Journal of Information Security and Applications 53 (2020) 102530

Contents lists available at ScienceDirect

Journal of Information Security and Applications

journal homepage: www.elsevier.com/locate/jisa

An architecture for resilient intrusion detection in ad-hoc networks

Mohammed Al Qurashi[a,*], Constantinos Marios Angelopoulos[a], Vasilios Katos[a]

Bournemouth University, Fern Barrow, Poole, UK

ARTICLE INFO

Keywords:
IDS
IoT
Wireless sensor networks
RPL Protocol
Sinkhole attack

ABSTRACT

We study efficient and lightweight Intrusion Detection Systems (IDS) for ad-hoc networks via the prism 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) has been developed taking into consideration the architecture of conventional computer networks, and as such they do not efficiently address the paradigm of ad-hoc networks, that is highly relevant in emergent networks, such as the Internet of Things (IoT). In this context, the network properties of resilience and redundancy have not been studied yet. In this work, we firstly identify a trade-off between the communication overhead and energy consumption 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 such networks as Random Geometric Graphs; 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 a 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. Our detailed experimental evaluation demonstrates significant performance gains in terms of communication overhead and energy consumption 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. Conducted experiments show that our proposed IDS architecture is resilient against frequent topology changs due to nodes failures.

1. Introduction

Internet of Things represents an emerging networking paradigm both in qualitative and quantitative terms. Following previous paradigm shifts - such as Web 2.0, Cloud Computing and the rise of Social Media – that enabled computers and people to be interconnected, now *things* and *machines* are able to seamlessly exchange information and data over the Internet.

The profound impact of IoT is highlighted by the number of IoT devices being deployed, as well as the projected growth of the corresponding market; a number of studies predict that by 2025 more than 55 billion IoT devices will be deployed and around 15USD trillion will be invested in IoT in aggregate between 2017 and 2025 [16]. Furthermore, IoT has found its way to several verticals such as healthcare, manufacturing and critical infrastructure [14]. The importance and critical role of IoT in the modern economy has naturally made such systems targets of malicious activity, such as the infamous Mirai botnet [11] or the case where hackers were able to affect the steering and braking systems of a Jeep car [28]. It is worth noting that IoT networks and systems are not only the subjects of but also the means for deploying attacks (such as in the case of Mirai). Ironically, the deployment of such attacks has also been facilitated - among other factors - by the use of standardised Internet technologies that have enabled the paradigm in the first place.

Broadly speaking, security controls can be taxonomized in three layers. At the first layer (also known as the first line of defence) lie preventative countermeasures, such as authentication and access control mechanisms, cryptography, firewalls, and so forth. At the second layer (also known as the second line of defence) lie detection countermeasures that are engaged *during* an attack, such as Intrusion Detection Systems. Finally, at the last layer lie recovery measures and processes for post-incident management, such as triaging, security information incident management and digital forensics. Due to the inherent and particular characteristics of

* Corresponding author.
E-mail addresses: malqurashi@bournemouth.ac.uk (M.A. Qurashi), mangelopoulos@bournemouth.ac.uk (C.M. Angelopoulos), vkatos@bournemouth.ac.uk (V. Katos).

https://doi.org/10.1016/j.jisa.2020.102530
2214-2126/

IoT (i.e. highly constrained, deployed in mass numbers and their ephemeral availability), the corresponding cyber security measures need to be revisited.

A considerable amount of research has been carried out in Intrusion Detection Systems (IDS) concerning deployment architectures, detection strategies and algorithms. However, currently available IDS are designed for "traditional" computer networks, thus making strong assumptions about the system the IDS will be deployed in; e.g. that each node of the network is powerful in terms of resources, is always available and that the nodes communicate over a reliable and high-capacity network. As such, there is a need for innovative, lightweight IDS that will efficiently address the IoT paradigm.

Our contributions. A wireless sensor actuator network (WSAN) consists of a set of small and inexpensive autonomous devices deployed over an area of interest. The devices - commonly referred to as motes or simply sensors - are able to wirelessly communicate with their peers and, in spite of their highly constrained nature, to collaboratively carry out complex tasks. WSNs are a key enabling technology for the IoT and as such share several common characteristics. For the same reason, WSNs have provided an ideal R&D platform for several IoT protocols and technologies, such as the CoAP [29], 6LoWPAN [30] and 6TiSCH [5]. In this work we study efficient and lightweight Intrusion Detection Systems for IoT deployments via the prism of IPv6-enabled WSNs.

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 inherent 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 there exists an underlying threshold behaviour in the efficiency of the IDS that is related to the connectivity threshold of the RGG model. This allows us to conjure that in peer-to-peer IoT networks, the number of IDS agents that need to be deployed in order to achieve a high detection rate is constant and a function of the ratio between the network area size and the communication range of the nodes. Furthermore, we show that the proposed architecture is able to efficiently cope with and mitigate the effects of nodes failures on the efficiency of the IDS.

The rest of the paper is organised as follows: Section 2 presents the current state-of-the-art with a special emphasis on the most important contributions in Intrusion Detection Systems in WSNs. Sections 3 and 4 introduce the proposed network model and adopted IDS architecture based on Random Geometric Graphs. Section 5 presents the performance evaluation of the proposed approach and discusses the simulation results and findings. In Section 6, we introduce possible directions and extensions for future research that are derived upon the contributions of this research. Finally, conclusions are discussed in Section 7.

2. Related work

IDS for WSNs and the IoT have attracted significant research interest in the past years; in this section we focus on the most important contributions in the area. Coarsely speaking, IDSs can be classified based on their architecture into systems following a centralized, distributed or a hybrid architecture. In centralized IDSs, all relevant monitoring and detection information has to be reported to a centrally located base station where sophisticated detection algorithms are executed. Here, the base station is considered powerful in terms of processing capabilities and available memory and energy. On the other hand, in distributed architectures each individual network node is running an IDS agent and in cooperation with other agents in the network, they collaboratively detect any on-going attacks. Finally, hybrid IDS architectures demonstrate a combination of the centralized and distributed architectures in an effort to exploit and combine the advantages of each individual approach.

2.1. Centralized IDS architectures

In [19], authors proposed a detection method for WSNs that aggregates the functions of the IDS with those of the intrusion prevention system. In this approach, symmetric encryption and oneâway hash functions are used to establish the routing path between the base station (BS) and other nodes of the network. Their results showed that their approach reduces the total amount of required energy, in some cases quite significantly. However, this comes at a cost of increased computational overhead due to the use of a symmetric key.

In [18], authors proposed an IDS 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.

In [10] authors proposed a centralised IDS 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 and EPL.The experiment result reveals that their approach had increased the computation overhead and consumed less memory comparing with traditional IDS.

