Journal of Network and Computer Applications 190 (2021) 103118

Contents lists available at ScienceDirect

Journal of Network and Computer Applications

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

Review

Secure data aggregation methods and countermeasures against various attacks in wireless sensor networks: A comprehensive review

Mohammad Sadegh Yousefpoor [a], Efat Yousefpoor [a], Hamid Barati [a], Ali Barati [a], Ali Movaghar [b], Mehdi Hosseinzadeh [c,*]

[a] Department of Computer Engineering, Dezful Branch, Islamic Azad University, Dezful, Iran
[b] Department of Computer Engineering, Sharif University of Technology, Iran
[c] Pattern Recognition and Machine Learning Lab, Gachon University, 1342 Seongnamdaero, Sujeonggu, Seongnam 13120, Republic of Korea

ARTICLE INFO

Keywords:
Wireless sensor networks (WSNs)
Data aggregation
Security
Attacks
Internet of Things (ioT)

ABSTRACT

Wireless sensor networks include a large number of sensor nodes, which monitor an environment. These networks have many applications in Internet of Things (IoT) and Industrial IoT (IIoT). In wireless sensor networks, data aggregation methods are known as a suitable solution that can reduce energy consumption. In addition, these networks are subject to many attacks due to their wireless communications. Therefore, it is very important to provide data security in the data aggregation process. In this paper, we introduce some secure data aggregation schemes in wireless sensor networks and express their strengths and weaknesses. Moreover, we categorize secure data aggregation methods based on network model, network topology, key cryptography technique, encryption method, application, authentication mechanism, and data recovery ability. We believe that this taxonomy can help researchers to design secure and efficient data aggregation methods in wireless sensor networks, identify problems in existing methods, and solve them. Also, familiarity with new techniques and challenges in this field helps researchers to identify future research directions.

1. Introduction

In the last century, one of the most important technologies is wireless sensor networks (WSNs) (Fahmy, 2020; Fei et al., 2016). These networks have a large number of sensor nodes that collaborate with each other to transmit their data to the sink node. Sensor nodes are tasked to sense and collect data from environment. These nodes are powered by a small battery, which cannot be recharged in most cases (Yetgin et al., 2017; Sah and Amgoth, 2020). These networks are applied in various fields such as environmental monitoring (Muduli et al., 2018), forest fire detection (Aslan et al., 2012), industrial monitoring and control (Salvadori et al., 2009), military applications (Bekmezci, 2009), and civilian applications (Kandris et al., 2020; Rani et al., 2020).

Today, wireless sensor networks play a key role in the Internet of Things (Kouicem et al., 2018). IoT is a new technology in which everything has a digital identifier (Dehkordi et al., 2020). This technology connects the digital world to the real world. In fact, IoT is a new model of WSN. In the IoT, sensor nodes can collect useful information such as location, light, temperature, etc. In fact, WSNs complement our environment knowledge and act as a bridge between the physical world and the digital world (Rani et al., 2020; Hassija et al., 2019). IoT has many applications such as transportation and logistics (Rani et al., 2020), health care (Dhanvijay and Patil, 2019), smart environment (Liyanage et al., 2020a). Fig. 1 displays some IoT applications. Furthermore, WSNs have many applications in Industrial IoT (IIoT). It is a subset of the Internet of Things. IIoT is known as an industrial revolution (Xu et al., 2020; Khan et al., 2020). It is an intelligent network, which includes the connected and intelligent industrial equipment. IIoT have been developed to enhance production rate and reduce production costs through real-time monitoring, efficient management and control of industrial processes, and scheduling. IIoT uses sensor nodes and smart devices. These sensors are installed on the intelligent factory systems to collect data. As a result, the factory manager can continuously analyze information to make more accurate decisions and increase production and productivity. For example, when the machinery fails, the connected sensors automatically report this problem and send a service request. Therefore, WSNs can help maintain industrial equipment through intelligent monitoring. Fig. 2 depicts

* Corresponding author.
E-mail addresses: ms.yousefpoor@iaud.ac.ir (M.S. Yousefpoor), eyousefpoor@iaud.ac.ir (E. Yousefpoor), hbarati@iaud.ac.ir (H. Barati), abarati@iaud.ac.ir (A. Barati), movaghar@sharif.edu (A. Movaghar), mehdi@gachon.ac.kr (M. Hosseinzadeh).

https://doi.org/10.1016/j.jnca.2021.103118
Received 12 December 2020; Received in revised form 21 May 2021; Accepted 24 May 2021
Available online 11 June 2021
1084-8045/© 2021 Elsevier Ltd. All rights reserved.

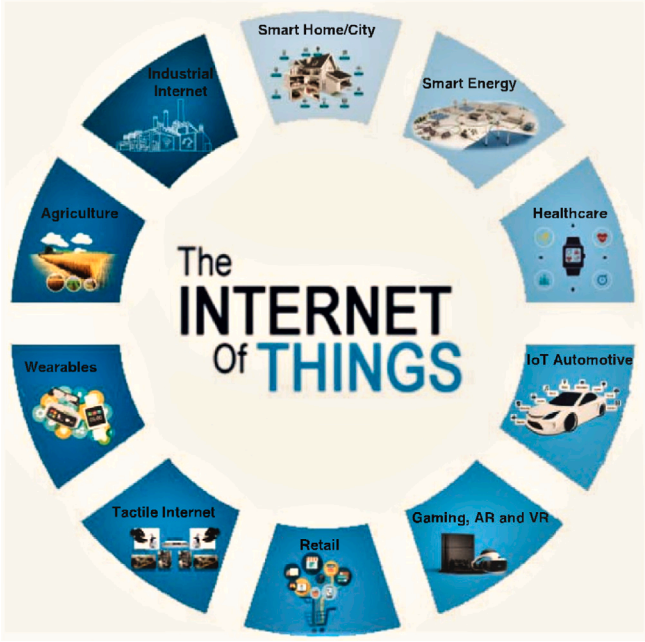

Fig. 1. IoT applications (Liyanage et al., 2020a).

some IIoT applications.

According to the mentioned notes, WSNs may generate a very high volume of data. It is very difficult to process this data using traditional data processing methods. In wireless sensor networks, conventional data collection techniques are not suitable because they flood data in the network and discharge the energy of sensor nodes quickly. Therefore, one of the appropriate solutions is to utilize data aggregation schemes (Randhawa and Jain, 2017; Mehrjoo and Khunjush, 2018). Data aggregation can be defined as merging the information (Cui et al., 2020). In WSNs, data aggregation processes can effectively lower communication overhead and energy consumption.

In addition, critical information is exchanged in WSNs (Dewal et al., 2018). Therefore, security is a very serious issue in WSNs due to their properties and applications such as IoT, IIoT, etc (Ghani et al., 2019; Gharib et al., 2017). Because any security gap may reveal confidential information and cause irreparable damages (Yousefpoor and Barati, 2019, 2020). In the following, some properties of these networks are presented (Dargie and Poellabauer, 2010).

- Sensor nodes include resource constraints (memory, computing power, bandwidth, and communication range) (Khan et al., 2016; Barati et al., 2015).
- WSNs have a dynamic topology. In addition, these networks are developed in inaccessible environments and have not a certain infrastructure. Therefore, it is impossible to control these networks continuously. As a result, they are exposed to many attacks (Hatamian et al., 2016; Belkhira et al., 2019).
- In WSNs, wireless communication channels are utilized to communicate between sensor nodes. As a result, attackers can eavesdrop on data packets exchanged on these channels (Karmaker et al., 2020).

Therefore, it can be deduced that researchers cannot ignore security in the data aggregation process. However, there is a natural contrast between security and data aggregation (Zhu et al., 2017). On the one hand, in security mechanisms, sensor nodes must encrypt their data and send it securely to the base station (BS), so that intermediate nodes cannot access their content. Then, BS receives and decrypts the data packets to access the original data (Bhushan and Sahoo, 2020). On the other hand, in data aggregation methods, aggregator nodes must execute data aggregation operations on raw data to obtain the aggregated data and send it to BS. Therefore, in designing WSN protocols, security and data aggregation must be efficiently combined so that the data aggregation process can be implemented without compromising security. Simultaneous access to security and data aggregation has been led researchers to focus on secure data aggregation (SDA) methods. It is an important challenge to design appropriate SDA methods that befit for wireless sensor networks. In data aggregation mechanisms, security means protecting aggregated data against any unauthorized alterations (Shah and Shukla, 2012; Liu et al., 2019).

Today, some researches have been done in this area because secure data aggregation is a very important concept. However, our studies show that there are few review papers in this area. As a result, it is necessary to do further researches on the SDA field to determine its challenges and future research directions. Table 1 summarizes a number of review papers related to the SDA methods in WSNs. Most review papers focused on the structure of secure data aggregation schemes, i.e. network topology, data aggregation operations, and cryptography techniques. Whereas, the main purpose of this survey is to provide a comprehensive perspective so that researchers can answer the question: *"How can they design appropriate secure data aggregation schemes in WSN?"* In this paper, we provide a detailed classification of SDA methods. Our proposed classification is as follows:

- Network models in SDA schemes (i.e. homogeneous, heterogeneous)
- Network topologies in SDA methods (i.e. hierarchical (cluster-based), flat, tree-based, and tree-cluster based)
- Key cryptography techniques in SDA schemes (i.e. symmetric key cryptography, asymmetric key cryptography, hybrid key cryptography)
- Encryption schemes in SDA methods (i.e. hop-by-hop encryption and end-to-end encryption)
- Types of SDA schemes based on application (i.e. low-risk applications and high-risk applications)
- Authentication mechanisms in SDA methods (i.e. end-to-end authentication and hop-by-hop authentication)
- Types of SDA schemes based on data recovery ability (i.e. recoverable SDA schemes and unrecoverable SDA schemes)

In Table 2, our review paper has been compared with previous review papers.

In the following, our major contributions are summarized:

- First, we define the most important security requirements in WSNs and illustrate the most common attacks in these networks.
- Then, we describe secure data aggregation and its evaluation scales and present our proposed classification.
- Next, we select the state-of-the-art SDA schemes and review them based on our proposed classification and focus on their weaknesses and strengths.
- Finally, we evaluate what security requirements have been addressed by these methods. Then, we introduce their security solution to counteract attacks that threaten these security requirements.

We believe that this review paper helps researchers to realize the security level, strengths, and weaknesses of the SDA methods. Discovering challenges related to secure data aggregation helps scholars to know the future research directions. Therefore, researchers can use this knowledge to address existing problems.

In the following, the paper is organized as follows: Section 2 describes some security requirements and introduces the most common attacks in WSNs. In Section 3, the data aggregation process is defined. In this section, we focus particularly on secure data aggregation methods, and their evaluation scales and present our proposed classification.

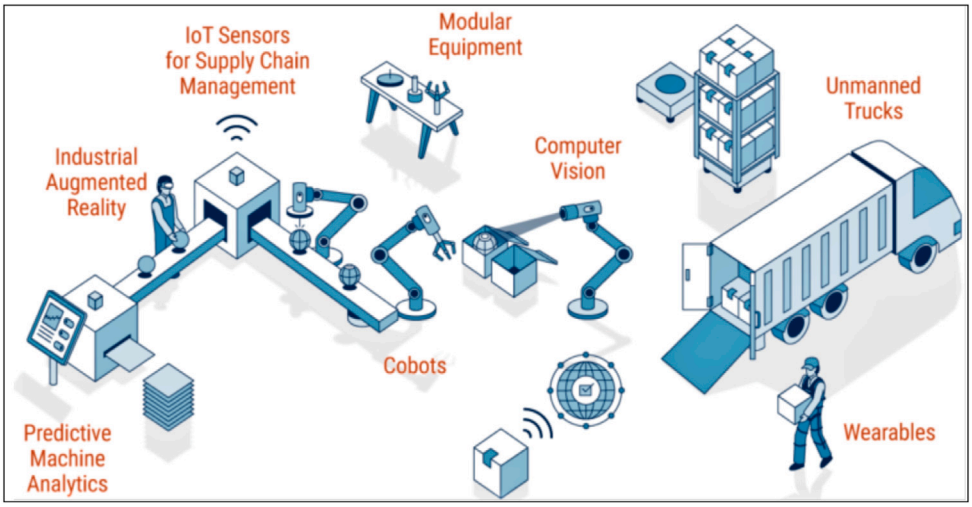

Fig. 2. IIoT applications.

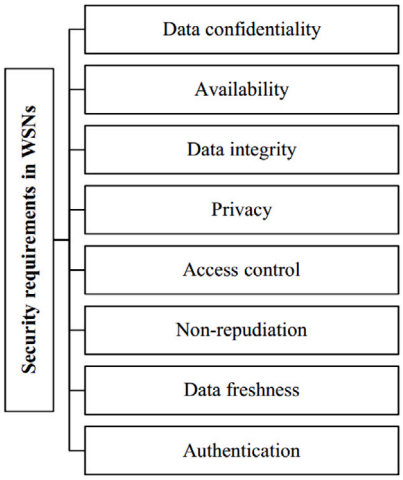

Fig. 3. Most important security requirements in WSNs.

Section 4 presents several SDA schemes and analyzes their strengths, weaknesses, and security status. Section 5 discusses SDA schemes generally. Section 6 demonstrates the most important challenges and open issues in the secure data aggregation schemes. Finally, in Section 7, the conclusion is stated.

2. Security requirements

In wireless sensor networks, data packets must be exchanged through valid sensor nodes. We know that communication channels are wireless in WSNs. Therefore, data exchanged between the sensor nodes may be corrupted due to data loss, data interference, or sabotage performed by the attacker. To protect data packets against various attacks, it is necessary to take into account the security requirements in WSNs (Stavroulakis and Stamp, 2010; Di Pietro et al., 2014). In the following, we introduce the most important security requirements in these networks. These security requirements are illustrated in Fig. 3. Also, we define the most common attacks in WSNs.

- **Availability:** This means that the information and services provided by the network must always be available (Penttinen, 2016).
- **Data confidentiality:** It protects sensitive information so that invalid nodes cannot access this information. When data packets are exchanged between sensor nodes or between the base station and sensor nodes, data confidentiality ensures that their contents are not disclosed in the network environment (Penttinen, 2016; Oreku and Pazynyuk, 2016).
- **Data integrity:** This prevents any changes in the information exchanged during the data transmission process (Di Pietro et al., 2014).
- **Access control:** This means that sensor nodes must be authenticated before accessing the network (Di Pietro et al., 2014).
- **Authentication:** It allows the receiver to verify the validity of the sender node. If an authentication mechanism is designed correctly, then only valid nodes receive messages transmitted on the network. The authentication process protects the network against various attacks and is known as the initial defense step against attackers (Di Pietro et al., 2014; Conti, 2015)
- **Data freshness:** This ensures that the data packets are new, and old data packets are not replayed (Di Pietro et al., 2014; Liu and Ning, 2007).
- **Non-repudiation:** This means that senor nodes cannot deny their participation in communication. Non-repudiation is related to the two communication parts (i.e. the source and destination) and proves that a valid node sends/receives data packets (Stavroulakis and Stamp, 2010; Conti, 2015).
- **Privacy:** This means that the personal data of one sensor node must be hidden from other sensor nodes in the network. Privacy and data confidentiality should not be confused with each other. Data confidentiality means hiding data from external entities (i.e. attackers). Whereas, privacy means avoiding interference and gathering data, which is not permitted to access (Stavroulakis and Stamp, 2010; Di Pietro et al., 2014).

2.1. Different attacks in wireless sensor networks

Today, WSNs are targeted by many attacks. These attacks are being complicated every day. These attacks disable network and disrupt its normal operation through unauthorized access, hacking, revealing secret information, altering data, or denying service. Security attacks may be active or passive. In passive attacks, attackers misemploy information or eavesdrop on communication channels without disrupting network performance. In contrast, attackers attempt to disrupt the normal network operation in active attacks (see Fig. 4(a)) (Nagireddy and Parwekar, 2019; Ni et al., 2010).

Furthermore, attacks are divided into two groups based on the attack location: internal attacks and external attacks. In internal attacks, attackers capture some sensor nodes to communicate with other nodes through the compromised nodes, and ultimately disrupt network

Table 1
Some review papers conducted in the SDA field.

Survey paper	Description
Vinodha and Anita (2019)	In Vinodha and Anita (2019), various secure data aggregation schemes had been evaluated in terms of different security requirements, including data confidentiality, data integrity, and authentication. Then, their countermeasures against some attacks were introduced. Also, SDA methods were divided into three groups in terms of network topology: ring topology, tree topology, and cluster topology. In addition, SDA approaches has been categorized into two classes in terms of cryptography techniques: hop-by-hop encryption and end-to-end encryption. In Vinodha and Anita (2019), a comprehensive review had been performed on secure data aggregation methods, we recommend that researchers study this survey.
Shah and Shukla (2012)	In Shah and Shukla (2012), it had been attempted to express the relationship between security and data aggregation process. First, the data aggregation process had been defined generally. Then, the SDA methods were grouped according to different views: (1) The data acquisition systems (query-based system or event-based system), (2) Data aggregation process (one aggregator-based system or hierarchical system). Moreover, hierarchical methods had been fallen into four groups: tree-based methods, cluster-based methods, multi-path methods, and hybrid methods. Furthermore, the authors introduced two types of data aggregation methods: (1) Lossy and Lossless methods (2) Duplicate sensitive and duplicate insensitive methods. However, the SDA schemes are classified from numerous aspects in Shah and Shukla (2012), it does not review these groups comprehensively and accurately and only presents a brief study in this area. It is the most important drawback of this survey. In addition, Priyanka et al. defined different security requirements in WSNs and introduced several attacks in these networks. Ultimately, some SDA approaches had been also presented. This survey gives a brief review of SDA methods and does not demonstrate their security techniques in the data aggregation process.
Liu et al. (2019)	In Liu et al. (2019), it had been explained why it is important to apply data aggregation methods in WSNs. Then, it describes some security issues that must be addressed in SDA methods. Next, Liu et al. reviewed the different types of data aggregation topologies, their characteristics, and differences. Then, they introduced the different security strategies utilized in SDA methods. This survey is very interesting. Studying this review paper can help researchers understand the challenges and issues related to secure data aggregation methods.
Parmar and Jinwala (2016)	In Parmar and Jinwala (2016), different the SDA methods were introduced. Parmer et al. first defined the data aggregation approaches. Then, they examined their effect on parameters such as network lifetime, delay, and accuracy. Next, various security requirements, including data confidentiality, data integrity and data freshness were presented. Furthermore, this survey said why it is important to take into account security requirements in SDA methods. In this paper, several SDA methods were evaluated and their security solutions were introduced. Finally, data aggregation schemes are classified according to homomorphic features, and their strengths and weaknesses were demonstrated. Studying this paper will be useful for researchers.
Ozdemir and Xiao (2009)	In Ozdemir and Xiao (2009), several SDA schemes were reviewed. Ozdemir et al. demonstrated some security requirements in WSNs, including data confidentiality, data integrity, authentication, availability, and freshness. Then, they categorized these approaches into two groups according to network topology: tree-based data aggregation methods and cluster-based data aggregation methods. However, we believe that these groups cannot cover all SDA schemes comprehensively. Then, the authors grouped the SDA methods into two categories in terms of cryptography techniques: symmetric encryption-based SDA methods and homomorphic encryption-based SDA methods. This survey provides a brief analysis of these methods.

Table 2
Comparison between our survey and other review papers.

Reference	Network models (homogeneous, heterogeneous)	Network topologies (hierarchical, flat, tree-based, and tree-cluster based)	Key cryptography techniques (symmetric, asymmetric, hybrid)	Encryption schemes (hop-by-hop and end-to-end)	Application	Authentication mechanisms (end-to-end and hop-by-hop)	Data recovery ability
Vinodha and Anita (2019)	×	✓	✓	✓	×	×	×
Shah and Shukla (2012)	×	✓	×	×	×	×	✓
Liu et al. (2019)	×	✓	✓	×	×	×	×
Parmar and Jinwala (2016)	×	×	✓	✓	×	×	×
Ozdemir and Xiao (2009)	×	✓	×	×	×	×	×
Our survey	✓	✓	✓	✓	✓	✓	✓

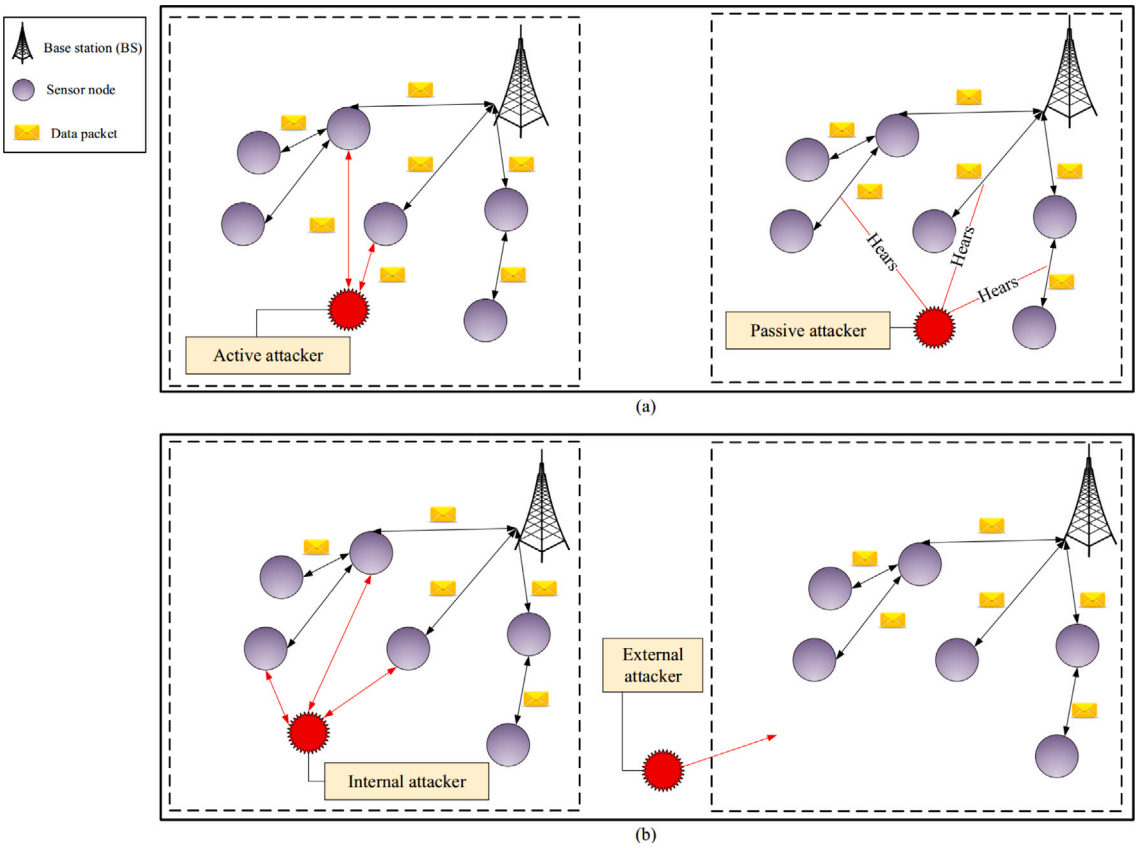

Fig. 4. Attack types: (a) Active attacks vs passive attacks; (b) Internal attacks vs external attacks.

performance. In contrast, external attacks are done by nodes that do not belong to the network (Nagireddy and Parwekar, 2019; Ni et al., 2010). See Fig. 4(b). In the following, the most common attacks on WSNs are introduced. Table 3 summarizes these attacks.

- **Black hole attack:** This is an active and external attack that threatens availability requirement. Black hole nodes claim that they have routes with zero cost (Wazid and Das, 2017). As a result, other nodes are encouraged to send their data packets through these malicious nodes. In this attack, a black hole node accesses all data packets received from other sensor nodes and removes them (Mehetre et al., 2019; Kaushik and Sharma, 2020). Fig. 5(a) displays a black hole attack.
- **Sinkhole attack:** This is an active and external attack that targets availability requirement. It is a kind of the black hole attack. but, its difference is that the attacker knows the position of the sink nod. In this case, it is more dangerous and more destructive than black hole attack because the malicious node tries to convince all sensor nodes to select it as next-hop to reach the sink node (Rehman et al., 2019; Wazid et al., 2016). The sinkhole node claims that this fake route has optimal parameters such as a shorter distance to the sink node, less traffic, best link quality and so on (Hamedheidari and Rafeh, 2013). Fig. 5(b) indicates a sinkhole attack.
- **Wormhole attack:** This is an active and external attack that threatens availability requirement. Usually, two malicious nodes participate in this attack so that they establish a wormhole channel between themselves (Dutta and Singh, 2019; Tamilarasi and Santhi, 2020). Fig. 5(c) demonstrates a wormhole attack. In wormhole tunnels, two nodes, which are actually very far apart, claim to be very close. These attackers create a tunnel between themselves and claim that it is an optimal route to another part of the network to deceive other nodes. As a result, the attacker is able to intercept the communications between sensor nodes, copy data packets, and manipulate network traffic (Ahutu and El-Ocla, 2020; Dong et al., 2011).
- **Selective forwarding attack:** This attack is also called the gray hole and is an active and external attack that menaces availability requirement. The gray hole node eliminates some received data packets selectively and forwards other data packets (Liu et al., 2015; Fu et al., 2020). A simple version of this attack is a black hole attack that removes all data packets (Yaseen et al., 2018). Fig. 5(d) displays a type of selective forwarding attack. This attack is implemented in two manners:
  - Removing a special type of data packets.
  - Removing data packets having a specific destination.
- **Sybil attack:** It is an active and external attack and can destruct network availability. In the Sybil attack, an attacker seizes multiple valid identifiers in the network. If other sensor nodes conclude that the Sybil node is their neighbor, they may select this node as next-hop to send their data. In a particular type of this attack, malicious nodes are arranged in a specific area (see Fig. 6(a)). The main purpose of this attack is to send fake data packets and ultimately disable the entire network. In another type of this attack, malicious nodes are scattered throughout the network, so that it is more difficult to detect this attack (see Fig. 6(b)) (Vasudeva and Sood, 2018; Angappan et al., 2020).
- **Flooding attack:** An active and external attack that aims to threaten network availability. In this attack, the malicious node constantly sends a connection request to the target node to fill its memory because the target node stores some information related to this request. As a result, the sensor node cannot reply to valid requests because its memory is full (Raymond and Midkiff, 2008; Perrig et al., 2004). Fig. 6(c) shows a flooding attack.

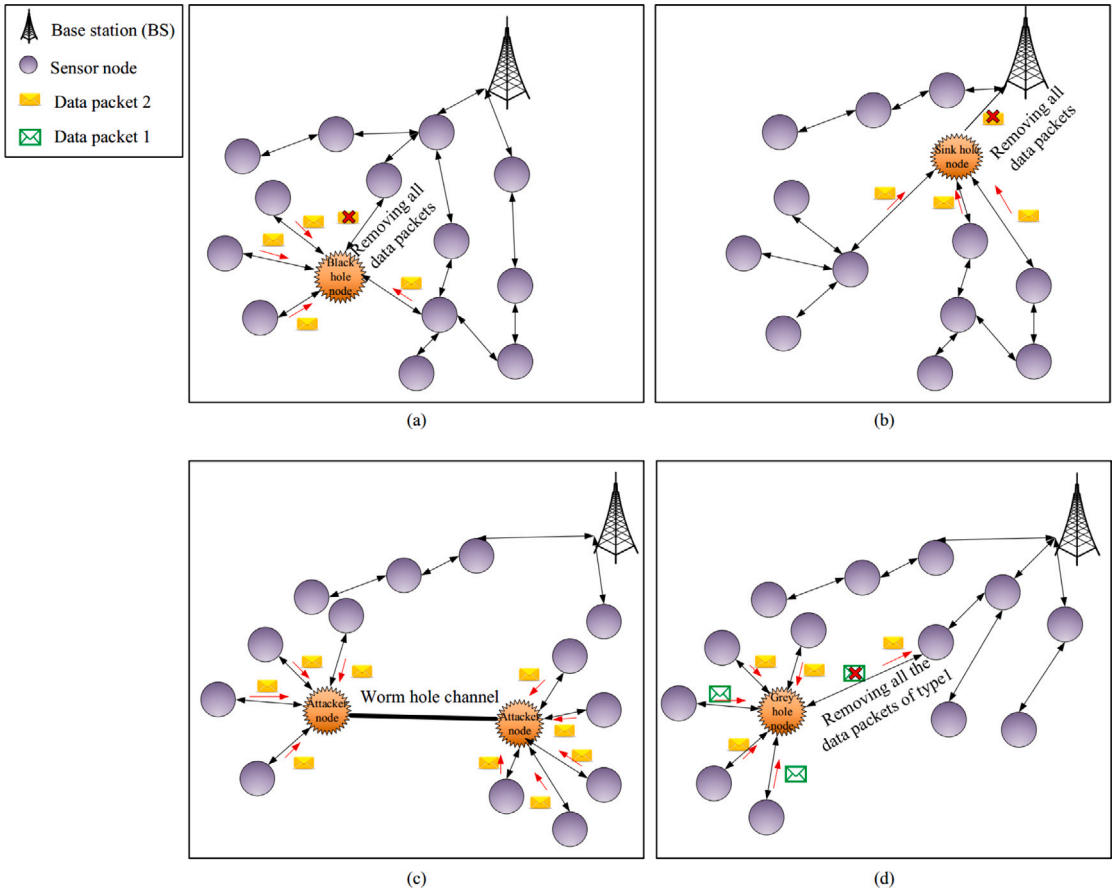

Fig. 5. Attacks related to availability: (a) Black hole attack; (b) Sinkhole attack; (c) Wormhole attack; (d) Selective forwarding attack.

