(Online) ISSN 2744-1741
Defense and Security Studies *Original Research*
Vol. 6, No. 1, 2025, pp.25-38
https://doi.org/10.37868/dss.v6.id278

Review of Wireless Sensor Network security

Jusuf Elfarahati[1*], Tarik Pašić[1], Emir Kurtanović[1], Admir Ferhatović[1], Ermin Šabotić[1], Ali Abd Almisreb*[1],
[1] International University of Sarajevo, Bosnia and Herzegovina

*Corresponding author E-mail: aalmisreb@ius.edu.ba

Received: Jan. 3, 2025
Revised: Feb. 5, 2025
Accepted: Feb. 7, 2025
Online: Feb. 13, 2025

Abstract

This review explores the security challenges and solutions associated with Wireless Sensor Networks (WSNs). It provides a comprehensive analysis of common security threats, including various types of attacks and their potential impacts on WSNs. The paper discusses critical security requirements such as data integrity, authentication, and secure communication, and examines defensive measures to mitigate these threats. Additionally, the review addresses trust management within WSNs, highlighting the importance of establishing reliable and secure networks. Emerging trends such as the integration of artificial intelligence, blockchain technology, quantum cryptography, edge computing, and secure multi-party computation are also discussed, showcasing the latest advancements aimed at enhancing WSN security. The findings underscore the necessity for ongoing research and development to improve the security and efficiency of WSNs, ensuring their viability for diverse applications.

Keywords: Wireless security; communication; Data integrity; Authentication

1. Introduction

Alongside the development of wireless networking comes the development of sensor nodes. By being small in size and low in price, sensor node networks represent an improvement over traditional sensors since they come with embedded processors and can exchange data with each other. They find themselves useful in many fields of human interest from civilian to military tasks and are becoming more and more popular. Throughout this paper, we will try to present the advantages and disadvantages of wireless network sensors, explain their function and usability together with ways to make the network more secure and able to defend against outside attacks.

Low cost and versatility are the main characteristics of wireless sensor networks. They manage to solve a great deal of problems that require monitoring, recording, and organization of collected data into one centralized node. With the advancement of self-learning and self-maintainable systems, they also aim to become self-organized through algorithms and network protocols, which represent the core concepts of these networks [1].

The paradigm of wireless network sensors sounds very futuristic and innovative on paper, but in reality, it is extremely hard to accomplish and connect it all into a meaningful system. Wireless network systems bring with

them a great deal of security risks and are far from ideal systems. They lack power and data storage which are crucial to the traditional techniques that are used for security. In these disadvantages we see similarities with very old machines (low processing power) and the trend of cost reduction will do little to none to help solve the problems above, while on the other hand, the increasing cost would solve various issues that would render wireless networks useless from a business standpoint. Another fatal security flow is communication channels which are often unreliable and operations that are ignored by the system and don't contribute it, but still waste precious processing time [2].

Engineers and scientists both are addressing issues from which these networks suffer [3]. Their goal is the maximization of processing capabilities and reduction of power consumption together with the implementation of security systems. Many different approaches to wireless sensor networking are being tested, some of which are: secure and efficient routing, data aggregation, group formation, and many more [2].

Up until recently, many researchers who were pioneering in the field of wireless networking sensors took for granted that all of the nodes that the system consists of will just agree to work with each other and that every node is trustworthy. This proved to be another security flaw of these networking systems and in order to solve these problems researchers had to work on the development of a trust model.

As we can clearly see from this introduction wireless network systems come with many problems, but are worth investing in. They can solve many problems at lower cost and more efficiency, but they require a great deal of investment and development. In the following paper, we will discuss obstacles to security, requirements of a secure network, attacks, and defensive systems of wireless network sensors.

Our literature was chosen from many different academic sources. All of it revolves around the usage of wireless sensor networks. Alongside many book segments and articles, we have chosen a number of different surveys that have the role of confirming importance of the wireless sensor networks. For our paper, we used [1] as our main guideline since the topic at hand covers a vast area of human expertise and we found that [1] comprises many different papers, books, and surveys into a meaningful whole which helped us great deal with making of this paper

2. Discussion

2.1. Obstacles to security

The system of wireless network sensors shares many similarities with older computer systems. Old and new usually don't fare well together and that is the case with these networks. Trying to overcome the liabilities of a network and combine it with new technologies regarding security and resource management is difficult, but before trying to solve this problem, the first step should be understanding it.

In order to provide authentication and data protection nodes must be aware of each other. They must establish communication between them and one of the solutions for the stated problem is public keys. Since this part is about problem understanding we should focus more on the problem statement and constraints of the system.

2.1.1. Ignored or unattended nodes

When nodes of the system are left unattended they serve no purpose to the system and they can easily become liability. Coming up with ways to deal with unattended nodes is extremely important, because we may face system failures because we are requesting some resource that we didn't know wasn't available. There are several scenarios in which a node can be left unattended:

- Physical damage – nodes are often faced with the danger of being physically damaged since they can be placed in various locations that are far away from the central node. This is not a liability with computer networks, because usually PCs are placed on secure locations and only worry about software attacks [4] [1].
- Remote Management – in various scenarios nodes might be unavailable for maintenance. In such scenarios, if they suffer physical damage it is impossible to fix it. We must let go of the node. One of

such example would be a node running out of battery while deep underwater where it cannot be reached, or suffering physical damage behind enemy lines [5].

