communicating nodes. Trust management and trust assessment has been extensively studied in distributed networks in general and the IoT in particular, but there are still outstanding pressing issues such as the bad-mouthing of trust values, addressing the memory constraints of IoT nodes etc. which prevent them from being used in practical IoT

applications. Furthermore, no research has been conducted to ensure that the developed IoT trust solutions are scalable across billions of IoT nodes.

As discussed in Chapter 2, none of the existing research addresses the significant gaps in the investigation of bad-mouthing attacks, IoT node memory constraints, and the scalability of IoT nodes.

To address these issues, this chapter proposes a methodology for a scalable trust management solution in the IoT. The methodology addresses practical and pressing issues related to IoT trust management such as trust-based IoT clustering, intelligent methods for countering bad-mouthing attacks on trust systems, issues of memory-efficient trust computation and the trust-based migration of IoT nodes from one cluster to another (to enable the scalability of the trust management solution). The experiment results demonstrate the effectiveness of the proposed approaches.

This chapter is organised as follows: Section 6.2 proposes the overall architecture of the scalable trust management approach of IoT (TM-IoT). Section 6.3 proposes four intelligent algorithms (to address the aforementioned gaps) on top of TM-IoT. These intelligent solutions are as follows:

1. Section 6.3.1 proposes Algorithm 6.1 which is a new mechanism of clustering (in TM-IoT) by calculating trust value boundaries for each cluster according to memory boundaries.
2. Section 6.3.2 proposes Algorithm 6.2 which defines the conditions in which a cluster node is able to change to a specified new master node in TM-IoT.
3. Section 6.3.3 proposes Algorithm 6.3 which is used to address the issue of the bad-mouthing of IoT nodes. It does so by eliminating the bad-mouthed values (outliers) of a set of floats using Tukey's fences. This algorithm is a proposed solution for extreme bad-mouthing attacks in TM-IoT.
4. Section 6.3.4 proposes Algorithm 6.4 which is a method by which master nodes monitor cluster node trust values and try to move some cluster nodes away. A

check is then made on the memory of the current master node to decide whether further cluster nodes need to be removed.

Sections 6.4 documents the experimentation and the results for the proposed TM-IoT algorithms and compares the results with other state-of-the-art approaches along various benchmarks. Section 6.5 concludes the chapter.

This chapter contains sections that have been earlier published by us in (Alshehri, Hussain & Hussain 2018).

6.2 Architecture of the approach for scalable trust management in IoT (TM-IoT)

In this section, I present the architectural overview of the scalable trust management solution (TM-IoT). Subsequently, in Section 6.3, I present the algorithmic working to address the key issues related to scalable trust solutions which are identified in Chapter 3.

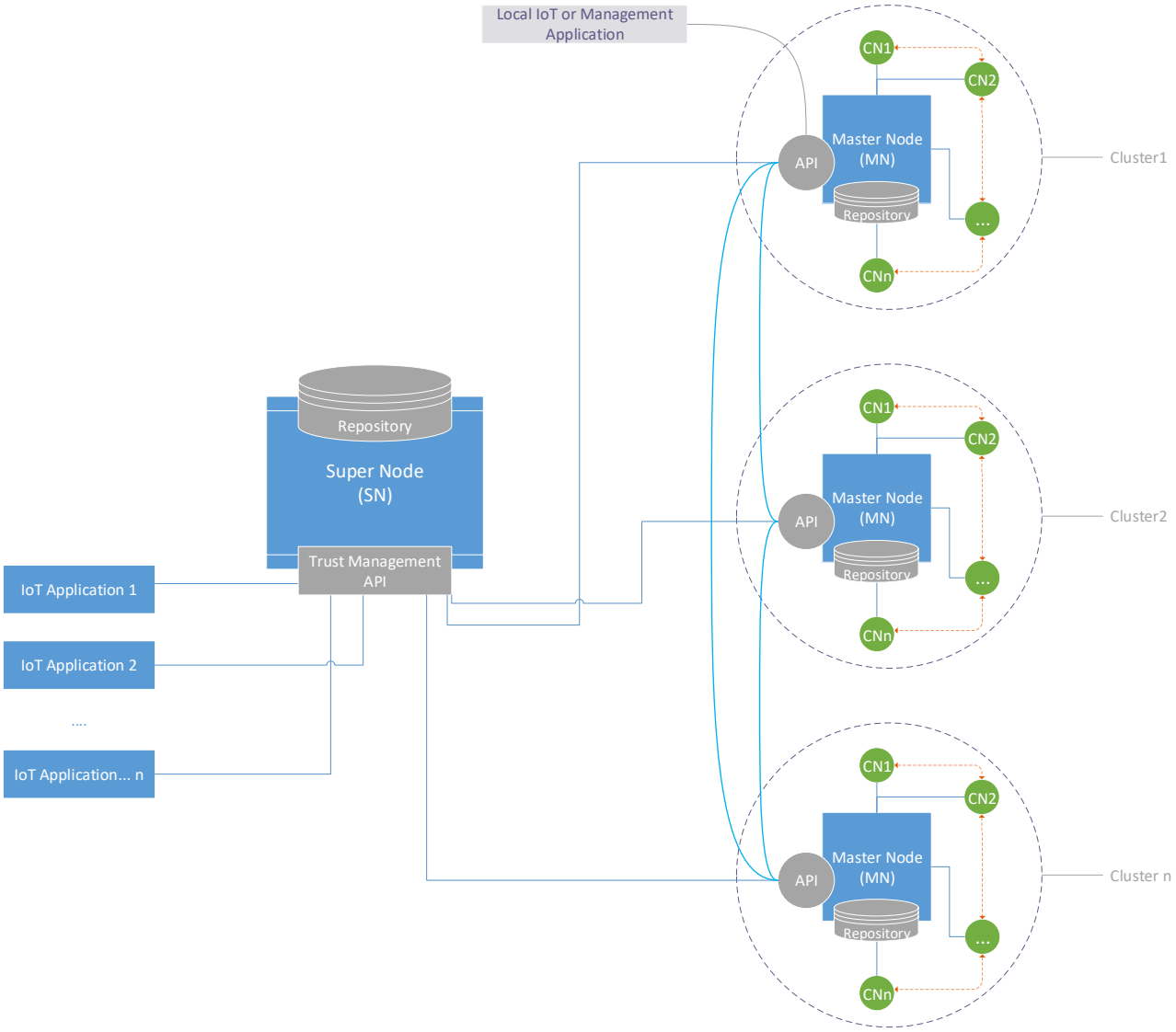

Figure 6-1 Architecture of the TM-IoT

The TM-IoT provides a trustworthy platform for communication between all the devices that communicate with other nodes in the IoT environment. In TM-IoT, the master node (MN) stores the trust values of all the nodes within its cluster. The super node (SN), in turn, stores the trust value of all the MNs. The TM-IoT architecture allows heterogeneous IoT devices and applications to contact each other in trusted heterogenic-device communication.

The TM-IoT architecture (Figure 6-1) is a distributed architecture that consists of *CN*, *MN*, *Clusters* and a *SN*. The TM-IoT architecture consists of a Cluster Node (CN), which resides on a node, and a Master Node (MN). The CNs interact with each other and assign a trust value based on the outcome of the interaction. The updated trust values are sent and stored by the Master Node (MN) The MN manages many CNs in the cluster, and additionally stores the received data sent by CNs in the MN memory. In the TM-IoT distributed architecture illustrated in Figure 6-1, the SN is the main node in the TM-IoT. It is responsible for all IoT-trusted environments and contains an API, referred to as the trust management API. The API allows the SN to communicate with many MNs in the clusters. The SN also has a repository that stores the trust value and addresses of each MN and CN. The SN repository is hierarchical (tree-structured), and each entry relevant to a CN is addressed through the MN's unique ID. Thus the trust data in the SN repository is collected from the MNs and not from the CNs. An IoT application running or making use of the SN can provide it's services based on the data from the various CNs. Therefore, IoT applications and services are built on top of the proposed TM-IoT by supporting communications between nodes via the SN.

The TM-IoT architecture provides a centralized model of several clusters and a MN, which allows for trust management of things over a local area network. The TM-IoT distributed architecture of several MNs and clusters creates a trust distributed system where CNs communicate with each other and MNs communicate with the CNs in their cluster and the SN in a cooperative manner. This architectural flexibility is specifically designed for the communication requirements of the IoT, given that most IoT devices may play different roles in both centralized and distributed operations setups, especially for trust management in the IoT. Additionally, it is this architectural flexibility that allows TM-IoT to scale to a large number of nodes.

6.3 INTELLIGENT ALGORITHMS FOR TM-IOT SCALABILITY AND RELIABILITY

To ensure that our proposed architectural approach for trust management is scalable and reliable, I propose the following four algorithms:

a) Algorithm 6.1 proposes a new mechanism of clustering (in TM-IoT) by calculating trust value boundaries for each cluster according to memory boundaries (Section 6.3.1).
b) Algorithm 6.2 defines the conditions in which a cluster node is able to change to a specified new master node in TM-IoT (Section 6.3.2).
c) Algorithm 6.3 is used to address the issue of the bad-mouthing of IoT nodes. It does so by eliminating the bad-mouthed values (outliers) of a set of floats using Tukey's fences. This algorithm is a proposed solution for extreme bad-mouthing attacks in TM-IoT (Section 6.3).
d) Algorithm 6.4 proposes methods by which master nodes monitor cluster node trust values and try to move some cluster nodes away. Further, after CNs interact with each other in each cluster, some CNs will receive a high or low trust value after which the MN will decide to move the CNs to the proper cluster which is equal to the new trust value obtained by the CNs. A check will then be made on the memory of the current master node to decide whether further cluster nodes need to be removed (Section 6.3.4).

In the following sub-sections, I present the workings of each of these four algorithms.

6.3.1 ALGORITHM FOR TRUST-BASED CLUSTER BOUNDARY CALCULATION

This algorithm is run by the SN to allocate trust value thresholds and memory thresholds for all MNs. The trust values of all MNs are collected from the other MNs. An average of these trust values is calculated and input into a set *trustMn*. The total memory *Mt* is then retrieved from the trust values from all MNs with memory permanently used by service (*Ms)*. *Ms* is calculated on the MNs by multiplying *Mt* with a global predefined service rate (*Rs*) to simulate a heavy use of memory.

$$Ms = Mt \times Rs \tag{6.1}$$

The total memory available (*Ma)* is calculated by subtracting *Ms* from *Mt*. *Ma* will be a part of set *Sma*.

$$Ma = Mt - Ms \tag{6.2}$$

Based on *Ma*, the maximum number of nodes connected to this master node *MNm* is calculated using Equation 6.3. The result is placed into the set *Smnm.*

$$MNm = \frac{Ma}{Me} \tag{6.3}$$

Me is a constant defining one set of connection information. It consists of the IP address and trust value of a cluster node. Next, *Sma* and *Smnm* are sorted according to *trustMn* from smallest to largest. In the latter part of this algorithm, this sorting ensures the consistency of trust values between the connected MNs and CNs. The next step is to compute the memory rates from *Nmc* and the sum of *Smnm* elements *Nmc* is the total

number of cluster nodes possible in the system. The memory minimum bound rate *Rmm* and memory upper bound rate *Rum* is calculated using the following equations.

$$Rmm = \frac{numCn}{Nmc} \tag{6.4}$$

$$Rum = \frac{Rmm + (1 - Rmm)}{256} \tag{6.5}$$

numCn is the total number of cluster nodes in the system. The intention of *Rum* is to give every MN a small amount of memory to allow some memory for cluster node movement between MNs. The number 256 is selected to provide a suitable memory allowance. The higher this number is, the lower the allowance. A high allowance leads to an unbalanced distribution of cluster nodes to master nodes. A low allowance results in a high level of restriction on cluster node movement. The set of minimum memory boundary and the set of upper memory boundary are computed using equations 6.9 and 6.10.

$$Smm = Sma \times Rmm \tag{6.6}$$

$$Sum = Sma \times Rum \tag{6.7}$$

All CN trust values stored on MNs are subsequently gathered and sorted from lowest to highest into a set *Stcn*. A temporary 2-dimensional set *tempTrust* is created. Trust

values in *Stcn* are considered from the middle to the boundaries and added to *tempTrust*. Values are first added to the middle of *tempTrust* and then towards the boundaries. During the process of adding a temporary memory, usage is calculated as follows:

$$Tmemory = (tempTrust[i].size + 1) \times Me \quad (6.8)$$

If *Tmemory* exceeds *Smm[i]*, it adds the value to the next set of *tempTrust*. Lastly, the trust upper bound *Tupper* and trust lower bound *Tlower* is computed. When calculating *Tupper* for *tempTrust[i], tempTrust[i+j]* is considered. It continues to look for the next *tempTrust* until the highest number of *tempTrust[i]* is not equal to the highest number of *tempTrust[i+j]*. *Tupper* is then calculated by equation 6.9.

$$Tupper = \frac{[max(tempTrust[i]) + min(tempTrust[i+j])]}{2} \quad (6.9)$$

Tlower is calculated in a similar way using *tempTrust[i-j]*. It continues to look for the next *tempTrust* until the lowest number of *tempTrust[i]* is not equal to the highest number of *tempTrust[i-j]*. *Tlower* is then calculated using equation 6.10.

$$Tlower = \frac{[min(tempTrust[i]) + max(tempTrust[i+j])]}{2} \quad (6.10)$$

Lastly, all *Smm, Sum, Tupper* and *Tlower* are assigned to the consistent MNs. The working of the algorithm for computing the trust-based thresholds and memory threshold for each cluster is presented in Algorithm 6.1.

Algorithm 6.1 Allocating trust value thresholds and memory thresholds for master nodes

Require: Number of master nodes as numMn, trust values of Master nodes as a set trustMn, Number of cluster nodes as numCn, trust values of cluster nodes as a set Stcn
for all master nodes **do**
get trust values of master node from other master nodes into set Strust
Algorithm 3 (Strust)
if Strust is not empty **then**
Add average of trust into set trustMn
else
Add 0.0 to set trustMn
end if
end for
for all master nodes **do**
$Ma \leftarrow Mt - Ms$
Add Ma to set Sma
$Mnm \leftarrow Ma \div Me$
Add Mnm to set Smnm
end for
Sort Sma, Smnm, according to trustMn from lowest to largest
$Nmc \leftarrow sumofSmnmelements$
$Rmm \leftarrow numCn \div Nmc$
$Rum \leftarrow Rmm + (1 - Rmm) \div 256$
$setSmm \leftarrow Sma \times Rmm$
$setSum \leftarrow Sma \times Rum$
Sort Stcn from lowest to largest
for all master nodes from the middle to the boundary in the sequence of trustMn **do**
repeat
add current float into the set tempTrust[i]
until $(tempTrust[i].size + 1) \times Me > Smm[i]$
end for
for all float sets in set tempTrust **do**
$max \leftarrow$ largest number in tempTrust[i]
$min \leftarrow$ smallest number in tempTrust[i]
j = 0
repeat
$maxj \leftarrow$ largest number in tempTrust[i+j]
$minj \leftarrow$ smallest number in tempTrust[i+j]
j++

```
        until max != maxj
        if i + j + 1 is the size of tempTrust then
            Add max to set STupper
        else
            Add average of max and minj to set STupper
        end if
        j = 0
        repeat
            maxj ← largest number in tempTrust[i-j]
            minj ← smallest number in tempTrust[i-j]
            j–
        until min != minj
        if i - j == 0 then
            Add min to a set STlower
        else
            Add average of maxj and min to STlower
        end if
    end for
    for all master nodes do
        Assign trust upper bound from STupper
        Assign trust lower bound from STlower
        Assign memory minimal bound from Smm
        Assign memory upper bound from Sum
    end for
```

6.3.2 ALGORITHM FOR TRUST-DRIVEN NODE MIGRATION FROM ONE CLUSTER TO ANOTHER TO ENABLE TM-IOT SCALABILITY

In our proposed framework (TM-IoT), scalability is achieved by clustering the IoT nodes into groups or clusters based on their trust values. An IoT node MN takes care of the trust management process of each CN within a given cluster. When the trust value of a given CN changes, the MN uses Algorithm 6.2 to transfer the IoT node to a different cluster. Algorithm 6.2 is used to accept a request to transfer a CN between clusters. It gathers the CN trust values from cluster peers of the current cluster and cluster peers of the target cluster into a list *trustValues*. Algorithm 6.3 is run for

trustValues to eliminate outliers (bad mouthing nodes), and *Tv* is calculated as the average of *trustValues*. If *Tv* is in the trust bound of the target MN and adding CN will not exceed the memory threshold of the target MN, the CN and its trust value record are removed from the current MN and transferred to the target MN. Consequently, the capability of moving CNs from one cluster to another based on their trust value results in a trust-based environment. This subsequently could be a key contributor to the collaboration and further growth in the number of nodes joining the network.. The details of our experiments are in Section 6.5

The working of the algorithm for trust-driven node migration from one cluster to another is presented in Algorithm 6.2.

Algorithm 6.2 Request to switch cluster

```
Require: cluster node CN, current master node MN, a new
  master node NMN, trust boundaries of NMN, memory used
  by NMN to store cluster node information as Mi, memory
  threshod as memThres
  for all cluster peers of CN do
      Send trust value of CN to the list trustValues on MN
  end for
  for all cluster nodes belongs to NMN do
      Send trust value of CN to the list trustValues on MN
  forward by NMN
  end for
  Algorithm 3 (trustValues)
  Tv ← average of values in trustValues
  if Tv not within the trust boundaries of NMN then
      return false
  else
      if Mi + Me <memThres then
          remove CN from its current cluster
          remove CN's trust value from MN's trust list
          add CN to the cluster of NMN
          add Tv as trust value of CN to NMN's trust list
          return true
      else
          return false
      end if
  end if
```

6.3.3 ALGORITHM TO ELIMINATE OUTLIERS FOR BAD-MOUTHING ATTACKS IN THE IOT

The first step in the algorithm to eliminate outliers (Algorithm 6.3) is to break the set of input float values from lowest to largest into two parts. Q1 is the medium of the first part and Q3 is the medium of the second part in the interquartile range. The interquartile range is expressed as follows:

$$Qrange = Q_3 - Q_1 \tag{6.11}$$

Then the upper threshold of the set is computed

$$Tupper = Q_3 + 1.5 \times Qrange \tag{6.12}$$

The lower threshold of the set is computed in a similar way

$$Tlower = Q_1 - 1.5 \times Qrange \tag{6.13}$$

Finally, all floats outside of the ranges *Tupper* and *Tlower* are eliminated.

The working of the algorithm for removing bad-mouthed values is presented in Algorithm 6.3. The experiment results of our proposed approach for removing bad-mouthed values are in Section 6.5.

Algorithm 6.3 Eliminate outliers

Require: A set of floats as Sinput
Sort Sinput from lowest to highest
if Sinput.size is event **then**
Slower ← first half set of Sinput
Supper ← second half set of Sinput
else
Slower ← first half set of Sinput eliminating the middle number
Supper ← second half set of Sinput eliminating the middle number
end if
if Slower.size is even **then**
Q1 ← average of the middle two numbers
else
Q1 ← the middle value
end if
if Supperr.size is even **then**
Q3 ← average of the middle two numbers
else
Q3 ← the middle value
end if
$Qrange \leftarrow Q3 - Q1$
$Thupper \leftarrow Q3 + 1.5 \times Qrange$
$Thlower \leftarrow Q1 - 1.5 \times Qrange$
for all float E in Sinput **do**
if E >Thupper or E <Thlower **then**
remove E from Sinput
end if
end for
return Sinput

6.3.4 ALGORITHM FOR UPDATING AND CHECKING THE TRUST VALUES OF CLUSTER AND MASTER NODES

The first step in the algorithm (Algorithm 6.4) is to update and check the trust values of the CNs of MNs. The memory usage of MNs is also checked. If the trust value or memory exceeds the pre-defined trust boundaries or thresholds, the MN uses Algorithm 6.2 to determine whether it can move certain CNs. First, the trust values of each CN are collected from the peer cluster nodes. The outliers are eliminated by Algorithm 6.3 and *Tv* is calculated as an average. The trust value of each CN is updated to *Tv* and stored on the MN. If *Tv* is not in the range of *Tupper* and *Tlower* of the current MN, it is added to the list *trustOut*. For all CNs in *trustOut*, a list of MNs where *Tv* of the CN is in range of the trust bounds of the MNs is computed. This list of MNs will be *trustMNodes*. Algorithm 6.1 is run for all MNs in *trustMNodes* with the minimum memory bound of MN. If CN is removed, the algorithm is terminated. If all MNs of *trustMNodes* are considered for Algorithm 6.2 and CN is not removed from the current master node, Algorithm 6.2 is run again for all MNs in *trustMNodes* with the upper memory bound for MNs.