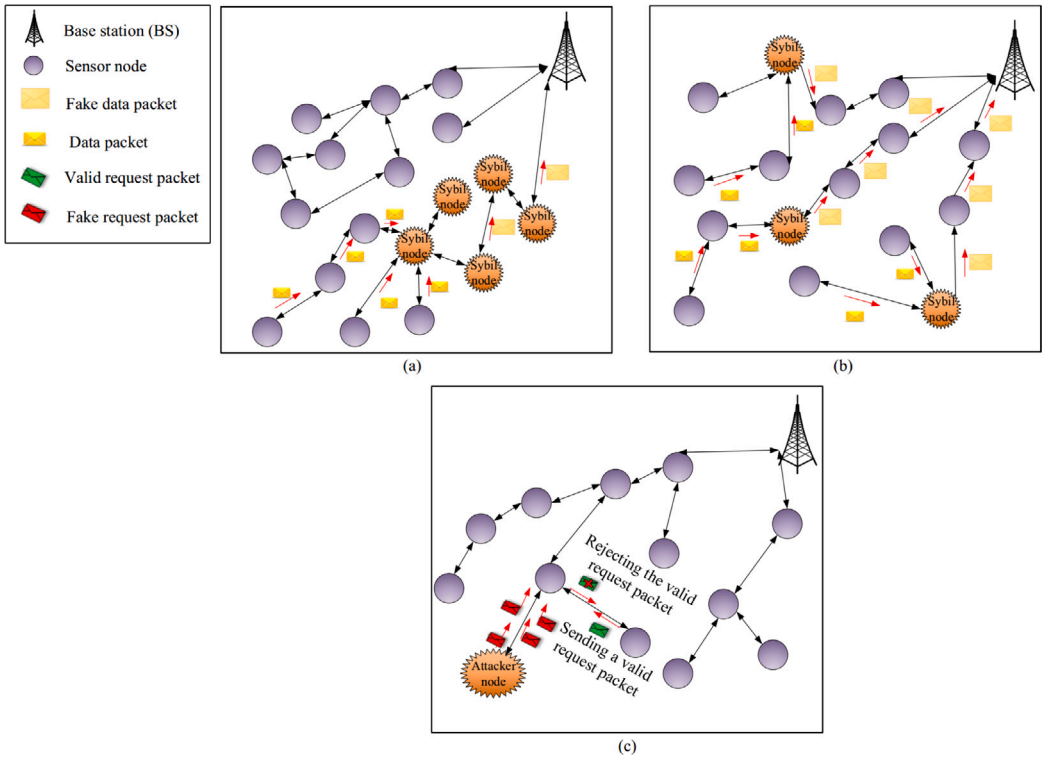

Fig. 6. Attacks related to availability: (a) Sybil attack with local effect on part of the network; (b) Sybil attack with global effect on the entire network; (c) Flooding attack.

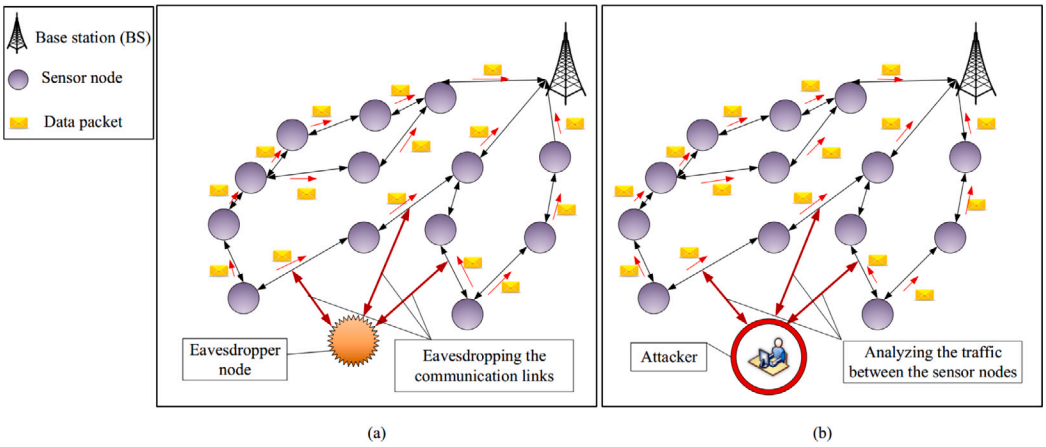

Fig. 7. Attacks related to data confidentiality: (a) Eavesdropping attack; (b) Traffic analysis attack.

- **Eavesdropping attack:** This is a passive and external attack that targets data confidentiality and privacy requirements. In this attack, the attacker tries to find out confidential data of sensor nodes by eavesdropping on wireless communication links. We know that communication nature is broadcast on WSNs (Perrig et al., 2004). Therefore, if no encryption mechanism is designed to protect important data of sensor nodes, it is very easy to trigger an eavesdropping attack successfully and eavesdrop on communication links between the sensor nodes (Chen and Lou, 2015; Pongaliur and Xiao, 2013). Fig. 7(a) represents this attack.
- **Traffic analysis attack:** A passive and external attack that threatens data confidentiality and privacy requirements. In this attack, the attacker analyzes the activities of a sensor node (Nagireddy and Parwekar, 2019; Ward and Younis, 2019). In this regard, the roles of sensor nodes and their operations are discovered. In this attack, the attacker seeks to explore traffic information, network topology, transferred message pattern, message length, message waiting time in buffer, and so on (Baroutis and Younis, 2016). Fig. 7(b) displays a traffic analysis attack.
- **Node replication attack:** It is an active and external attack that targets data integrity. This attack is very similar to the Sybil attack (Anitha et al., 2020). In a Sybil attack, there is a node having several identifiers; whereas, in the node replication attack, the attacker copies the memory of a sensor node. Then, the attacker can inject fake data packets into the network, remove data packets, and transmit modified data packets. Therefore, this attacker interferes with the network performance (Zhu et al., 2012; Numan et al., 2020). Fig. 8(a) shows a node replication attack.
- **Packet injection attack:** It is an active and internal–external attack, and its aim is to threaten data integrity. In this attack, attacker injects fake data packets into the network to disrupt data transmission process. To inject fake data packets into the network, an attacker forges valid messages on the network so that they cannot be easily distinguished from valid data packets (Illiano and Lupu, 2015; Zhu et al., 2007). Fig. 8(b) represents a packet injection attack.
- **Packet duplication attack:** It is an active and internal–external attack that jeopardizes data integrity requirement. In this attack, the malicious node replicates a valid data packet and constantly forwards it to the target node to drain its resources (memory and battery) and disrupt network performance (Nagireddy and Parwekar, 2019; Boubiche et al., 2020). Fig. 8(c) shows the packet duplication attack.
- **Packet alteration attack:** An active and internal–external attack that threatens data integrity requirement. In this attack, the attacker alters the data packets exchanged between the sensor nodes and sends the modified data packets on the network (Nagireddy and Parwekar, 2019; Boubiche et al., 2020). This attack is shown in Fig. 8(d).

3. Data aggregation in wireless sensor networks

Data aggregation is a new research field that is defined as an efficient data processing solution in large-scale WSNs (Goyal et al., 2019). In these networks, sensor nodes may generate a very large volume of data. Therefore, it is necessary to design efficient methods for data processing. The data aggregation methods can remove data redundancy, lower energy consumption, improve network lifetime, and utilize network resources optimally in large-scale WSNs (Pourghebleh and Navimipour, 2017; Boubiche et al., 2018). Therefore, data aggregation means gathering and merging useful information in a specific area of the network (Xiang et al., 2012; Kaur and Munjal, 2020). In the following, data aggregation is defined scientifically:

- **Data aggregation:** Assume that there is a data set $X = \{x_1, x_2, \ldots, x_N\}$, data aggregation is defined as $Y = F(X) = F(x_1, x_2, \ldots, x_N)$, where F is the aggregation function, Y indicates the aggregation result, and $\|Y\|$ is very smaller than $\|X\|$.
- **In-network data aggregation:** In a WSN, assume that each node generates an initial data set $X = \{x_1, x_2, \ldots, x_N\}$. Then, X can be subdivided into M subsets i.e. $\{X_1, X_2, \ldots, X_M\}$ where, $M < N$, $X_1 \cup X_2 \cup \cdots \cup X_M = X$ and $X_1 \cap X_2 \cap \ldots \cap X_M = \varnothing$. In-network data aggregation is defined as: $Y = F(X) = f(h(X_1), \ldots, h(X_M))$. In other words, the intermediate aggregator nodes execute the data aggregation process on a data subset. Finally, the sink node performs the final data aggregation process. This method optimizes the network lifetime and lowers the energy consumption of the sensor nodes (Akkaya and Ari, 2007).

Fig. 9 represents a simple example to illustrate the effect of data aggregation methods on the number of data packets transferred in a network. In this example, sensor nodes have been arranged linearly in the network. It is assumed that each node must transmit a data packet to the base station. In Fig. 9(a), network does not use the data aggregation process and the total number of transmitted data packets is equal to $\frac{N(N+1)}{2}$. Whereas, in Fig. 9(b), the data aggregation process is applied in the network. In this case, the total number of transmitted data packets is equal to N, because each node aggregates its own data packet with the data packet received from the neighboring nodes. As a result, the data aggregation process can reduce the number of data packets transferred in the network (Xiang et al., 2012; Kaur and Munjal, 2020).

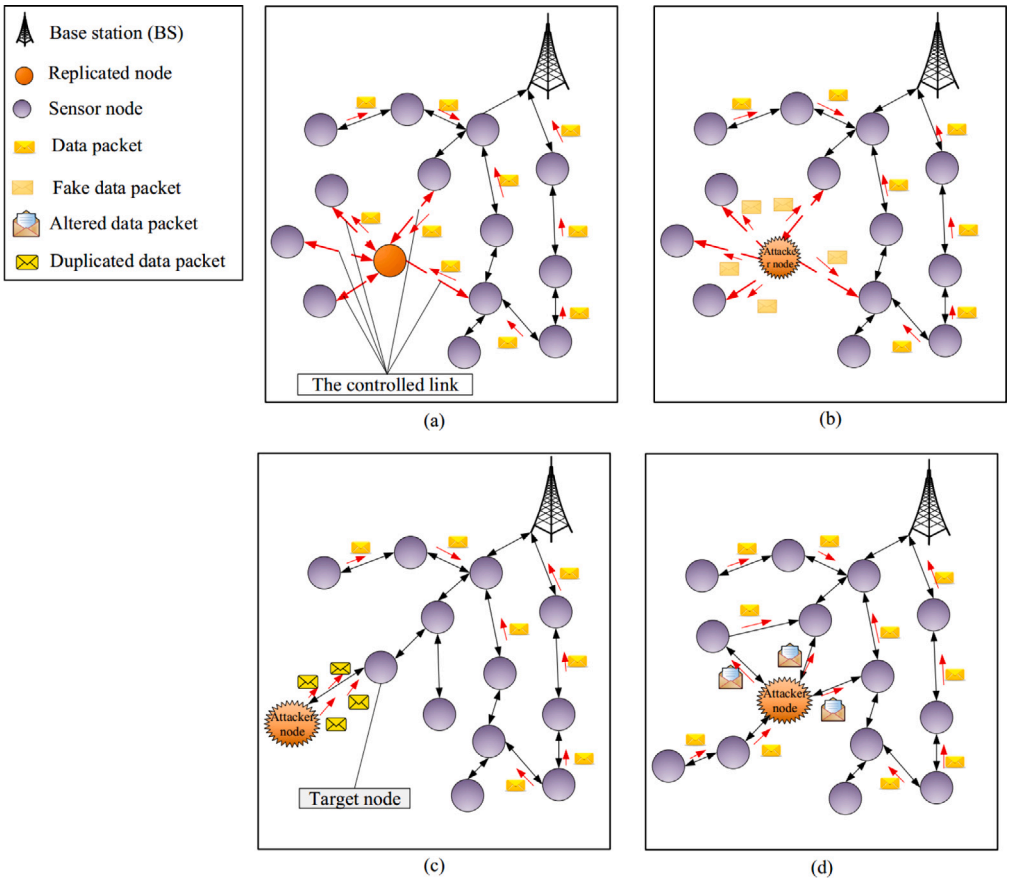

Fig. 8. Attacks related to data integrity: (a) Node replication attack; (b) Packet injection attack; (c) Packet duplication attack; (d) Packet alteration attack.

Table 3
Most common attacks in wireless sensor networks.

Attacks	Definition	Attack type	Attacker type	Target
Black hole	Attacker claims that it has optimal routes and removes all data packets.	Active	External	Availability
Sinkhole	Attacker forbids sending data packets to the sink node.	Active	External	Availability
Wormhole	Attacker interprets communications between sensor nodes, duplicates the data packets, and forwards fake data packets through a tunnel.	Active	External	Availability
Selective forwarding (Gray hole)	An attacker forwards some data packets and removes other data packets.	Active	External	Availability
Sybil	Attacker counterfeits the node ID to send fake data packets to the network.	Active	External	Availability
Flooding	Attacker constantly sends the connection request to the target node.	Active	External	Availability
Eavesdropping	Attacker eavesdrops on communication links between the two sensor nodes.	Passive	External	Data confidentiality and privacy
Traffic analysis	The attacker analyzes the activities of a sensor node.	Active	Internal	Data confidentiality and privacy
Node replication	Attacker copies the memory of a sensor node.	Active	External	Integrity
Packet injection	Attacker injects fake data packets into the network.	Active	Internal–External	Integrity
Packet duplication	The attacker node replays old data packets on the network.	Active	Internal–External	Integrity
Packet alteration	The attacker node sends the modified data packets on the network.	Active	Internal–External	Integrity

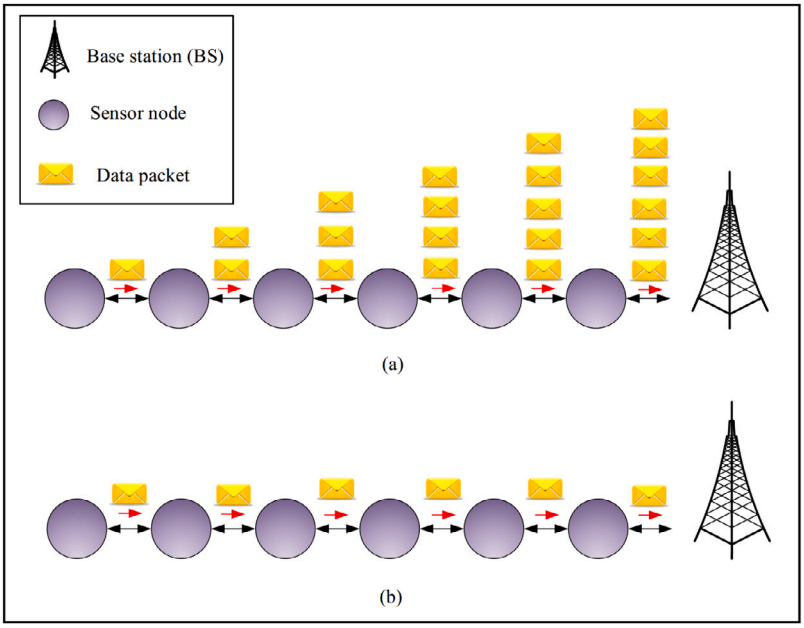

Fig. 9. Data transmission process, (a) without data aggregation mechanism (b) using a data aggregation mechanism.

3.1. Secure data aggregation (SDA)

WSNs are developed in insecure environments and are exposed to many attacks. Attackers may track or steal data packets that are forwarded to the base station. As a result, providing security is very important and challenging in designing SDA methods (Ozdemir and Çam, 2009; Lakshmi and Deepthi, 2019). Because this security mechanism must be adapted to specific characteristics of WSNs. Security in the data aggregation process means protecting the collected data and the aggregation results against any unauthorized access (Merad Boudia et al., 2018; Bodkhe and Tanwar, 2020).

3.1.1. Evaluation scales of secure data aggregation schemes in WSNs

In this section, we introduce some evaluation scales that are used to determine the efficiency of a SDA method:

- **Network lifetime:** This scale can express the efficiency of a secure data aggregation method. In WSNs, many attackers try to disable the network by increasing the energy consumption of sensor nodes. Therefore, network lifetime is a useful scale for evaluating the performance of a SDA method against attackers. In different researches, there are various definitions for network lifetime. These definitions can be based on the connectivity between nodes or the percentage of alive nodes in the network as follows:
  - *Connectivity-based (CB):* Network lifetime represents a time period from the start time (i.e. when the network is launched) until the first network partition occurs in the network (Abdollahzadeh and Navimipour, 2016). This occurs when one or more sensor nodes cannot communicate with the base station.
  - *Percentage of alive nodes (PAN):* Network lifetime indicates the time interval from the start time until the number of alive nodes is less than a certain threshold in the network (Abdollahzadeh and Navimipour, 2016).

  Moreover, network lifetime can be defined based on the three scales, namely first node die (FND), half of the nodes die (HND), and last node die (LND) as follows (Abdollahzadeh and Navimipour, 2016):
  - *FND:* Based on this scale, network lifetime is defined as a time interval from the start time until the first node dies in the network. In most secure data aggregation schemes, this scale is used to express the network lifetime.
  - *HND:* According to this scale, network lifetime is a time interval from the start time until half of the nodes die in the network. This scale is often used for scenarios where the density of nodes is high, so that the nodes are close to each other and neighboring nodes may sense similar data. In this case, the death of a small number of nodes has no effect on network performance.
  - *LND:* Based on this scale, network lifetime means the time interval from the start time until the last node dies in the network.
- **Accuracy:** In SDA methods, this scale is applied to measure the difference between the aggregated data result and the actual value. Accuracy is an important metric for evaluating a secure data aggregation scheme (Zhu et al., 2017). Because if the aggregated data result is false, it may cause a wrong decision by the system and lead to unpleasant results. In WSNs, attackers try to capture some sensor nodes in the network. They use these nodes to inject incorrect data into the network. This may cause a lot of errors in the aggregated data result.
- **Delay:** In the data transmission process, the delay is defined as the time interval since sensor nodes generate data until the base station receives this data (Shah and Shukla, 2012; Liu et al., 2019). A SDA method may increase the end-to-end delay because aggregator nodes must wait a time interval, before aggregating the data, to ensure that they have received all the data sent by the sensor nodes. If this time interval is short, then the accuracy of the aggregated data result will be decreased. Because some data of sensor nodes may be lost. On the other hand, if this time interval is long, then delay will be increased.
- **Scalability:** This scale indicates the ability to maintain network performance relative to network size (Fahmy, 2020; Khan et al., 2016). Scalability is very important in a SDA scheme, which mainly facilitates the data transmission process in large-scale wireless sensor networks. If a SDA scheme is not scalable, it is often considered as an inefficient secure data aggregation method.

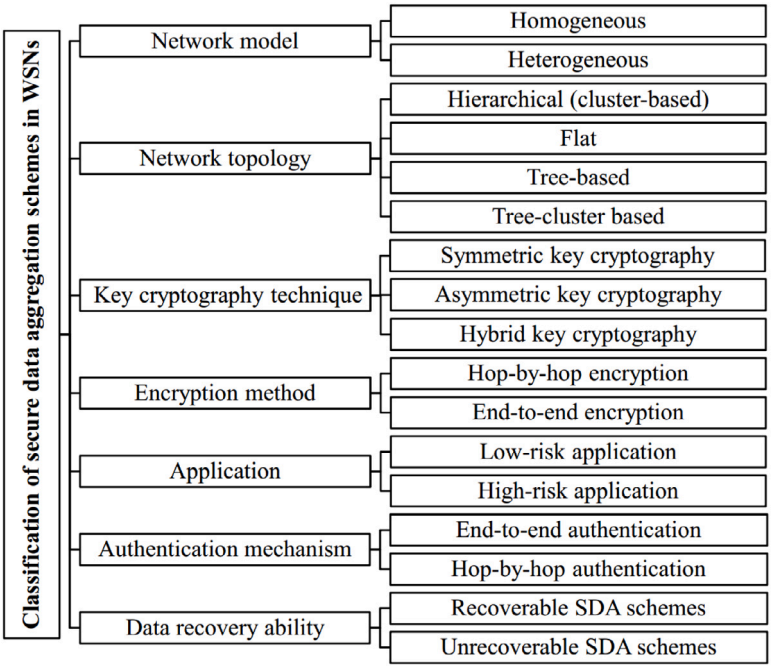

Fig. 10. Our proposed classification of SDA schemes.

- **Energy consumption:** To increase network lifetime, secure data aggregation methods must address energy issue because sensor nodes have limited energy resources (Fahmy, 2020). On the other hand, some sensor nodes, such as aggregator nodes, cluster head nodes, nodes close to BS, etc., have more overhead than other nodes and consume high energy. As a result, these nodes lose their energy quickly. Therefore, a SDA scheme should balance energy consumption in the network to extend network lifetime.
- **Security:** It is an important scale for evaluating SDA schemes in wireless sensor networks (Khan et al., 2016). If some sensor nodes are compromised in the network, they may affect the overall network performance. Hence, network security should be ensured when there are some compromised nodes such as cluster head (CH) node, aggregator nodes, cluster member nodes, etc. and they should not have any effect on the normal network operation.

3.1.2. Classification of secure data aggregation schemes in WSNs

In this section, we propose a detailed classification of SDA schemes. This classification is as follows:

- Network models in SDA schemes (i.e. homogeneous, heterogeneous)
- Network topologies in SDA schemes (i.e. hierarchical (cluster-based), flat, tree-based, and tree-cluster based)
- Key cryptography techniques in SDA methods (i.e. symmetric key cryptography, asymmetric key cryptography, hybrid key cryptography)
- Encryption methods in SDA schemes (hop-by-hop encryption and end-to-end encryption)
- Types of secure data aggregation methods based on application (i.e. low-risk application and high-risk application)
- Authentication mechanisms in SDA methods (end-to-end authentication and hop-by-hop authentication)
- Types of SDA schemes based on data recovery ability (i.e. recoverable SDA schemes and unrecoverable SDA schemes)

In the following, our proposed classification has been illustrated in Fig. 10.

Classification of secure data aggregation schemes based on network model. In our proposed classification, SDA methods are categorized into two classes based on the network model: homogeneous and heterogeneous. In the following, we explain these two classes:

- **Homogeneous wireless sensor networks:** In this model, the sensor nodes are similar in terms of energy, processing power, memory capacity and other hardware features (Abdollahzadeh and Navimipour, 2016). In SDA schemes, including homogeneous network model, selecting aggregator nodes (ANs) is a serious challenge because all sensor nodes have limited energy. Therefore, nodes, which have more residual energy than other nodes, should be selected as aggregator nodes. Because aggregator nodes require a lot of energy for data aggregation, data transmission, encryption, decryption, data integrity confirmation, fake data detection, and so on.
- **Heterogeneous wireless sensor networks:** In this model, sensor nodes have different hardware characteristics such as energy, memory, processing power, communication range, and so on (Abdollahzadeh and Navimipour, 2016). As a result, topology control and management as well as the deployment process in heterogeneous WSNs are more complex than homogeneous WSNs. In designing secure data aggregation methods, whose network model is heterogeneous; aggregator nodes should be selected from sensor nodes with more energy and high computing power. Furthermore, low-energy nodes are responsible for sensing the environment, collecting data, and transmitting this data to aggregator nodes.

Classification of secure data aggregation schemes based on network topology. In our proposed classification, we categorize SDA methods based on network topology as follows:

- **Hierarchical (cluster-based) topology:** In this topology, sensor nodes play different roles such as cluster head (CH) node, aggregator node (AN), and cluster member (CM) node (Randhawa and Jain, 2017; Khan et al., 2016). In cluster-based SDA schemes, the network is divided into several clusters, and each cluster has a CH node. In these schemes, selection of CH nodes is a very important challenge. The hierarchical topology may include heterogeneous nodes so that there are a large number

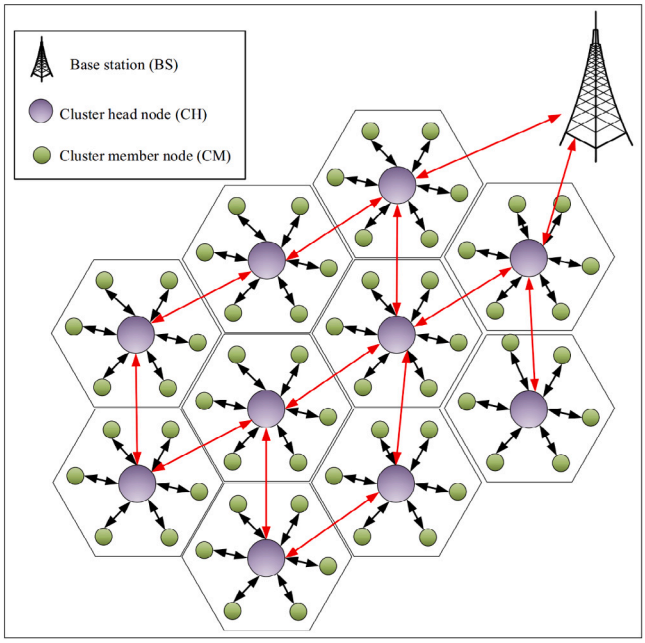

Fig. 11. Hierarchical (cluster-based) topology.

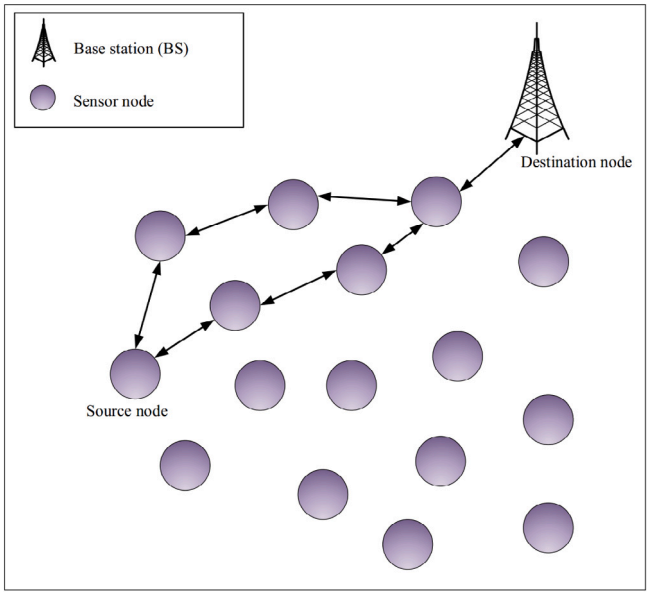

Fig. 12. Flat topology.

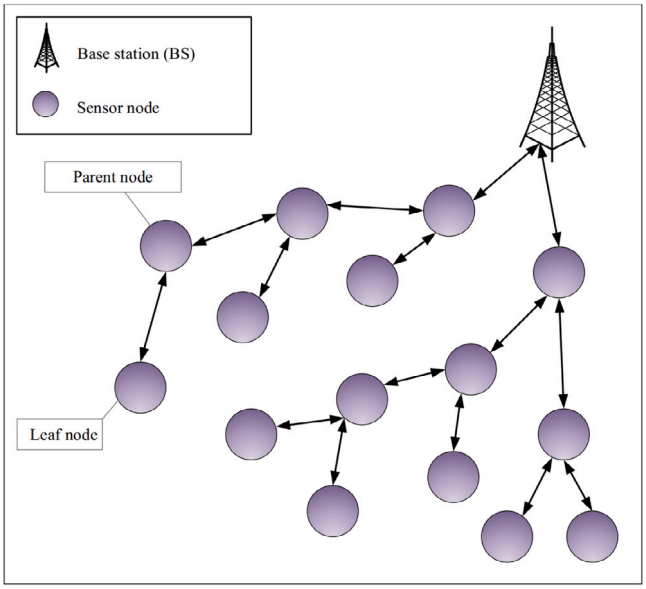

Fig. 13. Tree-based topology.

of low-energy sensor nodes and a small number of high-energy nodes in the network. In the lower levels of this hierarchy, low-energy nodes are applied. They are responsible for collecting data and transferring it to higher levels. In addition, high-energy nodes are used at higher levels of the hierarchy to aggregate data and perform more complex operations such as encryption, decryption, authentication, and so on. Generally, SDA methods, which have hierarchical topology, include two data aggregation processes: intra-cluster data aggregation and inter-cluster data aggregation. The CH node manages the intra-cluster data aggregation process. Hierarchical SDA schemes consume energy efficiently and improve network lifetime. These methods are also scalable. Therefore, they are a suitable solution for large-scale WSNs. Furthermore, these schemes decrease network traffic and reduce delay in the data transmission process. In this topology, the main challenge is the energy consumption of CH nodes, because a CH node is not only responsible for aggregating data of its CM nodes, but also collaborates in the inter-cluster aggregation process and must send data received from other CH nodes to the base station. Fig. 11 displays this topology.