- Decentralized architecture – The wireless sensor network should be a centralized system of connected nodes. They all should be connected to one central node. If that is not the case and nodes are managed in a decentralized fashion then the difficulty of organization, inefficiency, and unreliability of network rises [4].

We can conclude that we must find a way to handle unattended nodes. Losing one node would not cost us financially, but if left ignored then new possibilities are opened for malicious attacks and we risk destabilization and security of the whole system.

2.1.2. Unreliable communication

Communication between nodes is the most important feature of wireless sensor networks. It is strictly defined with network protocols. Deciding which protocol should be used sets quality and security standards and capabilities of the system. Without reliable communication systems will be easily overloaded with incomplete data which serves no purpose. Unreliable communication is usually caused by the following:

- Incorrect protocol – using the wrong protocol can prove devastating. In wired network systems dividing data into packets and sending them separately proved to be extremely reliable, but when it comes to wireless sensors this transfer type becomes connectionless and unreliable. With connectionless packet transfer, we risk a high percentage of damaged or missing packages. Furthermore, a protocol that we want to use must have extremely good error handling since we expect a higher rate of error with connectionless data transfer [6].
- Density – The density of the network affects communication between nodes. Even with reliable channels if packets meet while traveling their collision will create disturbance and communication will fail. This happens due to the broadcast nature of wireless sensors.
- Latency – many factors can affect a difference in latency between nodes. Different computation times and network congestion are examples of causes for the increase in latency. It becomes difficult to achieve synchronization of the system if latency stars vary. This becomes a huge problem in real-time systems [1].

2.1.3. Limited resources

We are long past the time when our PCs could run out of resources when it comes to coding. If we go through some older codes it becomes clear that they didn't care for their variable names. They usually make no sense and are represented with only a single letter. There is an actual reason that those variable names are so short and that is because there wasn't enough memory to store larger strings. Modern PCs allow us to write much more meaningful and beautiful codes, but wireless network sensors suffer from the same problem as older computers. They lack memory and because they are wireless they lack sufficient energy supply:

- Memory – sensors are physically small devices and more memory requires more space. Every device comes with a small amount of memory and storage space for code. We can take for example one of the common sensor types, TesloB which has: a 16-bit, 8 MHz RISC CPU; 10K RAM; 48K program memory, and 1024K flash storage. Memory size dictates the size of software code, which we can easily assume is not much. When it comes to security in the case of TesloB [1], we don't have much space for the code and must be very careful and smart not to overload it with unnecessary code.
- Energy supply – same as memory, energy supply is limited with the size of the device. Node lives while its battery has power in it. If we make an assumption that it is hard to replace batteries due to high maintenance cost, then ensuring long battery life becomes one of the most crucial tasks to deal with. Power capacity of the node is so small that we must consider the impact of security code that we add to the node. Functions that security systems usually use like encryption, decryption, signature verification,

and similar, can cost these small devices lots of power. Adding more security layers demands more power [2].

Wireless sensor networks come with many obstacles to secure communication. In the discussion following this chapter, we will consider different solutions to the given problems.

2.2. Security requirements

A sensor network is a special kind of system. It imparts a few shared traits to a run of a typical computer network, yet in addition, presents special prerequisites of its own. Along these lines, we can think about the necessities of a wireless sensor organization as including both the run-of-the-mill typical prerequisites and the unique prerequisites suited exclusively to wireless sensor networks.

There are many aspects of security requirements when it comes to wireless sensor network security. Some of them are: data privacy, data integrity, data freshness, availability, self-organization, time synchronization, secure localization, and authentication. Of course, there are more than these mentioned aspects, but those mentioned will be explained in detail.

2.2.1. Data privacy

Information privacy is the most vital issue in network security. Each network with any security focus will commonly address this issue first. In sensor networks, privacy is identified with the accompanying [7]:

- A sensor network ought not to spill sensor readings to its neighbors (in military applications, data can be very sensitive).
- It is critical to construct a protected path in a wireless sensor network since in many applications, very sensitive data is being processed.
- Public sensor data (sensor personalities and open keys) ought to likewise be scrambled to some degree to ensure against traffic examination assaults.
- The standard methodology for keeping sensitive information safe is to protect the information with a secret key that just proposed receivers have, in this way accomplishing privacy.

2.2.2. Data integrity

With the execution of privacy, an enemy might be not able to take data. Be that as it may, this doesn't mean the information is sheltered. The adversary can change the information, in order to send the sensor arrange into confusion. For instance, a malevolent node may include a few parts or control the information inside a packet. This new packet would then be able to be sent to the first receiver. Information loss or harm can even happen without the presence of a malignant node because of the cruel communication condition. In this way, information integrity guarantees that any obtained information has not been adjusted in travel.

2.2.3. Data freshness

Regardless of whether privacy and information integrity are guaranteed, we additionally need to guarantee the freshness of each message. Casually, information freshness recommends that the information is recent, and it guarantees that no old messages have been replayed. This necessity is particularly critical when there are shared-key systems utilized in the structure. Ordinarily, shared keys should be changed after some time. Be that as it may, it sets aside time for new shared keys to be spread to the whole system. In this situation, it is simple for the adversary to utilize a replay attack. Likewise, it is anything but difficult to disturb the typical work of the sensor, if the sensor is unconscious of the new key change time. To tackle this issue a nonce, or some other time-related counter, can be added into the packet to guarantee information freshness.