The second part of Algorithm 6.4 is triggered when the memory of the current MN is over the upper memory bound. In this case, the number of nodes that are required to be removed are computed. The current memory usage is required to calculate the value. This value does not consider the Service memory (memory used by the nodes for any kind of processing) and Ma is the current memory available.

$$Mi = Mt - Ms - Ma \tag{6.14}$$

The number of CNs that need to be moved will be *Num* and are calculated with the upper memory boundary *mum*.

$$Num = \frac{(Mi - mum)}{Me} \tag{6.15}$$

Algorithm 6.2 is then run for all CNs targeting all other MNs using their minimal memory bound until the *Num* of CNs is removed. If the *Num* of CNs is not removed following this process, Algorithm 6.2 is run again for all CNs, targeting all other MNs using their upper memory bound until the number of CNs that have been removed reaches *Num*. Consequently, this solution ensures the IoT platform is scalable and efficient, based on the current proposed approach. The working of the algorithm for updating and checking the trust values of the cluster and MNs is presented in Algorithm 6.4. This algorithm can be used to identity nodes who are delivering low quality services or bad service to other IoT nodes.

Algorithm 6.4 Check current cluster node status

Require: a master node MN, memory used by MN to store cluster node information as Mi, its cluster nodes CNs, trust boundaries of MN, a set of other master nodes MNs
for all CN in CNs **do**
if a trust value of CN exists on MN **then**
add trust value of CN stored on MN to set Stcn
end if
for all other CN in CNs **do**
add trust value of CN to set Stcn
end for
Algorithm 3 (Stcn)
$Tv \leftarrow$ average of values in Stcn
update trust value of CN on MN to Tv
if T **then**v is not within trust boundaries of MN
add CN to list trustOut
end if
end for
for all CN in trustOut **do**
Tcn is trust value of CN
for all Mn if other MNs **do**
$Tupper \leftarrow$ trust upper bound of MN
$Tlower \leftarrow$ trust lower bound of MN
if Tlower <Tcn <Tupper **then**
Add MN to list trustMNodes
end if
end for
for all MN in trustMNodes **do**
if Algorithm 2 (CN, current master node of CN, MN, trust boundaries of MN, memory used to store cluster node information in MN, minimum memory bound of MN) **then**
break
end if
end for
if CN not switched to other cluster **then**
if Algorithm 2 (CN, current master node of CN, MN, trust boundaries of MN, memory used to store cluster node information in MN, upper memory bound of MN) **then**
break
end if
end if
end for

```
if Mi >upper memory bound of current master node MN
mum then
    number of nodes needed to be removed Num ← (Mi −
mum) ÷ Me
end if
for all CN in current MN's cluster do
    if Num ≤ 0 then
        return
    end if
    for all MN in other MNs do
        if Algorithm 2 (CN, current master node of CN,
MN, trust boundaries of MN, memory used to store cluster
node information in MN, minimum memory bound of MN)
then
            Num−
            break
        end if
    end for
end for
for all CN in current MN's cluster do
    if Num ≤ 0 then
        return
    end if
    for all MN in other MNs do
        if Algorithm 2 (CN, current master node of CN,
MN, trust boundaries of MN, memory used to store cluster
node information in MN, minimum memory bound of MN)
then
            Num−
            break
        end if
    end for
end for
```

6.4 Experimentation and Results for the proposed scalability and reliability algorithms of TM-IoT

In this section, I experimentally evaluate the working of the proposed TM-IoT algorithms in Section 6.4. The parameters of the simulation set-up and testing are as follows:

Table 6-1 Base case simulation parameters

Parameter	Value
Number of nodes	343
Number of clusters	9
Simulation time	200s
Master/Cluster node memory	32 bytes
Memory rate	0.5
Trigger Outliers	True
Trigger Bad Mouths	False
Trigger Memory Thresholds	True
Trigger Balanced Node Distribution	False

Table 6-1 gives the simulation parameters in the base case. I define the *Memory Rate* as the percentage of total memory used on other services. In our simulation, this chunk

of memory is occupied permanently. I define the *Trigger Outliers triggers* (used in Algorithm 6.3) as a trigger that computes the trust values in all the nodes in the network. I define *Trigger Bad Mouths* as a trigger that initiates an extreme bad-mouthing, as a result of which some nodes give an extremely low trust value number for other nodes. I define *Trigger Memory Threshold* as a trigger that compares the current memory being used with the memory bounds used. This is to eliminate outliers for bad-mouthing trust attacks in IoT in Algorithm 6.3. Lastly, I define *Trigger Balanced Node Distributed* as a measure that captures the even distribution of CNs across MNs. If the *Trigger Balanced Node Distribution* is true, the cluster nodes are distributed evenly between the master nodes. If it is false, the cluster nodes are distributed to only two master nodes at the start of a simulation, otherwise it is evenly distributed across all the master nodes in the simulation.

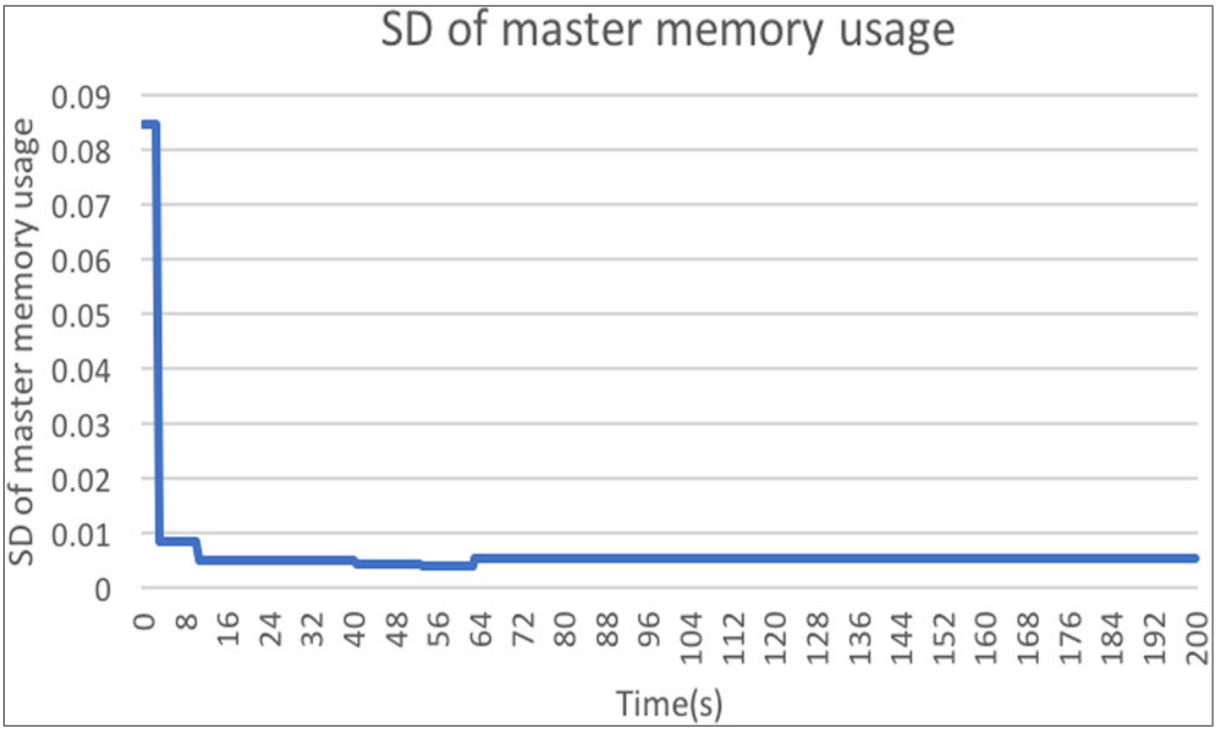

Figure 6-2 Base case SD of master memory usage

Figure 6-2 uses standard deviation (SD) to demonstrate the difference in memory usage on the MNs. Memory usage on every MN (for trust computations) based on memory without service occupied memory is computed by equation 6.16.

$$Musa = \frac{1 - Mavailable}{(Mtotal - Mservice)} \tag{6.16}$$

Standard deviation σ is then calculated using the mean μ of N number of $Musa$ in a set $SMusa$ by equation 6.17.

$$\sigma = \sqrt{\frac{1}{N}\sum(SMusa - \mu)^2} \tag{6.17}$$

A low σ means there is an even distribution of memory is being used for all MNs for trust computation. This demonstrates the effectiveness of TM-IoT along two dimensions as follows: (a) the ability of MNs to carry out trust computations without dedicating large amounts of resources such as memory to it; and (b) the ability to intelligently and automatically group CNs with similar values into a cluster.

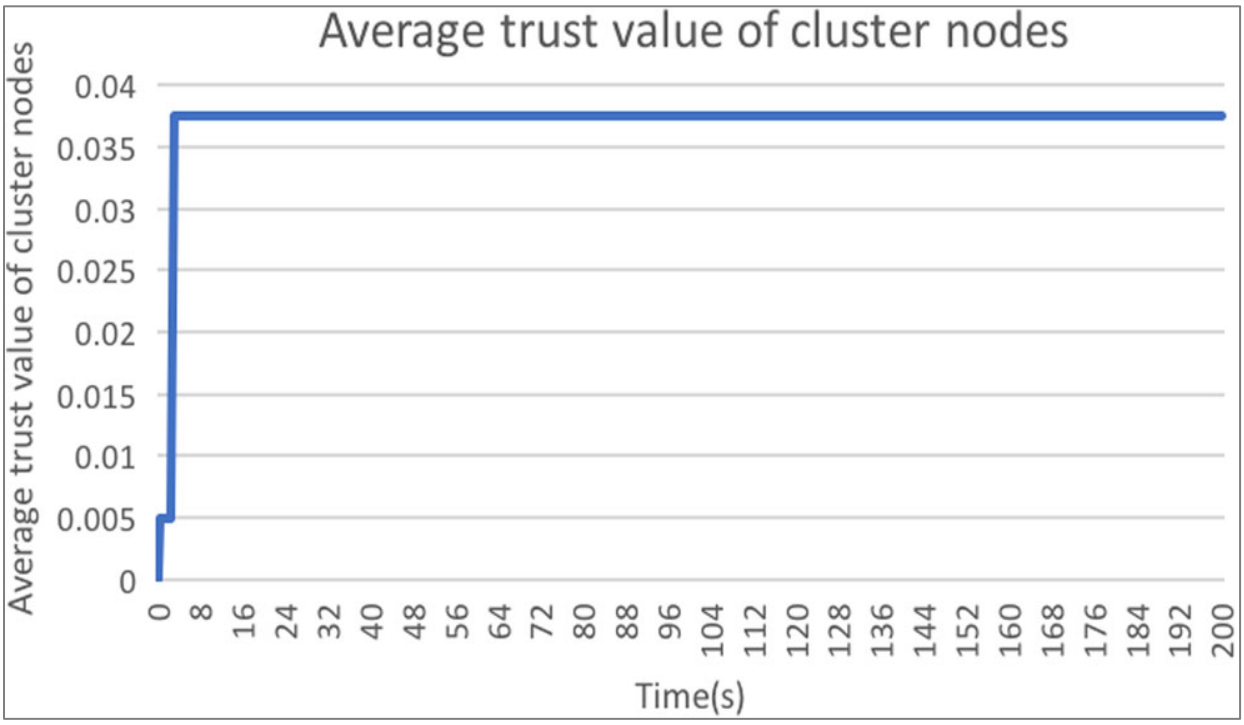

Figure 6-3 Base case average trust value of cluster nodes

The average of values from Figure 6-3 is computed by the trust values of all CNs stored on MNs using equation 6.18. *SclusterTrust* is a list of these trust values.

$$\mu = \frac{1}{N}\sum SclusterTrust \tag{6.18}$$

This measure (*SclusterT rust*) is used to capture the effectiveness of the algorithm in eliminating the effects of bad-mouthing attacks.

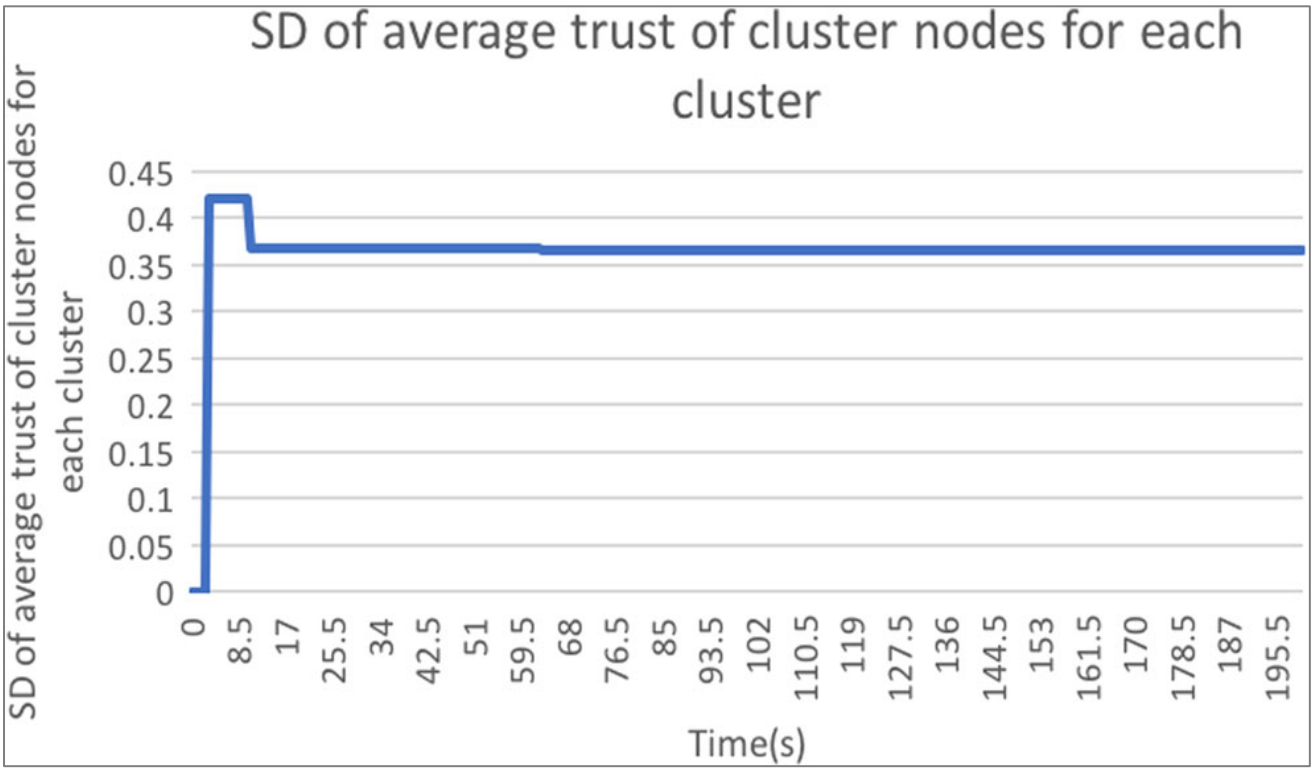

Figure 6-4 Base case SD of average trust of cluster nodes for each cluster

Figure 6-4 shows the difference between the average cluster trust values from each cluster. During the initial part of the simulation the difference is high and as the simulation progresses the average trust value of all the clusters converge. This shows the effectiveness of TM-IoT in intelligent and automatically clustering IoT with similar trust values.

The average trust value of CNs from each cluster μ will be calculated by *SclusterTrust* as a set of all trust values and put into a list $S\mu$ by equation 6.19.

$$s_u = \frac{1}{N}\sum SclusterTrust \tag{6.19}$$

An average of $S\mu$ will be computed by equation 6.20.

$$\mu s = \frac{1}{N}\sum s_u \tag{6.20}$$

Lastly, the standard deviation will be calculated using equation 6.21.

$$\sigma = \sqrt{\frac{1}{N}\sum (s_u - \mu s)^2} \tag{6.21}$$

6.4.1 RESULTS OF METHOD PROPOSED FOR COUNTERING BAD-MOUTHING ATTACKS IN TM-IOT

Table 6-2 shows the case simulation values of parameters for bad-mouthing attacks.

Table 6-2 Bad-mouthing attack case simulation parameters

Parameter	Value
Number of nodes	343
Number of clusters	9
Simulation time	200s
Master/Cluster node memory	32 bytes
Memory rate	0.5

Trigger Outliers	Switched
Trigger Bad Mouths	True
Trigger Memory Thresholds	True
Trigger Balanced Node Distribution	False

The purpose of this simulation is to set up and simulate an extreme bad-mouthing attack environment and demonstrate the effectiveness of determining statistical outliers (bad-mouthed trust values) using Algorithm 6.3. To determine the effectiveness of our proposed method for countering bad-mouthing attacks, I carry out the simulation with and without Algorithm 6.3. The blue line in Figures 6-5, 6-6, and 6-7 shows the results with the proposed bad-mouthing algorithm, and the orange line shows the results without our proposed bad-mouthing algorithm.

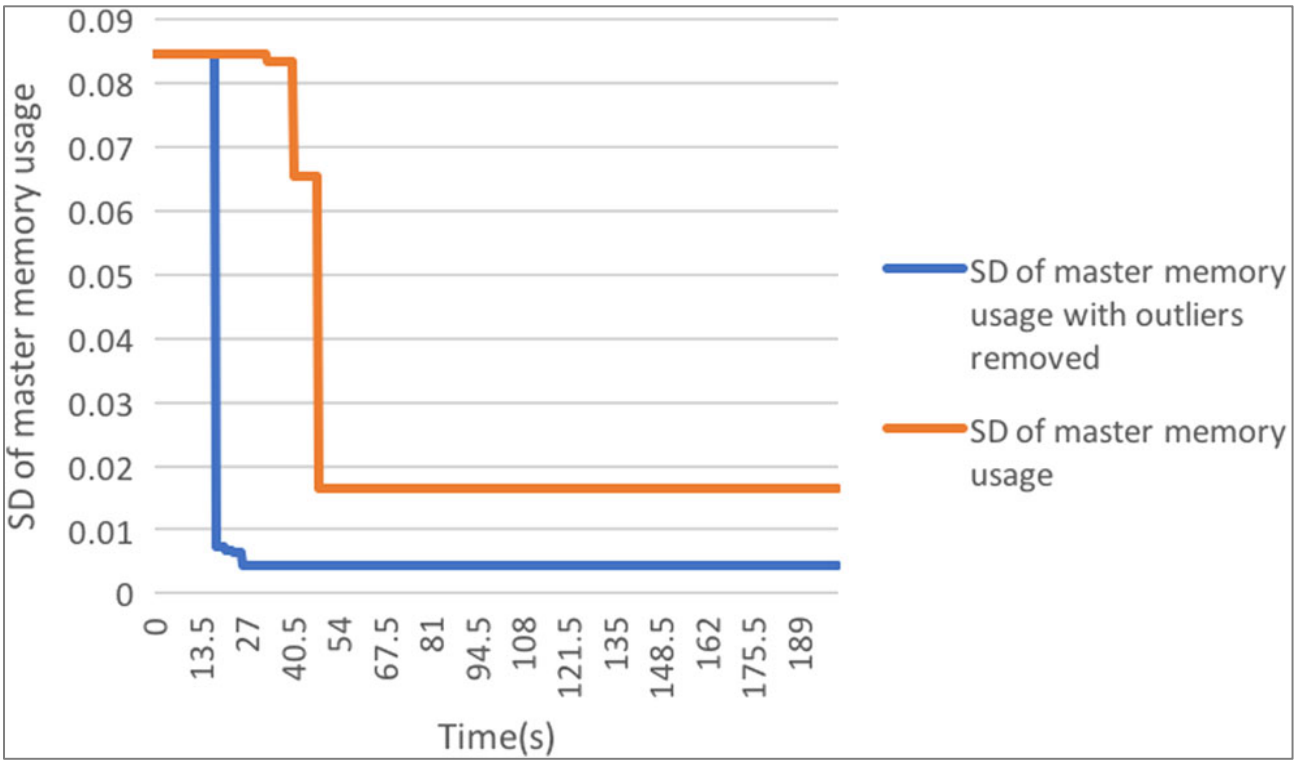

Figure 6-5 Comparison of SD of master memory usage with and without outliers (proposed bad-mouthing algorithm – Algorithm 6.3)

Figure 6-5 illustrates that without the outlier mechanisms, there is a higher SD of master memory usage. A higher SD means there is an unbalanced distribution of CNs to MNs based on master node memory. This comparison shows that in the case of extreme bad-mouthing attacks, the outlier mechanism ensures the balanced distribution of memory on the MNs.