- **Flat topology:** In this topology, all sensor nodes have the same role in the network (Randhawa and Jain, 2017; Khan et al., 2016). In a flat-based SDA method, each sensor node is responsible for sensing the data, aggregating its data with neighboring nodes, and sending the data to the BS. Sensor nodes send their data to the base station via multi-hop routes. Therefore, in these methods, there is a lot of delay in the data transmission process. Also, in these methods, the nodes, which are close to BS, have a lot of traffic and high communication overhead. Hence, they consume a lot of energy and die quickly. Therefore, in flat-based SDA methods, energy consumption is unbalance. It reduces network lifetime. However, these methods are simple, but they are not scalable. Therefore, they are not suitable for large-scale wireless sensor networks. This topology has been shown in Fig. 12.
- **Tree-based topology:** In SDA schemes, the simplest method is to select a number of aggregator nodes in the network and determine routes for sending data from sensor nodes to the BS through these aggregator nodes. Tree-based topology is one of the most common topologies for SDA methods in WSNs (Mehrjoo and Khunjush, 2018). In this topology, there are two types of sensor nodes: parent nodes (aggregator nodes) and leaf nodes. In tree-based SDA methods, an important challenge is that an aggregation tree is created between the sensor nodes in the network, so that its root is the BS. This aggregation tree must optimize energy consumption and increase network lifetime. In these schemes, a fixed and unique route is established between the BS and each sensor node. Therefore, aggregator nodes consume a lot of energy consumption and have a high communication overhead. In this topology, data packet loss is very high because data transmission routes are fixed. On the other hand, it reduces delay in the data transmission process greatly because the routes are predetermined, and there is no need to discover the route before the data transmission process. It should be noted that tree-based topology is suitable for low-density networks. This topology is illustrated in Fig. 13.
- **Tree-cluster based topology:** This topology combines both cluster-based topology and tree-based topology (Vinodha and

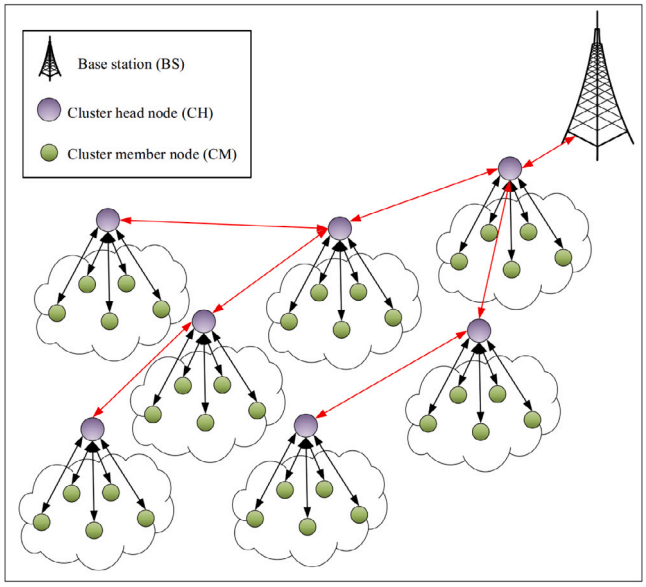

Fig. 14. Tree-cluster topology.

Anita, 2019; Liu et al., 2019). It utilizes advantages of these topologies and reduces their drawbacks. In this topology, the network is divided into a number of clusters and an aggregation tree is created between CH nodes. This topology can optimize energy consumption in the network and ensure scalability. Fig. 14 depicts tree-cluster topology.

In the following, the strengths and weaknesses of different topologies are summarized in Table 4.

Classification of secure data aggregation schemes based on key cryptography technique. In a SDA scheme, it is very important to support data confidentiality (Messai and Seba, 2016; Mustafa et al., 2018). Cryptography is one of the mechanisms that ensures data confidentiality (Choubey and Hashmi, 2018). Cryptography is an important research field that utilizes complex mathematical techniques. In the proposed classification, we categorize SDA methods into three groups based on key cryptography technique: symmetric key cryptography, asymmetric key cryptography, and hybrid key cryptography.

- **Symmetric key cryptography:** When the same key is applied in both encryption and decryption processes, it is known as symmetric key cryptography in which both communication parts (i.e. sender and receiver) agree on a common secret key. Symmetric key cryptography is a very fast method. Furthermore, it has a low computational overhead. Therefore, it is extensively used in secure data aggregation for WSNs. However, this method has a lower security than asymmetric key cryptography methods. Some examples of symmetric key cryptography techniques are: AES, DES, RC4, and so on.
- **Asymmetric key cryptography:** This method is also known as public key cryptography technique. This technique was introduced by Diffie and Hellman. In this scheme, two keys are used in the encryption and decryption processes:
  - Public keys are available to any sensor nodes on the network and are used in the message encryption process.
  - Private keys are secret and are applied in the message decryption process.

  Asymmetric key cryptography schemes provide higher security than symmetric key cryptography schemes. However, these methods have a lot of computational overhead. In this cryptography scheme, the encryption and decryption processes are slower than symmetric key cryptography method. Examples of these methods are RSA, ECC, and so on.
- **Hybrid key cryptography:** Some secure data aggregation methods use both symmetric and asymmetric key cryptography techniques. This improves energy consumption, computational overhead and network security.

In the following, various key cryptography techniques are compared in Table 5.

Classification of secure data aggregation schemes based on encryption methods. In our proposed classification, we classify SDA methods into two categories based on encryption methods:

- **Hop-by-hop encryption technique:** In SDA schemes, which apply hop-by-hop encryption method, the data encryption and decryption processes are executed in each hop (Halak, 2018; Lindell, 2017). In this regard, each aggregator node firstly receives the encrypted data packets, secondly decrypts them, then aggregates these data packets, and ultimately encrypts the aggregated data and forwards it to the next step. In this method, it is possible for an attacker to capture the intermediate nodes and achieve data packets. Therefore, network security is threatened in terms of data confidentiality and privacy. Also, this encryption method increases energy consumption in secure data aggregation schemes and delay in the data transmission process. However, it can be implemented easily. In Fig. 15, the hop-by-hop encryption process is shown.
- **End-to-end encryption technique:** In SDA schemes, which apply this encryption technique, the data decryption process is executed only at the base station (Halak, 2018; Lindell, 2017). Intermediate aggregator nodes perform data aggregation operations on encrypted data without any knowledge of the data content. Compared to the hop-by-hop encryption technique, this method increases data security and lowers energy consumption at intermediate nodes because it does not need to execute encryption and decryption processes in each hop. Moreover, the end-to-end encryption technique reduces delay in the data transmission process. However, implementing this encryption method is more complicated than the hop-by-hop encryption scheme. Fig. 16 demonstrates the end-to-end encryption process.

The most important features of these encryption methods are summarized in Table 6.

Classification of secure data aggregation schemes based on application. To design efficient SDA methods in WSNs, it is necessary to take into account the security requirements of an application. In the proposed classification, we classify secure data aggregation schemes in two classes based on application:

- **Low-risk applications:** In these applications, it is no matter to design robust security mechanisms because security is not a very critical requirement, meaning that modifying or losing some data packets has no effect on the network performance. An example is an application that collects ambient temperature. In this application, missing some data packets is not important. In fact, there are less likely to attack these applications. As a result, designing complex security mechanisms with a very high-security level for these applications only wastes resources, increases delay in the data transmission process, and boosts energy consumption in the network. In designing a SDA method, the first step is to protect data confidentiality. Therefore, this security requirement should be considered in low-risk applications. As a result, secure data aggregation methods designed for these applications must resist attacks such as Eavesdropping and Traffic Analysis.

Table 4
Comparison between different topologies in SDA methods.

Network topology	Strength	Weakness
Hierarchical (cluster-based) topology	Improving energy consumption, increasing network lifetime, scalability, reducing network traffic, lowering delay in the data transmission process	The main challenge in this network topology is high communication overhead and the high energy consumption in CH nodes.
Flat topology	Simplicity, being suitable for small networks with a small number of nodes	High delay in the data transmission process, high communication overhead in the nodes, which are close to BS, unbalanced energy consumption in network, lowering network lifetime, low scalability
Tree-based topology	Determining routes before the data transmission process, low energy consumption, low delay in the data transmission process	Establishing fixed routes for data transmission, increasing packet loss rate, low scalability
Tree-cluster topology	This method includes the advantages of both hierarchical topology and tree-based topology.	This method reduces the disadvantages of both hierarchical and tree-based topologies, but it still has these drawbacks.

Table 5
Comparison between different key cryptography techniques in SDA methods.

Key cryptography techniques	Computational overhead	Delay	Security
Symmetric key cryptography	Low	Low	Low
Asymmetric key cryptography	High	High	High
Hybrid key cryptography	Low	Low	High

Table 6
Comparison between hop-by-hop and end-to-end encryption techniques.

Encryption techniques	Security	Energy consumption	Delay	Implementation
Hop-by-hop encryption	Low	High	High	Simple
End-to-end encryption	High	Low	Low	Complicated

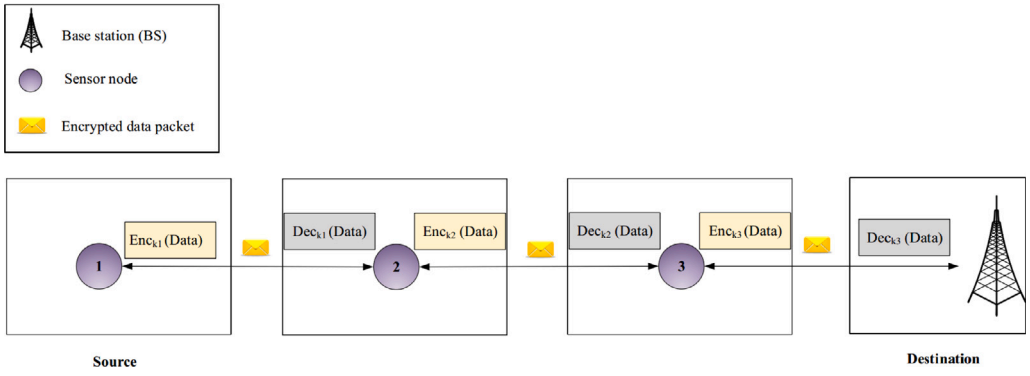

Fig. 15. Hop-by-hop encryption process.

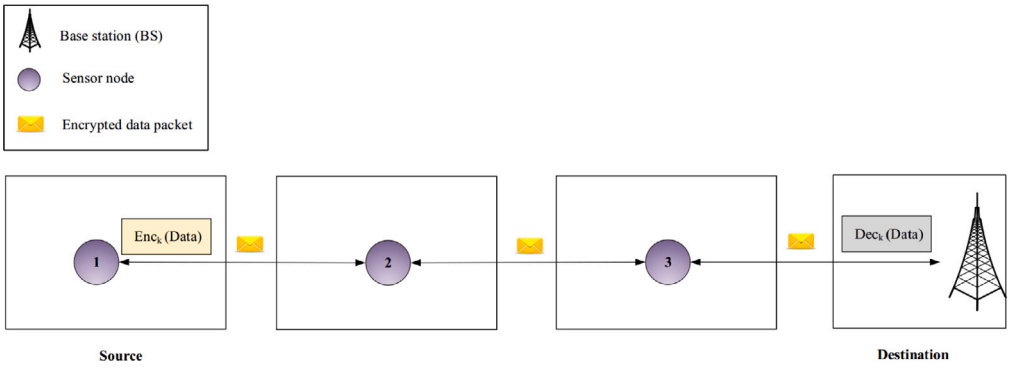

Fig. 16. End-to-end encryption process.

- **High-risk applications:** In these applications, network security is very vital. For example, data protection is very important in healthcare applications, military applications, IoT, IIoT, and so on; and if an attacker changes data slightly, then it can be very dangerous. Therefore, network designers must design robust security mechanisms for these applications. In this case, creating robust security mechanisms is very important and reasonable even if they increase energy consumption and reduce network

Table 7
Comparison of various authentication mechanisms in SDA schemes.

Authentication mechanism	Computational overhead	Delay	Energy consumption	Security
End-to-end authentication mechanism	Low	Low	Low	Low
Hop-by-hop authentication mechanism	High	High	High	High

Table 8
Comparison between recoverable SDA schemes and unrecoverable SDA schemes.

Scheme	Advantage	Disadvantage
Recoverable SDA scheme	Accessing to all original data and performing various aggregation operations on this data by the base station	Increasing energy consumption and delay in the data transmission process, low scalability, enlarging data packet size in each hop
Unrecoverable SDA scheme	Reducing data packet size, decreasing delay and energy consumption in the data transmission process, reducing data redundancy, scalability	The aggregation result may be incorrect.

lifetime. These applications can be attacked many times, and attackers may corrupt the encrypted data on the network by performing various attacks described in Section 2 so that data packets are not rightly received by BS. Therefore, when designing SDA schemes for high-risk applications, network designers must consider not only data confidentiality but also other security requirements such as data integrity, availability, authentication, etc. In these secure data aggregation methods, it is necessary to design appropriate authentication techniques to detect invalid nodes in the network.

Classification of secure data aggregation schemes based on authentication mechanism. Authentication helps a sensor node to authenticate the source of a data packet in the network. Also, it guarantees data integrity. In SDA schemes, an attacker can disrupt the data aggregation process and change the aggregation result via modifying data packets or injecting fake data packets into the network (Mishra and Turuk, 2016; Yugha and Chithra, 2020). Therefore, aggregator nodes (or base station) must ensure that data is sent by valid nodes in the network. Today, many authentication techniques have been proposed in WSNs such as digital signature, message authentication code (MAC), and so on (Atwady and Hammoudeh, 2017; Liyanage et al., 2020b). In the following, we describe these techniques briefly:

- **Digital signature:** Digital signature is a robust tool that authenticates messages sent on a communication channel. This method is originated from public key cryptography. It is applied to ensure some security requirements, including data integrity, non-repudiation, etc.
- **Message authentication code (MAC):** The purpose of the message authentication code is to authenticate the message sent between the receiver and the sender using a common symmetric key to ensure data integrity.

In this survey, we classify authentication mechanisms in two groups:

- **End-to-end authentication mechanism:** In this technique, a centralized authentication process is executed by the base station to validate sender nodes of data packets. This method has low computational overhead and energy consumption. Furthermore, this scheme decreases end-to-end delay in the data transmission process because other sensor nodes do not participate in the authentication process. However, it also has a major drawback: if an attacker captures a sensor node in the network. It can use this node to send fake data packets to the network. Other nodes do not know that this node is invalid. As a result, they participate in sending fake data packets to BS. Hence, sensor nodes consume a lot of energy and waste limited network resources.
- **Hop-by-hop authentication mechanism:** In this method, a decentralized authentication mechanism is performed by sensor nodes in each hop, meaning that aggregator nodes participate in the authentication process. This technique provides better security than the end-to-end authentication mechanism because if a sensor node is compromised, aggregator nodes are able to identify this node quickly. However, it increases computational overhead and energy consumption in aggregator nodes. Moreover, the end-to-end delay is increased in the data transmission process.

In Table 7, these authentication mechanisms have been compared with each other.

Classification of secure data aggregation schemes based on data recovery ability. In our proposed classification, we divide the SDA methods into two categories based on data recovery ability:

- **Recoverable SDA schemes:** In this category, the data aggregation process focuses on data compression so that all sensed data is sent to the base station. These SDA schemes have one major advantage: the base station has access to all sensed data and can execute any data aggregation operations on the raw data. However, they have one fundamental drawback: enlarging data packet size in each hop. This increases energy consumption and end-to-end delay, and reduces network efficiency in large-scale networks.
- **Unrecoverable SDA schemes:** In this category, the aggregator nodes locally execute an aggregation function (for example, MAX, MIN, SUM, etc.) on the data packets received from sensor nodes and forwards the aggregated data packet to the BS. In these methods, the base station can only access the aggregated data and cannot retrieve the original data. This category has several advantages, like reducing the data packet size, lowering the end-to-end delay in the data transmission process, minimizing data redundancy, and so on. However, these schemes have one main drawback: if the intermediate nodes (aggregator nodes) perform the aggregation operations inaccurately, incorrect aggregated data will be produced.

Table 8 lists the most important advantages and disadvantages of these schemes.

4. Investigating several secure data aggregation methods

In this section, we introduce some SDA approaches and present their strengths and weaknesses. Also, we describe their network model, network topology, key cryptography method, and other features based on the proposed classification. Next, we evaluate each SDA method in terms of the security requirements introduced in Section 2. Finally, we analyze their solution to counteract different attacks. Our purpose is to determine what security requirements can be met by a SDA approach. This helps us to understand what applications (i.e. low-risk application or high-risk application) can apply this SDA scheme.

Table 9
Main features of the MODA method.

Method	Network model	Network topology	Encryption technique	Weaknesses	Strengths
MODA (Zhang et al., 2018)	Homogeneous	Tree-based	Homomorphic encryption technique (an asymmetric key cryptography scheme)	Low scalability, high data redundancy, low network lifetime, not designing a mechanism for removing duplicated data	Using an end-to-end encryption method and designing a multi-functional aggregation scheme

4.1. MODA

Zhang et al. (2018) presented the multi-functional secure data aggregation scheme (MODA). For implementing MODA, a homogeneous WSN is applied. Furthermore, sensor nodes are arranged in a tree topology to transmit the aggregated data to the sink node. MODA uses the homomorphic encryption technique to provide the security of messages exchanged between sensor nodes. It is an asymmetric key cryptography method. MODA has five phases:

- **Mapping:** Each sensor node k (where, $k = 1, \ldots, N$ and N is equal to the total number of sensor nodes in the network) maps its raw data (i.e. $x_k \in (X_{LB}, X_{UB}]$, where, X_{LB} and X_{UB} are lower boundary and upper boundary, respectively) to y_k (Where, $y_k \in (0, L]$ and $L = \left\lceil \frac{X_{UB}-X_{LB}}{a} \right\rceil$, and a is a constant value) using a monotonic mapping function.
- **Encoding:** The sensor node k transforms y_k into an encoded vector $\overrightarrow{v_k}$ using a certain function. The vector $\overrightarrow{v_k}$ (where, $\overrightarrow{v_k} \in \{0,1\}^L$) has L elements so that the y_kth element is one and its other elements are equal to zero.
- **Encryption:** The sensor node k encrypts $\overrightarrow{v_k}$ using a homomorphic encryption method and ultimately forwards it to its parent node.
- **Aggregation:** Upon receiving data packets from child nodes, the parent node aggregates them. The data aggregation operation is executed directly on the encrypted data packets and the aggregator node is unaware of their content (end-to-end encryption).
- **Decryption:** Only sink node executes the decryption operation. The sink node decrypts data packets and extracts raw data to perform the desired operation on it.

4.1.1. Strengths and weaknesses

In this section, we state the most important strengths and weaknesses of MODA (Zhang et al., 2018). Moreover, we list the main features of this method in Table 9 briefly. Its strengths are:

- In Zhang et al. (2018), an end-to-end encryption method is applied. This method does not require to encrypt/decrypt data packets at each hop; hence, energy consumption will be reduced.
- It presents a novel idea called multi-functional aggregation in which the sink node can extract different statistical results from the received data packet.

In the following, we state the most important weaknesses of the MODA scheme:

- In MODA, sensor nodes are organized in a tree topology. Therefore, if the number of sensor nodes is increased, then it is necessary to be designed a tree with high depth and branches. In this situation, energy consumption is boosted in the sensor nodes located in the upper levels of the aggregation tree, because these sensor nodes must send, receive, and aggregate very high data packets related to the sensor nodes in their subtree and die quickly. As a result, it can be said that MODA does not guarantee scalability and is not suitable for large-scale WSNs.
- The encoding phase has high data redundancy. This increases communication overhead dramatically. Zhang et al. in Zhang et al. (2018) presented two improved versions called RODA and CODA. However, these optimized schemes do not achieve more success.
- In dense WSNs, several sensor nodes may overlap with each other and sense the same data. This wastes resources and increases energy consumption in the network. An efficient data aggregation method should design an appropriate mechanism for removing duplicate data in the data aggregation process. In MODA, there is no mechanism for detecting duplicate data. Therefore, it may boost communication overhead and energy consumption.
- In this method, sensor nodes close to the sink node have high communication overhead because other sensor nodes utilize these nodes to send their data packets to the sink node. As a result, these sensor nodes consume a lot of energy and die quickly. Therefore, connections between the sink node and other sensor nodes may be lost. It reduces network lifetime.

4.1.2. Evaluating the MODA method in terms of the security requirements

In this section, we evaluate MODA (Zhang et al., 2018) in terms of the security requirements introduced in Section 2 to determine what requirements are addressed by this method.

- **Data Confidentiality:** MODA presents a homomorphic encryption technique to protect data confidentiality. Sensor nodes encrypt their own data and then send their encrypted data packet to the parent node. Each sensor node has a unique key. Only the sink node is aware of these keys. As a result, even if an attacker captures some aggregator nodes, it cannot decrypt the data packets of other sensor nodes in the network. Overall, the MODA method adequately ensures data confidentiality.
- **Privacy:** As mentioned, this scheme applies an end-to-end encryption scheme to ensure the privacy and data confidentiality.

4.1.3. Countermeasures against various attacks

In this section, we introduce the attacks that MODA (Zhang et al., 2018) can detect and prevent. This analysis helps us to be aware of the security level of this method.

- **Eavesdropping:** As mentioned earlier, a homomorphic encryption technique has been designed to guarantee privacy and data confidentiality. This scheme can help sensor nodes to protect their secret data. An attacker cannot achieve private keys of all sensor node to decipher their data packets. Therefore, if an attacker eavesdrops on communication links, it only accesses encrypted data packets, and cannot find out their content.
- **Traffic analysis:** The encryption process used in the MODA method protects the network against this attack. If an attacker analyzes network traffic, it cannot access information such as message pattern, message length, and so on because only the sink node is aware of the decryption keys.

4.2. EHDA

Ullah et al. (2020) introduced the efficient healthcare data aggregation (EHDA) scheme. It utilizes the FoG server in the data aggregation process. This scheme was presented for homogeneous WSNs. In addition, sensor nodes have been arranged in a hierarchical topology. Sensor nodes transmit their data packets to aggregator nodes in a single-hop manner. The aggregator node is tasked to send data packets to the base station (FoG server) directly (single-hop manner) or through intermediate aggregator nodes (multi-hop manner). The FoG server extracts data from data packets and stores them in local memory. Ultimately, the FoG server uploads data saved in its local memory into a cloud memory at regular intervals. This method uses a symmetric key cryptography technique to secure communication links between sensor nodes. The EHDA scheme has three phases:

- **Local data transmission phase:** In this phase, a sensor node must send its data packets to the aggregator node. For this purpose, it first encrypts its data using a symmetric key. It should be noted that the FoG server has preloaded this key into the memory of the sensor node. Secondly, the sensor node calculates the hash value of the encrypted data based on its key and a timestamp. Then, the sensor node compresses the encrypted data along with the hash value. Finally, it forwards the data packet, including its ID, timestamp, and compressed data, to the aggregator node.
- **Data packet receipt phase:** In this phase, the aggregator node receives data packets sent from its cluster member (CM) nodes according to an algorithm called MRA (Ullah et al., 2020). Based on this algorithm, the aggregator node first checks the timestamp of the data packet received from a sensor node to ensure that the data packet is new. Therefore, the aggregator node can detect and remove old data packets. Then, the aggregator node recalculates the hash value and compares it with the value inserted into the data packet. If these two values are the same, the data packet is valid, otherwise, it is deleted. Finally, the aggregator node aggregates all received data packets, encrypts them, and finally sends the encrypted data.
- **Data extraction phase:** The FoG server extracts the data received from aggregator nodes based on an algorithm called MEA (Ullah et al., 2020). According to this algorithm, the FoG server first decrypts the received data packets and then decompresses them. Then, it checks their timestamps to ensure that they are new. Next, the FoG server recalculates the hash value and compares it with the value inserted into the data packets. If these two values are the same, then it verifies the data packets. Therefore, the FoG server can extract the data and finally stores it in its memory.

4.2.1. Strengths and weaknesses

In this section, we describe the most important strengths and weaknesses of EHDA (Ullah et al., 2020). Moreover, Table 10 summarizes the main characteristics of this method. In the following, we state its strengths:

- In the EHDA method, sensor nodes are organized in a cluster-based hierarchical topology. Therefore, this scheme is scalable because the clustering process reduces energy consumption and optimizes network lifetime.
- It presents a symmetric key cryptography technique. This technique has low computational overhead and reduces energy consumption.
- It applies the end-to-end encryption technique to secure the data transmission process. Hence, it reduces energy consumption in the network.
- This method presents a mechanism for detecting and removing old messages. This mechanism optimizes network performance.

In the following, the most important weaknesses of the EHDA method are demonstrated:

- In large-scale WSNs, some sensor nodes may overlap with each other and sense the same data. In EHDA, there is no mechanism for removing data redundancy.
- In this method, the aggregator nodes close to the FoG server have a very high communication overhead because other aggregator nodes apply these nodes to send their data to the FoG server. As a result, these sensor nodes have high energy consumption. EHDA cannot balance energy consumption in the network.

4.2.2. Evaluating the EHDA method in terms of security requirements

In this section, we analyze EHDA (Ullah et al., 2020) according to the security requirements introduced in Section 2. Our purpose of this analysis is to determine what requirements are solved by this method.

- **Availability:** The EHDA method cannot guarantee availability. In fact, this scheme does not present an authentication mechanism to detect the invalid nodes. Therefore, it cannot counteract many attacks such as black hole, sinkhole, wormhole, and selective forwarding. However, it can deal with some attacks like Sybil and flooding.
- **Data confidentiality:** This scheme utilizes a symmetric key cryptography technique to secure data packets. Sensor nodes first encrypt their data using a unique key and then send it to the aggregator node. The decryption process of data packets is executed only on the FoG server, which is aware of the keys of all sensor nodes in the network. Therefore, if an attacker captures some sensor nodes, it cannot discover the keys of other nodes in the network. As a result, the EHDA method can guarantee data confidentiality and privacy.
- **Data integrity:** In Ullah et al. (2020), each sensor node calculates a hash value. This value is inserted into data packet to guarantee integrity. Upon receiving this data packet, aggregator nodes (or FoG server) recalculate a hash value to verify the data packet and remove the modified data packets. Therefore, it can be said that the EHDA method ensures data integrity.
- **Data freshness:** In this scheme, each sensor node inserts a timestamp into its data packet. Thus, if an aggregator node receives old data packets, it checks their timestamps to detect that they are obsolete, and finally delete them.
- **Privacy:** As mentioned earlier, a symmetric end-to-end encryption technique is presented in Ullah et al. (2020). This technique can guarantee privacy.

4.2.3. Countermeasures against various attacks

In this section, we express the attacks that EHDA (Ullah et al., 2020) can be able to detect and prevent. This analysis helps us to be aware of the security level of this method.

- **Eavesdropping:** The EHDA scheme utilizes a symmetric encryption technique to guarantee privacy and data confidentiality. As a result, if an attacker eavesdrops on communication links, it cannot find out the contents of data packets.
- **Traffic analysis:** In Ullah et al. (2020), sensor nodes encrypt their data using a unique key. Only the FoG server knows the encryption keys of the sensor nodes in the network. As a result, if an attacker analyzes network traffic, it cannot threaten privacy and data confidentiality. However, the attacker may obtain information such as node ID, location of aggregator nodes, FoG server location, and so on, and use this information to launch attacks such as sinkhole, black hole, sybil, etc.
- **Black hole:** In this scheme, aggregator nodes may use intermediate aggregator nodes to send their data packets to the FoG server. As a result, black hole nodes can persuade aggregator nodes to send their data packets through these malicious nodes. In the EHDA method, there is no authentication process to detect malicious nodes in the network. Thus, this method fails against a black hole attack. This is also true for sinkhole, wormhole, and selective forwarding attacks.

Table 10
Main features of the EHDA scheme.

Scheme	Network model	Network topology	Encryption technique	Weaknesses	Strengths
EHDA (Ullah et al., 2020)	Homogeneous	Cluster-based hierarchical	A symmetric key cryptography scheme	Not designing a mechanism for removing data redundancy, High communication overhead in neighboring nodes with the server FoG	Using an end-to-end encryption method, designing a mechanism for removing old data packets, using a symmetric encryption scheme

- **Sybil:** Mechanisms designed in the EHDA method can prevent such attacks. According to the MRA algorithm, each sensor node must insert a hash value in the data packet so that the aggregator nodes can detect fake data packets in the network. If a fake data packet is detected, the aggregator node rejects it. As a result, the EHDA scheme can counteract the sybil attack.
- **Flooding:** When aggregator nodes receive a data packet, they first check the timestamp inserted into this data packet to determine that it is duplicate or not. If the data packets are duplicate, the aggregator nodes reject them. Therefore, EHDA can counteract a flooding attack.
- **Node replication:** In the EHDA scheme, if such attack occurs, the attacker only can access the secret key of the compromised node and cannot capture other sensor nodes in the network. Therefore, this attack has a local effect on the network. However, it should be noted that if an aggregator node is compromised, then the attacker could disrupt network performance via deleting or modifying valid data packets received from other sensor nodes. If this compromised aggregator node is close to the FoG server, then this attack is more dangerous. In general, EHDA cannot present an appropriate solution for detecting and isolating compromised nodes.
- **Packet alteration and packet injection:** The EHDA method can counteract this attack because each aggregator node or FoG server can detect fake data packets or modified data packets via checking hash value inserted into the data packet.
- **Packet duplication:** In EHDA (Ullah et al., 2020), if an attacker replays the old data packets on the network, the aggregator node or FoG server checks timestamp inserted into the data packet to detect whether it is an outdated data packet or not.