2.2.4. Availability

Altering the conventional encryption algorithms to fit inside the wireless sensor network isn't free, and will present some additional expenses. A few methodologies adjust the code to reuse however much code as could

be expected. A few methodologies attempt to make utilization of extra communication to accomplish a similar objective. In addition, a few methodologies drive strict restrictions on information access or propose an unsatisfactory plan (central point scheme) so as to rearrange the algorithm.

But all these approaches weaken the availability of a sensor and sensor network because additional computation consumes additional energy (no more energy exists means data will no longer be available), additional communication also consumes more energy, and a single point failure will be introduced if using the central point scheme.

The prerequisite of security influences the task of the network, as well as is profoundly critical in keeping up the accessibility of the entire network.

2.2.5. Time synchronization

Most sensor network applications depend on some type of time synchronization. So as to conserve power, an individual sensor's radio might be killed for time-frames. Moreover, sensors may wish to process the end-to-end postponement of a packet as it goes between two pairwise sensors. An increasingly cooperative sensor network may require group synchronization for following applications, and so forth. What is proposed is a lot of secure synchronization protocols for sender-receiver (pairwise), multihop sender-recipient (for use when the combined hubs are not inside single-hop range), and group synchronization [8].

2.2.6. Self-organization

A wireless sensor network is a commonly specially appointed system, which requires each sensor node to be autonomous and sufficiently adaptable to act self-organizing and self-mending as indicated by various circumstances. There is no settled infrastructure accessible with the end goal of network management in a sensor network. This intrinsic element conveys an incredible challenge to wireless sensor network security too. For instance, the elements of the entire network hinder the possibility of pre-establishment of a shared key between the base station and all sensors [9]. A few random key predistribution plans have been proposed with regard to symmetric encryption methods [10].

With regards to applying public key cryptography procedures in sensor networks, a productive component for public key distribution is essential too. Similarly, disseminated sensor networks must self-organize to help multihop directing, they should likewise self-organize to lead key administration and build trust connection among sensors. In the event that self-organization is inadequate in a sensor network, the harm coming about because of an assault or even a risky condition might be devastating.

2.2.7. Secure localization

Frequently, the utility of a sensor network will depend on its capacity to precisely and naturally find every sensor in the network. A sensor network organization intended to find shortcomings will require precise area data so as to pinpoint the area of an error. Shockingly, an attacker can without much of a stretch control non-secured area data by revealing false flag qualities, replaying signals, and so forth.

A well-known technique regarding this issue is Verifiable Multilateration (VM). In multilateration, a device's position is precisely registered from a progression of known reference points. Verified ranging and distance bounding are utilized to guarantee the precise area of a node. Given distance bounding, an assaulting node can just increase its guaranteed distance from a reference point. Notwithstanding, to guarantee location consistency, an assaulting node would likewise need to demonstrate that its distance from another reference point is shorter [11]. For large sensor networks, the SPINE (Secure Positioning for sensor NEtworks) algorithm is used. It is a three-phase algorithm based on verifiable multilateration.

SeRLoc ‘s oddity is its decentralized, range-independent nature. SeRLoc utilizes locators that transmit reference point data. It is expected that the locators are trusted and can't be imperiled. Besides, every locator is expected to know its very own location. A sensor figures its location by tuning in for the reference point data sent by

every locator. The reference points incorporate the locator's area. Utilizing the majority of the beacons that a sensor node detects, a node calculates an approximate location dependent on the coordinates of the locators. Utilizing a larger part vote scheme, the sensor at that point processes an overlapping antenna region. The last computed area is the "center of gravity" of the overlapping antenna region [12].

2.2.8. Authentication

An adversary isn't simply restricted to adjusting the information packet. It can change the entire packet stream by infusing extra packets. So the recipient needs to guarantee that the information utilized in any decision-making process starts from the correct source. Then again, while building the sensor network, authentication is vital for some administrative assignments. From the above-mentioned, we can see that message authentication is critical for some applications in sensor networks. Casually, information authentication enables a recipient to confirm that the information truly is sent by the guaranteed sender. On account of two-party communication, information authentication can be accomplished through an absolutely symmetric system: the sender and the receiver share a secret key to compute the message authentication code (MAC) of all conveyed information.

An enhancement is proposed to the μTESLA framework that utilizes broadcasting of the key chain responsibilities as opposed to μTESLA's unicasting system [13]. They present a series of plans beginning with a straightforward pre-determination of key chains and lastly settling on a multi-level key chain method. The multi-level key chain scheme utilizes pre-determination and broadcasting to accomplish an adaptable key distribution system that is intended to be impervious to refusal of service assaults, including jamming.

2.3. Attacks

Of course, we cannot talk about the security of networks and not mention the most common types of attacks used against these wireless-based connections. There are many ways in which an attacker may try to harm the effectiveness of the connection or to try to expose the packets that are sent between the nodes. When talking about this topic we must keep in mind that these networks do not and cannot have state-of-the-art security mostly because of two reasons. The first of them is that the nodes inside this network do not have that much computational power, which handicaps them from using heavy computational algorithms that provide more security. The second reason is that the nodes in these networks are usually exposed and reachable, their locations are usually known, for example inside of a traffic light if the network in question is used for traffic management or traffic information collection. In this part, we will explain the most used attacks on these networks. [14].