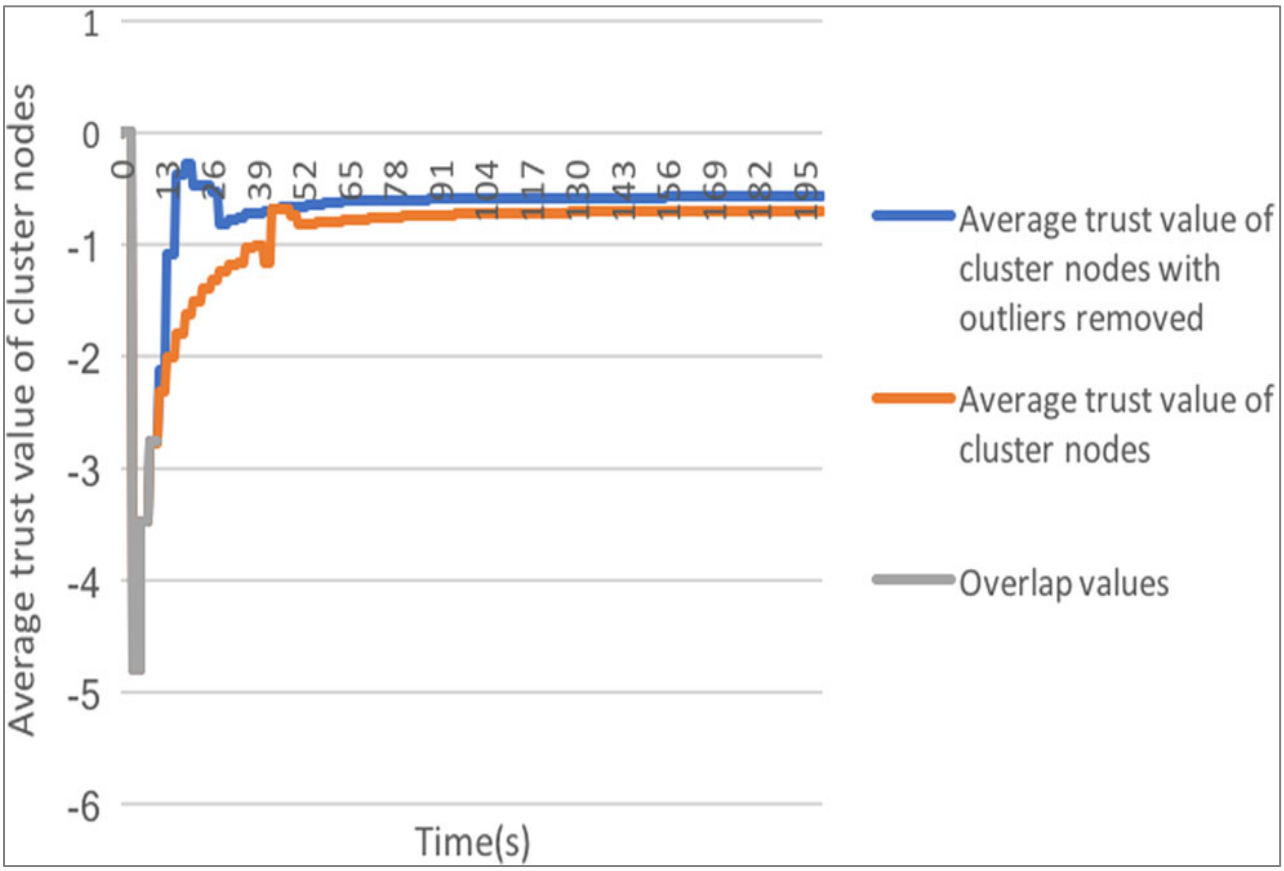

Figure 6-6 Comparison of average trust values of CNs with and without outliers

Figure 6-6 demonstrates that the outlier mechanism is able to filter out some of the bad-mouthing attack values, resulting in reliable trust values being available for each CNs for trust computation.

These simulation results also reveal a downside of the outlier mechanism, namely that a larger set of data is required for this mechanism to be effective. In the earlier stages of the simulation, fewer trust values accumulate, reducing the effect of the outlier mechanism. As the trust values accumulate, the outlier mechanism becomes more effective and can filter out bad-mouthed values.

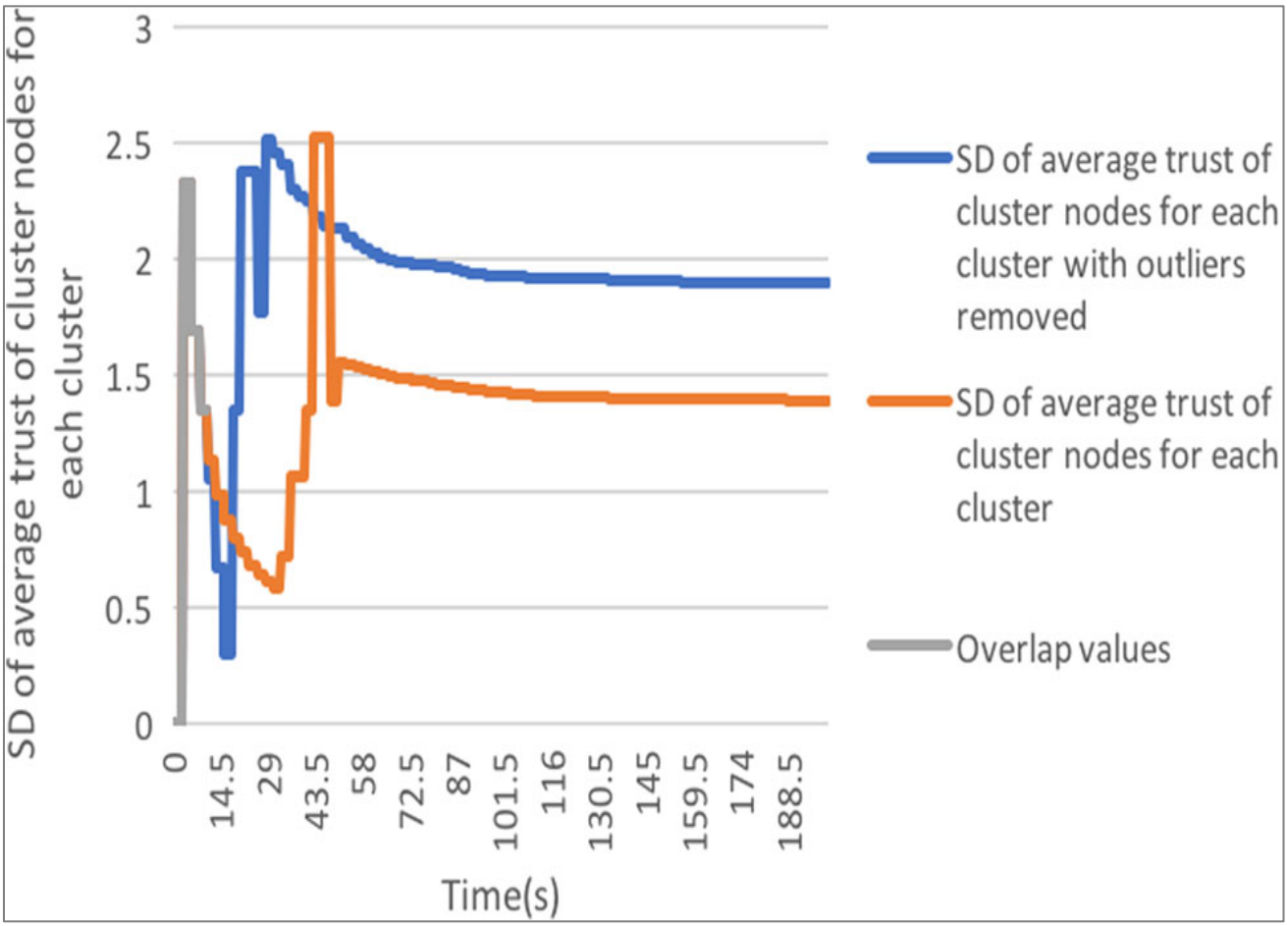

Figure 6-7 Comparison of SD of average trust of CNs for each cluster with and without outliers

Figure 6-7 shows that clustering is more effective with the outlier mechanism, based on the trust values. There is a difference of around 46 seconds in simulation time between our approach to detect bad-mouthing attacks and without badmouthing. With our approach, the simulation time is shorter by 46 seconds. This is the result of the uneven distribution of CNs. At the start of this simulation, CNs are distributed only to two MNs. After several interactions between the nodes, some clusters have extremely low trust values and the other empty clusters have 0 or the initial value of trust. In the

latter stage of the simulation, the nodes are evenly distributed based on their trust value, creating a smaller difference in average trust values between clusters.

6.4.2 RESULTS OF METHOD PROPOSED FOR COUNTERING EXTREME MEMORY

The parameters of the simulation approach used for countering extreme memory are shown in Table 6-3.

Table 6-3 Extreme memory simulation parameters

Parameter	Value
Number of nodes	343
Number of clusters	9
Simulation time	200s
Master/Cluster node memory	32 bytes
Memory rate	0.9775
Trigger Outliers	True
Trigger Bad Mouths	False
Trigger Memory Thresholds	Switched
Trigger Balanced Node Distribution	True

As shown in Table 6-3, compared to the base case (Table 6-1), the memory rate is switched from 0.5 to 0.9775, creating an extremely low memory condition. By using

our proposed algorithm instead of distributing the CNs to only two MNs, the CNs are distributed equally to all MNs. The memory thresholds are switched 'on' and 'off' for comparison. When the memory threshold is switched 'on', the memory boundary mechanisms will be turn off.

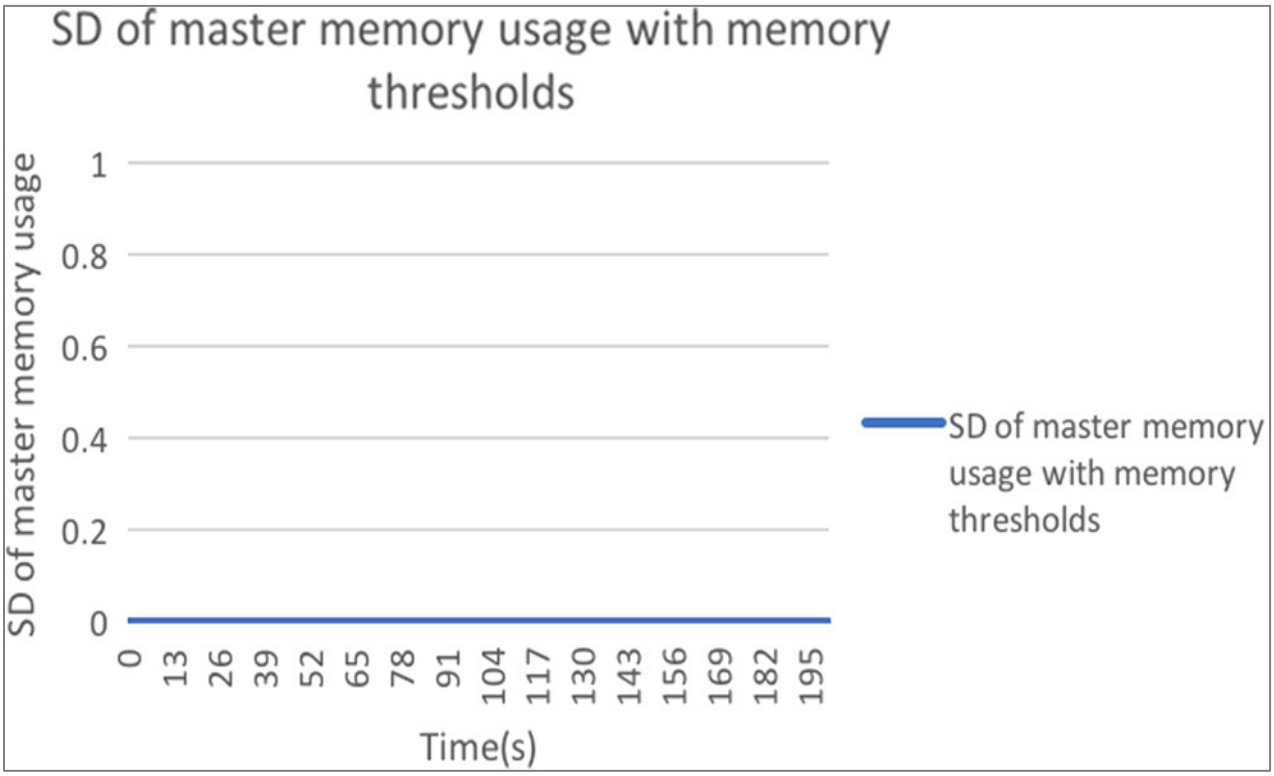

Figure 6-8 SD of master memory usage with memory thresholds (memory threshold 'on')

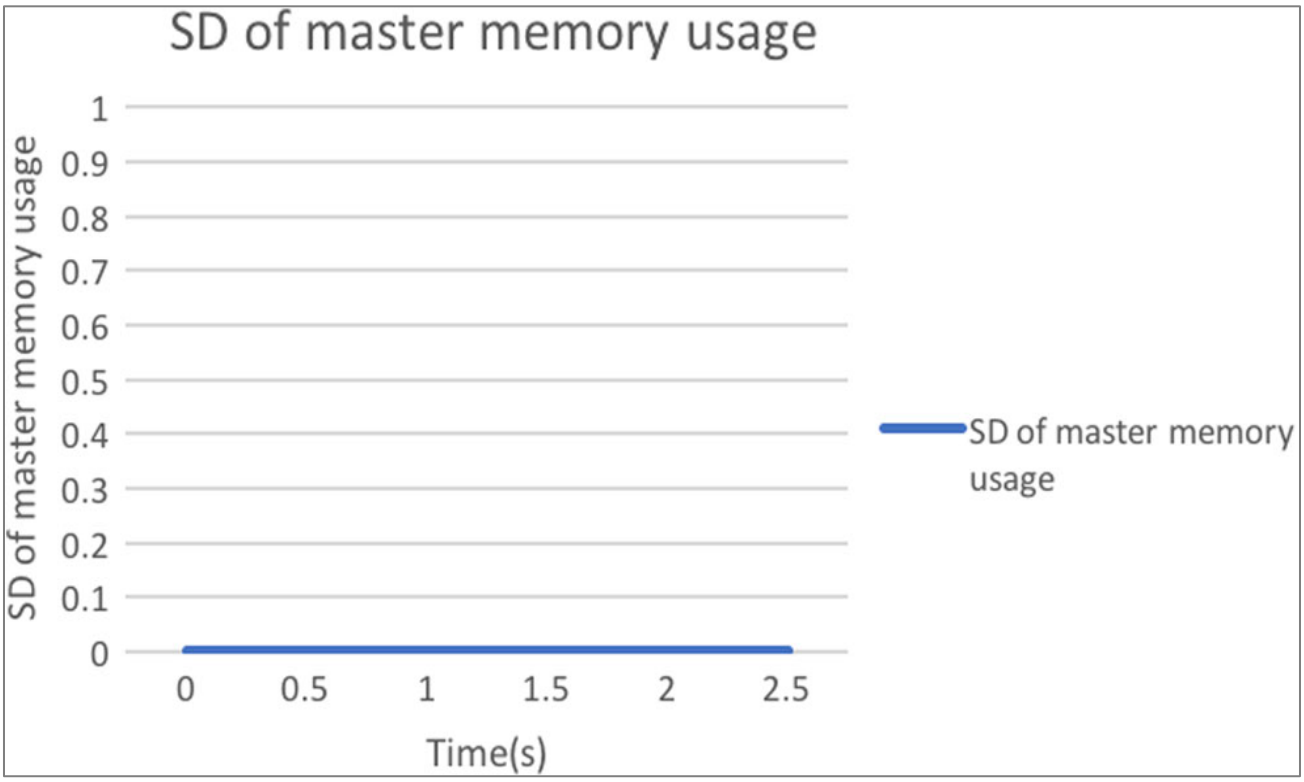

Figure 6-9 SD of master memory usage without memory thresholds (memory threshold 'off')

Figure 6-8 and Figure 6-9 respectively show the time it takes for the simulation to execute with and without the memory threshold. As can be seen from these two figures, the simulation successfully executes for much longer (195.5 seconds) with the memory threshold than without the memory threshold (2.5 seconds). As expected, the SD of master memory usage is 0 due to an even distribution of cluster nodes at the start. Figure 6-8 shows that, with memory thresholds, the simulation is able to run fully. Figure 6-9 illustrates that without memory thresholds, the simulation terminates due to the master node running out of memory. This difference between simulation times demonstrates the usefulness of memory thresholds under extreme memory conditions.

6.6 CONCLUSION

The chapter opened with the elaboration of the gaps related to trust management for IoT. The main concepts used in designing and evaluating the framework were defined. The proposed TM-IoT platform was described and four intelligent algorithms were proposed to ensure the scalability and reliability of the proposed TM-IoT trust management framework. The proposed solutions are: (i) intelligent approaches for countering bad-mouthing attacks in TM-IoT; (ii) intelligent approaches for the trust-based migration of IoT nodes from one cluster to another; (iii) intelligent approaches to determine the trust boundaries of the each cluster; and (iv) intelligent approaches by which master nodes monitor cluster node trust values and try to move some cluster nodes to ensure the scalability of IoT.
Furthermore, in order to evaluate our proposed algorithms in this chapter, I engineered a prototype and carried out the testing of the prototype. The obtained results demonstrate the applicability and good performance of our proposed methods using various benchmarks.

In the next chapter, I present a reliable fuzzy security protocol for trust management in TM-IoT. I term the proposed fuzzy security protocol Fuzzy-IoT and it is presented in detail in the next chapter.

CHAPTER 7

RELIABLE FUZZY-LOGIC BASED PROTOCOL FOR ENSURING THE RELIABILITY OF TM-IOT (FUZZY-IOT)

7.1 INTRODUCTION

Nowadays, the Internet of Things (IoT) is becoming used in every aspect of human life. Issues centred on cyber security are at the forefront of IoT deployments. One of the possible solutions to achieve robust security in IoT is enabling trusted communication between the IoT things (nodes). In Chapter 2, I inferred from the current literature there is lack of approaches for detecting malicious nodes in IoT. The malicious nodes in IoT can carry out various undesirable or untrustworthy activities or attacks that can comprise the reliability of the trust management solution. Also, in the existing literature there is no secure IoT communication protocol to enable a reliable and trustworthy communication between IoT nodes.

In the previous chapter, I presented the intelligent methods within TM-IoT that can ensure the scalability of the IoT network also the trust-based clustering of the IoT nodes

in the IoT paradigm. Building upon Chapter 6, in this chapter I propose fuzzy-logic based approaches for trust clustering of IoT nodes. Furthermore, I propose fuzzy-logic based algorithms to counter three different types of untrustworthy behaviours by the IoT nodes.

The main purpose or contributions of this chapter is to address the following IoT issues:

(a) First, I propose a fuzzy-logic based approach for trust-based clustering of the IoT nodes. The fuzzy-logic based clustering algorithm is based on the non-fuzzy counterpart clustering algorithm presented in Chapter 6.

(b) Second, there are lack of a standard messaging system similar to serial communication for secure encryption-based messaging in the IoT network. I propose a secure trust-based messaging system to address this shortcoming.

(c) Third, there are issues in detecting malicious nodes and restricting their untrusted function during communication with other nodes. To address this gap, the focus of this chapter is on proposing intelligent fuzzy-logic based solutions and techniques to address the four types of untrustworthy behaviours by the IoT nodes, namely bad-mouthing attacks, contradictory behaviour attacks, on-off attacks and bad service attack by IoT nodes communications. The fuzzy-logic based algorithms are based on their non-fuzzy counterpart algorithms that have been presented in Chapter 6.

I term the combination of all the above three contributions as Fuzzy-IoT. Fuzzy-IoT is the first work of its type in combining both intelligent approaches for countering untrustworthy behaviours and proposing fuzzy-logic based clustering methods. Further, the effectiveness of fuzzy logic in trust clustering and detection of untrustworthy behaviours is demonstrated. The contributions in this chapter can reduce the risk of the IoT nodes carrying out the above mentioned cyber-attacks on TM-IoT.

The chapter is divided as follows. Section 7.2 provides the technique of secure message system for the communication between IoT nodes. Section 7.3 presents the proposed fuzzy-logic based algorithms for both intelligent trust-based clustering and countering untrustworthy behaviours in TM-IoT. Section 7.4 provides a Trust-based communication protocol in TM-IoT. Section 7.5 describes the simulation results of the proposed solution along with IoT node mechanisms and demonstrate the results and analysis of this solution. Section 7.6 concludes this chapter.

7.2 A SECURE HEXA DECIMAL-BASED MESSAGING SYSTEM FOR TAMPER DETECTION

The structure of the message in our proposed protocol is shown in Figure 7-1 and Table 7-1. Each unit/code of the message is a two-digit hexadecimal number or an unsigned char with values from 0 - 255. *Message Length* refers to the length of the data section. *Check Code* is generated by processing other hexadecimals in the message. In our simulation, the *Check Code* is generated by adding the *Data Sections* together. If the result is greater than 255, reduce 256 from the result the result is reduced until it is smaller or equal to 255.

The messaging system provides two extra layers of security. If the *head code* of a received message is wrong, the current message will be discarded, protecting the system from an outside source or malicious nodes. The *check code* effectively serves as an authentication mechanism. A different *check code* implies a wrong *check code* generation mechanism, which means the message is from an unsecure source IoT node origin.