2.2. Distributed IDS architectures

In [12], authors proposed a distributed anomaly detection framework for industrial WSNs that uses in-network hierarchical processing. The nodes of the network are first clustered using fuzzy c-means clustering, and then run an incremental model to score local and global outliers. The proposed method was evaluated both on synthetic and real data and results showed a better trade-off to be achieved between achieving high-accuracy and introducing a smaller computational and communication overhead to the network.

In [17], authors proposed a lightweight IDS for WSNs that combines anomaly-based detection (using support vector machine algorithms) with signature based rules. The authors study cluster-based topologies that reduce communication costs thus leading to extending the network lifetime. Simulation results show that their proposed model can detect abnormal events efficiently and has a high detection rate with a lower false-alarm rate.

In [34], authors proposed an energy-efficient location-aware clone detection protocol in densely deployed WSNs, which can guarantee successful clone attack detection and maintain satisfactory network lifetime. Their approach is based on Specification Protocol Analysis that exploits location information of sensors to verify the legitimacy of other sensors in the proximity and detect clone attacks. Simulation results showed that this approach yielded high accuracy in detecting clone attacks and relatively lower energy consumption compared to other approaches.

In [4], authors proposed a distributed anomaly detection method for WSNs that uses both signature and anomaly-based detection techniques. Their framework is composed of a central agent and a number of local agents. The central agent employs data mining detection techniques whereas the local agents uses lighter detection techniques. Decision trees have been adopted as classification algorithm in the detection process of the central agent and

their behaviour has been analysed in selected attacks scenarios. The empirical results revealed that this method exhibits low detection accuracy and high false positive rates.

In [33], authors 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 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 limited to hierarchical network and my applied for flat network with additional requirements. And also it assumes static architecture that does not consider frequent dynamic changes of WSN.

In [8], authors proposed a distributed IDS for IoT. Their detection module is installed at each node, which monitors and detects attacks using a statistical analysis tool based on Binary Logistic Regression (BLR). Their results show that this approach yielded high accuracy in detecting attacks, but relatively high energy consumption compared to other approaches.

In [15], authors proposed a distributed IDS following blockchain model. They developed a blockchain-based framework called CB-SigIDS by combining blockchains with distributed signature-based IDS in IoT. Blockchain- based IDSs in IoT are in the infancy stage, authors highlight several issues with this approach. Further researches are needed to investigate the efficiency and effectiveness of blockchain distribution.

In [3], authors proposed a distributed IDS 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.

In [32], authors 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.

In [20], authors 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.

2.3. Hybrid IDS architectures

In [24] authors introduced a detection method that uses misuse and anomaly detection techniques to detect sinkhole and deprivation attacks in WSNs. This method provisions local agents that are installed in each wireless sensor node as well as a central detection agent deployed at the base station. The local IDS agents are responsible for analysing traffic flow handled by the local node, gathering control data and sending it to the central agent. With such an arrangement local nodes are able to detect suspicious activities and to collaboratively contribute to the global detection process. The authors evaluated their approach through simulations and noted that a high detection rate can be achieved. However, the proposed approach does not consider the highly-constraint nature of WSN and the limitations it poses on real-life systems.

In [27] the authors 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.

In [13] the authors proposed a specification-based intrusion detection system for IoT to detect attacks on RPL-based networks. Their approach leverages a hybrid and partially 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.

In [23] the authors proposed a hybrid intrusion detection system for IoT consisting 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 to monitor 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 sized networks.

In [31], authors proposed a hybrid IDS for IoT by extending [27]. 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 detection 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.

The current state-of-the-art on IDSs for WSN and IoT networks are still resource-intensive and do not seem to adequately address the highly constrained nature of the underlying devices (Table 1). Centralised IDS architectures introduce significant communication overheads to the network as the base station (also known as Sink) injects and receives large numbers of requests to and from the nodes related to IDS data collection. Moreover, in the special case of multi-hop peer-to-peer networks, bottleneck effect phenomena emerge in the areas close to the Sink as the corresponding nodes relay data from/to the rest of the network. Distributed IDS architectures largely 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. To the best of our

Table 1
Comparison of State-of-the-art IDSs.

Proposed approach	Detection technique	Architecture	Security Attacks	Limitations/ Drawbacks
Moon et al. [19]	Signature Based	centralized	specific	Energy consumption, Very low detection rate
Coppolino et al. [4]	Anomaly Signature Based	distributed	specific	complex detection algorithm
Maleh et al. [17]	Anomaly based	distributed	general	Communication overhead,
Raza et al. [27]	Anomaly based	Hybrid	Routing attack	Communication overhead
Kumarage et al. [12]	Anomaly based	distributed	general	Communication overhead
Xie et al. [33]	Anomaly based	Hybrid	specific	Communication overhead within local cluster
Zheng et al. [34]	Specification protocol analysis	distributed	specific	Cooperation between nodes increased Energy consumption
Xie et al. [33]	Anomaly based	distributed	general	Limited to hierarchical networks
Wang et al. [32]	Anomaly /Signature Based	distributed	specific	Communication overhead
Ponomarchuk et al. [24]	Traffic analysis	distributed	general	Communication overhead
Jun et al. [10]	Specification-based	centralized	general	computation overhead, Communication overhead
le at al. [13]	Specification-based	Hybrid	Routing attack	Limited to cluster based networks
Shreenivas et al. [31]	Hybrid	Hybrid	Routing attack	Communication overhead
Ponoglo et al. [23]	Anomaly based	Hybrid	Routing attack	Limited to small size networks.
Cervantes et al. [3]	Hybrid	distributed	Routing attack	No details provided
Oh et al. [20]	Signature Based	distributed	conventional attacks	Only detect known attacks
Ioannou et al. [8]	statistical analysis	distributed	Routing attack	Energy consumption,
li et al. [14]	Signature Based	distributed	Flooding attacks	Energy consumption, only detect known attacks,
Midi et al. [18]	Hybrid	centralised	DoS attack	Energy consumption,

knowledge, the subject of resilience of the IDS architecture has not yet been extensively investigated in current-state-of-the-art IDS in WSNs and IoT.

A possible solution to overcome these challenges is to use a hybrid architecture which combines centralised and distributed IDS agents thus avoiding the disadvantages of centralised and distributed architectures. Hybrid architecture would provide better control of the IDS architecture since the centralised IDS agent is more powerful and rich in resources that are able to perform more complex algorithms and processes. Furthermore, a hybrid architecture placement scheme may reduce the communication overheads and energy consumption associated with the detection processes and the communication between base station and nodes when it is integrated with partly distributed architecture. We use Graph theory to design our solution in order to properly solve the particular characteristics of the IoT and Wireless Sensor Network paradigms. The proposed system aims to address 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). In particular, Random Graphs have been used as they are 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. Random Geometric Graph represents the actual placing of the set of n vertices uniformly and randomly at the area of interest.

