Available: DOI: https://doi.org/10.54117/gjpas.v3i1.118 *Research article*

Gadau Journal of Pure and Allied Sciences
Gadau J Pure Alli Sci, 3(1): 15-29 (2024)
ISSN: 2955-1722
DOI: https://doi.org/10.54117/gjpas.v3i1.118

Analysis of Key Management Schemes for Centralized and Distributed Key Revocation in Wireless Sensor Networks

Taofeek Yusuf[*1], Victor Onomza Waziri[1], Morufu Olalere[1], Muhammad Bashir Abdullahi[2]

[1]Department of Cyber Security Science, Federal University Technology, Minna, Nigeria.
[2]Department of Computer Science, Federal University Technology, Minna, Nigeria.

**Correspondence:* yusuf.pg824211@st.futminna.edu.ng; *+2348035926745*

Abstract

Wireless sensor network (WSN) applications have been utilized for a variety of applications, such as monitoring environments, military functions, and patient status monitoring. In most of their applications, the security of these network is critical and needed robust support because of the sensitive nature of the information being transmitted over the unsecured wireless channel. Securing communications in wireless sensor networks is a challenging task. This is because they are mostly deployed in a hostile environment and coupled with their constrained resources. Data security in WSNs requires appropriate cryptographic techniques, usually implemented through secure and efficient key management schemes. Research on key management in WSN is a work in progress, majority of existing research work focus on key establishment and distribution aspect of the key management system. While less attention is devoted to key revocation and renewal techniques. For this reason, network and protocol designers encounter challenges in the selection and implementation of the most appropriate schemes for WSN applications. This research work presents an exhaustive review of centralized and distributed categories of key revocation schemes in existence. The study analyzed key revocation developments based on literature review and comparison of existing work, to explore how the categories of key revocation approaches help to improve security and efficiency in WSN. The results of comparative analysis of the security and performance requirements of the selected candidates of centralized and distributed schemes shows that KRRP scheme is the only solution among the selected candidates analyzed that can guarantee end-to-end data secrecy, because it merges the capability of the private and public key cryptography in their implementation.

Article History

Received: 20/11/2023
Accepted: 26/03/2024
Published:01/06/2024

Keywords
Wireless Sensor Networks; Key Management systems; Key Revocation Schemes; Algorithm; Intrusion Detection Systems; Key Establishment; Protocols.

How to cite this paper: Yusuf, T., Waziri, V. O., Olalere, M., Abdullahi, M. B. (2024). Analysis of key management schemes for centralized and distributed key revocation in wireless sensor networks. *Gadau J Pure Alli Sci, 2(2): 15-29*. https://doi.org/10.54117/gjpas.v3i1.118.

1.0 Introduction

Wireless Sensor Networks (WSN) has an extensive array of prospects and can be employed to virtually all part of human endeavors (Muruganandam *et al*., 2023). The network consists of tiny sensor nodes with low-cost and wireless transmission. They work together to sense and collect data about certain environmental phenomena (Mehmood *et al*., 2021). Wireless sensor networks offer rewarding solutions to varieties of real-life issues, thus, making them helpful in several application domains. There are various application areas that WSNs are applicable due to their flexibility of deployment in recent times (Huanan *et al*., 2021). WSNs are utilized in application domains like environmental monitoring, transportation systems, healthcare schemes for observing patients' status (Chinniah and Krishnamoorthi, 2019). Sensor nodes are vulnerable to different types of attacks because of the sensitive information they transmit and their constrained resources.

Seemingly, the wireless medium for transmission also leaves a wide range of security threats and assaults open. This makes it easier for anyone to participate or monitor communications (Xue *et al*., 2023). These limitations in WSNs call for the design of lightweight secure and efficient communication mechanisms using cryptographic schemes (Ahlawat and Dave, 2021). However, three instances could warrant node/key revocation. This includes when a network node run out of power as its battery's energy gets drained. Secondly, a node could be captured by an adversary and the secret credentials extracted and thirdly an adversary can join the network and disguise itself as a legitimate node (Mansour *et al*., 2014). When this occurs and the intrusion detection system report identifies a node with any of these compromise tendencies, the next step is to remove the compromised node and then substitute the secret keys used in the affected network (Mansour *et al*., 2015). One important technique that guarantees secure and efficient removal of compromised nodes is an aspect of the key management system. The key management system consists of the following process, key setup, key generation, key agreement, key distribution, and key revocation (Mall *et al*., 2013). As an aspect of the key management system, key revocation is one of the most important approaches to secure communication in the event of node or key compromise (Moara-Nkwe *et al*., 2018). However, the requirements for designing secure and efficient key revocation schemes are fundamental issues in the life of key management systems.