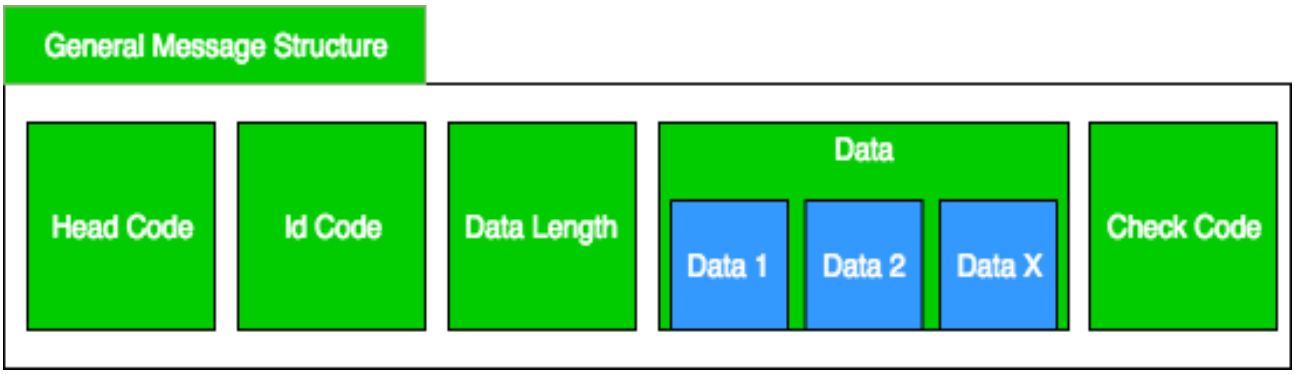

Figure 7-1 General structure of a message

Table 7-1 A description of the general structure of a message

Code	Description	Length (bytes)
Head Code	The head code is the same for all messages within the system	1
Id Code	Determines which operation should be performed with the current message	1
Data Length	Determines the length of the data section	1
Data	Data required for the operation performed by the current message	Depends on the operation
Check Code	Calculated by the code before sending the message. Will be calculated again at the receiver node and compared to verify the validity of the message	1

7.3 FUZZY LOGIC-BASED APPROACH FOR COUNTERING ATTACKS ON IOT

Our proposed solution (Fuzzy-IoT) comprises of five algorithms as follows. Algorithm 7.1 is used to classify the trust score values of cluster nodes into fuzzy sets. After determining the fuzzy sets, algorithm 7.2 uses these fuzzy sets and classifies the cluster nodes into three categories: trusted, semi-trusted and non-trusted. These categories are used to restrict node interaction between them. Algorithm 7.3 uses a direct trust score, indirect trust score and routing score to calculate the trust value of each cluster node. Algorithm 7.4 uses this trust value to determine if a cluster node is able to change to another cluster. Finally, Algorithm 7.5 checks the current condition of the cluster nodes and uses all the previous algorithms to update a new fuzzy status and new trust value. It also uses trust values for clustering. Consequently, the fuzzy state will be used to limit node functionality and the calculated trust values will be used for clustering with trust boundaries stored on a certain master nodes.

Algorithm 7.1 uses fuzzy boundaries to determine a low, medium or high value for the three different trust scores. The trust values in our system range between 0 to 1. The fuzzy boundaries are defined as in Figure 7-2.

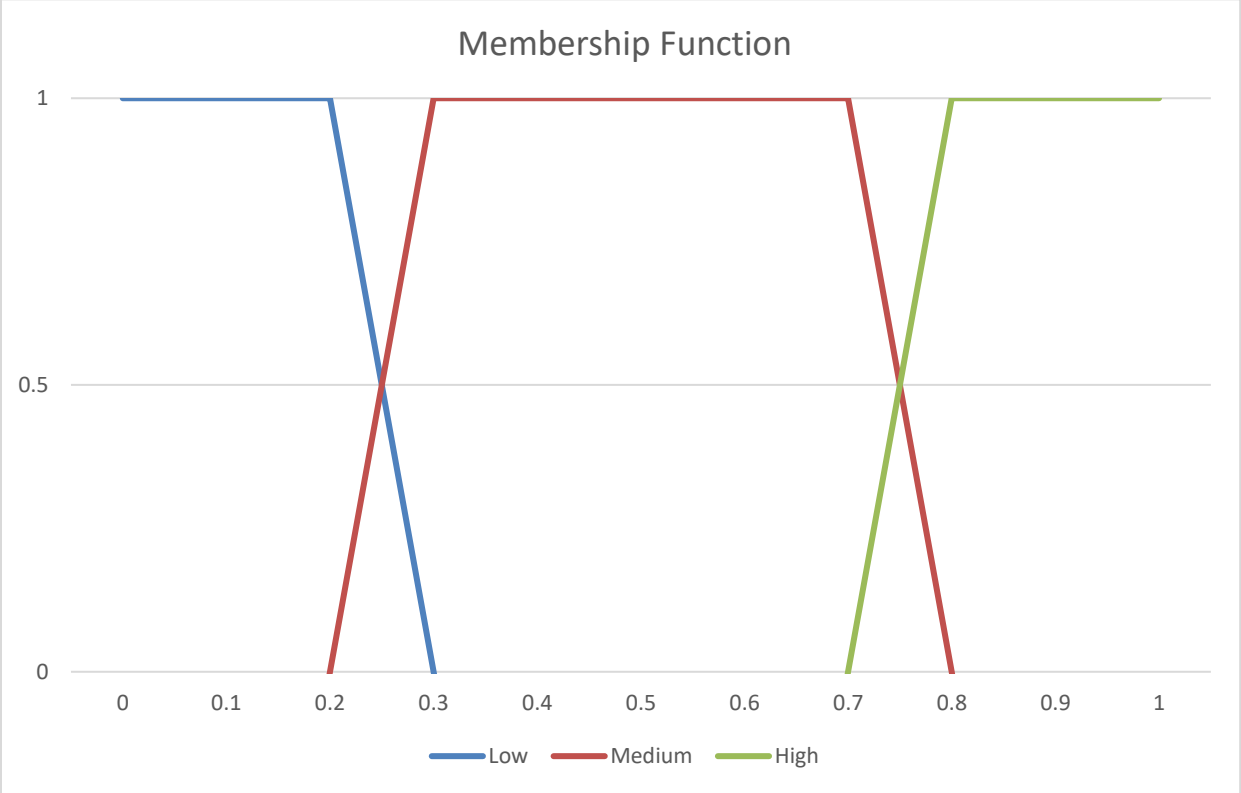

Figure 7-2 Membership function of fuzzy-logic

Algorithm 7.1 evaluates an input value with all three fuzzy sets (Low trust, Medium trust and high trust), where obtaining a set of three 1 or 0 results represents membership or non-membership of the sets. Each of the values is tested with a set of three equations with an upper bound and lower bound of trust. For example, with the medium number set, a value is compared that is between the lower bound of medium set *Ml* and the upper bound of medium set *Mu*.

$$Ml < value < Mu \qquad (7.1)$$

Algorithm 7.1 Fuzzification Algorithm

```
Require: float Sc; upper boundary for low, Lu; upper and lower boundaries for medium
  and high, Mu, Ml, Hu, Hl
  String result ← ""
  if Sc <Lu then
      result ← "1"
  else
      result ← "0"
  end if
  if Sc >Ml and Sc <Mu then
      result ← result + "1"
  else
      result ← result + "0"
  end if
  if Sc >Hl and Sc <Hu then
      result ← result + "1"
  else
      result ← result + "0"
  end if
  Return result
```

In our system, there will be three trust scores as components of the trust value for each IoT node. The three score are (direct, history and routing) scores. A direct score will be generated from the quality of a service delivered by that node. When a new direct score is stored, the last one will become the new history score. I regard routing scores as values that are generated by evaluating the quality of service responses from a node in a different cluster.

Putting these three scores into algorithm 7.1 generates the fuzzy membership for each of the three scores. Algorithm 7.2 uses the results of algorithm 7.1 to provide the fuzzy state of a node. A given node can have one of the three fuzzy states. All nodes start with the non-restricted state “trusted”. As more nodes give more trust scores, it could move down to more states that are restricted. A “semi-trusted” node can only provide

services to nodes within the same cluster. “Semi-trusted” nodes are restricted to provide services to an outside node (outside node of nodes outside the “semi-trusted” status). A service request to a “trusted” node or to a “non-trusted” node will be blocked by its master node.

Algorithm 7.2 is logic-based. If the routing score (Rr) is low AND the direct score (Rd) is NOT low AND the past score (Rp) is not low, this node is semi-trusted. If Rd is low OR Rp is low, then this node is non-trusted. All nodes start in a trusted state. These states can only downgrade towards the non-trusted state. The Algorithm 7.2 is used to intelligently detect on-off attacks in TM-IoT.

Algorithm 7.2 Fuzzy Trust state detection

Require: fuzzification result of Direct, Past, Routing Scores, Rd, Rp, Rr;

```
if Rr is Low AND Rd is not Low AND Rp is not Low then
    Return Semi-Trusted
end if
if Rd is Low OR Rp is Low then
    Return Non-trusted
end if
Return Trusted
```

By directly using the average of the trust scores obtained from the other cluster nodes and other master nodes, a trust value can be obtained. Three direct scores, a past score and routing score coefficients Cd, Cp and Cr are values with a sum of 1. These coefficients are used to provide a weighted average of direct, past and routing scores (Srd, Srp, Srr). In our approach, I use an average as in Equation 7.2 below, however a weighted approach could be used as well.

$$result = Cd \times Srd + Cp \times Srp + Cr \times Srr \quad (7.2)$$

Algorithm 7.3 Calculation of trust value

Require: Response, History, Routing Coefficients, Cd, Cp, Cr; Direct, Past, Routing Scores. Srd, Srp, Srr;

$result \leftarrow Cd \times Srd + Cp \times Srp + Cr \times Srr$

Return result

When a node's trust value is not within the boundaries of the current master node, Algorithm 7.4 uses the trust values calculated from algorithm 7.3 and checks if it is within the range of another master nodes' trust value boundaries (Tnl and Tnu). These boundaries can be requested from the super node. The working of the algorithm for switching a node from one cluster to another by the Master Node of the cluster to which the node belongs is presented in Algorithm 7.4 below.

Algorithm 7.4 Request to switch cluster

Require: cluster node, CN; CN's trust value, Tcn; CN's Non-trusted status, NTs; Tcn; trust boundaries of new master node, Tnu, Tnl;

if $Tnl \leq Tcn \leq Tnu$ **then**

Return Request Granted

end if

Return Request Declined

Algorithm 7.5 specifies the process by which a master node checks the status of the cluster nodes and decides whether it needs to move them. The direct, past and routing scores (Sdi, Spa) are obtained from other cluster nodes and Ssdi and Sspa are summed.

$$Ssdi = \sum Sdi \tag{7.3}$$

$$Sspa = \sum Spa \tag{7.4}$$

The average of these scores, Asdi and Aspa, will then be calculated. Then, routing scores Sro are obtained from the other master nodes to calculate Ssro.

$$Ssro = \sum Sro \tag{7.5}$$

Again, an average Asro will be calculated. Algorithm 7.2 and three will be used to produce a fuzzy state (NTs) and a trust value (Trust), respectively. The fuzzy state will be broadcasted to other same-cluster nodes. The trust value obtained will be checked against the upper and lower trust boundaries (Tu, Tl) of the current master node. If it is not within the boundary, algorithm 7.4 will be run to check if this node can move to another cluster. The working of the algorithm for to check the status of each Cluster Node is presented in Algorithm 7.5 below. Algorithm 7.5 is used to intelligently detect IoT nodes carrying out bad-mouthing attacks, contradictory behaviour attacks and low quality services in TM-IoT.

Algorithm 7.5 Check current cluster node status

Require: List of Cluster nodes CNs; upper and lower trust boundaries of current master node, Tu, Tl; other master nodes, MNs

for all CN in CNs **do**
  for all other cluster nodes in CNs **do**
    get Direct and Past scores, Sdi, Spa
    $Ssdi \leftarrow Ssdi + Sdi$
    $Sspa \leftarrow Sspa + Spa$
  end for
  $Asdi \leftarrow Average(Ssdi)$
  $Aspa \leftarrow Average(Sspa)$
  for all other master nodes in MNs **do**
    Request Routing Scores, Sro;
    $Ssro \leftarrow Ssro + Sro$
  end for
  $Asro \leftarrow Average(Ssro)$
  $NTs \leftarrow Algorithm2(Asdi, Aspa, Asro)$
  broadcast NTs to same cluster neighbours
  $Trust \leftarrow Algorithm1(Cd, Cp, Cr, Asdi, Aspa, Asro)$
  if $Tl \leq Trust \leq Tu$ **then**
    Continue to next CN
  else
    for all MN in MNs **do**
      if Algorithm4(CN, Trust, NTs, trust boundaries of MN) is Granted **then**
        Continue to next CN
      end if
    end for
  end if
end for

The following flowchart demonstrates the working of Algorithm 7.5.

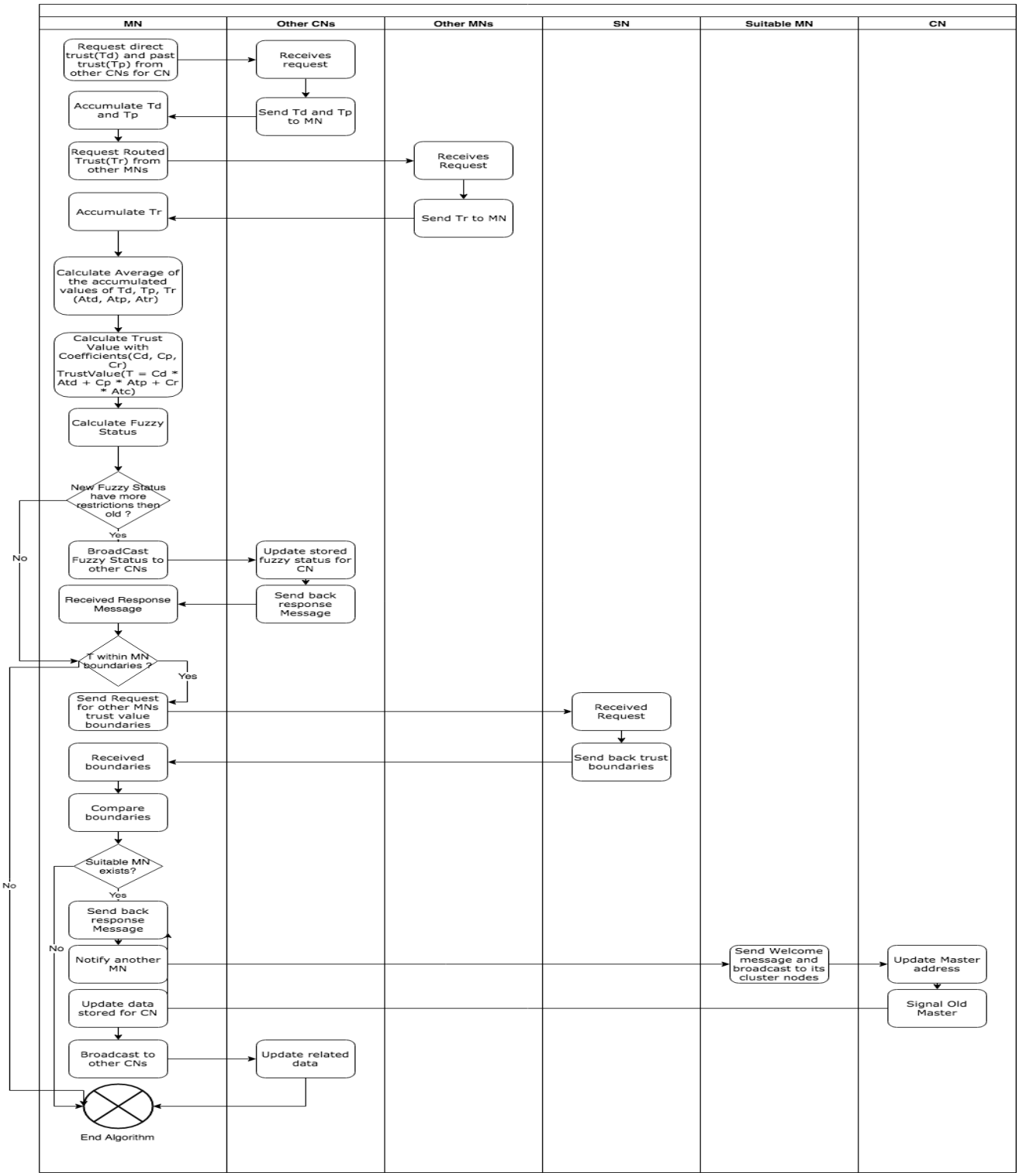

Figure 7-3 Flowchart showing pictorially the working of the algorithm 7.5

Figure 7-3 describes the flowchart of algorithm 7.5 specifies the process by which a master node checks the current status of the cluster nodes and decides whether it needs to move them. The direct and past trust scores (Sdi, Spa) are obtained from other cluster nodes and Ssdi and Sspa are summed, equations (7.3 and 7.4)

The average of these scores, Asdi and Aspa, will then be calculated. Then, routing scores Sro are obtained from the other master nodes to calculate Ssro, equation (7.5). Again, an average Asro will be calculated. Algorithm 7.2 and three will be used to produce a fuzzy state (NTs) and a trust value (Trust), respectively.

Algorithm 7.2 is based on fuzzy logic. If the routing score (Rr) is low AND the direct score (Rd) is NOT low AND the past score (Rp) is not low, this node is semi-trusted. If Rd is low OR Rp is low, then this node is non-trusted. All nodes start in a trusted state. These states can only downgrade towards the non-trusted state. Only if the new fuzzy status is lower than the current status, then update the current fuzzy status to the new fuzzy status.

The fuzzy state will be broadcasted to other same-cluster nodes. The trust value obtained will be checked against the upper and lower trust boundaries (Tu, Tl) of the current master node. If it is not within the boundary, Algorithm 7.4 will be run to check if this node can move to another cluster.

If the cluster node has a suitable new master node to move to. The current master will notify the new master node. The new master node will broadcast the welcome message to its cluster nodes and the moving cluster node. After changing the master node, the moving cluster node will signal the old master node to update the cluster node list. Finally, the old master node will broadcast to its cluster nodes to update their neighbour list.

7.4 TRUST-BASED COMMUNICATION PROTOCOL IN TM-IOT

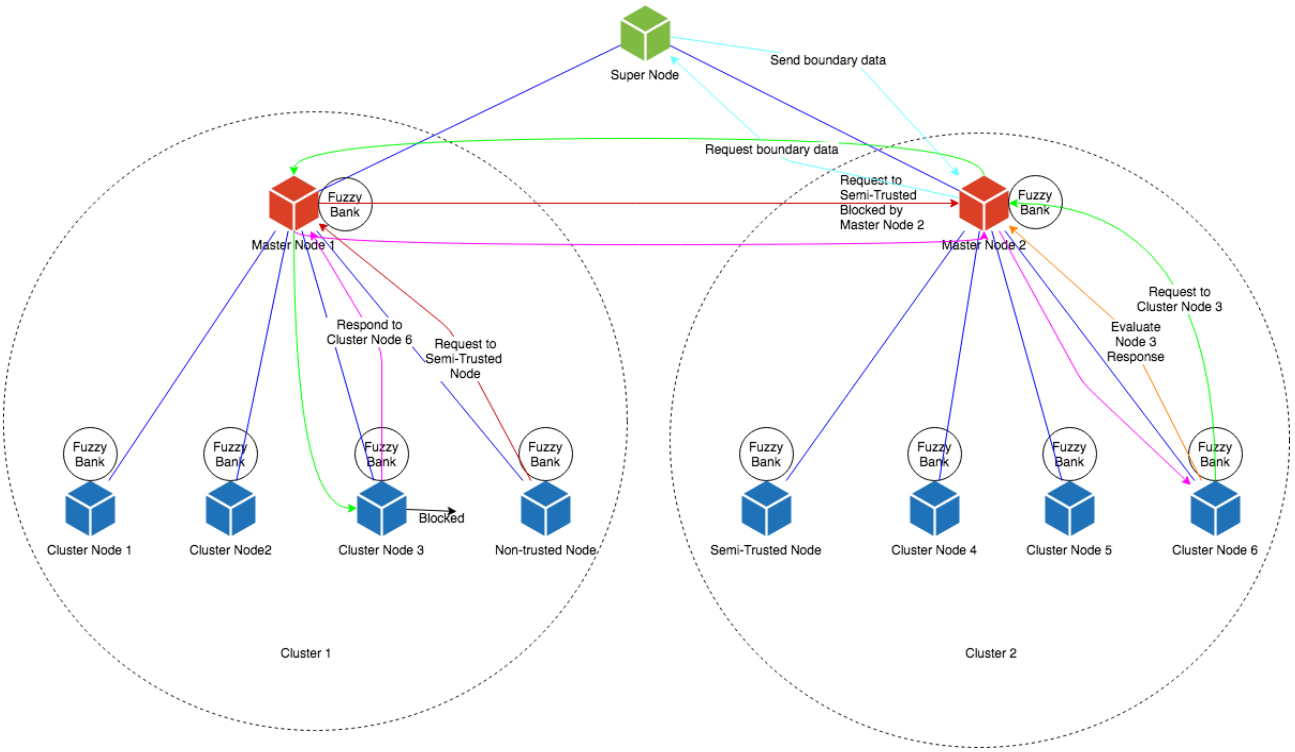

Figure 7-4 Trust-based message communication protocol in TM-IoT

Figure 7-4 pictorially represents the working of our proposed protocol for trust-based communication in an IoT environment. The blue lines represent the communication between the CN and its MN. The cluster nodes represented as the blue cubes can communicate with the master node (red cubes) of its cluster and with the neighbour nodes (i.e. IoT nodes within that cluster). The super node can only be contacted by the master nodes. The black lines demonstrate basic same-cluster interactions. Cluster nodes can be classified into three categories with restrictions as shown in Table 7-2.