4.3. ESRDA

Zhong et al. (2018) proposed an efficient and secure recoverable data aggregation (ESRDA) scheme, which combines a homomorphic encryption method with a signature scheme. In ESRDA, a heterogeneous WSN has been implemented. Also, the network topology includes a cluster-based hierarchical structure. Cluster member nodes communicate directly with the cluster head (CH) node. CHs are tasked to verify, aggregate, and send data packets of CM nodes to the base station. This method has five phases:

- **Setup phase:** In this phase, the base station generates system parameters, secret materials, and a key derivation function (KDF). Then, the base station loads these parameters into the memory of each sensor node.
- **Private key extraction phase:** When a sensor node (for example node i) wants to join the network, the base station generates several parameters, including a new identifier (ID_i), private key and secret key, and loads these parameters into memory of node i. It should be noted that the secret key is only known by sensor node i and the base station. After setting up the network, each cluster head node stores a list, including its CM nodes, called LCM_j, and sends it to the base station. In addition, the BS stores a list, including cluster head nodes, called LCH.
- **Encryption-signature phase:** When a CM node wants to transmit its data to the CH node, it first encodes its data, then generates a pseudo-random key and finally encrypts the encoded data using this key. Then, this node calculates an ID-based signature using its private key and a timestamp. Ultimately, the cluster member node sends its data packet, including encrypted data, generated signature, its ID, and timestamp, to the cluster head node.
- **Verification-aggregation-signature phase:** When a CH node receives a data packet from its CM nodes, it performs several steps: First, the CH node searches the LCM list to determine whether sender node is its cluster member or not. If CH does not find this identifier in its list, then it rejects this data packet. Otherwise, in the second step, the timestamp label is checked. If the timestamp is valid, then the CH node authenticates the data packet using batch signature verification. In the fourth step, if data is verified, the CH node aggregates the received data packets. Then, the CH node generates an ID-based signature using its private key and a timestamp. Eventually, the CH node forwards the data packet, including aggregated data, the generated signature, its ID, and a timestamp, to the base station.
- **Verification–decryption phase:** When BS receives a data packet from a cluster head node, it first searches the LCH list to verify that the sender node is valid. If its ID is not in this list, the base station rejects the data packet. Otherwise, in the next step, the validity of the timestamps are checked. If the base station verifies all timestamps, then BS must validate the received data via verifying the batch signature. If the signature based authentication is successful, then BS retrieves raw data using decryption keys.

4.3.1. Strengths and weaknesses

In this section, we introduce the most important strengths and weaknesses of ESRDA (Zhong et al., 2018). In addition, the main characteristics of this method have been listed in Table 11. In the following, some strengths of this method are expressed:

- The ESRDA method is scalable because sensor nodes are arranged in a cluster-based hierarchical structure. Thus, it is suitable for large-scale WSNs. It can reduce energy consumption and increase network lifetime.
- It uses an end-to-end encryption technique. This technique can reduce energy consumption, lower network delay, and improve network security.
- In Zhong et al. (2018), BS can recover all raw data. As a result, data aggregation operations are not limited by data aggregation functions.
- The ESRDA method can detect fake data packets and remove them.

Some disadvantages of the ESRDA method are:

- The ESRDA scheme has no mechanism for removing duplicate data. This may increase energy consumption and lower the network lifetime.
- In this scheme, all sensed data must be sent to the base station. As a result, the data packet size is increased in each hop. This reduces its scalability.

Table 11
Most important features of the ESRDA scheme.

Scheme	Network model	Network topology	Encryption technique	Weaknesses	Strengths
ESRDA (Zhong et al., 2018)	Heterogeneous	Cluster-based hierarchical	A symmetric key cryptography scheme	Not designing a mechanism for removing data redundancy, increasing size of data packets in each hop	Using an end-to-end encryption method, designing a mechanism for removing fake data packets, recovering all sensed data

4.3.2. Evaluating the ESRDA scheme in terms of security requirements

In this section, we assess ESRDA (Zhong et al., 2018) according to the security requirements defined in Section 2. This analysis helps us to determine what requirements have been solved by this method and there is what strategies to guarantee them.

- **Availability:** In Zhong et al. (2018), each sensor node calculates an ID-based signature and inserts it into its data packet. Upon receiving this data packet, the receiver node (aggregator node or base station) first checks the LCM and LCH lists to make sure that the sender node is valid. Then, it authenticates the sender node via verifying signature inserted into the data packet. Therefore, only valid data packets are received and others will be removed. In general, this method presents an appropriate authentication process that guarantees data availability.
- **Data confidentiality:** This method presents a symmetric homomorphic encryption method that can ensure data confidentiality successfully.
- **Data integrity:** In the ESRDA scheme, aggregator nodes or base station can detect and delete fake or modified data packets via verifying the signature inserted into the data packets. Therefore, it can be deduced that this method ensures data integrity.
- **Access control:** In Zhong et al. (2018), a signature-based authentication mechanism is proposed. As a result, only valid nodes participate in the data transmission process. Therefore, the ESRDA method guarantees access control.
- **Authentication:** In the ESRDA method, an authentication mechanism is presented to detect invalid nodes.
- **Data freshness:** In this scheme, each sensor node must insert a timestamp into its data packet. Thus, it can detect and remove old data packets.
- **Non-repudiation:** In the ESRDA scheme, each sensor node uses a signature in the authentication process. Also, upon receiving a data packet, cluster head nodes verify node ID to determine the validity of the sender nodes. In addition, the base station verifies the validity of CH nodes. Therefore, it can be deduced that non-repudiation is guaranteed in ESRDA.
- **Privacy:** As mentioned, this scheme uses a homomorphic end-to-end encryption technique to guarantee privacy.

4.3.3. Countermeasures against various attacks

In this section, we describe the attacks that ESRDA (Zhong et al., 2018) can detect and prevent. This analysis helps us to be aware of the security level of this method.

- **Eavesdropping:** In Zhong et al. (2018), before developing sensor nodes in the network, the base station loads the secret materials and the key derivation function (KDF) in their memory. Therefore, these encryption keys are only known by the corresponding sensor node and the base station. Sensor nodes apply this key to encrypt their data packets on the network. Therefore, an attacker does not have access to the encryption keys and cannot correctly interpret the data packets exchanged on the network by eavesdropping on the communication links.
- **Traffic analysis:** As mentioned, an attacker cannot discover encryption keys of sensor nodes. As a result, it cannot be able to find out the content of data packets exchanged in the network though analyzing its traffic.
- **Black hole, sinkhole, wormhole, and selective forwarding:** In the ESRDA scheme, when CH nodes receive data packet from a CM node, it first checks its LCM list before processing this data packet. As a result, the CH node can make sure that the CM node is valid. In addition, the CH node applies the batch signature verification to authenticate the sensor nodes. Thus, no attacker can send fake data packets to the CH node. On the other hand, after receiving a data packet from the CH node, the base station first check the LCH list, and if the CH node is valid then BS uses a signature verification method to verify this node. Therefore, in the data transmission process, the attackers cannot send fake data packets to the base station because it can detect malicious nodes and rejects data packets sent by them. In general, it is deduced that the ESRDA scheme can counteract such attacks.
- **Sybil:** In Zhong et al. (2018), it is impossible to occur such attack. According to this method, if a sensor node transmits a data packet to its CH node, it first controls the timestamp and ID of this node to make sure that the data packet is fresh and the sensor node is valid, respectively. Then, to validate the received data packet, the CH node verifies the signature inserted into the data packet. It should be noted that this signature was calculated based on a timestamp and the secret key of the sensor node. Therefore, if the attacker is not aware of this key, it cannot send data packets to the CH node successfully because the CH node can detect and remove fake data packets. There is a similar mechanism at the base station to verify the data packet sent by CH nodes. Therefore, an attacker cannot send a fake data packet to the base station.
- **Flooding:** In this scheme, if a sensor node receives duplicate data packets, it can detect and delete these data packets by controlling their timestamps. Therefore, ESRDA can counteract the flooding attack.
- **Node replication:** An attacker can launch such attack by capturing a sensor node. In Zhong et al. (2018), if a CM node is compromised, the attacker cannot disrupt the overall network performance. However, if a CH node is compromised, then this malicious node can make more problems, including removing all data packets in a cluster. However, the compromised CH node cannot send fake data packets (by modifying data packets of CM nodes) to the base station because it is not aware of the keys of its cluster member node. On the other hand, if this malicious node deletes all data packets received from other CH nodes, then the data integrity and availability will be threatened. In Zhong et al. (2018), the base station cannot prevent such conditions. Compromising CH nodes that are close to the base station, is more dangerous and can disrupt the security of the entire network.
- **Packet alteration and packet injection:** The ESRDA scheme has two countermeasures against such attacks. First, when receiving a data packet, the CH nodes (or BS) search the LCM (or LCH) list to determine that the sender node is valid. Second, the CH nodes (or BS) can detect modified data packets by checking the signature inserted in them. These countermeasures guarantee that

the ESRDA can counteract packet alteration and packet injection attacks.
- **Packet duplication:** In this data aggregation method, if an attacker replays an old data packet, the CH node (or BS) can counteract this attack because it examines timestamp inserted into the data packet to detect whether it is old or not.

4.4. SDAW

Boubiche et al. (2016) introduced the secure data aggregation watermarking-based scheme (SDAW) for homogeneous WSNs. In this method, the network topology is a cluster-based hierarchical structure. Sensor nodes sense data and send it to the CH node. CH nodes aggregate the received data packets and then forward the aggregated data packet directly (single-hop) to the base station. This scheme applies a lightweight and energy-efficient watermarking technique to secure the network. The SDAW scheme has two phases:

- **Intra-cluster data aggregation operations:** If a CM node wants to send its data to the CH node, it first generates a watermark using an embedding mechanism (Boubiche et al., 2016). Then, this watermark is inserted into the first 160 bits of the data packet. Next, raw data is added to the remaining space of the data packet. Then, the CM node transfers the data packet to the CH node via neighboring nodes (in a multi-hop manner). It should be noted that each next-hop node utilizes a detection and verification mechanism (Boubiche et al., 2016) to verify the accuracy of the watermark inserted into the received data packet. Next, this sensor node calculates a new watermark, replaces this new value with the previous watermark in the data packet, and sends the data to a new next-hop node. This process continues until this data packet reaches the CH node.
- **Inter-cluster data aggregation operations:** In this phase, the CH nodes must execute the inter-cluster data transmission process to send data packets received from their CM nodes directly (a single-hop manner) to the base station. In this regard, a CH node aggregates the data packets received from the CM nodes using an aggregation function, then generates a watermark and inserts it into the aggregated data packet. Next, the CH node forwards this data packet to the base station. Upon receiving this data packet, the base station extracts the inserted watermark and verify its validity. If the watermark is valid, the base station extracts data from the data packet. Otherwise, the base station rejects it.

4.4.1. Strengths and weaknesses

In this section, we express the most important strengths and weaknesses of SDAW (Boubiche et al., 2016). Furthermore, the main characteristics of this method have been summarized in Table 12. Some advantages of this method are:

- Sensor nodes are organized in a cluster-based hierarchical structure. As a result, the SDAW method is scalable. Therefore, it is an appropriate method for large-scale WSNs.
- This scheme uses a lightweight watermarking technique to secure the network. This technique can detect fake data packets and isolate malicious nodes.

In the following, the major drawbacks of the SDAW method are presented:

- In this method, the CH nodes communicate directly with the base station. Hence, if the distance between CH nodes and the base station is long, then they consume high energy for sending data packets to the base station. As a result, the network lifetime will be reduced. It can also threaten scalability.
- This scheme has a high memory overhead due to using a watermarking technique.
- SDAW increases delay and energy consumption in the data transmission process due to generating and verifying watermarks in each hop.

4.4.2. Evaluating the SDAW scheme in terms of security requirements

In this section, we evaluate SDAW (Boubiche et al., 2016) based on the security requirements defined in Section 2. This analysis helps us to determine what requirements have been solved by this method and there is what solutions to guarantee them.

- **Data confidentiality:** SDAW presents a lightweight watermarking scheme to guarantee data confidentiality. Each sensor node calculates a watermark that is appended to the data packet. Attackers are not aware of watermark and cannot interpret the content of the data packets correctly.
- **Data integrity:** Boubiche et al. propose an embedding mechanism for calculating watermark. As a result, each sensor node can detect and track any changes in data packets. Therefore, this method ensures the data integrity.
- **Privacy:** As mentioned, the SDAW method utilizes a lightweight watermarking scheme to protect privacy and data confidentiality.

4.4.3. Countermeasures against various attacks

In this section, we introduce the attacks that SDAW (Boubiche et al., 2016) can detect and prevent. This analysis helps us to be aware of the security level of this method.

- **Eavesdropping:** In this scheme, a watermark is generated based on several parameters, including the MAC address of a sensor node, XOR function, and a one-way hash function. An eavesdropper cannot discover these parameters. Therefore, it cannot interpret data packets exchanged in the network properly.
- **Traffic analysis:** If an attacker analyzes the network traffic and is aware of the presence of a watermark in the data packet, then it can extract the data from the data packet. This can threaten the data confidentiality. Boubiche et al. believe that an attacker does not have enough time to retrieve data because CH nodes are selected periodically and the cluster-based network structure is constantly changing. We believe that this argument is not very accurate because a watermark is added to the first 160 bits of the data packet, and if the attacker finds out this, then it can access the data.
- **Sybil:** This attack can be counteracted in the SDAW method. Accordingly, when a sensor node forwards a data packet to another sensor node, the receiver node can determine whether the sender node is valid or not, because it uses a detection and verification mechanism. Ultimately, the receiver node eliminates fake data packets.
- **Node replication:** In this scheme, if a sensor node (CM node or CH node) is captured, all its confidential information will be revealed such as the watermark generation mechanism and so on. An attacker can apply this compromised node to disrupt network performance. In the SDAW scheme, there is no mechanism to detect and remove this node from the network. As a result, the SDAW method is very vulnerable against this attack.
- **Packet alteration and packet injection:** In Boubiche et al. (2016), upon receiving a data packet, each sensor node (or base station) first verifies the accuracy of the data packet using the detection and verification mechanism. If the sender node is invalid then the received data packet will be rejected. Therefore, the SDAW method can deal with these attacks successfully.
- **Packet duplication:** In the SDAW scheme, if an attacker replays an old data packet, the sensor nodes (or base station) cannot detect it. Thus, it cannot counteract this attack. This attack can threaten data integrity, whereas the authors claim that their scheme could meet this security requirement.

Table 12
Most important features of the SDAW scheme.

Scheme	Network model	Network topology	Encryption technique	Weaknesses	Strengths
SDAW (Boubiche et al., 2016)	Homogeneous	Cluster-based hierarchical	A lightweight watermarking technique	Direct communication (single-hop) between the CH nodes and BS, high memory overhead, delay, and energy consumption due to generating and verifying watermark	Scalability, designing a mechanism for detecting fake data packets

4.5. Sign-share

Alghamdi et al. (2017) presented a secure data aggregation scheme called sign-share for homogeneous WSNs. The network topology is a cluster-based hierarchical structure. Each cluster has two aggregators. Aggregator nodes transmit data packets directly (a single-hop manner) to the base station. This scheme applies two techniques, including cryptography and digital signature, to secure the network. Each sensor node divides its data into several slices, then codes each slice, and finally sends each of them to an aggregator node in the cluster. This method includes four phases:

- **Setup phase:** In this phase, BS loads some system parameters, including a key set, pseudo-random binary sequence generator, public–private keys, and hash function into the memory of each sensor node.
- **Secret sharing-signature phase:** When a sensor node wants to forward its data to aggregator nodes, it performs several steps. First, the sensor node codes its data. Second, it divides this data into four slices and then encrypts each slice using its own key set. In the next step, the sensor node calculates a digital signature for each slice. This signature is appended to the corresponding data slice. Finally, the sensor node sends two data slices to one aggregator node and the other slices to another aggregator node in the cluster.
- **Aggregation phase:** Upon receiving all data packets from the CM nodes, the aggregator node aggregates the received data slices and transmits the aggregated data to the base station.
- **Verification and decryption phase:** When the base station receives data packets from the aggregator node, it first authenticates the data packets using the Boneh et al. algorithm. If the data packets are valid, the base station extracts the encrypted data from the data packets and decrypts them using the key set of each sensor node. Then, it merges the data slices to achieve the initial data.

4.5.1. Strengths and weaknesses

In this section, we demonstrate the most important strengths and weaknesses of sign-share (Alghamdi et al., 2017). Moreover, the main characteristics of this method have been summarized in Table 13. Some advantages of the sign-share scheme are presented as follows:

- This method has been designed for cluster-based hierarchical networks. This improves scalability and increases the network lifetime.
- It applies the end-to-end encryption scheme. Therefore, it reduces energy consumption, lowers end-to-end delay, and improves network security.
- In this scheme, the base station accesses all sensed data.

In the following, we present its major drawbacks:

- In this method, the aggregator nodes communicate directly with the base station. This threatens scalability because aggregator nodes consume high energy for communicating with BS.
- In Alghamdi et al. (2017), all sensed data must be sent to the base station. In this case, the size of the data packets is increased in each hop. It is not desirable for large-scale WSNs.
- In the Sign-share algorithm, the sensor nodes divide their data into several slices and forward a part of these data slices to the first aggregator node and another part to the second aggregator. As a result, in the data transmission process, if one of the aggregator nodes loses its data for reasons such as attackers, network congestion, and so on, then the data of another aggregator node will be inefficient. This is very undesirable in large-scale WSNs because if the base station asks the aggregator nodes to resend their data, then it increases the communication overhead, network congestion, and the packet loss rate (PLR).
- In the Sign-share method, authenticating sensor nodes and validating data packets is only done by the base station in a centralized manner.

4.5.2. Evaluating the sign-share scheme in terms of security requirements

In this section, we analyze Sign-share (Alghamdi et al., 2017) according to the security requirements introduced in Section 2. This analysis helps us to determine what requirements have been addressed by this method and there is what solutions to guarantee them.

- **Availability:** The Sign-share scheme can identify invalid nodes and prevent their hostile operations in the network because it utilizes a digital signature-based authentication mechanism. Thus, it guarantees availability.
- **Data confidentiality:** In the sign-share scheme, there are two techniques to guarantee data confidentiality: end-to-end encryption method and data slicing technique. As stated earlier, sensor nodes first split their data. Then, each data slice is encrypted and sent to one of the aggregator nodes. Aggregator nodes do not decrypt the data packets received from the CM nodes and is not aware of their encryption keys. They are tasked to aggregate the received data packets and send the aggregated data packet to the base station. Only the base station can decrypt the data received from the aggregator nodes and interpret its content. Note that if an aggregator node is captured, then the attacker node cannot access all data slices. Also, the data slices are encrypted and the attacker does not know the encryption keys to decrypt them. As a result, the attacker could not be informed of the contents of the data packets.
- **Data integrity:** The base station can detect the modified and fake data packets using digital signature inserted into them. As a result, data integrity is guaranteed.
- **Access control:** It ensures this security requirement using the digital signature-based authentication mechanism. Therefore, only valid nodes participate in the data transmission process.
- **Authentication:** In the Sign-share method, a digital signature-based authentication process has been proposed.
- **Non-repudiation:** This security requirement is met due to applying a digital signature.
- **Privacy:** Sign-share guarantees privacy using an end-to-end encryption method and the data slicing technique.

Table 13
Most important features of the Sign-share scheme.

Scheme	Network model	Network topology	Encryption technique	Weaknesses	Strengths
Sign-share (Alghamdi et al., 2017)	Homogeneous	Cluster-based hierarchical	An asymmetric key cryptography scheme	Direct communication (single-hop) between the aggregator nodes and BS, wasting the data of one aggregator node in case of losing the data of another aggregator, increasing data packet size in each hop, high communication overhead, using a centralized algorithm for the authentication process	Scalability, using an end-to-end encryption method, recovering all sensed data

4.5.3. Countermeasures against various attacks

In this section, we introduce the attacks that Sign-share (Alghamdi et al., 2017) can detect and prevent. This analysis helps us to be aware of the security level of this method.

- **Eavesdropping:** If an attacker eavesdrops on communication links, it cannot correctly infer the contents of the data packets. This has two reasons: (1) Using the data slicing technique and sending a part of the data slices (not all data slices) to aggregator nodes. (2) Encrypting data packets. Each sensor node first encrypts its data and then forwards the encrypted data packets in the network. Note that secret keys have been preloaded in memory of sensor nodes before setting up the network. Only the base station knows the secret keys of all sensor nodes. Thus, the attacker cannot correctly interpret data packets because it does not access the secret keys.
- **Traffic analysis:** The attacker cannot discover the contents of data packets by analyzing the network traffic. We explained its reasons in the eavesdropping attack. In general, this scheme can counteract this attack.
- **Black hole, sinkhole, wormhole and selective forwarding:** These attacks can be detected using two techniques, including the data slicing process and the digital signature. In this method, each sensor node divides its data into several slices, then encrypts and signs each slice, and finally transfers them to different aggregator nodes in the cluster. Assume that a black hole node deceives an aggregator node and removes its data packets. In this case, another aggregator node transmits its data packets to BS. When BS receives these data packets, it finds out that some data slices have not been sent. Therefore, the base station can detect such attacks. In addition, the digital signature inserted into data packets helps BS to detect and isolate invalid nodes.
- **Sybil:** The sign-share scheme can deal with such attacks because it uses the data slicing technique and the digital signature. However, there is a main drawback in the sign-share method: using a centralized authentication process. Hence, the aggregator nodes cannot detect fake data packets locally.
- **Node replication, packet alteration, and packet injection:** This scheme has a suitable solution to deal with such attacks: partitioning data of sensor nodes and sending data slices to different aggregator nodes in the cluster. As a result, if an aggregator node is compromised, the attacker has access to a subset of data of the CM nodes. Secondly, the base station has been equipped with an authentication mechanism and can detect the modified and fake data packets.

4.6. ASSDA

Hua et al. (2018) suggested an energy-efficient adaptive slice-based secure data aggregation (ASSDA) scheme for homogeneous WSNs. Sensor nodes are organized into a tree-based topology. In this method, a symmetric key cryptography method is applied to secure the data transmission process in the network. Moreover, it utilizes a hop-by-hop encryption technique. The ASSDA method has five phases:

- **Aggregation tree construction:** In this phase, an aggregation tree is built to organize the sensor nodes in the network. Note that the base station is located at the root of this tree. To build the aggregation tree, the base station broadcasts an initial message to its single-hop neighboring nodes. After receiving this message, neighboring nodes send back a join message to the base station. Then, the base station selects several nodes as its child nodes based on the signal strength of messages received from them. These nodes are known as aggregator nodes. This process continues until all sensor nodes select their parent nodes.
- **Determining the slicing number:** The ASSDA method determines several rules for sending data packets by sensor nodes:
  - All sensor nodes use only single-hop communications.
  - Each leaf node sends its data slices to its sibling node and the parent node.
  - Each sensor node sends its data slices in a time slot.
  - Sensor nodes cannot send and receive data simultaneously.
- **Determining the size of each slice:** In this phase, the leaf node divides its own data into a number of slices with different sizes. The size of each slice is calculated based on the distance between the sender node and the receiver nodes; if the distance between sender and receiver is short, the slice size is large. Otherwise, the slice size is small.
- **Mixing and assembling:** In this phase, each leaf node encrypts its data slices using a pairwise key shared with the receiver node. Then, the encrypted data slices are transmitted to the sibling node. After a certain time period, each sensor node decrypts the received data slices using a shared pairwise key and then aggregates these data slices with its own data. Finally, the aggregated data packet is sent to the parent node.
- **Aggregation:** Each sensor node calculates the total number of its data slices, then encrypts this value using a shared pairwise key, and sends it to the aggregator node. When the aggregator node receives all data packets from its child nodes, it decrypts them and then aggregates the received data. Finally, the aggregation result is sent to the base station via the aggregation tree.

4.6.1. Strengths and weaknesses

In this section, we state the most important strengths and weaknesses of ASSDA (Hua et al., 2018). Moreover, the main features of this method have been listed in Table 14. In the following, some advantages of the ASSDA scheme are presented:

- This method establishes an aggregation tree between sensor nodes. As a result, data transmission routes have been predetermined. Thus, there is no need to execute the route discovery process before sending data packets. This reduces delay.
- Using the data slicing technique maintains data privacy.
- In the data slicing process, each sensor node splits data into several slices with different sizes. Then, large-size data slices are transferred to near neighboring nodes and small-size data slices are transmitted to far neighboring nodes. This balances the energy consumption in the network.

Table 14
Most important features of the ASSDA scheme.

Scheme	Network model	Network topology	Encryption technique	Weaknesses	Strengths
ASSDA (Hua et al., 2018)	Homogeneous	Tree-based	A symmetric key cryptography scheme	Using a hop-by-hop encryption, low scalability, high communication overhead, high energy consumption, discharging the battery of sensor nodes close to the base station.	Creating an aggregation tree, guaranteeing privacy due to applying data slicing technique, generating data slices with different size

In the following, some drawbacks of the ASSDA method are mentioned:

- In this scheme, a hop-by-hop encryption technique is applied to secure data packets. This technique boosts delay and energy consumption in the data transmission process and provides low security level in the network.
- In the ASSDA method, all sensed data is sent to the base station. As a result, this threatens scalability because size of data packets is increased in each hop.
- Sensor nodes close to the base station have high communication overhead because they must send data packets of all sensor nodes to the BS. Therefore, they consume high energy.
- The data slicing mechanism has several drawbacks, including high communication overhead, large packet loss rate, high energy consumption, and heavy network congestion.

4.6.2. Evaluating the ASSDA scheme in terms of security requirements

In this section, we evaluate ASSDA (Hua et al., 2018) according to the security requirements introduced in Section 2. This analysis helps us to determine what requirements have been addressed by this method.

- **Data confidentiality:** ASSDA applies a hop-by-hop encryption technique to ensure data confidentiality. The sensor node encrypts its data slices using a pairwise key shared with the receiver node. Then, these data slices are transmitted to the receiver node. In addition, the data slicing technique presented in this method also guarantees data confidentiality. If an attacker captures a sensor node, it only access a subset of data slices of other sensor nodes.
- **Privacy:** The ASSDA method guarantees privacy by using a symmetric key cryptography technique and the data slicing method.

4.6.3. Countermeasures against various attacks

In this section, we introduce the attacks that ASSDA (Hua et al., 2018) can detect and prevent. This analysis helps us to be aware of the security level of this method.

- **Eavesdropping:** This scheme applies two techniques, including symmetric key cryptography and data slicing to counteract this attack. In this regard, the leaf nodes first split their own data, then encrypt their data slices, and finally transmit them to the desired sensor nodes. If an attacker eavesdrops on communication links, it only can achieve a subset of encrypted data slices and cannot interpret them correctly because it does not have access to entire data slices as well as it is not aware of secret keys.
- **Traffic analysis:** ASSDA can deal with this attack. We demonstrated some reasons for this in the eavesdropping attack. Assume that an attacker captures a sensor node. In this case, the attacker will only access pairwise keys shared between this node and its neighboring nodes and can decrypt data packets exchanged between them. However, this attacker cannot achieve the secret keys between other nodes. As a result, this attack has a local effect on the network.

4.7. OSM-EFHE

Shobana et al. (2020) introduced a SDA scheme called an optimized security model using enhanced fully homomorphic encryption (OSM-EFHE). This method has been designed for homogeneous WSNs. The network topology is a cluster-based hierarchical network. In the OSM-EFHE scheme, the size of the clusters is adjusted based on two parameters, including the distance between the CH nodes and the base station, and their energy. This data aggregation method includes four phases:

- **Clustering phase:** In this phase, a fuzzy rule-based clustering mechanism has been designed. The purpose of this mechanism is to balance energy consumption in the network. Obviously, CH nodes close to the base station have high communication overhead because they must receive and aggregate data packets from other CHs. Finally, they transmit aggregated results to the base station. This fuzzy system has two inputs, including the distance between the CH node and the base station, and the energy of the CH node. Its output is the cluster radius. In this regard, if a CH node is close to the base station and its energy is low, then its cluster radius is very small.
- **Data encryption and key generation phase:** In this phase, secret keys are generated using the DGHV public key compression technique (Shobana et al., 2020). Aggregator nodes do not have access to the content of the data packets because this scheme applies the EFHE method (Shobana et al., 2020). As a result, the data aggregation operation is performed on the encrypted data.
- **Secure data aggregation and integrity checking phase:** In this phase, an algorithm has been presented to detect fake data packets and verify data integrity. According to this algorithm, when the intermediate nodes receive a data packet in a cluster, they recalculate the SecMAC value and compare this value with the SecMAC value inserted into the data packet. If these two values are same, then data integrity is guaranteed. Aggregator nodes execute similar operations to verify data integrity in the inter-cluster data transmission process.
- **Decryption phase:** In this phase, when the base station receives the data packets, it decrypts the data packet and retrieves the original data.