The rest of the paper is organized as follows. Section 2 presents related work and literature survey, while section 3 described the methodology used in the work. Section 4 presents the categories of key revocation schemes. Section 5 covers the presentation of the representatives of centralized and distributed key revocation schemes selected for the study. In section 6, the security and performance analysis of centralized schemes. Section 7 describes the Security and performance analysis of the distributed key revocation schemes. Lastly, section 8 presents conclusion of the research work.

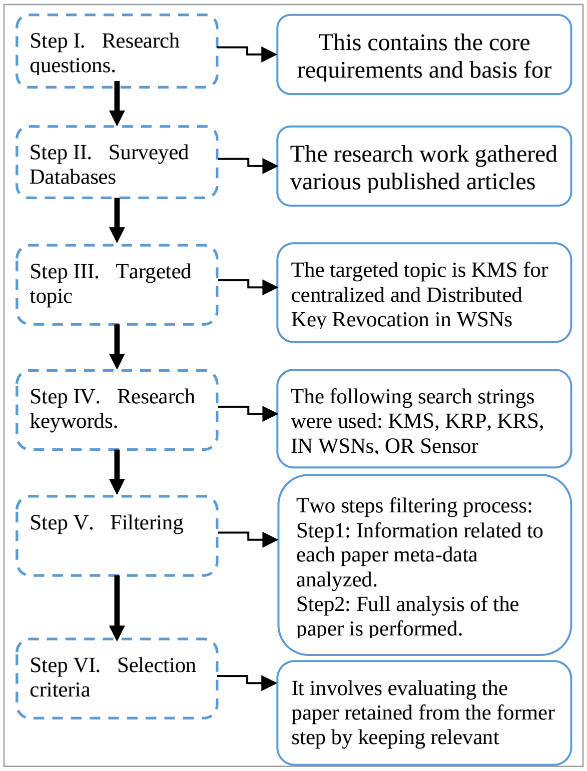
2.0 Material and methods

This section presents the methodology utilized to review key revocation schemes in WSNs. To perform a thorough review and analysis of the literature on key revocation methods, this study adopts a review methodology (Nabavi and Mousavi, 2018). The methodology process flow is presented in figure 2.

Figure 1. Methodology process flow diagram

The methodology aims at providing an all-inclusive overview as described in the following steps.

Step 1. Research questions: This contains the core requirements and basis for the contributions of the study. By proving answers to the following research questions.


Step I. Research questions.
This contains the core requirements and basis for
Step II. Surveyed Databases
The research work gathered various published articles
Step III. Targeted topic
The targeted topic is KMS for centralized and Distributed Key Revocation in WSNs
Step IV. Research keywords.
The following search strings were used: KMS, KRP, KRS, IN WSNs, OR Sensor
Step V. Filtering
Two steps filtering process: Step1: Information related to each paper meta-data analyzed. Step2: Full analysis of the paper is performed.
Step VI. Selection criteria
It involves evaluating the paper retained from the former step by keeping relevant

i. What are the security requirements needed in the major categories of key revocation schemes?
ii. What are the important performance metrics for comparison in centralized and distributed key revocation schemes?
iii. Which of the schemes outperforms in terms of the following parameters? energy consumption, computation time, storage cost, and communication overhead.
iv. What are the advantages and disadvantages of each of the representatives of the selected scheme?

Step 2. Surveyed databases: As a means of achieving the research objective. The work gathered various published articles spanning between 2003 to 2022, and also available on electronic databases, such as Science Direct, IEEE Xplore, ACM Digital library, SpringerLink, and from other domains like Google Scholar and Hindawi databases.

Step 3. Targeted topic: In this study, the targeted topic is "key management schemes for centralized and distributed key revocation in WSNs".

Step 4. Research keywords: To generate multiple strings for searching is an essential task. As such, the process must not leave anything from the research question. Combining different words can produce important expressions. For instance, the following search string is employed to find articles that relate to this study domain. "Key Management Schemes" OR "Key Management systems" OR "Key Revocation Protocols" OR "Key Revocation Schemes" IN "Wireless Sensor Networks" OR "WSNs" OR "Sensor Networks".