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 substantially reduced while maintaining sufficiently high detection rates.

3. The network model

The paradigm of IoT envisions the massive and seamless connection of embedded systems, smart devices and things over the Internet. Wireless Sensor & Actuator Networks (WSANs) apart from being a key enabling technology many industrial applications, also carry several characteristics that are typically found in several IoT systems. WSANs comprise of a big number of ultra-small sensor devices (which we also refer to as sensors), whose purpose is to monitor local environmental conditions (e.g. ambient luminance, temperature, etc.) and drive actuators (e.g. switches, valves, etc.). Each sensor is a fully-autonomous computing and communication device, characterized mainly by its constrained nature in terms of available power supply (battery), its transmission range r, the energy cost of data transmission and the (limited) processing and memory capabilities. In this work we focus our study on WSANs where the sensors are static and are deployed over the network area uniformly at random.

There is a special node within the network the called Sink S, that represents the gateway device located on the edge of the WSAN network. In contrast to the WSAN nodes, the Sink is assumed to be powerful in terms of computing power, memory and energy supply. It is also the device that initiates the self-organisation of the network.

We consider that the random uniform placement of the sensors inside the network area is abstracted by RGG as we had used in [1]. 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 quite realistic model for WSANs as it captures to a great extent the communication structure of real networks (at least their spatial aspects). For instance, in [2] authors employ the $\mathcal{G}(n; R)$ model in order to study the performance of a new data collection strategy in WSN by mobile sink. The same model can also be used to abstract emerging ad-hoc networks, such as WSNs using 6TiSCH and IEEE 802.15.4e, that are very relevant to time synchronized industrial networks [25,26].

More formally, considering an area $\mathcal{A} \subset \mathbb{R}^2$ in a two dimensional space, an instance of the RGG 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

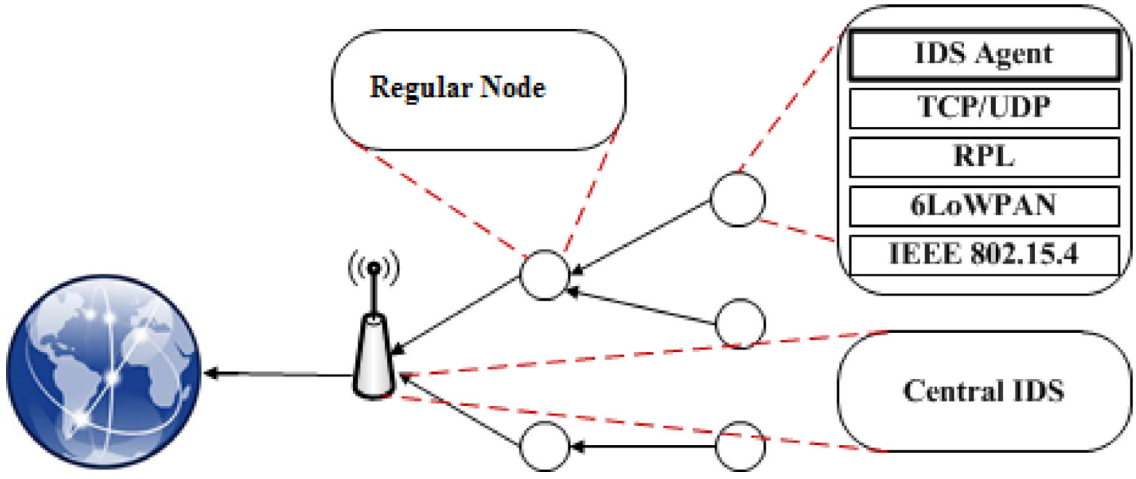


Fig. 1. The proposed IDS architecture.

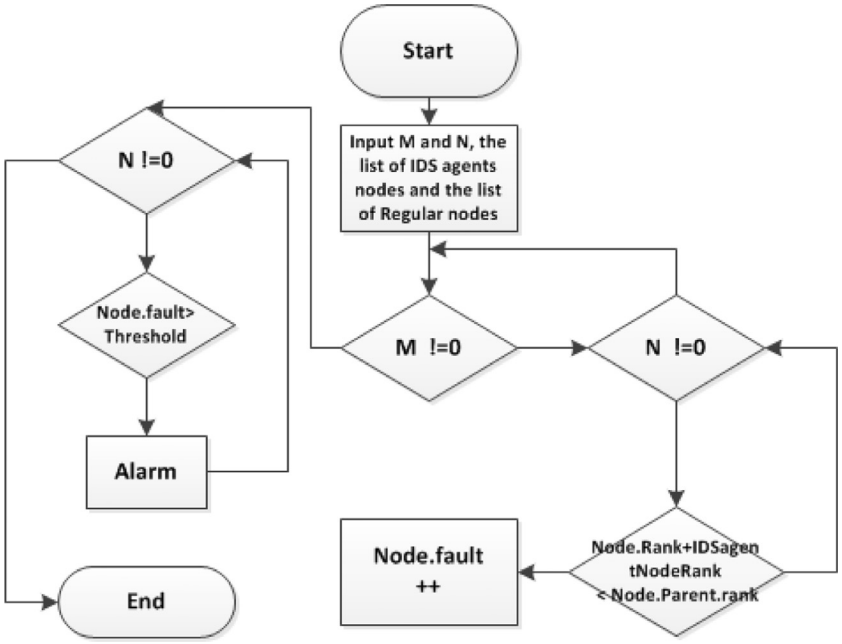


Fig. 2. Algorithm of detecting sinkhole attacks.

In [7,22] it is shown that the connectivity threshold for $\mathcal{G}(\mathcal{X}_n; r)$ is

$$r_c = \sqrt{\frac{\ln n}{\pi n}} \tag{1}$$

This way, the RGG model provides us with 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.

4. The proposed IDS architecture

We propose an 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 Fig. 1. 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 lest one 1-hop neighbour running the IDS agent. In graph-theoretical terms, such a subset would be a vertex cover 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. This set corresponds to the minimum vertex cover for the corresponding RGG graph.

