BOURNEMOUTH UNIVERSITY

Intrusion Detection in IPv6-enabled Sensor Networks

by

Mohammed Al Qurashi

A thesis submitted in partial fulfillment for the
degree of Doctor of Philosophy

in the
FACULTY OF SCIENCE AND TECHNOLOGY
COMPUTING AND INFORMATICS

September 2020

Declaration of Authorship

I, Mohammed Al Qurashi, declare that this thesis titled, 'Intrusion Detection in IPv6-Enabled Sensor Networks' and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed:

Date:

"The most valuable contribution of intellectuals to international reconciliation and to the lasting fraternity of man lies in their scientific and artistic creations, because these elevate the human spirit above personal and selfish nationalistic aims"

Albert Einstein

BOURNEMOUTH UNIVERSITY

Abstract

FACULTY OF SCIENCE AND TECHNOLOGY

COMPUTING AND INFORMATICS

Doctor of Philosophy

by Mohammed Al Qurashi

In this research, we study efficient and lightweight Intrusion Detection Systems (IDS) for ad-hoc networks through the lens of IPv6-enabled Wireless Sensor Actuator Networks. These networks consist of highly constrained devices able to communicate wirelessly in an ad-hoc fashion, thus following the architecture of ad-hoc networks. Current state of the art IDS in IoT and WSNs have been developed considering the architecture of conventional computer networks, and as such they do not efficiently address the paradigm of ad-hoc networks, which is highly relevant in emerging network paradigms, such as the Internet of Things (IoT). In this context, the network properties of resilience and redundancy have not been extensively studied.

In this thesis, we first identify a trade-off between the communication and energy overheads of an IDS (as captured by the number of active IDS agents in the network) and the performance of the system in terms of successfully identifying attacks. In order to fine-tune this trade-off, we model networks as Random Geometric Graphs; these are a rigorous approach that allows us to capture underlying structural properties of the network. We then introduce a novel IDS architectural approach that consists of a central IDS agent and set of distributed IDS agents deployed uniformly at random over the network area. These nodes are able to efficiently detect attacks at the networking layer in a collaborative manner by monitoring locally available network information provided by IoT routing protocols, such as RPL.

The detailed experimental evaluation conducted in this research demonstrates significant performance gains in terms of communication overhead and energy dissipation while maintaining high detection rates. We also show that the performance of our IDS in ad-hoc networks does not rely on the size of the network but on fundamental underling network properties, such as the network topology and the average degree of the nodes. The experiments show that our proposed IDS architecture is resilient against frequent topology changes due to node failures.

Acknowledgements

First and foremost, I would like to thank Almighty God for giving me the opportunity, ability, and knowledge to undertake this research study and to persevere and complete it satisfactorily.

I am grateful to my supervisors Prof.Vasilis Katos and Dr.Marios Angelopoulos, for motivating me along the whole project and for continued support and guidance throughout this research, which have made a valuable contribution to this thesis.

I also wish to thank Eng.Zeyad Alshaikh, for his excellent technical support.

Finally, I wish to thank my family: my parents, wife, parent in law, and siblings, for continuous and unparalleled love, help and support.

Contents

List of Figures

List of Tables

6LoWPAN	IPv6 for Low-Power Wireless Personal Area Network
AIDS	Anomaly based Intrusion Detection System
CoAP	Constrained Application Protocol
CTP	Collection Tree Protocol
CH	Cluster Head
DAG	Directed Acyclic Graph
DODAG	Destination Oriented DAG
DIO	DODAG Information Object
ETX	Expected Number of Transmissions
HTTP	Hypertext Transfer Protocol
HIDS	Host based Intrusion Detection System
IoT	Internet of Things
IAT	Inter-Arrival Time
ICPMv6	Internet Control Massage protocol v6
IETF	Internet Engineering Task Force
IDS	Intrusion Detection System
LLNs	low power and lossy networks
MP2P	Multipoint-to-Point
NIDS	Network based Intrusion Detection System
OSI	Open Systems Interconnection model
PRR	Packet Reception Rate
P2P	Point-to-Point

QoS	Quality of Service
RPL	Routing protocol for low power and lossy networks
RGG	Random Geometric Graphs
SN	Sensor Nodes
SIDS	Signature based Intrusion Detection System
WSN	Wireless Sensor Network
WSAN	Wireless Sensor Actuator Network
UDP	User Datagram Protocol

Dedicated to my family:
to my parents, wife and siblings.

Chapter 1

Introduction

1.1 Motivation

Internet of Things (IoT) represents a future networking paradigm that enables computers and people, *things* and *machines* to seamlessly exchange information and data over the Internet. It is estimated that by 2025 around 55 billion IoT devices will be deployed and more than 15 USD trillion will be invested in IoT in aggregate between 2017 and 2025 (Liu 2020). IoT has already been deployed in many sectors and environments such as healthcare, manufacturing and critical infrastructure (Li et al. 2015) and so forth. However, as promising as IoT seems in addressing and enabling business cases that shape the modern economy, there are observed systematic and increasing attacks on the underlying infrastructure. The boom of IoT development and adoption seems to cause a lack of security considerations. As sush, IoT infrastructures have been targeted by malicious actors, such as Mirai botnet (Kolias et al. 2017) or the case where hackers were able to affect the steering and braking systems of a Jeep car (Ring 2015). As such, there is an ongoing, topical and continuously maturing research activity in security for IoT ecosystems.

In this context, security controls can be grouped in three category types, namely preventative, detective, and recovery. Preventative controls include authentication and access control mechanisms, cryptography , firewalls, etc. Detection countermeasures refer to those that are engaged *during* an attack, such as Intrusion Detection Systems. Finally recovery measures and processes focus in post-incident management, such as security information incident management and digital forensics. Due to the inherent and particular characteristics of IoT (i.e. highly constrained, deployed in mass numbers and their ephemeral availability), the existing cyber security relating to conventional networks may not be applicable. One of the main reasons is that many existing security controls may require relatively intense computational resources and network connectivity. While such assumptions are guaranteed in "traditional" networks, they are not always met in IoT environments. Therefore the feasibility of the current security approaches needs to be revisited.

Particularly, in the domain of Intrusion Detection Systems (IDS) in IoT, a considerable amount of research has been carried out concerning deployment architectures, detection strategies and algorithms. However, available IDS in IoT are designed based on assumptions holding from “conventional” computer networks, e.g. that each node of the network is assumed to be adequately powerful in terms of resources (such as available energy, memory, CPU, etc.). Moreover, it is assumed that connectivity is always offered and the nodes can communicate over a reliable and high-capacity network. As such, the need of an IDS compatible with the IoT paradigm is evident.

A wireless sensor Actuator network (WSAN) consists of a set of nodes called sensors deployed over an area of interest. The devices communicate over the air with their peers collaboratively carrying out complex tasks. WSNs are a key enabling technology for the IoT and as such share several common characteristics. Therefore, WSNs have provided an ideal R&D platform to study several IoT protocols and technologies, such as the CoAP (Shelby et al. 2014), 6LoWPAN (Shelby and Bormann 2011).

Over the past decade, Intrusion Detection Systems for WSNs and the IoT have attracted significant research interest. IDSs can be classified based on their architecture, namely centralised, distributed or hybrid. In centralised IDSs, the detection algorithms are executed and performed on a designated node (host). The underlying monitoring and detection data have to be reported to a centrally located base station which is assumed to be powerful in terms of processing capabilities and available memory and energy. In a distributed architecture on the other hand, each individual network node runs an IDS agent cooperatively with other agents in the network. Finally,hybrid IDS architectures demonstrate a combination of the centralised and distributed architectures in an effort to avoid the disadvantages of each individual approach.

The current state-of-the-art on IDSs for WSN and IoT networks are still resource-intensive, as it is primarily stemming from assumptions holding from conventional computer networks. Centralised IDS architectures introduce significant communication overhead to the network as the base station(or Sink) due to large numbers of requests to and from the nodes related to IDS data collection. Distributed IDS architectures rely on the cooperation among the sensor nodes, thus increasing the communication load as well as energy dissipation. Lastly, hybrid IDS architectures achieve a better control and global overview of the network, but currently available solutions also introduce a significant communication overhead that increases proportionally to the number of network nodes. Furthermore, the resilience that refers to the ability of the network to recover its IDS architecture and capabilities have not been studied yet.

In this work we focus on hybrid IDS architectures but we show that by taking into account the specifics of IoT protocols, such as the ranking mechanism of RPL, as well as the spatial characteristics of such networks, the number of required IDS agents in the network (and therefore the corresponding overhead) can be greatly reduced while maintaining sufficiently high detection rates.

1.2 Aims of the Thesis

The aim of this research is to study and propose a novel IDS architecture that takes into consideration the particular characteristics and limitations of the IoT and Wireless Sensor Network paradigms. The proposed approach addresses the following characteristics of IoT and WSN: their highly distributed nature; their ad-hoc network structure; the peer-to-peer communication scheme among the devices; the highly constrained nature of the devices per se in terms of resources (computational power, available energy, limited memory). Sensor nodes in WSNs are constrained and limited in energy resource, which is related to the manufacturer and sensor nodes are usually powered by two AA batteries. And also, these sensor nodes are equipped with a few kilobytes memory and microprocessor. Furthermore, frequent changing of the network topology due to node communication failures and potentially changing network typology. Therefore, proposed IDS should consider auto-configurability to cope with the dynamic nature of WSNs and mitigate the effects of nodes failures on the efficiency of the IDS. Random Geometric Graphs (RGG) are used to design proposed IDS architecture as they are a well-studied model and a paradigm for wireless networks, such as sensor networks. The RGG model efficiently captures spatial characteristics of the network, and also adhere to different network typologies (e.g. mesh and ad-hoc networks). The WSNs are modelled as motes which in turn are represented as vertices in RGG, whereas the communication between these motes is represented by the edges. We study and consider a hybrid architecture of an Intrusion Detection System in Ad-hoc and wireless networks that combines centralised and distributed detection models. This combination is envisaged to reduce the massive communication overheads where the distributed IDS monitors a set of cluster member nodes activities through IDS agents to detect anomalies and malicious nodes and share information with a central IDS element. The latter is typically located at the so called *base station* or *sink*, in an WSN architecture. Moreover, this combination claims the advantages of both centralised and distributed detection models where centralised model provides better control on the detection processes than the distributed model, but the latter can offer better scalability, as well as communication overhead reduction. The distributed model is based on a small set of sensor nodes working as a local sink and monitor the neighbour nodes in its 1-hop or cluster. Besides, we consider a dynamic distribution and randomise the placement of the IDS agents among the various sensor nodes.

1.2.1 Research Questions

Based on conducted literature review on Intrusion Detection System in Wireless Sensor Networks, the following questions were derived:

- **RQ1:** What mechanisms/protocols/models are suitable for resource monitoring in WSN Environments?

- **RQ2:** How to reduce the communication overhead and energy consumption induced by the IDS procedure?
- **RQ3:** How to implement effective and efficient real-time IDS in WSN?

1.2.2 Research Objectives

- **Objective 1:** Review of the current state of the art IDS with particular emphasis on using IDS in WSN.
- **Objective 2:** Define new evaluation metrics for WSN-IDS efficiency. In particular, define the metrics that capture the trade-off between energy consumption /communication overhead with detection rate of events.
- **Objective 3:** Define a new WSN- IDS model/ architecture that properly addresses the distributed nature of WSN.
- **Objective 4** Develop efficient protocols/methods/algorithms for WSN-IDS based on objectives 2, 3.
- **Objective 5:** Develop a proof-of concept WSN- IDS testbed for experimentally evaluating objective 4.

1.3 Research Methodology

1.3.1 Literature Review

A systematic study of the literature was conducted to support the development of a conceptual and theoretical framework suitable for this study.

The review also encompassed the current state-of-the-art on Intrusion Detection System in Wireless Sensor Networks. The security aspects and challenges of wireless sensor networks, are presented along with the networking standard protocols and a summary of the related work in the fields of intrusion detection in WSNs.

The literature review also included Wireless Sensor Networks to analyse and capture the nature of tiny devices and their security challenges. Furthermore, the standardised wireless network protocols and their security weakness were also reviewed. Last but not least, the key contributions for IDS proposed for WSN and

IoT in the current state-of-the-art was reviewed in order to identify and map the shortcomings into research questions that were addressed throughout this research.

1.3.2 System Design

In this thesis, we introduce a hybrid IDS architecture for IoT networks that consist of centralised and distributed IDS agents integrated with a novel placement strategy. The proposed architecture can detect and mitigate the effect of node failures. Firstly, we model a WSN with the use of Random Geometric Graphs (RGG). The RGG model efficiently captures spatial characteristics of the network that are closely related to network connectivity; e.g. inter-dependencies on the existence of wireless links among neighbouring nodes. Then, motivated by how IoT networking protocols, such as RPL, manage and operate the network, we identify the trade-offs between the communication overhead introduced by an IDS and its detection rate of attacks such as the sinkhole attack.

The central IDS controls the entire IDS architecture and relevant data from the distributed agents. Each network node that runs an instance of the distributed agent, monitors and collects data on local network activity from its 1-hop neighbouring nodes. This implies that not all nodes need to run the IDS agent, but only a subset of them such that every node in the network has at least one 1-hop neighbour operating as IDS agent. In graph theory, such a subset is defined as a vertex cover of the corresponding RGG graph that captures the structure of the network. This subset of the nodes that act as IDS agents is selected based on the Vertex-cover algorithm (greedy algorithm) to find a subset of minimum cardinality with proper placement. Moreover, we propose a method to maintain and monitor the distributed IDS agents against node failures. The central IDS frequently checks the set of IDS agents against node failures. In case any of the IDS nodes fail to communicate with the central IDS, the central IDS agent will re-run our proposed algorithm to select a new subset of the nodes to act as IDS agents.

1.3.3 Prototyping

A prototype was implemented in order to evaluate the effectiveness and efficiency of the aforementioned approach. We extended the-state-of the-art IDS for WSN by Raza et al. called SVELTE Raza et al. (2013). In SVELTE, authors consider multi-hop peer-to-peer IPv6-enabled WSNs running the 6LoWPAN stack Shelby and Bormann (2011) on ContikiOS Dunkels et al. (2004). They develop a hybrid IDS architecture that consists of a centralised module running on the Sink and a distributed agent running on each individual sensor node.

In this thesis, we focus on experimentally evaluating the proposed approach on SVELTE as a representative example of hybrid architecture IDS for ad-hoc networks. Particularly, we evaluate the trade-off between the potentially reduced overhead of the IDS in successfully detection rate (due to lower number of active IDS agents in the network) versus the reduced communication overhead and increased energy efficiency of the network. We also evaluate the resilience and robustness of our proposed method against random node failures.

1.3.4 Experimental Results

We integrate our method on the state of the art on IDS for WSNs and conduct our performance evaluation via extensive emulations. We consider various network densities as they are formally defined by RGG model. Experimental results show that our proposed approach achieved high detection rates with a subset of the nodes running as IDS agents. The energy consumption and communication overhead introduced by the proposed IDS is reduced due to the reduction of the number of IDS agents.

The conducted experiments show that 1) indeed the IDS detection rates remain at very high levels (around 85%) even with a subset of the nodes as IDS agents; 2) that the required number of IDS agents in the network in order to achieve these levels is independent from the network population and in fact *constant*; 3) that the energy consumption and communication overhead introduced by the IDS is proportional to the number of IDS agents, therefore our method allows for massive energy gains while not affecting the detection rate of the IDS. Furthermore, results show that our proposed IDS architecture is resilient and robust against node failures. Centralised IDS able to monitor the distributed IDS agents and reallocates a new subset to run as IDS agents whenever node failure accrue.

This indicates that the energy efficiency and resilience of hybrid IDS architecture for ad-hoc networks is independent to the number of nodes acting as IDS agents and their placement. It suffices that only one of neighbouring nodes monitors and reports relevant information to the Sink. Thus, it reduces the number of nodes that are needed to operate as IDS agents.

1.4 Research Contributions

In this research we study efficient and lightweight Intrusion Detection Systems for ad-hoc networks via the prism of IPv6-enabled Wireless Actuator Sensor Networks. We first use Random Geometric Graphs (RGG) that allows to provide a formal model of WSNs. RGG capture the spatial characteristics of WSNs as such inter-dependencies on the existence of wireless links among neighbouring nodes. We focus on network attacks in IoT-specific networking protocols such as sinkhole attack in RPL. We identify the underline cause of communication overhead in state-of-the-art and try to optimise the trade-off between energy efficiency of IDS and detection rate. we propose a novel IDS architecture that requires only a subset of the nodes with proper placement to efficiently operate distributed IDS agents.

We proposed a hybrid architecture IDS in Ad-hoc networks that consist of centralised and distributed IDS agents integrated with a novel placement strategy. In our approach, the IDS architecture can be monitored against nodes failure. Initially, we model a WSN with the use of Random Geometric Graphs (RGG). The RGG model efficiently captures spatial characteristics of the network that are closely related to network connectivity; e.g. inter-dependencies on the existence of wireless links among neighbouring nodes. Then, motivated by how IoT networking protocols, such as RPL, manage and operate the network, we identify the trade-offs between the communication overhead introduced by an IDS and its detection rate of attacks such as the sinkhole attack. We investigate this trade-off via extended emulations and show that it is not necessary for all nodes to act as IDS agents. This allows us to establish that in order to achieve a high detection rate with minimal energy consumption overheads one needs to pick the minimum number of nodes running the IDS agent which is achieved through a strategic placement of these agents/nodes.

1.5 Publications Associated with this Thesis

1.5.1 Journals

- **JISA** Al Qurashi, M., Angelopoulos, C.M. and Katos, V., 2020. An architecture for resilient intrusion detection in ad-hoc networks. Journal of Information Security and Applications, 53, p.102-530.

1.5.2 Conferences

- **ICS-SCR** M. Al Qurashi, C. M. Angelopoulos, and V. Katos, “Efficient intrusion detection in ad-hoc networks,” in 6th International Symposium for ICS& SCADA Cyber Security Research 2019 6, 2019, pp. 117–125.

- **ICC2020** Al Qurashi, M., Angelopoulos, C.M. and Katos, V., 2020, June. An Architecture for Resilient Intrusion Detection in IoT Networks. In ICC 2020-2020 IEEE International Conference on Communications (ICC) (pp. 1-7). IEEE.

1.6 Thesis Structure

This thesis is structured as follows. Chapter 2 presents an overview of major attacks in WSNs, discusses IDS techniques and challenges of designing IDS for WSNs and discusses the state-of-the-art for the proposed IDS in WSNs. This involves Wireless Sensor Networks networking protocols which are included in the investigation of the nature of tiny devices and their security challenges. We consider the most important contributions in field of IDS in WSN and IoT in the current state-of-the-art to identify the shortcomings and map them into research questions that are addressed throughout this research.

Chapter 3 elaborates on the proposed methodology and outlines the different stages that the research went through to fulfil the objectives of this thesis. This chapter discusses the proposed network model and adopted IDS architecture based on Random Geometric Graphs. Chapter 4 explores in greater detail and depth the prototype of the proposed IDS architecture based on random IDS placement. The chapter presents the performance evaluation of the proposed approach, and also discusses the simulation results and findings. Chapter 5 presents a detailed explanation of an optimisation solution that enhances the proposed IDS placements strategy by using Random Graph optimisation algorithms. Further, the Chapter presents and discusses the simulation results and findings of proposed method that design to mitigate the effect of nodes failures. Finally the conclusion and recommendations for future research are presented in Chapter 6.

Chapter 2

Literature Review

2.1 Introduction

This chapter reviews the current state-of-the-art IDS in WSNs. We present the security aspects and challenges of WSNs, networking standard protocols and a summary of the related work in the field of intrusion detection in WSNs.

This literature investigates constrained-networks and the nature of tiny devices and their security challenges. Further, it reviews standardised wireless network protocols and their security aspects, followed by a review of the state-of-the-art contributions of IDS in WSN and IoT to identify that shortcomings and open issues that will be addressed throughout this research.

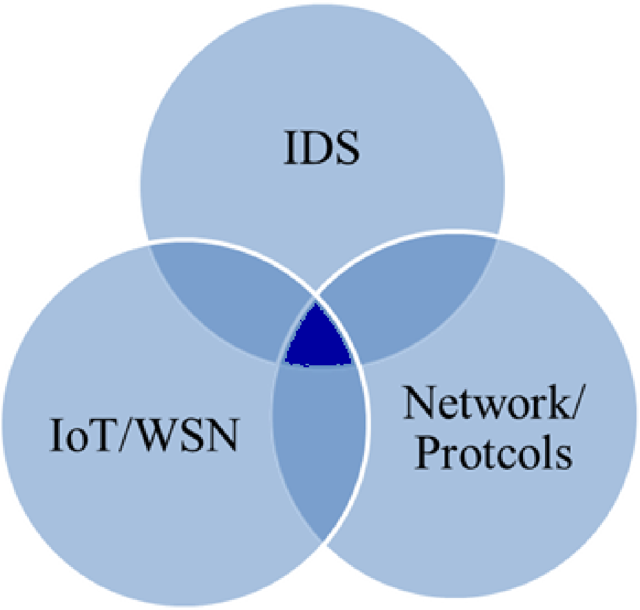


FIGURE 2.1: Literature Review Landscape

2.2 Wireless Networks

2.2.1 WSN Architecture

Typical WSNs are composed of a number of sensor nodes (SNs) which are distributed in a wide area, and a gateway. SNs consist of a sensing unit, a processing unit, a transceiver and a power unit. Each node has the ability to sense its surrounding environment and perform simple computations. Sensor nodes communicate wirelessly with their peers further to direct communication to based station (Khan 2014). Usually, data in WSNs is transmitting towards a base station. The packets may have to be forwarded over multi-hop routing since the radio chip is not powerful enough to communicate directly to base station when the node is too distant. Moreover, the base stations (access points) have much more computational, energy and commutation resources. They work as a gateway between sensor nodes and the end user since they forward data from a WSN to a host server, as shown in Figure 2.2.

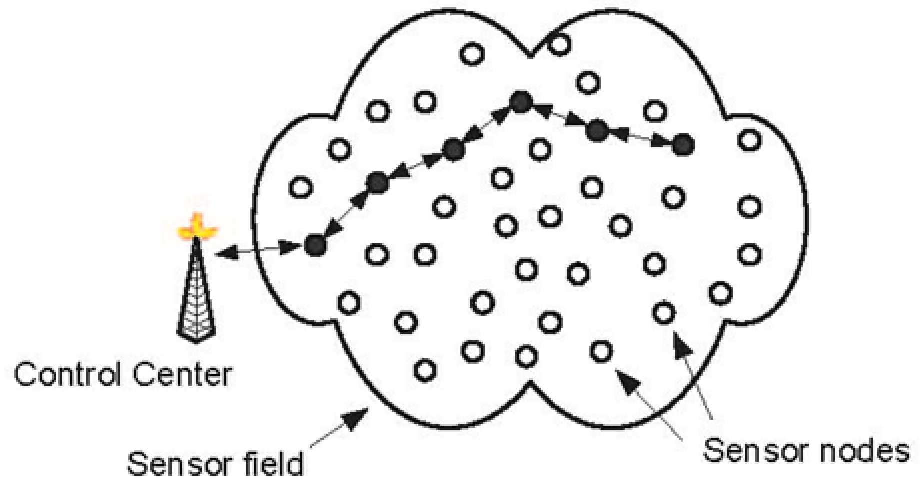


FIGURE 2.2: Typical Wireless Sensor Network (WSN)

Sensor and based stations use protocol stack similar to the OSI model, but it consists of five layers: application layer, transport layer, network layer, data link layer and the physical layer(Akyildiz et al. 2006). This protocol stack combines power and routing awareness, integrates data with networking protocols, communicates power efficiently through the wireless medium and promotes cooperative sensor node efforts (Chen et al. 2009) (Ghosal and Halder 2013). The physical layer is responsible for frequency selection, carrier frequency generation, signal detection and signal processing and data encryption. The data link layer is responsible for

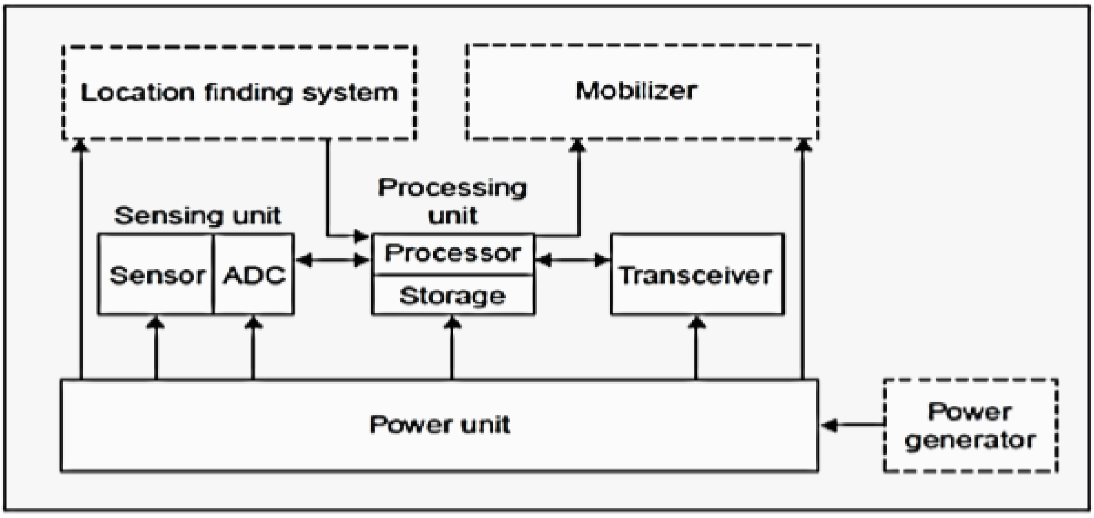


FIGURE 2.3: Common sensor node Architecture (Singh et al. 2014)

the multiplexing of data streams, data frame detection, medium access flow control and error control. It ensures reliable point-to-point and point-to-multipoint connections in a communication network. The network layer routes the data given by the transport layer. It is responsible for specifying the assignment of addresses and how packets are forwarded. The transport layer helps to preserve the flow of data if the sensor networks application requires it. This layer is especially needed when the system is planned to be accessed through the Internet or other external networks (Kolias et al. 2016),(Pathan 2016),(Lavric and Popa 2017).

2.2.2 Energy consumption in WSN Nodes

Energy consumption is a critical issue in WSNs because of the constrained energy resource, which is related to the manufacturer and sensor nodes are usually powered by two AA batteries. The sensor node in a WSN consists of several units: microprocessor, transceiver, sensor and power supply modules as shown in Figure 2.3 (Biswas et al. 2018),(Adu-Manu et al. 2018),(Zhou et al. 2011). These components work to gather to collect sensing data about the WSN environment and transmit these data over the base station (e.g.to the cloud, database, another system, process, etc.) (Seah et al. 2009)(Luo et al. 2014).

The energy consumption of WSN nodes can be summarised as follows:

- **Sensing Unit.** The energy consumption of a sensing unit is due to its operations, such as analogue-to-digital and signal modulation in both burst and periodic modes. For instance, the energy consumption in periodic mode can be expressed as follows:

  $$E_{sensor} = E_{on-off} + E_{off-on} + E_{sensor-run} \tag{2.1}$$

  where E_{on-off} is the energy consumption of the closing sensor operation, E_{off-on} is the energy consumption of the opening sensor operation and $E_{sensor-run}$ is the energy consumption of the sensing operation that is equal to the working voltage multiplied by the current of sensors and the time interval of the sensing operation (Abo-Zahhad et al. 2015).

- **Processing Unit.** The core activities of the processing unit are sensor controlling, protocol communication and data processing. Basically, this unit supports three operation states (sleep, idle, run). The processor energy consumption, represented as, E_{cpu} is the sum of the state energy consumption $E_{cpu-state}$ and the state-transition energy consumption, $E_{cpu-change}$ where $i = 1, 2, ..m$ is the processor operation state and m is the number of the processor state, $j = 1, 2, ..n$, is the type of state transition and n is the number of the state-transition (Tomić and McCann 2017)(Pantazis et al. 2013).

$$\begin{aligned} E_{cpu} &= E_{cpu-state} + E_{cpu-change} \\ &= \sum_{i=1}^{m} P_{cpu-stat}(i) T_{cpu-state}(i) + \sum_{i=1}^{n} P_{cpu-stat}(j) T_{cpu-state}(j) \end{aligned} \tag{2.2}$$

where $P_{cpu-state}(i)$ is the power of state i that can be found from the reference manual and $T_{cpu-state}(i)$ is the time interval in state i, which is a statistical variable. $P_{cpu-change}(j)$ is the frequency of state transition j and $T_{cpu-change}(j)$ is the energy consumption of one-time state transition j (Tomić and McCann 2017).

- **Communication Unit.** The communication unit includes the base-band (near-zero frequency) and radio frequency, which are responsible for sending and receiving node data. The transceiver usually has several operation states: $Tx,\ Rx, Off,\ Idle,\ Sleep\ and\ Clear\ Chanel\ Assessment$ (CCA/ED) (Zhou et al. 2011). The transceiver energy consumption can be calculated by summing the state energy consumption and state-transition energy consumption. Energy state consumption can be calculated from the expression below (Wei et al. 2017)(Pantazis et al. 2013)(Khanmirza and Yazdani 2016):

$$E_{trans-state} = E_{TX} + E_{RX} + E_{Idle} + E_{sleep} + E_{CCA} \tag{2.3}$$

Energy consumption in data processing is much lower compared to the data communication. For instance, the energy cost of transmitting 1 KB a distance of 100 m is approximately the same as that for executing three million instructions using a 100 million instructions per second (MIPS)/W processor(Dhand and Tyagi 2016).

2.2.3 WSN security

Security Challenges: According to our literature review, we can summarise security challenges of WSNs as presented below:

- **Resource constraint** Security solutions in WSNs add an expensive communication overhead and energy consumption.
- **The vulnerabilities of the communication medium** The wireless communication between WSN nodes can be intercepted easily.
- **Sensor nodes are vulnerable to physical attacks** Since wireless sensors are physically accessible, an adversary may gain full access to nodes and obtain sensitive information or even physically damage the nodes.

- **Dynamic changing of the topology** The topology of WSNs usually change quickly due to adding or nodes failures.

Security requirement: In a WSN, data are transmitted wirelessly over the air, which is an unreliable and unsecure communication medium for critical and confidential data. An adversary can intercept the communication of a WSN and may listen to or alter sensitive data. Security prevention solutions such as cryptography should cooperate with other security solutions to enhance the security of WSNs, and to ensure the security of wireless (Chen et al. 2009),(Grover and Sharma 2016),(Mendez et al. 2017),(Burhanuddin et al. 2018). In this research, we focus on network layer security threats. Based on the literature review, the basic security goals for WSNs were derived presented follows:

- **Confidentiality:** Data in WSNs travel wirelessly over an unreliable communication medium, therefore, adversaries can easily intercept the communication listening to sensitive and critical data. Data confidentiality in WSN forbids adversary to obtain data which is one of the crucial security requirements in WSNs.
- **Integrity:** Since adversaries can easily intercept the communication in WSNs, they are able to alter data and break into its integrity during communication. Date integrity helps to prevent data to be altered or modified by an adversary or unauthorised node.
- **Availability:** In WSNs, denial-of-service DoS attacks as an example can take place at any protocol layer of WSN causing an excessive communication overhead that drains constrained resources of WSN. As a consequence, DoS attacks effect the availability and the operation of many critical applications like those in the military and healthcare sectors. Availability ensures that services and information are available to an authorised user whenever it is required.
- **Non-repudiation:** Non-repudiation helps to identify and isolate compromised nodes. Further, non-repudiation ensure that any communication between

nodes within wireless networks is undeniable, and also it prevents malicious nodes from hide their activities.

- **Authentication:** In WSNs, data are transmitted over untrusted public wireless environment, thus malicious nodes might acquire some sensitive data. Each node must ensure the identity of peer nodes with which it communicates.

2.2.4 Security Attack in WSNs

Wireless sensor networks are vulnerable to a number of types of attacks due to their limited resources, and unreliable transmission medium(Khan 2014),(Deogirikar and Vidhate 2017),(Chelli 2015). WSNs attacks can be classified into five classes based on the protocol layer an attack can occur, as presented below:

- **Physical Layer Attacks** Jamming: Is the primary attack in physical layer, where a malicious node attempts to send a high energy signal toward sensor nodes causing interference with the radio frequencies of the nodes. This attack can eliminate the WSN services resulting into denial of service (DoS)(Can and Sahingoz 2015),(Zhu et al. 2017).

  Tampering: Since sensor nodes are physically accessible, adversaries could access a node by physical means and extract stored sensitive data such as encryption/decryption keys(Raju and Parwekar 2016).

- **Data link Layer Attacks** Collision: When two or more nodes attempt to transmit simultaneously on the same frequency, the data portion will be changed due to packet collision. As consequence, the packet will be rejected due to checksum failures (Yadav and Kumar 2016)(Miranda et al. 2014).

  Exhaustion: In this attack the adversary consumes all energy resources of the targeted node and disturbs the media access control protocol (MAC), by occupying the channel by sending and receiving unnecessary data (Yadav and Kumar 2016).