Step 5. Filtering: The filtering process is in two steps: (1) Information that relates to each paper's metadata is analyzed, concerning the title, abstract, and keywords, and (2) a Full analysis of the paper is performed.

Step 6. Selection criteria: The selection stage involves evaluating the papers retained from the former step, by keeping only relevant papers and excluding the irrelevant ones, duplicated and not written in English.

2.1 Categories of key revocation schemes in WSN

Different classification of key revocation schemes has been proposed in literature based on different criteria (Mall *et al*., 2013). This research identifies and presents categories of key revocation schemes in existence based on the work in (Wang *et al*., 2007). The classification of the protocols is in two broad categories, namely: Centralized and Distributed schemes as shown in figure 2. In key management systems, categorizing a scheme depends on the degree of its engagement with the central authority (CA). The CA is a base station or sink that can direct nodes in the network to discontinue association with any malicious node detected by the IDS.

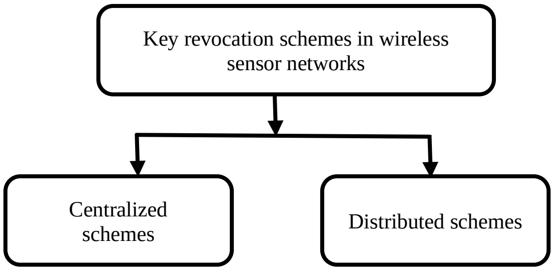


Figure 2. Categories of key revocation methods in WSNs

a. Centralized key revocation schemes

The protocol was first presented by (Eschenauer and Gildor, 2002). The approach is simple and involves a powerful central authority with adequate capability to monitor, detect and revoke the keying materials of compromised nodes (Chaib *et al*., 2016). The revocation decisions are held by a single assigned authority. The central authority is required to communicate with other sensor nodes to carry out the revocation task. In a situation a network node is detected to exhibit malicious behavior, the central authority must invalidate the compromised key and disconnect the node from the networks (Mall *et al*., 2013).

b. Distributed revocation schemes

In this scheme, a voting process is initiated based on the IDS report from the malicious nodes' neighbors, to determine the revocation decision. A revocation decision is taken only when the votes matched surpass the specified threshold (Mall *et al*., 2013). The node revocation process demands an alliance between nodes in the network. Thereby making the approach more complex, since the revocation decision is taken by many nodes. Accordingly, the distributed scheme has the benefit of working flexibility with a large-size ad hoc network.

3. Review of related literature

This section is divided into two subsections, the first subsection review articles that are related to the proposed research work. While the second subsection presents a general literature survey of relevant papers on key revocation schemes in sensor networks.

Many reviews on key management schemes have been conducted in the recent past (Nour *et al*., 2020). Prominent among them are the authors in (Mall *et al*., 2013; Nabavi and Mousavi, 2018; Moara-Nkwe *et al*., 2018). A review of modern distributed dynamic key management schemes in WSNs was proposed in Nabavi and Mousavi, (2018). The work investigates and discuses important requirements of distributed dynamic key management schemes in WSNs. The study highlighted their security and performance strengths and weaknesses. Finally, the work compared the reviewed schemes based on security metrics with emphasis on node revocations. The authors in (Gautam and Kumar, 2018) present a comparative analysis of current development in key management protocols, including details of major issues and future research directions. The work surveys key management advancement using a literature review and comparison of existing schemes. Gandino and Servetti (2019) presents and discusses the recoverability property, an important feature of security systems in WSNs. The work investigated state-of-the-art key management schemes, and their capacity to salvage secure communications, which follows the withdrawal of secret credentials of some nodes in the network. The study introduced recoverability analytical formulas, and the correctness of the presented formulas was validated through simulation. Hence, the study represents valuable support for the analysis of key management systems in WSNs. Similarly, the author in (Nithya, 2020) present a survey of various cluster-based key management schemes in WSN. The survey carries out a comparative analysis of selected schemes and the result indicated the need to focus on energy, communication, and memory overheads to design reliable scalable and adaptive key management schemes. Hussain and Kumar (2021) present a review of separate key agreement schemes in WBAN, concerning some identified attacks. The work discusses WBAN architecture, standard specifications, and security essentials. The study further classifies key agreement schemes and carries out an extensive review based on the following requirements, data confidentiality, authentication, data integrity, scalability, and forward and backward secrecy. The authors in (Dabhade and Alvi, 2021) discussed various aspects of WSN to understand the best design practice for WSN schemes. The study presents several WSN application techniques, summarizes security issues, and analyzed different key management schemes. However, the authors only discuss key revocation briefly. The work in (Khan *et al*., 2021) surveys the principal roots for motivating nodes to adopt selfish behavior and the solution for handling such nodes. Accordingly, experimental results show that the selfishness of nodes can be managed through the use of incentive-based or evolutionary-based mechanisms. However, the authors observed that the experiments are carried out based on their comprehension, therefore, cannot be considered perfect. Sheu et al., (2022) presents a server-less mutual authentication scheme for edge networks, the work solves security issues in autonomous devices and applications. The protocol utilizes public-key algorithm, challenge-response mechanism, identifier, time-stamps, and session keys. The approach eliminates the need for secret keys and reduces infrastructure requirements. The results evaluation shows the scheme's effectiveness when compared with other existing works. (Hegde and Andrew, 2023) presents a lightweight fault-tolerant secure data communication framework for WSNs that utilized ECDH and ECC primitives, together with Message Passing Interface (MPI) parallel program platform. Though, the technique demonstrates improved execution time and memory usage. Nevertheless, the assessment of the proposed technique is reduced to a small number of sensor nodes, and it is uncertain how the framework would work in larger-scale distributions.