Table 7-2 Types of nodes in the security protocol

Node type	Restrictions	Service quality
Normal nodes (trusted nodes)	No restrictions	Provides an acceptable quality service to other nodes
Semi-trusted nodes	Cannot send a routed service response to a service request from a node in another cluster. Able to send out service requests.	Reliable service for the same cluster nodes. Bad quality service to nodes in another cluster
Non-trusted nodes	Cannot respond to any service requests. Able to send out service requests	Bad quality service

All cluster nodes have a fuzzy bank is a cache of trust value stored in its memory about the fuzzy status or 'node type' of its neighbours. For the same cluster request, the sender will firstly check the node type of the target node. If the target node is non-trusted, such as cluster node 3, the node will correspondingly make a trust-based decision and not send the request to the non-trusted node.

A service request initiating from a node from a cluster (assume cluster 1) to a node in a different cluster (assume cluster 2) will be routed through the master nodes of both the clusters. The green line in Figure 7-4 shows a request reaching from cluster node 6 to cluster node 3, routed from master node 2 to master node 1 and finally to cluster node 3. The purple lines show a response which is sent back from cluster 3 to cluster 6

routed by master node 1 and master node 2. Finally, the orange line shows a score from cluster node 6 being sent to master node 2 for cluster node 3's services.

The trust repository of the master nodes stores the trust value of its cluster nodes. It also stores the trust scores provided by the cluster nodes in another cluster. The red lines show a non-trusted node sending a service request to the semi-trusted nodes. Master node 2 checks the trust status of the semi-trusted node and blocks the request.

The light blue lines (annotated by 'Send Boundary Data') show that the super node provides trust value boundaries for the master nodes to be used in a future cluster change algorithm (Algorithm 7.5).

An IoT node can directly transmit a message to another node within its cluster, but it needs a master node to redirect a message to another master node, then to a node within another cluster. For direct transmission, between two nodes if the fuzzy status of the target node stored on the sending node is non-trusted, the sending node will stop sending the message. For a redirected message if the master node of the target determines if the recipient CN has one of these two trust values (semi-trusted or non-trusted) in such case the MN will block the message immediately.

7.5 Simulation, Node Mechanisms, Results and Analysis

This section presents the details of simulation, node mechanisms and the analysis of results.

7.5.1 SIMULATION SETTINGS: BASIC CONCEPT

The simulation is run with the Cooja simulator from the Contiki system. Contiki is an event-driven and light-weight system for the IoT. In this simulation, Rime addresses from Contiki are used for node identification. By default, Rime addresses can be 2 bytes or 8 bytes. Considering the small size of our clusters, 2-byte addresses will be sufficient. There are two major sets of simulations: the base case with the fuzzy detection mentioned in the node mechanism section; and the second case without fuzzy detection. These two sets of simulations are further developed in three different simulation scenarios. The first scenario only has 15 nodes to test the basic counter-attack concept of the algorithm. The second scenario utilises 200 nodes to ensure the algorithm is functional under a large number of nodes. The final scenario contains 2000 nodes showing the algorithm's performance on a large-sized network.

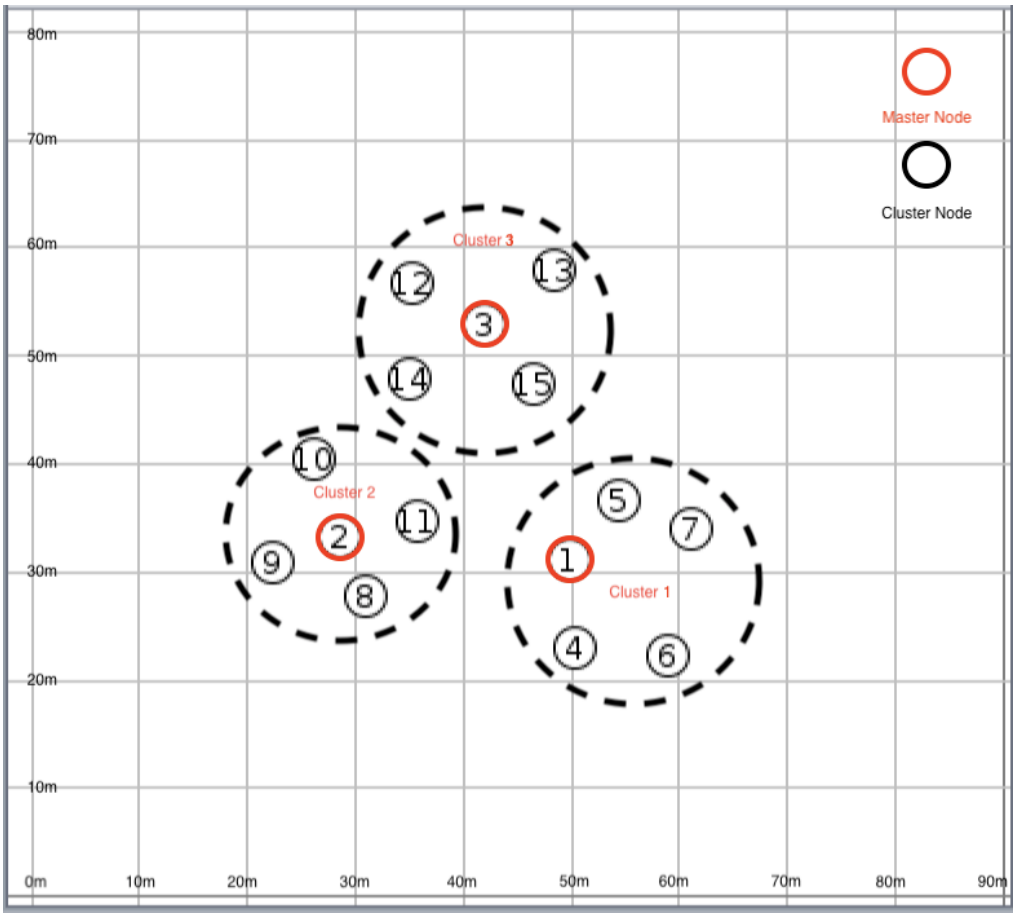

Figure 7-5 Basic concept initial simulation settings

Table 7-3 Basic concept node initial clustering of the nodes in the simulation

Master Node	Cluster nodes
1	4,5,6,7
2	8,9,10,11
3	12,13,14,15

Table 7-4 The basic concept properties of the nodes in the simulation

Node	Rime Address	Node Type
1	1.0	Master node
2	2.0	Master node
3	3.0	Master node
4	4.0	Cluster node
5	5.0	Cluster node
6	6.0	Cluster node
7	7.0	Bad node
8	8.0	Cluster node
9	9.0	Cluster node
10	10.0	Cluster node
11	11.0	On-Off attack node
12	12.0	Cluster node
13	13.0	Cluster node
14	14.0	Cluster node
15	15.0	Contradictory behaviour node

In our algorithm, the function of a super node is to send trust boundaries to the master nodes. Because this is a small system, I simplified the simulation and stored the boundaries on the master nodes. Three malicious nodes are in the system for detection: 1) a bad service provider which sends low scoring service responses; 2) an on-of f attack node which provides a bad service in a certain cycle of time; and 3) a contradictory behaviour node which sends a bad service response redirected to nodes within other clusters.

Table 7-3 demonstrates the initial clustering of the 15 nodes. Node 1, Node 2 and Node 3 act as master nodes for the three initial clusters. Table 7-4 shows the details of the types of nodes within the simulation. Nodes 7, 11 and 15 are malicious nodes acting as a bad service provider providing bad responses, a node performing on-off attacks and a node performing contradictory-behaviour attacks, respectively. These nodes are distributed across a 90m × 80m surface into three clusters as shown in Figure 7-5.

This simulation consists of two cases. Table 7-5 details the base case. The base case takes place on a 90m × 80m surface for 60s with the Fuzzy Trigger on. If the Fuzzy Trigger is on, the cluster nodes and master nodes block the requests sent to the non-trusted and semi-trusted nodes, as discussed in section 7.5. Table 7-6 shows the settings for the second case. The only difference between the cases is the second case runs with the Fuzzy Trigger off. In this case, the master nodes and cluster nodes will not block any requests in light of the fuzzy status.

Table 7-5 Basic concept base case parameters

Parameter	Value
Number of nodes	15
Number of clusters	3
Simulation time	60s
Simulation area	90m × 80m
Fuzzy Trigger	On

Table 7-6 Basic concept non-fuzzy case parameters

Parameter	Value
Number of nodes	15
Number of clusters	3
Simulation time	60s
Simulation area	90m × 80m
Fuzzy Trigger	Off

7.5.2 SIMULATION SETTINGS: (200 NODES)

The second simulation scenario is used to test the proposed system in a network of 200 IoT nodes with 60 being malicious.

Table 7-7 (200 nodes) IoT network node initial clustering

Mater Node (MN)	Cluster Nodes (CNs)
1	21 – 29
2	30 – 38
3	39 – 47
4	48 – 56
5	57 – 65
6	66 – 74
7	75 – 83
8	84 – 92
9	93 – 101
10	102 – 110
11	111 – 119

12	120 – 128
13	129 – 137
14	138 – 146
15	147 – 155
16	156 – 164
17	165 – 173
18	174 – 182
19	183 – 191
20	192 - 200

Table 7-8 (200 nodes) IoT network node properties

Node	Rime Address	Node Type
1 – 20	1.0 – 20.0	Master node
The first six cluster nodes in each cluster		Cluster node
The seventh cluster node in each cluster		Bad Service Provider
The eighth cluster node in each cluster		On-off Attack node
The ninth cluster node in each cluster		Contradictory behaviour node

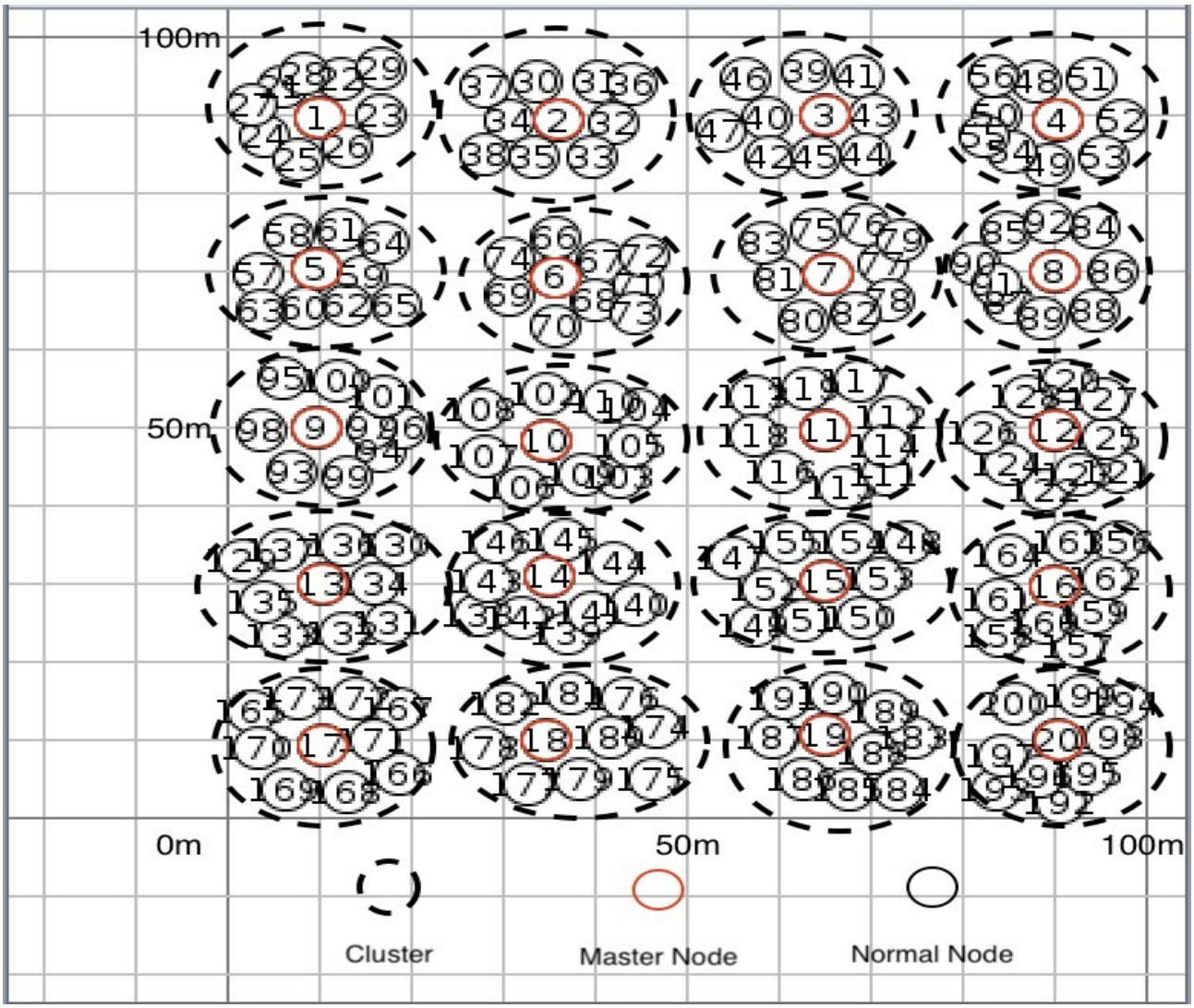

Figure 7-6 200 nodes IoT network initial simulation setting

Table 7-7 shows the initial cluster settings of the network. The first 20 nodes in the network are all master nodes, each carrying nine cluster nodes in their cluster. Table 7-8 shows the node property composition of every cluster. In every cluster, the last three nodes are malicious nodes. These three nodes are the bad service provider, on-off attack node and contradictory behaviour node. A topology of this case is shown in Figure 7-6.

This simulation scenario also consists of two cases. Table 7-9 demonstrates the base case. The base case takes place on a 100m × 100m surface for 60s with the Fuzzy Trigger on. Table 7-10 shows the settings for the second case with the *Fuzzy Trigger* off.

Table 7-9 (200 Nodes) IoT network base case parameters

Parameter	Value
Number of nodes	200
Number of clusters	20
Simulation time	60s
Simulation area	100m x 100m
Fuzzy Trigger	On

Table 7-10 (200 nodes) IoT network non-fuzzy case parameters

Parameter	Value
Number of nodes	200
Number of clusters	20
Simulation time	60s
Simulation area	100m x 100m
Fuzzy Trigger	Off

7.5.3 SIMULATION SETTINGS: LARGE SCALE IOT NETWORK

The second simulation scenario is used to test the proposed system in a large scale IoT network with 2000 nodes to prove the proposed approach is able to scale to any number of IoT nodes. Of these 2000 nodes, 60 are malicious.

Table 7-11 (Large-scale) IoT network node initial clustering

Master Node	Normal node number	Cluster property	Description
1 – 20	6	Malicious cluster	Last three nodes are bad service, on-off and contradictory behaviour node.
21 – 200	9	Normal Cluster	All nodes are not malicious

Figure 7-7 Large-scale IoT network sample malicious cluster setting

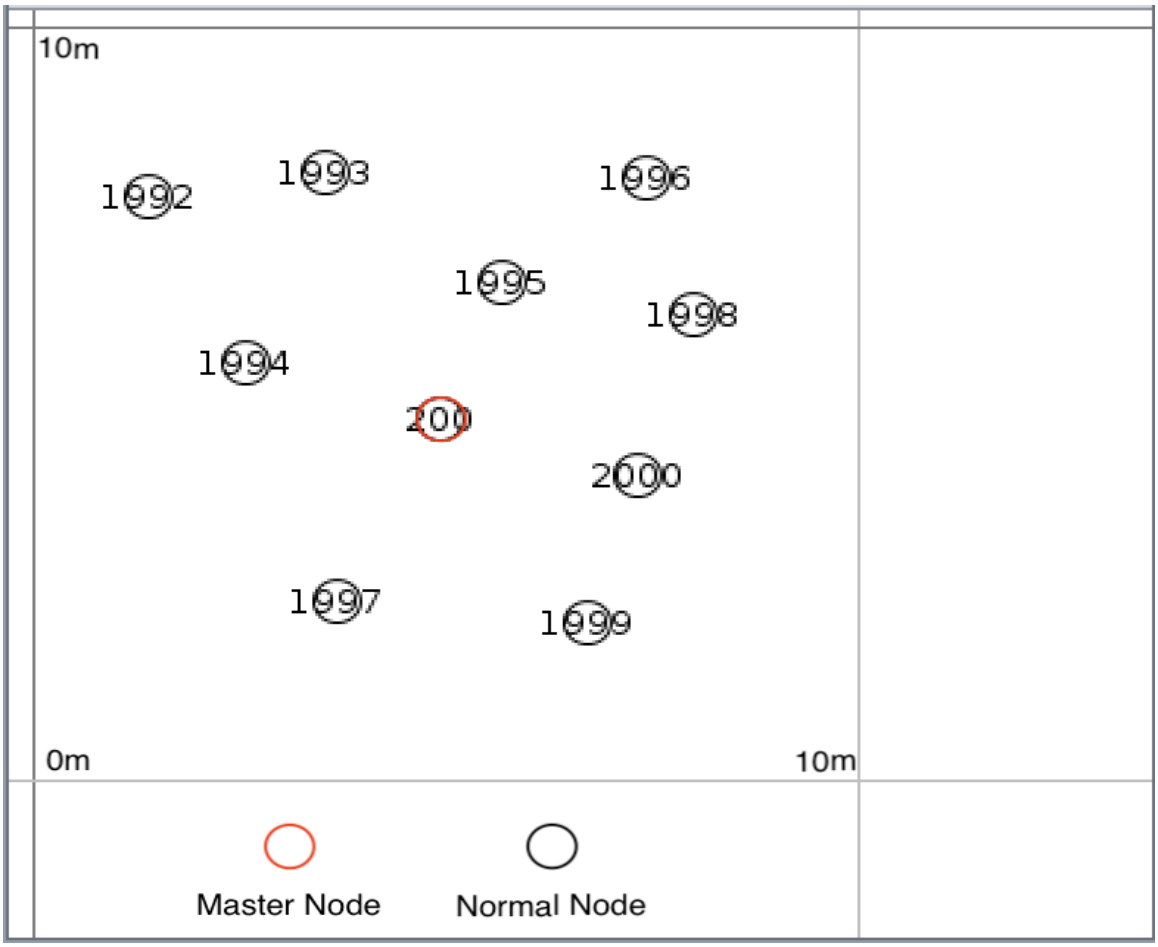

Figure 7-8 Large-scale IoT network sample normal cluster setting

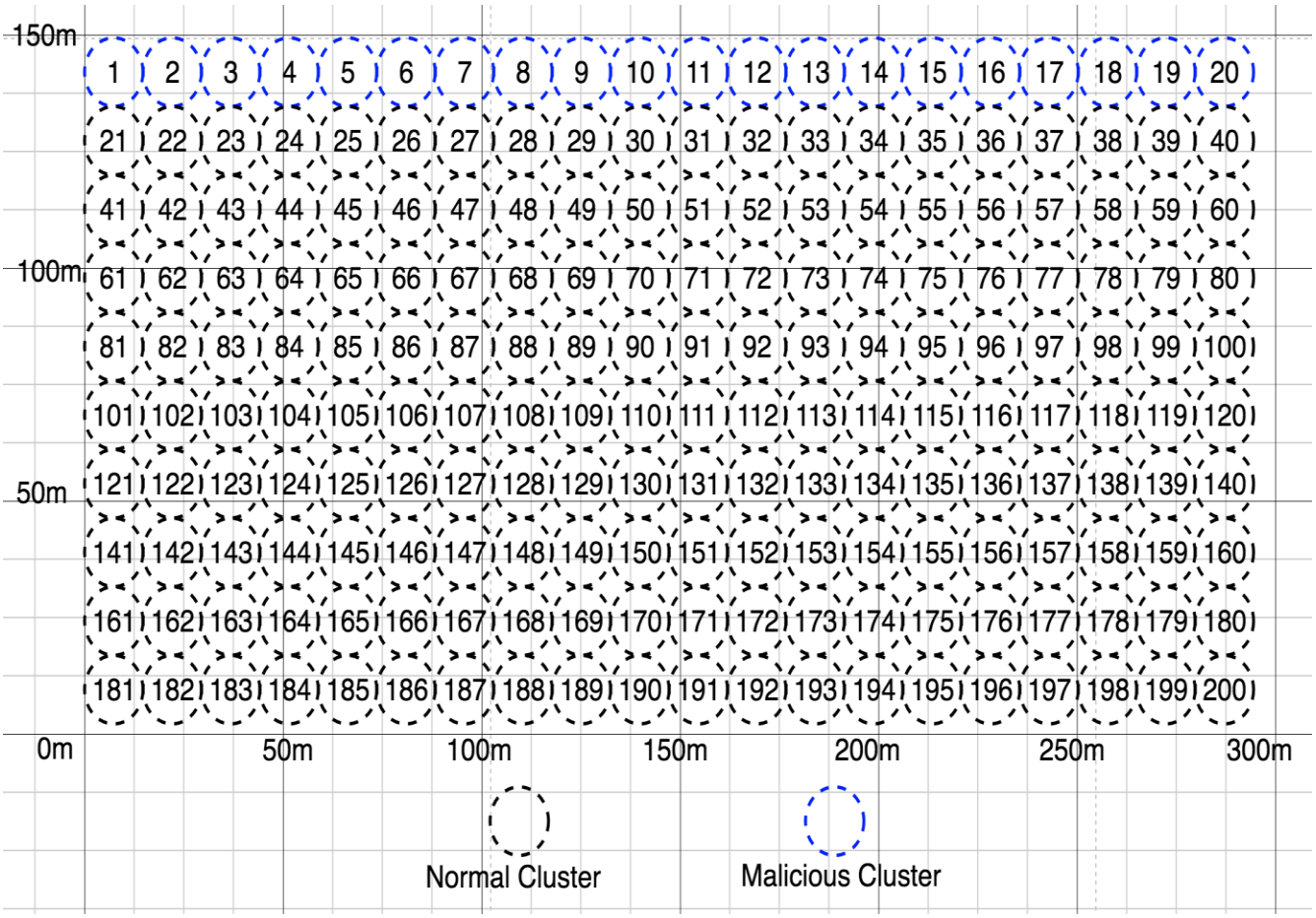

Figure 7-9 Large IoT network initial simulation setting

Table 7-11 presents the initial properties of the clusters. The first 20 clusters are malicious clusters, each of which contains three malicious nodes. These clusters have a similar topology to that shown in Figure 7-7. The 21st to the final cluster are normal clusters, all with non-malicious cluster nodes. Figure 7-8 illustrates this setting. Figure 7-9 reveals the big picture of cluster distribution for a large IoT network with 2000 nodes. Each cluster contains a master node and nine cluster nodes.