4.7.1. Strengths and weaknesses

In this section, we introduce the most important strengths and weaknesses of OSM-EFHE (Shobana et al., 2020). Furthermore, the main features of this method have been listed in Table 15. In the following, some advantages of the OSM-EFHE scheme are expressed:

- This method has been designed for the cluster-based hierarchical network. Therefore, it improves network scalability and lowers energy consumption in the network.
- In Shobana et al. (2020), CH nodes, which are close to the BS, have small-size clusters. As a result, they consume less energy for the intra-cluster data aggregation process. Therefore, they can conserve more energy to do the inter-cluster data aggregation process. This balances energy consumption in the network.

Table 15
Most important features of the OSM-EFHE scheme.

Scheme	Network model	Network topology	Encryption technique	Weaknesses	Strengths
OSM-EFHE (Shobana et al., 2020)	Homogeneous	Cluster-based hierarchical	An asymmetric key cryptography scheme (homomorphic encryption)	Increasing the data packet size in each hop, not designing a mechanism for removing data redundancy	Scalability, adjusting the cluster radius based on distance and energy, applying an end-to-end encryption scheme

- It applies an end-to-end encryption technique that improves network security and lowers energy consumption in sensor nodes.

This SDA scheme has several drawbacks:

- In the OSM-EFHE scheme, all sensed data are transferred to the base station. Thus, the data packet size is increased in each hop. This reduces the network scalability.
- In dense networks, sensor nodes may overlap with each other and sense the same data. This increases communication overhead and energy consumption in the network. This scheme does not provide an efficient solution for this issue.

4.7.2. Evaluating the OSM-EFHE scheme in terms of security requirements

In this section, we evaluate OSM-EFHE (Shobana et al., 2020) according to the security requirements introduced in Section 2. This analysis helps us to determine what requirements have been addressed by this method and there is what solutions to guarantee them.

- **Data confidentiality:** This scheme applies a fully homomorphic encryption scheme called DGHV to guarantee data confidentiality. Sensor nodes exchange their encrypted data packet. Only the base station executes the decryption process. Attackers cannot access secret keys in the network and cannot interpret data packets correctly.
- **Data integrity:** This scheme presents a SecMAC-based algorithm to verify data integrity in the network.
- **Access control:** OSM-EFHE proposes the SecMAC-based authentication mechanism and the DGHV encryption technique to guarantee this security requirement. As a result, invalid nodes cannot collaborate in the data transmission process and forward fake data packets to the network.
- **Authentication:** In this scheme, a message authentication mechanism has been presented based on the SecMAC structure to guarantee data integrity.
- **Privacy:** As mentioned earlier, the OSM-EFHE scheme applies a fully homomorphic encryption technique called DGHV. It guarantees privacy.

4.7.3. Countermeasures against various attacks

In this section, we introduce the attacks that OSM-EFHE (Shobana et al., 2020) can detect and prevent. This analysis helps us to be aware of the security level of this method.

- **Eavesdropping:** This scheme can counteract the eavesdropping attack because it uses a homomorphic encryption technique. Therefore, an attacker cannot access the content of data packets exchanged on the network because the data packets have been encrypted and the decryption process is only executed by the base station.
- **Traffic analysis:** This attack can be counteracted via techniques discussed in the eavesdropping attack. If a sensor node is captured in the network, then the attacker only achieves its secret keys. Note that the attacker cannot compromise other sensor nodes using this node. On the other hand, only the BS has decryption keys. As a result, the attacker cannot interpret data packets of the compromised node in the network.
- **Sybil:** The OSM-EFHE scheme includes a SecMAC-based algorithm. This algorithm can detect fake data packets. As a result, the attacker cannot launch a Sybil attack to disrupt network performance. Based on this algorithm, each sensor node is to send a data packet to another sensor node, it calculates a SecMAC value and inserts it into its data packet (Refer to Shobana et al. (2020) for more details). Then, upon receiving the data packet, the receiver node first recalculates the SecMAC value and compares it with the SecMAC value in the data packet. If these values are not the same, the sensor node detects that the data packet is fake and deletes it.
- **Node replication:** In this scheme, if a sensor node (CH node or CM node) is compromised, it is impossible to detect this node or prevent its operation in the network. However, capturing a sensor node cannot disrupt the overall network performance (this node can only have a local effect on the network). This is because the compromised sensor nodes are not aware of the secret keys of the other nodes (only the BS and corresponding node know its secret keys). In general, it can be deduced that the OSM-EFHE method has good resistance against this attack.
- **Packet alteration and packet injection:** The OSM-EFHE scheme can counteract this attack using the SecMAC-based integrity analysis algorithm. We demonstrated this algorithm in Section 4.7.
- **Packet duplication:** This method cannot detect this attack. This is a drawback because Shobana et al. claim that their scheme can guarantee data integrity. Whereas, this attack threatens this security requirement.

4.8. SAPDA

Goyal et al. (2020) proposed the secure authentication and protected data aggregation (SAPDA) scheme for homogeneous underwater wireless sensor networks (UWSN). The sensor nodes are arranged in a cluster-based hierarchical topology. Gateway nodes are tasked to authenticate CH nodes to ensure that valid CH nodes manage the clusters. This method has two phases: secure authentication of CH nodes and secure data aggregation.

- **Secure authentication of CH nodes:** In this phase, it must be ensured that each cluster has a valid CH node. After clustering, CH nodes are authenticated by the gateway node (GW). In the following, we express the authentication process according to the algorithm introduced in Goyal et al. (2020):
  - The CH node generates a secret key and creates a registration request message, which includes its ID and the ID of the GW node.
  - In the next step, it calculates a hash value based on the request message and a timestamp, then signs this value using its own key, and finally transmits this message to the GW node.
  - After receiving this message, the GW node decrypts it using the public key of the CH node and extracts the request message and timestamp. Then, the GW node recalculates a hash value based on the request message and timestamp and compares it with the hash value inserted into the request message. If the two hash values are the same, the GW verifies the validity of the CH node.

  - In the last step, the GW node sends back a reply message to the CH node.

- **Protected data aggregation:** In this phase, a secure data aggregation algorithm has been presented. According to this algorithm, the GW node generates a unique symmetric key and sends it to each sensor node in the cluster. When a sensor node wants to send its data to the CH node, it first calculates an HMAC value based on its own data and a timestamp. Next, the sensor node generates an encrypted data packet, including its data, HMAC value, and its ID, and transmits it to the CH node. The CH is tasked to aggregate the data packets received from the CM nodes without decrypting them. Then, it sends the aggregated data packet to the base station. The BS decrypts this data packet using the secret keys and checks their timestamp. If data freshness is verified, then the base station checks data integrity using the HMAC value inserted into the data packet. If the data packet is valid, then the BS extracts the data; otherwise, it removes the fake data packets.

4.8.1. Strengths and weaknesses

In this section, we demonstrate the most important strengths and weaknesses of SAPDA (Goyal et al., 2020). In addition, the main features of this method have been presented in Table 16. In the following, some advantages of the SAPDA scheme are stated:

- In Goyal et al. (2020), the network topology is a cluster-based hierarchical network. It improves scalability and reduces energy consumption in the network.
- This method utilizes a symmetric end-to-end encryption technique that improves network security, conserves network resources, optimizes delay in the data transmission process due to not using encryption and decryption process in each hop.
- The SAPDA scheme can guarantee that the clusters are handled by valid CH nodes. This enhances network security.

In the following, we express some disadvantages of the SAPDA method:

- In this method, all sensed data is forwarded to the base station. Hence, it is not scalable because the size of the data packets are increased in each hop.
- In Goyal et al. (2020), CM nodes are not authenticated in a distributed manner. Therefore, a malicious node may be in a cluster. However, the BS can detect the adversary nodes using an HMAC-based authentication mechanism described in Section 4.8, other sensor nodes (CH or CM nodes) cannot identify them. As a result, sensor nodes must consume high energy to transfer data packets, whereas it is not clear whether they are valid or fake. This wastes network resources and increases communication overhead.

4.8.2. Evaluating the SAPDA scheme in terms of security requirements

In this section, we evaluate SAPDA (Goyal et al., 2020) according to the security requirements introduced in Section 2. This analysis helps us to determine what requirements have been addressed by this method and there is what solutions to guarantee them.

- **Availability:** In this scheme, after the clustering process, the GW node authenticates all CH nodes. Therefore, CM nodes send their data only to the authenticated CH nodes. In general, it is impossible to manage a cluster by a malicious node. On the other hand, the inter-cluster data transmission process is performed by trusted CH nodes. Thus, malicious nodes cannot disrupt this process. Also, a centralized integrity verification process is executed by BS. This process helps BS to detect malicious nodes in the network. As a result, the SAPDA method guarantees data availability.
- **Data confidentiality:** The GW node generates a unique key for each sensor node and loads it into its memory. Each sensor node only knows its own key and is not aware of the encryption keys of other nodes in the network. The base station is tasked to decrypt data packets and extract original data from them. If a sensor node (CH node or CM node) is compromised in the network, the attacker only has access to the secret key of the captured node. The attacker cannot interpret data packets of other nodes because it cannot achieve their key through a compromised node and cannot disrupt the network performance. Therefore, the SAPDA method guarantees data confidentiality.
- **Data integrity:** As stated in Section 4.8, this scheme has a secure data aggregation phase to ensure data integrity. According to this phase, whenever the sensor nodes want to forward their data packets to the CH node, they first calculate an HMAC value using the raw data and a timestamp. Next, they transmit the encrypted data packets to the CH node to reach the base station. After receiving these data packets, the base station executes two steps: (1) Checking data freshness using timestamp inserted into these data packets. (2) Verifying data integrity through recalculating the HMAC values and matching them with the values inserted into the data packets. Therefore, the base station can detect any modified data packets. Of course, this method has a major drawback: designing a centralized data integrity verification mechanism.
- **Access control:** It ensures access control because only authenticated CH nodes participate in the data transmission process to the base station.
- **Authentication:** In this scheme, GW node is tasked to validate CH nodes. Also, a message authentication mechanism has been proposed in the SAPDA scheme.
- **Data freshness:** In Goyal et al. (2020), each sensor node inserts a timestamp into its data packets. Therefore, the BS can detect old data packets. Thus, this security requirement has been guaranteed.
- **Non-repudiation:** This security requirement has been met due to applying a message authentication mechanism and validating CH nodes described in Section 4.8.
- **Privacy:** As stated earlier, the SAPDA method can protect privacy due to using a symmetric key cryptography method.

4.8.3. Countermeasures against various attacks

In this section, we introduce the attacks that SAPDA (Goyal et al., 2020) can detect and prevent. This analysis helps us to be aware of the security level of this method.

- **Eavesdropping:** The SAPDA scheme can counteract this attack. In this regard, when a CM node wants to send its data to the CH node, it first encrypts its data packets using a symmetric key. Then, the CM node transfers the encrypted data packet to the CH node. Similar operations are performed in the inter-cluster data transmission process to send data packets from a CH node to the BS. Therefore, if an eavesdropper listens to communication links, it cannot correctly interpret data packets exchanged on the network because this attacker does not know the secret keys of sensor nodes.
- **Traffic analysis:** This attack can be deactivated by the SAPDA scheme. We expressed some reasons for this in the eavesdropping attack. If an attacker captures a sensor node (CH node or CM node) in the network, it only achieves the secret key of this node. The compromised node can locally influence the network; however, this can be ignored because the attacker cannot capture other sensor nodes using this captured node and can achieve no keys of other nodes in the network. This because each sensor node only knows its own key. Therefore, the attacker cannot disrupt overall network performance via this attack.

Table 16
Most important features of the SAPDA scheme.

Scheme	Network model	Network topology	Encryption technique	Weaknesses	Strengths
SAPDA (Goyal et al., 2020)	Homogeneous	Cluster-based hierarchical	A symmetric key cryptography scheme	Increasing the data packet size in each hop, designing a centralized algorithm for verifying data integrity	Scalability, applying an end-to-end encryption scheme, managing clusters by valid CHs

- **Black hole, sinkhole, wormhole, and selective forwarding:** We believe that the SAPDA scheme can counteract these attacks. Based on this scheme, CM nodes send their data to the trusted CH nodes, which were authenticated by GW node according to an algorithm described in Section 4.8. In the inter-cluster data aggregation process, only the valid CH nodes can communicate with each other. In fact, a valid CH node aggregates the data packets received from its CM nodes and forwards the aggregated data packet to the base station through other valid CH nodes. If an attacker is located in this route, the CH nodes do not send any data packet to it because the attacker has not been authenticated by the GW node.
- **Sybil:** Based on the SAPDA scheme, the base station can detect any fake data packet. When a CM node is to send its data to the CH node, it first calculates an HMAC value using the original data and timestamp and inserts it in the data packet. Then, the CM node sends the encrypted data packet to the CH node. Next, the CH node aggregates all data packets received from CM nodes and transmits the aggregated data to the base station. BS is tasked to decrypt and verify the data packets. It validates the data packets by checking the HMAC value inserted in them. If a data packet is valid, the base station extracts the data. Otherwise, the base station rejects this data packet and the corresponding node is considered as a malicious node. This process ensures that the SAPDA method can neutralize the Sybil attack successfully.
- **Flooding:** According to this scheme, each sensor node inserts a timestamp into its data packets. Therefore, old data packets can be detected through checking timestamps. When BS receives the data packets, it checks the timestamp inserted into them. If the base station detects that a duplicate data packet has been replayed, it takes into account the corresponding node as a malicious node and isolates it in the network. This proves that the SAPDA method can deactivate this attack. However, it should be noted that if the timestamp checking process is performed as a distributed manner, the malicious node is identified rapidly and the communication overhead is reduced. Whereas, the SAPDA method does not design a distributed process.
- **Node replication:** This attack can be counteracted using a similar operation stated in the Sybil attack. Therefore, an attacker cannot execute this attack and disrupt network performance by sending fake data packets.
- **Packet alteration and packet injection:** Whenever the base station receives a data packet, it recalculates the HMAC value based on the original data and timestamp, and then matches it with the HMAC value in the data packet. If these two values are not the same, it means that the data packet has been modified or a fake packet has been injected into the network. As a result, the SAPDA scheme can detect these attacks.
- **Packet duplication:** The timestamp inserted into the data packets helps BS to detect duplicate data packets. Thus, this attack can be counteracted by this method.

4.9. EPDA

Zhou et al. (2019) suggested an energy-efficient and privacy-preserving data aggregation algorithm (EPDA) for homogeneous WSNs. Sensor nodes have been organized in a tree-based topology. The EPDA scheme applies a symmetric key cryptography called random key management technique to secure data packets. This method has three phases:

- **Tree establishment and optimization phase:** In this phase, an aggregation tree is established between the sensor nodes in the network. In this tree, the number of leaf nodes is minimized to reduce the communication overhead caused by the data slicing process performed by leaf nodes.
- **Slicing and mixing phase:** In this phase, to protect data privacy, each node divides its data into several slices. Then, the node stores one data slice. Next, it encrypts other data slices and sends them to its neighboring nodes. After a certain time interval, each sensor node deciphers the data slices received from its neighboring nodes and mixes them with its own data slice.
- **Aggregation phase:** In this phase, sensor nodes cipher the mixed data slices and transfers them to their parent node in the aggregation tree. Then, the parent node aggregates all data received from its child nodes. Ultimately, it encrypts and forwards the aggregated data to the base station through the aggregation tree.

4.9.1. Strengths and weaknesses

In this section, we introduce the most important strengths and weaknesses of EPDA (Zhou et al., 2019). In addition, the main features of this method have been summarized in Table 17. In the following, some advantages of the EPDA scheme are described:

- In Zhou et al. (2019), an aggregation tree is created between the sensor nodes. This tree has a minimum number of leaf nodes to reduce the communication overhead.

In the following, we demonstrate the most important disadvantages of the EPDA scheme:

- This method applies a hop-by-hop encryption technique. It increases the energy consumption and delay in the data transmission process.
- The tree creation process has a high communication overhead.
- The tree establishment process may cause an unbalanced energy distribution between the sensor nodes because a parent node may have many child nodes and consumes high energy, whereas other parent node has less child nodes and consumes less energy.
- The EPDA method does not provide a solution to detect/prevent sensing duplicate data.

4.9.2. Evaluating the EPDA scheme in terms of security requirements

In this section, we evaluate EPDA (Zhou et al., 2019) according to the security requirements introduced in Section 2. This analysis helps us to determine what requirements have been addressed by this method and there is what solutions to guarantee them.

- **Data confidentiality:** In Zhou et al. (2019), a symmetric key cryptography scheme called random key management technique has been applied to generate a shared pairwise key between two sensor nodes. Therefore, when a sensor node wants to transmit its data to the destination node, it first encrypts its original data using a pairwise key shared with the destination node and

Table 17
Most important features of the EPDA scheme.

Scheme	Network model	Network topology	Encryption technique	Weaknesses	Strengths
EPDA (Zhou et al., 2019)	Homogeneous	Tree-based	A symmetric key cryptography scheme (random key management technique)	Applying a hop-by-hop encryption scheme, high communication overhead in the tree creation process, unbalanced energy distribution, not designing a mechanism for removing data redundancy	Creating an aggregation tree with minimum leaf nodes for reducing communication overhead in the data slicing process

then forwards the encrypted data to it. This process ensures data confidentiality because attackers cannot interpret data packets exchanged on the network using eavesdropping the communication channels.
- **Privacy:** This scheme guarantees privacy using two techniques: random key management technique and data slicing process.

4.9.3. Countermeasures against various attacks

In this section, we demonstrate the attacks that EPDA (Zhou et al., 2019) can detect and prevent. This analysis helps us to be aware of the security level of this method.

- **Eavesdropping:** In this scheme, if an attacker eavesdrops on the wireless communication links, it cannot achieve data exchanged on them because communication between source node and destination node has been secured using a random key management technique.
- **Traffic analysis:** This attack has been counteracted using two schemes: (1) Using random key management for ciphering data packets, and (2) The data slicing process. We explained these schemes in Section 4.9 in detail. However, if an attacker captures a sensor node in the network, it can achieve the pairwise keys shared between compromised node and its neighboring nodes. This attacker can capture other nodes using the compromised node. This is a major drawback that has not been addressed in this scheme.

4.10. RCDA

Chen et al. (2011) presented a recoverable concealed data aggregation (RCDA) for cluster-based WSNs. This scheme has two versions namely, RCDA-HOMO (for homogeneous WSNs) and RCDA-HETE (for heterogeneous WSNs). In this scheme, the base station can retrieve all sensed data. In the following, both RCDA-HOMO and RCDA-HETE methods are described in detail.

RCDA-HOMO. This scheme has four phases: setup, encryption-signature, aggregation, and authentication.

- **Setup phase:** In this phase, the base station generates the secret keys and the hash function and loads them into the memory of each sensor node.
- **Encryption-signature phase:** In this phase, whenever a sensor wants to send its data to the CH node, it first encrypts its data and calculates a signature. Finally, the sensor node transmits a data packet, including the encrypted data and digital signature, to the CH node.
- **Aggregation phase:** When the CH node receives data packets from its CM nodes, it executes the aggregation process on the encrypted data packets and calculates an aggregated signature. Finally, the CH node transfers the aggregated data packet to the base station.
- **Authentication phase:** After receiving the aggregated data packets, the base station authenticates them. In this regard, the BS decrypts the aggregated data packets and then verifies them via checking the signature inserted in them. If the data packets are valid, the base station retrieves the original data. Otherwise, it rejects invalid data packets.

4.10.1. Strengths and weaknesses

In this section, we introduce the most important strengths and weaknesses of RCDA-HOMO (Chen et al., 2011). In addition, the main features of this method have been presented in Table 18. In the following, some benefits of the RCDA-HOMO scheme are stated:

- The network topology is a cluster-based hierarchical structure that improves scalability.
- In this scheme, an end-to-end encryption technique is applied. It enhances network security, reduces delay in the data transmission process, and lowers energy consumption.
- In the RCDA-HOMO scheme, the base station can achieve all sensed data. Hence, it can execute different aggregation operations on the raw data.

In the following, we state some disadvantages of RCDA-HOMO method:

- As mentioned earlier, in this scheme all sensed data is sent to the base station. As a result, the size of the data packet is enlarged in each hop. This can threaten network scalability.
- In the RCDA-HOMO scheme, a centralized data verification process has been presented. As a result, only the BS can validate sensor nodes in the network. Therefore, sensor nodes must send all received data packets while it is not clear whether these data packets are valid or not. This may cause high energy consumption in the data transmission process.

RCDA-HETE. This data aggregation method includes five phases: setup, intra-cluster encryption, inter-cluster encryption, aggregation, and authentication.

- **Setup phase:** In this phase, the base station generates the secret keys and some essential functions and loads them into the memory of sensor nodes. Also, each CM node shares a pairwise key with its CH node.
- **Intra-cluster encryption phase:** In this phase, a secure communication is established between the CM nodes and their corresponding CH node. When a CM node wants to send its data to the CH node, it first encrypts its data using a pairwise key shared with CH node, and transfers the encrypted data packets to the CH node. Upon receiving the data packet, the CH node decrypts it and extracts the raw data.
- **Inter-cluster encryption phase:** When the CH node receives all data packets from the CM nodes, it executes the desired aggregation operation on these data packets. Next, the CH node encrypts the aggregated data and calculates a signature. Eventually, this aggregated data packet is transmitted to neighboring CH nodes.

Table 18
Most important features of the RCDA-HOMO scheme.

Scheme	Network model	Network topology	Encryption technique	Weaknesses	Strengths
RCDA-HOMO (Chen et al., 2011)	Homogeneous	Cluster-based hierarchical	An asymmetric key cryptography scheme called elliptic curve ElGamal (EC-EG)	Enlarging data packet size in each hop, applying a centralized data verification process	Scalability, using an end-to-end encryption scheme, sending all sensed data to the BS

- **Aggregation phase:** When a CH node receives data packets from other CH nodes, it aggregates them. Then, the CH node encrypts the aggregated data and calculates an aggregated signature. Eventually, the CH nod forwards the aggregated data packet to its neighbors. This phase continues until data packets reach the base station.
- **Authentication phase:** After receiving the aggregated data packet, the base station must first authenticate it. Therefore, BS decrypts the data packet and then verifies the signature inserted in it. If the data verification process is successful, the base station extracts the data. Otherwise, the BS rejects it.

4.10.2. Strengths and weaknesses

In this section, we introduce the most important strengths and weaknesses of RCDA-HETE (Chen et al., 2011). In addition, the main features of this method have been presented in Table 19. In the following, some advantages of the RCDA-HETE scheme are stated:

- This scheme has been designed for cluster-based hierarchical networks. Therefore, the RCDA-HETE scheme is scalable.
- In the inter-cluster data aggregation process, CH nodes apply the end-to-end encryption, which improves network security and reduces delay and energy consumption.
- In this scheme, CH nodes have been tasked to execute the intra-cluster data aggregation process. Therefore, all sensed data is not sent to the base station. In fact, this scheme can manage the data packet size in each hop and improves scalability.

In the following, we express the most important disadvantage of the RCDA-HETE scheme:

- In this scheme, a centralized data verification process has been proposed. Thus, it has low security and wastes the network resources.

4.10.3. Evaluating the RCDA scheme in terms of security requirements

In this section, we evaluate RCDA (Chen et al., 2011) according to the security requirements introduced in Section 2. This analysis helps us to determine what requirements have been addressed by this method and there is what solutions to guarantee them.

- **Availability:** In the RCDA scheme, the base station authenticates the validity of the sensor nodes by checking the digital signature inserted into the data packets. However, it is a centralized mechanism so that sensor nodes are only verified by the base station.
- **Data confidentiality:** In the RCDA-HOMO scheme, the base station loads a private key into the memory of each sensor node. Sensor nodes encrypt data packets using this key. Then, these data packets are sent to the CH node. The CH node does not perform any decryption process. It only aggregates the data packets and then forwards the aggregated data packet to the BS in a multi-hop manner. When the base station receives the data packets, it decrypts them and extracts the raw data. In this process, secret key of each sensor node is only known by the base station and the corresponding node. Therefore, the RCDA-HOMO scheme guarantees data confidentially. In the RCDA-HETE method, the base station loads a private key into the memory of each CH node. Each CM node shares a symmetric pairwise key with the corresponding CH node. This key is applied to secure the intra-cluster data aggregation process. After receiving data packets, the CH node decrypts them and executes an aggregation operation on them. Then, the CH node encrypts the aggregated data packet using its private key. Next, the CH node sends it to the BS in a multi-hop manner. In fact, a CH node cannot decrypt data packet received from other CH. It is only tasked to forward this data packet to the BS. Only the base station is aware of the private keys of the CH nodes. Therefore, it decrypts the received data packets and extracts data. In general, the RCDA-HETE method guarantees data confidentiality.
- **Data Integrity:** The RCDA scheme ensures data integrity because it proposes a digital signature-based authentication mechanism.
- **Access control:** The proposed authentication mechanism can guarantee access control in the network.
- **Authentication:** In RCDA, a centralized digital signature-based authentication process has been designed. Please, refer to Section 4.10.
- **Non-repudiation:** This security requirement is guaranteed due to applying an appropriate authentication mechanism.
- **Privacy:** The RCDA ensures this security requirement. We explained some reasons for this in the data confidentiality.

4.10.4. Countermeasures against various attacks

In this section, we introduce the attacks that RCDA (Chen et al., 2011) can detect and prevent. This analysis helps us to be aware of the security level of this method.

- **Eavesdropping:** In the RCDA-HOMO scheme, the encryption keys have been preloaded into the memory of sensor nodes before setting up the network. These keys are only known by the BS and the corresponding node. In the intra-cluster data transmission process, a CM node must transfer its data to the CH node in an encrypted form. In the inter-cluster data transmission, a similar procedure is also executed. Only the BS knows decryption keys and can decrypt data packets exchanged on the network. Therefore, an attacker cannot interpret data packets through eavesdropping the wireless communication links. Overall, the RCDA-HOMO method can counteract this attack. On the other hand, in the RCDA-HETE scheme, in the intra-cluster data transmission process, each sensor node has a pairwise key shared with the CH node. It is applied to secure communications between them. Also, the BS has preloaded a private key into the memory of each CH node to secure inter-cluster communications. It can be deduced that in the RCDA-HETE scheme, the attackers cannot discover the content of the data packets exchanged in the network via listening to the communication channels.
- **Traffic analysis:** As stated in the eavesdropping attack, the encryption techniques applied in RCDA can deactivate this attack. In the RCDA-HOMO method, if a sensor node (CM node or CH node) is compromised, the attacker only achieves its cipher keys and cannot disrupt the overall network performance. On the other hand, in the RCDA-HETE scheme, if a CM node is compromised, it can influence the network locally. However, if a CH node is captured, the attacker achieves all pairwise key shared with its CM nodes and disrupts a cluster. However, it does not influence the entire network.

Table 19
Most important features of the RCDA-HETE scheme.

Scheme	Network model	Network topology	Encryption technique	Weaknesses	Strengths
RCDA-HETE (Chen et al., 2011)	Heterogeneous	Cluster-based hierarchical	A hybrid cryptography scheme (both symmetric and asymmetric key cryptography)	Applying a centralized data verification process	Scalability, using hybrid encryption (both end-to-end and hop-by-hop encryption), applying a local data aggregation process

- **Black hole, sinkhole, wormhole and selective forwarding:** The RCDA scheme presents an authentication mechanism based on the digital signature to protect the network against these attacks. However, this is a centralized mechanism. We believe that this mechanism cannot detect/prevent many attacks. Assume that there is a sinkhole node in the network. This malicious node removes all received data packets. In RCDA methods, there is no solution to detect this malicious node.
- **Sybil:** In the RCDA-HOMO scheme, each sensor node uses a hash function and its own private key to calculate a digital signature and insert it into the data packet. When the BS receives this data packet, its digital signature is checked to ensure that the data is valid. If the data is verified, the base station extracts the data. Otherwise, it rejects the data packet. This process ensures that the RCDA-HOMO method can counteract the Sybil attack. In the RCDA-HETE scheme, there is a similar process to validate data packets exchanged in inter-cluster communications. However, in the intra-cluster data transmission process, there is no mechanism to verify data packets. Therefore, a Sybil node may threaten the performance of a cluster. However, it influences the network performance locally. Therefore, it can be deduced that the RCDA-HETE scheme can counteract a Sybil attack.
- **Node replication, packet alteration and packet injection:** This attack can be deactivated using the digital signature-based authentication process proposed in RCDA methods because this mechanism can detect the modified and fake data packets and remove them.
- **Packet duplication:** In the RCDA method, there is no solution for detecting duplicate data packets. This is a drawback because Chen et al. claim that their proposed scheme can guarantee data integrity, whereas this attack threatens this security requirement.