3.1 Centralized category

Literature on centralized key revocation was foremost presented in (Eschenauer and Gligor, 2002). The scheme adopts the centralized key revocation category based on probability and a random graph model. Some of the strength of the scheme includes ease of implementation, scalability, and robustness. However, the scheme is not capable of providing adequate protection when extra nodes are compromised in the network. The work in (Dini and Savino, 2006) tried to improve and extend the work in (Eschenauer and Gligor, 2002) or apply it in different contexts. The work in (Yao *et al*., 2015) proposed a secure and efficient low-power group key management scheme based on LKH++. The scheme supports key renewal to enhance network security against node compromise and node capture attacks. However, the verification time for connectivity between cluster members and their cluster heads will lead to communication overhead. Mansour *et al*. (2014) proposed a centralized protocol for the revocation and renewal of compromised keys based on symmetric and asymmetric cryptographic mechanisms. The authors used Scyther to evaluate the security of their scheme. However, the key renewal aspect of the protocol cannot ensure message integrity.

The authors in (Guermazi *et al*., 2017) proposed a protocol for key management that used Diffie-Hellman key exchange, to address the initial key exchange problem associated with a symmetric cryptosystem. However, the execution time for the renewal of the session key is high. This is due to the extra public key encryption of the protocol. Won *et al*. (2017) proposed a suite of a cryptographic protocols to handle three separate communication scenarios. The scheme utilized a lightweight technique for node integration and revocation. Nevertheless, the evaluation result focuses only on performance parameters. Similarly, the work in (Wazid *et al*., 2019) proposed a secure key management and authentication technique for fog computing tag SAKA-FC. Nevertheless, the commutative communication overhead of the protocol is relatively high. The number of exponentiation operations is considered large, which will invariably result in high energy consumption. The work in (Zarezadeh and Mala, 2019) proposed a new method that aimed at improving the process of detecting the honesty of the accuser node on a nonhomogeneous poisson process. However, the performance evaluation of the scheme did not include some important efficiency metrics like computation, communication, and memory overheads. Furthermore, the study did not present a security evaluation of the procedure. The scheme ensures secure key revocations, with a single short broadcast message, to replace a linear number of unicast messages with several nodes.

3.2 Distributed category

As an improvement over the work of Eschenauer and Gilgor, the authors in (Chan *et al*., 2005) proposed a scheme that allows two nodes to share common keys to establish a secure bond between them. The scheme aimed at improving defense against attacks that used suitable values of common keys, when the number of compromised nodes is less than a crucial value. However, the scheme cannot ensure network connectivity among one-hop neighbor nodes due to the network scale. The authors in (Wang *et al*., 2008) proposed a key management protocol that can efficiently remove malicious nodes from the network. Experimental results show that the protocol is secure and efficient against malicious base stations and nodes. The work in (Zhang *et al*., 2009) proposed a simple but contentious idea that is built on pre-distribution and local collaboration-based group keys rekeying schemes. The scheme updates the compromised group keys and preloads future group keys in sensor nodes before deployment. However, the major setback of the scheme is that each node in the network must save the secret shares of its direct neighbor's e-polynomials. which may not be feasible in a network with a compact deployment of sensor nodes.