Table 7-12 demonstrates the base case of this simulation scenario. The base case takes place on a 300m × 150m surface for 60s with the *Fuzzy Trigger* on. Table 7-13 shows the settings for the second case with the *Fuzzy Trigger* off.

Table 7-12 (Large-scale) IoT network base case parameter

Parameter	Value
Number of nodes	2000
Number of clusters	200
Simulation time	60s
Simulation area	300m x 150m
Fuzzy Trigger	On

Table 7-13 (Large-scale) IoT network non-fuzzy case parameters

Parameter	Value
Number of nodes	2000
Number of clusters	200
Simulation time	60s
Simulation area	300m x 150m
Fuzzy Trigger	Off

7.5.4 RESULTS AND ANALYSIS

An analysis of the experimentation resulted in the following: Three bar graphs to measure the time is takes to intelligently detect the nodes carrying out three types of attacks - bad service provider, on-off attack and contradictory behaviour attack.

i) A performance evaluation and comparison to detect on-off attacks. The performance evaluation and comparison are carried out for both fuzzy and non-fuzzy cases.

ii) A performance evaluation and comparison to detect contradictory behaviour attacks. The performance evaluation and comparison are carried out for both fuzzy and non-fuzzy cases.

iii) Three diagrams comparing the average trust values for fuzzy and non- fuzzy cases. The first has similar settings to prove the basic concept. The second diagram is obtained in an environment with 200 nodes. The final diagram

is obtained in an environment with 2000 nodes. The second and third diagrams are used to prove that the proposed algorithms can operate with a large number of IoT nodes.

The bar graph in Figure 7-10 shows the number of rounds required to detect a certain IoT node carrying out on-off attacks, contradictory behaviour attacks and bad-mouthing attacks. A round is defined when a master node has run the first five algorithms (Algorithm 7.1 to Algorithm 7.5). This process includes gathering the trust scores from the master nodes and cluster nodes, calculating the trust values and fuzzy status, assigning a fuzzy status and finally moving out of the trust value boundary cluster nodes to a suitable master node. As can be seen in Figure 7-10, all of the bad service providers, the on-off attack nodes and the contradictory behaviour attack nodes are detected during round 1 of the master nodes' cycle. This demonstrates that this algorithm is quite efficient in detecting these malicious node types.

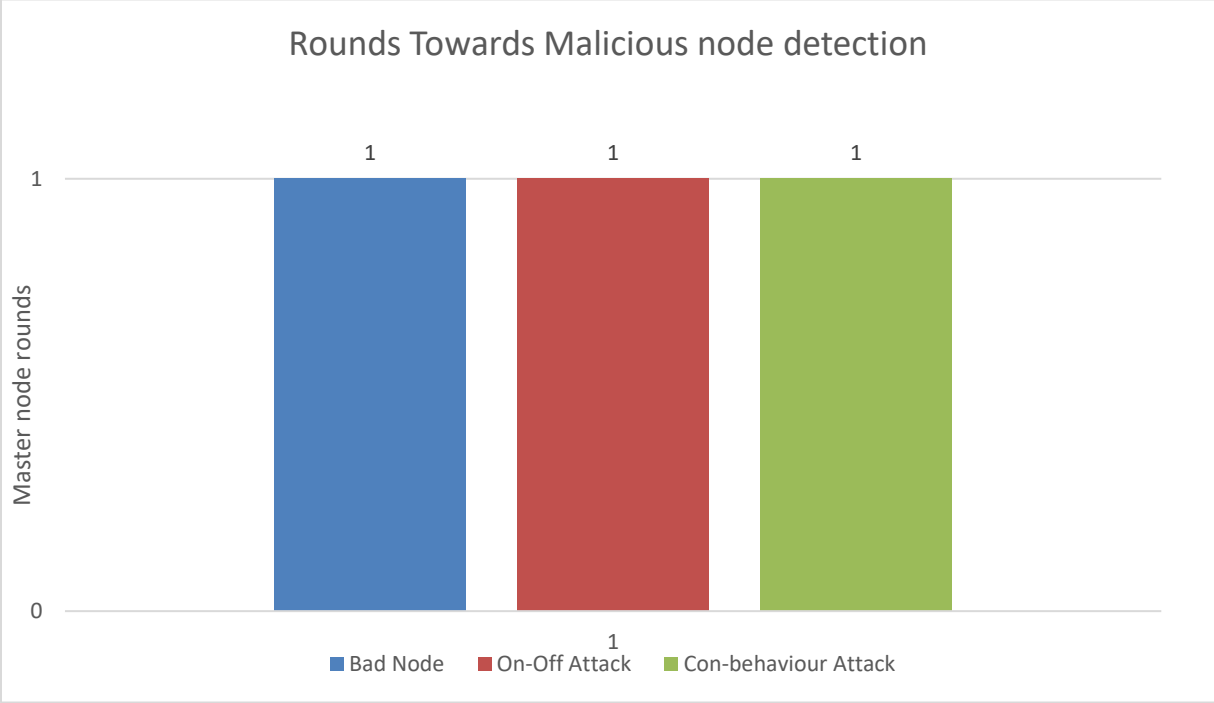

Figure 7-10 Rounds taken to detect malicious nodes

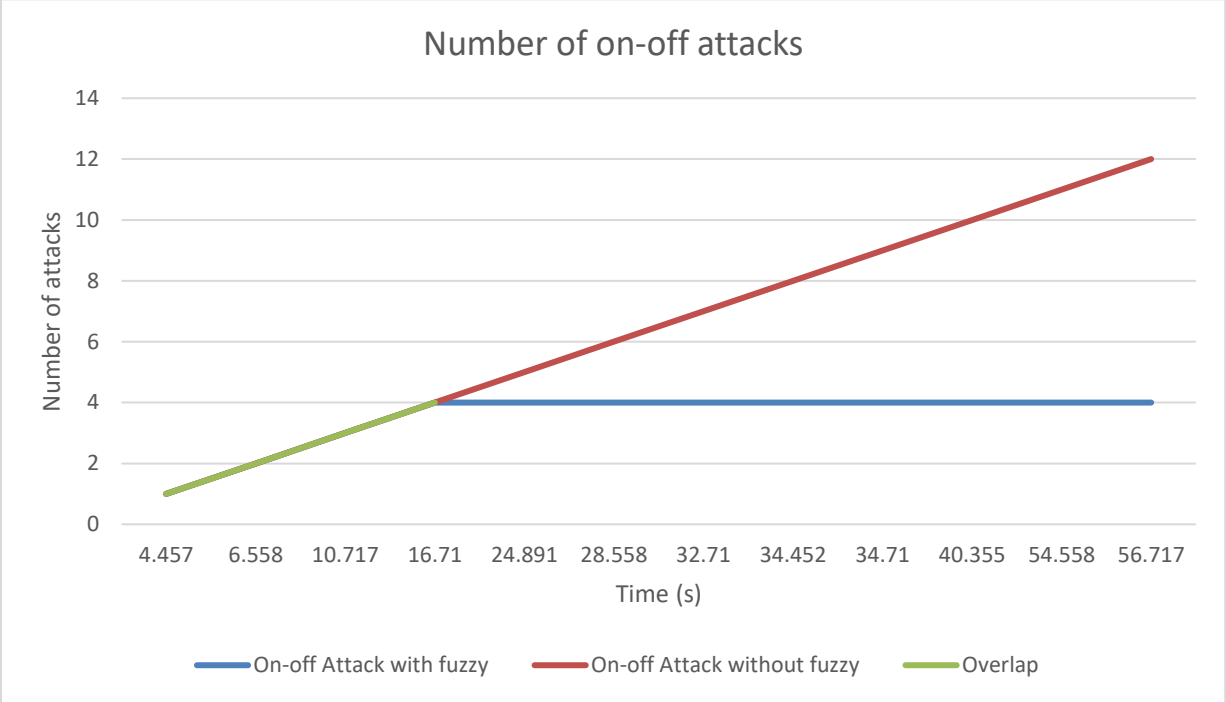

Figure 7-11 Time taken to detect a node carrying out on-off attacks

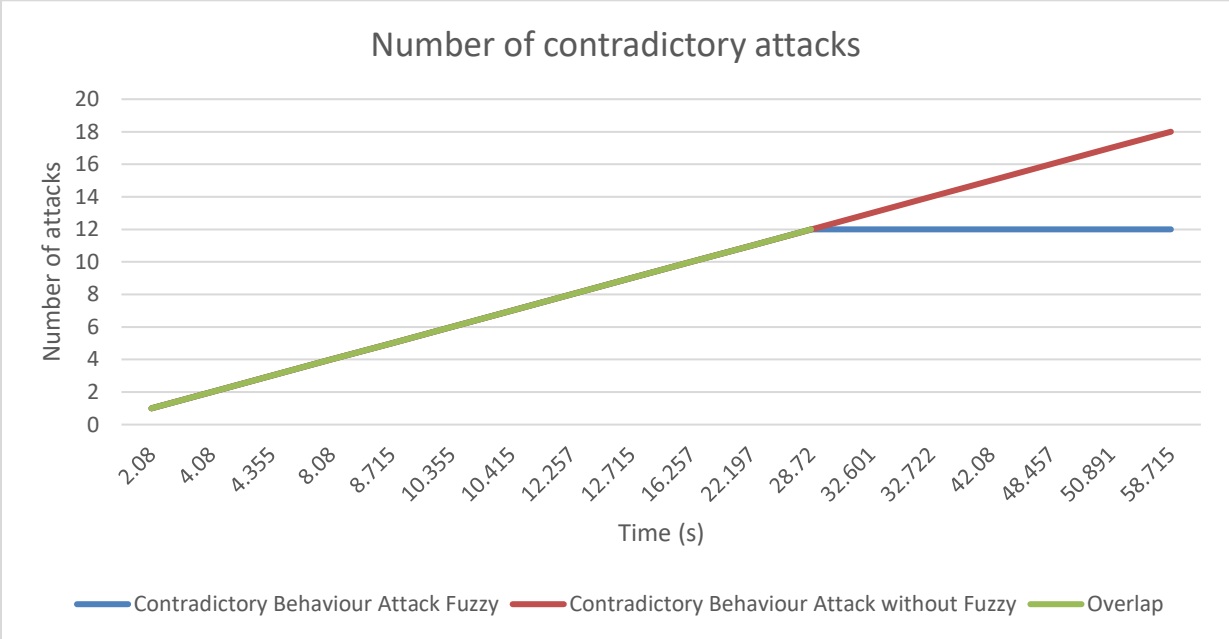

Figure 7-12 Number of contradictory behaviour attacks

Figure 7-11 compares the performance of our proposed fuzzy-logic based and non-fuzzy logic based algorithms to counter 'on-off attacks' in TM-IoT. The Y-axis dimension (Number of attacks) is cumulative since the start of the simulation and the X-axis denotes the time dimension. Both the fuzzy and the non-fuzzy version of our algorithms to detect on-off attacks are applied from the start of the simulation. As shown by the blue line, at 16.71 seconds since the start of the simulation, the fuzzy-logic based algorithm is able to detect the IoT nodes carrying out the on-off attacks and remove them completely. The fuzzy-logic based algorithms is able to successfully quarantine the TM-IoT against any further on-off attacks.

As I can see in Figure 7-11 using our proposed fuzzy-logic based algorithm proposed in this chapter, no new on-off attacks occur after 16.71 seconds. In the case without fuzzy mechanisms, these attacks are not detected (i.e., go un-detected unlike the fuzzy-logic based counterpart algorithm), as indicated by the orange on-off attack line which shows a constant increase in these attacks. This demonstrates that the fuzzy-logic based algorithm presented in this chapter is effective in detecting on-off attacks and is able to block the attack after the initial bootstrapping time.

Similar to on-off attacks, as shown in Figure 7-12, a contradictory behaviour attacks node is determined at 28.72 seconds using the approach proposed in this chapter. The X-axis in Figure 7-12 shows the time dimension and the Y-axis is the cumulative number of contradictory behaviour attacks since the start of the simulation. Both the fuzzy and non-fuzzy algorithms are applied from the beginning of the simulation. As shown by the blue line, at 28.72 seconds since the start of the simulation, the fuzzy-logic based algorithm is able to detect the IoT nodes carrying out the contradictory attacks and remove them completely. The fuzzy-logic based algorithms is able to successfully quarantine the TM-IoT against any further contradictory attacks.

In contract, the red line indicating contradictory behaviour attacks without fuzzy mechanisms continues to increase as the attacks are not blocked. This demonstrates the

effectiveness of the proposed fuzzy-logic based mechanism to block contradictory behaviour attacks.

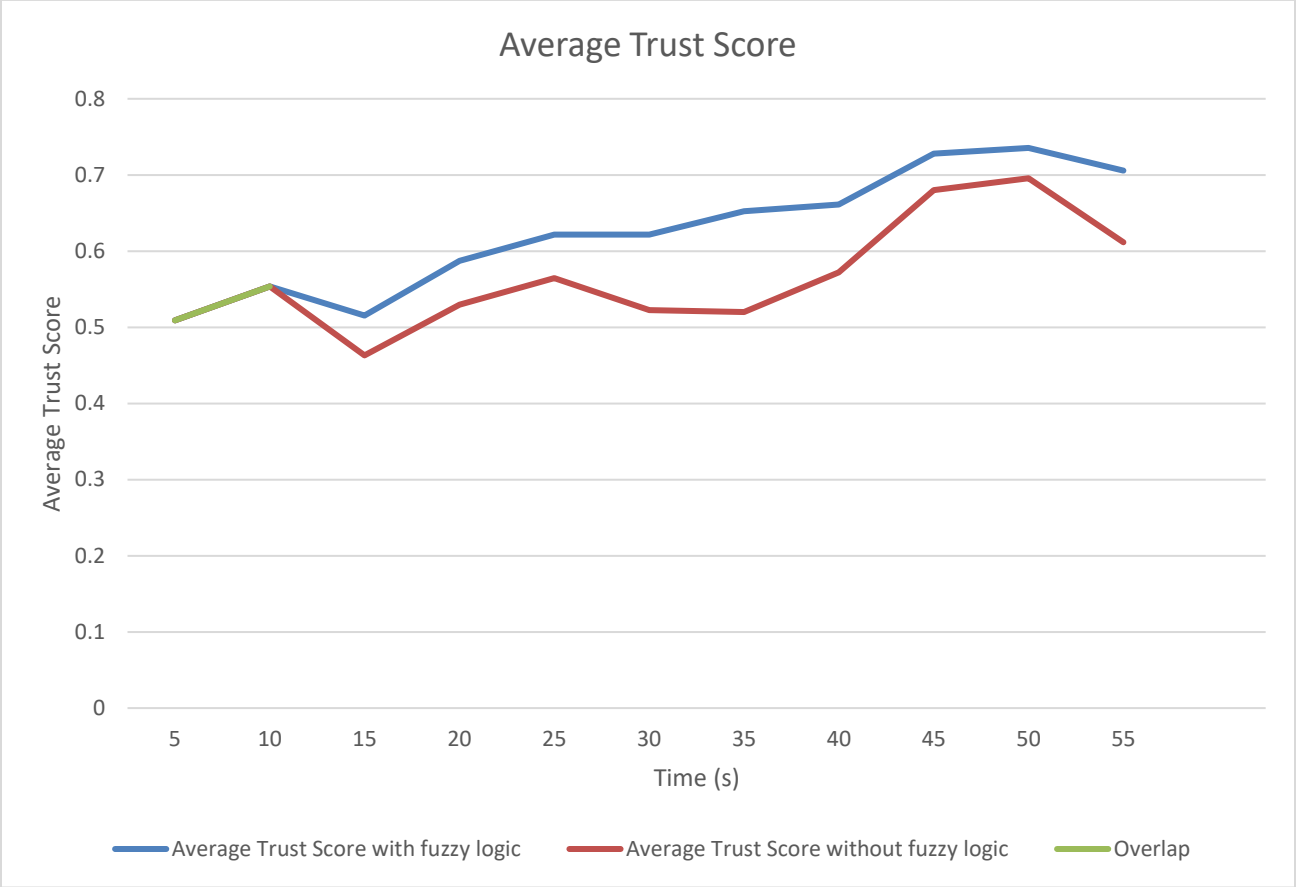

Figure 7-13 Average trust score of all the IoT nodes during the simulation (n=20 nodes)

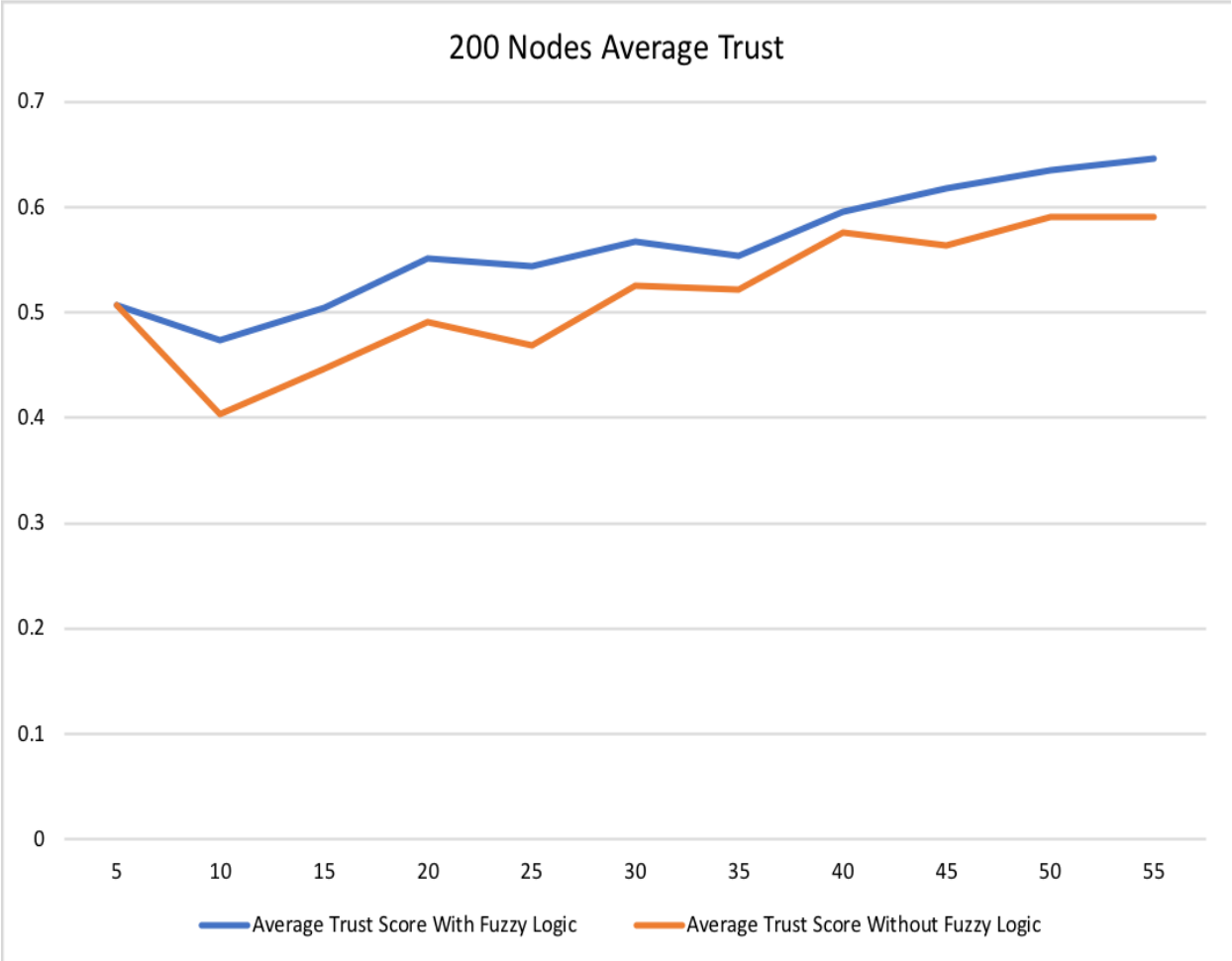

Figure 7-14 Average trust score of all the IoT nodes during the simulation (n=200 nodes)