We apply the aforementioned approach on the state-of-the-art IDS for WSN by Raza et al. called SVELTE [27]. In their work, the authors consider multi-hop peer-to-peer IPv6-enabled WSNs running the 6LoWPAN stack [30] on ContikiOS [6]. They develop an IDS following a hybrid architecture that consists of a centralized module running on the Sink and a distributed agent running on each individual sensor node. The centralized 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

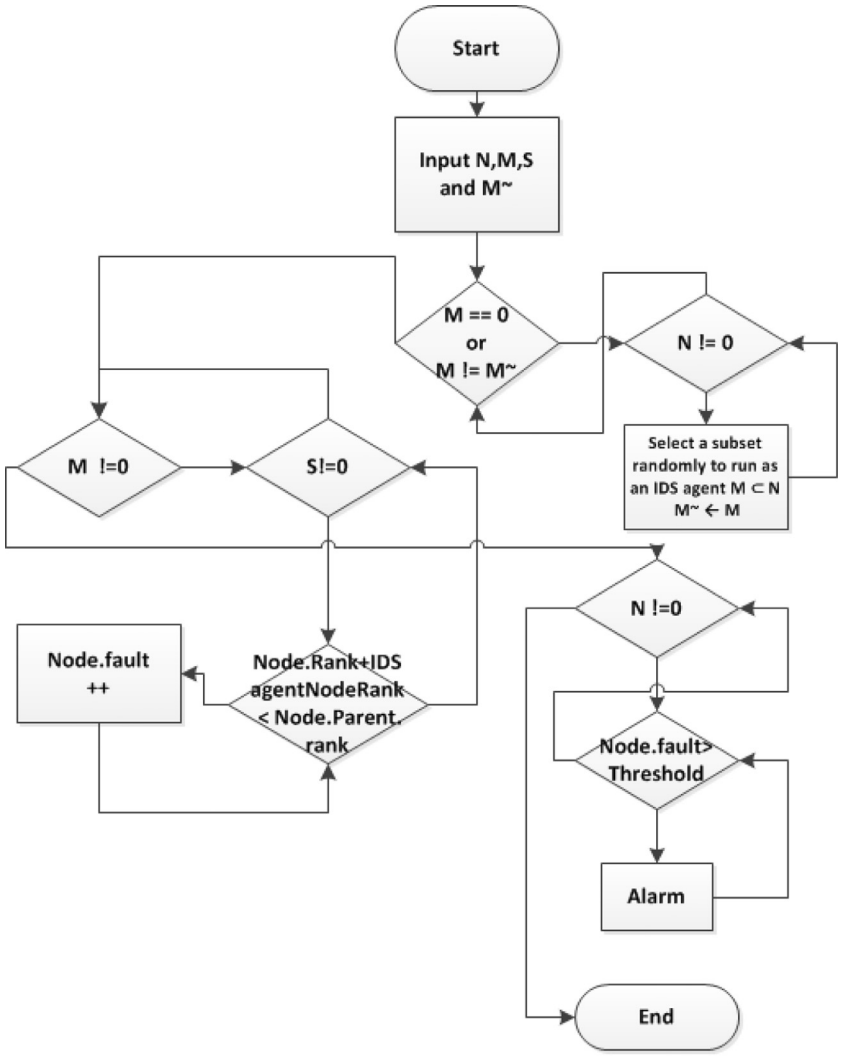


Fig. 3. Process of monitoring the distributed IDS agents and detecting sinkhole attacks.

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 to its neighbours a significantly smaller rank. 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 to go through the compromised node.

As already mentioned, 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 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 the sinkhole attack - by detecting corresponding anomalies as shown in Algorithm 1 lines 9 and 10, for example, checking 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 5 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

Algorithm 1 Algorithm for detecting sinkhole attacks.

Require: $M \leftarrow$ the list of IDS agents nodes
Require: $N \leftarrow$the list of Regular nodes
1: **for** Node in M **do**
2: **for** Node in N **do**
3: **if** (Node.Rank+IDSagentNodeRank
4: < Node.Parent.rank) **then**
5: Node.fault=Node.fault+1
6: **end if**
7: **end for**
8: **end for**
9: **for** Node in N **do**
10: **if** Node.fault>Threshold **then**
11: Alarm
12: **end if**
13: **end for**

availability of the network as 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 our 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, as in the case of RPL. This network information can be easily shared with or even be monitored by 1-hop neighbouring nodes (Algorithm 1 lines 1–5). 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. Furthermore, we propose an algorithm to select at random a subset of the nodes as shown in Algorithm 2

Algorithm 2 Algorithm for monitoring the distributed IDS agents and detecting sinkhole attacks.

Require: $N \leftarrow$the list of nodes
Require: $M \leftarrow$ the list of selected nodes to run as IDS agents
Require: $S \leftarrow$the list of regular nodes
Require: $M\prime \leftarrow$M
1: **while** $M = \emptyset$ OR $M \neq M\prime$ **do**
2: **while** $N \neq \emptyset$ **do**
3: Select a subset randomly to run as an IDS agent $M \subset N$
4: $M\prime \leftarrow M$
5: $S \leftarrow N - M$
6: **return** $A, A\prime, S$
7: **end while**
8: **end while**
9: **for** Node in M **do**
10: **for** Node in S **do**
11: **if** (Node.Rank+IDSagentNodeRank
12: < Node.Parent.rank) **then**
13: Node.fault++
14: **end if**
15: **end for**
16: **end for**
17: **for** Node in N **do**
18: **if** Node.fault>Threshold **then**
19: Alarm
20: **end if**
21: **end for**

line 3 based on the connectivity threshold, and also to maintain and monitor the distributed IDS agents against node failures. The number of nodes in this subset should be equal to the connectivity threshold where they are able to efficiently monitor the network and detect malicious nodes. As expressed in Algorithm 2 line 1, the central IDS frequently checks the set of IDS agents against node failures. Firstly, the central IDS constructs the network and then randomly selects a subset of the node to perform the distributed IDS agent. In case where any of the IDS nodes fail to communicate with the central IDS, the central IDS agent will re-run our proposed algorithm to select at random a new subset of the nodes to act as IDS agents.

In this work, we focus on experimentally investigating and empirically evaluating our approach using SVELTE as an indicative example of an IDS for ad-hoc networks. We note that this choice is made without any loss of generality. In particular, we focus on evaluating the trade-off between the potentially reduced 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. And also, we evaluate the reliance and recovery of the proposed approach against nodes failure to communicate the central IDS.

5. Performance evaluation

5.1. Simulation set-up

We ran our experiments using the Cooja emulator [21], which provides a detailed cross-layer simulation for WSNs running the 6LoWPAN stack. We ran our experiments in two parts. Firstly, we evaluate the efficiency and effectiveness of our proposed approach. Then, we study the resilience and redundancy of the proposed architecture.