The authors in (Zhang *et al*., 2011) presents a distributed deterministic energy-efficient key management scheme for WSN. The scheme emphasizes pairwise key, local clusters key establishment and protection. The scheme is highly flexible and has a very low-performance overhead. In another study, the work in (Rehman *et al*., 2022) concentrate on the security of wireless sensor networks (WSNs) using blockchain technology, to address the privacy and security challenges in IoT systems. Highlights the gains of employing blockchain in WSNs, such as ensuring high-level security, non-rewritable data transmission, and resistance against attacks like self-mining was presented. Nevertheless, the possible impact of blockchain on the scalability and resource constraints of wireless sensor networks, that could be a major drawback in real-world applications need to be address. Zhang and Cao, (2023) presents a framework for authenticating edge computing Internet of Things (IoT) devices to establish secure communication between devices and edge servers, and also between devices themselves. The work fulfils wide range of security attributes and resist several security threats through formal and informal security analysis. Additionally, the protocol demonstrates good performance in terms of computation and transmission consumption.

4. Selected representatives of the key revocation schemes

For both representatives of the key revocation protocols considered. The selection was based on whether the task of key revocation has been designated to multiple nodes or a single node. Vis a vis their level of emphasis on key revocation and renewal. Other criteria include their resilience to revocation attacks and timely completion of the revocation procedure.

Note that, the names assigned to each selected candidate are at the discretion of the authors as featured in the title of the selected schemes.

4.1 Centralized Key Revocation Scheme

i. Key management and distribution framework (KMMR)

The KMMR protocol was proposed in (Guermazi *et al*., 2017). The scheme gives priority to IDS activities in the base station in other to offer secure communication in the sensor network. A packet labeled Secure Report Messages (SRM) that comprise local events and monitored information from neighbors is encrypted with the individual key and then send to the base station by the sensor nodes.

$$SRM(S \rightarrow BS): RouterIDTimestamp, Enc_{INK}, MAC_{PWK}(M) \quad (1)$$

The moment compromised nodes are detected in the IDS server. An authenticated broadcast of an alarm broadcast message (ABAM) containing a blacklist, is broadcasted by the base station. The delayed disclosure of the hash key K_{j+1} authenticates the ABAM message. The inclusion of the blacklist in the ABAM, permits the sensor node's speedy discovery of the malicious nodes list.

$$ABAM(BS \Rightarrow^{*}): K_j, Enc_{GBK}(Backlist), MAC_{Kj+1}\ MAC_{LBK}\ (M) \quad (2)$$

Upon the discovery of the downward transmitting traffic of the ABAM sufficiently out of reach. The base station then sends the Hash key Disclosure of Alarm Message (HKDAM). As soon as the hash key K_{j+1} is unveiled, it becomes feasible for sensor node S to verify the authenticity of the ABAM. In the first instance, the healthful node deletes the PWK shared with the malicious nodes and then computes a fresh LBK. Consequently, to share the new LBK, a Secure key Update Message (SKUM) defended with the PWK is transmitted to each healthy neighbor.

$$S'SKUM(S \rightarrow S'): NonseS, Enc_{PWK}(LBK), MAC_{PWK}(M) \quad (3)$$

In conclusion, the base station advanced with the GBK update.

ii. Key revocation and renewal protocol (KRRP)

The work in (Doer *et al*., 2019) presents a symmetric and asymmetric cryptographic key revocation and renewal protocol for WSNs. Before the nodes are deployed in the network, each node is loaded with a public and private pair of keys expressed as $pk(N)$ and $sk(N)$ and the public key of sink $pk(S)$. The IDS result establishes the bases for identifying malicious nodes by the base station, the step marks the beginning of the revocation process. Once a compromised neighbor of node N is detected by the sink, a message comprising a list of malicious nodes and a nonce ns encrypted with $K_{DH}\ (S,N)$ is transmitted as a revocation message to node N.