2.3.1. Denial of service attacks

The first type of attack that we will address is denial of service attacks. These attacks are based on jamming a node or set of nodes. The jamming of a network can come in two forms: constant or intermittent jamming. Constant jamming means the complete jamming of the entire network. No messages can be sent or received. However, if the jamming is only partial or intermittent, then nodes can exchange messages periodically, but not consistently.

These attacks can be used on the link layer, routing layer, and even the transport layer. Link layer attacks are based on breaking the protocol of the network (for example IEEE 801.11b) and this will cause constant retransmissions which will deplete the power supply of the node. Routing layer attacks use the fact that these networks use multi-hop to exchange messages and then an attacker can use their node to purposely route messages to wrong nodes. Lastly, transport layer attacks are based on flooding a certain node. What this means is that the attacker sends too many connection requests to a node and the node allocates resources to handle these requests and if too many resources are allocated it is rendered useless [15].

2.3.2. The Sybil attack

The Sybil attack is defined as a “malicious device illegitimately taking on multiple identities” [16]. This attack is known for bypassing redundancy mechanisms of networks, routing algorithms, data aggregation, and fair

resource allocation. The Sybil algorithm utilizes multiple identities to generate false nodes or creates multiple false routes to render communication slow or impossible [17].

2.3.3. Traffic analysis attacks

These attacks are like observations on a certain network so that the attacker can know “where to strike”. These attacks are founded on the principle that these networks usually have a base node that forwards packets more than the other nodes. There are certain algorithms that can define with a high probability of success, which node is a base node, and when the attacker knows this, he can actively try to jam or disable this node to weaken the performance or disable the network completely [18].

2.3.4. Node replication attacks

The description of this type of attack is already given in the name. The attacker tries to imitate a node inside the network and by doing this he may infiltrate the network successfully. This can lead to many problems, for example, corrupted packets, false network readings, misrouting packets, and many more [19].

2.3.5. Physical attacks

As previously stated, nodes in these networks are vulnerable to physical attacks since they are usually in an exposed place. This means that an attacker can use force to open the node and mess with the code and hardware inside. These attacks usually involve forcibly opening the sensor, deleting and then inserting modified code to do false instructions, get information, and more [20].

2.4. Defensive measures

2.4.1. Key establishment

Using keys for encrypting and decrypting data is a common practice in all of Computer Science and related fields. Wireless sensor networks are no exception to that. They are the foundation of securing networks in general and that is why I am going to start the defensive measures with concepts regarding key establishment.

Devices used in wireless sensor networks are of course specific and have specific characteristics of performance, energy, efficiency, and processing power. These are all attributes that need to be taken into consideration when trying to apply a key management technique.

Usually, the most secure and widely used way for key management is public key encryption algorithms. Public Key encryption is an asymmetric type of encryption. It works in the following way. Two keys are generated which have a mathematical relation (the relation depends on the algorithm used). One of the keys is the public key and the other one is the private key. The public key can then safely be distributed to any external clients and these clients can then encrypt data using that key. Data encrypted using the public key can then only be decrypted using a combination of the public and private keys, and the only owner of the private key is the source that gave out the public keys.

There exist various algorithms for the public-private key generation and this method is extremely secure because it would take an unreal amount of processing power to crack the private key. Currently, we lack that processing power. One of the most popular public key encryption cryptosystems is RSA (Rivest-Shamir-Adleman) which is heavily used throughout the web and application network layer. But here comes the issue. As I mentioned earlier, devices in wireless sensor networks are specific in terms of power. To be cheap these devices lack computational power and public key encryption requires a lot of computational power when compared to these devices. That is why we often need to use the simpler, symmetric approach called Shared Key Encryption.

Shared key encryption, as the name implies, works like this. You encrypt data using some encryption algorithm and some key, now to access this data a client needs the same key, so the key needs to be shared, hence the name. And it is symmetric because both sides need the same key. The most used algorithm for symmetric encryption is DES (Data Encryption Standard). One disadvantage is that the key is easier to crack and therefore

it is less safe than other algorithms. But there are of course other algorithms for symmetric encryption, because as I said, they are computationally easier and suited for sensor devices, but still they all have one major problem, and that is that both hosts need the key to establish communication. The obvious problem with this is how to be sure that the two hosts that need to exchange data are legitimate and that no third party tried to be in the middle of it.

Now let us talk about proposed solutions to the mentioned problem.

The first solution is proposed by Eschenaur and Gligor [21]. It is the idea of randomly pre-distributing key rings to nodes in a wireless sensor network [22]. Each keyring is to contain a set of randomly chosen keys from a larger set of keys. Although not a perfect solution, this way it is probabilistically very likely that each node would have another pair node sharing the same key from their respective keyrings and therefore would be able to establish a direct link. The idea is that this would make it possible for full-fledged communication in larger networks.

The next solution is the LEAP protocol. Its idea is to preload the nodes with a key. The nodes then use that key to establish a link generate a new key for communication and then delete the preloaded key.