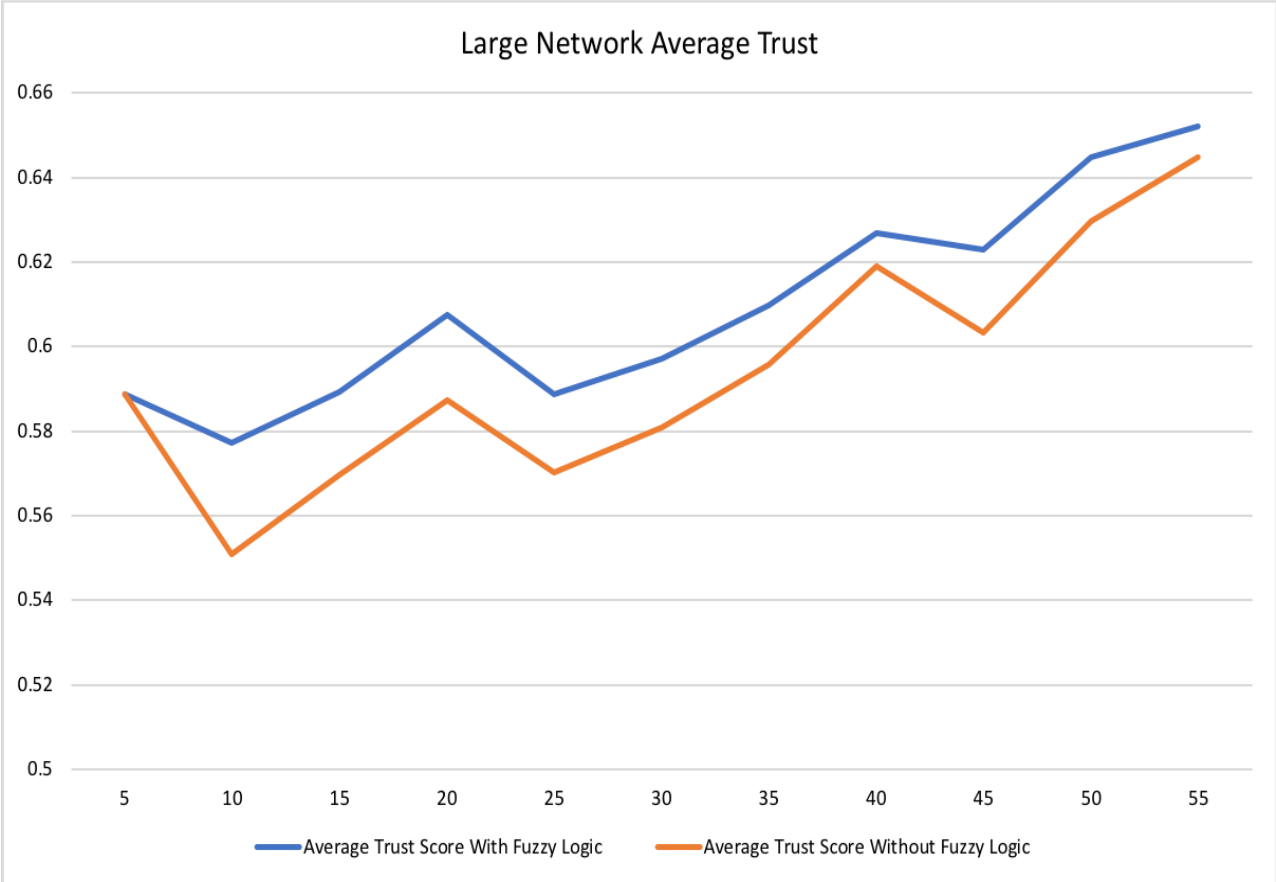

Figure 7-15 Average trust score of all the IoT nodes during the simulation (n=2000 nodes)

Figure 7-13 shows the average trust ratings for all the entire population of IoT nodes in the simulation. In our model, a node gives a score after receiving a service response which is the basis of the trust value. This score represents the service quality of the received service response. The lines indicting the average of the trust values using the fuzzy-logic based approach and the non-fuzzy logic based approach divergence occurs around 7 seconds. This is when the bad service provider, is detected by master node 1 (using fuzzy-logic based approach). A larger divergence occurs around 15 seconds and 30 seconds when node 11, the on-off attack node, and node 15, the contradictory

behaviour node, are detected, respectively. This shows there is an increase in the average service quality of the whole system when a fuzzy mechanism is added.

In this scenario of 200 nodes (Figure 7-14), the fuzzy logic-based approach is better is modelling the trust value than the non-fuzzy logic based approach. This figure is similar to the scenario with the large network with 2000 nodes (Figure 7-15). The only difference is the initial trust value and the difference between the fuzzy trust values and the non-fuzzy trust values. The base scenario has three malicious nodes in twelve cluster nodes, hence 25% of the nodes are malicious. However, in the scenario with 200 nodes, there are 60 malicious nodes in 10 cluster nodes, hence 33.3% of the nodes are malicious. With a greater number of malicious nodes, the initial trust value of 200 nodes is slightly lower than the base scenario. In the base scenario, one malicious node represents a third of all malicious nodes, on the other hand, one malicious node only represents 1.67% of total malicious nodes of the scenario 200 nodes. In this case, although there are more malicious nodes in every cluster of the 200 node network, the difference between the two lines is smaller than the base scenario. Figure 7-15 shows that the average trust of the large IoT network is similar. Only 3.33% of the 1800 cluster nodes are malicious. This equates to a significantly higher initial average trust value than the other two scenarios, but there is a smaller difference between the two cases of fuzzy logic, as each detection of a malicious nodes is not that significant.

7.6 CONCLUSION

In this chapter I presented fuzzy-logic based approaches for trust management in TM-IoT. I termed the collection of all the fuzzy-logic based approaches presented in this thesis as Fuzzy-IoT.

Building on TM-IoT (discussed in chapter 5) and CITM-IoT (discussed in chapter 6), in this chapter I presented fuzzy-logic based algorithms for trust clustering of the IoT

nodes. Furthermore, I presented fuzzy-logic based approached for intelligently detecting and countering on-off attacks, contradictory behaviour attacks, bad-mouthing attacks and bad service attacks. I also presented in this chapter, a secure messaging system for IoT nodes using hexadecimal values for communicating trust values. The experimental results and evaluation of the proposed approaches prove that they are able to achieve their intended goals. They are outline the superiority of fuzzy-logic for detecting untrustworthy behaviour by the IoT nodes.

The next chapter is the conclusion of the thesis.

CHAPTER 8

CONCLUSION AND FUTURE WORK

8.1 INTRODUCTION

I conclude the thesis in this chapter by presenting an overview of the outcomes achieved and by outlining the suggested future work.

The growth in the number of IoT devices has led to the need for intelligent, scalable and reliable trust management solutions in IoT. IoT is being deployed and used in various sectors such as medicine, transportation etc. This deployment and the increasing number of IoT devices (nodes) creates new challenges in ensuring the trustworthiness of the environment. The need for comprehensive trust management for IoT platforms is crucial from two perspectives: (a) it provides a reliable mechanism for deciding with which nodes to interact; and (b) it assists in creating a trusted environment among IoT entities thereby creating a secure and trustworthy IoT environment. Presenting a trust management approach for enabling a trustworthy IoT environment is a key part of the overall strategy of achieving cyber security in IoT. However, there are concomitant issues that need to be addressed, such as the scalability of the IoT trust management approach, resilience of the IoT trust management approach against inherent attacks such as bad mouthing by the malicious nodes etc.

A number of researchers have focused on the field of trust management in IoT. Furthermore, it is evident from the investigation described in Chapter 2 that a number of researchers have proposed different approaches to address some of the issues in the field of trust management in IoT. However, at the time of writing, trust management approaches in the existing body of literature do not address key issues such as the scalability of the proposed IoT trust management approaches, approaches to counter and detect untrustworthy behaviour by the IoT nodes such as on-off attacks, bad-mouthing attacks etc. Moreover, as is evident from Chapter 2 and Chapter 3, the current literature has made significant advances in trust management for WSN and some focused partially on trust management solutions for IoT. At the same time however, the major shortcoming of the current approaches is that none of them present a comprehensive approach for trust management in IoT. The reason for this is that none of the proposed approaches provide an integrated and comprehensive solution that has:

(a) Intelligent and robust mechanisms to enable the scalability of the IoT trust management approaches
(b) Intelligent and robust approaches to counter and detect non-compliant behaviour. IoT nodes or entities can carry out cyber-security attacks as a part of their non-compliant behaviour.
(c) Experimental evaluation for the solutions mentioned in (a) and (b) above.

This thesis presents a trusted, lightweight and secure communication mechanism between IoT entities in the environment. In order to propose a comprehensive solution for a trust management platform for IoT, this thesis presents solutions for the aforementioned shortcomings in the literature.

The chapter is organized as follows: this section provides an overview of the chapter. The next section discusses the research issues related to trust management in IoT that have been addressed in this thesis. Based on these research issues, the contributions of

the thesis are subsequently summarized in Section 8.3. In Section 8.4, I identify the future research work and conclude this chapter.

8.2 PROBLEMS ADDRESSED IN THIS THESIS

This thesis aims to fill the critical gaps related to trust management for IoT in the existing literature body. Based on the literature review in Chapter 2, the research issues that were addressed in this thesis are summarized as follows:

1. Propose a comprehensive platform for trust management for IoT that includes intelligent approaches to ensure the scalability of the proposed trust management solution (TM-IoT)
2. Propose fuzzy-logic based and non-fuzzy logic based algorithms that will ensure the resilience and reliability of the TM-IoT against untrustworthy behaviour by the IoT nodes. In this thesis I have compared the performance of both the fuzzy-logic based algorithms with their non-fuzzy logic based counterparts to benchmark them. I have presented and evaluated approaches to detect bad-mouthing attacks, on-off attacks, contradictory attacks, and bad service attacks on the TM-IoT
3. Propose an IoT trust management platform that is able to carry out memory efficient trust management activities.

In order to address these issues, I propose a comprehensive solution for trust management (TM-IoT) in IoT that encompasses and addresses the aforementioned challenges.

8.3 Contributions of this Thesis to the Existing Literature

Based on the identified research issues, the principal contribution of this thesis to the existing literature is that it proposes a comprehensive trust management platform for IoT based on intelligent clustering techniques, taking into account the intelligent approaches for enabling the scalability of the proposed trust management solution and intelligent approaches to detect certain types of untrustworthy behaviour by the IoT nodes (to enhance the reliance and resilience of the proposed trust management solution). A brief overview of the five principal contributions of this thesis to address the gaps in the existing literature follows.

8.3.1 Contribution 1: State-of-the-art Comprehensive Survey of the Existing Literature

This thesis documents an extensive state-of-the-art survey of the existing literature in the areas of trust management in IoT which is documented in Chapter 2. To the best of the researcher's knowledge, the survey of the literature carried out in this research is most comprehensive and extensive in the existing literature so far.

For the purposes of discussion and evaluation, the existing literature was divided into the following five categories based on their features and working attributes:

1. Trust management and scalable approaches in the IoT.
2. Context-aware trust assessment for the IoT.

3. Security protocol for reliable trust management for IoT.
4. Clustering-Based Trust for IoT.
5. Fuzzy-logic based mechanisms for trust management in the IoT.

The salient features and shortcomings of each of these approaches were identified and discussed. Based on the extensive analysis of the existing literature, research gaps were identified with the existing approaches.

8.3.2 Contribution 2: Methodology of Trust Management Platform for IoT (TM-IoT)

As discussed in previous chapters, trust management in IoT is different to trust management in other types of networks, due to the heterogeneous nature of the IoT node (Ammar, Russello & Crispo 2018). The existing literature lacks a comprehensive platform for IoT trust management that has the attributes mentioned in Section 1 of this chapter. Therefore, before creating a comprehensive platform of trust management in IoT, a prescribed definition of trust and a trust management platform for IoT was proposed that focuses on facilitating lightweight and trustworthy communication among IoT nodes in the proposed platform, as presented in Chapter 4 of this thesis. The key elements of the comprehensive platform of trust management in IoT are as follows (TM-IoT):

1. Clustering: This is a technique that is used to intelligently group the IoT nodes based on their trust value. Clustering the IoT nodes is a key enabler for the scalability of the trust management approach.
2. Cluster Node (CN): A cluster node is a regular IoT node that interacts and communicates with other nodes in the IoT platform.

3. Master Node (MN): A master node is the main IoT node in each cluster and is responsible for the trust management activities in its cluster (CN nodes). Some of the activities that the MN carries out are monitoring and storing the trust values of the CNs, communication between the MN and the SN and updating the SN in relation to its cluster.
4. Super Node (SN): The super node is the central node responsible for the activities of the entire IoT trust management platform for IoT. It oversees and carries out the grouping of the IoT nodes in the IoT application into clusters and the selection of the MN for each cluster. Furthermore, it stores the trust values from the trust values from the MN and is also responsible for migrating the IoT nodes from one cluster to another based on their evolving or dynamic trust value. The process of selecting the MN and the SN is explained in Chapter 5. The process by which the trust updates are communicated from the MN to the SN has also been explained in Chapter 5.

The detailed working of the TM-IoT platform has been presented in Chapter 5. To the best of the researcher's knowledge, and also based on the experimentation results obtained in this thesis TM-IoT trust management platform for IoT contributes to both the scalability of the proposed trust management solution and also to its resilience against non-compliant behaviour by the IoT nodes (such as on-off attacks etc.) has not been explored in the existing literature.

8.3.3 Contribution 3: Clustering-Driven Intelligent, Scalable and Reliable Trust Management for IoT (CITM-IOT)

The third contribution of this thesis is that it proposes an intelligent clustering-based intelligent, scalable and reliable algorithms in TM-IoT. In CITM-IoT the proposed

algorithms are not based on fuzzy-logic. The clustering-based approach clusters the IoT nodes into groups based on their trust value. Furthermore, to ensure the resilience and robustness of TM-IoT against non-compliant or untrustworthy behaviour by the IoT nodes, I propose a series of algorithms to intelligently detect the IoT nodes which will carry out attacks, such as on-off attacks, bad-mouthing attacks etc. on the TM-IoT. I term the collection of all the algorithms for building resilience against the cyber-attacks in TM-IoT and also the algorithms to ensure scalability of the TM-IoT as *CITM-IoT*. To achieve the objective of this contribution, Chapter 6 discusses the clustering-based intelligent approach to achieve scalability in TM-IoT, and proposes four intelligent algorithms to ensure the reliability and resilience of the IoT against cyber-attacks as follows:

1. Algorithm 1 proposes a new mechanism of clustering in TM-IoT by calculating trust value boundaries for each cluster according to memory boundaries.
2. Algorithm 2 defines the conditions in which a cluster node is able to change to a specified new master node in TM-IoT.
3. Algorithm 3 is used to address the issue of bad-mouthing of IoT nodes. It does so by eliminating the bad-mouthed values (outliers) of a set of floats using Tukey's fences. This algorithm is a proposed solution for extreme bad-mouthing attacks in TM-IoT. Furthermore, this algorithm proposes approaches to intelligently detect the intentional bad delivery of service attacks.
4. Algorithm 4 proposes methods by which master nodes can monitor cluster node trust values and try to move some cluster nodes away. A check is then made on the memory of the current master node to decide whether further cluster nodes need to be removed.

To the best of the researcher’s knowledge, there is no holistic approach in the existing literature that provides a trust management approach for IoT that is both scalable and resilient against the aforementioned cyber-attacks on the trust management approaches.

8.3.4 CONTRIBUTION 4: FUZZY LOGIC-BASED ALGORITHMS FOR THE RELIABILITY OF TM-IOT (FUZZY-IOT)

The fourth contribution of this thesis is that it proposes an intelligent fuzzy-logic based algorithms for clustering and ensuring reliability of TM-IoT. Chapter 7 details the working of a suite of fuzzy logic-based algorithms, which I term as Fuzzy-IoT, to ensure both the scalability and the reliability of TM-IoT. Fuzzy-IoT comprises a sequence of fuzzy logic-based algorithms to intelligently detect any untrusted function of the IoT nodes. In particular, it comprises algorithms to counter bad-mouthing attacks, on-off attacks, contradictory attacks and bad service attacks. Furthermore, I propose a secure messaging protocol as part of the trust management methodology to ensure the integrity of the transmitted messages in TM-IoT. The messaging system provides a mechanism to ensure that the messages in transit are not tampered with whilst they are in transit. To address this gap, I propose a serial communication for secure message encryption in the IoT network.

To the best of the researcher's knowledge, experiments which use a huge number of IoT nodes to ensure scalability and reliability have not been performed before and there is no existing literature that provides a reliable IoT protocol with fuzzy logic techniques to ensure both the security and scalability of the entire IoT environment.

8.3.5 Contribution 5: Evaluation of the Proposed Trust Management Platform for IoT

To evaluate the applicability and effectiveness of the proposed trust management platform for IoT, in Chapters 5, 6 and 7, each component of the trust management platform for IoT was evaluated and tested using the trust metrics.

To build the prototype platform, I use Cooja (IoT simulator tool), Java and C++. Chapters 6 and 7 present the detailed implementation of the developed prototype to demonstrate the overall working of the TM-IoT platform. Furthermore, it also evaluates the developed prototype using benchmarks or metrics. The detailed experimental results can be found in Chapters 5, 6 and 7. Here were present some key findings that I obtained:

(a) I found that TM-IoT is able to scale very well with increasing population of nodes. I repeated the experiments with various population sizes of the IoT nodes (20, 200 and 2000) and I found no deterioration in the performance of TM-IoT.

(b) I found our proposed approach for trust-based clustering is effective in grouping the IoT nodes based on their trust value for ensuring IoT nodes scalability and reliability

(c) I found that TM-IoT is memory efficient in terms of the memory requirements from the Master Nodes for trust management.

(d) I found that our fuzzy-logic based approach for countering bad-mouthing attacks is better than the non-fuzzy approach

(e) I found that our fuzzy-logic based approach for detecting on-off behaviour attacks is better than the non-fuzzy approach

(f) I found that our fuzzy-logic based approach for detecting contradictory behaviour attacks is better than the non-fuzzy approach

(g) I found that our proposed fuzzy-logic based approach is able to intelligently detect nodes that engage in bad service attacks.

8.4 Conclusion and Future Work

The work that was undertaken in this thesis has been published extensively as a part of the proceedings in peer-reviewed international conferences and journals. At the time of writing this thesis, one JCR Q1 journal article with impact factor (3.25) has been published, another ERA-CORE A journal article has been accepted, one ERA-CORE A conference paper has been published, two conference papers have been published in prestigious conferences and four conference papers have been published and accepted with other authors in the same field of IoT. The list of publications arising as a result of the work documented in this thesis can be found at the beginning of the thesis.

Although I have undertaken a lot of research on the topic of this study, the researcher feels there are still several future directions that would strengthen the proposed trust management platform for IoT. It is our intention to continue working on this topic, primarily along, but not limited to, the following lines:

1. The development of intelligent techniques which enable new reliable IoT entities to bootstrap themselves for the number of IoT services
2. To apply the proposed TM-IoT platform in a real domain, such as healthcare systems, transportation systems, smart cities and buildings, smart banking systems and so on.
3. To develop intelligent and memory-efficient Artificial Intelligence (AI)-based methods that can be deployed on SN and MN to carry out trust prediction, context-based trust assessments, advanced cyber security assessments etc… for the CNs. These areas have not been explored in this thesis and remain an avenue for further research.

REFERENCES

Abedin, S.F., Alam, M.G.R., Haw, R. & Hong, C.S. 2015, 'A system model for energy efficient green-IoT network', *Information Networking (ICOIN), 2015 International Conference on*, IEEE, pp. 177-82.