Upon receiving the revocation message by node N, the season keys of all its neighbor's nodes on the revocation list are deleted. And a nonce encrypted with $K_{DH}(N,S)$ is communicated back to the sink as a confirmation of the receipt of the revocation message. Subsequently, the network keys must be renewed in the event of any network compromise. Thus, a new network key $NK'$and a nonce $n_{(si)}$ is computed for each non-compromised node I, encrypted with $K_{DH}\,(S, I)$. The sink observed a waiting period to receive all nonces from node I before it commences the use of the new network keys.

iii. Efficient pairwise and group key management protocol (PGMP)

The work in (Rahman and Sampalli, 2015) proposed a scheme, in which the base station initiates the revocation process, on the bases of a list of compromised nodes transmitted from the IDS in the followings.

a. The base station computes a random $(\lambda + 1) \cdot (\lambda + 1)$ symmetric matrix Ḋ over $GF_{(q)}$. Then computes Ň $\cdot$ $(\lambda + 1)$ matrix Ä = (Ḋ $\cdot\, G)^{\intercal}$. Where (Ḋ $\cdot\, G)^{\intercal}$ is the transpose of (Ḋ $\cdot\, G$).

b. Generate a bit vector ȷ of length Ň, where Ň represent the nodes count in the network, given that, ȷ[ĭ] = 0, for node ID ĭ is in Ꞧ, otherwise 1.

c. Base station computes Ň $\cdot$ $(\lambda + 1)$ matrix H = Ä +A, and performs Row_{i} (H) $\cdot$ ȷ[ĭ], where Row_{i} (H) are the elements in the ith row of H.

d. The base station again computes $M_{ĭ}$ = ĭ Row_{i} (H)]\\MAC[\\Kĭĭ, ĭ\\Ëk Row_{i} (H))], for each non-revoke node ĭ.

e. On the final note, the base station computes M = M\\$M_{ĭ}$, where ĭ ∉ Ꞧ and broadcast Ꞧ\\M. And updates D as Ḋ for the forthcoming revocation.

f. After the non-revoked node ť receives the revocation broadcast, it first takes out its share from the message that was broadcasted ť $Row_{ť}$ (H)]\\MAC[ť \\ Ëk $Row_{ť}$ (H))], it then computes.

Ꝟ= MAC[ť \\ Ëk $Row_{ť}$ (H))] (4)

Where the node's private share is determined as ℂol$_{\intercal}$ (G) $\cdot$ $Row_{ť}$ (A), if Ꝟ corresponds with the received MAC, hence, it shows that the broadcast is from the base station. Thereafter the encrypted part of the massage $Row_{ť}$ (H))] is decrypted by the node with. ťť, then brings up to date its private share $Row_{ť}$ (A) = $Row_{ť}$ (H) - $Row_{ť}$ (A). Else, it rejects the broadcast message.

iv. Energy-aware key management framework (EKMF)

The scheme was proposed by (Omar *et al.*, 2018) The detection of the compromised node is performed through IDS. In the event, the IDS detects a sensor node Πs is compromised. The base station extracts from the list $\mathcal{L}$. Then notify the cluster head ($\mathbb{C}_h$) supervising the sensor node Πs by sending a message (ComPromiD_noDe (Πs)) to the$\mathbb{C}_h$. The $\mathbb{C}_h$ removes the store public key of the sensor node Πs, and commences the rekeying process of the cluster key. Based on this, the $\mathbb{C}_h$ encrypts a fake key $(\mathrm{F}_k)pk$ to disconnect the compromised sensor, by blocking the links that it can use in establishing a connection with the $\mathbb{C}_h$. Also, if the base detects the $\mathbb{C}_h$ is compromised, the base station notifies the sensor nodes supervised by the $\mathbb{C}_h$ via a message (ComPromiD_$\mathbb{C}_h$). Lastly, the sensor nodes withdraw the stored $\mathbb{C}_h$'s public key, and cluster key and move to join another cluster.

4.2 Selected representatives of the distributed key revocation protocols

i. Hierarchical key management scheme with probabilistic security (HKMS)

The scheme was proposed in (Albakri *et al.*, 2017). In this revocation approach, the cluster head $\mathbb{C}\mathcal{H}$ monitors the activities of sensor nodes and detects the misbehaving ones. The IDs of the detected misbehaving nodes are added to the node revocation list (NRL) of the $\mathbb{C}\mathcal{H}$. The CH must update the local broadcast key $\mathrm{K}_{\mathcal{L}\mathcal{B}}$, and then transmit it to the non-compromised node in its cluster independently.