Sybil Attack: A malicious node in this attack assumes the identities of many, legitimate sensors through MAC spoofing. A Sybil attack can be detrimental to the services offered by a sensor network (Benzarti et al. 2017).

- **Network layer attacks** Spoofing, Replaying Routing: This is a direct attack against routing protocols. A malicious node can modify routing information of the entire or part of a WSN to disrupt the network traffic and flow. This kind of attack can create new routing paths, generate false error messages or shortening or extending source routes (Sinha et al. 2017).

  Selective Forwarding: The aim of selective forwarding attacks is to interrupt the communication process, where a malicious node performs forwarding of the received packets to a neighbouring node in a selective manner. Selective forwarding is considerably more difficult to detect, when compared to an approach of a malicious node systematically dropping all packets (Butun et al. 2014). Wormhole attacks: In a wormhole attack, there is one, two or more malicious nodes are placed on a network at different locations; these nodes create tunnels and forward and replay packets through these tunnels. The aim of the attacker is to deceive the sender and receiver nodes by convincing them that they are situated at a distance of one or two hops, instead of the actual distance which is actually higher (Amish and Vaghela 2016),(Patel et al. 2015).

  Sinkhole/Blackhole attack: In a Sinkhole attack a malicious node advertises itself as the best route to the sink node. A malicious node would listen to route requests and then it would falsely advertise that it has the shortest path to the base station. When a malicious node succeeds in attracting sufficient network traffic, it can start dropping all or selected packets or even change the content of some of the packets(Cervantes et al. 2015),(Deogirikar and Vidhate 2017),(Abdelshafy and King 2016),(Jain and Tokekar 2015).

  Hello flood attack: Many routing protocols in WSN require nodes to announce their state through a hello broadcast message. In this attack, a malicious node would broadcast random packets to cause congesting to the channel (Alanazi et al. 2015),(Bouabdellah et al. 2018).

Acknowledgement Spoofing: Each node in wireless sensor network uses a static routing information table, and it updates this routing table by sending or receiving Hello and ACK messages. In this kind of attack, a malicious node would capture a packet from its neighbouring nodes and spoof an acknowledgment by providing false information to the nodes(Xiao et al. 2016),(Gai et al. 2017),(Taylor and Johnson 2015).

Homing attack: In a homing attack, adversaries analyse network traffic in order to deduce the geographic location of a critical node such as cluster head nodes or the neighbour nodes of the sink node in order to physically disable these critical nodes and eventually shut down the entire network. The aim of this kind of attack is to block the traffic to the sink node, also to provide better environment for launching other attacks (Butun et al. 2014)(Ghosal and Halder 2013).

- **Transport Layer Attacks** Flooding: In this attack a malicious node would continuously send connection requests until the WSN resources drained out (Benzarti et al. 2017). De-synchronization: Attackers try to disturb and swindle existing connections by repeatedly sending fake packets which contain control flags or sequence numbers causing de-synchronization between the two ends (Liu et al. 2005)(Hossain et al. 2015),(Ioannou and Vassiliou 2016).

- **Application Layer Attacks** Overwhelm attack: Adversaries in this attack try to overwhelm the entire wireless sensor network by sending appropriate stimuli to cause the sensors to send huge amounts of traffic to the base station. As a consequence, this attack drains out the network resources (Nawir et al. 2016),(Kasinathan et al. 2013).

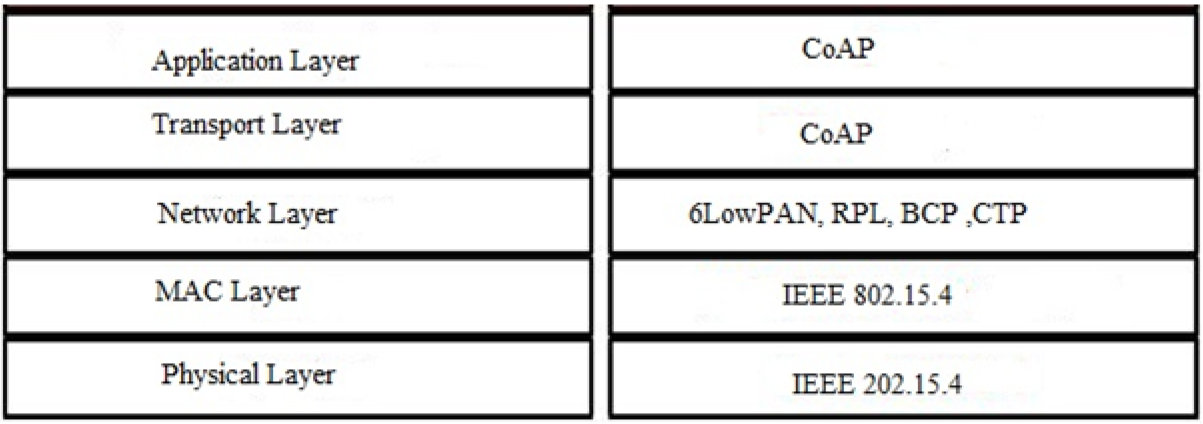
  Path-based DoS attack: Malicious nodes in this attack are placed along the path between a source and the base station and inject fake packets that drain out the network resources. Therefore, the pathways to base station will be exhausted and denied for the legitimate nodes (Sonar and Upadhyay 2014),(Andrea et al. 2015)(Gope et al. 2016),(Goyal et al. 2010).

Layers	**Attacks**	**Attacks Impact**	**Countermeasure**
Physical Layer	Jamming Tempering	Radio interference Physically access nodes and extract sensitive information	Channel hopping and Blacklisting Protection and Changing of key
Data Link Layer	Collision Exhausting Sybil	Consumes network resources Impersonate other legitimate identities	Link layer encryption Change the key regularly
Network Layer	Spoofed routing selective forwarding Wormholes Sinkhole Sybil Hello Flood Acknowledgment Spoofing	Exploit weakness in routing protocol	Effective encryption and authentication Multi path routing Secure routing protocol Hierarchal and geographic routing protocol Effective encyption
Transport Layer	Homing Flooding De synchronization	Exploit weakness in communication protocols	Client puzzles
Application layer	Overwhelming	Drains out network resources	Isolate malicious node

Table 2.1: Characteristic of Security attacks in WSNs and their Countermeasure

2.3 WSN Network and Communication Model

In order to understand the security requirements of the communication and standard protocols, we considered the protocol stack under IETF (Internet Engineering Task Force) with respect to network layer standard communication protocols and routing protocols. In this section, we present the network layer standard communication protocols and the security analysis of these protocols.



Application Layer	CoAP
Transport Layer	CoAP
Network Layer	6LowPAN, RPL, BCP ,CTP
MAC Layer	IEEE 802.15.4
Physical Layer	IEEE 202.15.4

Figure 2.4: Communication Standards and Protocols

2.3.1 WSN Standards Communication Protocols

1. **RPL**(Routing Protocol for Low-power and lossy network) is a distance-vector dynamic routing protocol that supports link layer technologies (e.g. IEEE 802.15.4, Wireless Heart, etc.) and manages the communication from wireless nodes to the base station (sink node). RPL has two main components: wireless sensor node and border router which is responsible for the translation of packets for Internet. Furthermore, RPL organizes the network topology as Directed Acyclic Graph (DAG) that can be directed to one or more sink node as Destination Oriented DAG (DODAG) where Border Router is the root (Sink node). The best path is computed usually based on the number of hops as routing metric and node energy consumption or the expected number of transmissions (ETX). RPL protocol can

support MP2P (Multipoint-to-Point), P2MP (Point-to-Multipoint) and P2P (Point-to-Point) topologies. The information exchange maintenance of the topology is supported by RPL control messages (e.g. DODAG Information Object)(Silva et al. 2018) (Zhao et al. 2017),(Porambage et al. 2015). RPL organize the network topology as Directed Acyclic Graph (DAG) that can be directed to one or more sink node as Destination Oriented DAG (DODAG) Figure 2.5. RPL use Internet Control Message protocol v6 (ICMPv6) that

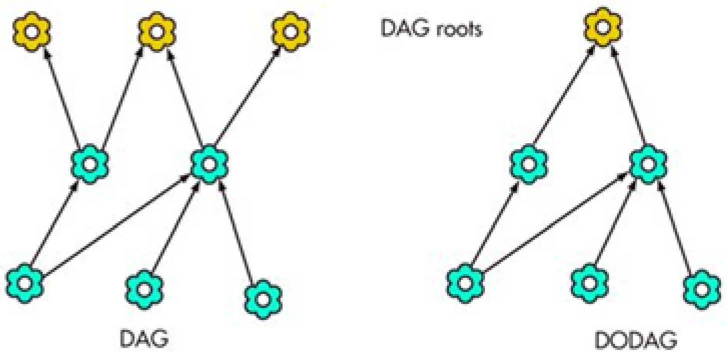


Figure 2.5: DOG and DODAG Topology (Sutaria 2014)

enable nodes to send messages through wireless networks. It has three control massages as presented below (Al-Fuqaha et al. 2015)(Tomić and McCann 2017),(Yang et al. 2015),(Farooq et al. 2015),(Kharrufa et al. 2017):

- **DODAG Information Object (DIO)** : DIO is a multicast message provides information about the network for other nodes to discover and join the network as shown in figure above (Shaffer et al. 2015).
- **DODAG Information Selection (DIS):** DIS is a message used by node to broadcast a request for DIO when it has not received within specific time.
- **Destination Advertisement Solicitation (DOA):** DAO is unicast from node to DODAG root to confirm that the sending node has accepted the DIO to join the network. This message is routed upward across the DODAG (Accettura et al. 2011).

There are several attacks that had exposed new security issues and vulnerabilities in WSNs protocols (e.g. IEEE 802.15.4, 6LoWPAN, RPL). For instance, the network layer attacks exploit the weaknesses in routing mechanism in PRL protocol. There are number of attacks protocol that exploit the weakness in RPL by altering control massages in WSNs (Granjal et al. 2015) (Chamudeeswari and Sumathi 2017).

2. **6LoWPAN** (IPv6 for Low-Power Wireless Personal Area Network)is a network protocol that enables constrained wireless nodes to connect to the internet. In essence, this protocol modifies IPv6, as the standard version of the latter is impractical for deployment in a constrained wireless communication network. In particular, 6LoWPAN is enabling the efficient use of IPv6 where it fragments IPv6 into small parts, the minimum size of IPv6 packet is 1280 bytes and IEEE 802.15.4 supports 127 bytes packet. 6LoWPAN protocol supports multi-hop communication where the nodes can forward packets to each other, however it faces the problems of routing solution that supports different communication patterns and deals with limited resources, low-data rates, link failures, and nodes mobility which led to design RPL (Rahman and Shah 2016)(Raghunandan and Lakshmi 2011),(Mayzaud et al. 2014),(Pu 2018).

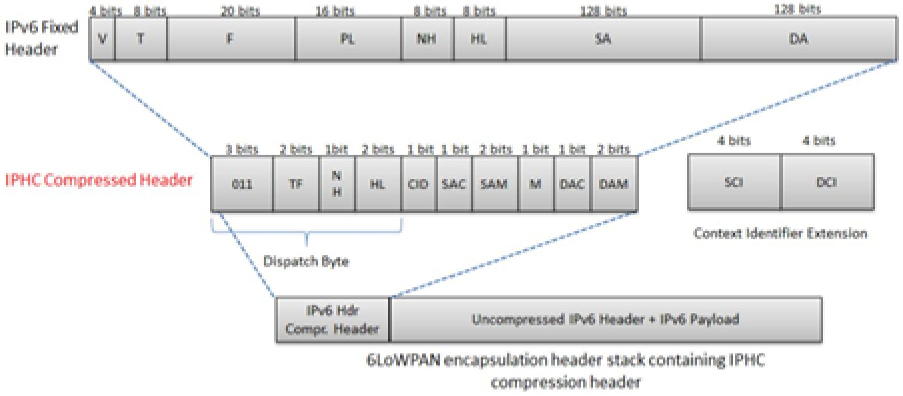


Figure 2.6: 6LoWPAN compressed header (Ishaq et al. 2013)

3. **IEEE 802.15.4** IEEE 802.15.4 is a standard radio technology and low-cost protocol for PHY and MAC layer. It has been used widely in many wireless applications due to its low-cost and low power consumption. It has three

different modulation techniques as presented below (Molisch et al. 2004),(Liu et al. 2017):

- **IEEE 802.15.4 for PHY layer:** It supports 11 low-frequency band (868/915 MHZ) channels and 16 high-frequency ISM (Industrial, Scientific and Medical) radio band (2.4 GHZ).
- **IEEE 802.15.4 for MAC layer:** It manages the access to physical channels and time slots, frame detection and node association. Further, it uses CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance) method
- **IEEE 802.15.4.e for MAC layer:** It is revised version of IEEE 802.15.4 for MAC layer that supports multi-hop communications by using (TSMP) Time Synchronized Mesh Protocol.

4. **CTP** (Collection Tree Protocol) CTP is a tree-based routing protocol designed for low traffic rates providing many-to-one or one to-many communication. CTP is a best-effort protocol that does not guarantee reliable delivery. Furthermore, it addresses routes to tree where roots are generated by using the expected transmissions (ETX). A single root is allowed which has ETX of 0, while a node's ETX is a sum of the ETX of its parent and the ETX of its link to the parent (Tomić and McCann 2017)(Granjal et al. 2015).

5. **BCP** (Backpressure Collection Protocol) is a low cost routing protocol that designed based on backpressure routing where there is no explicit path computation between source and destination. The routing paths are chosen based on the link backpressure weight that is a function of the queue and link state information. When the forwarding queue is non-empty, the node computes weights for all of its neighbours (Jiao et al. 2016).

6. **CoAP** (Constrained Application Protocol) CoAP is a standard protocol supports application-layer communication and it has been developed for web transfer within the IoT. It is similar to the Hypertext Transfer Protocol (HTTP), CoAP depends on the representational state transfer (REST)

architecture which is embedded in User Datagram Protocol (UDP) for transactions in the 6LoWPAN environments. CoAP uses very limited resources which reduced complexity and provides the same services as HTTP (Tomić and McCann 2017).

2.3.2 WSN Routing Protocols

The network layer in WSNs is responsible for the communication in the network. Furthermore, the communication model is integrated into the routing protocols with additional requirements and challenges compared to the conventional network. In particular, the distributed nature, dynamic topology and constrained energy source introduce new requirement that must be considered in designing routing protocols. There are several routing mechanisms have been developed and proposed to overcome the challenging of designing routing protocol in WSNs. These mechanisms have considered the constraint and other requirements of WSNs. Therefore, an efficient routing protocol will offer significant reductions of energy consumption that prolong the life time of WSNs. The routing protocols can be classified into four main schemes: Communication, Network Structure, Topology Based and Reliable Routing Schemes (Pantazis et al. 2013),(Zhou et al. 2011),(Al-Fuqaha et al. 2015), (Manjeshwar and Agrawal 2001).

1. **Communication Scheme**. This scheme adapts the communication model into a routing protocol as the main operation of the protocol in order to route packets in the network. The protocols in this scheme deliver more data for a given energy amount and can perform close to the theoretical optimum point-to-point and broadcast networks in terms of dissemination rate and energy usage(Ancillotti et al. 2013). However, the Communication Scheme protocols do not have high delivery rates and do not guarantee the data delivery. The protocols in communication scheme can be classified as follows:

- **Query-Based Protocols** The destination nodes of this type of protocols propagate a query for data (e.g. sensing task) from a node through the network, and a node having the data send back this data to that initiated the query(Singh et al. 2009).

- **Negotiation-Based Protocols** These protocols use negotiation to eliminate redundant transmission in the network by exchanging a series of negotiation messages before real data transmission(Siddiqui et al. 2006).

2. **Network Structure Scheme** Depending on whether the nodes have uniform or distinct and different roles, the structure of a network can be classified into flat and hierarchical structure. Nodes route packets and are connected based on the network structure they implement. The underlying protocols in this category can be classified as follows:

- **Flat Protocols** In Flat protocols all the nodes in the network are equal which minimize the overhead to maintain the infrastructure between communicating nodes (Zhang et al. 2014).

- **Hierarchical protocols** The hierarchical routing protocols impose a structure on the network in order to achieve energy efficiency, stability and scalability. Protocols in this class organize the network nodes into clusters where each cluster has cluster head node which is nominated based on higher remaining energy. As a consequence, clustering protocols achieve a significant reduction of energy consumption and prolong lifetime of the network (Liu 2015).

3. **Topology based Scheme**The protocols in this scheme require every node in the network to maintain the topology information. The topology base protocols can be classified as follows:

- **Location-based protocols** these protocols use the position information to find a path from source to destination that has minimum energy consumption(Camp et al. 2002).

- **Mobile Agent-based protocols** The mobile agent protocols have a mobile agent that travels among the nodes in the network to perform a task based on the environment conditions (Rath et al. 2019).

4. **Reliable Routing Protocols**Reliable Routing protocols provide the resilience to route failures by achieving load balancing or satisfy certain QoS metrics such as energy consumption and bandwidth. However, the scheme may add more overhead on the nodes since nodes need to maintain the routing table and QoS metrics at every node. The protocols in this scheme can be classified as follows:

   - **Multipath –Based Protocols:** They are more resilient in case of routing failure(Singh 2018).
   - **QoS-Based Protocols:** QoS-Based protocols balance between energy consumption and data quality when the sink node requests sensory data from any node in the network (Yessad et al. 2018).

Protocol	Classification	Advantages	Drawbacks
DREAM	Network Topology-Based	Efficient packet transition	Consumes the network bandwidth
SPEED	Reliable Routing	Performs well in end to end communication	Low performance in heavy traffic networks
LEACH	Network Structure -based	Compatible with ad-hoc, distributed,low energy consumption	Not applicable for large networks
CBMPR	Reliable Routing	Simplicity, limited Interference	Path joining
TEEN	Network Structure -based	Handles sudden changes in the network	Energy Consumption and overhead in large networks
ZRP	Network Structure -based	Low routing traffic	Expensive delay Message
ROAM	Reliable Routing	Avoid sending unnecessary search packets,	Required to keep sending Hello
GDSTR	Reliable Routing	Able to find the shortest routes	High energy consumption and communication overhead
SELAR	Reliable Routing	Select the node with highest energy level	Does not perform well when node keep change its location
RPL	Reliable Routing	Low Energy Consumption	Does not support multicast traffic
SPIN	Communication	Minimal cost when stat up	Low delivery rate
ACQUIRE	Communication	Ideal for complex queries	Flooding
LEACH-C	Network Structure -based	Less Energy consumption	Overhead
PEGASIS	Network Structure -based	Reduces The transmission distance	No consideration for of the sink node
ELCH	Network Structure -based	Minimize the transmission energy consumption	Cannot handle clusters exceed certain amount

Table 2.2: Comparison of Network layer routing protocols

2.3.3 Eenrgy Efficiency Performance Evalaution of Communication Protocols

Energy consumption is among the most critical issues in WSNs, as these can be found in deployments that do not have connectivity to power supplies and may require batteries as energy sources. Therefore, the existing communication protocols are designed based on residual energy and transmission energy consumption.

In WSNs, there are three major activity catagories that affect the level of energy consumption: sensing, data processing and the communication of sensor nodes which is the major factor of the energy consumption. Consequently, we consider the energy consumption factors in WSNs with particular emphasis on communication factor. There are some metrics that can be used to evaluate the performance and energy efficiency of communication routing protocols in WSNs (Tomić and McCann 2017) (Le et al. 2013) (Ahmed et al. 2015). These metrics can be summarized as follows:

1. **Energy per Packet.** It refers to the amount of energy that packet spends from a source to a destination.

2. **Network lifetime.** In many cases this metric denotes the time when the first node or certain section of the network's nodes is dead.

3. **Average Energy Dissipated.** This metric is related to the average dissipation of energy per node over time in the network as it performs functions such as transmitting, receiving, sensing and aggregation of data.

4. **Low Energy Consumption.** An efficient protocol should consider the remaining energy of the nodes and selects routes that prolong the network's lifetime.

5. **Total Number of Nodes Alive.** This metric is related to network lifetime and it shows the area coverage of the network by alive nodes over time.

6. **Average Packet Delay.** This metric is calculated as the average one-way latency that is observed between the transmission and reception of a data packet at the sink.

7. **Packet Delivery Ratio.** It is the ratio of successfully delivered packets to destination over the total number of packets sent from source sensors.

8. **Time until the First Node Dies.** This metric indicates the duration until the first node runs out of energy.

9. **Energy Spent per Round.** This metric is related to the total amount of energy spent in routing messages in a round.

10. **Packet Size.** The size of a packet determines the time that a transmission will last.

11. **Distance.** The distance between the sender and receiver that can affect the required power to send and receive packets. The routing protocols can select the shortest paths between nodes in order to reduce energy consumption.

2.3.4 Attacks and Security Analysis of Network Layer Communication Protocols

1. **6LoWPAN protocol** is vulnerable to a number threats since there are no indigenous security mechanisms defined in context of this protocol. For instance, through a fragmentation attack one can change the packet fragment fields in the fragmentation chain. This is feasible due to the lack of node authentication when joining the network. An unauthorised node can perform the attack on the receiver buffer since the receiver waits for all fragments for reassembly. Moreover, 6LoWPAN does not secure end-to-end communication between IP sensor nodes and the Internet which makes them vulnerable to spoofing and man in the middle attacks. The IPsec (Internet protocol Security) specification which enables the authentication and encryption of each IP packet at the network layer may enhance the security as prevention solution (Pongle and Chavan 2015*b*)(Wallgren et al. 2013)(Pantazis et al. 2013).

2. **RPL** is a Routing protocol for LLNs low power and lossy networks in network layer (e.g. RPL, AODV). It may overcome the routing challenges in constrained networks since it has been designed based on resources constraints awareness of LLNs. Nevertheless, a wide majority of WSNs and IoT attacks can occur in network layers due to routing protocols vulnerabilities. Malicious and adversarial nodes would typically exploit the protocols weaknesses in order to affect communication, routing and networking in general. RPL is designed for the multipoint to point communication with three basic security modes namely: unsecured mode, preinstalled mode and authenticated mode. None of RPL security mode has been implemented which makes RPL protocol prone to number of internal network layer attacks. The attacks on RPL are summarised below: (Kamble et al. 2017)(Weekly and Pister 2012) (Al-Fuqaha et al. 2015) (Airehrour et al. 2016)

   - **Selective Forwarding Attack:** This attack occurs when malicious or compromised nodes selectively forward packets that may lead to DoS (denial of services) attack. In RPL protocol, this attack disrupts the routing path by forwarding all RPL control messages and drops the rest of the traffic (Le et al. 2011)(Stehlik et al. 2016).

- **Sinkhole Attack:** Sinkhole attack is an insider attack were an attacker can compromise a node inside the network and launch an attack. The malicious node in this attack advertises a beneficial path in order to attract its neighbour nodes to route the traffic through it based on the routing metric that used in the underlying routing protocol. This kind of attack does not disrupt the network operation but it breaks into the data integrity and confidentiality in the network (Rehman et al. 2016) (Deogirikar and Vidhate 2017).

Protocol	Attacks	Attack Characteristic	Security Solution
LoWPAN	Fragmentation attack	Changes the packet fragment fields	
	Authentication attack Man in middle attack	Exploit security weaknesses in end-to-end communication	Network access control
	Spoofing attack		IPsec
			Bot analysis module
RPL	Selective forwarding attack	Forwarding all RPL control messages and drops the rest of the traffic	Heartbeat
		Advertise beneficial path in order to attract its neighbor	Failover/Rank authentication
	Sinkhole Attack	Impersonates other nodes identities	
			Distributed hash tables
	Sybil Attack	Broadcasting Hello message with strong routing metrics	
	Hello Flooding Attack	Creating tunnel between the two or more malicious nodes	
			Separate keys for network segments
	Wormhole Attack		
		Attracts some child node to be parents of other nodes	
	Rank Attack		
		Uses DIS (DODAG Information Solicitation) to generate more control DIS messages	Limit the rate of tickle timer resets
	DIS Attack		
CTP	Blackhole Attack	Advertise beneficial path to attract network's traffic to pass through malicious node	Secure CTP
	Selective forwarding attack	Alter some packet header and data	
	False routes Attacks		Kinesis
	Header/data Modification		
BCP	Sinkhole	Advertise beneficial path in order to attract its neighbor	VAR trust model
	Data alteration	Selectively drop some packets	
	Selective forwarding attack		Virtual trust queuing

TABLE 2.3: Attacks on Network layer protocols

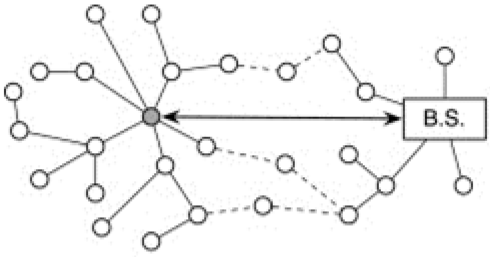


FIGURE 2.7: Sinkhole Attack (Kaur and Singh 2016)

- **Sybil Attack:** In this attack, a malicious node impersonates several identities of the same physical node. It should be noted that this type attack is not evaluated yet on the RPL routing protocol (Jan et al. 2015) to the best of our knowledge.
- **Hello Flooding Attack:** According to this attack, a malicious node introduces itself as neighbor node to many nodes by broadcasting Hello message with strong routing metrics when joining the network. In RPL protocol, the DIO message is the equivalent Hello message that is used to advertise information about DODAG (Moudni et al. 2016).
- **Wormhole Attack:** RPL can be vulnerable to the wormhole attack. This kind of attack occurs by creating a tunnel between the two or more malicious nodes and forward and replay packets through these tunnels(Khan et al. 2013).
- **Rank Attack:** Rank value in RPL increases from root to leaves nodes. An attacker attracts some child node to be parents of other nodes in order to attract network traffic going toward the root (Base Station) or by advertising false rank value which may lead to establish a non-optimal route(Le et al. 2013).
- **DIS Attack:** This attack uses DIS (DODAG Information Solicitation) to generate more control DIS messages. Malicious nodes periodically send the DIS messages to its neighbors. As a result, the receiver nodes upon receiving DIS message reset the DIO timer assuming something went wrong with the topology around. In multicast DIS attacks, the evaluation showed the highest increase in end to end delays which generates

more control overheads and is energy exhausting (Pongle and Chavan 2015*b*)(Kamble et al. 2017).

3. **CTP** routing protocol was designed to achieve minimum power consumption, but the security measures was not considered. As a consequence, CTP is vulnerable to routing attacks such as sinkhole, wormhole, and so forth. Malicious nodes can alter CTP operation rules such as ETX which could lead to presence of loops and inconsistencies within the network. Therefore, the number of beacon frames will be increased which in turn will cause more communication overheads and increased energy consumption (Elyengui et al. 2015).

4. **BCP** routing protocol does not adopt any security mechanism that makes BCP vulnerable to network layer security threats. A malicious node can perform selective forward attacks by dropping some received packets or altering massage header or data which disturb the network operation. Moreover, a malicious node may exploit BCP operating rules by advertising a wrong queue size and link quality to attract or prevent any incoming traffic (Pantazis et al. 2013).

2.4 Intrusion Detection System in WSNs

In conventional networks there is a limited concern about the computational complexity and network resources required in the deployment of the security defence mechanisms. On the contrary, WSNs have constrained resources, such as limited computational power on the sensor side, low energy life time and hardware constraints. As such, designing a suitable and effective IDS in a WSN environment is a challenging task.

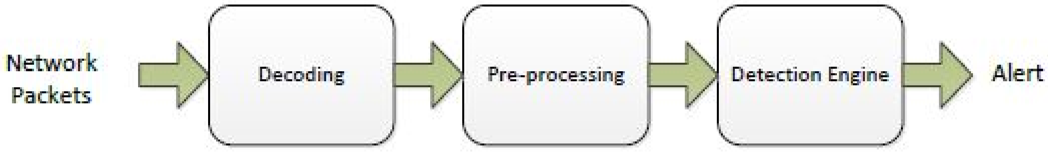


Figure 2.8: Basic components of an Intrusion Detection System

2.4.1 Detection technique

With regards to IDS approaches, there are four main categories used in the detection of attacks:

Signature based (S-IDS): With this type of detection method, the patterns or signatures of known attacks are stored in a database, and the detection process matches the incoming events against these signatures (Shamshirband et al. 2014)(Liao et al. 2013)(Ma et al. 2015).

Anomaly based (A-IDS): Anomaly based methods can potentially detect zero day attacks or unknown attacks, as they attempt to capture normal, legitimate behaviour and alert the administrator when activity outside the "normal" envelope is observed (Kushwaha et al. 2017)(Van et al. 2017).

Specification protocol analysis (SPA): The specification in SPA focuses on analysing and tracing the protocol states and their transitions in order to detect whether there is any deviation from expected protocol behaviour. This is done, for example, by pairing request messages with replies (Patel et al. 2013) (Abduvaliyev et al. 2013)(Song et al. 2016).

Hybrid IDS approaches: Hybrid techniques combine both anomaly-based and signature-based approaches. Hybrid mechanisms contain two detection models, signature based to detect known attacks and anomaly based for detecting unknown attacks (Guerroumi et al. 2015)(Pan et al. 2015).

The anomaly detection approach is more flexible than rule-based detection, and could potentially detect zero-day attacks and new kinds of attacks. However, it may raise a higher number of false positive alarms because of legitimate network traffic may change slightly or usual behaviour has not been analysed completely.

	Pros	Cons
SIDS	Simple and effective methods to detect known attack. Detailed contextual analysis	Cannot detect unknown attacks. Hard to keep signatures up to date. Time consuming to maintain the knowledge
AIDS	Effective to detect new and unknown attacks Less dependent on OS Facilitates detections privilege abuse.	Weak profile accuracy due to observed events being constantly changed. Unavailable during rebuilding of behaviour profiles. Difficult to trigger alerts in right time.
Stateful protocol analysis (specification-based)	Trace the protocol stats Distinguish unexpected sequences of commands	Resource consuming to protocol state tracing and examination. Unable to inspect attacks looking like benign protocol behaviour. Might incompatible to dedicated Oss or Aps.
Hybrid	Exploit the benefits of both techniques and detect unknown attack as well as known attack	and detect unknown attack as well as known attack

Table 2.4: Comparison of IDSs Detection Technique

2.4.2 Machine Learning in IoT Security

Machine learning (ML) algorithms have been widely used in a variety of real-world applications because of their ability to learn and solve problems from experience without direct human interaction. Machine learning begins the learning process with observations, direct experience or instructions to look for specific patterns in datasets to make a decision without human interference. Particularly, in intrusion detection learning algorithms, the main function is to distinguish and classify normal behaviour and detect any system deviation. The performance improvement of learning algorithms can be achieved by improving the accuracy of the classification and learning experience from which the algorithms collect the data of system normal behaviour (Al-Garadi et al. 2020).

Machine learning algorithms can be classified into three main categories: Supervised, unsupervised and reinforcement learning (RL). Supervised learning algorithms learn from existing or training data and can apply what has been learned to new data using labelled dataset. Training data consist of inputs labelled with correct output. Learning algorithm search for pattern in training data that correlate with the correct outputs. The learning algorithm can also compare its output to modify the model accordingly. After training, a supervised learning algorithm is applied to new data to determine which label the new data will be classified based on prior training data (Jordan and Mitchell 2015). On the other hand, unsupervised learning algorithms have originated from supervised learning methods for learning representations, learn important representations of the input without labelled and training data. These learning algorithms aim to analyse unlabelled data and to classify the input data into distinctive groups by identifying commonalities in the data and the similarity between them. The last but not least, reinforcement learning (RL) are trained by interacting with an environment. RL aims to understand an environment and discover the best approaches to a given agent in different environments. An RL agent learns from the consequences of its actions, rather than from being explicitly trained. Usually, it selects its actions based on its past experiences and new choices (Hastie et al. 2009),(Schmidhuber 2015).

2.4.2.1 Common Machine Learning algorithms in IoT Security and IDSs

This section discusses the common ML algorithms that have been used in IoT security applications and IDSs.

- Decision Trees (DTs) methods form a tree hierarchy of branches to classify samples according to their feature. Each path from the root to a leaf represent a classification decision, and each internal node in a tree represents a feature. Tree branches in decision tree represent an outcome of the test and denote a value that the node can have in a sample to be classified. The samples are classified starting at the origin vertex and with respect to their feature values. The feature that is able to optimally split the training samples is considered as origin node of the tree. Authors in (Alharbi et al. 2017), used DT-based classifier in their proposed intrusion detection to analyse network traffic to detect suspicious traffic behaviour of DDoS. Nevertheless, DT-based supervised machine learning methods require large storage due to the nature of DT data (Al-Garadi et al. 2020).

- Support Vector Machines (SVMs) are used supervised machine learning as classification algorithms. SVMs create a splitting hyperplane in the data attributes between classes, where the distance between the hyperplane and the most adjacent sample points of each class is maximised (Tong and Koller 2001). SVMs ML algorithms have been widely used in intrusion detection algorithms, such as the proposed attack detection in a smart grid (Ozay et al. 2015) . This research showed that the proposed detection algorithm based on SVM is effective in detecting known and unknown attacks, and also has better performance comparing with traditional methods used for attack detection in smart grids.