We consider three qualitatively distinct network densities as these 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 Eq. (1), for each value of n, the corresponding network connectivity threshold is r_c: {18.5; 14.3; 11} respectively. Therefore, by setting the sensors' communication range to be $r = 20$, we get three network set ups where r is (a) almost equal to; (b) $\times$ 1.5 and (c) $\times$ 2 the connectivity threshold (Eq. (1)), thus resulting in (a) *sparse*, (b) *normal* and (c) *dense* networks. Figs. 4–6 provide a visual representation of the various network densities.

In the first part of our experiment, we consider five scenarios for each network density 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). Moreover, 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).

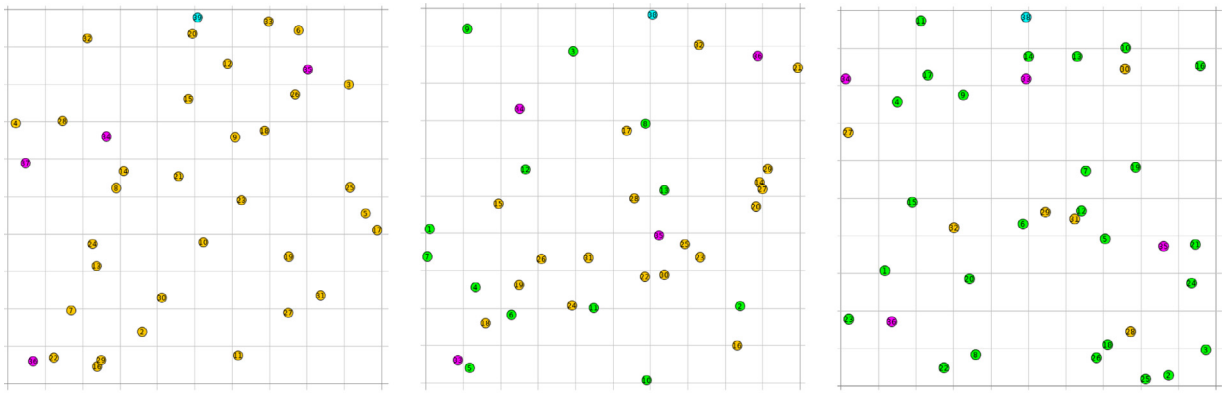
Furthermore, for each network density in the second part of our experiment we consider the number of IDS agents based on the connectivity threshold that we find later. The result we found from the first experiment that the connectivity threshold indicates the number of IDS agents needed to efficiently monitor the ad-hoc wireless network. The required number of IDS agents in the network to achieve these levels of high detection rates is independent of the network population and in fact constant. In the second part of our experiments, we consider the number of IDS agents randomly placed in the network based on the connectivity threshold which in our case is 25 IDS agents for all network densities $n \in \{32, 64, 128\}$ and any remaining nodes are regular nodes. We gradually drop off some nodes that perform IDS agent 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 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 s where nodes generate and transmit data approximately every second. For each scenario and each performance metric we compute the average values with 95% confidence intervals.

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 (Eqs. (2, 3) and (4)).

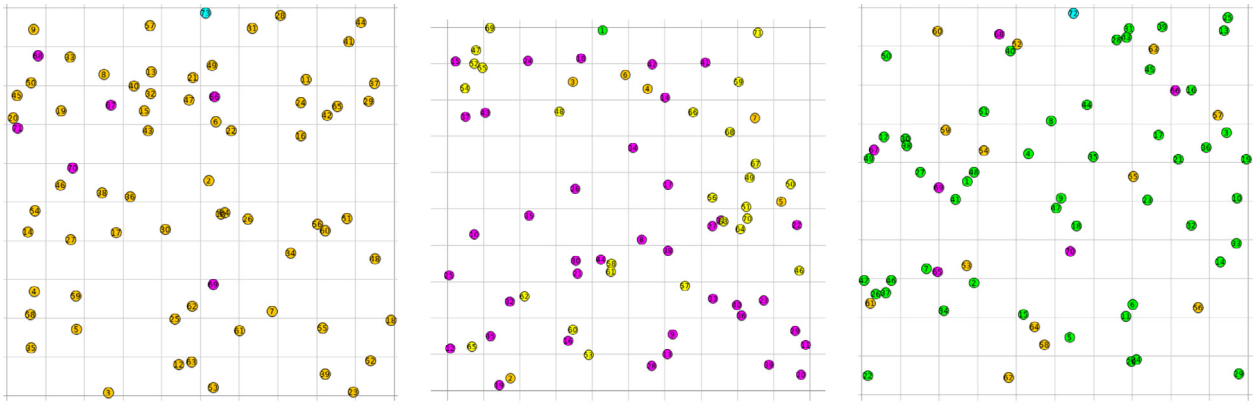
$$E_{total} = E_{sen \times 0.0545mA} + E_{cpu \times 1.8mA} + E_{transmit \times 19.5mA + listen \times 21.8mA} \quad (2)$$

$$E(mJ) = \frac{E_{total} \times 3V}{4396 \times 8} \quad (3)$$



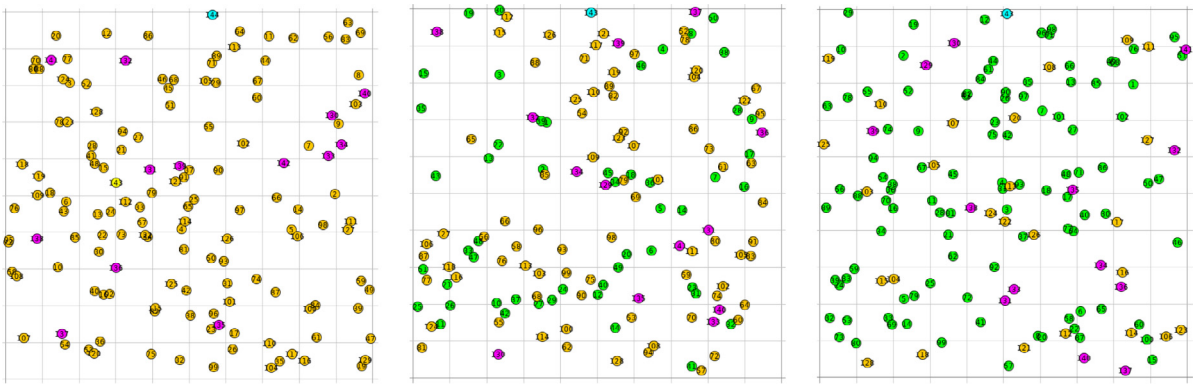
(a) 100% IDS agents (b) 60% IDS agents (c) 20% IDS agents

Fig. 4. Indicative topology of sparse network in WSANs.



(a) 100% IDS agents (b) 60% IDS agents (c) 20% IDS agents

Fig. 5. Indicative topology of normal density network in WSANs.