Ahmed, A., Bakar, K.A., Channa, M.I. & Khan, A.W. 2016, 'A secure routing protocol with trust and energy awareness for wireless sensor network', *Mobile Networks and Applications*, vol. 21, no. 2, pp. 272-85.

Al-Fuqaha, A., Guizani, M., Mohammadi, M., Aledhari, M. & Ayyash, M. 2015, 'Internet of things: A survey on enabling technologies, protocols, and applications', *IEEE Communications Surveys & Tutorials*, vol. 17, no. 4, pp. 2347-76.

Alexopoulos, N., Habib, S.M. & Mühlhäuser, M. 2018, 'Towards Secure Distributed Trust Management on a Global Scale: An analytical approach for applying Distributed Ledgers for authorization in the IoT', *Proceedings of the 2018 Workshop on IoT Security and Privacy*, ACM, pp. 49-54.

Alshehri, M.D. & Hussain, F.K. 2015, 'A Comparative Analysis of Scalable and Context-Aware Trust Management Approaches for Internet of Things', *International Conference on Neural Information Processing*, Springer, pp. 596-605.

Alshehri, M.D. & Hussain, F.K. 2017, 'A Centralized Trust Management Mechanism for the Internet of Things (CTM-IoT)', *International Conference on Broadband and Wireless Computing, Communication and Applications*, Springer, pp. 533-43.

Alshehri, M.D., Hussain, F.K. & Hussain, O.K. 2018, 'Clustering-Driven Intelligent Trust Management Methodology for the Internet of Things (CITM-IoT)', *Mobile Networks and Applications*, pp. 1-13.

Ammar, M., Russello, G. & Crispo, B. 2018, 'Internet of Things: A survey on the security of IoT frameworks', *Journal of Information Security and Applications*, vol. 38, pp. 8-27.

Babar, S., Stango, A., Prasad, N., Sen, J. & Prasad, R. 2011, 'Proposed embedded security framework for internet of things (iot)', *Wireless Communication, Vehicular Technology, Information Theory and Aerospace & Electronic Systems Technology (Wireless VITAE), 2011 2nd International Conference on*, IEEE, pp. 1-5.

Banković, Z., Vallejo, J.C., Fraga, D. & Moya, J.M. 2011, 'Detecting bad-mouthing attacks on reputation systems using self-organizing maps', *Computational Intelligence in Security for Information Systems*, Springer, pp. 9-16.

Bao, F. & Chen, I.-R. 2012a, 'Dynamic trust management for internet of things applications', *Proceedings of the 2012 international workshop on Self-aware internet of things*, ACM, pp. 1-6.

Bao, F. & Chen, R. 2012b, 'Trust Management for the Internet of Things and its Application to Service Composition', *IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, IEEE, pp. 1-6.

Bao, F., Chen, R. & Guo, J. 2013, 'Scalable, adaptive and survivable trust management for community of interest based internet of things systems', *Autonomous Decentralized Systems (ISADS), 2013 IEEE Eleventh International Symposium on*, IEEE, pp. 1-7.

Bellavista, P. & Zanni, A. 2016, 'Towards better scalability for IoT-cloud interactions via combined exploitation of MQTT and CoAP', *Research and Technologies for Society and Industry Leveraging a better tomorrow (RTSI), 2016 IEEE 2nd International Forum on*, IEEE, pp. 1-6.

Bello, O. & Zeadally, S. 2016, 'Intelligent device-to-device communication in the internet of things', *IEEE Systems Journal*, vol. 10, no. 3, pp. 1172-82.

Bělohlávek, R. & Klir, G.J. 2011, *Concepts and fuzzy logic*, MIT Press.

Boswarthick, D., Elloumi, O. & Hersent, O. 2012, *M2M communications: a systems approach*, John Wiley & Sons.

Burstein, F. & Gregor, S. 1999, 'The systems development or engineering approach to research in information systems: An action research perspective', *Proceedings of the 10th Australasian Conference on Information Systems*, Victoria University of Wellington, New Zealand, pp. 122-34.

Castellani, A.P., Gheda, M., Bui, N., Rossi, M. & Zorzi, M. 2011, 'Web Services for the Internet of Things through CoAP and EXI', *Communications Workshops (ICC), 2011 IEEE International Conference on*, IEEE, pp. 1-6.

Chae, Y., DiPippo, L.C. & Sun, Y.L. 2015, 'Trust management for defending on-off attacks', *IEEE Transactions on Parallel and Distributed Systems*, vol. 26, no. 4, pp. 1178-91.

Chang, K.-D. & Chen, J.-L. 2012, 'A survey of trust management in WSNs, internet of things and future internet', *KSII Transactions on Internet & Information Systems*, vol. 6, no. 1.

Chen, C. & Helal, S. 2011, 'A Device-Centric Approach to a Safer Internet of Things', *International workshop on networking and object memories for the internet of things (NOMe-IoT)*, ACM, pp. 1-6.

Chen, D., Chang, G., Sun, D., Li, J., Jia, J. & Wang, X. 2011, 'TRM-IoT: A Trust Management Model Based on Fuzzy Reputation for Internet of Things', *Computer Science and Information Systems*, vol. 8, no. 4, pp. 1207-28.

Chen, R., Guo, J. & Bao, F. 2016, 'Trust Management for SOA-Based IoT and its Application to Service Composition', *IEEE Transactions on Services Computing*, vol. 9, no. 3, pp. 482-95.

Contiki 2018, *Contiki IoT simulator tool (Cooja)* viewed 02/01/2017 2017, <http://www.contiki-os.org/start.html>.

Di Francesco, M., Li, N., Raj, M. & Das, S.K. 2012, 'A storage infrastructure for heterogeneous and multimedia data in the internet of things', *2012 IEEE International Conference on Green Computing and Communications*, IEEE, pp. 26-33.

Evans, D. 2012, 'The internet of everything: How more relevant and valuable connections will change the world', *Cisco IBSG*, vol. 2012, pp. 1-9.

Fenye, B. 2012, 'Dynamic Trust Management for Internet of Things Applications', *International Workshop on Self-Aware Internet of Things (Self-IoT'12)*, ACM.

Galliers, R.D. 1990, 'Choosing appropriate information systems research approaches: a revised taxonomy', *In Proceedings of the IFIP TC8 WG8. 2*, Citeseer.

Gharbieh, M., ElSawy, H., Bader, A. & Alouini, M.-S. 2017, 'Spatiotemporal stochastic modeling of IoT enabled cellular networks: Scalability and stability analysis', *IEEE Transactions on Communications*.

Goldhaber, A.S. & Nieto, M.M. 2010, 'Photon and graviton mass limits', *Reviews of Modern Physics*, vol. 82, no. 1, p. 939.

Granjal, J., Monteiro, E. & Silva, J.S. 2015, 'Security for the internet of things: a survey of existing protocols and open research issues', *IEEE Communications Surveys & Tutorials*, vol. 17, no. 3, pp. 1294-312.

Gubbi, J., Buyya, R., Marusic, S. & Palaniswami, M. 2013, 'Internet of Things (IoT): A vision, architectural elements, and future directions', *Future generation computer systems*, vol. 29, no. 7, pp. 1645-60.

Guo, J., Chen, R. & Tsai, J.J. 2017, 'A survey of trust computation models for service management in internet of things systems', *Computer Communications*, vol. 97, pp. 1-14.

Gupta, A. & Awasthi, L.K. 2010, 'Toward a quality-of-service framework for peer-to-peer applications', *International Journal of Distributed Systems and Technologies (IJDST)*, vol. 1, no. 3, pp. 1-23.

Hamadeh, H., Chaudhuri, S. & Tyagi, A. 2017, 'Area, energy, and time assessment for a distributed TPM for distributed trust in IoT clusters', *Integration, the VLSI Journal*, vol. 58, pp. 267-73.

Han, S. & Woo, H. 2016, 'NDN-Based Pub/Sub System for Scalable IoT Cloud', *Cloud Computing Technology and Science (CloudCom), 2016 IEEE International Conference on*, IEEE, pp. 488-91.

Höller, J., Boyle, D., Karnouskos, S., Avesand, S., Mulligan, C. & Tsiatsis, V. 2014, *From machine-to-machine to the internet of things*, Elsevier.

Hossain, M.M., Fotouhi, M. & Hasan, R. 2015, 'Towards an analysis of security issues, challenges, and open problems in the internet of things', *Services (SERVICES), 2015 IEEE World Congress on*, IEEE, pp. 21-8.

Intelligence, B. 2015, 'The Internet of Everything. Available online'.

Jabeur, N., Yasar, A.U.-H., Shakshuki, E. & Haddad, H. 2017, 'Toward a bio-inspired adaptive spatial clustering approach for IoT applications', *Future Generation Computer Systems*.

Jiang, H., Shen, F., Chen, S., Li, K.-C. & Jeong, Y.-S. 2015, 'A secure and scalable storage system for aggregate data in IoT', *Future Generation Computer Systems*, vol. 49, pp. 133-41.

Kaplan, B. & Maxwell, J.A. 2005, 'Qualitative research methods for evaluating computer information systems', *Evaluating the organizational impact of healthcare information systems*, Springer, pp. 30-55.

Khan, Z.A. & Herrmann, P. 2017, 'A Trust Based Distributed Intrusion Detection Mechanism for Internet of Things', *Advanced Information Networking and Applications (AINA), 2017 IEEE 31st International Conference on*, IEEE, pp. 1169-76.

Kokoris-Kogias, E., Voutyras, O. & Varvarigou, T. 2016, 'TRM-SIoT: A scalable hybrid trust & reputation model for the social Internet of Things', *Emerging Technologies and Factory Automation (ETFA), 2016 IEEE 21st International Conference on*, Ieee, pp. 1-9.

Kotis, K., Athanasakis, I. & Vouros, G.A. 2018, 'Semantically enabling IoT trust to ensure and secure deployment of IoT entities', *International Journal of Internet of Things and Cyber-Assurance*, vol. 1, no. 1, pp. 3-21.

Lee, I. & Lee, K. 2015, 'The Internet of Things (IoT): Applications, investments, and challenges for enterprises', *Business Horizons*, vol. 58, no. 4, pp. 431-40.

Li, F., Vögler, M., Claeßens, M. & Dustdar, S. 2013, 'Efficient and scalable IoT service delivery on cloud', *Cloud Computing (CLOUD), 2013 IEEE Sixth International Conference on*, IEEE, pp. 740-7.

Li, X., Lu, R., Liang, X., Shen, X., Chen, J. & Lin, X. 2011, 'Smart Community: An Internet of Things Application', *IEEE Communications Magazine*, vol. 49, no. 11.

Lin, J., Yu, W., Zhang, N., Yang, X., Zhang, H. & Zhao, W. 2017, 'A survey on internet of things: Architecture, enabling technologies, security and privacy, and applications', *IEEE Internet of Things Journal*, vol. 4, no. 5, pp. 1125-42.

Lize, G., Jingpei, W. & Bin, S. 2014, 'Trust management mechanism for internet of things', *China Communications*, vol. 11, no. 2, pp. 148-56.

Lund, D., MacGillivray, C., Turner, V. & Morales, M. 2014, 'Worldwide and regional internet of things (iot) 2014–2020 forecast: A virtuous circle of proven value and demand', *International Data Corporation (IDC), Tech. Rep*, vol. 1.

Lyu, S., Liu, J., Tang, M., Xu, Y. & Chen, J. 2015, 'Efficiently predicting trustworthiness of mobile services based on trust propagation in social networks', *Mobile Networks and Applications*, vol. 20, no. 6, pp. 840-52.

Ma, T., Liu, Y. & Zhang, Z.-j. 2015, 'An energy-efficient reliable trust-based data aggregation protocol for wireless sensor networks', *Int. J. Control Autom*, vol. 8, no. 3, pp. 305-18.

Mahalle, P.N., Thakre, P.A., Prasad, N.R. & Prasad, R. 2013, 'A fuzzy approach to trust based access control in internet of things', *Wireless Communications, Vehicular Technology, Information Theory and Aerospace & Electronic Systems (VITAE), 2013 3rd International Conference on*, IEEE, pp. 1-5.

Mainetti, L., Patrono, L. & Vilei, A. 2011, 'Evolution of wireless sensor networks towards the internet of things: A survey', *Software, Telecommunications and Computer Networks (SoftCOM), 2011 19th International Conference on*, IEEE, pp. 1-6.

Malina, L., Hajny, J., Fujdiak, R. & Hosek, J. 2016, 'On perspective of security and privacy-preserving solutions in the internet of things', *Computer Networks*, vol. 102, pp. 83-95.

Miao, J. & Wang, L. 2012, 'Rapid Identification Authentication Protocol for Mobile Nodes in Internet of Things with Privacy Protection', *JNW*, vol. 7, no. 7, pp. 1099-105.

Miorandi, D., Sicari, S., De Pellegrini, F. & Chlamtac, I. 2012, 'Internet of things: Vision, applications and research challenges', *Ad hoc networks*, vol. 10, no. 7, pp. 1497-516.

Mosenia, A. & Jha, N.K. 2017, 'A comprehensive study of security of internet-of-things', *IEEE Transactions on Emerging Topics in Computing*, vol. 5, no. 4, pp. 586-602.

Naik, S. & Maral, V. 2017, 'Cyber security—IoT', *Recent Trends in Electronics, Information & Communication Technology (RTEICT), 2017 2nd IEEE International Conference on*, IEEE, pp. 764-7.

Neisse, R., Steri, G., Baldini, G., Tragos, E., Fovino, I.N. & Botterman, M. 2014, 'Dynamic context-aware scalable and trust-based IoT security, privacy framework', *Chapter in Internet of Things Applications-From Research and Innovation to Market Deployment, IERC Cluster Book*.

Neisse, R., Steri, G., Fovino, I.N. & Baldini, G. 2015, 'SecKit: a model-based security toolkit for the internet of things', *Computers & Security*, vol. 54, pp. 60-76.

Nguyen, K.T., Laurent, M. & Oualha, N. 2015, 'Survey on secure communication protocols for the Internet of Things', *Ad Hoc Networks*, vol. 32, pp. 17-31.

Ning, H. 2016, *Unit and ubiquitous internet of things*, CRC press.

Ortiz, A.M., Hussein, D., Park, S., Han, S.N. & Crespi, N. 2014, 'The cluster between internet of things and social networks: Review and research challenges', *IEEE Internet of Things Journal*, vol. 1, no. 3, pp. 206-15.

Parwekar, P. 2011, 'From internet of things towards cloud of things', *Computer and Communication Technology (ICCCT), 2011 2nd International Conference on*, IEEE, pp. 329-33.

Pasqualetti, F., Dörfler, F. & Bullo, F. 2011, 'Cyber-physical attacks in power networks: Models, fundamental limitations and monitor design', *Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on*, IEEE, pp. 2195-201.

Peffers, K., Tuunanen, T., Rothenberger, M.A. & Chatterjee, S. 2007, 'A design science research methodology for information systems research', *Journal of management information systems*, vol. 24, no. 3, pp. 45-77.

Peña-López, I. 2005, 'ITU Internet report 2005: the internet of things'.

Perera, C., Zaslavsky, A., Christen, P. & Georgakopoulos, D. 2012, 'Ca4iot: Context awareness for internet of things', *Green Computing and Communications (GreenCom), 2012 IEEE International Conference on*, IEEE, pp. 775-82.

Pfister, C. 2011, *Getting started with the Internet of Things: connecting sensors and microcontrollers to the cloud*, " O'Reilly Media, Inc.".

Ray, B.R., Abawajy, J. & Chowdhury, M. 2014, 'Scalable RFID security framework and protocol supporting Internet of Things', *Computer Networks*, vol. 67, pp. 89-103.

Raza, S., Duquennoy, S., Höglund, J., Roedig, U. & Voigt, T. 2014, 'Secure communication for the Internet of Things—a comparison of link-layer security and IPsec for 6LoWPAN', *Security and Communication Networks*, vol. 7, no. 12, pp. 2654-68.

Ren, W. 2011, 'QoS-Aware and Compromise-Resilient Key Management Scheme for Heterogeneous Wireless Internet of Things', *International Journal of Network Management*, vol. 21, no. 4, pp. 284-99.

Renubala, S. & Dhanalakshmi, K. 2014, 'Trust based secure routing protocol using fuzzy logic in wireless sensor networks', *Computational Intelligence and Computing Research (ICCIC), 2014 IEEE International Conference on*, IEEE, pp. 1-5.

Roman, R., Najera, P. & Lopez, J. 2011, 'Securing the Internet of Things', *Computer*, vol. 44, no. 9, pp. 51-8.

Saied, Y.B., Olivereau, A., Zeghlache, D. & Laurent, M. 2013, 'Trust management system design for the Internet of Things: A context-aware and multi-service approach', *Computers & Security*, vol. 39, pp. 351-65.

Sarkar, C., SN, A.U.N., Prasad, R.V., Rahim, A., Neisse, R. & Baldini, G. 2015, 'DIAT: A scalable distributed architecture for IoT', *IEEE Internet of Things journal*, vol. 2, no. 3, pp. 230-9.

Sarobin, V.R. & Ganesan, R. 2016, 'Bio-inspired, Cluster-based Deterministic Node Deployment in Wireless Sensor Networks', *International Journal of Technology* vol. 7, no. 4, pp. 673-82.

Shifeng, Y., Chungui, F., Yuanyuan, H. & Shiping, Z. 2011, 'Application of IOT in agriculture', *Journal of Agricultural Mechanization Research*, vol. 7, pp. 190-3.

Sicari, S., Rizzardi, A., Grieco, L.A. & Coen-Porisini, A. 2015, 'Security, privacy and trust in Internet of Things: The road ahead', *Computer Networks*, vol. 76, pp. 146-64.

Sirisala, N. & Bindu, C.S. 2015, 'Uncertain rule based fuzzy logic QoS trust model in manets', *Advanced Computing and Communications (ADCOM), 2015 International Conference on*, IEEE, pp. 55-60.

Talwar, S., Choudhury, D., Dimou, K., Aryafar, E., Bangerter, B. & Stewart, K. 2014, 'Enabling technologies and architectures for 5G wireless', *Microwave Symposium (IMS), 2014 IEEE MTT-S International*, IEEE, pp. 1-4.

Tselentis, G., Domingue, J. & Galis, A. 2009, *Towards the future internet: A European research perspective*, IOS press.

Tuna, G., Kogias, D.G., Gungor, V.C., Gezer, C., Taşkın, E. & Ayday, E. 2017, 'A survey on information security threats and solutions for Machine to Machine (M2M) communications', *Journal of Parallel and Distributed Computing*, vol. 109, pp. 142-54.

Uckelmann, D., Harrison, M. & Michahelles, F. 2011, 'An architectural approach towards the future internet of things', *Architecting the internet of things*, Springer, pp. 1-24.

Varghese, R., Chithralekha, T. & Kharkongor, C. 2016, 'Self-organized cluster based energy efficient meta trust model for internet of things', *Engineering and Technology (ICETECH), 2016 IEEE International Conference on*, IEEE, pp. 382-9.

Vögler, M., Schleicher, J.M., Inzinger, C. & Dustdar, S. 2016, 'A scalable framework for provisioning large-scale IoT deployments', *ACM Transactions on Internet Technology (TOIT)*, vol. 16, no. 2, p. 11.

Wang, X., Zhou, H., Su, J., Wang, B., Xing, Q. & Li, P. 2018, 'T-IP: A self-trustworthy and secure Internet protocol', *China Communications*, vol. 15, no. 2, pp. 1-14.

Xia, F., Yang, L.T., Wang, L. & Vinel, A. 2012, 'Internet of things', *International Journal of Communication Systems*, vol. 25, no. 9, pp. 1101-2.

Yan, Z., Zhang, P. & Vasilakos, A.V. 2014, 'A survey on trust management for Internet of Things', *Journal of network and computer applications*, vol. 42, pp. 120-34.

Yaqoob, I., Ahmed, E., Hashem, I.A.T., Ahmed, A.I.A., Gani, A., Imran, M. & Guizani, M. 2017, 'Internet of things architecture: Recent advances, taxonomy, requirements, and open challenges', *IEEE Wireless Communications*, vol. 24, no. 3, pp. 10-6.

Yu, Y., Jia, Z., Tao, W., Xue, B. & Lee, C. 2017, 'An efficient trust evaluation scheme for node behavior detection in the internet of things', *Wireless Personal Communications*, vol. 93, no. 2, pp. 571-87.

Zanella, A., Bui, N., Castellani, A., Vangelista, L. & Zorzi, M. 2014, 'Internet of things for smart cities', *IEEE Internet of Things journal*, vol. 1, no. 1, pp. 22-32.

Zhou, L. & Chao, H.-C. 2011, 'Multimedia Traffic Security Architecture for The Internet of Things', *IEEE Network*, vol. 25, no. 3.

Zorzi, M., Gluhak, A., Lange, S. & Bassi, A. 2010, 'From today's intranet of things to a future internet of things: a wireless-and mobility-related view', *IEEE Wireless communications*, vol. 17, no. 6.