4.11. QPPDA

Liu et al. (2020) introduced the queries privacy-preserving mechanism for data aggregation (QPPDA) in homogeneous WSNs. In this scheme, the network topology is a grid structure in which the sensor nodes are divided into a number of cells. These nodes sense data from an environment and send it to an aggregator node. The aggregator node has two tasks: (1) Responding to queries of the base station in a multi-hop manner, and (2) Aggregating the data received from the sensor nodes. In the QPPDA method, a homomorphic encryption scheme has been applied to ensure data confidentiality. This method includes three phases:

- **Grid division phase:** In this phase, the network is divided into several cells. Each sensor node can directly communicate with its neighboring cells. To create a grid topology, the base station first broadcasts its spatial coordinates and the length of each cell to all sensor nodes in the network. The QPPDA scheme proposes a grid division algorithm to calculate the cell coordinates based on the BS location and cell length information for each sensor node. After determining the grid structure, an aggregator node is selected for each cell. Then, an aggregation tree is established. The base station is considered as the root of this tree. Next, the cell member nodes transmit their data to the aggregator node. Finally, aggregator nodes forward the aggregated data to the base station through this tree.
- **Key generation phase:** In this phase, a homomorphic encryption technique is proposed to secure communication links. This technique has been originated from the elliptic curve scheme. In the data transmission process, different keys are applied to encrypt the data of sensor nodes. The number of private–public keys is equal to the number of queries supported by the network.
- **Query processing phase:** In this phase, the base station sends a query to the aggregator nodes. Then, they broadcast a message, including the query type, query epoch, and response time, to their cell member nodes. This process has four steps: (1) Data collection step. In this step, the cell member nodes collect the sensed data according to the received query. (2) Data encryption step. In this step, nodes encrypt the sensed data using a homophobic encryption technique. (3) Data aggregation step. In this step, aggregator nodes aggregate data received from its cell member nodes. (4) Data decryption step. In this step, the base station decrypts the data received from the aggregator nodes.

4.11.1. Strengths and weaknesses

In this section, we introduce the most important strengths and weaknesses of QPPDA (Liu et al., 2020). Also, Table 20 presents the main features of this method. In the following, some advantages of the QPPDA scheme are stated:

- In this scheme, the network topology is a hierarchical network. It improves energy consumption and scalability in the network.
- In Liu et al. (2020), an end-to-end encryption technique has been proposed. This technique reduces delay in the data transmission process and improves the energy consumption in the network.
- In the grid division phase, an aggregation tree is created between aggregator nodes. This tree is applied for sending data packets to the base station. This means that data transmission routes have been predetermined. This reduces the end-to-end delay in the data transmission process.

In the following, we introduce the main weaknesses of the QPPDA scheme:

- In this method, aggregator nodes close to the base station have high communication overhead because they must aggregate the data received from their cell member nodes, also aggregate the data received from other aggregator nodes, and send this data to the base station. As a result, they consume high energy and die rapidly. The QPPDA method does not provide a solution to solve this problem. After the death of these aggregator nodes, communications between other aggregator nodes and the base station will be disconnected and ultimately the network will be disabled.
- In Liu et al. (2020), the key generation process has high computational overhead.
- In large-scale networks, if sensor nodes overlap with each other, then they sense similar data. This wastes resources on the network. The QPPDA scheme does not provide a solution to address this problem.

Table 20
Most important features of the QPPDA scheme.

Scheme	Network model	Network topology	Encryption technique	Weaknesses	Strengths
QPPDA (Liu et al., 2020)	Homogeneous	Grid	An asymmetric cryptography scheme (a homomorphic encryption technique)	High computational overhead, not designing a mechanism for removing data redundancy, high communication overhead in aggregator nodes close to the BS	Scalability, using an end-to-end encryption scheme, creating an aggregation tree between aggregator nodes

4.11.2. Evaluating the QPPDA scheme in terms of security requirements

In this section, we evaluate QPPDA (Liu et al., 2020) according to the security requirements introduced in Section 2. This analysis helps us to determine what requirements have been addressed by this method and there is what solutions to guarantee them.

- **Data confidentiality:** In this method, a homomorphic encryption technique has been presented to guarantee data confidentiality. In data transmission process, various keys are applied to secure wireless communication links. Decryption keys are only known by the base station. Therefore, an attacker cannot discover the content of data packets through capturing sensor nodes (cell member node or aggregator node). As a result, it ensures data confidentiality.
- **Privacy:** As mentioned earlier, the homomorphic encryption technique proposed in the QPPDA scheme can ensure privacy and data confidentiality.

4.11.3. Countermeasures against various attacks

In this section, we introduce the attacks that QPPDA (Liu et al., 2020) can detect and prevent. This analysis helps us to be aware of the security level of this method.

- **Eavesdropping:** The elliptic curve-based encryption technique can secure messages exchanged on the network. As a result, if an attacker listens to the communication links, it cannot interpret data packets correctly. Therefore, the QPPDA scheme can deactivate this attack.
- **Traffic analysis:** This attack can be counteracted via the homomorphic encryption technique. If an attacker captures a sensor node in the network, it cannot disrupt the overall network performance because it cannot achieve encryption keys of other sensor nodes. On the other hand, it does not access the decryption keys because they have only been saved in the BS.

4.12. LSDAR

Haseeb et al. (2020) offered the lightweight structure based data aggregation routing (LSDAR) protocol for homogeneous WSNs. The sensor nodes have been organized in a cluster-based hierarchical topology. In this protocol, clusters have different sizes. The size of each cluster is calculated based on the distance between CH nodes and the base station. Clusters close to the base station have a small size whereas clusters, which have a long distance to the BS, have a large size. In this method, data security is ensured using the XOR encryption function. The LSDAR method includes three phases:

- **Initial topology construction:** In this phase, the base station first broadcasts an advertisement message, including its ID and location. Upon receiving this message, each sensor node calculates its distance to BS based on the RSSI index. This distance is used for determining the cluster radius. In a cluster, node with the most energy is selected as the CH node. Next, the CH node broadcasts an advertisement message on the network. Upon receiving this message, ordinary sensor nodes join the nearest CH node via sending a join message.
- **Routing tree creation:** In this phase, the A-star algorithm is applied to construct a routing tree. In this tree, the best node is selected as the next step node. In this scheme, an objective function has been proposed to select the optimal node. This objective function has two parameters: RSSI and the residual energy of the desired node.
- **Data security:** In this phase, the base station generates t random keys and sends them to each CH node. If CH nodes are close to the BS, they encrypt their data and transmit their encrypted data packets to the BS in a single hop manner. In contrast, if the distance between a CH node and the BS is high, it encrypts its data using the XOR function and sends its encrypted data packet to the BS via routing tree. Finally, the decryption operation of the data packets is only executed by the BS.

4.12.1. Weaknesses and strengths

In this section, we present the most important strengths and weaknesses of LSDAR (Haseeb et al., 2020). Furthermore, Table 21 summarizes the main features of this method. In the following, some advantages of the LSDAR scheme are stated:

- This protocol utilizes a cluster-based hierarchical topology. Thus, LSDAR is scalable.
- In Haseeb et al. (2020), a routing tree is created between sensor nodes. Therefore, data transmission routes have been predetermined and there is no need for discovering route before the data transmission process. As a result, the end-to-end delay is reduced in this process.
- In this method, the network is divided into clusters with different sizes. This balances energy consumption on the network because clusters close to the base station are smaller and their CH nodes consume less energy for intra-cluster communications. As a result, these nodes can consume more energy for the inter-cluster data transmission process.
- In the LSDAR scheme, all sensed data is sent to the base station.
- In this method, an end-to-end encryption technique is used. In fact, only the base station executes the decryption process. This reduces energy consumption and delay in the data transmission process.

In the following, we express some weaknesses of the LSDAR method:

- The LSDAR scheme cannot eliminate data redundancy.
- In this method, all sensed data is sent to the BS. Thus, the size of the data packets is increased in each hop. This reduces the scalability and increases the end-to-end delay in the data transmission process.
- In the LSDAR scheme, the tree construction process is done dynamically. This increases the communication overhead.

Table 21
Most important features of the LSDAR scheme.

Scheme	Network model	Network topology	Encryption technique	Weaknesses	Strengths
LSDAR (Haseeb et al., 2020)	Homogeneous	Cluster-based hierarchical	A symmetric cryptography scheme (XOR function)	Not designing a mechanism for removing data redundancy, high communication overhead, increasing data packet size in each hop	Scalability, designing an aggregation tree for sending data to BS, using an end-to-end encryption scheme, sending all sensed data to the BS, balancing energy consumption on the network

4.12.2. Evaluating the LSDAR scheme in terms of security requirements

In this section, we analyze LSDAR (Haseeb et al., 2020) according to the security requirements introduced in Section 2. This analysis helps us to determine what requirements have been addressed by this method and there is what solutions to guarantee them.

- **Data confidentiality:** The LSDAR protocol utilizes a symmetric key cryptography method to ensure data security. Although this method ensures data confidentiality, network performance may be dropped over time because the encryption keys are fixed. Thus, attackers can capture some sensor nodes and reveal their encryption keys. Considering the rekeying process or improving the encryption process can efficiently increase network security in this scheme.
- **Privacy:** The LSDAR method ensures this security requirement using the XOR-based encryption method.

4.12.3. Countermeasures against various attacks

In this section, we introduce the attacks that LSDAR (Haseeb et al., 2020) can detect and prevent. This analysis helps us to be aware of the security level of this method.

- **Eavesdropping:** In this method, a symmetric encryption technique is presented to secure communication links in the data transmission process. Each sensor node first encrypts its data and then sends the encrypted data packet to the next-hop node. As a result, the attacker cannot properly interpret the data packets exchanged on the network without having the encryption keys.
- **Traffic analysis:** As mentioned, this attack can be counteracted using the encryption technique presented in the LSDAR method. However, if an attacker captures a sensor node in the network, it can compromise all sensor nodes and reveal their encryption keys. As a result, all network will be disabled. It can be deduced that this method has low resilience.

4.13. SDAPA

Parmar and Kadhiwala (2016) proposed the secure data aggregation protocol using AES (SDAPA) for homogeneous WSNs. This scheme utilizes the AES symmetric encryption algorithm to secure the data aggregation process. Network model is a tree-based topology. SDAPA applies a hop-by-hop encryption technique and includes four phases:

- **Bootstrapping:** In this phase, the BS loads an initial key into the memory of the sensor nodes. This key is used to secure the key distribution process and is removed from the memory of sensor nodes after bootstrapping the network.
- **Tree construction:** In this phase, an aggregation tree is built between the sensor nodes. The base station broadcasts a tree beacon message to start the tree creation process. Upon receiving this message, sensor nodes send back a tree join request message to the BS. If a node successfully joins this tree, the base station forwards a tree join success message to it. This process continues until the aggregation tree is established between all sensor nodes in the network.
- **Key establishment:** In the aggregation tree, each node creates two keys: a pairwise key shared with its parent node and a pairwise key shared with its grandparent node. To establish these keys, the sensor node first transmits a key exchange message, including a nonce value, to the parent node (or grandparent node). Upon receiving this message, the parent node (or grandparent node) calculates a pairwise key shared with this node. Next, the parent node (or grandparent node) transfers its nonce value to the sensor node for calculating this shared key.
- **Data aggregation:** When a sensor node wants to transmit its data, it must perform two data transmission processes: (1) Sending its data packets to the parent node, and (2) Sending its data packet to the grandparent node. It should be noted that a MAC value is calculated using the pairwise key shared with the parent node (or grandparent node) and is inserted into the data packet. Upon receiving the data packet, the parent node first checks the MAC value inserted in it. If the data packet is valid, the parent node decrypts it. Then, the parent node aggregates the data received from the child nodes and sends the aggregated data packets to its parent node. On the other hand, when the grandparent node receives the data packets from the child node and the grandchild node, it first verifies the validity of the sender nodes using the MAC inserted in the data packets. If they are valid, then it decrypts the data packets. The grandparent node compares the data received from the grandchild node with the aggregated data received from the child node. If these values are not the same, the grandparent node rejects the data packets and sends a warning message to the child nodes to retrieve the data correctly. Ultimately, the parent node also removes the data packet received from its child node to complete the data recovery process.

4.13.1. Strengths and weaknesses

In this section, we present the most important strengths and weaknesses of SDAPA (Parmar and Kadhiwala, 2016). Furthermore, Table 22 summarizes the main features of this method. In the following, some advantages of the SDAPA scheme are introduced:

- In this method, the hop-by-hop authentication process is executed. As a result, if there is a malicious node in the network, it can be quickly removed from the network. This can helps sensor nodes to conserve their resources.
- In Parmar and Kadhiwala (2016), an aggregation tree is created between the sensor nodes in the network. Therefore, the data transmission routes have been predetermined. Thus, it reduces the end-to-end delay in the data transmission process.

In the following, we express some disadvantages of this method:

- In Parmar and Kadhiwala (2016), nodes close to the base station (i.e. sensor nodes located in the upper level of the aggregation tree) consume a lot of energy because they have a high communication overhead.

Table 22
Most important features of the SDAPA scheme.

Scheme	Network model	Network topology	Encryption technique	Weaknesses	Strengths
SDAPA (Parmar and Kadhiwala, 2016)	Homogeneous	Cluster-based	A symmetric cryptography scheme based on AES algorithm	High communication overhead, low scalability, applying a hop-by-hop encryption technique, not designing a mechanism for removing data redundancy	Designing an aggregation tree for sending data to the BS, presenting a hop-by-hop authentication mechanism

- In this scheme, each sensor node transmits its data packets to two nodes (i.e. the parent node and the grandparent node). It increases the communication overhead in the network.
- The SDAPA scheme applies a hop-by-hop encryption technique that increases the end-to-end delay and energy consumption in the data transmission process and reduces the network lifetime.
- In the dense networks, if sensor nodes overlap with each other, then they may sense the same data. It increases data redundancy and wastes resources in the network. This method does not provide any solution for this issue.

4.13.2. Evaluating the SDAPA scheme in terms of security requirements

In this section, we evaluate SDAPA (Parmar and Kadhiwala, 2016) according to the security requirements introduced in Section 2. This analysis helps us to determine what requirements have been addressed by this method and there is what solutions to guarantee them.

- **Availability:** SDAPA proposes an authentication process based on message authentication code (MAC) to authenticate the data packets sent over the network. This process guarantees data availability.
- **Data confidentiality:** In the SDAPA scheme, a symmetric encryption technique based on the AES algorithm has been presented to ensure data confidentiality.
- **Data integrity:** The MAC-based authentication mechanism guarantees data integrity. On the other hand, each sensor node transmits its data packets to two nodes, namely parent node and grandparent node. As a result, if an attacker modifies the data packets, the grandparent node can detect and remove these data packets by matching the data packets received from the child node and the grandchild node. Therefore, this scheme guarantees data integrity.
- **Access control:** Access control is guaranteed due to applying the MAC-based authentication mechanism.
- **Authentication:** In the SDAPA scheme, a MAC-based authentication mechanism has been provided to authenticate the sensor nodes.
- **Data freshness:** In this method, a timestamp is inserted into the data packets. Thus, it guarantees data freshness.
- **Non-repudiation:** This security requirement has been guaranteed due to the authentication mechanism and the data transmission process presented in this scheme.
- **Privacy:** This security requirement has been ensured using an AES-based encryption mechanism.

4.13.3. Countermeasures against various attacks

In this section, we introduce the attacks that SDAPA (Parmar and Kadhiwala, 2016) can detect and prevent. This analysis helps us to be aware of the security level of this method.

- **Eavesdropping:** As mentioned earlier, each sensor node first encrypts its data using an encryption key shared with the parent node (or grandparent node). Then, it transmits the encrypted data packet to the parent node (or grandparent node). Therefore, all data packets are exchanged on communication links in an encrypted form. Thus, if an attacker listens to wireless communication channels, it cannot discover the contents of the data packets because it does not have access to the encryption keys. As a result, the SDAPA method can counteract this attack.
- **Traffic analysis:** This attack can be deactivated using the encryption technique introduced in this scheme. If a sensor node is compromised, the attacker can achieve all keys stored in its memory, including the pairwise key shared with the parent node, the key shared with the grandparent, the keys shared with the child nodes, and the keys shared with the grandchild nodes. As a result, some secret information is revealed on the network. It should be noted that the sensor nodes use pairwise keys. Thus, the attacker node cannot capture other nodes via the compromised node and does not access the keys of other sensor nodes in the network. It can be deduced that capturing a sensor node has a local effect on the network performance and cannot threaten the entire network.
- **Black hole, sinkhole, wormhole and selective forwarding:** Each sensor node sends its encrypted data packets to two nodes, namely parent node and grandparent node. These data packets include a message authentication code (MAC). When the parent node (or grandparent node) receives a data packet, it checks the MAC value inserted into this data packet to authenticate the sender node. On the other hand, the grandparent node can detect any change in the data packets received from the grandchild nodes by comparing them with the data packet received from its child nodes. Assume that the parent node is a black hole node and removes all data packets received from its child nodes. In this case, the grandparent node can detect this attack because it receives data packets from the grandchild nodes directly, and executes a data recovery process.
- **Sybil:** This attack can be counteracted using the MAC-based authentication process in the SDAPA scheme. Thus, any malicious node cannot send fake data packets. If a node receives a data packet, it first checks the validity of this data packet using the MAC value inserted in it. Therefore, a Sybil node cannot launch this attack.
- **Flooding:** Each sensor node can detect duplicate data packets via checking timestamp inserted in them. Therefore, this method can successfully deal with the flooding attack.
- **Node replication:** In this attack, the attacker captures a sensor node and attempts to disrupt network performance using this compromised node. It can locally disrupt network performance. However, since each node sends its data packets to two nodes (parent node and grandparent node), the grandparent node can detect the compromised node by matching the data received from the child node and the grandchild node. Therefore, it can be deduced that this scheme can counteract the node replication attack.
- **Packet alteration and packet injection:** In this method, each sensor node can detect fake or modified data packets via verifying the MAC value inserted into them. As a result, these attacks can be deactivated by the SDAPA scheme.

- **Packet duplication:** Timestamp inserted into the data packets is an effective solution for detecting old data packets. Thus, this method can deal with the packet duplication attack.

4.14. LIPDA

Zhao et al. (2016) introduced a lightweight and integrity-protecting oriented data aggregation (LIPDA) scheme in homogeneous WSNs. Sensor nodes have been arranged in a cluster-based hierarchical structure. The LIPDA method applies the additive homomorphic encryption technique to secure the data packets. This method has six phases:

- **CH selection process:** Obviously, if the distance between sensor nodes increases, then the energy consumed for sending/receiving data packets will also increase. In this phase, a distance-based cluster selection protocol has been proposed to balance the energy consumption in the CH nodes. Based on this, each sensor node calculates a probability, which indicates its chance for being CH. This probability is based on the distance between a node and its neighboring nodes. The received signal strength index (RSSI) is applied to calculate the distance.
- **Formation of tree-cluster network topology:** In this phase, the sink node first broadcasts a Hello message. Upon receiving this message, each sensor node calculates its chance to be selected as a CH. After determining the CH node, it selects the sender node of the Hello message as its parent node. Then, the CH node replays the Hello message to determine other CH nodes and their child nodes. This process continues until the tree-cluster topology is completed.
- **Key generation and distribution:** In this phase, the RC4 encryption algorithm is applied to secure the key distribution process. The sink node is tasked to generate the cluster key and send it to the corresponding CH node. CM nodes utilize this key to encrypt their data and forward the encrypted data packets to the CH node. It should be noted that the additive homomorphic encryption is applied to secure the data aggregation process, please refer to Zhao et al. (2016) for more details. The CH node also aggregates the received data packets and sends them to the sink node. After receiving the data packets, the sink node decrypts them.
- **Data aggregation:** This phase has two steps: intra-cluster data aggregation and inter-cluster data aggregation. In the intra-cluster data aggregation, each sensor node frames its data into a complex number structure, which is a combination of encrypted data and privacy factor. Then, the sensor node sends this data packet to the CH node. In the inter-cluster data aggregation step, the CH node aggregates all data packets received from CM nodes and its own data. Then, the CH node transmits the aggregated data packet to its parent node in the aggregation tree. This process continues until the data reaches the sink node.
- **Integrity verification:** When the sink node receives data packets, it first decrypts them. As mentioned, each data packet includes two parameters: the encrypted data and the privacy factor used for message authentication. After decrypting the data packet, the sink node recalculates the privacy factor and compares it with the value inserted into the data packet. If these two values are the same, then this data packet is verified; otherwise, it is rejected.
- **Dynamic cluster adjustment:** When a CH node dies, its CM nodes cannot send their data. One solution is to adjust clusters dynamically. In the LIPDA scheme, when a CH node has less energy than a certain threshold, the network topology will be reset.

4.14.1. Weaknesses and strengths

In this section, we state the most important strengths and weaknesses of LIPDA (Zhao et al., 2016). Furthermore, Table 23 summarizes the main features of this method. In the following, some advantages of the LIPDA scheme are introduced:

- This scheme applies the end-to-end encryption technique, which improves security, reduces end-to-end delay and energy consumption in the network.
- In Zhao et al. (2016), an aggregation tree is created between CH nodes. Thus, the data transmission routes have been predetermined. As a result, there is no need to discover routes between CH nodes during transferring data packets.
- In the LIPDA scheme, a dynamic topology adjustment process is taken into account. Thus, if the CH nodes die, communications between the sensor nodes are not disturbed.
- This method is scalable due to applying a hierarchical topology.

In the following, we describe the most important weaknesses of the LIPDA method:

- The LIPDA scheme does not provide any solution to eliminate data redundancy.
- The message authentication process is only executed by the sink node. Therefore, other sensor nodes must send all received data packets while it is not clear whether these data packets are valid or not. This can increase energy consumption and congestion in the network.

4.14.2. Evaluating the LIPDA scheme in terms of security requirements

In this section, we evaluate LIPDA (Zhao et al., 2016) according to the security requirements introduced in Section 2. This analysis helps us to determine what requirements have been addressed by this method and there is what solutions to guarantee them.

- **Data confidentiality:** The LIPDA encryption mechanism has two steps: (1) RC4 encryption algorithm to protect the key distribution process (2) Additive homomorphic encryption to protect the aggregated data. All data packets are exchanged on the network in an encrypted form. Therefore, an attacker cannot interpret them. Thus, this scheme ensures data confidentiality.
- **Data integrity:** In the LIPDA method, the sink node is tasked to check data integrity. If the sink node finds out that the data packets have been modified, it rejects them. This is executed via recalculating the privacy factor and comparing it with the value inserted into the data packets. Therefore, the LIPDA scheme guarantees data integrity.
- **Authentication:** In this method, the sink node performs a message authentication process.
- **Privacy:** This security requirement is guaranteed using two encryption algorithms, namely RC4 and additive homomorphic.

4.14.3. Countermeasures against various attacks

In this section, we introduce the attacks that LIPDA (Zhao et al., 2016) can detect and prevent. This analysis helps us to be aware of the security level of this method.

- **Eavesdropping:** This attack can be counteracted using the additive homomorphic encryption scheme and the RC4 algorithm. In this scheme, the key distribution process is secured by the RC4 encryption. Also, all data packets are encrypted and the decryption process is only executed by the sink node. Therefore, if an attacker eavesdrops on communication links, it cannot discover the contents of data packets exchanged on the networks.
- **Traffic analysis:** There are three techniques to deal with this attack: (1) Using the RC4 algorithm to protect keys generated, (2) Applying the additive homomorphic encryption to encrypt the sensed data. (3) Using a complex number structure that

Table 23
Most important features of the LIPDA scheme.

Scheme	Network model	Network topology	Encryption technique	Weaknesses	Strengths
LIPDA (Zhao et al., 2016)	Homogeneous	Cluster-tree hierarchical	A symmetric cryptography scheme based on RC4 algorithm, and an additive homomorphic encryption technique	Not designing a mechanism for removing data redundancy, applying a centralized message authentication mechanism	Using an end-to-end encryption technique, selecting CH nodes dynamically, designing an aggregation tree for sending data to the sink node, scalability

includes two parameters, including the data and privacy factor. If an attacker captures a sensor node in the network, it cannot compromise other nodes via the captured node because attacker cannot achieve their private keys.

- **Sybil:** The LIPDA scheme can deactivate this attack. An attacker requires the private key and ID of a sensor node to inject fake data packets into the network. However, the key distribution process is secured by the RC4 algorithm. Therefore, it is not simple to discover the generated keys. Also, the sink node can detect all fake data packets by checking privacy factor.
- **Node replication:** In this attack, the attacker captures some sensor nodes to disrupt the network performance. In this scheme, each sensor node has a private key that is only known by the sink node and the corresponding node. Therefore, it is not simple for the attacker to capture other sensor nodes via the compromised node. Therefore, capturing a sensor node has a local effect on the network.
- **Packet alteration and packet injection:** This attack can be counteracted because if an attacker modifies the aggregated data packets or injects fake data packets into the network, the sink node can detect and remove invalid data packets using the integrity verification process introduced in the LIPDA scheme.

4.15. CSDA

Fang et al. (2019) suggested a cluster-based private data aggregation (CSDA) scheme for homogeneous WSNs. The network model is a tree-cluster hierarchical structure. The CSDA scheme uses the random pairwise key encryption technique to ensure data security. If two neighboring nodes share a same key, they can communicate with each other directly. Otherwise, they communicate with each other in a multi-hop manner. In this method, the data slicing technique has been applied to protect data privacy.

- **Clustering phase:** In this phase, the sensor nodes are arranged in several clusters. To establish a cluster, each node broadcasts a Hello message to its neighbors. Upon receiving this message, each sensor node calculates a probability to be selected as a CH. Then, the CH nodes rebroadcast the Hello message. In addition, non-CH nodes send a request message to join a cluster. Eventually, the network is clustered and an aggregation tree is created between the CH nodes.
- **Intra-cluster data aggregation phase:** In this phase, a data slicing technique is used. Assume that a cluster has m CM nodes. Thus, each CM node divides its data into m slices and encrypts $m-1$ data slices using pairwise keys shared with its neighboring nodes and sends the encrypted data packets to them. Upon receiving a data slice, each sensor node decrypts it and aggregates this data slice with its own data. Finally, the sensor node broadcasts the aggregated data to reach the CH node.
- **Inter-cluster data aggregation phase:** In this phase, the CH node aggregates the data packets received from the CM nodes with its data. Finally, the CH node forwards the aggregated data packet to its parent node in the aggregation tree.

4.15.1. Strengths and weaknesses

In this section, we demonstrate the most important strengths and weaknesses of CSDA (Fang et al., 2019). Furthermore, Table 24 lists the main features of this method. In the following, some advantages of the CSDA scheme are presented:

- This method is scalable and improves energy consumption in the network due to applying a tree-cluster hierarchical topology.
- In Fang et al. (2019), an aggregation tree is created between the CH nodes. Therefore, the data transmission routes are predetermined. This reduces end-to-end delay in the data transmission process.

In the following, we describe some disadvantages of the CSDA method:

- In this scheme, CH nodes close to BS have high communication overhead and consume a lot of energy.
- The CSDA scheme has high communication overhead due to applying the data slicing technique.
- This method uses a hop-by-hop encryption technique that increases energy consumption and delay in the data transmission process.

4.15.2. Evaluating the CSDA scheme in terms of security requirements

In this section, we evaluate CSDA (Fang et al., 2019) according to the security requirements introduced in Section 2. This analysis helps us to determine what requirements have been addressed by this method and there is what solutions to guarantee them.

- **Data confidentiality:** The CSDA scheme uses a symmetric key cryptography method called random pairwise key to ensure data security. If two sensor nodes can share a secret key, they can communicate with each other directly and use this key to encrypt their data packets. Otherwise, they must use a path key to communicate securely with each other. Therefore, this method ensures data confidentiality.
- **Privacy:** This security requirement can be guaranteed using two techniques: (1) Applying a random pairwise key-based encryption method and (2) Utilizing the data slicing technique. It is not simple for attackers to decrypt data packets because they must capture all sensor nodes in the network to achieve the encryption keys. On the other hand, if attackers can capture some sensor nodes, they will only obtain a subset of original data because each sensor node uses the data slicing technique and sends its data slices to the different nodes.

4.15.3. Countermeasures against various attacks

In this section, we introduce the attacks that CSDA (Fang et al., 2019) can detect and prevent. This analysis helps us to be aware of the security level of this method.

- **Eavesdropping:** This attack can be counteracted because the CSDA scheme uses a random pairwise key-based symmetric encryption method to ensure security in the data transmission process. Therefore, all data packets are exchanged on the network in an encrypted form. As a result, an attacker cannot interpret data

Table 24
Most important features of the CSDA scheme.