(a) 100% IDS agents (b) 60% IDS agents (c) 20% IDS agents

Fig. 6. Indicative topology of dense network in WSANs.

$$P(mW) = \frac{Emj}{Time(s)} \tag{4}$$

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

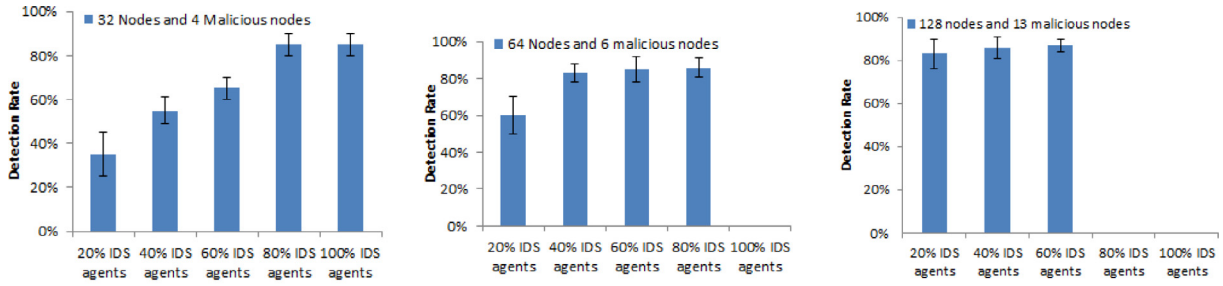
5.2. Evaluation metrics

In the following subsections we present the metrics adopted for the evaluation of the proposed IDS architecture.

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.

$$\text{Detection rate} = \frac{\text{number of true positive detections}}{\text{total number of malicious nodes}} \tag{5}$$

2. *Communication Overhead.* 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 Eq. (6). We follow the practice of [27] 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



(a) Sparse network (b) Normal density network (c) Dense network

Fig. 7. IDS detection rate against the number of active IDS agents in the network as percentage of the node population. Notice that in normal density and dense networks the overhead induced by the IDS is that high that the network gets disconnected prematurely.

	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

Fig. 8. Tmote Sky operation conditions [9].

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 said nodes with no IDS running in the network. Then,

$$\text{Communication overhead} = \frac{E_{\text{IDS}} - E_{\overline{\text{IDS}}}}{E_{\overline{\text{IDS}}}} \quad (6)$$

3. *Total Energy Consumption in the Network.* We measure the total energy consumption ΔE_{total} (Eq. (7)) 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}. Then,

$$\Delta E_{total} = \Sigma_{i \in n}(E^i_{\text{init}} - E^i_{\text{final}}) \quad (7)$$

5.3. Simulation findings

Fig. 7a shows that in sparse networks the detection rate remains as high as 85% for the scenarios where 100% and 80% of the node population operates as an IDS agent. However, the detection rate drops at 60% for 60% of the population as IDS agents, and 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.

Fig. 7b shows the findings for networks of normal density. We note two points. Firstly, 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, we note that for 100% of nodes as 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.

This could also form the reason why other works in the literature on IoT and WSAN IDS limit their simulation studies in networks with small populations. Fig. 7c further highlights these findings as the simulations failed to complete for scenarios considering big numbers of IDS agents (percentages of 100% and 80%). Also, in dense networks the detection rate of the IDS remained at very high levels (circa 80–85%).

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 7a, 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 80% of the population for sparse networks, %60 of the population for medium dense networks, %20 of the population for dense networks). This implies that only *a constant number of IDS agents* is needed to effectively and efficiently monitor the network.

This is a very strong indication that the efficiency of 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 Random Geometric Graphs, we conjecture 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.

Figs. 9 and 10 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 the massive gains that can be achieved with respect to 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.

In the second part of our experiments we initially dropped off gradually and randomly some IDS agent nodes during the simulation to study the resilience and how that affects the efficiency of our approach. We then evaluated the proposed IDS algorithm

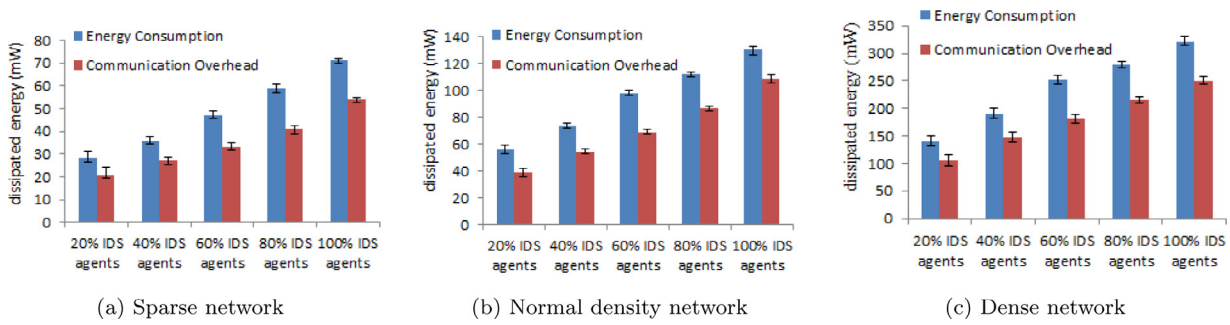


Fig. 9. Energy consumption and communication overhead for the entire network.

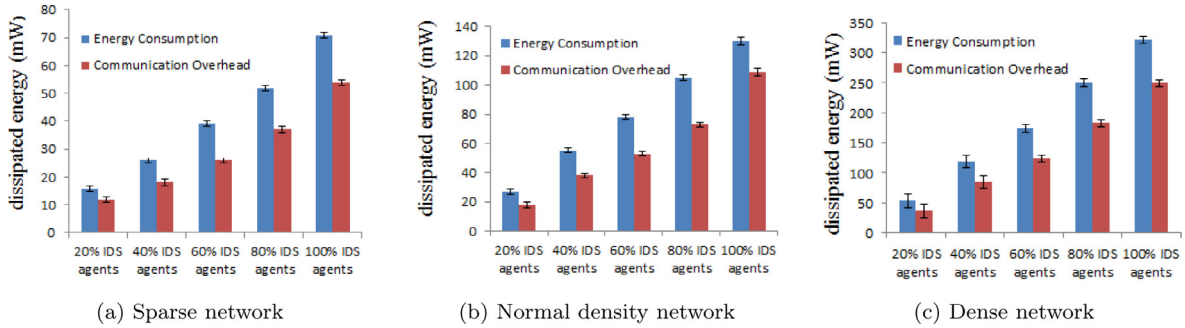


Fig. 10. Energy consumption and communication overhead introduced by the IDS .