- Beyesian theorem-based ML algorithms describe the probability of an event based on previous related information of the incident. For instance, Naive Bayes (NB) classifier is used in a supervised classifier that uses the Bayesian theorem to forecast the probability that can be used by intrusion detection to classify the traffic as normal or abnormal. The network feature such as connection duration, connection protocol (e.g. TCP) and connection status flag can be used for traffic classification(Mukherjee and Sharma 2012),Jordan and Mitchell (2015).

- k-Nearest neighbour (KNN) is a simple supervised machine learning algorithm that often uses the Euclidean distance to solve both classification and regression problems. KNN has been used in statistical estimation and pattern as a non-parametric technique. KNN algorithm use feature similarity to predict the values of new datapoints, where the new data point will be assigned a value based on how closely it matches the points in the training set (Al-Garadi et al. 2020). Authors in (Li et al. 2018), proposed a classifier based on KNN that have been used for network intrusion detection and anomaly detection in IoT environment. Furthermore, authors in (Sicari et al. 2015), proposed intrusion detection to classify nodes as normal or abnormal in WSN. The proposed system showed that the detection algorithm achieved a high level of accuracy comparing to other ML algorithms.

- K-Means clustering is an unsupervised ML algorithm. This method is designed to discover clusters in the data, and k refers to the number of clusters to be generated by the algorithm. The aim of this algorithm is to find groups in the data, with the number of groups represented by the variable K. The algorithm works iteratively by assigning each data point to one of K groups based on the features that are provided. Data points are clustered based on feature similarity. The inputs of the algorithm are the number of clusters (k) and dataset, which contains a set of features for each sample in the dataset (Al-Garadi et al. 2020).

- Artificial Neural Network (ANN) is a computational model in machine learning that emulate the human mind's ability to identify patterns and interpret perceptual information. It is the foundation of artificial intelligence (AI) that is able to solve problems that would prove impossible or difficult by human or statistical standards. ANNs have self-learning capabilities that enable them to produce better results as more data becomes available. ANNs are arranged in multiple layers, where the information enters the neural network through the input layer, which is the primary outermost layer. Authors in (Hodo et al. 2016) proposed a threat analysis based on multi-level perceptron ANN for IoT. Their approach focuses on the classification of normal and threat patterns on an IoT Network. Their experimental results showed that the proposed detection method successfully detect various DDoS and DoS attacks with a high level accuracy.

- Convolutional neural network (CNN) is a class of deep neural network that was introduced to reduce the data parameters in ANN. The parameters are reduced by utilising three concepts: sparse interaction, parameter sharing and equivariant representation. The training time complexity of a CNN is reduced due to the reduction of the connections between layers. (Goodfellow et al. 2016). CNN has been applied to the training approaches in DL, and also allows for the automatic learning of features from raw data with high performance. However, it has a high computational cost, therefore; implementing CNN in resource-constrained devices is challenging (Al-Garadi et al. 2020).

- Recurrent neural network (RNN) is another class of neural networks that allow previous outputs to be used as inputs, thus the output of the neural network depends on the present and past inputs. RNNs have achieved good performance in many applications with sequential data , such as machine translation (Pascanu et al. 2013).Authors in (Torres et al. 2016), use RNN in analysing network traffic behaviour to detect malicious behaviour, their approach achieved accurate malicious behaviour detection.

2.4.3 Network Traffic Source

With regards to the source of traffic or data that IDSs use to monitor the resources of a computerised networked system, IDSs can be categorised into three groups (Arias et al. 2015)(Liao et al. 2013)(Hu et al. 2018)(Igbe et al. 2016):

Network based Intrusion Detection System (NIDS): NIDS can detect malicious activity by monitoring network traffic, capture packets and analyse entire packets payload.

Host based Intrusion Detection System (HIDS): A host based intrusion detection system (HIDS) analyses information collected from a particular host such as file systems, network events and system calls.

Hybrid Intrusion Detection System: It consists of both HIDS and NIDS components typically involving a central agent who checks the entire network traffic while a mobile agent checks a system through log file inspection.

	Strength/pros	Limitation/cons
HIDS	Identify intrusion by monitoring hosts file system and system calls events of network.	Needs to be installed on each machine. Can only monitor attacks on the host where it is deployed
NIDS	Identify intrusion by monitoring network traffic. Can monitors deferent types of system at the sometime.	Hard to detect encrypted traffic. Difficult to detect intrusions in virtual network. Difficult to detect internal attack.
Hybrid-IDS	Only need to place underlying network. Uses characteristics of NIDS and HIDS, thus has the strength of both of them.	Require high communication computational cost

Table 2.5: Comparison of IDSs traffic source

2.4.4 Detection Frequency

Following the current state of the literature, there are in general three conventions on the detection of frequency that influence the communications and energy consumption overhead.

Continuous: the Intrusion Detection System monitors the network continuously and work without interruption(Arias et al. 2015).

Periodical: the detection algorithm is executed in certain and specified periods of time (Butun et al. 2014).

Event-based: in this case, the detection algorithms are activated only when a particular critical event has occurred (Abduvaliyev et al. 2013) (Butun et al. 2014).

Continuous monitoring in terms of energy consumption is most expensive as it comes with a long execution duration. From an energy consumption perspective, periodical and event-based techniques consume lower energy than continuous monitoring since they are activated in specific points in time. Nevertheless, they may not be efficient in critical applications (Liao et al. 2013).

2.4.5 Architecture and placement strategy

IDSs can be classified according to their architecture and computation location to Centralised, Distributed (Cooperative or Cluster-based) and hybrid (consist of Centralised and Distributed architecture) IDS. First, a centralised system requires all relevant monitoring and detection information to be reported to the base station. The detection algorithm is executed only in base station which is more powerful in terms of processing power and memory and is thus supports complex detection method. Centralised systems have an overview of the entire network that provides the ability to detect suspicious nodes and activities in the network. However, the communication to the base station is increased due to the huge amount of reported traffic (Abduvaliyev et al. 2013) (Butun et al. 2014) (Benzarti et al. 2017)(Sharma and Gupta 2015)(Duhan and Khandnor 2016)(Kenkre et al. 2015)(Sabahi and Movaghar 2008). In distributed or cooperative systems, each node is equipped with an IDS agent, which cooperates with other agents and participates in the detection of the overall network. When a node detects an intrusion with incomplete evidence, it may cooperate with the detection procedure further, and, it can alert an entire the network about an intrusion. Distributed and cooperative techniques are suitable for designing lightweight IDS for WSNs; however, an node agent IDS cannot apply a powerful analysis algorithm since wireless sensor nodes are constrained in resources and performance. Finally, there is hybrid IDS, which is an architecture that combines centralised and distributed systems.

In these systems cluster head nodes or nodes equipped with an IDS agent, cluster head nodes (CHs) monitor their member node and cooperate with the base station as part of the intrusion detection mechanism. This technique inherits the advantages of both centralised and distributed architectures, which may reduce the overhead on WSN resource constraints (Abduvaliyev et al. 2013) (Liao et al. 2013)(Patel et al. 2010)(Peddabachigari et al. 2007)(Arrington et al. 2016)(Sabahi and Movaghar 2008).

2.4.6 Combination of IDS with prevention system in WSNs

Intrusion Prevention System (IPS) are more proactive types of software systems aiming to prevent the progress of an attack by responding to security incidents through actions such as firewall policy and Access Control List (ACL) updates and so forth. The combination of both IDS and IPS introduced a new security solution that is defined as Intrusion Detection and Prevention System. IDPSs are mainly focused on identifying, detecting and reporting a security incident to administrators to rapidly respond to the incident and mitigate the impact caused by the incident and update the security measures. IDSP can also monitor file transfers and identify suspicious ones. IDPS in wireless networks monitor network wireless communications against malicious activities, events characteristic of a malware attack, behaviour deviation, and performs mitigating actions. Particularly, a wireless IDPS monitors wireless network traffic and analyses wireless networking protocols against malicious activities. Wireless IDPSs can enhance the security of Wireless Networks. However, they cannot detect certain type of attacks (e.g. offline traffic possessing) and the problem refers to the deployment of sensors due to the lack of stander architecture. Furthermore, IDPSs require massive communication overheads in constraint networks and impractical with weak security prevention measures.

2.4.7 Challenges of IDS in WSNs

Designing an Intrusion Detection in WSNs is challenging for several reasons. First of all, the communication in WSNs is multi-hop, the topology of the network is dynamic topology and very often not known. Moreover, nodes are constrained in CPU, memory, bandwidth and energy. Due to constraints resource, traditional techniques used in conventional networks are impractical in WSNs.

In addition, transmitting data is very costly in terms of energy for the sensor nodes and thus should be minimised. More specifically, the challenges in intrusion detection systems for wireless sensor networks can be summarised as follows:

1. Sensor nodes have limited power resources, processing capability, memory and radio range.
2. DoS attacks can be more successful than other attacks since WSNs communicate over open environment (Coppolino et al. 2013).
3. Logging on a WSN node level may be limited due to the nodes' low storage capabilities (Patel et al. 2013).
4. Limited bandwidth requires more effort in data transmission. As consequence, data transmission may consume a significant amount of power (Liao et al. 2013).
5. Frequent changing of the topology, due to adding and failure node.

The challenges altogether require Intrusion Detection systems specifically tailored to wireless sensor networks.

2.5 Related Work and Current State-of-the-art IDS in WSNs

We have grouped the most contributed IDSs in field of Wireless Sensor Network based on the centralised, distributed and hybrid architecture. In centralised IDSs, all relevant monitoring and detection information has to be reported to the base station. Detection algorithm is executed only in base station where based station is more powerful in terms of possessor and memory and able to perform complex detection methods. On the other hand, in distributed systems each node equipped with an IDS agent cooperates with other agents and participates in the detection of the overall network. When a node detects an intrusion with incomplete evidence it may cooperate with components through global detection procedures. If a node detects an intrusion, it can alert entire the network regarding that intrusion. Finally hybrid IDSs combine Centralised and Distributed in order to exploit the advantages of these two architectures.

2.5.1 Centralised Architecture IDSs

Moon et al. proposed a detection method for WSNs in terms of energy consumption and aggregate IDS with intrusion prevention systems. In this approach, symmetric encryption and one–way hash functions have been used to construct the routing path between BSs and nodes. Their results showed that the total amount of required energy was reduced in some cases. However, the use of symmetric key increases the computation overheads. Further, the initial construction phase requires the BS and every node in the network communicate with each other using the topology and route construction message (TRC message) and the neighbour information response message (NIR message) which increases the communication overhead all over the network (Moon et al. 2014).

Midi et al. proposed an intrusion detection system for IoT named Kalis. Kalis is placed at the border router to collect features of the network and use these to dynamically configure appropriate detection techniques. The authors claim that this approach can be applied and extended to new protocols as it is a protocol independent method being based on features(Midi et al. 2017).

Jun et al. proposed a centralised intrusion detection system for IoT. They introduced Complex Event-Processing (CEP) techniques to monitor network packets that is placed at the router border. Their approach is a specification-based IDS relies on stored rules in Rules Pattern Repository using SQL. The experiment result reveals that their approach had increased the computation overheads and consumed less memory compared to traditional IDS (Jun and Chi 2014).

2.5.2 Distributed Architecture IDSs

Coppolino et al. proposed a distributed anomaly detection method to detect anomaly events in WSNs. They have used both signature and anomaly based detection techniques and their framework is composed of a central agent and number of local agents. The central agent performs data mining detection techniques whereas the local agents deploy lighter anomaly detection techniques. The empirical results showed that this method had low detection accuracy and high false positive rate. This approach was designed and evaluated based on QoS aspects, whereas it inherited assumptions from conventional networks; the performance of the proposed IDS was measured in terms of false positive rate, false negative rate, and accuracy. Moreover, it does not consider the constrained resources and nature of WSN since it use complex anomaly detection algorithm that required maintaining the profile of the network behaviour (Coppolino et al. 2013).

Kumarage et al. proposed a distributed anomaly detection method for WSNs that used unsupervised data partitioning adaptive with fuzzy c-means clustering that distributes the sensor nodes into clusters. This approach classifies the data and identifies anomalies through maximizing distributed in-network processing in a hierarchical architecture as a distributed cluster. This proposed method was evaluated by both synthetic and real data. The results showed this distributed method achieved a high detection accuracy and less communication overheads comparing with existing centralized algorithms. However, it still required communication overheads since each node is in charge of local observation and cooperation with neighbour nodes for local anomaly detection (Kumarage et al. 2013*a*).

Maleh et al. proposed a lightweight IDS for WSNs that combined anomaly based detection by employing support vector machine algorithms augmented by signature rules. In addition, this approach was applied to a cluster-based topology to reduce communication costs leading to network lifetime improvements. The simulation result showed that the detection of abnormal events has a high rate and lower false alarms. However, this approach combines anomaly detection (SVM) with signature based algorithm requiring more computation overhead and huge storage space to store attacks signatures. Furthermore, this approach requires all node members (local agents) to cooperate with cluster heads (global agent) in detection mechanism which increases the local communication overhead (Maleh et al. 2015).

Zhenge et al. proposed an approach based on Specification Protocol Analysis that exploits location information of a sensor node to verify the legitimacy of sensors and detect clone attacks. Furthermore, they proposed a distributed clone detection protocol (ERCD) protocol where neighbour nodes observe a suspect node and report potential divergent behaviour. The simulation results showed that this approach yielded a very high accuracy in detecting clone attacks and relatively low energy consumption comparing with existing approaches. Nevertheless, this approach is limited to clone attack and it may increase the communication overhead since the nodes required cooperating with neighbour nodes in detection mechanism (Zheng et al. 2016).

Cervantes et al. proposed a distributed Intrusion Detection System on 6LoWPAN called (INTI) for IoT. INTI is a distributed IDS combines trust and reputation concepts in which each node monitors exchange packet with neighbour nodes. In their approach, nodes are classified into leader, association and member nodes following hierarchical or cluster based network structure, also the node's category can be changed over the time due attack occurrence or network reconfiguration. When a node detects an attack, a broadcast alert is sent to other nodes in the networks. The performance and effectiveness of this approach were not presented (Cervantes et al. 2015).

Xie et al. introduced an Intrusion Detection method that uses hyper grid k-nearest neighbour(KNN) based anomaly detection for WSNs. This approach considered the distribution of computation load evenly all over the network where it distributed the nodes into clusters. Each node in the cluster monitors its local traffic and sends the summary to its cluster head (CH). Then, the cluster head collects all local summaries and aggregate from its member nodes in the cluster to broadcasts collected summaries as a global normal behaviour of the network. Member nodes will perform the detection mechanism locally based on global normal profile. The experiments with the real WSN data set have revealed the proposed detection method is robust for WSNs comparing with other existing approach where it features low computation and communication overhead. This approach may reduce the global communication overhead; however it may drain out the cluster head nodes resources due to the massive local communication overhead between member nodes and their cluster heads (Xie et al. 2012).

Wang et al. had proposed a cluster-based detection approach for WSNs that consists of three different IDS agents for sink node, cluster head and sensor node. Each agent is designed based on the capability of node that is designed for in order to exploit the advantages of distributing the detection efforts over all nodes. Additionally, they used hybrid detection technique that combine rule-based with back- propagation network (BPN) to detect anomalies. The simulation results showed that the proposed approach achieved low resource consumption in some cases; however the proposed BPN anomaly detection technique is not accurate enough to minimize the false positive alarm. This approach consists of three different IDS agents that increase the complexity of detection mechanism and increase the communication overhead (Wang et al. 2011).

Xie et al. proposed a segment-based anomaly detection method to detect anomalies in WSNs. This algorithm has combined the Distributed Segment-Based and Kullback–Leibler divergence measures and distributed the sensor nodes in clusters and each cluster has a cluster head node. The algorithm is executed separately by each cluster head node and main nodes. This method was evaluated using a real-world data set and the result had revealed a high performance and low communication costs. However, this proposed detection method is limited to hierarchical network and may be applied for flat network with additional requirements. It also assumes static architecture that does not consider frequent dynamic changes of WSN (Xie et al. 2017).

Oh et al. proposed a distributed signature-based detection system for IoT. In their approach, each single node in charge of monitoring the network traffic extracts the packet payloads in order to reduce unnecessary matching to reduce the computational overhead. The adopted detection algorithm in each node matches against conventional attacks signatures in Snort. Their detection approach may detect attacks faster than other algorithms, but only detects known conventional attacks (Oh et al. 2014).

2.5.3 Hybrid Architecture IDSs

Raza et al. introduced an IDS for WSN that follows a hybrid architecture. Their solution focuses on routing attacks and consists of a central IDS module (running computationally intensive processes) that runs on the Sink node and a lightweight distributed agent that is deployed on sensor motes. The proposed IDS has three main modules: a central module called mapper, a lightweight intrusion detection module, and a firewall. The proposed solution shows a good performance in small networks, but it introduces a massive communication overhead in larger networks. This is mainly due to the fact that the lightweight agent is deployed on every single sensor mote of the network, thus leading to bottleneck phenomena to emerge around the Sink as the diameter of the network increases (Raza et al. 2013).

Shareenivas et al. proposed a hybrid IDS for IoT by extending (Raza et al. 2013). They extended the detection module of SVELTE by using ETX(Expected Transmissions) metric with the detection process. Moreover, they proposed detection methods with geographical information to detect malicious nodes that attempt to attack ETX based networks. Their evaluation experiment revealed that their approach achieved better detection rates. However, they evaluated their approach in small networks with few nodes (Shreenivas et al. 2017).

Ponomarchuk et al. proposed a lightweight detection method to detect malicious packets in WSN based on the traffic analysis. Their detection technique was based on monitoring of packet reception rate (PRR) and inter-arrival time (IAT) of packets. Moreover, this method analyses the neighbouring nodes' behaviour using the thresholding technique against packet reception rate and inter-arrival time. The simulation result showed that this approach yielded low resource consumption and better accuracy in comparison with other algorithms. Nevertheless, monitoring of incoming traffic from nodes at their one-hop neighbours along the path towards the BS may require more communication overheads and the detection metrics used in the approach are limited to a few threats (Coppolino et al. 2013).

Le et al. proposed a specification-based intrusion detection system for IoT to detect attacks on RPL-based networks. Their approach is based on hybrid and partly distributed architecture that divides the network into clusters where each cluster has a cluster head. The distributed IDS agents are placed on the cluster heads to monitor member nodes within their cluster. And then, each cluster head reports all relevant detection information to the central IDS that is placed on base station. The simulation results revealed that their approach achieved high detection rate and low overhead. However, this proposed IDS is limited to cluster based networks (Le et al. 2016).

Pongle et al. proposed a hybrid intrusion detection system for IoT that consist of central and distributed IDS agents. The centralised IDS agent was placed on the border router and the distributed IDS agent is equipped with other network nodes. The distributed IDS agents are in charge of monitoring their neighbour nodes and report to the central IDs in the sink node. The simulation result showed that their approach achieved low energy consumption and high detection rate, but still limited to small size networks. (Pongle and Chavan 2015*a*).

.

Proposed approach	Detection technique	Architecture	Security Attacks	Limitations/ Drawbacks
Moon et al.	Signature Based	centralized	specific	Energy consumption, Very low detection rate
Coppolino et al.	Anomaly Signature Based	distributed	specific	complex detection algorithm
Maleh et. Al	Anomaly based	distributed	general	Communication overhead,
Raza et al.	Anomaly based	Hybrid	Routing attack	Communication overhead
Kumarage et. al	Anomaly based	distributed	general	Communication overhead
Xie et. Al	Anomaly based	Hybrid	specific	Communication overhead within local cluster
Zhenge et. Al	Specification protocol analysis	distributed	specific	Cooperation between nodes increased Energy consumption
Xie et. al	Anomaly based	distributed	general	Limited to hierarchical networks
Wang et. Al	Anomaly /Signature Based	distributed	specific	Communication overhead
Ponomarchuk et. A	Traffic analysis	distributed	general	Communication overhead
Jun et al.	Specification-based	centralized	general	computation overhead, Communication overhead
le at al.	Specification-based	Hybrid	Routing attack	Limited to cluster based networks
Shreenivas et al	Hybrid	Hybrid	Routing attack	Communication overhead
Ponoglo et al.	Anomaly based	Hybrid	Routing attack	Limited to small size networks.
Cervantes et al	Hybrid	distributed	Routing attack	No details provided
Oh et al.	Signature Based	distributed	conventional attacks	Only detect known attacks
Midi et. al	Hybrid	centralised	DoS attack	Energy consumption,

Table 2.6: Comparison of State-of-the-art IDSs

2.6 Evalution of the IDS

The proposed IDS can be evaluated through a simulation, an actual implementation environment (testbed) or a combination of both simulation and implementation in a real-world setting. Simulation is helpful for effectively analysing large networks, different topologies and investigate scalability issues. However, implementation allows better assessment of the actual behaviour in the real world. The current state of the art in IDSs, WSNs, have been evaluated using simulation, implementation or a combination of both. Nevertheless, there is no universal agreement nor standard evolution metrics to evaluate the proposed IDSs. IDSs can be evaluated from two different aspects: effectiveness and efficiency. Effectiveness metrics measure the detection accuracy and the ability of a system to distinguish between normal and malicious activity while efficiency focuses on the resources that systems use and consume (e.g. available energy, CPU, etc). The proposed IDSs in IoT and WSNs in the published literature are evaluated differently. The majority of the proposed systems in the state of the art focused on measuring the accuracy and effectiveness of the IDSs in terms of detection accuracy using different datasets with limited consideration to the efficiency metrices. Moreover, available works in the literature on IoT and WSAN IDS limit their simulation studies in networks with small populations which is not sufficient to evaluate the efficiency. Therefore, it is difficult to determine the better IDS. In table 2.6, we present a comparison of the current state-of-the-art IDS designed for WSNs with respect to their evaluation metrics.

2.7 Summary and Findings

During the last decade, a number of IDSs have been proposed in WSN and IoT. In the literature review, we focused on the most important contributions in area of IDS in WSNs and IoT. The current-state-of-the-art IDSs in WSNs and the IoT can be classified based on their architectures and placement strategies as centralised, distrusted and hybrid architecture. This classification helps in investigating the open issues and shortcomings in the current sate of the art since the security solution's architecture is an essential factor in constrained networks.

The majority of the proposed IDSs in WSNs are decentralised (distributed) solutions, in which the sensor nodes are responsible for detecting anomalies. They require cooperation with other nodes in wireless networks networks in the detection procedure, which increases the communication overhead between sensor nodes in WSNs. (Kumarage et al. 2013*b*) proposed a distributed anomaly IDS for WSNs. Their results showed this distributed method achieved less communication overhead when compared with existing centralised algorithms, but the communication overhead is still required since each node is in charge of local observation and cooperation with neighbour nodes for local anomaly detection. As a consequence, the distributed (decentralised) IDS architecture consumes the constraint resources in sensor nodes. (Xie et al. 2017) proposed a distributed IDS for WSNs, in which the IDS agents are placed on the cluster heads of the network. This approach is limited to hierarchical WSNs, as the detection algorithm is executed separately in each cluster.Due to the absent of a central IDS agent in this approach, each cluster works as an individual network and is responsible for monitoring its own traffic.

The centralised IDSs are typically placed on border routers (base stations), where they are more powerful and rich in resources (e.g. constant energy, powerful CPU, etc). The centralised approaches provide better control of the detection procedure, but they require intensive communication with other nodes to monitor node activities and manage the detection procedure. (Moon et al. 2014) proposed a centralised detection method for WSNs that considered energy consummation. Their centralised architecture requires BS and every node in the network to communicate with each other, which increases the communication overhead throughout the network the network.

(Coppolino et al. 2013) introduced a hybrid IDS that combines decentralised and centralised intrusion detection architectures for WSNs. This method consists of local agents that are installed in each wireless sensor node and a central detection agent located in the base station. The hybrid architecture may distribute the computation overheads among all the sensor nodes and base station; however, it introduces substantial communication overheads between the nodes and the base station that drains constrained resources in WSNs. (Le et al. 2016) proposed an IDS for the IoT based on hybrid and partly distributed architectures that divides the network into clusters, where each cluster has a cluster head. The distributed IDS agents are placed on the cluster heads to monitor member nodes within their cluster. Their approach showed low communication overheads and energy consumption, but it is limited to cluster-based networks. The fact is that the real-world IoT devices and Wireless sensors are not strict in the placement and may be placed randomly in the area of interest. Thus, these approaches target specific network typologies. The proper design of IDS architecture for the IoT and WSNs should consider the nature of constraint device and real-world node placement.

The current state-of-the-art IDSs in WSNs reveal resource-intensive IDS solutions that overburden the constrained resources. It relies on cooperation between sensor nodes, which increases the communication overhead and energy consumption since the limited bandwidth in sensor nodes requires more effort in data transmission. According to (Zhou et al. 2011), the energy cost of transmitting 1 KB a distance of 100 m is approximately the same as that for executing three million instructions using a 100 (MIPS)/W processor. Thus, communication overheads is a critical factor that plays a pivotal role in the energy consumption of WNSs and the IoT.

Moreover, the existing research has focused on IDS in WSNs the nature of WSN constraints (computational power, available energy, limited memory), but the frequent changing of the network topology due to node or communication failures has not yet been studied. As such, IDS in a dynamic network with a potentially changing network typology is a very challenging task. Accordingly, IDSs should consider account auto-configurability to cope with the dynamicity of WSNs. Furthermore, the current state-of-the-art IDSs in WSNs have been evaluated differently due the lack of universal agreement regarding the effectiveness and efficiency of evaluation metrics. For instance, (Coppolino et al. 2013) proposed a distributed anomaly detection method to detect anomalous events in WSNs, which was evaluated based on Quality of Service (QoS) with inherited assumptions from conventional wired networks. Most existing IDS for WSNs do not provide comprehensive analyses of real-world implementation and simulations, which makes analysing the effectiveness of an IDS mechanism a difficult task.

The shortcomings and open issues of the current state of the art in related work on designing IDS for WSN can be summarised as follows:

1. Existing IDS in WSNs and the IoT are still resource-intensive and, the constrained nature of WSNs and the IoT has not been adequately addressed.
2. The IDS placement strategy according to the current state-of-the-art, is based on assumptions holding from conventional computer networks, and thus a proper IDS placement strategy that addresses the nature of WSNs and the IoT is needed.
3. The robustness of the current state-of-the-art IDS in WSNs and IoT has not yet been extensively investigated.
4. The current-state-of-the-art IDSs in WSNs have been evaluated differently due the lack of universal agreement on the effectiveness and efficiency evaluation metrics.

A possible solution to overcome these challenges is to use a hybrid architecture that combines centralised and distributed IDS agents to avoid the disadvantages of both architectures separately. Such a hybrid architecture would provide better control of the IDS architecture since the centralised IDS agent is more powerful and rich in resources for performing complicated algorithms,procedures etc. Furthermore, the hybrid architecture placement scheme may reduce the communication overheads and energy consumption caused by the detection procedure and the communication between the base station and the nodes when it is integrated with the partly distributed architecture. Our IDS will be based on a small subset of the nodes of the network (distributed IDS agents) to monitor local network activity (e.g. by monitoring their 1-hop neighbours) and report this information to the sink node (centralised) for post-processing. In addition, we will consider dynamic distribution and randomise the IDS agent placement among the various sensor nodes. In particular, we will use graph theory to design our solution in order to formally investigate the particular characteristics of the IoT and WSN paradigms. Our designed system will address the following characteristics of the IoT and WSNs: their highly distributed nature; their ad-hoc network structure; the peer-to-peer communication scheme among the devices; the highly constrained nature of the devices per se in terms of resources (computational power, available energy, limited memory etc.). Random Geometric Graphs have been used as a well-studied model and a paradigm for wireless networks, such as sensor networks. Motes are represented as vertices in RGG, and the communication between these motes is represented by the edges. In other words, RGG can represent the actual placing of the set of n vertices uniformly and randomly in the area of interest. Following our network modelling with the use of RGG, we intend to investigate and employ more formal and rigorous methods from graph theory.

Chapter 3

Proposed Method

3.1 Introduction

This research introduces a hybrid intrusion detection architecture for IoT ad-hoc networks that combines centralised and distributed models with a novel placement strategy. The proposed approach respects the limitations and characteristics of IoT and sensor networks in particular, such as: their highly distributed nature; their ad-hoc network structure; the peer-to-peer communication scheme among the devices; and the highly constrained nature of the devices per se in terms of resources (computational power, available energy, limited memory, etc.). Random Graphs have been employed as they are a well-studied model and a paradigm for wireless networks, such as sensor networks.

This chapter describes the theoretical aspects of the proposed IDS for WSN and ad-hoc networks. In particular, we present and describe the underlying design of the proposed approach using a Random Geometric Graph followed by an optimisation solution and algorithm for better IDS placement and resilience. The latter is explored further in Chapters 4 and 5.

3.2 System Design Overview

According to the current state of the literature, IDSs designed for WSNs are classified according to their architecture as centralised, decentralised (distributed) and hybrid architecture IDSs. The majority of existing IDSs are designed with a decentralised architecture, where the sensor node is responsible for detecting local malicious nodes and attacks and is required to cooperate with neighbour nodes to detect anomalies. The drawback of the decentralised architecture is that a sensor node alone might be unable to detect some kinds of attacks since it has only a local view of the network. Moreover, decentralised architecture approaches require cooperation with other nodes in wireless networks in the detection procedure, that increases the communication overhead between sensor nodes in WSNs. Meanwhile, a centralised IDS architecture requires the transfer of all relevant detection information to the base station, which results in an intensive communication overhead. In particular, all sensor nodes transfer relevant detection information to the base station (sink node), which leads to base station failure. Finally, there is a hybrid IDS architecture that combines centralised and distributed IDS agents. This approach has been introduced in the current state of the art. It gives all sensor nodes part of the responsibility for detecting malicious activities, and all sensor nodes participate in the detection procedure, which increases the combined overhead due the massive communication between sensor nodes and base station.

In this research, we adopted a hybrid architecture combining centralised and distributed IDS agents. This combination helps to avoid the disadvantages of both centralised and distributed architectures while capitalising on their advantages. The hybrid architecture provides better control of the IDS architecture since the centralised IDS agent is more powerful and rich in resources for performing complicated algorithms and procedures. Furthermore, the hybrid architecture placement scheme may reduce the communication overheads and energy consumption caused by the detection procedure and the communication between the base station and nodes when it is integrated with the partly distributed architecture. Our IDS will be deployed on a small subset of the nodes of the network (distributed IDS agents) to monitor local network activity (e.g. by monitoring their 1-hop neighbours) and report this information to the sink node (centralised) for post-processing. Furthermore, we consider random placement of IDS agents among the various sensor nodes. In particular, we use graph theory to design our solution in order to properly account for the particular characteristics of the IoT and WSN paradigms. The proposed system considers the characteristics of the IoT and WSNs outlined earlier, whereas Random Geometric Graphs are used as they are a well-studied formal analysis tool for wireless and sensor networks. Motes are represented as vertices, and the communication between these motes is represented by the edges. In other words, a random geometric graph can represent the actual placing of the set of n vertices uniformly and randomly at the area of interest. By modelling the netwiork with the use of RGG, we investigate Vertex cover algorithms and employ more formal and rigorous methods from graph theory.

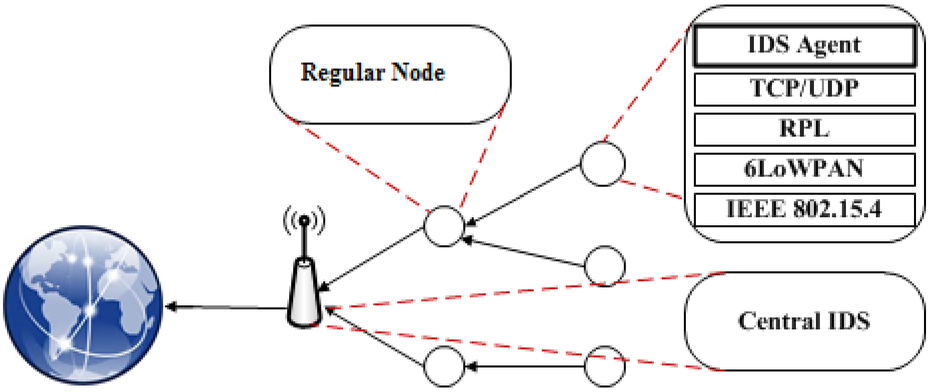


Figure 3.1: The Proposed IDS Architecture

In the proposed IDS architecture, the global communication overheads between sensor nodes and the base station will be reduced to the minimum since the small set of sensor nodes work as a local sink and monitor the neighbour nodes in its 1-hop or cluster. We propose two different placement strategies based on graph theories. The First placement algorithm is based on RGG. The RGG model efficiently captures spatial characteristics of the network that are closely related to network connectivity e.g. inter-dependencies on the existence of wireless links among neighbouring nodes. Then, motivated by the way in which IoT networking protocols, such as RPL, manage and operate the network, we identify inherent trade-offs between the communication overhead introduced by an IDS and its detection rate of attacks, such as sinkhole attacks. We investigate this trade-off via extended emulations and show there exists an underlying threshold behaviour in the efficiency of the IDS that is related to the connectivity threshold of the RGG model. Second, an IDS agent placement method is proposed as an optimisation approach for IDS placement. We used the existing Vertex-cover algorithm to find the minimum or close to minimum number of IDS agents with respect to the node placement. Thus, we can have a lower number of IDS agents that helps to improve the communication overhead and energy dissipation. Furthermore, we proposed an algorithm that integrates with the Vertex-cover approach to maintain and monitor the distributed IDS agents against node failures. The central IDS frequently checks the IDS agents against node failure. In the event any IDS node failing to communicate with the central IDS, the central IDS agent will re-run the proposed algorithm to select a new subset (cover set that covers other nodes) of the nodes to act as IDS agents.