Scheme	Network model	Network topology	Encryption technique	Weaknesses	Strengths
CSDA (Fang et al., 2019)	Homogeneous	Cluster-tree hierarchical	A symmetric cryptography scheme (random pairwise key encryption)	High communication overhead in sensor nodes close to BS, high communication overhead due to using data slicing technique, using hop-by-hop encryption scheme	Designing an aggregation tree for sending data to the sink node, scalability

packets through listening to the wireless communication links because it does not access the encryption keys that the BS has preloaded into the memory of sensor nodes.

- **Traffic analysis:** To deactivate this attack, two solutions have been proposed in this method, namely the random pairwise key technique and the data-slicing scheme. If a sensor node is compromised in the network, all pairwise keys shared with its neighboring nodes will be revealed. However, the attacker node cannot discover communications between other sensor nodes via this node because each sensor node only knows its encryption keys. On the other hand, each sensor node only receives a subset of data of its neighboring nodes. Therefore, the attacker cannot access the original data. Overall, this attack has a local effect on the network.

5. Discussion

In this section, according to the SDA schemes discussed in Section 4, it can be deduced that key cryptography techniques are the most common solution to guarantee data confidentiality. Based on the studies conducted in this paper, we found out that some SDA methods utilize symmetric key cryptography techniques. For example, EHDA (Ullah et al., 2020), ASSDA (Hua et al., 2018), SAPDA (Goyal et al., 2020), EPDA (Zhou et al., 2019), LSDAR (Haseeb et al., 2020), SDAPA (Parmar and Kadhiwala, 2016), and CSDA (Fang et al., 2019). It should be noted that this technique has several advantages such as low energy consumption, simpler algorithm, less computational and communication overhead. However, its security level is lower than that in asymmetric key cryptography technique. On the other hand, some SDA schemes like MODA (Zhang et al., 2018), Sign-share (Alghamdi et al., 2017), OSM-EFHE (Shobana et al., 2020), RCDA-HOMO (Chen et al., 2011), and QPPDA (Liu et al., 2020), also apply asymmetric key cryptography techniques to secure data packets. Compared to symmetric key cryptography technique, this technique provides a robust security level. However, it has high energy consumption and more computational and communication overhead. Hence, this technique is not suitable for WSNs with limited resources. In addition, some SDA approaches use hybrid key cryptography techniques (i.e. both symmetric and asymmetric methods) to secure data packets. For example, ESRDA (Zhong et al., 2018), RCDA-HETE (Chen et al., 2011), and LIPDA (Zhao et al., 2016) schemes. They seek to take advantage of both symmetric and asymmetric key cryptography techniques and reduce their disadvantages. In Table 25, the SDA methods are compared in terms of the key cryptography technique.

Obviously, designing security mechanisms reduces network lifetime because they increase energy consumption in WSN. For this reason, researchers are trying to make a tradeoff between security and network lifetime. To improve network lifetime in WSNs, one solution is to use a suitable topology to organize sensor nodes in the network. For this reason, most researchers apply hierarchical (cluster-based) topology in SDA schemes, such as EHDA (Ullah et al., 2020), ESRDA (Zhong et al., 2018), SDAW (Boubiche et al., 2016), Sign-share (Alghamdi et al., 2017), OSM-EFHE (Shobana et al., 2020), SAPDA (Goyal et al., 2020), RCDA (Chen et al., 2011), LSDAR (Haseeb et al., 2020), and SDAPA (Parmar and Kadhiwala, 2016). In this topology, the energy consumption of cluster member nodes is drastically reduced. Because they send their data only to the CH node, which is in a short distance from them. On the other hand, CH nodes are responsible for aggregating data packets received from cluster member nodes and sending the aggregated data to the BS. However, CH nodes have high overhead and die quickly. In RCDA-HETE (Chen et al., 2011), the network model is heterogeneous and CH nodes are selected from sensor nodes with more energy and higher processing power. This scheme presents promising results. High-energy nodes have more robust hardware than other nodes. Hence, they are expensive and their number is less than other nodes in the network. Therefore, a serious research challenge is to determine accurate and appropriate location of these nodes so that other nodes can access them. It should be noted that hierarchical networks have poor performance for dynamic environments, which include mobile sensor nodes; Because the network topology is constantly changing. Therefore, a future research direction is to propose SDA schemes for dynamic environments. Based on our studies in this survey, some secure data aggregation methods such as MODA (Zhang et al., 2018), ASSDA (Hua et al., 2018), and EPDA (Zhou et al., 2019), use tree-based topology to organize sensor nodes in the network. The main drawback of this topology is its low scalability. In this topology, fixed routes are used to transmit data. This can increase the packet loss rate. In SDA schemes, an attractive idea is to use a hybrid tree-cluster-based topology, which decreases the drawbacks of both tree-based topology and cluster-based topology. LIPDA (Zhao et al., 2016) and CSDA (Fang et al., 2019) have used tree-cluster based topology. In Tables 26 and 27, secure data aggregation schemes are classified according to the network model and network topology, respectively.

In SDA methods, encryption technique also has a critical effect on network performance and efficiency in terms of energy consumption, delay, etc. According to our studies in this paper, most secure data aggregation schemes apply the end-to-end encryption method. For example MODA (Zhang et al., 2018), EHDA (Ullah et al., 2020), ESRDA (Zhong et al., 2018), Sign-share (Alghamdi et al., 2017), OSM-EFHE (Shobana et al., 2020), SAPDA (Goyal et al., 2020), RCDA-HOMO (Chen et al., 2011), QPPDA (Liu et al., 2020), LSDAR (Haseeb et al., 2020), and LIPDA (Zhao et al., 2016). In these methods, the most important advantage is that they can efficiently protect privacy. Because the intermediate nodes cannot access the contents of the data packets of other nodes in the network; and the data aggregation process is performed on the encrypted data. The end-to-end encryption also has other benefits such as enhancing security and reducing delay in the data transmission process and reducing energy consumption of sensor nodes. Whereas, ASSDA (Hua et al., 2018), EPDA (Zhou et al., 2019), SDAPA (Parmar and Kadhiwala, 2016), CSDA (Fang et al., 2019) schemes have used the hop-by-hop encryption, which has a weak performance in protecting privacy and data confidentiality. Because, the encryption and decryption processes of data is executed at each hop in these methods. As a result, the intermediate nodes are informed about the contents of the data packets of other nodes in the network. Therefore, if the intermediate nodes are compromised, the data of other sensor nodes will also be revealed. In Table 28, SDA schemes are compared in terms of encryption technique.

Table 25
Classification of SDA schemes based on key cryptography technique.

Number	SDA scheme		Symmetric key cryptography	Asymmetric key cryptography
1	MODA (Zhang et al., 2018)		×	✓
2	EHDA (Ullah et al., 2020)		✓	×
3	ESRDA (Zhong et al., 2018)		✓	×
4	SDAW (Boubiche et al., 2016)		×	×
5	Sign-share (Alghamdi et al., 2017)		×	✓
6	ASSDA (Hua et al., 2018)		✓	×
7	OSM-EFHE (Shobana et al., 2020)		×	✓
8	SAPDA (Goyal et al., 2020)		✓	×
9	EPDA (Zhou et al., 2019)		✓	×
10	RCDA (Chen et al., 2011)	RCDA-HOMO	×	✓
		RCDA-HETE	×	×
11	QPPDA (Liu et al., 2020)		×	✓
12	LSDAR (Haseeb et al., 2020)		✓	×
13	SDAPA (Parmar and Kadhiwala, 2016)		✓	×
14	LIPDA (Zhao et al., 2016)		✓	×
15	CSDA (Fang et al., 2019)		✓	×

Table 26
Classification of SDA schemes based on network model.

Number	SDA scheme		Homogeneous	Heterogeneous
1	MODA (Zhang et al., 2018)		✓	×
2	EHDA (Ullah et al., 2020)		✓	×
3	ESRDA (Zhong et al., 2018)		✓	×
4	SDAW (Boubiche et al., 2016)		✓	×
5	Sign-share (Alghamdi et al., 2017)		✓	×
6	ASSDA (Hua et al., 2018)		✓	×
7	OSM-EFHE (Shobana et al., 2020)		✓	×
8	SAPDA (Goyal et al., 2020)		✓	×
9	EPDA (Zhou et al., 2019)		✓	×
10	RCDA (Chen et al., 2011)	RCDA-HOMO	✓	×
		RCDA-HETE	×	✓
11	QPPDA (Liu et al., 2020)		✓	×
12	LSDAR (Haseeb et al., 2020)		✓	×
13	SDAPA (Parmar and Kadhiwala, 2016)		✓	×
14	LIPDA (Zhao et al., 2016)		✓	×
15	CSDA (Fang et al., 2019)		✓	×

Table 27
Comparison of SDA schemes in terms of network topology.

Number	SDA scheme		Flat topology	Cluster-based topology	Tree-based topology	Tree-cluster based topology
1	MODA (Zhang et al., 2018)		×	×	✓	×
2	EHDA (Ullah et al., 2020)		×	✓	×	×
3	ESRDA (Zhong et al., 2018)		×	✓	×	×
4	SDAW (Boubiche et al., 2016)		×	✓	×	×
5	Sign-share (Alghamdi et al., 2017)		×	✓	×	×
6	ASSDA (Hua et al., 2018)		×	×	✓	×
7	OSM-EFHE (Shobana et al., 2020)		×	✓	×	×
8	SAPDA (Goyal et al., 2020)		×	✓	×	×
9	EPDA (Zhou et al., 2019)		×	×	✓	×
10	RCDA (Chen et al., 2011)	RCDA-HOMO	×	✓	×	×
		RCDA-HETE	×	✓	×	×
11	QPPDA (Liu et al., 2020)		×	×	×	×
12	LSDAR (Haseeb et al., 2020)		×	✓	×	×
13	SDAPA (Parmar and Kadhiwala, 2016)		×	✓	×	×
14	LIPDA (Zhao et al., 2016)		×	×	×	✓
15	CSDA (Fang et al., 2019)		×	×	×	✓

Table 28
Classification of SDA schemes based on encryption technique.

Number	SDA scheme		End-to-end encryption	Hop-by-hop encryption
1	MODA (Zhang et al., 2018)		✓	×
2	EHDA (Ullah et al., 2020)		✓	×
3	ESRDA (Zhong et al., 2018)		✓	×
4	SDAW (Boubiche et al., 2016)		×	×
5	Sign-share (Alghamdi et al., 2017)		✓	×
6	ASSDA (Hua et al., 2018)		×	✓
7	OSM-EFHE (Shobana et al., 2020)		✓	×
8	SAPDA (Goyal et al., 2020)		✓	×
9	EPDA (Zhou et al., 2019)		×	✓
10	RCDA (Chen et al., 2011)	RCDA-HOMO	✓	×
		RCDA-HETE	✓	✓
11	QPPDA (Liu et al., 2020)		✓	×
12	LSDAR (Haseeb et al., 2020)		✓	×
13	SDAPA (Parmar and Kadhiwala, 2016)		×	✓
14	LIPDA (Zhao et al., 2016)		✓	×
15	CSDA (Fang et al., 2019)		×	✓

Table 29
Comparison of different SDA schemes in terms of security requirements.

Number	Scheme	Security requirements							
		Availability	Data confidentiality	Data integrity	Access control	Authentication	Data freshness	Non-repudiation	Privacy
1	MODA (Zhang et al., 2018)	×	✓	×	×	×	×	×	✓
2	EHDA (Ullah et al., 2020)	×	✓	✓	×	×	✓	×	✓
3	ESRDA (Zhong et al., 2018)	✓	✓	✓	✓	✓	✓	✓	✓
4	SDAW (Boubiche et al., 2016)	×	✓	✓	×	×	×	×	✓
5	Sign-share (Alghamdi et al., 2017)	✓	✓	✓	✓	✓	×	✓	✓
6	ASSDA (Hua et al., 2018)	×	✓	×	×	×	×	×	✓
7	OSM-EFHE (Shobana et al., 2020)	×	✓	✓	✓	✓	×	×	✓
8	SAPDA (Goyal et al., 2020)	✓	✓	✓	✓	✓	✓	✓	✓
9	EPDA (Zhou et al., 2019)	×	✓	×	×	×	×	×	✓
10	RCDA (Chen et al., 2011)	✓	✓	✓	✓	✓	×	✓	✓
11	QPPDA (Liu et al., 2020)	×	✓	×	×	×	×	×	✓
12	LSDAR (Haseeb et al., 2020)	×	✓	×	×	×	×	×	✓
13	SDAPA (Parmar and Kadhiwala, 2016)	✓	✓	✓	✓	✓	✓	✓	✓
14	LIPDA (Zhao et al., 2016)	×	✓	✓	×	✓	×	×	✓
15	CSDA (Fang et al., 2019)	×	✓	×	×	×	×	×	✓

When designing SDA methods in WSNs, the security requirements should be considered based on needs of an application. In this paper, we divide SDA schemes based on applications into two categories, namely low-risk application and high-risk application. Accordingly, if a secure data aggregation method only considers data confidentiality and privacy, it is suitable for low-risk applications. For example, MODA (Zhang et al., 2018), SDAW (Boubiche et al., 2016), ASSDA (Hua et al., 2018), EPDA (Zhou et al., 2019), QPPDA (Liu et al., 2020), LSDAR (Haseeb et al., 2020), and CSDA (Fang et al., 2019). It is not recommended that these SDA schemes be applied to critical applications such as military, healthcare, IoT, IIoT, etc., because they are vulnerable to many attacks. In contrast, EHDA (Ullah et al., 2020), ESRDA (Zhong et al., 2018), Sign-share (Alghamdi et al., 2017), OSM-EFHE (Shobana et al., 2020), SAPDA (Goyal et al., 2020), RCDA (Chen et al., 2011), SDAPA (Parmar and Kadhiwala, 2016), and LIPDA (Zhao et al., 2016) are suitable for high-risk applications, because they meet different security requirements and are resistant to many attacks on the network. Table 29 compares secure data aggregation schemes in terms of various security requirements. It should be noted that a detailed analysis of these methods was presented in Section 4.

As shown in Table 29, almost all SDA schemes guarantee data confidentiality and privacy. Eavesdropping and traffic analysis are the most important attacks that threaten these security requirements. Table 30 analyzes SDA methods in terms of these attacks briefly. As shown in Table 30, SDAW (Boubiche et al., 2016) cannot counteract the traffic analysis attack because this method applies an inefficient encryption technique to protect data confidentiality. Therefore, the attacker can analyze the data packets exchanged on the network to discover the watermark inserted in them and access their content.

As stated in Section 4, some SDA methods apply the data slicing technique to protect data privacy. For example, Sign-share (Alghamdi et al., 2017), ASSDA (Hua et al., 2018), EPDA (Zhou et al., 2019), and CSDA (Fang et al., 2019). The purpose of this technique is to guarantee data privacy. According to the data slicing technique, each sensor node divides its original data into several data slices, and then the data slices are sent to different aggregator nodes. Now, if an attacker captures an aggregator node, it can only access a subset of the data of the other nodes. It is true that this technique is an effective solution to ensure data privacy, but it contradicts the main goal of the data aggregation process. This purpose is to reduce data transmission to decrease energy consumption, congestion, traffic, packet collision, and delay in the

Table 30
Analyzing different SDA methods in terms of attacks related to data confidentiality.

Number	Scheme	Attacks related to data confidentiality	
		Eavesdropping	Traffic analysis
1	MODA (Zhang et al., 2018)	✓	✓
2	EHDA (Ullah et al., 2020)	✓	✓
3	ESRDA (Zhong et al., 2018)	✓	✓
4	SDAW (Boubiche et al., 2016)	✓	×
5	Sign-share (Alghamdi et al., 2017)	✓	✓
6	ASSDA (Hua et al., 2018)	✓	✓
7	OSM-EFHE (Shobana et al., 2020)	✓	✓
8	SAPDA (Goyal et al., 2020)	✓	✓
9	EPDA (Zhou et al., 2019)	✓	✓
10	RCDA (Chen et al., 2011)	✓	✓
11	QPPDA (Liu et al., 2020)	✓	✓
12	LSDAR (Haseeb et al., 2020)	✓	✓
13	SDAPA (Parmar and Kadhiwala, 2016)	✓	✓
14	LIPDA (Zhao et al., 2016)	✓	✓
15	CSDA (Fang et al., 2019)	✓	✓

network. However, if the data slicing technique is used, not only the number of data transmission is not decreased, but it is also increased. As a result, it is impossible to employ this technique for large-scale WSNs. Therefore, researchers must study in this field to provide more effective techniques for data privacy.

It should be noted that many SDA approaches provide an appropriate authentication technique, which guarantees data integrity and availability. Table 31 compares SDA methods in terms of attacks related to availability. Table 32 also presents a comparison between SDA schemes in terms of attacks related to data integrity. Also, secure data aggregation schemes are categorized based on application in Table 33.

In some applications, authentication is an essential requirement. This is because an attacker can disrupt the data aggregation process by altering data packets or injecting fake data packets into the network. This can change the aggregation result. According to our proposed classification in this paper, SDA schemes can apply two end-to-end and hop-by-hop authentication mechanisms. For example, Sign-share (Alghamdi et al., 2017), SAPDA (Goyal et al., 2020), RCDA-HOMO (Chen et al., 2011), and LIPDA (Zhao et al., 2016) use the end-to-end authentication mechanism. The most important benefits of this mechanism are low computational overhead, energy consumption, and end-to-end delay in the data transmission process. However, its major drawback is low security that may waste network resources. On the other hand, ESRDA (Zhong et al., 2018), SDAPA (Parmar and Kadhiwala, 2016), OSM-EFHE (Shobana et al., 2020), EHDA (Ullah et al., 2020), and SDAW (Boubiche et al., 2016) apply the hop-by-hop authentication mechanism in the data aggregation process. This mechanism increases the computational overhead, energy consumption, and end-to-end delay in the data transmission process. Whereas, it provides better security. This can be a future research direction for researchers to provide an appropriate solution, which makes a tradeoff between security and delay. For example, RCDA-HETE (Chen et al., 2011) presented a proper technique so that the BS authenticates CH nodes (end-to-end authentication mechanism). Furthermore, CH nodes locally authenticate their CM nodes (hop-by-hop authentication mechanism). This method provides an appropriate tradeoff between security and delay. It is an effective authentication mechanism for large-scale WSNs. However, it requires further research to address its weaknesses. In Table 34, different SDA schemes are categorized in terms of the authentication mechanism.

Based on our studies in this paper, it can be deduced that EHDA (Ullah et al., 2020), ESRDA (Zhong et al., 2018), Sign-share (Alghamdi et al., 2017), OSM-EFHE (Shobana et al., 2020), SAPDA (Goyal et al., 2020), RCDA-HOMO (Chen et al., 2011), LSDAR (Haseeb et al., 2020), and SDAPA (Parmar and Kadhiwala, 2016) are recoverable SDA schemes. The most important benefit of these methods is that the BS has access to all sensed data and can perform various data aggregation operations on raw data. However, these methods are not scalable. Because when transmitting data to the BS, data packet size is enlarged in each hop. This increases energy consumption and delay in the data transmission process. On the other hand, SDAW (Boubiche et al., 2016), ASSDA (Hua et al., 2018), EPDA (Zhou et al., 2019), RCDA-HETE (Chen et al., 2011), QPPDA (Liu et al., 2020), LIPDA (Zhao et al., 2016), and CSDA (Fang et al., 2019) are unrecoverable SDA schemes. These methods are scalable and reduce delay in the data transmission process. In MODA (Zhang et al., 2018), the authors have introduced an interesting idea called multi-functional data aggregation technique. According to the technique proposed in this method, each sensor node transforms its own original data into well-defined vectors. Then, the data aggregation process is performed on these vectors. This preserves value, order, and context of the original data. In this condition, the base station is able to execute various statistical operations on the aggregated data. Refer to Section 4 for more details. In fact, MODA (Zhang et al., 2018) offered a hybrid scheme (i.e. both recoverable and unrecoverable techniques). It seeks to take advantages of both categories and addresses their drawbacks. However, in the future, researchers must study this method to complete this idea and address its weaknesses. In Table 35, SDA schemes are classified based on the data recovery ability.

6. Open issues and research challenges

Secure data aggregation is a very important research subject. It is expected that this subject will be improved by researchers in the future. In the wireless sensor network, there are various challenges for providing an appropriate SDA method. In this section, we present the most important challenges and open issues in this area:

Restricted resources. Sensor nodes have limited resources such as processing power, communication range, memory, energy, and so on. As a result, designing a light-weight and energy-efficient SDA scheme is a very important research area that should be considered by researchers.

Scalability. The SDA scheme should be suitable for networks with different sizes. Obviously, when the size of the network increases, the delay will be increased in the data transmission process from sensor nodes to the base station (BS). This can undermine the performance of the SDA scheme. Therefore, designing the scalable SDA schemes is an important challenge.

Table 31
Comparisons between SDA methods in terms of attacks related to availability.

Number	Scheme	Attacks related to availability					
		Black hole	Sinkhole	Wormhole	Selective Forwarding	Sybil	Flooding
1	MODA (Zhang et al., 2018)	×	×	×	×	×	×
2	EHDA (Ullah et al., 2020)	×	×	×	×	✓	✓
3	ESRDA (Zhong et al., 2018)	✓	✓	✓	✓	✓	✓
4	SDAW (Boubiche et al., 2016)	×	×	×	×	✓	×
5	Sign-share (Alghamdi et al., 2017)	✓	✓	✓	✓	✓	×
6	ASSDA (Hua et al., 2018)	×	×	×	×	×	×
7	OSM-EFHE (Shobana et al., 2020)	×	×	×	×	✓	×
8	SAPDA (Goyal et al., 2020)	✓	✓	✓	✓	✓	✓
9	EPDA (Zhou et al., 2019)	×	×	×	×	×	×
10	RCDA (Chen et al., 2011)	×	×	×	×	✓	×
11	QPPDA (Liu et al., 2020)	×	×	×	×	×	×
12	LSDAR (Haseeb et al., 2020)	×	×	×	×	×	×
13	SDAPA (Parmar and Kadhiwala, 2016)	✓	✓	✓	✓	✓	✓
14	LIPDA (Zhao et al., 2016)	×	×	×	×	×	×
15	CSDA (Fang et al., 2019)	×	×	×	×	×	×

Table 32
Comparison between SDA schemes in terms of attacks related to data integrity.

Number	Scheme	Attacks related to data integrity			
		Node replication	Packet injection	Packet duplication	Packet alteration
1	MODA (Zhang et al., 2018)	×	×	×	×
2	EHDA (Ullah et al., 2020)	×	✓	✓	✓
3	ESRDA (Zhong et al., 2018)	×	✓	✓	✓
4	SDAW (Boubiche et al., 2016)	×	✓	×	✓
5	Sign-share (Alghamdi et al., 2017)	✓	✓	×	✓
6	ASSDA (Hua et al., 2018)	×	×	×	×
7	OSM-EFHE (Shobana et al., 2020)	×	✓	×	✓
8	SAPDA (Goyal et al., 2020)	✓	✓	✓	✓
9	EPDA (Zhou et al., 2019)	×	×	×	×
10	RCDA (Chen et al., 2011)	×	✓	✓	✓
11	QPPDA (Liu et al., 2020)	×	×	×	×
12	LSDAR (Haseeb et al., 2020)	×	×	×	×
13	SDAPA (Parmar and Kadhiwala, 2016)	✓	✓	✓	✓
14	LIPDA (Zhao et al., 2016)	×	✓	×	✓
15	CSDA (Fang et al., 2019)	×	×	×	×

Table 33
Classification of SDA schemes based on application.

Number	SDA scheme		Low-risk application	High-risk application
1	MODA (Zhang et al., 2018)		✓	×
2	EHDA (Ullah et al., 2020)		×	✓
3	ESRDA (Zhong et al., 2018)		×	✓
4	SDAW (Boubiche et al., 2016)		✓	×
5	Sign-share (Alghamdi et al., 2017)		×	✓
6	ASSDA (Hua et al., 2018)		✓	×
7	OSM-EFHE (Shobana et al., 2020)		×	✓
8	SAPDA (Goyal et al., 2020)		×	✓
9	EPDA (Zhou et al., 2019)		✓	×
10	RCDA (Chen et al., 2011)	RCDA-HOMO	×	✓
		RCDA-HETE	×	✓
11	QPPDA (Liu et al., 2020)		✓	×
12	LSDAR (Haseeb et al., 2020)		✓	×
13	SDAPA (Parmar and Kadhiwala, 2016)		×	✓
14	LIPDA (Zhao et al., 2016)		×	✓
15	CSDA (Fang et al., 2019)		✓	×

Table 34
Classification of the SDA methods in terms of the authentication mechanism.

Number	Scheme		Hop-by-hop authentication mechanism	End-to-end authentication mechanism
1	MODA (Zhang et al., 2018)		×	×
2	EHDA (Ullah et al., 2020)		✓	×
3	ESRDA (Zhong et al., 2018)		✓	×
4	SDAW (Boubiche et al., 2016)		✓	×
5	Sign-share (Alghamdi et al., 2017)		×	✓
6	ASSDA (Hua et al., 2018)		×	×
7	OSM-EFHE (Shobana et al., 2020)		✓	×
8	SAPDA (Goyal et al., 2020)		×	✓
9	EPDA (Zhou et al., 2019)		×	×
10	RCDA (Chen et al., 2011)	RCDA-HOMO	×	✓
		RCDA-HETE	✓	✓
11	QPPDA (Liu et al., 2020)		×	×
12	LSDAR (Haseeb et al., 2020)		×	×
13	SDAPA (Parmar and Kadhiwala, 2016)		✓	×
14	LIPDA (Zhao et al., 2016)		×	✓
15	CSDA (Fang et al., 2019)		×	×

Table 35
Classification of the SDA methods in terms of the data recovery ability.

Number	Scheme		Recoverable SDA schemes	Unrecoverable SDA schemes
1	MODA (Zhang et al., 2018)		✓	✓
2	EHDA (Ullah et al., 2020)		✓	×
3	ESRDA (Zhong et al., 2018)		✓	×
4	SDAW (Boubiche et al., 2016)		×	✓
5	Sign-share (Alghamdi et al., 2017)		✓	×
6	ASSDA (Hua et al., 2018)		×	✓
7	OSM-EFHE (Shobana et al., 2020)		✓	×
8	SAPDA (Goyal et al., 2020)		✓	×
9	EPDA (Zhou et al., 2019)		×	✓
10	RCDA (Chen et al., 2011)	RCDA-HOMO	✓	×
		RCDA-HETE	×	✓
11	QPPDA (Liu et al., 2020)		×	✓
12	LSDAR (Haseeb et al., 2020)		✓	×
13	SDAPA (Parmar and Kadhiwala, 2016)		✓	×
14	LIPDA (Zhao et al., 2016)		×	✓
15	CSDA (Fang et al., 2019)		×	✓

Dynamic environment. In most SDA schemes, it is assumed that sensor nodes and the base station are immobile in the network. However, this assumption is not correct in many real environments. The movement of sensor nodes in the network may change the network topology. Therefore, the SDA protocol must be able to operate in dynamic environments appropriately.

Delay. Secure data aggregation can increase delay in the network. Whereas, in real-time applications, it is very important to consider the delay issue. Because in these applications, the data transmission process must be done with the lowest delay. Therefore, designing the time-sensitive SDA schemes is an open issue that should be studied by researchers.

Testbeds. Many secure data aggregation methods are tested using various simulation tools such as NS2, NS3, OMNET++ and so on. However, these tools cannot properly evaluate the various aspects of a SDA scheme. For this reason, it is necessary to implement these methods in real environments. However, it is very costly.

Using the new techniques. Today, machine learning (ML) and artificial intelligence (AI) techniques are used to design different protocols. They provide the promising results. However, these techniques are rarely applied in designing SDA methods. It is an open issue that should be discussed in the future by researchers.

7. Conclusion

In this survey, we studied and evaluated several SDA schemes comprehensively. We first introduced the most important security requirements in WSNs and described the most common attacks in these networks. Next, we categorized secure data aggregation methods based on the network model, topology, key cryptography technique, encryption method, application, authentication mechanism, and data recovery ability. Then, we introduced some SDA methods in WSNs. We analyzed these methods in terms of different security requirements and evaluated their countermeasures against different attacks. We also expressed their strengths and weaknesses. These SDA schemes consider different security requirements and are suitable for a specific application. In this paper, we attempted to give researchers an appropriate perspective on SDA schemes and future research directions. This paper also helps researchers gain a correct understanding for designing appropriate SDA methods, and addressing the current issues. In future research, we will examine secure data aggregation schemes that use artificial intelligence (AI) and machine learning (ML) techniques, and analyze their effect on the performance of WSNs. Today, these new techniques have been used in many fields and have shown promising results. We believe that these new techniques can be used to design SDA protocols for WSNs and increase their efficiency and performance.

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.

References

Abdollahzadeh, S., Navimipour, N.J., 2016. Deployment strategies in the wireless sensor network: A comprehensive review. Comput. Commun. 91, 1–16. http://dx.doi.org/10.1016/j.comcom.2016.06.003.

Ahutu, O.R., El-Ocla, H., 2020. Centralized routing protocol for detecting wormhole attacks in wireless sensor networks. IEEE Access 8, 63270–63282. http://dx.doi.org/10.1109/ACCESS.2020.2983438.