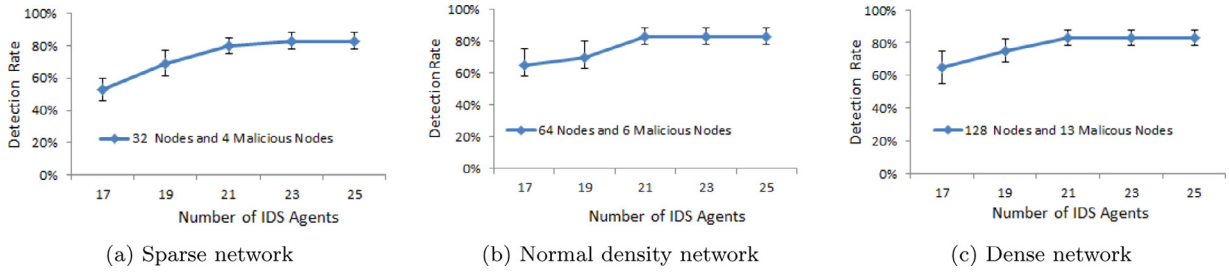


Fig. 11. Detection rate over IDS agent nodes failures.

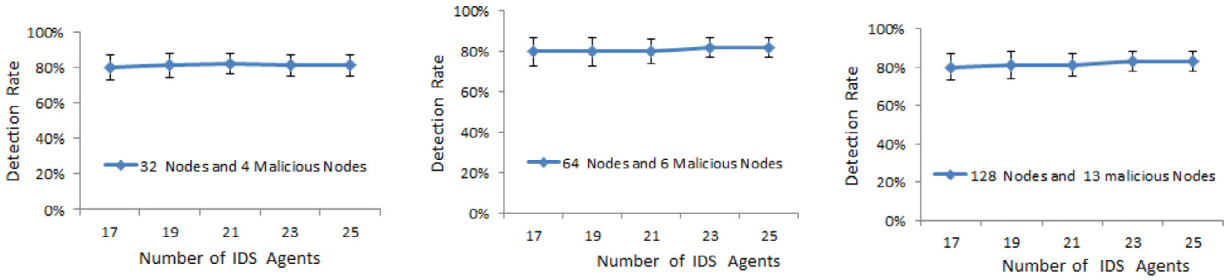
deployed at the monitoring agents against node failures. Fig. 11 show for the varying network densities the detection rates remain systematically high for all scenarios of the different IDS agent numbers (25,23,21). That is, even if the number of IDS agents drops slightly below the connectivity threshold (which is equal to 25 in this case), the effectiveness and performance of the IDS is not significantly affected. For even lower numbers (19–17 IDS agents) the detection rates would drop gradually and gracefully affecting only areas of the networks left un-monitored. This indicates that the IDS agents placement based on the threshold connectivity provides a level of resilience against nodes failure, but it is not guaranteed with more node failures. It is therefore proposed to introduce and include lightweight processes and algorithms to monitor IDS agents against nodes failure. Fig. 12 shows that the detection rate after activating the resilience mechanism integrated with the proposed architecture as shown in algorithm 2 remains high even after the nodes dropping. This indicates that our approach provides a level of resilience against nodes failures or whenever nodes fail to communicate with the central IDS.

Fig. 13 shows the energy consumption and the communication overhead introduced into the network by our proposed algorithm that monitors the distributed IDS agents against nodes failure. The energy consumption is not affected after random IDS agents nodes dropping due to that centralised IDS reallocate the distributed IDS agents in order to recover the area that left un-monitored.

This indicates that the energy efficiency 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.

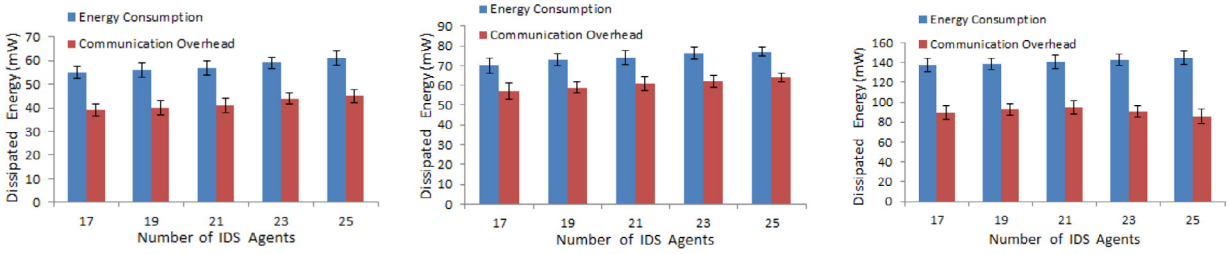
6. Future work

In this work detection accuracy was outside the scope of the research. We investigates the underline cause of the communication and energy consumption overhead introduce to constrained networks by an IDS. In particular, study the communication overheads introduced by hybrid IDS and seek to 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



(a) Sparse network (b) Normal density network (c) Dense network

Fig. 12. Detection rate after IDS agents failure recovery.



(a) Sparse network (b) Normal density network (c) Dense network

Fig. 13. Energy consumption and communication overhead introduced by the IDS with recovery mechanism.

should be conducted to enhance the accuracy of detection algorithm 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 as specified by the 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 protocols.

The IDS architecture for WSNs proposed in this work can be extended and consider mobile Wireless Networks. In the proposed approach we considered static 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 research should be conducted to investigate the novel findings of this work on Mobile Wireless Sensor Networks (MWSNs) that are used recently in critical settings and real-world applications. MWSN 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, MWSN are characterised by their mobility feature that adds more challenging on Wireless Networks security solutions. Therefore, future research should be conducted to study standardised routing protocols of mobile Wireless Networks.

Furthermore, we intend to employ rigorous graph-theoretical methods and tools to formally prove proposed approach for real-world networks. We will also work in providing efficient algorithms for choosing in a distributed way which nodes should operate as IDS agents as well as balancing this role among all the nodes.

7. Conclusions

In this work we study efficient and lightweight Intrusion Detection Systems for static ad-hoc networks via the prism of IPv6-enabled Wireless Actuator Sensor Networks. We first provide a formal model for WSNs with the use of Random Geometric Graphs, a graph-theoretical model to capture the spatial characteristics of WSNs such as inter-dependencies on the existence of wireless links among neighbouring nodes. Then, motivated from the operation of IoT-specific networking protocols such as RPL, we focus on network attacks such as the sinkhole or man-in-the-middle. We identify and try to optimise the trade-off between energy efficiency and communication overhead on one hand, and the IDS detection rate on the other. By levering upon the distributed nature of such protocols and locally available network information.