3.3 Proposed IDS Architecture Based on Random Placement

We model a WSN with the use of RGG. The RGG model efficiently captures spatial characteristics of the network that are closely related to network connectivity; e.g. interdependencies on the existence of wireless links among neighbouring nodes. Then, motivated by the way in which IoT networking protocols, such as RPL, manage and operate the network, we identify inherent trade-offs between the communication overhead introduced by an IDS and its detection rate of attacks, such as the sinkhole attacks. We investigate this trade-off through extended emulations and show there exists an underlying threshold behaviour in the efficiency of the IDS that is related to the connectivity threshold of the RGG model.

3.3.1 Graph Theory

Within the domain of computer science and computer networks, graph theory studies the relations among objects and their connection, which is used to represent network communications, data organisation etc. A graph is a representation of a set of objects (vertices or nods) that are interconnected over the links called edges. Mathematically, a graph is a pair $G = (V, E)$, where V is a finite set of vertices and E is a finite set of edgesWest et al. (1996),Bollobás (2013). For instance, the set V might be $\{a, b, c, d, e, f, g, h\}$, and E can be

$\{\{a, d\}, \{a, c\}, \{b, c\}, \{b, e\}, \{b, g\}, \{c, f\}, \{d, e\}, \{g, h\}\}$. Together, V and E are a graph G, as shown in Figure 3.2

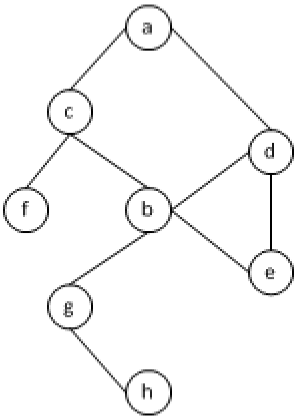


FIGURE 3.2: Example of a Graph

.

3.3.2 Random Geometric Graph

Real-world networks are complex, and thus it is difficult to identify their exact structure. Indeed, we only have incomplete information about their structure to use for analysis. Random network models provide a better representation of random and dynamic networks in order to determine long-term behaviour. Moreover, random graphs allow researchers to design algorithms that can adhere to numerous different structures Gupta and Kumar (2000),Durrett (2007).

RGG consists of a set of vertices that are randomly distributed in a d-dimensional area, where an edge (u, v) exists only if the Euclidean distance of u and v is less than R (the communication range). In wireless networks, R captures the ability of a node to communicate with its 1-hop neighbour.

In random graphs with n vertices, the possibility of an edge is determined randomly and independently of other edges with probability $p \in [0; 1]$. The probability p is the same as for all edges of $G \in G(n; p)$.

The $G(n, p)$ model has two parameters, n which is the number of vertices of the graph, and p which is the edge probability. For each vertice, v and w, p

is the probability that the edge (v, w) is present. The presence of each edge is statistically independent of other edges (e.g. when uv and vw exist, then uw possibly also exists)Godsil and Royle (2013).

3.3.3 Connectivity threshold of RGG

In our proposed IDS, we consider an area $\mathcal{A} \subset \mathbb{R}^2$ in a two-dimensional space. An instance of the *random geometric graphs model* $\mathcal{G}(\mathcal{X}_n; r)$ is constructed as follows: select n points $\mathcal{X}_n$ uniformly at random in $\mathcal{A}$. The set $V = \mathcal{X}_n$ is the set of vertices of the graph, and we connect two vertices iff their Euclidean distance is at most r. For any vertex $v \in V$, we will denote by $N(v)$ the set of neighbours of v and by $\deg(v) = |N(v)|$ its degree. Furthermore, we will denote by $\|u - v\|$ the Euclidean distance between the points corresponding to vertices v, u.

In Gupta and Kumar (1999) Penrose et al. (2016) it is shown that the connectivity threshold for $\mathcal{G}(\mathcal{X}_n; r)$ is

$$r_c = \sqrt{\frac{\ln n}{\pi n}} \tag{3.1}$$

3.3.4 Description of proposed system

Our proposed hybrid IDS architecture consists of two different IDS agents: a centralised IDS and distributed IDS agent. These two different agents are the main components of the proposed architecture; the central IDS located in the base station (the sink node), and distributed IDS agent is installed in every single sensor node. The distributed IDS agent monitors its neighbour nodes (e.g. by monitoring their cluster) and collects data on local network activity from its 1-hop neighbouring nodes for the central IDS agent, which is located at the sink node. Meanwhile, the central IDS controls distributed IDS agents and manages the detection procedure of the entire network.

Specifically, the proposed IDS architecture consists of a central detection agent located in the base station and a distributed lightweight intrusion detection agent deployed on a subset of the network nodes, as shown in Figure 3.1. This combination is useful since it exploits the advantages and avoids the limitations of both central and distributed architectures. The central agent manages the entire detection process and collects relevant data from the distributed agents. Each network node that runs an instance of the distributed agent monitors and collects data on local network activity from its 1-hop neighbouring nodes. This implies that not all nodes need to run the IDS agent, but only a subset of them, such that every node in the network has at least one 1-hop neighbour running the IDS agent. In graph-theoretical terms, such a subset would be able to capture the structure of the network. This also implies that there exists a minimum set of nodes that are able to efficiently monitor the network without compromising the performance of the IDS. This conjecture is explored empirically later on.

3.3.4.1 Central IDS

Central IDS is the main component of our proposed IDS, which is responsible for monitoring the entire network and detecting anomalies. In addition, it is responsible for the self-organisation of the network and contains a global network monitor that manages the monitoring task of other distributed agents. The detection engine in the central IDS is responsible for the detection policy and manages other IDS agents. The centralised IDS is placed in a special node within the network called a sink node or base station. This node represents the border gateway device located on the edge of the WSAN network, and it is powerful in terms of computing power, memory and energy supplies. These features of the sink node allow the centralised IDS to achieve better control and management of the monitoring procedure and the entire IDS architecture.

Algorithm 1 Detect Sinkhole Attacks

```
Require: $M \leftarrow$ the list of IDS agents nodes
Require: $N \leftarrow$the list of Regular nodes
  for Node in M  do
      for Node in N do
          if (Node.Rank+IDSagentNodeRank
  < Node.Parent.rank) then
              Node.fault=Node.fault+1
          end if
      end for
  end for
  for Node in N do
      if Node.fault>Threshold then
          Alarm
      end if
  end for
```

3.3.4.2 Distributed IDS agent

The distributed IDS agent is the lightweight component of the proposed IDS architecture that is placed on the sensor side. The sensor nodes in WSANs are highly constrained in terms of resources (computational power, available energy, limited memory, etc.). The distributed IDS agent is responsible for monitoring and analysing the traffic within its cluster or 1-hop neighbours. Additionally, it periodically provides the required and relevant detection information to the central IDS.

3.3.4.3 Distributed IDS agent placement strategy

We focus on the placement strategy of distributed IDS agents since it is the critical factor in designing security solution for WNSs and the IoT. Therefore, we adopt a hybrid IDS architecture, in which the central IDS is placed on the base station and the distributed IDS agent is placed only on a subset of sensor nodes. The IDS agent monitors traffic information within 1-hop neighbours. This network information is available to be monitored by 1-hop neighbouring nodes. For any set of neighbouring nodes, it suffices that only one of them is actively collecting and reporting relevant information to the Sink. Therefore, it is not necessary for all the sensor nodes to perform as IDS agents. This greatly reduces the number of nodes that need to operate as IDS agents. At this point, we intend to study and investigate the efficiency and effectiveness of our proposed IDS architecture. We focus on evaluating the trade-off between the potentially reduced (percentage of nodes acting as IDS agents is 100%, 80%, 60%, 40% and 20% of the total population) accuracy of the IDS in successfully detecting attacks due to the smaller number of active IDS agents in the network versus the reduced communication overhead and increased energy efficiency of the network.

3.3.5 The Network Topology

Real-world networks have incomplete information about their structure that is used for analysis. Random network models provide a suitable representation of random and dynamic networks in order to determine long-term behaviour. Moreover, Random geometric graphs enable the design of algorithms that can be applied to many different structures. RGG represent the actual placing of the set of n vertices uniformly and randomly at the area of interest Bollobás (2004).

We consider that the random uniform placement of the sensors inside the network area is abstracted by a *Random Geometric Graph*. RGG is formed by n vertices that are placed uniformly at random in the $[0,1]^2$ square. An edge (u, v) exists iff the Euclidean distance of vertices u and v is at most r, where r corresponds to the wireless communication radius r of the sensors. This holds assuming a disc radio model; two sensors can communicate with each other iff each one lies inside the communication range of the other. Random Geometric Graphs also have an important nice property: unlike other random graphs, like $G_{n,p}$, edges are not statistically independent of each other. That is, the existence of an edge (u, v) is not independent of the existence of edges (u, w) and (w, v). This property makes RGG a realistic model for WSANs, as it captures to a great extent the communication structure of real networks (at least their spatial aspects).

Particularly, we consider an area $\mathcal{A} \subset \mathbb{R}^2$ in two dimensional space. An instance of the *random geometric graphs model* $\mathcal{G}(\mathcal{X}_n; r)$ is constructed as follows: select n points $\mathcal{X}_n$ uniformly at random in $\mathcal{A}$. The set $V = \mathcal{X}_n$ is the set of vertices of the graph and we connect two vertices iff their euclidean distance is at most r. For any vertex $v \in V$, we will denote by $N(v)$ the set of neighbours of v and by $\deg(v) = |N(v)|$ its degree. Furthermore, we will denote by $\|u - v\|$ the Euclidean distance between the points corresponding to vertices v, u.

The RGG model provides a formal tool of constructing and characterising networks as "*sparse*, *dense* or *normal*". We also later find that this threshold also indicates the number of IDS agents needed in order to efficiently monitor a peer-to-peer ad-hoc wireless network Gross and Tucker (2001).

3.3.6 Implementation Plan

We apply our proposed approach to the state-of-the-art IDS for WSN of Raza et al. called SVELTE Raza et al. (2013). Their proposed IDS has a hybrid architecture that consists of a centralised module running on the Sink and a distributed agent running on each individual sensor node. The centralised module contains the 6LoWPAN Mapper (6Mapper), which is responsible for gathering information from the sensor nodes on the network topology. In particular, 6Mapper collects information on the rank assigned to each node by the RPL protocol (responsible for constructing and maintaining a global tree-like network structure in a distributed manner), which is closely related to the hop distance of each node from the Sink. This allows a second component - the intrusion detection component - to reconstruct and monitor the network topology for anomalies that indicate an intrusion.

For instance, a sinkhole attack could be deployed via a compromised node by having this node falsely announcing a significantly smaller rank to its neighbours. This causes its neighbouring nodes to assume that its distance to the Sink is much smaller than it actual is, thus directing all network traffic through the compromised node.

3.3.7 Evaluation Environment

We will evaluate the proposed approach experimentally using the Cooja emulator Osterlind et al. (2006). The Cooja emulator provides a detailed cross-layer simulation for WSNs running the 6LoWPAN stack. In our experiments, we consider three qualitatively distinct network densities as they are indicated by the RGG model. In particular, we consider a network area $\mathcal{A} = [0, 100]^2$, where n sensor motes are deployed uniformly at random, for $n \in \{32, 64, 128\}$. Following from equation 3.2, for each value of n, the corresponding network connectivity threshold is $r_c : \{18.5; 14.3; 11\}$, respectively. Thus, the communication range will be be $r = 20$. Thus, the RGG model provides us with a formal tool for constructing and characterising networks as "*sparse*", "*dense*" or "*normal*".

$$r_c = \sqrt{\frac{\ln n}{\pi n}} \tag{3.2}$$

Then, for each network density, we consider five scenarios, where we will reduce the percentage of nodes acting as IDS agents to 80%, 60%, 40% and 20% of the total population. In each case, we will set 10% of the node population to act as malicious nodes deploying sinkhole attacks by exploiting the rank mechanism of RPL. Any remaining nodes are regular nodes. We will consider 10 random instances of the network; this allows us to effectively mitigate in our simulations any issues that might occur due to the random network topology. We consider 10 iterations for each instance running for 3600 seconds, where nodes generate and transmit data approximately every second. For each scenario and each performance metric, we will compute average values and 95% confidence intervals.

3.3.8 Attacker Model

In our experiments, we consider sinkhole attacks because of this kind of attack is the one of most critical in the network layer. A sinkhole attack is an insider attack where an attacker compromises a node inside the network and launches an attack. The malicious node in this attack advertises a beneficial path in order to attract its neighbour nodes to route the traffic through it based on the routing metric used in the routing protocol. This kind of attack does not disrupt the network operation, but it affects the data integrity and confidentiality in the network Kamble et al. (2017).

A sinkhole attack could be deployed over a compromised node by having this node falsely announce a significantly lower rank to its neighbours. This would have its neighbouring nodes assume that its distance to the Sink is much smaller than the actual one, thus directing all network traffic through the compromised node.

3.3.9 Evaluation metrics

The aim of the proposed approach is to address the constrained nature of WSNs, especially in terms of resources (computational power, available energy, limited memory, etc). Therefore, we focus on the IDS architecture, with a focus on IDS placement since it is an essential factor of communication overheads and energy consumption in constraint networks. Thus, we define evaluation metrics that capture the trade-off between energy consumption /communication overheads with the detection rate of malicious activities.

3.3.9.1 Detection Rate

We define the detection rate as the number of true positives of malicious nodes over the total number of malicious nodes in the network.

$$\text{Detection rate} = \frac{\text{number of true positive detections}}{\text{total number of malicious nodes}} \tag{3.3}$$

3.3.9.2 Communication Overhead

The communication unit in WSNs nodes is responsible for sending and receiving node data. The transceiver usually has several operation states: $Tx, Rx, Off,$ $Idle$, $Sleep$ and $Clear$ Chanel Assessment/Energy Detect (CCA/ED) (Zhou et al. 2011). The transceiver energy consumption can be calculated by summing the state energy consumption and state-transition energy consumption.

Energy state consumption can be calculated from the following expression Zhou et al. (2011)Pantazis et al. (2013)Khanmirza and Yazdani (2016):

$$E_{trans-state} = E_{TX} + E_{RX} + E_{Idle} + E_{sleep} + E_{CCA} \tag{3.4}$$

Energy consumption in data communication is the most expensive factor in WSNs energy dissipation. For instance, the energy cost of transmitting 1 KB a distance of 100 m is approximately the same as that for executing three million instructions by a 100 (MIPS)/W processor(Zhou et al. 2011). Therefore, we define the communication overhead as the additional volume of data communication introduced in the network as a result of the operation of the IDS. We follow the practice of Raza et al. (2013) and monitor this metric only to the 1-hop neighbouring nodes of the Sink (the rationale is that any network traffic will have to go through these nodes prior to reaching the Sink). We denote by E_{IDS} the energy consumption of the said nodes with the IDS running and with $E_{\overline{\mathrm{IDS}}}$ the energy consumption of the said nodes with no IDS running in the network.

$$\text{Communication overhead} = \frac{E_{\mathrm{IDS}} - E_{\overline{\mathrm{IDS}}}}{E_{\overline{\mathrm{IDS}}}} \tag{3.5}$$

3.3.9.3 Total Energy Consumption in the Network

Energy consumption is a critical issue in WSNs because of the constrained energy resource related to the manufacturer, and sensor nodes are usually powered by two AA batteries. The sensor node in WSN consists of a power unit, sensing unit, processing unit and communication unit. These components work together to gather sensing data from the WSN environment and transmit these data to end user via the base station. Ying et. al Zhou et al. (2011) presented the energy consumption in the processing unit, sensing unit and wireless communication unit. We measure the total energy consumption ΔE_{total} in the network as the difference between the total available energy in the network at the beginning of a simulation and at the end. We denote the initial available energy for sensor i by E^i_{init} and the initial available energy for sensor i by E^i_{final}:

$$\Delta E_{total} = \Sigma_{i \in n}(E^i_{\text{init}} - E^i_{\text{final}}) \tag{3.6}$$

3.4 Proposed IDS Architecture Based on the Cover Set Algorithm

We introduce a hybrid architecture IDS for IoT networks that consists of centralised and distributed IDS agents integrated with a novel placement strategy. In our approach, the IDS architecture can detect and mitigate the effect of node failures. First, we model a WSN with the use of RGG. The RGG model efficiently captures spatial characteristics of the network that are closely related to network connectivity (e.g. inter-dependencies on the existence of wireless links among neighbouring nodes). Then, motivated by how IoT networking protocols, such as RPL, manage and operate the network, we identify the trade-offs between the communication overhead introduced by an IDS and its detection rate of attacks, such as the sinkhole attack. We investigate this trade-off via extended emulations and show that it is not necessary for all nodes to act as IDS agents. Furthermore, we show that the proposed architecture is able to efficiently cope with and mitigate the effects of node failures on the the efficiency of the IDS.

3.4.1 Vertex Cover in Graph Theory

A vertex cover (node cover) in a graph is a subset of vertices in which every edge in the graph is connected to at least one vertex in the cover. The purpose of the vertex cover algorithm is to find the smallest set of vertices that covers the whole graph with the minimum number of nodes. Mathematically, a vertex cover $V\prime$ of a graph $G = (V, E)$ is a subset of V such that $uv \in E \Rightarrow u \in V\prime \vee v \in V\prime$.

For instance, in a given graph $G = \{1, 2, 3, 4, 5, 6\}$, as in Figure 3.3, and a set of edges $E = \{\{1, 2\}, \{1, 3\}, \{2, 4\}, \{3, 4\}, \{4, 5\}, \{4, 6\}\}$, the union of E is equal to G. However, it is possible to cover all the vertices in G with $S = \{1, 4\}$, as shown in Figure 3.4 since vertex number 1 is in touch with vertices $\{2, 3\}$ and vertex number 4 in touch with $\{2, 3, 5, 6\}$.

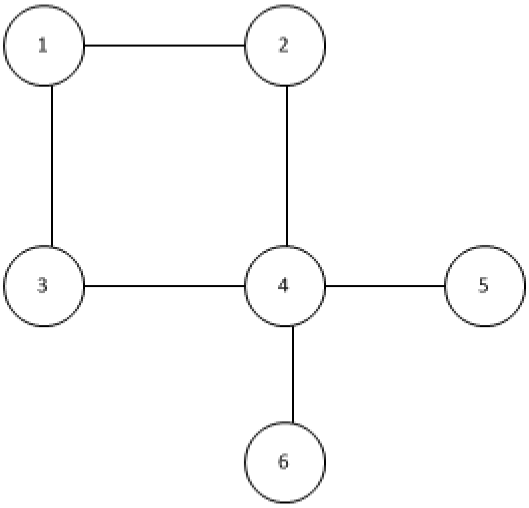


FIGURE 3.3: Indicative Example of Graph

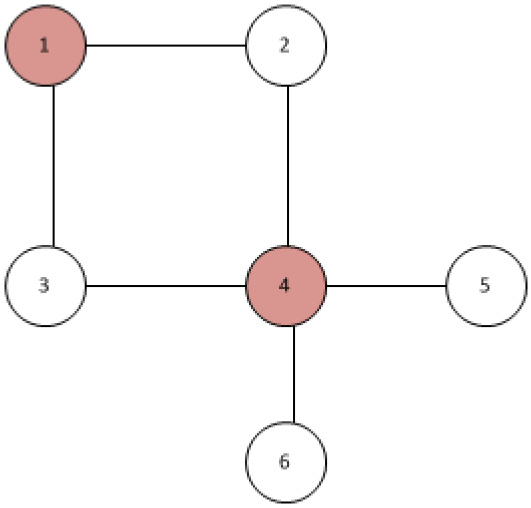


Figure 3.4: The Vertex Cover of the Graph

3.4.2 Description of the proposed IDS Architecture

We introduce a hybrid IDS architecture that consists of a central detection agent located in the base station and a distributed lightweight intrusion detection agent deployed on a subset of the network nodes, as shown in Figure 3.1. The central IDS controls the entire IDS architecture and relevant data from the distributed agents. Each network node that runs an instance of the distributed agent monitors and collects data on local network activity from its 1-hop neighbouring nodes. This implies that not all nodes need to run the IDS agent, but only a subset of them, such that every node in the network has at least one 1-hop neighbour operating as an IDS agent. In graph theory, such a subset is defined as a vertex cover of the corresponding RGG graph that captures the structure of the network. In our approach (Algorithm 1), the subset of nodes that act as IDS agents is selected based on the Vertex-cover algorithm Asgeirsson and Stein (2007) (greedy algorithm) to find a subset of minimum cardinality with proper placement. Moreover, we propose a method to maintain and monitor the distributed IDS agents against node failures. As shown in Algorithm 1, the central IDS frequently checks the set of IDS agents against node failures. In case any of the IDS nodes fail to communicate with the central IDS, the central IDS agent will re-run the proposed algorithm to select a new subset of nodes to act as IDS agents. Centralised IDS and distributed IDS agents are the main components of the proposed architecture; central IDS agent is located in the base station (the sink node) and distributed IDS agents are installed in every single sensor node. A Distributed IDS agent monitors its neighbour nodes (e.g. by monitoring their cluster) monitors and collects data on local network activity from its 1-hop neighbouring nodes for the central IDS agent, which is located at the sink node. Meanwhile, the central IDS controls distributed IDS agents and manages the detection procedure of the entire network. Our proposed architecture is able to efficiently cope with and mitigate the effects of node failure on the efficiency of the IDS.

3.4.2.1 Central IDS

The central IDS agent is located at the base station, and is and is responsible for monitoring the entire network and detecting anomalies. It is also responsible for locating distributed IDS agents based on the vertex-cover algorithm, which allow nodes to act as IDS agents with proper placement to cover the network with minimum number of nodes. The detection engine in the central IDS is responsible for enforcing the detection policy and manages other IDS agents. The centralised IDS is placed on the Sink node or base station. This node represents the border gateway device located on the edge of the WSAN network and is powerful in terms of computing power, memory and energy supplies. Furthermore, the centralised IDS maintains and monitors the distributed IDS agents against node failure. The central IDS frequently checks the set of IDS agents against node failures. In case any of the IDS nodes fail to communicate with the central IDS, the central IDS agent will re-run the proposed Algorithm 6 to select a new subset of the nodes to act as IDS agents.

3.4.2.2 Distributed IDS agent

The distributed IDS agent is the lightweight component of the proposed IDS architecture that is placed on the sensor's side. The distributed IDS agent is responsible for monitoring and analysing the traffic within its cluster or 1-hop neighbours. In addition, it periodically provides the required and relevant detection information to the central IDS.

3.4.2.3 Placement Strategy of the Distributed IDS Agent

The main idea of the proposed method is to improve the energy efficiency and maintain the level of performance of an IDS in IoT and ad-hoc networks. IDS placement strategy is one of the critical factors that effects the efficiency and effectiveness of the IDS security solution. Networking protocols for IoT and wireless networks are designed to address the distributed ad-hoc nature of IoT networks using local network information available to the nodes such as RPL. This network information can be monitored by 1-hop neighbouring nodes. Therefore, for a given set of neighbouring nodes, it suffices that only one of them is actively collecting and reporting relevant information to the Sink. This greatly reduces the number of nodes that need to operate as IDS agents, thus mitigating any scalability and performance issues. This subset of nodes in the network can monitor the entire network properly. The subset of nodes that act as IDS agents is selected based on the vertex-cover algorithm Asgeirsson and Stein (2007) (greedy algorithm)6 to find a subset of minimum cardinality with proper placement.

Algorithm 2 IDS Agent Placement

```
Require: $N \leftarrow$ the list of nodes
Require: $S \leftarrow$the list of neighbour nodes
Require: $A \leftarrow$IDS Agents
  while $N \neq \emptyset$ do
      Select a set of neighbours $\in S$ that maximise $S \cap N$
      $N \leftarrow N - S$
      $A \leftarrow A \cup \{S\}$
  end while
  return $A$
```

3.4.2.4 Resiliency algorithms

A method to maintain and monitor the distributed IDS agents against node failure is proposed. As shown in Algorithm 3, the central IDS frequently checks the set of IDS agents against node failure. In case any of the IDS nodes fail to communicate with the central IDS, the central IDS agent will re-run our proposed algorithm to select a new subset of nodes to act as IDS agents in order to efficiently cope with and mitigate the effects of node failure on the the efficiency of the IDS.

Algorithm 3 Detect Sinkhole Attacks and Placement/Resilience Methods

```
Require: $N \leftarrow$ the list of nodes
Require: $S \leftarrow$ the list of neighbour nodes
Require: $A \leftarrow$ the list of selected Node to run as an IDS agent
Require: $A\prime \leftarrow$ A
while $A = \emptyset$ OR $A \neq A\prime$ do
    while $N \neq \emptyset$ do
        Select a set of neighbours $\in S$ that maximise $S \cap N$
        $N \leftarrow N - S$
        $A \leftarrow A \cup \{S\}$
        $A\prime \leftarrow A$
        return $A, A\prime$
    end while
end while
for Node in A do
    for Node in S do
        if (Node.Rank+IDSagentNodeRank
< Node.Parent.rank) then
            Node.fault++
        end if
    end for
end for
for Node in N do
    if Node.fault>Threshold then
        Raise Alarm
    end if
end for
```

3.4.3 The Network Model

We model such networks as RGG, a formal model capturing underlying structural properties of the network. We then introduce a novel IDS architectural approach, in which only a minimum subset of the nodes act as IDS agents. These nodes are able to monitor the network and detect attacks at the networking layer in a collaborative manner by monitoring 1-hop network information provided by routing protocols, such as RPL. We consider the random placement of the sensor nodes in the network area by a RGG. RGGs are constructed by n vertices that are placed at random in the area of interest $[0,1]^2$. An edge (u, v) exists iff the Euclidean distance of vertices u and v is at most r, where r is the radius (wireless communication range) r of the sensors. RGG represents a realistic model for WSANs since it captures the extent of the communication structure of real networks including spatial aspects . Specifically, we consider an area $\mathcal{A} \subset \mathbb{R}^2$ in two-dimensional space. An instance of the *RGG model* $\mathcal{G}(\mathcal{X}_n; r)$ is constructed as follows: for n points $\mathcal{X}_n$ is uniformly place at random in $\mathcal{A}$. The set $V = \mathcal{X}_n$ is the set of vertices of the graph, and we connect two vertices iff their Euclidean distance is at most r. For any vertex $v \in V$, we denote by $N(v)$ the set of neighbours of v and by $\deg(v) = |N(v)|$ its degree. Further, we denote by $\|u - v\|$ the Euclidean distance between the points corresponding to vertices v, u. The RGG model provides us with a formal tool of constructing and characterising networks as "*sparse*", "*dense*" or "*normal*".

3.4.4 Implementation Plan

As with the random placement algorithm described earlier, we apply this approach using SVELTE Raza et al. (2013).

The RPL routing protocol maintains the routing paths between the Sink and the rest of the network nodes by constructing a global network topology using the Destination Oriented Directed Acyclic Graph (DODAG). Initially, the Sink broadcasts messages to its immediate neighbouring nodes, which in turn reiterate the process to their neighbouring nodes lying further away in the network. The process is run recursively and eventually results in each node being assigned a rank that depends on its actual hop-distance to the Sink as well as the link quality between neighbouring nodes (as measured by an objective function, such as the ETX metric). In SVELTE, the 6Mapper periodically collects these ranks to reconstruct the DODAG centrally at the Sink in order to monitor the network against relevant attacks - like sinkhole attack - by detecting corresponding anomalies, as shown in Algorithm 1; for example, if the rank of a node significantly deviates from the rank of its neighbours. The central IDS periodically broadcasts requests requests that are five bytes in length, while the IDS agents in each individual sensor node send responses that are 17 bytes long.

3.4.5 Evaluation Environment

We ran our experiments in two parts. The first part of the experiment was conducted to evaluate the efficiency and effectiveness of our proposed IDS architecture. Our proposed approach aims to find a minimum subset of nodes that are able to monitor the network efficiently. The second part of our experiment was conducted to evaluate our algorithm, which was integrated with the proposed IDS architecture in order to mitigate IDS agent failure. For both experiments, we used Cooja Osterlind et al. (2006), which provides a detailed cross-layer simulation for WSNs running the 6LoWPAN stack. We consider three qualitatively distinct network densities as they are indicated by the RGG model. Particularly, we consider a network area $\mathcal{A} = [0, 100]^2$, where n sensor motes are deployed uniformly at random, for $n \in \{32, 64, 128\}$. We get three network setups in which r is (a) almost equal to (b) $\times 1.5$ and (c)$\times 2$ the connectivity threshold, thus resulting in (a) *sparse*, (b) *normal* and (c) *dense* networks (see Appendix A).

For each network density, we consider algorithm 6, where the nodes act as IDS agents selected based on the cover set algorithm, namely, the greedy algorithm. In each case, we set 10% of the node population to act as malicious nodes deploying sinkhole attacks by exploiting the rank mechanism of RPL. Any remaining nodes are regular nodes. Furthermore, in the second part of our experiment we gradually drop off some nodes that perform as IDS agents during the simulation to evaluate the redundancy of our approach.

For each network configuration, we also run a scenario with no nodes operating as IDS nodes. For each scenario, we create 10 random instances of the network; this allows us to effectively mitigate any issues in our simulations that might occur due to the random network topology (in other words, we sample the space of RGG instances). For each instance, we run 10 iterations of simulating the network operation for a simulation time of 3600 seconds where nodes generate and transmit data approximately every second. For each scenario and each performance metric, we compute average values and 95% confidence intervals.

3.4.6 Attacker Model

In this research, we consider network layer type of attacks. Therefore, we consider in conducted experiments for this research sinkhole attack on RPL routing protocol for constrained networks as an attacker model and an use case. We consider the IDS architecture and the IDS agents' placement strategies regardless of the detection accuracy. We investigate the underline cause of the communication and energy consumption overheads introduced to constrained networks by Intrusion Detection System. The proposed IDS architecture and the IDS agent placement strategies can be extended to detect attacks in other routing protocols.

A sinkhole attack is an insider attack, in which an attacker compromises a node inside the network and launches an attack. The malicious node in this attack advertises a beneficial path in order to attract its neighbour nodes to route the traffic through it based on the routing metric used in the routing protocol. This kind of attack does not disrupt the network operation but it affects the integrity and confidentiality aspects Kamble et al. (2017).

A sinkhole attack could be deployed through a compromised node by having this node falsely announcing a significantly smaller rank to its neighbours. This will cause the neighbouring nodes to assume that its distance to the Sink is much smaller than the actual one, thus directing all network traffic through the compromised node.

3.4.7 Evaluation Metrics

The aim of our proposed approach is to properly address the constrained nature of WSNs, especially in terms of resources (computational power, available energy, limited memory, etc.). Therefore, we focus on IDS architecture with a special emphasis on IDS placement since it is an essential factor in communication overheads and energy consumption in constraint networks. Thus, we define evaluation metrics that capture the trade-off between energy consumption/communication overhead with the detection rate of malicious activities.

3.4.7.1 Detection Rate

We define the detection rate as the number of true positive detections of malicious nodes over the total number of malicious nodes in the network.

$$\text{Detection rate} = \frac{\text{number of true positive detections}}{\text{total number of malicious nodes}} \tag{3.7}$$

3.4.7.2 Communication Overheads

The communication unit in WSN nodes is responsible for sending and receiving nodes data. The transceiver usually has several operation states: $Tx, Rx, Off, Idle$ Sleep and Clear Chanel Assessment/Energy Detect (CCA/ED) (Zhou et al. 2011)Ishmanov et al. (2011)Xu et al. (2012)Huang et al. (2009)Chouhan et al. (2009). The transceiver energy consumption can be calculated by summing the state energy consumption and state-transition energy consumption.