Another protocol is the PIKE protocol, which introduces the idea of using a trusted third-party node for any two sensor nodes. The sensor nodes that need to communicate do not share the keys directly but share their key with the third-party node.

The last idea for establishing a secure connection between nodes is to have a base station in the wireless sensor network. The base station should have more computational power and be able to do public key cryptography operations. Then, the sensor nodes would communicate with the base station through some symmetric approach, but the base station would take care of public key cryptography for the sensor nodes when they communicate with each other. This is in my opinion the most secure way, but probably the most expensive too.

It is important not to forget that public key encryption is not completely ruled out as a method of key management inside wireless sensor networks. There are cases where so-called ECC – elliptic curve cryptography can be used to secure a wireless sensor network. ECC is an asymmetric algorithm like RSA. The main difference is that ECC generates shorter keys than RSA, 160 compared to 1024 bits, and that is why ECC is more suited for sensor nodes than RSA.

2.4.2. Defense against DoS attacks

DoS (Denial of Service) attack is probably the most common type of attack on any network and on any network layer. In the case of wireless sensor networks, a couple of approaches can be used. Let us start with the simplest one.

The logical approach to a DoS attack would be to find the jammed node and simply route around it [21]. The hard part here is to find out which node is the attacked one. Proposals have been made for an algorithm where the nodes that are near the jammed one, report their status to their neighbors and then declare the jammed node as unavailable and construct new routes around it.

The next approach is to handle the DoS attack at the data-link layer by using MAC addresses. The idea is to impose a rate limit for a sensor node to start ignoring resource-heavy requests. This approach has one issue though, there is no way to tell the difference between malicious intent and high-volume traffic.

Our next issue is malicious nodes that try to miscarry information. To overcome them we can simply send the same message over more routes and hope that at least one of the routes has no malicious node in it. This approach introduces of course redundancy but is simple and efficient.

The last type of DoS attack that we will talk about is the flood attack. Flooding is a transport layer attack where so many requests are sent that the server cannot handle legitimate requests because it is out of resources. Sometimes there is nothing you can do here, but some engineers have made a proposal to force clients to solve

a puzzle given out by the server before entering communication. This puzzle is supposed to be computationally intensive to solve. This way a client would have always to commit resources before the server, thus rendering a DoS attack impossible unless an unreal amount of resources is used. It is an interesting idea, at least. It reminds of how blockchain works, but I personally do not like it because it would slow down the communication between client and server.

2.4.3. Combating traffic analysis attacks

Surely there are ways to defend against traffic analysis attacks. One of the most common combats is using a random walk forwarding technique, which sends packets from time to time to a node that is not the sensor's parent node. But again, this does not cover the time correlation attacks.

For this to be covered Dent et al. suggest a fractal propagation strategy. Using this strategy a node will generate a packet that is fake when its neighbor is forwarding a packet to the base station. This packet which is fake is sent to another neighbour who may also generate a packet that is also fake.

There are several defense techniques with which to combat sensor attacks, they are listed below [23].

2.4.4. Anonymity mechanisms

Information about location which is described in detail can give the opportunity for an intruder to get user details or to track future user movements, this is a big problem and it presents a threat to a user. The mechanism is to depersonalize the data before they are released [24].

There are three approaches:

- Decentralize Sensitive Data – Distribute data of location to a spanning tree.
- Secure Communication Channel – using SPINS protocol the active attacks can be prevented
- Change Data Traffic – Inserting some extra clear data can change the traffic pattern
- Node Mobility – Changing node positions can defend against attacking privacy.

2.4.5. Information flooding

There are many proposed anti-traffic analysis mechanisms to defend against attackers from outside tracking the node locations since that would uncover the information about the location of sensed objects. [25] The methods that are used to defend against that are: the randomized data routing and phantom traffic generation mechanism.

They are used to cover the real data traffic so that it is difficult for the attacker to track the data source by analyzing the traffic of the network. There are many methods for single-path routing which are trying to solve privacy problems. Some of them are: baseline flooding, probabilistic flooding, flooding with fake messages, and phantom flooding.

2.4.6. Intrusion detection

There are many techniques that are used to prevent intruders from viewing the network data. Knowing the importance of detecting intruders, there are still no good solutions for this.

The two main categories of intrusion detection are: anomaly-based intrusion detection (AID), and misuse intrusion detection (MID) [26]. Anomaly-based intrusion detection is a detection that assumes that the intruders will do something ambiguous to the legitimate nodes, so unusual system behavior will be present. While in misuse intrusion detection the whole system has a database of the intrusion signatures which can help the system to easily detect the intrusions.

The intrusion detection in wireless sensor networks is all done with cryptography. Intrusion detection can be divided into two categories: host-based and network-based. Host-based IDS systems are doing the combats related to operating systems audit trails, logs, system call audit trails, and so on. A network-based IDS operates with packets that are captured from the network.