Akkaya, K., Ari, I., 2007. In-network data aggregation in wireless sensor networks. In: Handbook of Computer Networks: LANs, MANs, WANs, the Internet, and Global, Cellular, and Wireless Networks, vol. 2. Wiley Online Library, pp. 1131–1146.

Alghamdi, W.Y., Wu, H., Kanhere, S.S., 2017. Reliable and secure end-to-end data aggregation using secret sharing in wsns. In: 2017 IEEE Wireless Communications and Networking Conference (WCNC). IEEE, pp. 1–6. http://dx.doi.org/10.1109/WCNC.2017.7925558.

Angappan, A., Saravanabava, T., Sakthivel, P., Vishvaksenan, K., 2020. Novel sybil attack detection using RSSI and neighbour information to ensure secure communication in WSN. J. Ambient Intell. Humaniz. Comput. 1–12. http://dx.doi.org/10.1007/s12652-020-02276-5.

Anitha, S., Jayanthi, P., Chandrasekaran, V., 2020. An intelligent based healthcare security monitoring schemes for detection of node replication attack in wireless sensor networks. Measurement 167, 108272. http://dx.doi.org/10.1016/j.measurement.2020.108272.

Aslan, Y.E., Korpeoglu, I., Ulusoy, Ö., 2012. A framework for use of wireless sensor networks in forest fire detection and monitoring. Comput. Environ. Urban Syst. 36 (6), 614–625. http://dx.doi.org/10.1016/j.compenvurbsys.2012.03.002.

Atwady, Y., Hammoudeh, M., 2017. A survey on authentication techniques for the internet of things. In: Proceedings of the International Conference on Future Networks and Distributed Systems. http://dx.doi.org/10.1145/3102304.3102312.

Barati, H., Movaghar, A., Rahmani, A.M., 2015. EACHP: Energy aware clustering hierarchy protocol for large scale wireless sensor networks. Wirel. Pers. Commun. 85 (3), 765–789. http://dx.doi.org/10.1007/s11277-015-2807-2.

Baroutis, N., Younis, M., 2016. A novel traffic analysis attack model and base-station anonymity metrics for wireless sensor networks. Secur. Commun. Netw. 9 (18), 5892–5907. http://dx.doi.org/10.1002/sec.1744.

Bekmezci, I., 2009. Wireless Sensor Networks: A Military Monitoring Application. VDM Verlag, http://dx.doi.org/10.5555/1643673.

Belkhira, S.A.H., Boukli Hacene, S., Lorenz, P., Belkheir, M., Gilg, M., Bouziani, M., 2019. WRE-OLSR, a new scheme for enhancing the lifetime within ad hoc and wireless sensor networks. Int. J. Commun. Syst. 32 (11), e3975. http://dx.doi.org/10.1002/dac.3975.

Bhushan, B., Sahoo, G., 2020. Requirements, protocols, and security challenges in wireless sensor networks: An industrial perspective. In: Handbook of Computer Networks and Cyber Security. Springer, pp. 683–713. http://dx.doi.org/10.1007/978-3-030-22277-2_27.

Bodkhe, U., Tanwar, S., 2020. Secure data dissemination techniques for IoT applications: Research challenges and opportunities. Softw. - Pract. Exp. http://dx.doi.org/10.1002/spe.2811.

Boubiche, D.E., Athmani, S., Boubiche, S., Toral-Cruz, H., 2020. Cybersecurity issues in wireless sensor networks: current challenges and solutions. Wirel. Pers. Commun. 1–37. http://dx.doi.org/10.1007/s11277-020-07213-5.

Boubiche, S., Boubiche, D.E., Bilami, A., Toral-Cruz, H., 2018. Big data challenges and data aggregation strategies in wireless sensor networks. IEEE Access 6, 20558–20571. http://dx.doi.org/10.1109/ACCESS.2018.2821445.

Boubiche, D.E., Boubiche, S., Toral-Cruz, H., Pathan, A.-S.K., Bilami, A., Athmani, S., 2016. SDAW: secure data aggregation watermarking-based scheme in homogeneous WSNs. Telecommun. Syst. 62 (2), 277–288. http://dx.doi.org/10.1007/s11235-015-0047-0.

Chen, C.-M., Lin, Y.-H., Lin, Y.-C., Sun, H.-M., 2011. RCDA: Recoverable concealed data aggregation for data integrity in wireless sensor networks. IEEE Trans. Parallel Distrib. Syst. 23 (4), 727–734. http://dx.doi.org/10.1109/TPDS.2011.219.

Chen, H., Lou, W., 2015. On protecting end-to-end location privacy against local eavesdropper in wireless sensor networks. Pervasive Mob. Comput. 16, 36–50. http://dx.doi.org/10.1016/j.pmcj.2014.01.006.

Choubey, R.K., Hashmi, A., 2018. Cryptographic techniques in information security.

Conti, M., 2015. Secure Wireless Sensor Networks. Springer, http://dx.doi.org/10.1007/978-1-4939-3460-7.

Cui, J., Boussetta, K., Valois, F., 2020. Classification of data aggregation functions in wireless sensor networks. Comput. Netw. 107342. http://dx.doi.org/10.1016/j.comnet.2020.107342.

Dargie, W., Poellabauer, C., 2010. Fundamentals of Wireless Sensor Networks: Theory and Practice. John Wiley & Sons.

Dehkordi, S.A., Farajzadeh, K., Rezazadeh, J., Farahbakhsh, R., Sandrasegaran, K., Dehkordi, M.A., 2020. A survey on data aggregation techniques in IoT sensor networks. Wirel. Netw. 26 (2), 1243–1263. http://dx.doi.org/10.1007/s11276-019-02142-z.

Dewal, P., Narula, G.S., Jain, V., Baliyan, A., 2018. Security attacks in wireless sensor networks: A survey. In: Cyber Security. Springer, pp. 47–58. http://dx.doi.org/10.1007/978-981-10-8536-9_6.

Dhanvijay, M.M., Patil, S.C., 2019. Internet of things: A survey of enabling technologies in healthcare and its applications. Comput. Netw. 153, 113–131. http://dx.doi.org/10.1016/j.comnet.2019.03.006.

Di Pietro, R., Guarino, S., Verde, N.V., Domingo-Ferrer, J., 2014. Security in wireless ad-hoc networks–a survey. Comput. Commun. 51, 1–20. http://dx.doi.org/10.1016/j.comcom.2014.06.003.

Dong, D., Li, M., Liu, Y., Li, X.-Y., Liao, X., 2011. Topological detection on wormholes in wireless ad hoc and sensor networks. IEEE/ACM Trans. Netw. 19 (6), 1787–1796. http://dx.doi.org/10.1109/TNET.2011.2163730.

Dutta, N., Singh, M.M., 2019. Wormhole attack in wireless sensor networks: A critical review. In: Advanced Computing and Communication Technologies. Springer, pp. 147–161. http://dx.doi.org/10.1007/978-981-13-0680-8_14.

Fahmy, H.M.A., 2020. Wireless Sensor Networks. Springer, http://dx.doi.org/10.1007/978-981-10-0412-4.

Fang, W., Wen, X., Xu, J., Zhu, J., 2019. CSDA: a novel cluster-based secure data aggregation scheme for WSNs. Cluster Comput. 22 (3), 5233–5244. http://dx.doi.org/10.1007/s10586-017-1195-7.

Fei, Z., Li, B., Yang, S., Xing, C., Chen, H., Hanzo, L., 2016. A survey of multi-objective optimization in wireless sensor networks: Metrics, algorithms, and open problems. IEEE Commun. Surv. Tutor. 19 (1), 550–586. http://dx.doi.org/10.1109/COMST.2016.2610578.

Fu, H., Liu, Y., Dong, Z., Wu, Y., 2020. A data clustering algorithm for detecting selective forwarding attack in cluster-based wireless sensor networks. Sensors 20 (1), 23. http://dx.doi.org/10.3390/s20010023.

Ghani, A., Mansoor, K., Mehmood, S., Chaudhry, S.A., Rahman, A.U., Najmus Saqib, M., 2019. Security and key management in IoT-based wireless sensor networks: An authentication protocol using symmetric key. Int. J. Commun. Syst. 32 (16), e4139. http://dx.doi.org/10.1002/dac.4139.

Gharib, M., Moradlou, Z., Doostari, M.A., Movaghar, A., 2017. Fully distributed ECC-based key management for mobile ad hoc networks. Comput. Netw. 113, 269–283. http://dx.doi.org/10.1016/j.comnet.2016.12.017.

Goyal, N., Dave, M., Verma, A.K., 2019. Data aggregation in underwater wireless sensor network: Recent approaches and issues. J. King Saud Univ.-Comput. Inf. Sci. 31 (3), 275–286. http://dx.doi.org/10.1016/j.jksuci.2017.04.007.

Goyal, N., Dave, M., Verma, A.K., 2020. SAPDA: Secure authentication with protected data aggregation scheme for improving QoS in scalable and survivable UWSNs. Wirel. Pers. Commun. 1–15. http://dx.doi.org/10.1007/s11277-020-07175-8.

Halak, B., 2018. A primer on cryptographic primitives and security attacks. In: Physically Unclonable Functions. Springer, pp. 1–15. http://dx.doi.org/10.1007/978-3-319-76804-5_1.

Hamedheidari, S., Rafeh, R., 2013. A novel agent-based approach to detect sinkhole attacks in wireless sensor networks. Comput. Secur. 37, 1–14. http://dx.doi.org/10.1016/j.cose.2013.04.002.

Haseeb, K., Islam, N., Saba, T., Rehman, A., Mehmood, Z., 2020. LSDAR: A light-weight structure based data aggregation routing protocol with secure internet of things integrated next-generation sensor networks. Sustainable Cities Soc. 54, 101995. http://dx.doi.org/10.1016/j.scs.2019.101995.

Hassija, V., Chamola, V., Saxena, V., Jain, D., Goyal, P., Sikdar, B., 2019. A survey on IoT security: application areas, security threats, and solution architectures. IEEE Access 7, 82721–82743. http://dx.doi.org/10.1109/ACCESS.2019.2924045.

Hatamian, M., Barati, H., Movaghar, A., Naghizadeh, A., 2016. CGC: centralized genetic-based clustering protocol for wireless sensor networks using onion approach. Telecommun. Syst. 62 (4), 657–674. http://dx.doi.org/10.1007/s11235-015-0102-x.

Hua, P., Liu, X., Yu, J., Dang, N., Zhang, X., 2018. Energy-efficient adaptive slice-based secure data aggregation scheme in WSN. Procedia Comput. Sci. 129, 188–193. http://dx.doi.org/10.1016/j.procs.2018.03.033.

Illiano, V.P., Lupu, E.C., 2015. Detecting malicious data injections in wireless sensor networks: A survey. ACM Comput. Surv. 48 (2), 1–33. http://dx.doi.org/10.1145/2818184.

Kandris, D., Nakas, C., Vomvas, D., Koulouras, G., 2020. Applications of wireless sensor networks: an up-to-date survey. Appl. Syst. Innov. 3 (1), 14. http://dx.doi.org/10.3390/asi3010014.

Karmaker, A., Alam, M.S., Hasan, M.M., Craig, A., 2020. An energy-efficient and balanced clustering approach for improving throughput of wireless sensor networks. Int. J. Commun. Syst. 33 (3), e4195. http://dx.doi.org/10.1002/dac.4195.

Kaur, M., Munjal, A., 2020. Data aggregation algorithms for wireless sensor network: A review. Ad Hoc Netw. 100, 102083. http://dx.doi.org/10.1016/j.adhoc.2020.102083.

Kaushik, I., Sharma, N., 2020. Black hole attack and its security measure in wireless sensors networks. In: Handbook of Wireless Sensor Networks: Issues and Challenges in Current Scenario's. Springer, pp. 401–416. http://dx.doi.org/10.1007/978-3-030-40305-8_20.

Khan, S., Pathan, A.-S.K., Alrajeh, N.A., 2016. Wireless Sensor Networks: Current Status and Future Trends. CRC press.

Khan, W.Z., Rehman, M., Zangoti, H.M., Afzal, M.K., Armi, N., Salah, K., 2020. Industrial internet of things: Recent advances, enabling technologies and open challenges. Comput. Electr. Eng. 81, 106522. http://dx.doi.org/10.1016/j.compeleceng.2019.106522.

Kouicem, D.E., Bouabdallah, A., Lakhlef, H., 2018. Internet of things security: A top-down survey. Comput. Netw. 141, 199–221. http://dx.doi.org/10.1016/j.comnet.2018.03.012.

Lakshmi, V., Deepthi, P., 2019. A secure channel code-based scheme for privacy preserving data aggregation in wireless sensor networks. Int. J. Commun. Syst. 32 (1), e3832. http://dx.doi.org/10.1002/dac.3832.

Lindell, Y., 2017. Tutorials on the Foundations of Cryptography: Dedicated to Oded Goldreich. Springer, http://dx.doi.org/10.1007/978-3-319-57048-8.

Liu, A., Dong, M., Ota, K., Long, J., 2015. PHACK: An efficient scheme for selective forwarding attack detection in WSNs. Sensors 15 (12), 30942–30963. http://dx.doi.org/10.3390/s151229835.

Liu, D., Ning, P., 2007. Security for Wireless Sensor Networks, vol. 28. Springer Science & Business Media, http://dx.doi.org/10.1007/978-0-387-46781-8.

Liu, X., Yu, J., Li, F., Lv, W., Wang, Y., Cheng, X., 2019. Data aggregation in wireless sensor networks: from the perspective of security. IEEE Internet Things J. http://dx.doi.org/10.1109/JIOT.2019.2957396.

Liu, X., Zhang, X., Yu, J., Fu, C., 2020. Query privacy preserving for data aggregation in wireless sensor networks. Wirel. Commun. Mob. Comput. 2020, http://dx.doi.org/10.1155/2020/9754973.

Liyanage, M., Braeken, A., Kumar, P., Ylianttila, M., 2020a. IoT Security: Advances in Authentication. John Wiley & Sons.

Liyanage, M., Braeken, A., Kumar, P., Ylianttila, M., 2020b. IoT Security: Advances in Authentication. John Wiley & Sons, http://dx.doi.org/10.1002/9781119527978.

Mehetre, D.C., Roslin, S.E., Wagh, S.J., 2019. Detection and prevention of black hole and selective forwarding attack in clustered WSN with active trust. Cluster Comput. 22 (1), 1313–1328. http://dx.doi.org/10.1007/s10586-017-1622-9.

Mehrjoo, S., Khunjush, F., 2018. Optimal data aggregation tree in wireless sensor networks based on improved river formation dynamics. Comput. Intell. 34 (3), 802–820. http://dx.doi.org/10.1111/coin.12132.

Merad Boudia, O.R., Senouci, S.M., Feham, M., 2018. Secure and efficient verification for data aggregation in wireless sensor networks. Int. J. Netw. Manag. 28 (1), e2000. http://dx.doi.org/10.1002/nem.2000.

Messai, M.-L., Seba, H., 2016. A survey of key management schemes in multi-phase wireless sensor networks. Comput. Netw. 105, 60–74. http://dx.doi.org/10.1016/j.comnet.2016.05.005.

Mishra, A.K., Turuk, A.K., 2016. A comparative analysis of node replica detection schemes in wireless sensor networks. J. Netw. Comput. Appl. 61, 21–32. http://dx.doi.org/10.1016/j.jnca.2015.12.001.

Muduli, L., Mishra, D.P., Jana, P.K., 2018. Application of wireless sensor network for environmental monitoring in underground coal mines: A systematic review. J. Netw. Comput. Appl. 106, 48–67. http://dx.doi.org/10.1016/j.jnca.2017.12.022.

Mustafa, G., Ashraf, R., Mirza, M.A., Jamil, A., 2018. A review of data security and cryptographic techniques in IoT based devices. In: Proceedings of the 2nd International Conference on Future Networks and Distributed Systems. pp. 1–9. http://dx.doi.org/10.1145/3231053.3231100.

Nagireddy, V., Parwekar, P., 2019. Attacks in wireless sensor networks. In: Smart Intelligent Computing and Applications. Springer, pp. 439–447. http://dx.doi.org/10.1007/978-981-13-1927-3_47.

Ni, J., Zhou, L., Ravishankar, C.V., 2010. Dealing with random and selective attacks in wireless sensor systems. ACM Trans. Sensor Netw. 6 (2), 1–40. http://dx.doi.org/10.1145/1689239.1689245.

Numan, M., Subhan, F., Khan, W.Z., Hakak, S., Haider, S., Reddy, G.T., Jolfaei, A., Alazab, M., 2020. A systematic review on clone node detection in static wireless sensor networks. IEEE Access 8, 65450–65461. http://dx.doi.org/10.1109/ACCESS.2020.2983091.

Oreku, G.S., Pazynyuk, T., 2016. Security in Wireless Sensor Networks. Springer, http://dx.doi.org/10.1007/978-3-319-21269-2.

Ozdemir, S., Çam, H., 2009. Integration of false data detection with data aggregation and confidential transmission in wireless sensor networks. IEEE/ACM Trans. Netw. 18 (3), 736–749. http://dx.doi.org/10.1109/TNET.2009.2032910.

Ozdemir, S., Xiao, Y., 2009. Secure data aggregation in wireless sensor networks: A comprehensive overview. Comput. Netw. 53 (12), 2022–2037. http://dx.doi.org/10.1016/j.comnet.2009.02.023.

Parmar, K., Jinwala, D.C., 2016. Concealed data aggregation in wireless sensor networks: A comprehensive survey. Comput. Netw. 103, 207–227. http://dx.doi.org/10.1016/j.comnet.2016.04.013.

Parmar, P., Kadhiwala, B., 2016. Secure data aggregation protocol using AES in wireless sensor network. In: Emerging Research in Computing, Information, Communication and Applications. Springer, pp. 421–432. http://dx.doi.org/10.1007/978-981-10-0287-8_39.

Penttinen, J.T., 2016. Wireless Communications Security: Solutions for the Internet of Things. Wiley Online Library.

Perrig, A., Stankovic, J., Wagner, D., 2004. Security in wireless sensor networks. Commun. ACM 47 (6), 53–57. http://dx.doi.org/10.1145/990680.990707.

Pongaliur, K., Xiao, L., 2013. Sensor node source privacy and packet recovery under eavesdropping and node compromise attacks. ACM Trans. Sensor Netw. 9 (4), 1–26. http://dx.doi.org/10.1145/2489253.2489267.

Pourghebleh, B., Navimipour, N.J., 2017. Data aggregation mechanisms in the internet of things: A systematic review of the literature and recommendations for future research. J. Netw. Comput. Appl. 97, 23–34. http://dx.doi.org/10.1016/j.jnca.2017.08.006.

Randhawa, S., Jain, S., 2017. Data aggregation in wireless sensor networks: Previous research, current status and future directions. Wirel. Pers. Commun. 97 (3), 3355–3425. http://dx.doi.org/10.1007/s11277-017-4674-5.

Rani, S., Maheswar, R., Kanagachidambaresan, G., Jayarajan, P., 2020. Integration of WSN and IoT for Smart Cities. Springer, http://dx.doi.org/10.1007/978-3-030-38516-3.

Raymond, D.R., Midkiff, S.F., 2008. Denial-of-service in wireless sensor networks: Attacks and defenses. IEEE Pervasive Comput. 7 (1), 74–81. http://dx.doi.org/10.1109/MPRV.2008.6.

Rehman, A.-u., Rehman, S.U., Raheem, H., 2019. Sinkhole attacks in wireless sensor networks: A survey. Wirel. Pers. Commun. 106 (4), 2291–2313. http://dx.doi.org/10.1007/s11277-018-6040-7.

Sah, D.K., Amgoth, T., 2020. Renewable energy harvesting schemes in wireless sensor networks: a survey. Inf. Fusion 63, 223–247. http://dx.doi.org/10.1016/j.inffus.2020.07.005.

Salvadori, F., de Campos, M., Sausen, P.S., de Camargo, R.F., Gehrke, C., Rech, C., Spohn, M.A., Oliveira, A.C., 2009. Monitoring in industrial systems using wireless sensor network with dynamic power management. IEEE Trans. Instrum. Meas. 58 (9), 3104–3111. http://dx.doi.org/10.1109/TIM.2009.2016882.

Shah, P.K., Shukla, K.V., 2012. Secure data aggregation issues in wireless sensor network: A survey. J. Inf. Commun. Technol. 2 (1).

Shobana, M., Sabitha, R., Karthik, S., 2020. An enhanced soft computing-based formulation for secure data aggregation and efficient data processing in large-scale wireless sensor network. Soft Comput. 1–12. http://dx.doi.org/10.1007/s00500-020-04694-1.

Stavroulakis, P., Stamp, M., 2010. Handbook of Information and Communication Security. Springer Science & Business Media.

Tamilarasi, N., Santhi, S., 2020. Detection of wormhole attack and secure path selection in wireless sensor network. Wirel. Pers. Commun. 114 (1), 329–345. http://dx.doi.org/10.1007/s11277-020-07365-4.

Ullah, A., Said, G., Sher, M., Ning, H., 2020. Fog-assisted secure healthcare data aggregation scheme in IoT-enabled WSN. Peer-To-Peer Netw. Appl. 13 (1), 163–174. http://dx.doi.org/10.1007/s12083-019-00745-z.

Vasudeva, A., Sood, M., 2018. Survey on sybil attack defense mechanisms in wireless ad hoc networks. J. Netw. Comput. Appl. 120, 78–118. http://dx.doi.org/10.1016/j.jnca.2018.07.006.

Vinodha, D., Anita, E.M., 2019. Secure data aggregation techniques for wireless sensor networks: a review. Arch. Comput. Methods Eng. 26 (4), 1007–1027. http://dx.doi.org/10.1007/s11831-018-9267-2.

Ward, J.R., Younis, M., 2019. Cross-layer traffic analysis countermeasures against adaptive attackers of wireless sensor networks. Wirel. Netw. 25 (5), 2869–2887. http://dx.doi.org/10.1007/s11276-019-02003-9.

Wazid, M., Das, A.K., 2017. A secure group-based blackhole node detection scheme for hierarchical wireless sensor networks. Wirel. Pers. Commun. 94 (3), 1165–1191. http://dx.doi.org/10.1007/s11277-016-3676-z.

Wazid, M., Das, A.K., Kumari, S., Khan, M.K., 2016. Design of sinkhole node detection mechanism for hierarchical wireless sensor networks. Secur. Commun. Netw. 9 (17), 4596–4614. http://dx.doi.org/10.1002/sec.1652.

Xiang, L., Luo, J., Rosenberg, C., 2012. Compressed data aggregation: Energy-efficient and high-fidelity data collection. IEEE/ACM Trans. Netw. 21 (6), 1722–1735. http://dx.doi.org/10.1109/TNET.2012.2229716.

Xu, X., Han, M., Nagarajan, S.M., Anandhan, P., 2020. Industrial internet of things for smart manufacturing applications using hierarchical trustful resource assignment. Comput. Commun. 160, 423–430. http://dx.doi.org/10.1016/j.comcom.2020.06.004.

Yaseen, Q., Albalas, F., Jararwah, Y., Al-Ayyoub, M., 2018. Leveraging fog computing and software defined systems for selective forwarding attacks detection in mobile wireless sensor networks. Trans. Emerg. Telecommun. Technol. 29 (4), e3183. http://dx.doi.org/10.1002/ett.3183.

Yetgin, H., Cheung, K.T.K., El-Hajjar, M., Hanzo, L.H., 2017. A survey of network lifetime maximization techniques in wireless sensor networks. IEEE Commun. Surv. Tutor. 19 (2), 828–854. http://dx.doi.org/10.1109/COMST.2017.2650979.

Yousefpoor, M.S., Barati, H., 2019. Dynamic key management algorithms in wireless sensor networks: A survey. Comput. Commun. 134, 52–69. http://dx.doi.org/10.1016/j.comcom.2018.11.005.

Yousefpoor, M.S., Barati, H., 2020. DSKMS: a dynamic smart key management system based on fuzzy logic in wireless sensor networks. Wirel. Netw. 26 (4), 2515–2535. http://dx.doi.org/10.1007/s11276-019-01980-1.

Yugha, R., Chithra, S., 2020. A survey on technologies and security protocols: Reference for future generation IoT. J. Netw. Comput. Appl. 102763. http://dx.doi.org/10.1016/j.jnca.2020.102763.

Zhang, P., Wang, J., Guo, K., Wu, F., Min, G., 2018. Multi-functional secure data aggregation schemes for WSNs. Ad Hoc Netw. 69, 86–99. http://dx.doi.org/10.1016/j.adhoc.2017.11.004.

Zhao, X., Zhu, J., Liang, X., Jiang, S., Chen, Q., 2016. Lightweight and integrity-protecting oriented data aggregation scheme for wireless sensor networks. IET Inf. Secur. 11 (2), 82–88. http://dx.doi.org/10.1049/iet-ifs.2015.0387.

Zhong, H., Shao, L., Cui, J., Xu, Y., 2018. An efficient and secure recoverable data aggregation scheme for heterogeneous wireless sensor networks. J. Parallel Distrib. Comput. 111, 1–12. http://dx.doi.org/10.1016/j.jpdc.2017.06.019.

Zhou, L., Ge, C., Hu, S., Su, C., 2019. Energy-efficient and privacy-preserving data aggregation algorithm for wireless sensor networks. IEEE Internet Things J. 7 (5), 3948–3957. http://dx.doi.org/10.1109/JIOT.2019.2959094.

Zhu, S., Setia, S., Jajodia, S., Ning, P., 2007. Interleaved hop-by-hop authentication against false data injection attacks in sensor networks. ACM Trans. Sensor Netw. 3 (3), 14–es. http://dx.doi.org/10.1145/1267060.1267062.

Zhu, L., Zhang, Z., Xu, C., 2017. Secure data aggregation in wireless sensor networks. In: Secure and Privacy-Preserving Data Communication in Internet of Things. Springer, pp. 3–31. http://dx.doi.org/10.1007/978-981-10-3235-6_2.

Zhu, W.T., Zhou, J., Deng, R.H., Bao, F., 2012. Detecting node replication attacks in wireless sensor networks: a survey. J. Netw. Comput. Appl. 35 (3), 1022–1034. http://dx.doi.org/10.1016/j.jnca.2012.01.002.

Mohammad Sadegh Yousefpoor received B.Sc. degree in computer science from Dezful Branch, Payame Noor University, Dezful, Iran, in 2010. He received the M.Sc. degree in Artificial Intelligence (AI) Computer Engineering at Department of Computer Engineering, Dezful Branch, Islamic Azad University, Dezful, Iran, in 2017. His research interests include Wireless sensor networks (WSNs), Internet of things (IoT), Machine learning, Pattern recognition, Cryptography and Network security.

Efat Yousefpoor received B.Sc. degree in Computer Hardware Engineering from Jundishapoor University, Dezful, Iran, in 2013. She is currently working toward the M.Sc. degree in Artificial Intelligence Computer Engineering at Department of Computer Engineering, Dezful Branch, Islamic Azad University, Dezful, Iran. Her research interests include Wireless sensor networks (WSNs), Cryptography and Network security.

Hamid Barati is an Assistant Professor in the Department of Computer Engineering at Dezful Branch, Islamic Azad University, Dezful, Iran. He received his B.S. degree in Computer Hardware Engineering, M.S. degree in Computer Systems Architecture Engineering and Ph.D. degree in Computer Systems Architecture Engineering in 2005, 2007 and 2015 respectively. Currently he is Faculty of Islamic Azad University, Dezful Branch, Iran. His major research experiences and interests include mobile adhoc networks, interconnection networks and energy-efficient routing and security issues in wireless sensor networks.

Ali Barati is an Assistant Professor in the Department of Computer Engineering at Dezful Branch, Islamic Azad University, Dezful, Iran. He received his B.Sc. degree in Computer Hardware Engineering and M.Sc. degree in Computer Software Engineering in 2001 and 2004 respectively, and he received a Ph.D. degree in Computer Software Engineering at Department of Computer Engineering and Information Technology, Qazvin Branch, Islamic Azad University, Qazvin, Iran in 2014. Currently he is faculty of Department of Computer Engineering, Dezful Branch, Islamic Azad University, Dezful, Iran. Between 2007 and 2011 he was appointed as the Head of Department of Computer Science at the University. His research interests include wireless sensor networks, VANET, high speed networks and fault-tolerant systems.

Ali Movaghar (Senior Member, IEEE) received the B.S. degree in electrical engineering from the University of Tehran in 1977, and the M.S. and Ph.D. degrees in computer, information, and control engineering from the University of Michigan in 1979 and 1985. He is currently a Professor with the Department of Computer Engineering, Sharif University of Technology. His research interests include performance/dependability modeling and formal verification of wireless networks and distributed real-time systems.

Mehdi Hosseinzadeh received the B.E. degree in computer hardware engineering from the Dezful Branch, Islamic Azad University (IAU), Tehran, Iran, in 2003, and the M.Sc. and Ph.D. degrees in computer system architecture from the Science and Research Branch, IAU, in 2005 and 2008, respectively. He is currently an Associate Professor with the Iran University of Medical Sciences, Tehran. His research interests include SDN, information technology, data mining, big data analytics, E-commerce, E-marketing, and social networks.