The energy state consumption can be calculated as follows Zhou et al. (2011)Pantazis et al. (2013)Khanmirza and Yazdani (2016):

$$E_{trans-state} = E_{TX} + E_{RX} + E_{Idle} + E_{sleep} + E_{CCA} \tag{3.8}$$

Energy consumption in data communication is the most expensive factor in WSN energy dissipation. For instance, the energy cost of transmitting 1 KB a distance of 100 m is approximately the same as that for executing three million instructions using a 100 (MIPS)/W processor(Zhou et al. 2011)Bouabdallah et al. (2008)Abo-Zahhad et al. (2015)Alkalbani et al. (2013). Therefore, we define the communication overhead as the additional volume of data communication introduced in the network as a result of the operation of the IDS. We follow the practice of Raza et al. (2013) and monitor this metric only to the 1-hop neighbouring nodes of the Sink (the rationale is that any network traffic will have to go through these nodes prior to reaching the Sink). We denote by E_{IDS} the energy consumption of the nodes with the IDS running and with $E_{\overline{\mathrm{IDS}}}$ the energy consumption of the nodes with no IDS running in the network.

$$\text{Communication overhead} = \frac{E_{\mathrm{IDS}} - E_{\overline{\mathrm{IDS}}}}{E_{\overline{\mathrm{IDS}}}} \tag{3.9}$$

3.4.7.3 Total Energy Consumption in the Network

Energy consumption is the critical issue in WSNs because of the constrained energy resource, which is related to the manufacturer, and sensor nodes are usually powered by two AA batteries. The sensor node in WSNs consists of a power unit, sensing snit, processing unit and communication unit. These components work togather to collect sensing data about the WSN environment and transmit these data to the end user through the base station. Ying et. al Zhou et al. (2011) identified the energy consumption in the processing unit, sensing unit and wireless communication unit. We measure the total energy consumption ΔE_{total} in the network as the difference between the total available energy in the network at the beginning of a simulation and at the end. We denote initial available energy for sensor i by E^{i}_{init} and the initial available energy for sensor i by E^{i}_{final}.

$$\Delta E_{total} = \Sigma_{i \in n}(E^{i}_{\text{init}} - E^{i}_{\text{final}}) \tag{3.10}$$

3.5 Conclusion

This chapter describes a hybrid IDS architecture that combines centralised and distributed architectures for ad-hoc and wireless networks. This combination makes it possible to reduce the massive communication required for the distributed IDS monitor its immediate neighbours nodes and report relevant detection information to the Sink. Moreover, this combination uses the advantages of both the centralised and distributed detection models, where the centralised model provides better control on the detection procedure than the distributed model. In addition, the distributed model is beneficial in communication overheads reduction. The distributed model is based on a small set of sensor nodes, which function as a local sink and monitor the neighbour nodes in its 1-hop or cluster. Dynamic distribution and randomisation of the IDS agent placement among the various sensor nodes has been considered in the proposed IDS architecture. Furthermore, the distributed model placement based on the vertex cover algorithm from graph theory has been used as an optimisation solution to enhance the placement and resilience of the distributed model.

Chapter 4

Evaluation of the Proposed IDS Architecture

4.1 Introduction

This chapter reports on the detailed experimental evaluation and results of the proposed IDS architecture for WSN and ad-hoc networks based on random placement using RGG.

The trade-off between the communication and energy overheads of an IDS (as captured by the number of active IDS agents in the network) and the performance of the system in terms of successfully identifying attacks is studied. As outlined in the previous chapter, we model such networks as a RGG, a rigorous approach that formally expresses the underlying structural properties of the network. Then, we introduce a novel IDS architectural approach by having only a subset of the nodes function as IDS agents. These nodes are able to efficiently detect attacks at the networking layer in a collaborative manner by monitoring locally available network information provided by IoT routing protocols, such as RPL.

This Chapter presents the experimental evaluation of the proposed approach through simulation. We evaluate the efficiency and effectiveness of the proposed approach and discuss the experimental results and findings.

4.2 The Proposed IDS

As mentioned in the previous chapters, RPL establishes and maintains routing paths between the Sink and the rest of the network nodes by constructing a global tree-like network structure in a distributed way; the Destination Oriented Directed Acyclic Graph (DODAG). The process is initiated by the Sink broadcasting exploratory messages to its immediate neighbouring nodes, which in turn reiterates the process to their neighbouring nodes lying further away in the network. The process is run recursively and eventually results in each node being assigned a rank that depends on its actual hop-distance to the Sink as well as the link quality between neighbouring nodes (as measured by an objective function, such as the ETX metric). In SVELTE, the 6Mapper periodically collects these ranks to reconstruct the DODAG centrally at the Sink in order to monitor the network against relevant attacks - like sinkhole - by detecting corresponding anomalies; for example, if the rank of a node significantly deviates from the rank of its neighbours.

While for each individual node the introduced communication overhead may be small (the messages carrying the 6Mapper requests are five bytes long while each response from the nodes is 17 bytes long), engaging each individual node in the process introduces a communication overhead that is proportional to the size of the network. This poses significant scalability issues and adversely affects the connectivity and availability of the network because, in multi-hop peer-to-peer networks, nodes closer to the Sink also serve traffic coming from the rest of the network.

They key idea behind the proposed approach is that networking protocols designed to address the distributed ad-hoc nature of peer-to-peer IoT networks (such as IPv6-enabled WSNs) make use of network information that is *locally available* to the nodes, such as in the case of RPL. This network information can be easily shared or even be monitored by 1-hop neighbouring nodes. Therefore, for a given set of neighbouring nodes, it suffices that only one of them is actively collecting and reporting relevant information to the Sink. This greatly reduces the number of nodes that need to operate as IDS agents, thus mitigating any scalability and performance issues.

In this work, we focus on experimentally investigating and evaluating our approach by extending SVELTE (the current state-of-the-art IDS for IPv6-enabled WSNs). In particular, we focus on evaluating the trade-off between the potentially reduced accuracy of the IDS in detecting attacks (due to the smaller number of active IDS agents in the network) versus the reduced communication overhead and increased energy efficiency of the network.

Algorithm 4 Detect Sinkhole Attacks

```
Require: M ← the list of IDS agents nodes
Require: N ← the list of Regular nodes
for Node in M  do
    for Node in N do
        if (Node.Rank+IDSagentNodeRank
< Node.Parent.rank) then
            Node.fault=Node.fault+1
        end if
    end for
end for
for Node in N do
    if Node.fault>Thershold then
        Alarm
    end if
end for
```

4.3 Experimental Evaluation

4.3.1 Implementation Plan

We ran our experiments using the Cooja Osterlind et al. (2006) emulator, which provides a detailed cross-layer simulation for WSNs running the 6LoWPAN stack. We consider three qualitatively distinct network densities as they are indicated by the RGG model. In particular, we consider a network area $\mathcal{A} = [0, 100]^2$, where n sensor motes are deployed uniformly at random, for $n \in \{32, 64, 128\}$. Following equation 3.2, for each value of n, the corresponding network connectivity threshold is $r_c : \{18.5; 14.3; 11\}$. Therefore, by setting the sensors' communication range to be $r = 20$, we get three network setups, where r is (a) almost equal to, (b) $\times 1.5$ and (c)$\times 2$ the connectivity threshold, thus resulting in (a) *sparse*, (b) *normal* and (c) *dense* networks. Figures ??, ?? and ?? provide a visual representation of the various network densities.

For each network density, we consider five scenarios, where the percentage of nodes acting as IDS agents is 100%, 80%, 60%, 40% and 20% of the total population (yellow nodes in the corresponding figures, see Appendix ??). Furthermore, in each case, we set 10% of the node population to act as malicious nodes (nodes in purple) deploying sinkhole attacks by exploiting the rank mechanism of RPL. Any remaining nodes are regular nodes (nodes in green). For each network configuration, we also run a scenario with no nodes operating as IDS nodes. For each scenario we create 10 random instances of the network; this allows us to effectively mitigate in our simulations any issues that might occur due to the random network topology (in other words we sample the space of RGG instances). For each instance, we run 10 iterations of simulating the network operation for a simulation time of 3600 seconds, in which the nodes generate and transmit data approximately every second. For each scenario and each performance metric, we compute average values and 95% confidence intervals.

Our findings demonstrate a strong concentration around the mean and are therefore deemed statistically significant.

4.3.2 Evaluation Metrics

In the following subsections, we define the metrics that are used to evaluate our proposed IDS architecture.

4.3.2.1 Detection Rate

We define the detection rate as the number of true positive detections of malicious nodes over the total number of malicious nodes in the network.

$$\text{Detection rate} = \frac{\text{number of true positive detections}}{\text{total number of malicious nodes}} \tag{4.1}$$

4.3.2.2 Communication Overhead

We define the communication overheads as the additional volume of data communication introduced in the network as a result of the operation of the IDS. We follow the practise of Raza et al. (2013) and monitor this metric only to the 1-hop neighbouring nodes of the Sink (the rationale is that any network traffic will have to go through these nodes prior to reaching the Sink). We denote by E_{IDS} the energy consumption of the said nodes with the IDS running and with $E_{\overline{\text{IDS}}}$ the energy consumption of the nodes with no IDS running in the network. Then,

$$\text{Communication overhead} = \frac{E_{\text{IDS}} - E_{\overline{\text{IDS}}}}{E_{\overline{\text{IDS}}}} \tag{4.2}$$

4.3.2.3 Total Energy Consumption in the Network

We measure the total energy consumption ΔE_{total} in the network as the difference between the total available energy in the network at the beginning of a simulation and at the end. We denote initial available energy for sensor i by E^{i}_{init} and the initial available energy for sensor i by E^{i}_{final}. Then,

$$\Delta E_{total} = \Sigma_{i \in n}(E^{i}_{\text{init}} - E^{i}_{\text{final}}) \tag{4.3}$$

4.3.3 Results

Figure 4.1 shows that in sparse networks the detection rate remains as high as 85% for the scenarios where 100% and 85% of the node population operates as an IDS agent. However, the detection rate drops to 60% for 60% of the population as IDS agents, and it continues to drop further as the percentage of the agents is reduced. This demonstrates that the IDS performance in sparse networks quickly drops due to the fact that areas of the network remain un-monitored. We note, however, that there is a certain level of resilience for high percentages of IDS agents.

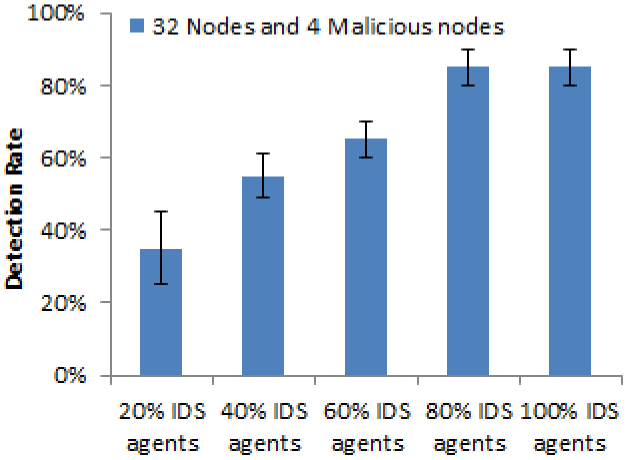


FIGURE 4.1: Sparse network

Figure 4.2 shows the findings for networks of normal density. We note two points. First, the IDS demonstrates a greater degree of resilience, as it achieves high detection rates even for percentages of IDS agents as low as 40% of the node population. Second, when 100% of the nodes are IDS agents the simulation was not completed due to the fact that the network was rendered disconnected, as the motes lying close to the Sink were not able to handle the increased network traffic. This highlights the network strain that even light-weight IDSs introduce.

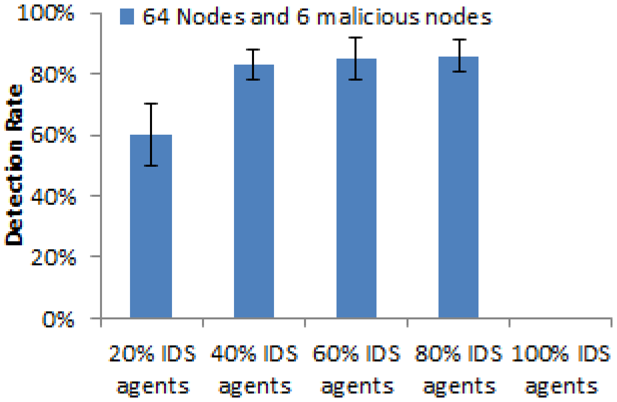


FIGURE 4.2: Normal density network

This is also the reason why other works in the literature on IoT and WSN IDS limit their simulation studies to networks with small populations. Figure 4.3 further highlights these findings, as the simulations failed to complete for scenarios with large numbers of IDS agents (percentages of 100% and 80%). Additionally, in dense networks the detection rate of the IDS remained at very high levels (approximately 80%-85%).

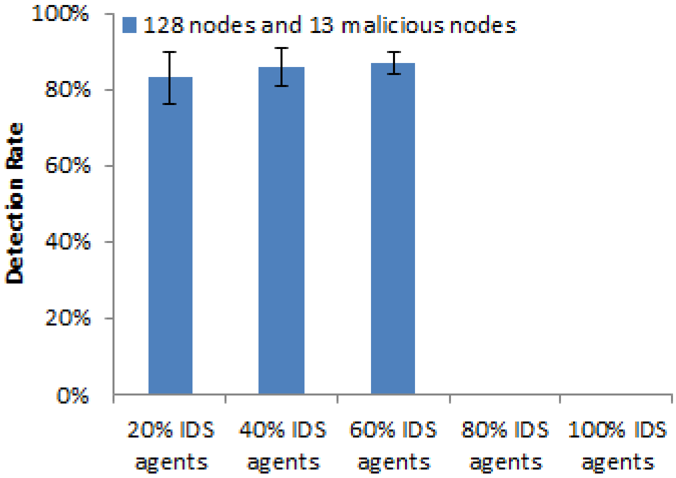


FIGURE 4.3: Dense network

At this point, we make another important observation. For all three network densities, the detection rate of the IDS starts to deteriorate significantly (and as shown in Figure 4.1, proportionally to the reduction in IDS agents percentage) once the absolute number of IDS agents in the network drops below a constant threshold; in this case, below 25 IDS nodes (corresponding to 32·80%; 64·40%; 128 · 20%). This implies that only *a constant number of IDS agents* is needed to effectively and efficiently monitor the network.

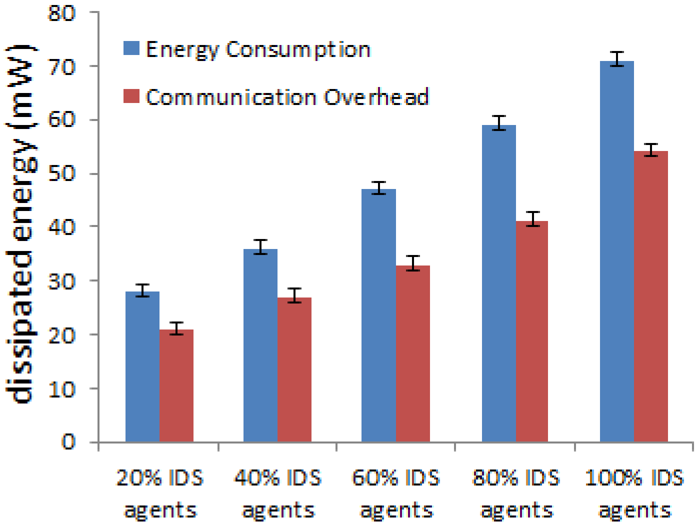


FIGURE 4.4: Energy consumption and communication overhead for the entire network of sparse networks

This is a very strong indication that the efficiency of a hybrid/distributed IDS for peer-to-peer ad-hoc networks is independent of the number of nodes but related to *underlying fundamental properties of the network.* Following our network modelling with the use of RGG, we notice that this property is the size of the minimum vertex cover of the corresponding RGG instance. We intend to investigate this in our future work employing more formal and rigorous methods from graph theory.

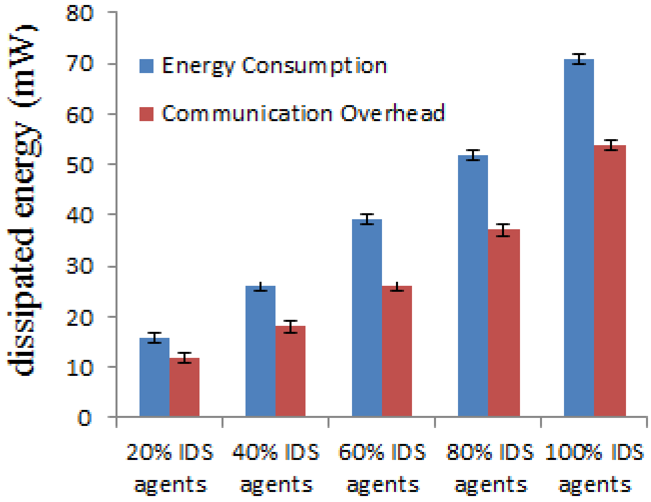


FIGURE 4.5: Energy consumption and communication overhead introduced by the IDS of sparse networks

Figures 4.4 and 4.5 show the average energy consumption and the communication overheads introduced by the IDS of sparse networks. We note that for all five scenarios (the percentage of nodes acting as IDS agents is 100%, 80%, 60%, 40% and 20% of the total population) the overall communication overhead and energy consumption of the network is proportional to the number of nodes operating as IDS agents. Correlating to Figure 4.1, the detection rate in the scenario where 80% of nodes acting as IDS agents remain at high level as 100% IDS agents scenario. This shows that our approach achieved lower communication overhead and energy consumption while maintaining a high detection rate.

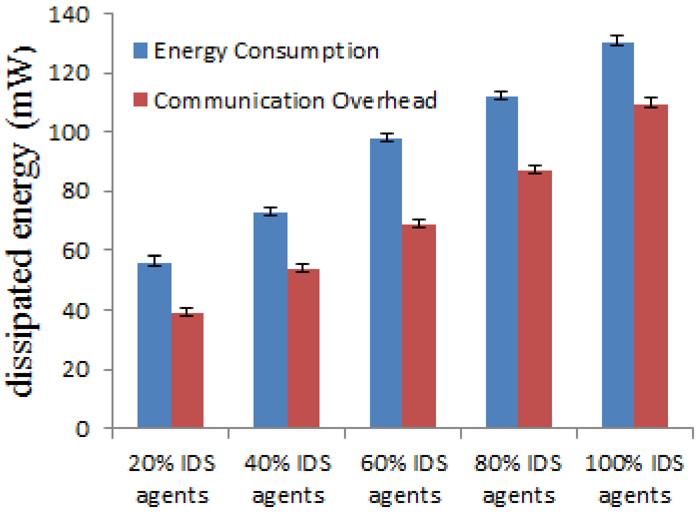


FIGURE 4.6: Energy consumption and communication overhead for the entire moderate dense networks

Figures 4.6 and 4.9 show the average energy consumption and the communication overhead of moderately dense networks. The communication overhead and energy consumption of networks is also proportional to the number of nodes operating as IDS agents for all five scenarios (the percentage of nodes acting as IDS agents is 100%, 80%, 60%, 40% and 20% of the total population). In Figure 4.2, the IDS achieves high detection detection rates at percentages of 40%, 60% and 80% of the node population. However, the detection rate drops quickly in scenarios with 20% of nodes acting as IDS agents. In moderate dense networks, the IDS with 40% agents achieved high detection rate with lower energy consumption compared with the other scenarios.

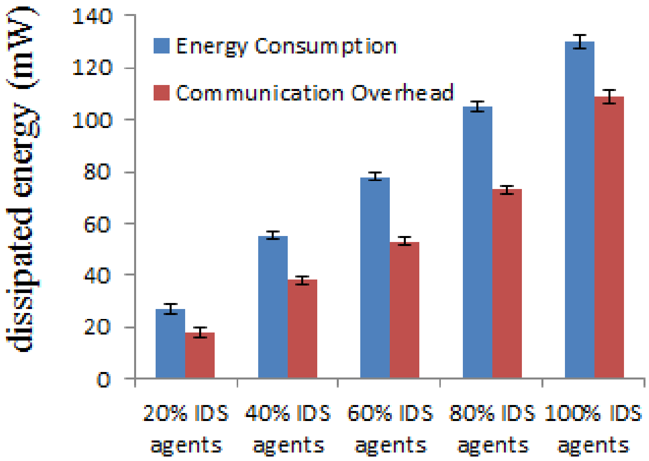


FIGURE 4.7: Energy consumption and communication overhead introduced by the IDS of moderate dense networks

Figures 4.7 and 4.8 show the average energy consumption and communication overhead of dense networks. The IDS achieves high detection rates as shown in Figure 4.3 for for IDS percentages of 20%, 40% and 60% of of the node population. However, the simulations with 80% and 100% failed to complete due to the massive communication overheads generated by the large number of IDS agents. For dense networks, the IDS with 20% of the nodes population performing as IDS agent achieved lower energy consumption scenarios while maintaining a high detection rate.

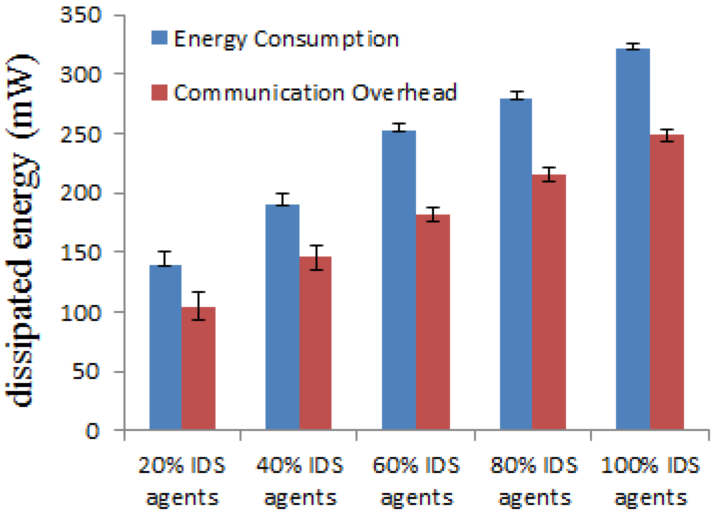


FIGURE 4.8: Energy consumption and communication overhead for the entire network of dense networks

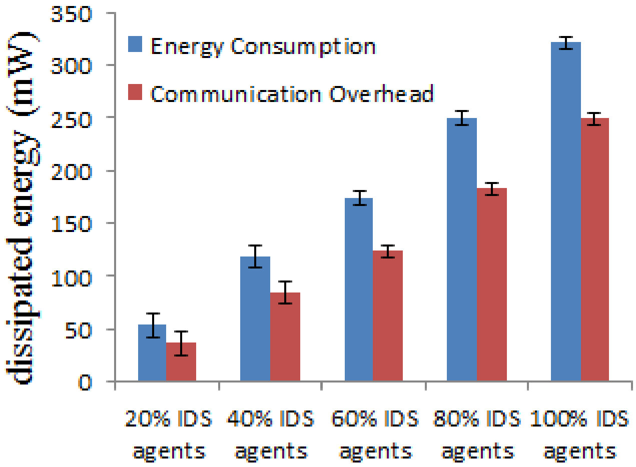


Figure 4.9: Energy consumption and communication overhead introduced by the IDS of dense networks

This shows the massive gains that can be achieved by fine-tuning the trade-off between energy efficiency and the achieved high detection rate as a result of using a constant number of IDS agents. Detailed experimental evaluation demonstrates significant performance gains in terms of communication overheads and energy dissipation while maintaining high detection rates. We also show that the performance of our IDS in ad-hoc networks not only relies on the size of the network but also on fundamental underlying network properties, such as the network topology and the average degree of the nodes.

4.4 Conclusion

This work studies efficient and lightweight IDS for ad-hoc networks via the prism of IPv6-enabled WSN. First, it provides a formal model for WSNs with the using RGG, a graph-theoretical model that captures the spatial characteristics of WSNs, such as interdependencies on the existence of wireless links among neighbouring nodes. Then, motivated by the operation of IoT-specific networking protocols, such as RPL, we focus on network attacks, such as the sinkhole or man-in-the-middle attack. We identify and attempt to optimise the trade-off between energy efficiency and communication overhead on one hand and the IDS detection rate on the other. By leveraging upon the distributed nature of such protocols and locally available network information, we propose a method for IDS that requires only a subset of the nodes to operate as IDS agents.

We extend state-of-the-art IDS for WSNs by integrating our method and conduct our performance evaluation via extensive emulations. We consider various network densities (as these are formally defined through the RGG model) and show that 1) the IDS detection rates remain at very high levels (around 85%) even with a subset of the nodes as IDS agents; 2) the required number of IDS agents in the network in order to achieve these levels is independent from the network population and in fact *constant*; 3) the energy consumption and communication overhead introduced by the IDS is proportional to the number of IDS agents, thus our method allows for massive energy gains while not affecting the detection rate of the IDS.

.

Chapter 5

Offering Resilience to an Intrusion Detection System through IDS Agent Placement Optimisation

5.1 Introduction

This chapter explores an approach of improving the placement strategy and the resilience of an IDS through the deployment of the IDS agents using a placement optimisation algorithm. Node failures can be commonplace in sensor networks. Sensor nodes are constrained in power resources, therefore they are vulnerable to frequent topology changes due to nodes failures. In the context of this research, resilience refers to the ability of the network to recover its IDS architecture and capabilities in order to avoid any loss of IDS agent that may affect the detection rate.

In essence, there are two approaches for offering resilience in sensor network environments. One would be through IDS agent redundancy, i.e. to increase the number of IDS agents so that this would be above the connectivity threshold. However, this solution has a few drawbacks. First, the overheads will be constantly higher, outweighing the benefits in the long term, particularly if the nodes have a relatively low probability of failure. Second, there is no guarantee that the detection rate will not drop as the placement strategy for the redundant nodes will not be optimum in the first place.

Another strategy for injecting resiliency into the IDS would be to run a placement protocol upon an IDS node failure. This chapter evaluates empirically this approach, adopting an IDS placement strategy using the vertex-cover algorithm described in 3.

Building upon the findings of Chapter 4, we present a new placement strategy for the distributed model of the proposed IDS architecture. The aim is to find a minimal cardinality of nodes that are able to efficiently detect attacks at the networking layer, and also able to detect and mitigate the effect of node failures.

This Chapter presents the studies and experimental evaluation of the proposed approach through simulation, following previous work that as described in Chapter 4. First, we evaluate the efficiency and effectiveness of the proposed approach, and then we evaluate its resilience against frequent topology changes due to node failures. Eventually, we discuss the experimental results and highlight the key findings of the studies.

5.2 Vertex Cover in Graph Theory

Vertex cover algorithms have been applied on numerous real-world problems and domains (electrical engineering, circuit design, network flow and so forth) as an optimisation solution since the aim is to find a subset with minimal cardinality.

A vertex cover (node cover) in a graph is a subset of Vertices where every edge in the graph is connected to at least one vertex in the cover. The goal of the vertex cover is to find the smallest set of vertices that covers the whole graph with the minimum number of nodes. Formally, a vertex cover $V\prime$ of a graph $G = (V, E)$ is a subset of V such that $uv \in E \Rightarrow u \in V\prime \vee v \in V\prime$.

For instance, given a graph $G = \{1, 2, 3, 4, 5, 6\}$ as in Figure 3.3 and a set of edges $E = \{\{1, 2\}, \{1, 3\}, \{2, 4\}, \{3, 4\}, \{4, 5\}, \{4, 6\}\}$, obviously, the union of E is equal to G. However, it is possible to cover all the vertices in G with $S = \{1, 4\}$, as shown in Figure 3.4 since vertex number 1 in touch with vertices $\{2, 3\}$ and vertex number 4 is in touch with $\{2, 3, 5, 6\}$.

5.3 The Proposed Placement IDS Architecture

We propose an optimising placement algorithm of IDS architecture consisting of a central detection agent located in the base station and a distributed lightweight intrusion detection agent deployed on a subset of the network nodes, as shown in algorithm 2, based on *vertex cover optimising algorithms.* The central agent manages the entire detection process and collects relevant data from the distributed agents. Each network node that runs an instance of the distributed agent monitors and collects data on local network activity from its 1-hop neighbouring nodes. This implies that not all nodes need to run the IDS agent, but only a subset of them, such that every node in the network has at least one 1-hop neighbour running the IDS agent. In graph-theoretical terms, such a subset of nodes are selected to be IDS agents by vertex-cover greedy algorithm of the corresponding RGG graph, capturing the structure of the network. This also implies that there exists a minimum set of nodes that are able to efficiently monitor the network without compromising the performance of the IDS with respect to proper placement of IDS agent nodes. This set corresponds to the minimum or close to vertex cover for the corresponding RGG graph.

In this optimising approach, we also consider that the random uniform placement of the sensors inside the network area is abstracted by a RGG. RGG is formed by n vertices that are placed uniformly at random in the $[0, 1]^2$ square. An edge (u, v) exists iff the Euclidean distance of vertices u and v is at most r, where r corresponds to the wireless communication radius r of the sensors. This holds assuming a disc radio model; two sensors can communicate with each other iff each one lies inside the communication range of the other. RGG have an important property: unlike other random graphs, like $G_{n,p}$, the edges are not statistically independent of each other. That is, the existence of an edge (u, v) is not independent of the existence of edges (u, w) and (w, v). This property makes RGG a realistic model for WSANs, as it captures to a great extent the communication structure of real networks (at least their spatial aspects).

Algorithm 5 IDS Agent Placement

```
Require: $N \leftarrow$ the list of nodes
Require: $S \leftarrow$ the list of neighbour nodes
Require: $A \leftarrow$ IDS Agents
while $N \neq \emptyset$ do
    Select a set of neighbours $\in S$ that maximise $S \cap N$
    $N \leftarrow N - S$
    $A \leftarrow A \cup \{S\}$
end while
return $A$
```

5.4 The Proposed Resilient IDS

We consider the hybrid IDS architecture presented in Chapter 3, that consists of a central detection agent located in the base station and a distributed lightweight intrusion detection agent deployed on a subset of the network nodes. The central IDS controls the entire IDS architecture and processes relevant data from the distributed agents. Each network node that runs an instance of the distributed agent monitors and collects data on local network activity from its 1-hop neighbouring nodes. This implies that not all nodes need to run the IDS agent, but only a subset of them, such that every node in the network has at least one 1-hop neighbour running the IDS agent. In graph theory, such a subset is defined as a vertex cover of the corresponding RGG graph that captures the structure of the network. In our Algorithm 6, the subset of nodes that function as IDS agents is selected based on vertex-cover Algorithm (greedy algorithm) to find a subset with a minimum cardinality with proper placement.

Moreover, we proposed a method to maintain and monitor the distributed IDS agents against node failures. As shown in Algorithm 1, the central IDS frequently checks the of IDS agents against node failure. the event that any IDS node fail to communicate with the central IDS, the central IDS agent re-runs our proposed algorithm to select a new subset (cover set that covers other nodes) of nodes to act as IDS agents.

Algorithm 6 Detect Sinkhole Attacks

```
Require: $N \leftarrow$ the list of nodes
Require: $S \leftarrow$the list of neighbour nodes
Require: $A \leftarrow$ the list of selected Node to run as an IDS agent
Require: $A\prime \leftarrow$ A
while $A = \emptyset$ OR $A \neq A\prime$ do
    while $N \neq \emptyset$ do
        Select a set of neighbours $\in S$ that maximise $S \cap N$
        $N \leftarrow N - S$
        $A \leftarrow A \cup \{S\}$
        $A\prime \leftarrow A$
        return $A, A\prime$
    end while
end while
for Node in A do
    for Node in S do
        if (Node.Rank+IDSagentNodeRank
< Node.Parent.rank) then
            Node.fault=Node.fault+1
        end if
    end for
end for
for Node in N do
    if Node.fault>Threshold then
        Alarm
    end if
end for
```

5.5 Performance Evaluation

5.5.1 Simulation Setup

We ran our experiments in two parts. The first part was conducted to evaluate the energy efficiency and effectiveness of our proposed approach. The second part of our experiments was run to study the resilience and redundancy of our approach. For both experiments, we use the Cooja Osterlind et al. (2006) emulator, which provides a detailed cross-layer simulation for WSNs running the 6LoWPAN stack. We consider three qualitatively distinct network densities, as they are indicated by the RGG model. Specifically, we consider a network area $\mathcal{A} = [0, 100]^2$, where n sensor motes are deployed uniformly at random, for $n \in \{32, 64, 128\}$. We get three network setups in which r is (a) almost equal to (b) $\times 1.5$ and (c)$\times 2$ the connectivity threshold, thus resulting in (a) *sparse*, (b) *normal* and (c) *dense* networks (see Appendix **??**).

For each network density, we consider Algorithm 1 in which the nodes acting as IDS agents are selected based on the cover set algorithm, namely, the greedy algorithm. With each case, we set 10% of the node population to act as malicious nodes deploying sinkhole attacks by exploiting the rank mechanism of RPL. Any remaining nodes are regular nodes. Furthermore, in the second part of our experiment we gradually drop off some nodes that perform IDS agent during the simulation to evaluate the resilience of the proposed approach.

For each network configuration, we also run a scenario with no nodes operating as IDS nodes. For each scenario, we create 10 random instances of the network. This allows us to effectively mitigate any issues in our simulations due to the random network topology (in other words, we sample the space of RGG instances). For each instance, we run 10 iterations simulating the network operation for a simulation time of 3600 seconds where nodes generate and transmit data approximately every second. For each scenario and each performance metric, we compute average values and 95% confidence intervals.

5.5.2 Evaluation Metrics

In the following subsections, we define the metrics that are used to evaluate our proposed IDS architecture.