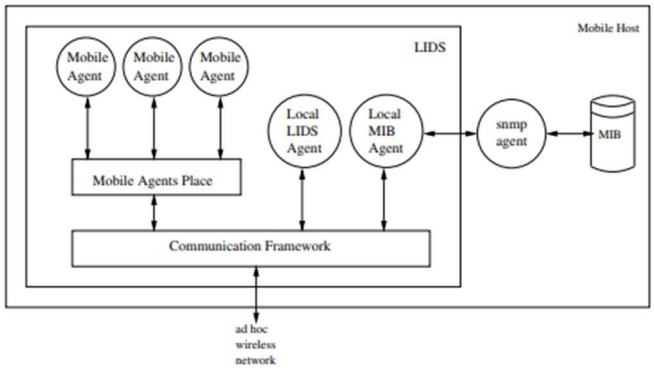

Figure 1. The LIDS architecture [2]

2.4.7. Secure data aggregation

As the amount of data the sensor networks are able to sense, the WSN continues to grow in size.

Because every sensor has constraints that are computational, a single sensor is responsible for a small part of the overall data. So this is the reason why the raw data is being returned from the wireless sensor network. It is the priority for the raw data to firstly be processed in order to have and fetch useful data from the network and it is done using many aggregators. The job of aggregators is to process the raw data into useful data collected from the subsets of nodes. However, this technique is vulnerable to attacks because one single node is just used to process and aggregate the data.

There are many techniques used today in secure data aggregation. The localized algorithm uses the directed diffusion technique using only local nodes. Using this technique the nodes that are at a higher level are able to communicate across clusters, while lower-level nodes are not able to communicate.

We have one problem with the techniques which are the standard ones. The problem is that they assume that the nodes are trustworthy, which is not the case, so for that secure data aggregation techniques will be required.

Stealthy attacks are attacks where an intruder creates fake aggregation data which are represented to the user by providing the incorrect aggregation results without the user spotting it. So the goal is to make sure that the accepted aggregate value from the user has a high probability that the value is near true to one aggregation value. So the user has the option to reject or accept the aggregation value. This approach is called the aggregate-commit-prove technique.

Hu and Evans gave a secure data aggregation technique that uses the (mi)TESLA protocol for security. In the description of their technique, it is stated that the nodes are organized into a tree-based hierarchy where the internal nodes are aggregators [27]. It is not, however, guaranteed in this technique that all nodes and aggregators provide the correct values [23].

2.4.8. Defending against physical attacks

Physical attacks represent a great problem for Wireless Sensor Networks because of their unattended features and limited resources. The physical hardware would help to enhance protection against many attacks from outside. To protect, for example, against tempering with the sensors, one defense involves tamper-proofing the node's physical package.

One possible way of protecting the sensor nodes with the physical hardware is self-termination. As the name tells us, the sensor destroys itself (including destroying all data and keys), when it spots a possible attack. This is specifically used in large wireless sensor networks where the cost is much cheaper for the sensors compared to the loss of being broken (attacked). The solution which seems the simplest one is periodically conducting neighborhood checking in static deployment.

Andersen et al. gave examples based on these attacks with low-cost protection:

- Randomized Clock Signal – Inserting random-time delays
- Randomized Multithreading – multithread processor architecture that schedules the processor by hardware between two or more threads of execution randomly at a per-instruction level.
- Restricted Program Counter – avoiding program counter which can traverse through the whole address space.

2.4.9. Trust management

Trust is very important in any network environment, social networking, or computer networking. Trust can be used in front of traditional cryptographic security.

For example, if we want to know whether the sensor nodes and the quality of their services are of high quality, also whether the aggregator is performing aggregation as it should perform.

Finding of Liang and Shi focus on trust model development and the rating aggregation algorithms analysis in the untrusted environments that are open. Their findings can be applied to Wireless Sensor Networks. They found out that rating is not useful for always knowing the limitations of the other factors [21] [7].

2.5. Emerging trends in Wireless Sensor Network security

In recent years, the field of wireless sensor network (WSN) security has seen significant advancements driven by the increasing complexity and deployment of WSNs in various critical applications. This section explores the latest trends and innovative approaches that are shaping the future of WSN security.

2.5.1. Artificial Intelligence and Machine Learning

Artificial intelligence (AI) and machine learning (ML) have emerged as powerful tools in enhancing the security of WSNs. These technologies are being leveraged to develop intelligent intrusion detection systems (IDS) that can identify and mitigate security threats in real-time. AI and ML algorithms can analyze vast amounts of data generated by sensor nodes to detect anomalies and predict potential attacks, thereby improving the overall resilience of WSNs [28].

2.5.2. Blockchain technology

Blockchain technology is gaining traction as a means to secure WSNs by providing a decentralized and tamper-proof ledger for recording transactions and data exchanges. This technology ensures data integrity and authenticity, making it difficult for attackers to alter or forge information. Blockchain can also facilitate secure key management and authentication processes within WSNs [29].

2.5.3. Quantum cryptography

Quantum cryptography is an emerging field that promises to revolutionize WSN security by leveraging the principles of quantum mechanics. Quantum key distribution (QKD) enables the secure exchange of cryptographic keys, ensuring that any attempt to intercept or tamper with the keys is detectable. This technology offers a high level of security that is theoretically immune to computational attacks [30].

2.5.4. Edge computing

Edge computing involves processing data closer to the source, i.e., at the edge of the network, rather than relying on centralized cloud servers. This approach reduces latency and enhances the security of WSNs by minimizing the exposure of sensitive data to potential attacks during transmission. Edge computing also enables real-time data analysis and decision-making, which is crucial for timely threat detection and response [31].