We extend the state of the art on 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 via the RGG model) and 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 monitors the distributed IDS agents and reallocates a new subset to run as IDS agents whenever node failure accrue.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This work has received funding from the EU's Horizon 2020 research and innovation programme under the Marie SkÅowska-Curie grant agreement No 778229 (Ideal-Cities) and under the grant agreement No. 830943 (ECHO).

References

[1] Al Qurashi M, Angelopoulos CM, Katos V. Efficient intrusion detection in ad-hoc networks. In: 6th international symposium for ICS & SCADA cyber security research 2019 6; 2019. p. 117–25.
[2] Angelopoulos CM, Nikoletseas S, Patroump D, Rapropoulos C. A new random walk for efficient data collection in sensor networks. In: Proceedings of the 9th ACM international symposium on Mobility management and wireless access; 2011. p. 53–60.
[3] Cervantes C, Poplade D, Nogueira M, Santos A. 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; 2015. p. 606–11.
[4] Coppolino L, D'Antonio S, Garofalo A, Romano L. 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; 2013. p. 247–54.
[5] Dujovne D, Watteyne T, Vilajosana X, Thubert P. 6Tisch: deterministic ip-enabled industrial internet (of things). IEEE Commun Mag 2014;52(12):36–41.
[6] Dunkels A, Gronvall B, Voigt T. Contiki-a lightweight and flexible operating system for tiny networked sensors. In: Local computer networks, 2004. 29th annual IEEE international conference on. IEEE; 2004. p. 455–62.
[7] Gupta P, Kumar P. Critical power for asymptotic connectivity in wireless networks. Boston: Stochastic Analysis, Control, Optimization and Applications; 1998.
[8] Ioannou C, Vassiliou V. An intrusion detection system for constrained WSN and IoT nodes based on binary logistic regression. In: Proceedings of the 21st ACM international conference on modeling, analysis and simulation of wireless and mobile systems; 2018. p. 259–63.
[9] Janbu y.. Tmotesky tmote sky block diagram cc2420 data sheet sentilla. 2020. URL https://fccid.io/TOQTMOTESKY/Block-Diagram/Block-Diagram-613122.
[10] Jun C, Chi C. Design of complex event-processing ids in internet of things. In: 2014 sixth international conference on measuring technology and mechatronics automation. IEEE; 2014. p. 226–9.
[11] Kolias C, Kambourakis G, Stavrou A, Voas J. Ddos in the iot: mirai and other botnets. Computer 2017;50(7):80–4.
[12] Kumarage H, Khalil I, Tari Z, Zomaya A. Distributed anomaly detection for industrial wireless sensor networks based on fuzzy data modelling. J Parallel Distrib Comput 2013;73(6):790–806.
[13] Le A, Loo J, Chai K, Aiash M. A specification-based ids for detecting attacks on RPL-based network topology. Information 2016;7(2):25.
[14] Li S, Da Xu L, Zhao S. The internet of things: a survey. Inf. Syst. Front. 2015;17(2):243–59.
[15] Li W, Tug S, Meng W, Wang Y. Designing collaborative blockchained signature-based intrusion detection in IoT environments. Future Gener Comput Syst 2019;96:481–9.
[16] Liu S.. IoT market size worldwide 2017–2025. 2020. URL https://www.statista.com/statistics/976313/global-iot-market-size/.
[17] Maleh Y, Ezzati A, Qasmaoui Y, Mbida M. A global hybrid intrusion detection system for wireless sensor networks. Procedia Comput Sci 2015;52:1047–52.
[18] Midi D, Rullo A, Mudgerikar A, Bertino E. Kalisa system for knowledge-driven adaptable intrusion detection for the internet of things. In: Distributed computing systems (ICDCS), 2017 IEEE 37th international conference on. IEEE; 2017. p. 656–66.
[19] Moon SY, Kim JW, Cho TH. 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; 2014. p. 467–70.
[20] Oh D, Kim D, Ro W. A malicious pattern detection engine for embedded security systems in the internet of things. Sensors 2014;14(12):24188–211.
[21] Osterlind F, Dunkels A, Eriksson J, Finne N, Voigt T. Cross-level sensor network simulation with COOJA. In: Local computer networks, proceedings 2006 31st IEEE conference on. IEEE; 2006. p. 641–8.
[22] Penrose M. Random geometric graphs. Oxford University Press; 2003.
[23] Pongle P, Chavan G. Real time intrusion and wormhole attack detection in internet of things. Int J Comput Appl 2015;121(9).
[24] Ponomarchuk Y, Seo D-W. Intrusion detection based on traffic analysis in wireless sensor networks. In: Wireless and optical communications conference (WOCC), 2010 19th annual. IEEE; 2010. p. 1–7.
[25] Raptis TP, Formica A, Pagani E, Passarella A. On the performance of data distribution methods for wireless industrial networks. In: 20th IEEE international symposium on "A World of Wireless, Mobile and Multimedia Networks", WoWMoM 2019, Washington, DC, USA, June 10–12, 2019. IEEE; 2019. p. 1–6. doi:10.1109/WoWMoM.2019.8793020.
[26] Raptis TP, Passarella A, Conti M. Data management in industry 4.0: state of the art and open challenges. IEEE Access 2019;7:97052–93. doi:10.1109/ACCESS.2019.2929296.
[27] Raza S, Wallgren L, Voigt T. Svelte: real-time intrusion detection in the internet of things. Ad Hoc Netw 2013;11(8):2661–74.
[28] Ring T. Connected cars–the next targe TFOR hackers. Netw Secur 2015;2015(11):11–16.
[29] Shelby Z, Hartke K, Bormann C. The constrained application protocol (COAP). Tech. Rep.; 2014.
[30] Shelby Z, Bormann C. 6LoWPAN: the wireless embedded internet, 43. John Wiley & Sons; 2011.
[31] Shreenivas D, Raza S, Voigt T. Intrusion detection in the RPL-connected 6lowpan networks. In: Proceedings of the 3rd ACM international workshop on IoT privacy, trust, and security. ACM; 2017. p. 31–8.
[32] Wang S-S, Yan K-Q, Wang S-C, Liu C-W. An integrated intrusion detection system for cluster-based wireless sensor networks. Expert Syst Appl 2011;38(12):15234–43.
[33] Xie M, Hu J, Guo S, Zomaya AY. Distributed segment-based anomaly detection with kullback-leibler divergence in wireless sensor networks. IEEE Trans Inf Forensics Secur 2017;12(1):101–10.
[34] Zheng Z, Liu A, Cai LX, Chen Z, Shen XS. Energy and memory efficient clone detection in wireless sensor networks. IEEE Trans Mob Comput 2016;15(5):1130–43.