5.5.2.1 Detection Rate

We define the detection rate as the number of true positive detections of malicious nodes over the total number of malicious nodes in the network.

$$\text{Detection rate} = \frac{\text{number of true positive detections}}{\text{total number of malicious nodes}} \tag{5.1}$$

5.5.2.2 Communication Overheads

We define the communication overhead as the additional volume of data communication introduced in the network as a result of the operation of the IDS. We follow the practise of Raza et al. (2013) and monitor this metric only to the 1-hop neighbouring nodes of the Sink (the rationale is that any network traffic will have to go through these nodes prior to reaching the Sink). We denote by E_{IDS} the energy consumption of the nodes with the IDS running and with $E_{\overline{\mathrm{IDS}}}$ the energy consumption of the nodes with no IDS running in the network. Then,

$$\text{Communication overhead} = \frac{E_{\mathrm{IDS}} - E_{\overline{\mathrm{IDS}}}}{E_{\overline{\mathrm{IDS}}}} \tag{5.2}$$

5.5.2.3 Total Energy Consumption in the Network

We measure the total energy consumption ΔE_{total} in the network as the difference between the total available energy in the network at the beginning of a simulation and at the end. We denote initial available energy for sensor i by E^{i}_{init} and the initial available energy for sensor i by E^{i}_{final}. Then,

$$\Delta E_{total} = \Sigma_{i \in n}(E^i_{\text{init}} - E^i_{\text{final}}) \tag{5.3}$$

5.5.3 Simulation Findings

5.5.3.1 Results of Sparse Networks

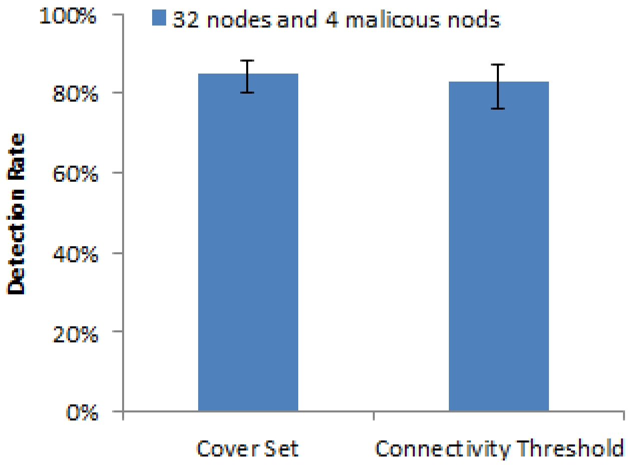


Figure 5.1: IDS detection rate of sparse network

Figure 5.1 shows that the average detection rate in sparse networks in which the subset of the node population functioning as IDS agents selected by our approach remains high, around 80%. Furthermore, the detection rate with random placement of distributed IDS agents based on the connectivity threshold remains as high as 85%. However, in very rare cases areas of the network remain unmonitored because of the random placement of IDS agents.

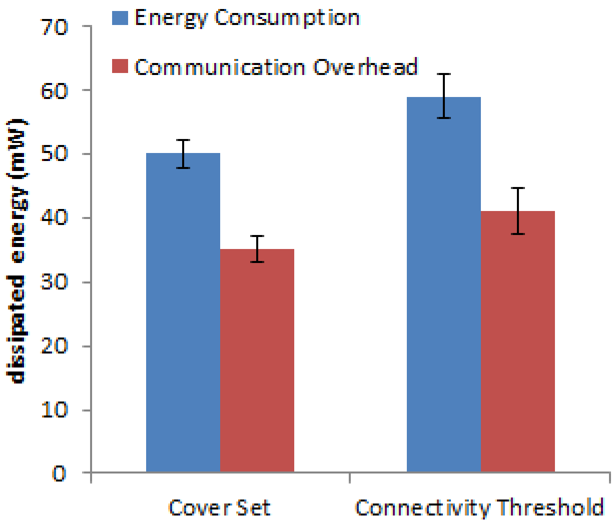


FIGURE 5.2: Energy consumption and communication overhead of sparse networks

Figures 5.2 and 5.3 show that the energy consumption and the communication overhead introduced to the network by the IDS on a sparse networks is proportional to the number of nodes operating as IDS agents. This shows that our approach achieved lower communication overheads and energy consumption rate when compared with random placement based on the connectivity threshold, where a constant number of nodes function as IDS agents. As shown in Figure 5.1 our approach achieved a high detection rate with lower energy consumption and communication overheads.

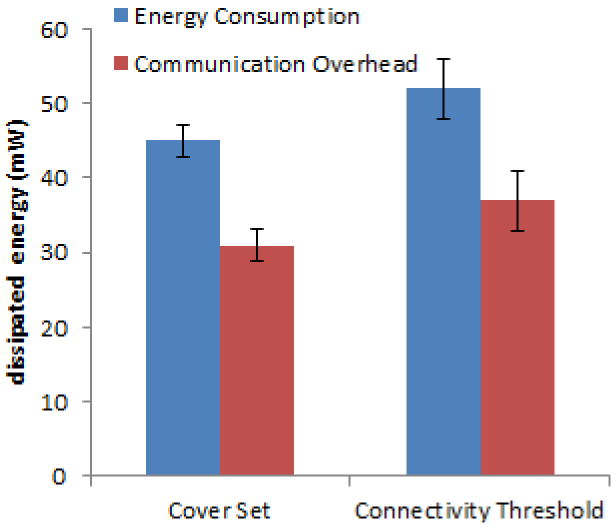


FIGURE 5.3: Energy consumption and communication overhead introduced by the IDS of sparse networks

5.5.3.2 Results of Moderate Dense Networks

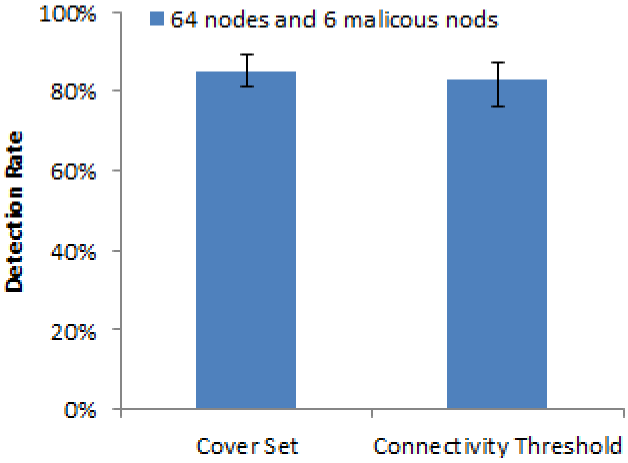


FIGURE 5.4: IDS detection rate of moderate dense network

The IDS placement method based on the cover-set algorithm also achieves a high detection rate in moderate dense networks, as shown in Figure 5.4, it remains high around 80%. Further, the detection rate with random placement of distributed IDS agents based on connectivity threshold is around 85%. In very rare cases, areas of the network remain unmonitored because of the random placement of IDS agents.

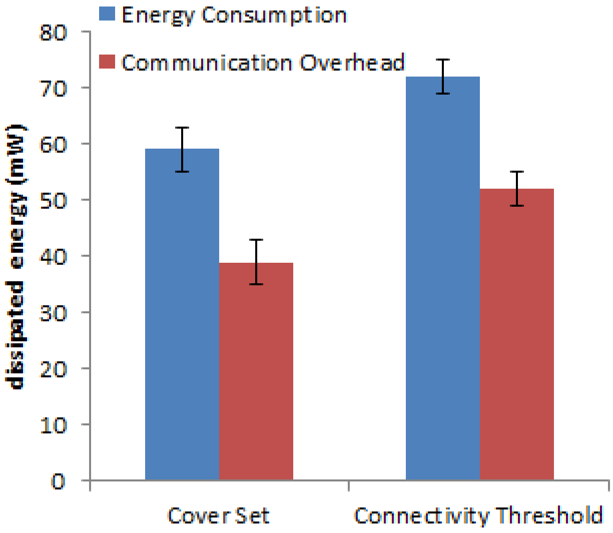


Figure 5.5: Energy consumption and communication overhead of moderate dense networks

Figures 5.5 and 5.6 show the energy consumption and the communication overheads introduced to the network by the IDS of moderate dense networks. The communication overheads and energy consumption in random IDS agent placement scenarios are higher than Cover Set algorithm placement due to random placement allows more nodes to function as IDS agents. This demonstrates that the placement of distributed IDS agents using cover Set algorithm allows fewer nodes with proper placement to function as IDS agents.

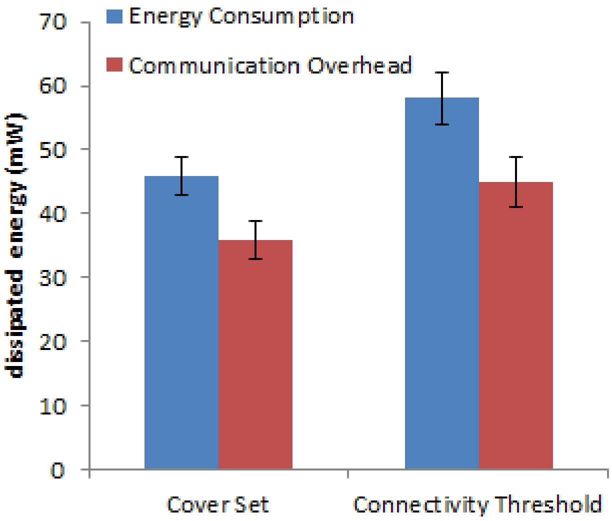


Figure 5.6: Energy consumption and communication overhead introduced by the IDS of moderate dense networks

5.5.3.3 Results of Dense Networks

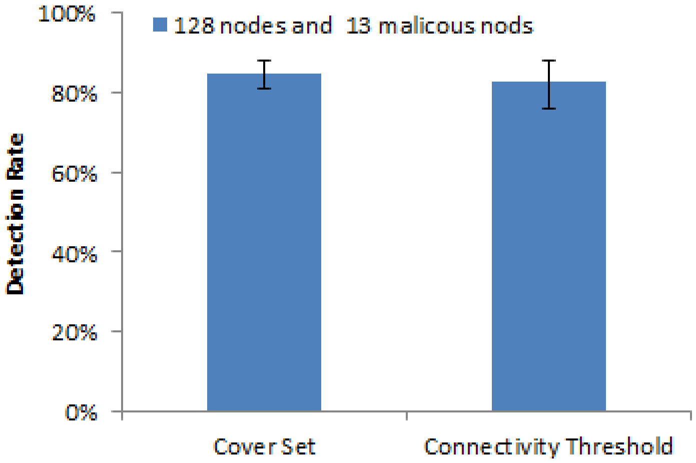


Figure 5.7: IDS detection rate of dense network

Figure 5.7 shows the average detection rate of dense networks. The detection rate in both placement strategies, random placement of distributed IDS agents based on connectivity threshold and the placement of IDS agents based on cover set algorithm, remains high at around 85%. Nevertheless, as with other network densities, there are cases in which areas of the network remain unmonitored because the random placement of IDS agents does not always result in complete or adequate coverage of the network area. This demonstrates that our approach provides better placement of the distributed IDS architecture since it allows a subset of nodes with proper location to perform as IDS agents.

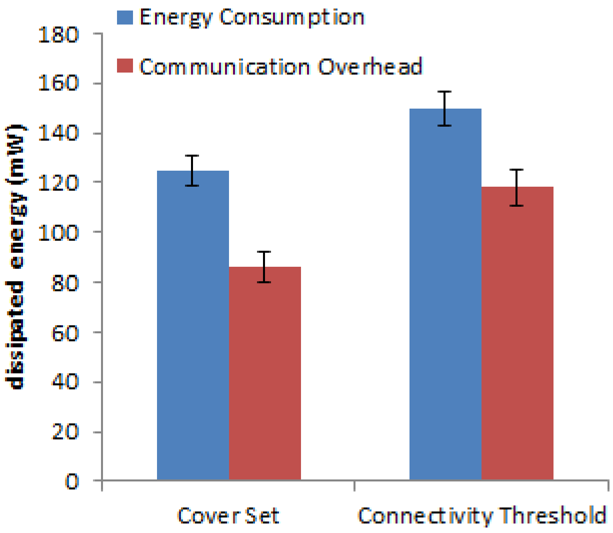


FIGURE 5.8: Energy consumption and communication overhead of dense networks

Figures 5.8 and 5.9 show the energy consumption and the communication overheads introduced to the network by the IDS of dense networks. The communication overhead and energy consumption in random IDS agent placement scenarios are also higher than with placement based on the cover set algorithm.

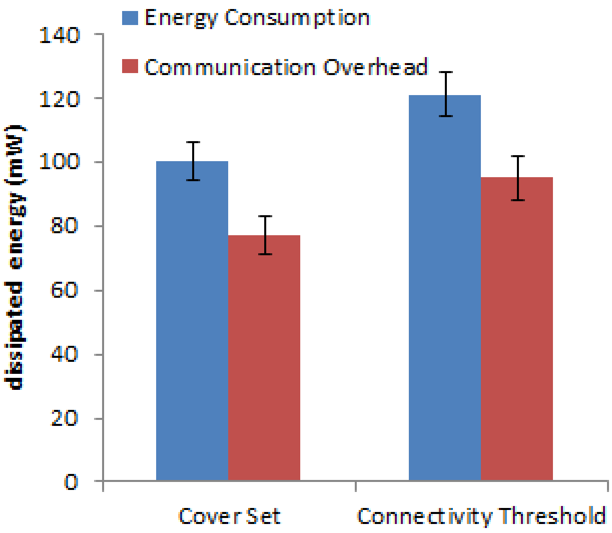


FIGURE 5.9: Energy consumption and communication overhead introduced by the IDS of dense networks

For all network densities, the energy consumption and the communication overhead introduced to the network by the IDS is proportional to the number of nodes operating as IDS agents. This shows that our approach achieved a lower communication overhead and energy consumption rate compared with random placement based on the connectivity threshold, where a constant number of nodes function as IDS agents. Moreover, our approach achieved a high detection rate with lower energy consumption, indicating that a minimal number of nodes with proper placement were allowed to function as IDS agents.

5.5.4 Simulation Findings of Resilience Study

The series of the following experiments study the resilience of the proposed approach that refers to the ability of the network to recover its IDS architecture and capabilities in order to avoid any loss of IDS agent. We gradually and arbitrarily dropped some IDS agent nodes during the simulation to study the resilience and effect on the efficiency of the IDS. For each network density, we adopted five different scenarios. In the first scenario, all IDS agents were selected by cover set algorithm, and then we dropped one additional IDS agent in each subsequent scenario as shown in Figures 5.10, 5.11 and 5.12.

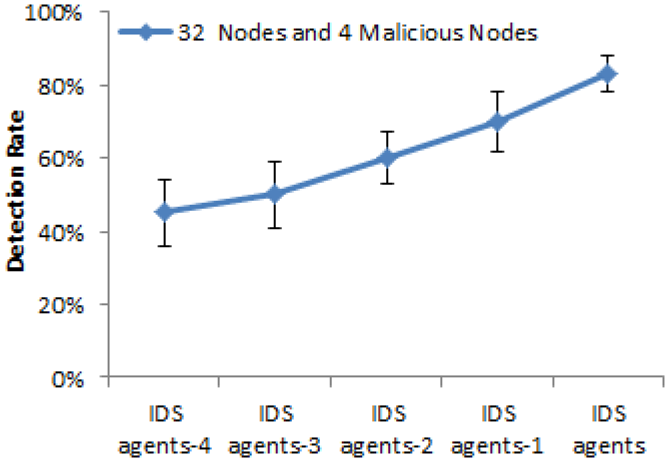


FIGURE 5.10: Detection rate over IDS agent node failure of sparse networks

Figure 5.10 shows the detection of sparse networks. The average detection rate in the scenario with all IDS agents is around 85%. However, the detection rate quickly drops in the other scenarios in which the IDS agents are gradually dropped.

The reason for the lower detection rate is that there were networks that were unmonitored when the IDS agents were dropped.

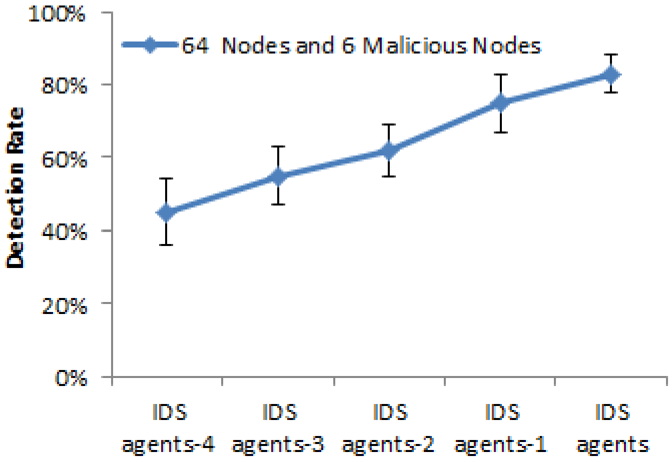


FIGURE 5.11: Detection rate over IDS agent node failure of moderate dense networks

In moderate dense networks, the average detection rate also quickly drops as the number of IDS agents is reduced as shown in Figure 5.11. The reason for the lower detection rate is that there also were networks that were unmonitored when the IDS agents were dropped.

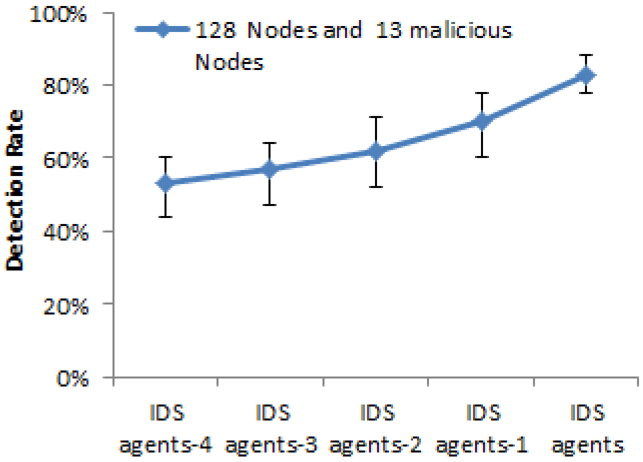


FIGURE 5.12: Detection rate over IDS agent node failure of dense networks

Figure 5.12 shows the detection of dense networks. The average detection rate with the scenario where with all IDS agents is around 85%. Further, the detection rate quickly drops as the number of IDS agents is reduced.

Then, we evaluated the proposed algorithm (IDS agent placement based on cover set algorithm). The algorithm underpins an invocation of a process that monitors the health of IDS agents, to detect when a node failure occurs. The centralised IDS reallocates a new subset (new cover set) of nodes to function as IDS agents in the event that an IDS agent node was dropped. In order to study the effectiveness of our Algorithm, we dropped off some IDS agent nodes gradually and arbitrarily during the simulation and analysed the detection rate after the simulation.

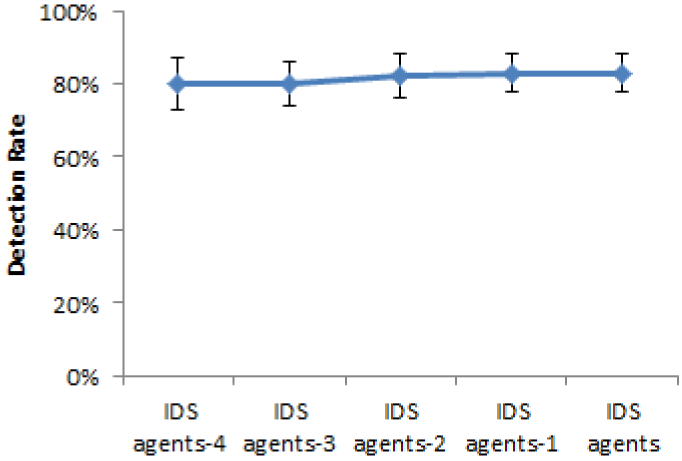


FIGURE 5.13: Detection rate after IDS agents failed recovery of sparse networks

Figure 5.13 shows the detection rate of sparse networks. The detection rate of scenarios in which some IDS agent nodes were dropped gradually remains as high as 85% compared to the scenario in which no IDS agent was dropped.

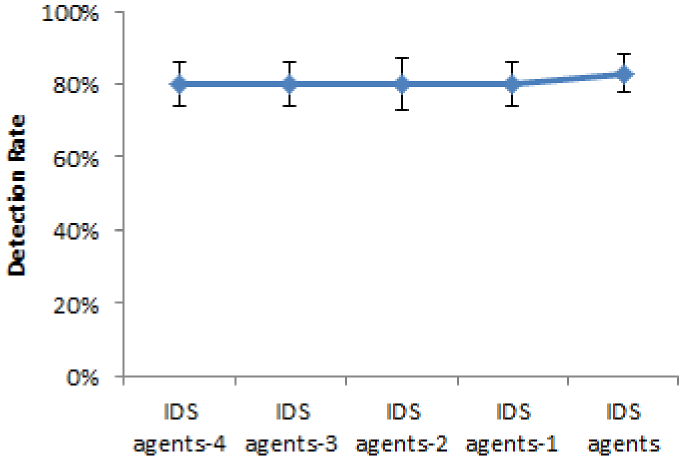


FIGURE 5.14: Detection rate after IDS agents failed recovery of moderate networks

In moderate dense networks, the detection rate in scenarios in which some IDS agent nodes were dropped gradually also remained high, as shown in Figure 5.14.

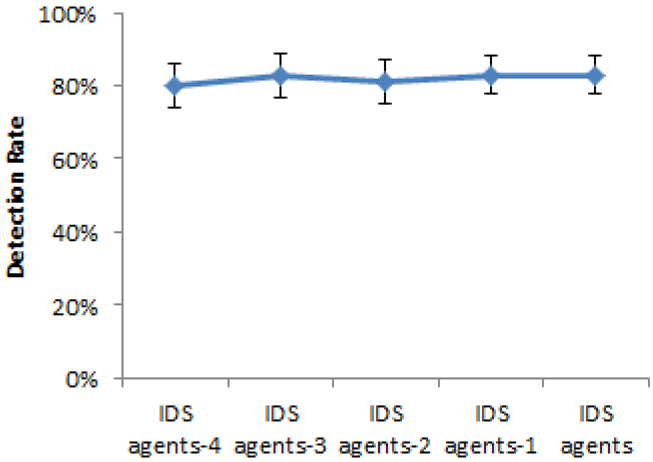


FIGURE 5.15: Detection rate after after failed IDS agent recovery of dense networks

Figure 5.15 shows that the detection rate of the dense network remained high even after the nodes were dropped. The detection rate in scenarios in which some IDS agent nodes were dropped gradually remained high (around 80%), compared to the scenario in which no IDS agent was dropped. This indicates that our approach provides a level of resilience against node failures.

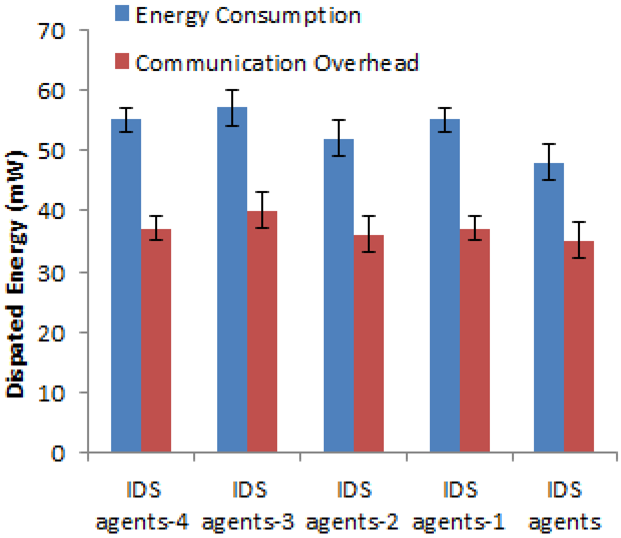


FIGURE 5.16: Energy consumption and communication overhead introduced by the IDS with the recovery mechanism of sparse networks

Figure 5.16 shows the energy consumption and the communication overheads in sparse networks. The energy and communication overheads are slightly increased in scenarios in which IDS agent nodes were dropped. This is because the centralised IDS reallocates the distributed IDS agents to recover any area left unmonitored.

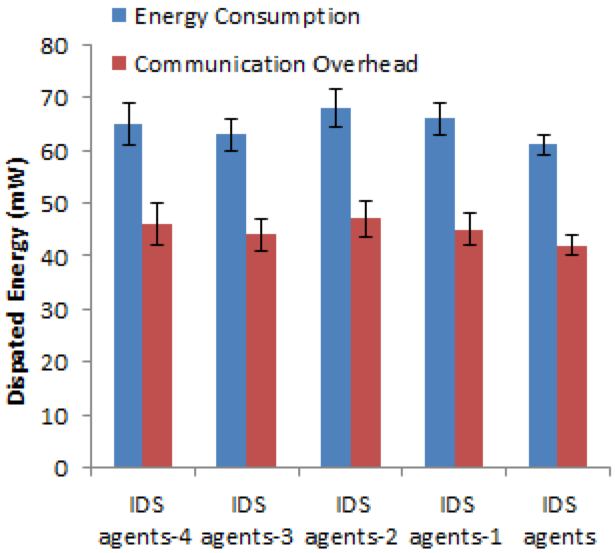


FIGURE 5.17: Energy consumption and communication overhead introduced by the IDS with the recovery mechanism of moderate dense networks

Figures 5.17 and 5.18 show the energy consumption and the communication overheads of moderately dense and dense networks. The energy consumption also increased slightly due to distributed IDS agent reallocation.

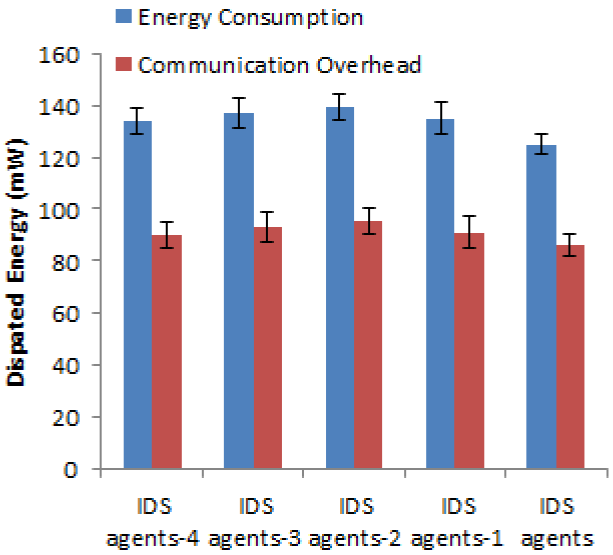


Figure 5.18: Energy consumption and communication overhead introduced by the IDS with recovery mechanism of dense networks

For all network densities, the energy consumption and the communication overheads were introduced to the network by the proposed algorithm, which monitors the distributed IDS agents against node failure. The energy consumption increased slightly after random IDS agent nodes were dropped because the centralised IDS reallocated the distributed IDS agents to recover the area that left unmonitored. This small increase is acceptable, and the effort results in increased resilience of the distributed IDS architecture against node failure.

This indicates that the energy efficiency and resilience of the hybrid IDS architecture for ad-hoc networks is independent of the number of nodes acting as IDS agents and their placement. It suffices that only one of the neighbouring nodes monitors and reports relevant information to the Sink. Thus, it reduces the number of nodes that must operate as IDS agents.

Approach	Overheads	Scalability	Resilience	Comments
SVELETE	high	small WSNs	-	high communication overhead
CT	depends on the density	larger networks	limited	achieved lower Energy Consumption and communication overhead
CS	depends on the density	larger networks	resilient	achieved lower communication overhead and resilient against node failures

Table 5.1: Comparison of proposed approaches with relevant approach

Table 5.1 provides a brief comparison of proposed approaches of this thesis with the relevant approach in the published literature and highlights the main contributions of this thesis. As shown in the table, proposed approaches allow to archived the aims of this thesis. The proposed approach that is described in Chapter 4 has achieved lower communication overhead and energy consumption since it allows a only a a random subset of the nodes based on connectivity threshold of RGG to act as IDS agents instead of all nodes as in SVELTE by (Raza et al. 2013). This greatly reduces the number of nodes that need to operate as IDS agents, thus mitigating any scalability and performance issues. And also, this is the reason why other works in the literature on IoT and WSAN IDS limit their simulation studies in networks with small populations. Moreover, the approach described in Chapter 5 achieved lower communication overhead and energy consumption because of use of cover-set algorithm that allows a subset on nodes to perform the IDS agent with minimum cardinality. Furthermore, this approach study the resilience that refers to the ability of the network to recover its IDS architecture and capabilities in order to avoid any loss of IDS agent that may affect the detection rate.

5.6 Conclusion

In this research, we used vertex-cover theory to minimise the number of IDS agents with respect to node placement. Thus, we can have a lower number of IDS agents which helps to improve the communication overheads and energy dissipation. Compared to the random placement of IDS agents introduced in Chapter 3 based on connectivity threshold in RGG, vertex-cover algorithm results in lower communication overheads and energy consumption Chein and Mugnier (2008).

This work studies an efficient and lightweight IDS for ad-hoc networks through the lens of IPv6-enabled WASNs. We first use RGG that which makes it possible to construct a formal model of WSNs. RGG capture the spatial characteristics of WSNs such as inter-dependencies on the existence of wireless links among neighbouring nodes. We focus on network attacks in IoT-specific networking protocols, such as sinkhole attacks in RPL. We identify the underlying causes of communication overheads and attempt to optimise the trade-off between the energy efficiency and detection rate of the IDS. We propose a novel IDS architecture that requires only a subset of the nodes with proper placement to efficiently operate as distributed IDS agents.

The experimental results show that our proposed approach achieved a high detection rate with a subset (cover set) of the nodes functioning as IDS agents. Therefore, the energy consumption and communication overhead introduced by the IDS is reduced since the energy consumption is proportional to the number of IDS agents. Furthermore, the results show that our proposed IDS architecture is resilient and robust against node failure. The centralised IDS monitors the distributed IDS agents and reallocates a new subset to function as IDS agents whenever node failure accrues.

Chapter 6

Conclusions

6.1 Introduction

This chapter summarises the results and the important findings of this thesis, and reviews this thesis with critical analyses to determine how this research address the aforementioned research question in Chapter 1, followed by recommendations and future directions for extending the research carried out in this thesis.

6.2 Achievement of Objectives

The objectives of this thesis were:

- **O1:** to review the current state-of-the-art-IDS with a particular emphasis on using IDS in WSN.
- **O2:** to define new evaluation metrics for WSN-IDS efficiency. In particular, define the metrics that capture the trade-off between energy consumption /communication overhead with detection rate of events.
- **O3:** to define a new WSN- IDS model/ architecture that properly addresses the distributed nature of WSN.
- **O4** to develop efficient protocols/methods/algorithms for WSN-IDS based on objectives 2, 3.
- **O5:** to develop a proof-of-concept WSN-IDS testbed for experimentally evaluating objective 4.

Overall the objectives were fully met. Following a thorough review of the current state of the art (**O1**), it was established that many IDS algorithms that are efficient in conventional networks, cannot be practically deployed in sensor networks as they may impede the actual operations and purpose of the network. Acknowledging that WSNs have a number of characteristics and constraints, we focus on establishing metrics to measure the efficiency of the IDS in such networks. These metrics were primarily linked to energy and computational efficiency, as well as the communication overheads (**O2**). There were cases in fact where a generous deployment of an IDS over a number of WSN agents (motes) resulted into the network being non operational. This lead to researching and proposing architectures that will implement a hybrid IDS model capable of balancing the IDS detection performance (or the IDS' indigenous upper bound of detection rate) and resulting energy and communication overheads (**O3**). The research resulted into two main architectures - one offering effective energy trade-offs and another one offering resilience - following a rigorous and formal analysis using RGGs (**O3 & O4**). Finally the above architectures were empirically evaluated through a series of simulations (**O5**).

6.3 Evaluation

6.3.1 Literature

The systematic literature review involved the investigation of scientific published work in the area of IDS, IoT networks and WSNs in particular, found in online digital repositories. The online resources provided an extensive source of relevant information on the current state of research and body of knowledge. The literature study was instrumental for developing the specific conceptual and theoretical framework. The current state-of-the-art Intrusion Detection System in Wireless Sensor Networks was presented. Furthermore, the security aspects and challenges of wireless sensor networks, networking standard protocols and shortcoming/limitations of the related work in the fields of Intrusion Detection in WSNs were explored and investigated.

More specifically, the literature review process commenced with the research in the domain of Wireless Sensor Networks and specifically to understand the characteristics of tiny devices and their security challenges. Then, we reviewed standardised wireless network protocols, their security weakness and how these can potentially, or have been exploited by malicious activities. Lastly, we reviewed the current state-of-the-art IDSs in WSN and IoT to identify and map the shortcomings into research questions that guided this research.

6.3.2 Analysis

In the domain of Intrusion Detection Systems (IDS) in IoT, a considerable amount of research has been carried out on deployment architectures, detection strategies and algorithms. However, available IDS in IoT are designed based on assumptions holding from "conventional" computer networks, e.g. that each node of the network is assumed to be powerful in terms of resources (available energy, memory, CPU, etc.). They are also assumed to be always available and the nodes to communicate over a reliable and high-capacity network, which is not the case for WSN environments. As such, an IDS that efficiently addresses the IoT paradigm is needed.