2.5.5. Secure multi-party computation

Secure multi-party computation (SMPC) is a cryptographic technique that allows multiple parties to jointly compute a function over their inputs while keeping those inputs private. In the context of WSNs, SMPC can be

used to perform secure data aggregation and analysis without revealing individual sensor data. This approach enhances privacy and security, particularly in applications involving sensitive information [32].

These emerging trends highlight the dynamic nature of WSN security and the continuous efforts to address the evolving threats and challenges. By integrating advanced technologies such as AI, blockchain, quantum cryptography, edge computing, and SMPC, researchers and practitioners can develop more robust and resilient WSNs capable of withstanding sophisticated cyber-attacks.

3. Conclusions

This paper provides a comprehensive review of the security challenges and solutions associated with wireless sensor networks (WSNs). It outlines the most common attacks and the defensive measures required to protect WSNs, covering topics such as obstacles to security, security requirements, attacks, defensive measures, and trust management. Additionally, the paper explores emerging trends in WSN security, including the integration of artificial intelligence, blockchain technology, quantum cryptography, edge computing, and secure multi-party computation. These advancements highlight the dynamic nature of WSN security and the continuous efforts to address evolving threats. The findings underscore the necessity for ongoing research and development to enhance the security and efficiency of WSNs, ensuring their viability for diverse applications.

Declaration of competing interest

The authors declare that they have no known financial or non-financial competing interests in any material discussed in this paper.

Funding information

No funding was received from any financial organization to conduct this research.

References

[1] J. P. Walters, Z. Liang, W. Shi, and V. Chaudhary, "Wireless sensor network security: A survey," in *Security in Distributed, Grid, Mobile, and Pervasive Computing*. Boca Raton, FL, USA: Auerbach Publications, 2007.

[2] D. W. Carman, P. S. Kruus, and B. J. Matt, *Constraints and Approaches for Distributed Sensor Network Security*, NAI Labs, Glenwood, MD, USA, Tech. Rep. 00-010, 2000.

[3] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *IEEE Communications Magazine*, vol. 40, no. 8, pp. 102–114, Aug. 2002. https://doi.org/10.1109/MCOM.2002.1024422

[4] A. R. Beresford and F. Stajano, "Location privacy in pervasive computing," *IEEE Pervasive Computing*, vol. 2, no. 1, pp. 46–55, Jan.–Mar. 2003. https://doi.org/10.1109/MPRV.2003.1186725

[5] D. Estrin, R. Govindan, J. Heidemann, and S. Kumar, "Next century challenges: Scalable coordination in sensor networks," in *Proc. 5th Annu. ACM/IEEE Int. Conf. Mobile Computing and Networking (MobiCom '99)*, Seattle, WA, USA, pp. 263–270, 1999. https://doi.org/10.1145/313451.313556

[6] A. Perrig, R. Szewczyk, V. Wen, D. E. Culler, and J. D. Tygar, "SPINS: Security protocols for sensor networks," *Wireless Networks*, vol. 8, nos. 5–6, pp. 521–534, 2002. https://doi.org/10.1023/A:1016598314198

[7] L. Lazos and R. Poovendran, "Secure broadcast in energy-aware wireless sensor networks," DTIC, Tech. Rep., 2002. [Online]. Available: https://apps.dtic.mil/sti/tr/pdf/ADA459885.pdf

[8] S. Ganeriwal, S. Capkun, C.-C. Han, and M. B. Srivastava, "Secure time synchronization service for sensor networks," in *Proc. ACM Workshop Wireless Security (WiSe)*, Cologne, Germany, pp. 97–106, 2005. https://doi.org/10.1145/1080829.1080843

[9] L. Eschenauer and V. D. Gligor, "A key-management scheme for distributed sensor networks," in *Proc. 9th ACM Conf. Computer and Communications Security (CCS '02)*, Washington, DC, USA, pp. 41–47, 2002. https://doi.org/10.1145/586110.586117

[10] H. Chan, A. Perrig, and D. Song, "Random key predistribution schemes for sensor networks," in *Proc. IEEE Symp. Security and Privacy*, Berkeley, CA, USA, pp. 197–213, 2003. https://doi.org/10.1109/SECPRI.2003.1199337

[11] S. Capkun and J.-P. Hubaux, "Secure positioning in wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 2, pp. 221–232, Feb. 2006. https://doi.org/10.1109/JSAC.2005.861387

[12] L. Lazos and R. Poovendran, "SeRLoc: Secure range-independent localization for wireless sensor networks," *ACM Trans. Sensor Networks*, vol. 1, no. 1, pp. 73–100, 2005. https://doi.org/10.1145/1077391.1077398

[13] D. Liu and P. Ning, *Efficient Distribution of Key Chain Commitments for Broadcast Authentication in Distributed Sensor Networks*, North Carolina State Univ., Raleigh, NC, USA, Tech. Rep., 2002.

[14] S. Plosz, A. Farshad, M. Tauber, C. Lesjak, T. Ruprechter, and N. Pereira, "Security vulnerabilities and risks in the industrial usage of wireless communication," in *Proc. IEEE Int. Conf. Emerging Technology and Factory Automation (ETFA)*, Barcelona, Spain, pp. 1–8, 2014. https://doi.org/10.1109/ETFA.2014.7005257

[15] A. D. Wood and J. A. Stankovic, "Denial of service in sensor networks," *Computer*, vol. 35, no. 10, pp. 54–62, Oct. 2002. https://doi.org/10.1109/MC.2002.1039518

[16] J. R. Douceur, "The Sybil attack," in *Proc. Int. Workshop Peer-to-Peer Systems (IPTPS)*, Cambridge, MA, USA, 2002. https://doi.org/10.1007/3-540-45748-8_24

[17] J. Newsome, E. Shi, D. Song, and A. Perrig, "The Sybil attack in sensor networks: Analysis & defenses," in *Proc. 3rd Int. Symp. Information Processing in Sensor Networks (IPSN)*, Berkeley, CA, USA, pp. 259–268, 2004. https://doi.org/10.1145/984622.984660

[18] J. Deng, R. Han, and S. Mishra, "Countermeasures against traffic analysis attacks in wireless sensor networks," in *Proc. 1st Int. Conf. Security and Privacy for Emerging Areas in Communications Networks (SecureComm)*, Athens, Greece, pp. 113–126, 2005. https://doi.org/10.1109/SECURECOMM.2005.38

[19] H. Chan and A. Perrig, "Security and privacy in sensor networks," *Computer*, vol. 36, no. 10, pp. 103–105, Oct. 2003. https://doi.org/10.1109/MC.2003.1236475

[20] S. Mohammadi and H. Jadidoleslamy, "A comparison of physical attacks on wireless sensor networks," *Int. J. Peer-to-Peer Networks*, vol. 2, no. 2, pp. 1–11, Apr. 2011.

[21] Z. Liang and W. Shi, "PET: A personalized trust model with reputation and risk evaluation for P2P resource sharing," in *Proc. 38th Annu. Hawaii Int. Conf. System Sciences (HICSS)*, Big Island, HI, USA, 2005. https://doi.org/10.1109/HICSS.2005.201

[22] F. Armknecht, A. Hessler, J. Girao, A. Sarma, and D. Westhoff, "Security solutions for wireless sensor networks," *NEC Technical Journal*, vol. 1, no. 3, 2006.

[23] Y. Okazaki, I. Sato, and S. Goto, "A new intrusion detection method based on process profiling," in *Proc. IEEE Symp. Applications and the Internet (SAINT)*, pp. 82–90, 2002. https://doi.org/10.1109/SAINT.2002.994460

[24] M. Gruteser and D. Grunwald, “Anonymous usage of location-based services through spatial and temporal cloaking,” in *Proc. 1st Int. Conf. Mobile Systems, Applications and Services (MobiSys ’03)*, San Francisco, CA, USA, pp. 31–42, 2003. https://doi.org/10.1145/1066116.1066120

[25] C. Ozturk, Y. Zhang, W. Trappe, and M. Ott, “Source-location privacy for networks of energy-constrained sensors,” in *Proc. IEEE Workshop Software Technologies for Future Embedded and Ubiquitous Systems*, Vienna, Austria, pp. 68–72, 2004. https://doi.org/10.1109/SEUS.2004.1369748

[26] J. Deng, R. Han, and S. Mishra, “INSENS: Intrusion-tolerant routing for wireless sensor networks,” *Computer Communications*, vol. 29, no. 2, pp. 216–230, 2006. https://doi.org/10.1016/j.comcom.2005.06.010

[27] N. Gura, A. Patel, A. Wander, H. Eberle, and S. C. Shantz, “Comparing elliptic curve cryptography and RSA on 8-bit CPUs,” in *Lecture Notes in Computer Science*, vol. 3156, pp. 119–132, 2004. https://doi.org/10.1007/978-3-540-28632-5_9

[28] S. Suhag and A. Sehwag, “Challenges and potential approaches in wireless sensor network security,” *Journal of Electrical Engineering & Technology*, vol. 19, no. 4, pp. 2693–2700, 2024. https://doi.org/10.1007/s42835-024-01621-6

[29] M. H. Behiry and M. Aly, “Cyberattack detection in wireless sensor networks using a hybrid feature reduction technique with AI and machine learning methods,” *Journal of Big Data*, vol. 11, no. 1, 2024. https://doi.org/10.1186/s40537-024-00887-9

[30] G. Mohan Ram and E. Ilavarasan, “Security challenges in wireless sensor networks: Current status and future trends,” *Wireless Personal Communications*, vol. 139, pp. 1173–1202, 2024. https://doi.org/10.1007/s11277-024-11206-0

[31] Y. Ghadi, T. Mazhar, T. Al Shloul, T. Shahzad, U. A. Salaria, A. Ahmed, and H. Hamam, “Machine learning solution for the security of wireless sensor networks,” *IEEE Access*, vol. 12, pp. 12699–12719, 2024. https://doi.org/10.1109/ACCESS.2024.3360485

[32] A. Akinsola, T. K. Njoku, O. Ejiofor, and A. Akinde, “Enhancing data privacy in wireless sensor networks: Investigating techniques and protocols to protect privacy of data transmitted over wireless sensor networks in critical applications of healthcare and national security,” *Int. J. Network Security & Its Applications*, vol. 16, no. 2, 2024.