The current state-of-the-art on IDSs for WSN and IoT networks are persistent in proposing resource-intensive solutions, as they have been inheriting solutions from conventional computer networks. Centralised IDS architectures introduce significant communication overhead to the network as the base station (or Sink) due to large numbers of requests to and from the nodes related to IDS data collection. Distributed IDS architectures rely on the cooperation between the sensor nodes, thus increasing the communication load as well as energy dissipation. Lastly, hybrid IDS architectures achieve a better control and global overview of the network, but currently available solutions also introduce a significant communication overhead that increases proportionally to the number of network nodes. Furthermore, resilience and robustness in such environments have not been extensively studied.

The shortcomings and open issues of the current state-of-the-art in related work contributing to designing proposed IDS for WSN can be summarised as follows:

1. Existing IDS in WSNs and IoT remain relatively resource-intensive and the constrained nature of WSNs and IoT have not been fully explored.

2. The IDS placement strategy in the current state of the art, is based on assumptions inherited from conventional computer networks; as such a more suitable IDS placement strategy that addresses the nature of WSNs and IoT is needed.
3. Robustness of the current state-of-the-art IDS in WSNs and IoT has not yet been fully investigated.

In this work we focus on hybrid IDS architectures but we show that by taking the specifics of IoT protocols into account, such as the ranking mechanism of RPL, as well as the spatial characteristics of such networks, the number of required IDS agents in the network (and therefore the corresponding overhead) can be greatly reduced while maintaining sufficiently high detection rates.

6.3.3 Design

The identified shortcomings and open issues in current state of research were translated into research questions that in turn specified the aim and objectives of this research. The aims of this thesis were derived from a conducted literature study to fulfill the shortcomings of the current state-of-the-art IDS in WSNs and to address the constraint and distributed nature of WSN.

This thesis introduces an IDS architecture that efficiently addresses the particular characteristics of the IoT and Wireless Sensor Network paradigms. In particular, the proposed system addresses the following characteristics of IoT and WSN: their highly distributed nature; their ad-hoc network structure; the peer-to-peer communication scheme among the devices; the highly constrained nature of the devices per se in terms of resources (computational power, available energy, limited memory, etc).

First, we determine the impact of IDS mechanisms towards the communication and energy overheads and attempt to model the trade-off between energy efficiency of IDS and detection rate. Then, we propose a novel IDS architecture that requires only a subset of the nodes with proper placement to efficiently operate distributed IDS agents. We model a WSN with the use of Random Geometric Graphs (RGG). Random Graphs have been used as a well-studied model and a paradigm for wireless networks, such as sensor networks. Motes are represented a vertices in RGG, and the communication between these motes is represented by the edges. The RGG model efficiently captures spatial characteristics of the network that are closely related to network connectivity; e.g. inter-dependencies on the existence of wireless links among neighbouring nodes. Random Geometric Graphs can represent the actual placing of the set of n vertices uniformly and randomly at a given area of interest. Then, motivated by how IoT networking protocols, such as RPL, manage and operate the network, we identify the trade-offs between the communication overheads introduced by an IDS and its detection rate of attacks such as the sinkhole attack. The central IDS controls the entire IDS architecture and relevant data from the distributed agents. Each network node that runs an instance of the distributed agent, monitors and collects data on local network activity from its 1-hop neighbouring nodes. This implies that not all nodes need to run the IDS agent, but only a subset of them such that every node in the network has at least one 1-hop neighbour operating as an IDS agent. In graph theory, such a subset is defined as a vertex cover of the corresponding RGG graph that captures the structure of the network. This subset of the nodes that act as IDS agents is selected based on the Vertex-cover algorithm (greedy algorithm) to find a subset of minimum cardinality with proper placement. Moreover, we propose a method to maintain and monitor the distributed IDS agents against node failures. Our designed system successfully addresses the following characteristics of IoT and WSN: their highly distributed nature; their ad-hoc network structure; the peer-to-peer communication scheme among the devices; the highly constrained nature of the devices per se in terms of resources(computational power, available energy, limited memory, etc).

6.3.4 Prototype

A prototype was implemented by extending the state-of-the-art IDS for WSN by Raza et al. called SVELTE Raza et al. (2013). In SVELTE, authors consider multi-hop peer-to-peer IPv6-enabled WSNs running the 6LoWPAN stack Shelby and Bormann (2011) on ContikiOS Dunkels et al. (2004). They develop a hybrid IDS architecture that consists of a centralised module running on the Sink and a distributed agent running on each individual sensor node. We applied our methods on the state of the art on IDS for WSNs and conduct our performance evaluation via extensive emulations. We focus on experimentally evaluating our approach on SVELTE as a representative example of hybrid architecture IDS for ad-hoc networks. Particularly, we evaluate the trade-off between the potentially reduced overhead of the IDS in successfully detection rate (due to lower number of active IDS agents in the network) versus the reduced communication overhead and increased energy efficiency of the network. We also evaluate the reliance and robustness of our proposed method against random node failures.

6.4 Experimental Results & Findings

We consider various network densities as they are formally defined by RGG model. Experimental results show that our proposed approach achieved high detection rates with a subset of the nodes running as IDS agents. The energy consumption and communication overhead introduced by the IDS reduced since the energy consumption is proportional to the number of IDS agents.

We show that 1) indeed the IDS detection rates remain at very high levels (around 85%) even with a subset of the nodes as IDS agents; 2) that the required number of IDS agents in the network in order to achieve these levels is independent from the network population and in fact *constant*; 3) that the energy consumption and communication overhead introduced by the IDS is proportional to the number of IDS agents, therefore our method allows for massive energy gains while not affecting the detection rate of the IDS.

Conducted experiments show that in all network densities the average detection rate where the subset of the node population operates as an IDS agent selected by our approach that uses cover set (greedy algorithm) remains high (around 80%). Also, the detection rate in random placement of distributed IDS agents based on connectivity threshold remains at high levels. However in rare cases, areas of the network remain unmonitored due to the random placement of IDS agents. This demonstrates that our approach provides better placement of the distributed IDS architecture since it allows the proper nodes to perform the IDS agent.

Moreover we show that the energy consumption and the communication overhead introduced to the network by the IDS is proportional to the number of nodes operating as IDS agents. This shows that our approach achieved lower communication overhead and energy consumption rate compared to random placement based on the connectivity threshold where a constant number of nodes performing the IDS agent. Our approach achieved high detection rate with lower energy consumption which reveals that minimal number of nodes were allowed to run the IDS agent.

At the second part of our experiment we first dropped off gradually and arbitrarily some IDS agent nodes during the simulation to study the resilience and efficiency of the proposed approach. The rsults show that the detection rate quickly drops due to that areas of the networks when nodes were drooped off left un-monitored. Then, we evaluated our algorithm that integrated with our approach that monitors the health of IDS agents. We also dropped off gradually and arbitrary some IDS agent nodes during the simulation. The detection rate remains high even after the nodes dropping which indicates that our approach provides a level of resilience against nodes failure.

Furthermore, results show that our proposed IDS architecture is resilient and robust against node failures. Centralised IDS monitors the distributed IDS agents and reallocates a new subset to run as IDS agents whenever node failure accrue.

For all network densities the energy consumption and the communication overhead introduced to the network by our proposed algorithm that monitors the distributed

IDS agents against nodes failure. The energy consumption slightly increased after random IDS agents nodes dropped due to the fact that the centralised IDS reallocated the distributed IDS agents in order to recover the area that was left un-monitored. We argue that the effort that provides the resilience of distributed IDS architecture against node failures is worthwhile.

Experimental results show that the proposed approach achieved high detection rates with a subset of the nodes running as IDS agents. The energy consumption and communication overheads introduced by the IDS were reduced since the energy consumption is proportional to the number of IDS agents. Furthermore, results show that the proposed IDS architecture is resilient and robust against node failures. Centralised IDS monitors the distributed IDS agents and reallocates a new subset to run as IDS agents whenever node failure accrue.

This indicates that the energy efficiency and resilience of hybrid IDS architecture for ad-hoc networks is independent to the number of nodes acting as IDS agents and their placement. It is sufficient to set only one of neighbouring nodes to monitor and report relevant information to the Sink. Thus, the number of nodes that are needed to operate as IDS agents is effectively reduced.

6.5 Future Work and Recommendations

In this section, we propose three possible directions and extensions for future research. These recommendations are set out in the order of importance with the most important being presented first.

In this work detection accuracy was outside the scope of the research. We investigate the underline cause of the communication and energy consumption overheads introduced to constrained networks by Intrusion Detection System. In particular, we consider the underline cause of communication overhead in state-of-the-art and optimise the trade-off between energy efficiency of IDS and detection rate. We consider the IDS architecture and the IDS agents' placement strategies regardless of the detection accuracy. we propose a novel IDS architecture that requires only a subset of the nodes with proper placement to efficiently operate distributed IDS agents. Further studies should be conducted to enhance the accuracy of detection algorithms that we used with proposed IDS architecture. For instance, we consider in conducted experiments for this research sinkhole attack on RPL routing protocol for constrained networks as an attacker model. Sinkhole attack is an insider attack were an attacker compromises a node inside the network and launches an attack. The malicious node in this attack advertises a beneficial path in order to attract its neighbour nodes to route the traffic through it based on the routing metric that used in routing protocol. The existing detection algorithm we used on proposed IDS architecture achieved high detection rates. However, further work is certainly required to improve the detection accuracy of this detection algorithm and also an extension is required to detect other attacks on other routing protocol.

This thesis considers ad-hoc networks excluding the mobility of wireless networks. In the proposed approach, we consider the static positioning of ad-hoc Wireless Sensor Networks. We provide a formal model for WSNs with the use of Random Geometric Graphs; a graph-theoretical model that properly captures the spatial characteristics of WSNs such as inter-dependencies on the existence of wireless links among neighbouring nodes. Future studies should carried out to investigate the novel findings of this work on Mobile Wireless Sensor Networks (MWSNs) that are been used recently in many critical setting and real-world applications. MWSNs share the following characteristics of IoT and WSN: their highly distributed nature; their ad-hoc network structure; the peer-to-peer communication scheme among the devices; the highly constrained nature of the devices per se in terms of resources (computational power, available energy, limited memory, etc). Moreover, MWSNs are characterised by their mobility feature that adds more challenging security solutions for sensor networks. Therefore, future research should be considering a study of standardised routing protocols of mobile Wireless Networks and investigate the aforementioned challenges. The future research should be carried out to extend the proposed IDS WSNs to include the mobility of WSNs. The

IDS architecture for WSNs proposed on this thesis can be extended to consider mobile Wireless Networks.

Furthermore, future reserach activities can focus on employing rigorous graph-theoretical methods and tools to formally prove the proposed approach on real-world networks. Moreover, more efficient and specific algorithms for choosing in a distributed way which nodes should operate as IDS agents as well as balancing this role among all the nodes can be researched and developed.

6.6 Conclusion

This research overall has successfully achieved its aims and objectives, providing a novel Intrusion Detection System architecture for ad-hoc networks that properly addresses the distributed nature of constrained wireless networks.

In this thesis, we studies efficient and lightweight Intrusion Detection Systems for ad-hoc networks via the prism of IPv6-enabled Wireless Actuator Sensor Networks. We first used Random Geometric Graphs (RGG) that make it possible to provide a formal model of WSNs. RGGs capture the spatial characteristics of WSNs as such inter-dependencies on the existence of wireless links among neighbouring nodes. We focus on network attacks in IoT-specific networking protocols such as sinkhole attacks in RPL. We identify the underline cause of communication overhead in state-of-the-art and try to optimise the trade-off between the energy efficiency of IDS and detection rate. We propose a novel IDS architecture that requires only a subset of the nodes with proper placement to efficiently operate distributed IDS agents.

The proposed methods were integrated on the state-of-the-art IDS for WSNs and we conducted our performance evaluation through extensive emulations. Experiment results show that the proposed approach achieved high detection rates with a subset of the nodes running as IDS agents. The energy consumption and communication overhead introduced by the IDS reduced since the energy consumption is proportional to the number of IDS agents. Various network densities we considered (as these are formally defined via the RGG model) and it was shown that indeed the IDS detection rates remain at very high levels even with a subset of the nodes as IDS agents.

The experimental results show the energy consumption and communication overheads introduced by the IDS is proportional to the number of IDS agents. Therefore, the proposed method allows for massive energy gains while not affecting the detection rate of the IDS. Furthermore, results show that our proposed IDS architecture is resilient and robust against node failures. Because the centralised IDS monitors the distributed IDS agents and reallocates a new subset to run as IDS agents whenever node failure accrues.

Appendix A

A selection of indicative network topology alternatives

The following figures depict the configurations used with varying density of nodes and IDS agents that were used for the empirical evaluation of the proposed methods.

A.1 Sparse Network

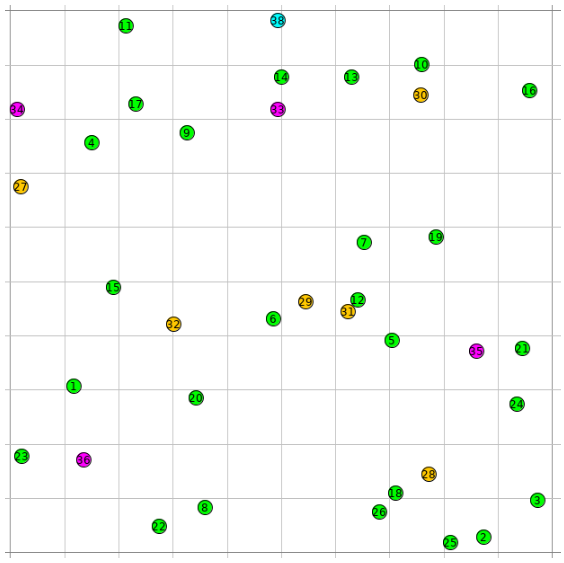


Figure A.1: 20% IDS agents

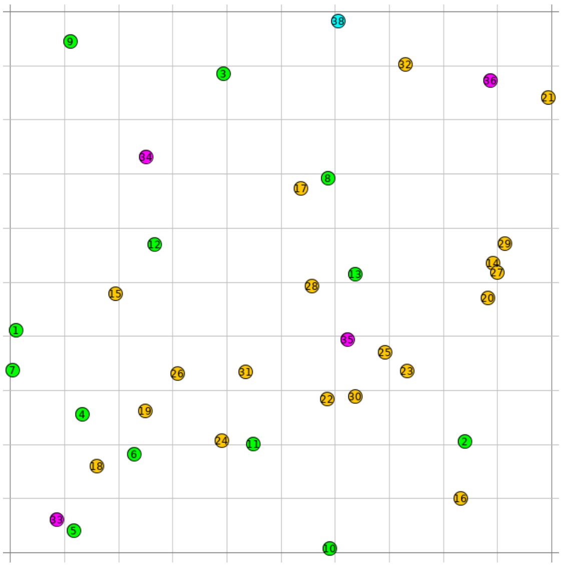


Figure A.2: 60% IDS agents

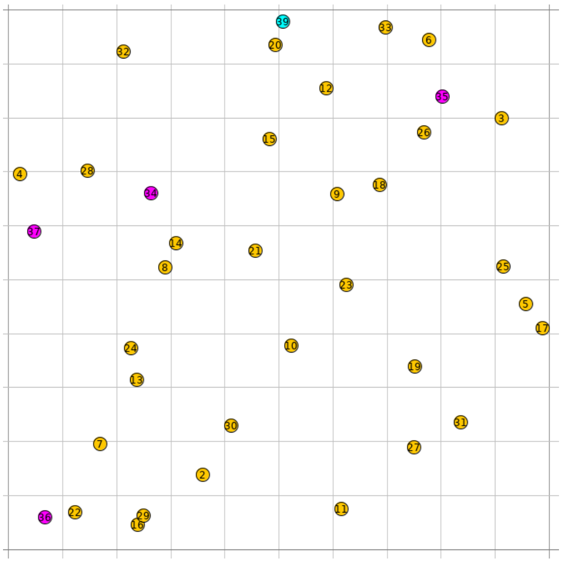


Figure A.3: 100% IDS agents

A.2 Moderate Dense Networks

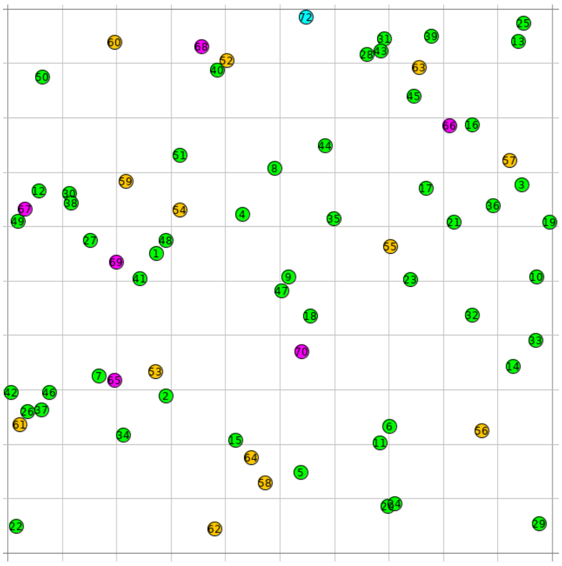


Figure A.4: 20% IDS agents

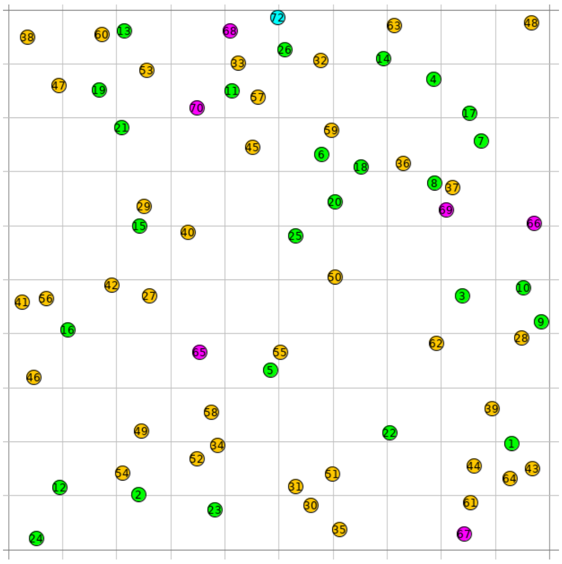


Figure A.5: 60% IDS agents

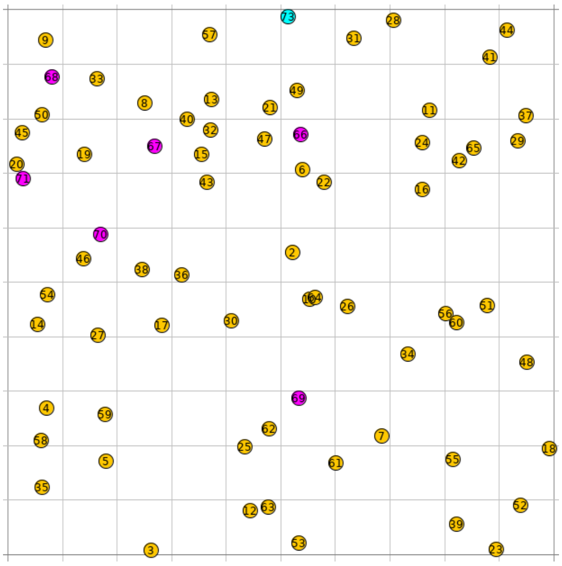


FIGURE A.6: 100% IDS agents

A.3 Dense Networks

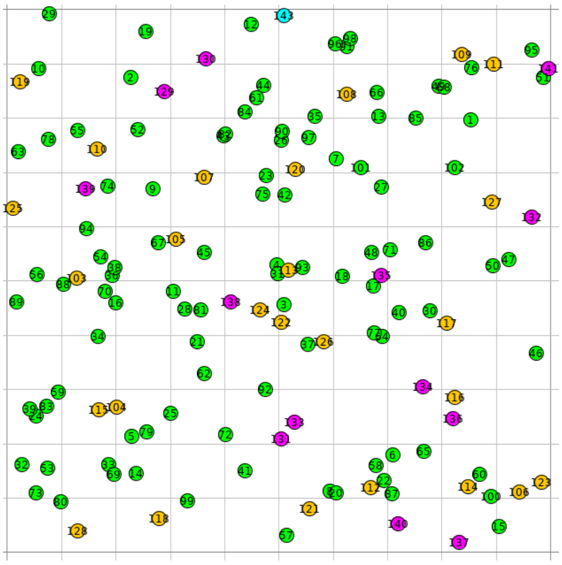


FIGURE A.7: 20% IDS agents

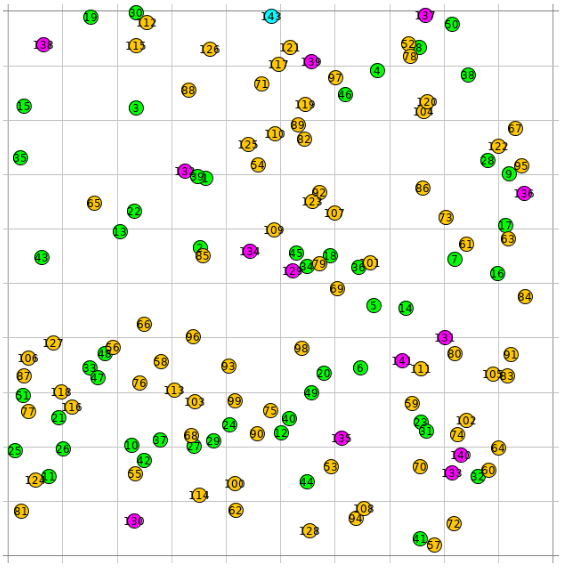


Figure A.8: 60% IDS agents

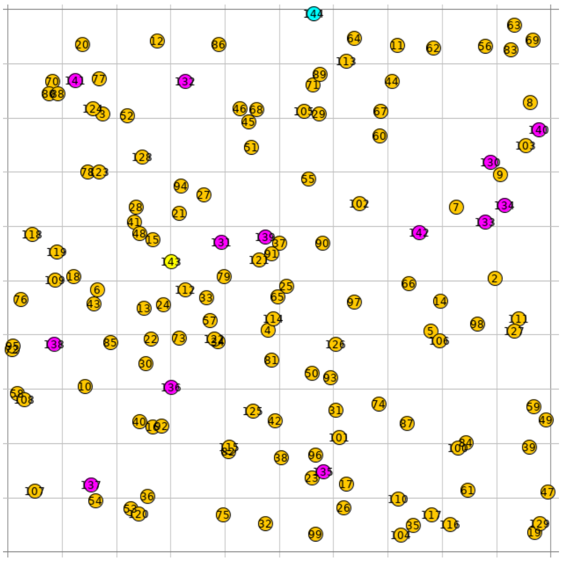


Figure A.9: 100% IDS agents

Appendix B

Energy Consumption Calculations

B.1 Sensor Node Energy Consumption

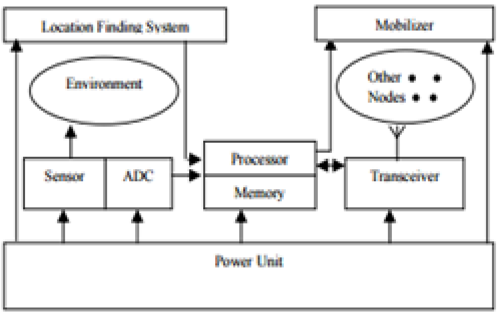


FIGURE B.1: The basic components of motes

We calculated the energy consumption of WSN nodes as follows:

- **Sensing Unit.** E_{on-off} is the energy consumption of the closing sensor operation, E_{off-on} is the energy consumption of the opening sensor operation and $E_{sensor-run}$ is the energy consumption of the sensing operation.

$$E_{sen} = E_{on-off} + E_{off-on} + E_{sensor-run} \tag{B.1}$$

- **Processing Unit.** In this unit, there are three operation states (sleep, idle, run). E_{cpu} is the sum of the state energy consumption $E_{cpu-state}$ and the state-transition energy consumption, $E_{cpu-change}$ where $i = 1, 2, ..m$ is the processor operation state and m is the number of the processor state, $j = 1, 2, ..n$, is the is the type of state transition and n is the number of the state-transition.

$$\begin{aligned} E_{cpu} &= E_{cpu-state} + E_{cpu-change} \\ &= \sum_{i=1}^{m} P_{cpu-stat}(i) T_{cpu-state}(i) + \sum_{i=1}^{n} P_{cpu-stat}(j) T_{cpu-state}(j) \end{aligned} \tag{B.2}$$

$P_{cpu-state}(i)$ is the power of state i that can be found from the reference manual and $T_{cpu-state}(i)$ is the time interval in state i. $P_{cpu-change}(j)$ is the frequency of state transition j and $T_{cpu-change}(j)$ is the energy consumption of one-time state transition j (Tomić and McCann 2017).

- **Communication Unit.** The transceiver operation states: $Tx,\ Rx, Off,\ Idle,\ Sleep\ and\ Clear\ Chanel\ Assessment$ (CCA/ED) (Zhou et al. 2011). We calculate the total energy consumption of the communication unit by summing the state energy consumption and state-transition.

$$E_{com} = E_{TX} + E_{RX} + E_{Idle} + E_{sleep} + E_{CCA} \tag{B.3}$$

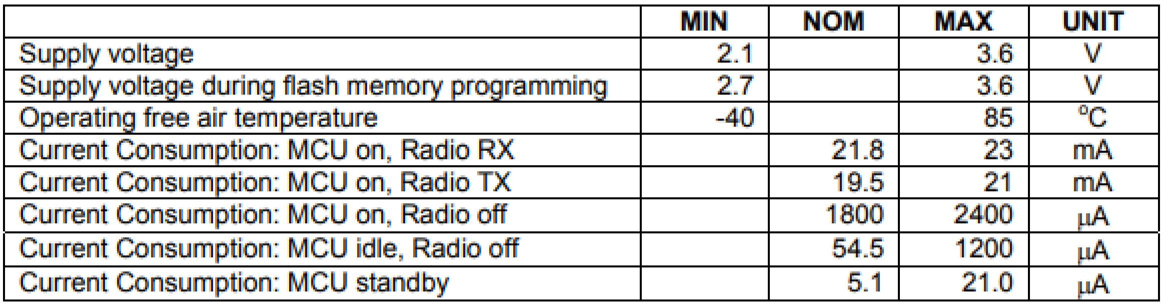
Then, The total energy consumption all over the network =

$$E_{total} = E_{sen} + E_{cpu} + E_{com} \tag{B.4}$$

B.2 Energy Consumption Calculations in Contiki OS

B.2.1 Tmote Sky energy consumption calculation

For all conducted experiments, we use Tmote Sky as it has been used widely in real-wold systems. Thus, we followed the operation conditions of Tmote Sky provided by the manufacturer to trace and calculate the energy consumption during the simulations.



	MIN	NOM	MAX	UNIT
Supply voltage	2.1		3.6	V
Supply voltage during flash memory programming	2.7		3.6	V
Operating free air temperature	-40		85	°C
Current Consumption: MCU on, Radio RX		21.8	23	mA
Current Consumption: MCU on, Radio TX		19.5	21	mA
Current Consumption: MCU on, Radio off		1800	2400	µA
Current Consumption: MCU idle, Radio off		54.5	1200	µA
Current Consumption: MCU standby		5.1	21.0	µA

Figure B.2: Tmote Sky operation conditions (TmoteSky data sheet)

$$E_{total} = E_{sen \times 0.0545mA} + E_{cpu \times 1.8mA} + E_{transmit \times 19.5mA + listen \times 21.8mA} \tag{B.5}$$

$$E(mJ) = \frac{E_{total} \times 3V}{4396 \times 8} \tag{B.6}$$

$$P(mW) = \frac{Emj}{Time(s)} \tag{B.7}$$

B.2.2 Power-trace of Cooja simulation

```
import csv
import pandas as pd
import numpy as np

filename=’motes output’

data=np.array([’s’,’clock_time’,’P’,’rimeaddr_node_addr’,’seqno’,’all_cpu’,’all_lpm’,

with open(filename,’rb’) as f:
    reader = csv.reader(f,delimiter=’ ’)
    for row in reader:
        if len(row)==17:
            data=np.vstack([data,row])

d=pd.DataFrame(data=data[1:,1:],index=data[1:,0],columns=data[0,1:])
for c in d.columns:
     if c!=’P’:
            d[c]=d[c].map(float)

#extract ID and store in new colunn
def get_id():
       return [int(s.split("ID:")[1].strip())  for s in d.index]

d[’ID’]=get_id()

# start producing output...
print "primary data extracted (interval read):"
for i in d.ID.unique():
      if len(d[d.ID==i])>1:
        (cpu,lpm,tx,tr)=d[d.ID==i].iloc[0][’all_cpu’:’all_listen’]
```

```
             print "node",i,cpu,lpm,tx,tr

print "\n\nCommunication overhead (0 to 30 mins):"
for i in d.ID.unique():
     if len(d[d.ID==i])>2:
           (cpu,lpm,tx,tr)=d[d.ID==i].iloc[1]['all_cpu':'all_listen']
           (cpu1,lpm1,tx1,tr1)=d[d.ID==i].iloc[2]['all_cpu':'all_listen']
             comm_over=(((tx1-tx)*19.5+(tr1-tr)*21.8)*3/(4096*8))/1800
             communication_overhead30[i]=comm_over
         print "node",i,comm_over
print "Total communication overhead for all nodes:",
np.sum(communication_overhead30.values())

print "\nTotal commnunication overhead for 30 mins
(using the  calculation):", np.sum(communication_overhead.values())
#d.all_energy.plot()
#d.all_energy.hist
```

Bibliography

Abdelshafy, M. A. and King, P. J. (2016), Resisting blackhole attacks on manets, *in* '2016 13th IEEE Annual Consumer Communications & Networking Conference (CCNC)', IEEE, pp. 1048–1053.

Abduvaliyev, A., Pathan, A.-S. K., Zhou, J., Roman, R. and Wong, W.-C. (2013), 'On the vital areas of intrusion detection systems in wireless sensor networks', *IEEE Communications Surveys & Tutorials* **15**(3), 1223–1237.

Abo-Zahhad, M., Farrag, M., Ali, A. and Amin, O. (2015), An energy consumption model for wireless sensor networks, *in* '5th International Conference on Energy Aware Computing Systems & Applications', IEEE, pp. 1–4.

Accettura, N., Grieco, L. A., Boggia, G. and Camarda, P. (2011), Performance analysis of the rpl routing protocol, *in* '2011 IEEE International Conference on Mechatronics', IEEE, pp. 767–772.

Adu-Manu, K. S., Adam, N., Tapparello, C., Ayatollahi, H. and Heinzelman, W. (2018), 'Energy-harvesting wireless sensor networks (eh-wsns): A review', *ACM Transactions on Sensor Networks (TOSN)* **14**(2), 10.

Ahmed, A., Bakar, K. A., Channa, M. I., Haseeb, K. and Khan, A. W. (2015), 'Terp: A trust and energy aware routing protocol for wireless sensor network', *IEEE Sensors Journal* **15**(12), 6962–6972.

Airehrour, D., Gutierrez, J. and Ray, S. K. (2016), Securing rpl routing protocol from blackhole attacks using a trust-based mechanism, *in* '2016 26th International Telecommunication Networks and Applications Conference (ITNAC)', IEEE, pp. 115–120.

Akyildiz, I. F., Vuran, M. C. and Akan, O. B. (2006), A cross-layer protocol for wireless sensor networks, *in* '2006 40th Annual Conference on Information Sciences and Systems', Ieee, pp. 1102–1107.

Al-Fuqaha, A., Guizani, M., Mohammadi, M., Aledhari, M. and Ayyash, M. (2015), 'Internet of things: A survey on enabling technologies, protocols, and applications', *IEEE Communications Surveys & Tutorials* **17**(4), 2347–2376.

Al-Garadi, M. A., Mohamed, A., Al-Ali, A., Du, X., Ali, I. and Guizani, M. (2020), 'A survey of machine and deep learning methods for internet of things (iot) security', *IEEE Communications Surveys & Tutorials* .

Alanazi, S., Al-Muhtadi, J., Derhab, A., Saleem, K., AlRomi, A. N., Alholaibah, H. S. and Rodrigues, J. J. (2015), On resilience of wireless mesh routing protocol against dos attacks in iot-based ambient assisted living applications, *in* '2015 17th International Conference on E-health Networking, Application & Services (HealthCom)', IEEE, pp. 205–210.

Alharbi, S., Rodriguez, P., Maharaja, R., Iyer, P., Subaschandrabose, N. and Ye, Z. (2017), Secure the internet of things with challenge response authentication in fog computing, *in* '2017 IEEE 36th International Performance Computing and Communications Conference (IPCCC)', IEEE, pp. 1–2.

Alkalbani, A., Tap, A. M. and Mantoro, T. (2013), Energy consumption evaluation in trust and reputation models for wireless sensor networks, *in* '2013 5th International Conference on Information and Communication Technology for the Muslim World (ICT4M)', IEEE, pp. 1–6.

Amish, P. and Vaghela, V. (2016), 'Detection and prevention of wormhole attack in wireless sensor network using aomdv protocol', *Procedia computer science* **79**, 700–707.

Ancillotti, E., Bruno, R. and Conti, M. (2013), 'The role of the rpl routing protocol for smart grid communications', *IEEE Communications Magazine* **51**(1), 75–83.

Andrea, I., Chrysostomou, C. and Hadjichristofi, G. (2015), Internet of things: Security vulnerabilities and challenges, *in* '2015 IEEE Symposium on Computers and Communication (ISCC)', IEEE, pp. 180–187.

Arias, O., Wurm, J., Hoang, K. and Jin, Y. (2015), 'Privacy and security in internet of things and wearable devices', *IEEE Transactions on Multi-Scale Computing Systems* **1**(2), 99–109.

Arrington, B., Barnett, L., Rufus, R. and Esterline, A. (2016), Behavioral modeling intrusion detection system (bmids) using internet of things (iot) behavior-based anomaly detection via immunity-inspired algorithms, *in* '2016 25th International Conference on Computer Communication and Networks (ICCCN)', IEEE, pp. 1–6.

Asgeirsson, E. and Stein, C. (2007), Vertex cover approximations on random graphs, *in* 'International Workshop on Experimental and Efficient Algorithms', Springer, pp. 285–296.

Benzarti, S., Triki, B. and Korbaa, O. (2017), A survey on attacks in internet of things based networks, *in* 'Engineering & MIS (ICEMIS), 2017 International Conference on', IEEE, pp. 1–7.

Biswas, S., Das, R. and Chatterjee, P. (2018), Energy-efficient connected target coverage in multi-hop wireless sensor networks, *in* 'Industry interactive innovations in science, engineering and technology', Springer, pp. 411–421.

Bollobás, B. (2004), *Extremal graph theory*, Courier Corporation.

Bollobás, B. (2013), *Modern graph theory*, Vol. 184, Springer Science & Business Media.

Bouabdallah, F., Bouabdallah, N. and Boutaba, R. (2008), 'On balancing energy consumption in wireless sensor networks', *IEEE Transactions on Vehicular Technology* **58**(6), 2909–2924.

Bouabdellah, M., Kaabouch, N., El Bouanani, F. and Ben-Azza, H. (2018), 'Network layer attacks and countermeasures in cognitive radio networks: A survey', *Journal of information security and applications* **38**, 40–49.

Burhanuddin, M., Mohammed, A. A.-J., Ismail, R., Hameed, M. E., Kareem, A. N. and Basiron, H. (2018), 'A review on security challenges and features in wireless sensor networks: Iot perspective', *Journal of Telecommunication, Electronic and Computer Engineering (JTEC)* **10**(1-7), 17–21.

Butun, I., Morgera, S. D. and Sankar, R. (2014), 'A survey of intrusion detection systems in wireless sensor networks', *IEEE communications surveys & tutorials* **16**(1), 266–282.

Camp, T., Boleng, J., Williams, B., Wilcox, L. and Navidi, W. (2002), Performance comparison of two location based routing protocols for ad hoc networks, *in* 'Proceedings. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies', Vol. 3, IEEE, pp. 1678–1687.

Can, O. and Sahingoz, O. K. (2015), A survey of intrusion detection systems in wireless sensor networks, *in* 'Modeling, Simulation, and Applied Optimization (ICMSAO), 2015 6th International Conference on', IEEE, pp. 1–6.

Cervantes, C., Poplade, D., Nogueira, M. and Santos, A. (2015), Detection of sinkhole attacks for supporting secure routing on 6lowpan for internet of things, *in* '2015 IFIP/IEEE International Symposium on Integrated Network Management (IM)', IEEE, pp. 606–611.

Chamudeeswari, R. and Sumathi, P. (2017), 'Security attacks on routing protocols and intrusion detection in manet', *International Journal of Scientific Research and Management* **5**(9), 7067–7073.

Chein, M. and Mugnier, M.-L. (2008), *Graph-based knowledge representation: computational foundations of conceptual graphs*, Springer Science & Business Media.

Chelli, K. (2015), Security issues in wireless sensor networks: Attacks and countermeasures, *in* 'Proceedings of the World Congress on Engineering', Vol. 1.

Chen, X., Makki, K., Yen, K. and Pissinou, N. (2009), 'Sensor network security: A survey.', *IEEE Communications Surveys and Tutorials* **11**(2), 52–73.

Chouhan, S., Bose, R. and Balakrishnan, M. (2009), 'A framework for energy-consumption-based design space exploration for wireless sensor nodes', *IEEE transactions on Computer-aided design of integrated circuits and systems* **28**(7), 1017–1024.

Coppolino, L., D'Antonio, S., Garofalo, A. and Romano, L. (2013), Applying data mining techniques to intrusion detection in wireless sensor networks, *in* 'P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC), 2013 Eighth International Conference on', IEEE, pp. 247–254.

Deogirikar, J. and Vidhate, A. (2017), Security attacks in iot: A survey, *in* '2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC)', IEEE, pp. 32–37.

Dhand, G. and Tyagi, S. (2016), 'Data aggregation techniques in wsn: Survey', *Procedia Computer Science* **92**, 378–384.

Duhan, S. and Khandnor, P. (2016), Intrusion detection system in wireless sensor networks: A comprehensive review, *in* '2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT)', IEEE, pp. 2707–2713.

Dunkels, A., Gronvall, B. and Voigt, T. (2004), Contiki-a lightweight and flexible operating system for tiny networked sensors, *in* 'Local Computer Networks, 2004. 29th Annual IEEE International Conference on', IEEE, pp. 455–462.

Durrett, R. (2007), *Random graph dynamics*, Vol. 200, Cambridge university press Cambridge.

Elyengui, S., Bouhouchi, R. and Ezzedine, T. (2015), A comparative performance study of the routing protocols rpl, loadng and loadng-ctp with bidirectional traffic for ami scenario, *in* '2015 International Conference on Smart Grid and Clean Energy Technologies (ICSGCE)', IEEE, pp. 43–49.

Farooq, M. O., Sreenan, C. J., Brown, K. N. and Kunz, T. (2015), Rpl-based routing protocols for multi-sink wireless sensor networks, *in* '2015 IEEE 11th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)', IEEE, pp. 452–459.

Gai, K., Qiu, M., Ming, Z., Zhao, H. and Qiu, L. (2017), 'Spoofing-jamming attack strategy using optimal power distributions in wireless smart grid networks', *IEEE Transactions on Smart Grid* **8**(5), 2431–2439.

Ghosal, A. and Halder, S. (2013), Intrusion detection in wireless sensor networks: Issues, challenges and approaches, *in* 'Wireless networks and security', Springer, pp. 329–367.

Godsil, C. and Royle, G. F. (2013), *Algebraic graph theory*, Vol. 207, Springer Science & Business Media.

Goodfellow, I., Bengio, Y. and Courville, A. (2016), *Deep learning*, MIT press.

Gope, P., Lee, J. and Quek, T. Q. (2016), 'Resilience of dos attacks in designing anonymous user authentication protocol for wireless sensor networks', *IEEE Sensors journal* **17**(2), 498–503.

Goyal, P., Batra, S. and Singh, A. (2010), 'A literature review of security attack in mobile ad-hoc networks', *International Journal of Computer Applications* **9**(12), 11–15.

Granjal, J., Monteiro, E. and Silva, J. S. (2015), 'Security for the internet of things: a survey of existing protocols and open research issues', *IEEE Communications Surveys & Tutorials* **17**(3), 1294–1312.

Gross, J. L. and Tucker, T. W. (2001), *Topological graph theory*, Courier Corporation.

Grover, J. and Sharma, S. (2016), Security issues in wireless sensor network—a review, *in* '2016 5th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions)(ICRITO)', IEEE, pp. 397–404.

Guerroumi, M., Derhab, A. and Saleem, K. (2015), Intrusion detection system against sink hole attack in wireless sensor networks with mobile sink, *in* 'Information Technology-New Generations (ITNG), 2015 12th International Conference on', IEEE, pp. 307–313.

Gupta, P. and Kumar, P. R. (1999), Critical power for asymptotic connectivity in wireless networks, *in* 'Stochastic analysis, control, optimization and applications', Springer, pp. 547–566.

Gupta, P. and Kumar, P. R. (2000), 'The capacity of wireless networks', *IEEE Transactions on information theory* **46**(2), 388–404.

Hastie, T., Tibshirani, R. and Friedman, J. (2009), *The elements of statistical learning: data mining, inference, and prediction*, Springer Science & Business Media.

Hodo, E., Bellekens, X., Hamilton, A., Dubouilh, P.-L., Iorkyase, E., Tachtatzis, C. and Atkinson, R. (2016), Threat analysis of iot networks using artificial neural network intrusion detection system, *in* '2016 International Symposium on Networks, Computers and Communications (ISNCC)', IEEE, pp. 1–6.

Hossain, M. M., Fotouhi, M. and Hasan, R. (2015), Towards an analysis of security issues, challenges, and open problems in the internet of things, *in* '2015 IEEE World Congress on Services', IEEE, pp. 21–28.

Hu, X., Jang, J., SCHALES, D., Stoecklin, M. and Wang, T. (2018), 'Detection of beaconing behavior in network traffic'. US Patent 10,044,737.

Huang, F., Jin, X., Zhang, Y. and TANG, J. (2009), 'Energy consumption balanced wsn routing protocol based on gasa', *Chinese Journal of Sensors and Actuators* **22**(4), 586–592.

Igbe, O., Darwish, I. and Saadawi, T. (2016), Distributed network intrusion detection systems: An artificial immune system approach, *in* '2016 IEEE First International Conference on Connected Health: Applications, Systems and Engineering Technologies (CHASE)', IEEE, pp. 101–106.

Ioannou, C. and Vassiliou, V. (2016), The impact of network layer attacks in wireless sensor networks, *in* '2016 International Workshop on Secure Internet of Things (SIoT)', IEEE, pp. 20–28.

Ishaq, I., Carels, D., Teklemariam, G., Hoebeke, J., Abeele, F., Poorter, E., Moerman, I. and Demeester, P. (2013), 'Ietf standardization in the field of the internet of things (iot): a survey', *Journal of Sensor and Actuator Networks* **2**(2), 235–287.

Ishmanov, F., Malik, A. S. and Kim, S. W. (2011), 'Energy consumption balancing (ecb) issues and mechanisms in wireless sensor networks (wsns): a comprehensive overview', *European Transactions on Telecommunications* **22**(4), 151–167.

Jain, A. K. and Tokekar, V. (2015), Mitigating the effects of black hole attacks on aodv routing protocol in mobile ad hoc networks, *in* '2015 International Conference on Pervasive Computing (ICPC)', IEEE, pp. 1–6.

Jan, M. A., Nanda, P., He, X. and Liu, R. P. (2015), A sybil attack detection scheme for a centralized clustering-based hierarchical network, *in* '2015 IEEE Trustcom/BigDataSE/ISPA', Vol. 1, IEEE, pp. 318–325.

Jiao, Z., Zhang, B., Li, C. and Mouftah, H. T. (2016), 'Backpressure-based routing and scheduling protocols for wireless multihop networks: A survey', *IEEE Wireless Communications* **23**(1), 102–110.

Jordan, M. I. and Mitchell, T. M. (2015), 'Machine learning: Trends, perspectives, and prospects', *Science* **349**(6245), 255–260.

Jun, C. and Chi, C. (2014), Design of complex event-processing ids in internet of things, *in* '2014 Sixth International Conference on Measuring Technology and Mechatronics Automation', IEEE, pp. 226–229.

Kamble, A., Malemath, V. S. and Patil, D. (2017), Security attacks and secure routing protocols in rpl-based internet of things: Survey, *in* '2017 International Conference on Emerging Trends & Innovation in ICT (ICEI)', IEEE, pp. 33–39.

Kasinathan, P., Pastrone, C., Spirito, M. A. and Vinkovits, M. (2013), Denial-of-service detection in 6lowpan based internet of things, *in* '2013 IEEE 9th international conference on wireless and mobile computing, networking and communications (WiMob)', IEEE, pp. 600–607.

Kaur, M. and Singh, A. (2016), Detection and mitigation of sinkhole attack in wireless sensor network, *in* '2016 International conference on micro-electronics and telecommunication engineering (ICMETE)', IEEE, pp. 217–221.

Kenkre, P. S., Pai, A. and Colaco, L. (2015), Real time intrusion detection and prevention system, *in* 'Proceedings of the 3rd International Conference on Frontiers of Intelligent Computing: Theory and Applications (FICTA) 2014', Springer, pp. 405–411.

Khan, F. (2014), Secure communication and routing architecture in wireless sensor networks, *in* 'Consumer Electronics (GCCE), 2014 IEEE 3rd Global Conference on', IEEE, pp. 647–650.

Khan, F. I., Shon, T., Lee, T. and Kim, K. (2013), Wormhole attack prevention mechanism for rpl based lln network, *in* '2013 Fifth International Conference on Ubiquitous and Future Networks (ICUFN)', IEEE, pp. 149–154.

Khanmirza, H. and Yazdani, N. (2016), 'Game of energy consumption balancing in heterogeneous sensor networks', *Wireless Communications and Mobile Computing* **16**(12), 1457–1477.

Kharrufa, H., Al-Kashoash, H., Al-Nidawi, Y., Mosquera, M. Q. and Kemp, A. H. (2017), Dynamic rpl for multi-hop routing in iot applications, *in* '2017 13th Annual Conference on Wireless On-demand Network Systems and Services (WONS)', IEEE, pp. 100–103.

Kolias, C., Kambourakis, G., Stavrou, A. and Gritzalis, S. (2016), 'Intrusion detection in 802.11 networks: empirical evaluation of threats and a public dataset', *IEEE Communications Surveys & Tutorials* **18**(1), 184–208.

Kolias, C., Kambourakis, G., Stavrou, A. and Voas, J. (2017), 'Ddos in the iot: Mirai and other botnets', *Computer* **50**(7), 80–84.

Kumarage, H., Khalil, I., Tari, Z. and Zomaya, A. (2013*a*), 'Distributed anomaly detection for industrial wireless sensor networks based on fuzzy data modelling', *Journal of Parallel and Distributed Computing* **73**(6), 790–806.

Kumarage, H., Khalil, I., Tari, Z. and Zomaya, A. (2013*b*), 'Distributed anomaly detection for industrial wireless sensor networks based on fuzzy data modelling', *Journal of Parallel and Distributed Computing* **73**(6), 790–806.

Kushwaha, P., Buckchash, H. and Raman, B. (2017), Anomaly based intrusion detection using filter based feature selection on kdd-cup 99, *in* 'TENCON 2017-2017 IEEE Region 10 Conference', IEEE, pp. 839–844.

Lavric, A. and Popa, V. (2017), Internet of things and lora™ low-power wide-area networks: a survey, *in* '2017 International Symposium on Signals, Circuits and Systems (ISSCS)', IEEE, pp. 1–5.

Le, A., Loo, J., Chai, K. and Aiash, M. (2016), 'A specification-based ids for detecting attacks on rpl-based network topology', *Information* **7**(2), 25.

Le, A., Loo, J., Luo, Y. and Lasebae, A. (2011), Specification-based ids for securing rpl from topology attacks, *in* '2011 IFIP Wireless Days (WD)', IEEE, pp. 1–3.

Le, A., Loo, J., Luo, Y. and Lasebae, A. (2013), The impacts of internal threats towards routing protocol for low power and lossy network performance, *in* 'Computers and Communications (ISCC), 2013 IEEE Symposium on', IEEE, pp. 000789–000794.

Li, L., Zhang, H., Peng, H. and Yang, Y. (2018), 'Nearest neighbors based density peaks approach to intrusion detection', *Chaos, Solitons & Fractals* **110**, 33–40.

Li, S., Da Xu, L. and Zhao, S. (2015), 'The internet of things: a survey', *Information Systems Frontiers* **17**(2), 243–259.

Liao, H.-J., Lin, C.-H. R., Lin, Y.-C. and Tung, K.-Y. (2013), 'Intrusion detection system: A comprehensive review', *Journal of Network and Computer Applications* **36**(1), 16–24.

Liu, S. (2020), 'Iot market size worldwide 2017-2025'.
URL: *https://www.statista.com/statistics/976313/global-iot-market-size/*

Liu, X. (2015), 'Atypical hierarchical routing protocols for wireless sensor networks: A review', *IEEE Sensors Journal* **15**(10), 5372–5383.

Liu, X., Sheng, Z., Yin, C., Ali, F. and Roggen, D. (2017), 'Performance analysis of routing protocol for low power and lossy networks (rpl) in large scale networks', *IEEE Internet of Things Journal* **4**(6), 2172–2185.

Liu, Y., Li, Y. and Man, H. (2005), Mac layer anomaly detection in ad hoc networks, *in* 'Proceedings from the Sixth Annual IEEE SMC Information Assurance Workshop', IEEE, pp. 402–409.

Luo, F., Jiang, C., Zhang, H., Wang, X., Zhang, L. and Ren, Y. (2014), Node energy consumption analysis in wireless sensor networks, *in* '2014 IEEE 80th Vehicular Technology Conference (VTC2014-Fall)', IEEE, pp. 1–5.

Ma, J., Le, F., Russo, A. and Lobo, J. (2015), Detecting distributed signature-based intrusion: The case of multi-path routing attacks, *in* '2015 IEEE Conference on Computer Communications (INFOCOM)', IEEE, pp. 558–566.

Maleh, Y., Ezzati, A., Qasmaoui, Y. and Mbida, M. (2015), 'A global hybrid intrusion detection system for wireless sensor networks', *Procedia Computer Science* **52**, 1047–1052.

Manjeshwar, A. and Agrawal, D. P. (2001), Teen: a routing protocol for enhanced efficiency in wireless sensor networks, *in* 'null', IEEE, p. 30189a.

Mayzaud, A., Sehgal, A., Badonnel, R., Chrisment, I. and Schönwälder, J. (2014), A study of rpl dodag version attacks, *in* 'IFIP international conference on autonomous infrastructure, management and security', Springer, pp. 92–104.

Mendez, D. M., Papapanagiotou, I. and Yang, B. (2017), 'Internet of things: Survey on security and privacy', *arXiv preprint arXiv:1707.01879* .

Midi, D., Rullo, A., Mudgerikar, A. and Bertino, E. (2017), Kalis—a system for knowledge-driven adaptable intrusion detection for the internet of things, *in* 'Distributed Computing Systems (ICDCS), 2017 IEEE 37th International Conference on', IEEE, pp. 656–666.

Miranda, J., Gomes, T., Abrishambaf, R., Loureiro, F., Mendes, J., Cabral, J. and Monteiro, J. L. (2014), A wireless sensor network for collision detection on guardrails, *in* '2014 IEEE 23rd International Symposium on Industrial Electronics (ISIE)', IEEE, pp. 1430–1435.

Molisch, A. F., Balakrishnan, K., Chong, C.-C., Emami, S., Fort, A., Karedal, J., Kunisch, J., Schantz, H., Schuster, U. and Siwiak, K. (2004), 'Ieee 802.15. 4a channel model-final report', *IEEE P802* **15**(04), 0662.

Moon, S. Y., Kim, J. W. and Cho, T. H. (2014), An energy-efficient routing method with intrusion detection and prevention for wireless sensor networks, *in* 'Advanced Communication Technology (ICACT), 2014 16th International Conference on', IEEE, pp. 467–470.

Moudni, H., Er-Rouidi, M., Mouncif, H. and El Hadadi, B. (2016), Attacks against aodv routing protocol in mobile ad-hoc networks, *in* '2016 13th international conference on computer graphics, imaging and visualization (cgiv)', IEEE, pp. 385–389.

Mukherjee, S. and Sharma, N. (2012), 'Intrusion detection using naive bayes classifier with feature reduction', *Procedia Technology* **4**, 119–128.

Nawir, M., Amir, A., Yaakob, N. and Lynn, O. B. (2016), Internet of things (iot): Taxonomy of security attacks, *in* '2016 3rd International Conference on Electronic Design (ICED)', IEEE, pp. 321–326.

Oh, D., Kim, D. and Ro, W. (2014), 'A malicious pattern detection engine for embedded security systems in the internet of things', *Sensors* **14**(12), 24188–24211.

Osterlind, F., Dunkels, A., Eriksson, J., Finne, N. and Voigt, T. (2006), Cross-level sensor network simulation with cooja, *in* 'Local computer networks, proceedings 2006 31st IEEE conference on', IEEE, pp. 641–648.

Ozay, M., Esnaola, I., Vural, F. T. Y., Kulkarni, S. R. and Poor, H. V. (2015), 'Machine learning methods for attack detection in the smart grid', *IEEE transactions on neural networks and learning systems* **27**(8), 1773–1786.

Pan, S., Morris, T. and Adhikari, U. (2015), 'Developing a hybrid intrusion detection system using data mining for power systems', *IEEE Transactions on Smart Grid* **6**(6), 3104–3113.

Pantazis, N. A., Nikolidakis, S. A. and Vergados, D. D. (2013), 'Energy-efficient routing protocols in wireless sensor networks: A survey', *IEEE Communications surveys & tutorials* **15**(2), 551–591.

Pascanu, R., Gulcehre, C., Cho, K. and Bengio, Y. (2013), 'How to construct deep recurrent neural networks', *arXiv preprint arXiv:1312.6026* .

Patel, A., Patel, N. and Patel, R. (2015), Defending against wormhole attack in manet, *in* '2015 Fifth International Conference on Communication Systems and Network Technologies', IEEE, pp. 674–678.

Patel, A., Qassim, Q. and Wills, C. (2010), 'A survey of intrusion detection and prevention systems', *Information Management & Computer Security* **18**(4), 277–290.

Patel, A., Taghavi, M., Bakhtiyari, K. and JúNior, J. C. (2013), 'An intrusion detection and prevention system in cloud computing: A systematic review', *Journal of network and computer applications* **36**(1), 25–41.

Pathan, A.-S. K. (2016), *Security of self-organizing networks: MANET, WSN, WMN, VANET*, CRC press.

Peddabachigari, S., Abraham, A., Grosan, C. and Thomas, J. (2007), 'Modeling intrusion detection system using hybrid intelligent systems', *Journal of network and computer applications* **30**(1), 114–132.

Penrose, M. D. et al. (2016), 'Connectivity of soft random geometric graphs', *The Annals of Applied Probability* **26**(2), 986–1028.

Pongle, P. and Chavan, G. (2015*a*), 'Real time intrusion and wormhole attack detection in internet of things', *International Journal of Computer Applications* **121**(9).

Pongle, P. and Chavan, G. (2015*b*), A survey: Attacks on rpl and 6lowpan in iot, *in* 'Pervasive Computing (ICPC), 2015 International Conference on', IEEE, pp. 1–6.

Porambage, P., Braeken, A., Schmitt, C., Gurtov, A., Ylianttila, M. and Stiller, B. (2015), 'Group key establishment for enabling secure multicast communication in wireless sensor networks deployed for iot applications', *IEEE Access* **3**, 1503–1511.

Pu, C. (2018), Mitigating dao inconsistency attack in rpl-based low power and lossy networks, *in* '2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC)', IEEE, pp. 570–574.

Raghunandan, G. and Lakshmi, B. (2011), A comparative analysis of routing techniques for wireless sensor networks, *in* '2011 national conference on innovations in emerging technology', IEEE, pp. 17–22.

Rahman, R. A. and Shah, B. (2016), Security analysis of iot protocols: A focus in coap, *in* '2016 3rd MEC international conference on big data and smart city (ICBDSC)', IEEE, pp. 1–7.

Raju, I. and Parwekar, P. (2016), Detection of sinkhole attack in wireless sensor network, *in* 'Proceedings of the Second International Conference on Computer and Communication Technologies', Springer, pp. 629–636.

Rath, M., Pati, B., Panigrahi, C. R. and Sarkar, J. L. (2019), Qtm: A qos task monitoring system for mobile ad hoc networks, *in* 'Recent Findings in Intelligent Computing Techniques', Springer, pp. 543–550.

Raza, S., Wallgren, L. and Voigt, T. (2013), 'SVELTE: Real-time intrusion detection in the Internet of Things', *Ad hoc networks* **11**(8), 2661–2674.

Rehman, A., Khan, M. M., Lodhi, M. A. and Hussain, F. B. (2016), Rank attack using objective function in rpl for low power and lossy networks, *in* '2016 International Conference on Industrial Informatics and Computer Systems (CIICS)', IEEE, pp. 1–5.

Ring, T. (2015), 'Connected cars–the next targe tfor hackers', *Network Security* **2015**(11), 11–16.

Sabahi, F. and Movaghar, A. (2008), Intrusion detection: A survey, *in* '2008 Third International Conference on Systems and Networks Communications', IEEE, pp. 23–26.

Schmidhuber, J. (2015), 'Deep learning in neural networks: An overview', *Neural networks* **61**, 85–117.

Seah, W. K., Eu, Z. A. and Tan, H.-P. (2009), Wireless sensor networks powered by ambient energy harvesting (wsn-heap)-survey and challenges, *in* '2009 1st International Conference on Wireless Communication, Vehicular Technology, Information Theory and Aerospace & Electronic Systems Technology', Ieee, pp. 1–5.

Shaffer, S., Vasseur, J.-P. and Shetty, S. J. (2015), 'Dynamic reroute scheduling in a directed acyclic graph (dag)'. US Patent 8,937,886.

Shamshirband, S., Amini, A., Anuar, N. B., Kiah, M. L. M., Teh, Y. W. and Furnell, S. (2014), 'D-ficca: A density-based fuzzy imperialist competitive clustering algorithm for intrusion detection in wireless sensor networks', *Measurement* **55**, 212–226.

Sharma, S. and Gupta, R. (2015), 'Intrusion detection system: A review', *International Journal of Security and Its Applications* **9**(5), 69–76.

Shelby, Z. and Bormann, C. (2011), *6LoWPAN: The wireless embedded Internet*, Vol. 43, John Wiley & Sons.

Shelby, Z., Hartke, K. and Bormann, C. (2014), The constrained application protocol (coap), Technical report.

Shreenivas, D., Raza, S. and Voigt, T. (2017), Intrusion detection in the rpl-connected 6lowpan networks, *in* 'Proceedings of the 3rd ACM International Workshop on IoT Privacy, Trust, and Security', ACM, pp. 31–38.

Sicari, S., Rizzardi, A., Grieco, L. A. and Coen-Porisini, A. (2015), 'Security, privacy and trust in internet of things: The road ahead', *Computer networks* **76**, 146–164.

Siddiqui, M., Villazón, A. and Fahringer, T. (2006), Grid capacity planning with negotiation-based advance reservation for optimized qos, *in* 'SC'06: Proceedings of the 2006 ACM/IEEE conference on Supercomputing', IEEE, pp. 21–21.

Silva, B. N., Khan, M. and Han, K. (2018), 'Internet of things: A comprehensive review of enabling technologies, architecture, and challenges', *IETE Technical Review* **35**(2), 205–220.

Singh, A., Ramakrishnan, C. and Smolka, S. A. (2009), Query-based model checking of ad hoc network protocols, *in* 'International Conference on Concurrency Theory', Springer, pp. 603–619.

Singh, B. (2018), 'Load balancing for multipath groups routed flows by re-associating routes to multipath groups'. US Patent 10,097,467.

Singh, S. K., Paulus, R., Jaiswal, A. and Kumar, M. (2014), 'Analysis of energy model and performance of ieee 802.15. 4 wsns under different duty cycle', *J. Electron. Commun. Eng* **9**, 48–54.

Sinha, P., Jha, V., Rai, A. K. and Bhushan, B. (2017), Security vulnerabilities, attacks and countermeasures in wireless sensor networks at various layers of osi reference model: A survey, *in* '2017 International Conference on Signal Processing and Communication (ICSPC)', IEEE, pp. 288–293.

Sonar, K. and Upadhyay, H. (2014), 'A survey: Ddos attack on internet of things', *International Journal of Engineering Research and Development* **10**(11), 58–63.

Song, H. M., Kim, H. R. and Kim, H. K. (2016), Intrusion detection system based on the analysis of time intervals of can messages for in-vehicle network, *in* '2016 international conference on information networking (ICOIN)', IEEE, pp. 63–68.

Stehlik, M., Matyas, V. and Stetsko, A. (2016), Towards better selective forwarding and delay attacks detection in wireless sensor networks, *in* '2016 IEEE 13th International Conference on Networking, Sensing, and Control (ICNSC)', IEEE, pp. 1–6.

Taylor, C. and Johnson, T. (2015), Strong authentication countermeasures using dynamic keying for sinkhole and distance spoofing attacks in smart grid networks, *in* '2015 IEEE Wireless Communications and Networking Conference (WCNC)', IEEE, pp. 1835–1840.

Tomić, I. and McCann, J. A. (2017), 'A survey of potential security issues in existing wireless sensor network protocols', *IEEE Internet of Things Journal* **4**(6), 1910–1923.

Tong, S. and Koller, D. (2001), 'Support vector machine active learning with applications to text classification', *Journal of machine learning research* **2**(Nov), 45–66.

Torres, P., Catania, C., Garcia, S. and Garino, C. G. (2016), An analysis of recurrent neural networks for botnet detection behavior, *in* '2016 IEEE biennial congress of Argentina (ARGENCON)', IEEE, pp. 1–6.

Van, N. T., Thinh, T. N. and Sach, L. T. (2017), An anomaly-based network intrusion detection system using deep learning, *in* '2017 International Conference on System Science and Engineering (ICSSE)', IEEE, pp. 210–214.

Wallgren, L., Raza, S. and Voigt, T. (2013), 'Routing attacks and countermeasures in the rpl-based internet of things', *International Journal of Distributed Sensor Networks* **9**(8), 794326.

Wang, S.-S., Yan, K.-Q., Wang, S.-C. and Liu, C.-W. (2011), 'An integrated intrusion detection system for cluster-based wireless sensor networks', *Expert Systems with Applications* **38**(12), 15234–15243.

Weekly, K. and Pister, K. (2012), Evaluating sinkhole defense techniques in rpl networks, *in* '2012 20th IEEE International Conference on Network Protocols (ICNP)', IEEE, pp. 1–6.

Wei, W., Sun, Z., Song, H., Wang, H., Fan, X. and Chen, X. (2017), 'Energy balance-based steerable arguments coverage method in wsns', *IEEE Access* **6**, 33766–33773.

West, D. B. et al. (1996), *Introduction to graph theory*, Vol. 2, Prentice hall Upper Saddle River, NJ.

Xiao, L., Li, Y., Han, G., Liu, G. and Zhuang, W. (2016), 'Phy-layer spoofing detection with reinforcement learning in wireless networks', *IEEE Transactions on Vehicular Technology* **65**(12), 10037–10047.

Xie, M., Chen, H.-H., Hu, J. and Han, S. (2012), 'Scalable hyper-grid k-NN-based online anomaly detection in wireless sensor networks', *IEEE Transactions on Parallel and Distributed Systems* **99**(1), 1.

Xie, M., Hu, J., Guo, S. and Zomaya, A. Y. (2017), 'Distributed Segment-Based Anomaly Detection With Kullback–Leibler Divergence in Wireless Sensor Networks', *IEEE Transactions on Information Forensics and Security* **12**(1), 101–110.

Xu, J., Jin, N., Lou, X., Peng, T., Zhou, Q. and Chen, Y. (2012), Improvement of leach protocol for wsn, *in* '2012 9th International Conference on Fuzzy Systems and Knowledge Discovery', IEEE, pp. 2174–2177.

Yadav, J. and Kumar, M. (2016), Detection of wormhole attack in wireless sensor networks, *in* 'Proceedings of International Conference on ICT for Sustainable Development', Springer, pp. 243–250.

Yang, X., Deng, D. and Liu, M. (2015), An overview of routing protocols on wireless sensor network, *in* '2015 4th International Conference on Computer Science and Network Technology (ICCSNT)', Vol. 1, IEEE, pp. 1000–1003.

Yessad, N., Omar, M., Tari, A. and Bouabdallah, A. (2018), 'Qos-based routing in wireless body area networks: A survey and taxonomy', *Computing* **100**(3), 245–275.

Zhang, M., Bingham, J. D., Erickson, J. and Sorin, D. J. (2014), Pvcoherence: Designing flat coherence protocols for scalable verification, *in* '2014 IEEE 20th International Symposium on High Performance Computer Architecture (HPCA)', IEEE, pp. 392–403.

Zhao, M., Kumar, A., Chong, P. H. J. and Lu, R. (2017), 'A comprehensive study of rpl and p2p-rpl routing protocols: Implementation, challenges and opportunities', *Peer-to-Peer Networking and Applications* **10**(5), 1232–1256.

Zheng, Z., Liu, A., Cai, L. X., Chen, Z. and Shen, X. S. (2016), 'Energy and memory efficient clone detection in wireless sensor networks', *IEEE Transactions on Mobile Computing* **15**(5), 1130–1143.

Zhou, H.-Y., Luo, D.-Y., Gao, Y. and Zuo, D.-C. (2011), 'Modeling of node energy consumption for wireless sensor networks', *Wireless Sensor Network* **3**(01), 18.

Zhu, J., Zou, Y. and Zheng, B. (2017), ‘Physical-layer security and reliability challenges for industrial wireless sensor networks’, *IEEE Access* **5**, 5313–5320.