Therefore, the updated NRL encrypted with the local broadcast key $\mathbb{E}k_{\mathcal{L}\mathcal{B}}$(NRL) must be transmitted by the $\mathbb{C}\mathcal{H}$ to all the nodes in its cluster. Hence, all the non-compromised nodes can decrypt the message with the aid of the local broadcast key. Which authenticates the $\mathbb{C}\mathcal{H}$ and updates the NRL. Thereafter, a request is sent to the sink by the $\mathbb{C}\mathcal{H}$ to update the global broadcast key $\mathrm{K}_{\mathbb{G}\mathcal{B}}$, And also send an updated NRL encrypted with a pairwise key K_{sc} as $\mathbb{E}k_{sc}$(NRL) to the sink. The sink conducts an authentication check on the received message from the $\mathbb{C}\mathcal{H}$. Updates its NRL and establishes a new global broadcast key. The GBK is encrypted with the pairwise key between the $\mathbb{C}\mathcal{H}$ and the sink, collectively with each $\mathbb{C}\mathcal{H}$ in the network, denoted by $\mathbb{E}k_{sc}(\mathrm{K}_{\mathbb{G}\mathcal{B}})$. As soon as each $\mathbb{C}\mathcal{H}$ receives the message sent by the sink. It uses the pairwise key to authenticate the received message and updates the global broadcast key.

ii. Multi-basestation key management protocol (MKMP)

The protocol was proposed by the authors in (Ferng *et al.*, 2014). The scheme used a random polynomial mechanism. The revocation process starts with the generation of random polynomials $\mathbb{S}_{total}$ of $\mathbb{S}_{total} > t$ degree t for each sensor node in the compromised cell. Where $\mathbb{S}_{total}$ represents the attempted number of revocation sessions against a targeted node in the compromised cell. Consequently, a revocation votes from sensor node $\mathcal{U}$ is loaded against sensor node $\mathcal{V}$, based on the presented random polynomial $q_v(\mathcal{X}_{u,v})$

for node $\mathcal{U}$ and $\mathcal{V}$, and each revocation session against target node $\mathcal{V}$. The revocation vote cast comprises secret share $(q_v(\mathcal{X}_{u,v}),\ \mathcal{X}_{u,v})$ encrypted with $\mathcal{K}_{lu}$. Hence, $\mathbb{E}\mathcal{K}_{lu}\ (q_v(\mathcal{X}_{u,v}), \mathcal{X}_{u,v})$ represents the loaded information. The encryption process guarantees every node’s ability to revoke sensor nodes around its neighborhood. Furthermore, for every vote, a log t authentication hash value is loaded for the Merkle tree with leaves $(q_v(\mathcal{X}_{i,v}), \mathcal{X}_{i,v})$. For sensor node i of sensor node $\mathcal{V}'$s neighbors within a similar cell. To permit the neighbors of sensor node $\mathcal{V}$ to validate the authenticity of the revocation votes cast. the hash value of the Merkle tree is computed and equated with the known root value of node $\mathcal{V}$, and a log t authentication value is attached to the message.

iii. Novel distributed key revocation scheme (DKRS)

The distributed revocation protocol proposed in (Chao *et al*., 2013) consists of four phases, that is, offline initialization, connection establishment, voting, and revocation completion.

Offline initialization: In this stage, public and private matrices are computed for every node in the network. The public matrices enclose freely accessible public information for verification of votes for every voting session. During each session, the encrypted votes from the matrices and the related column information of the public matrices are preloaded by the nodes. Consequently, to confirm the authenticity of the revocation verdict conceived for the compromised node, each node stores a set of hash values.

Connection establishment: When a message is received, showing a vote against a compromised node from one of their neighbors. Or by one of the nodes, clearly identifying a compromised node with unusual behavior, the node moves into the connection establishment phase. The phase involves the exchange of activation masks utilize to decrypt the votes among the participating and targeted nodes at the start of every session.

Each participating pair of nodes has information on the vote cast against the other. Thus, the votes are encrypted with their counterpart stored mask, hence a mutual substitute of the mask is needed to establish a connection. The participating nodes easily drop the links between them and the compromised node, after several attempts to establish a connection. Thereby causing a decline in the node degree. Furthermore, the moment the degree drops below the predestine benchmark, the node is discarded by a central degree-counting method.

Stages of the revocation session: In this session $\mathcal{S}$, the nodes' positions are defined in two specific modes, that is, the waiting and the voting states. Each participating node J is programmed to a waiting state at the commencement of each session $\mathcal{S}$. It then waits to receive a voting message from any of its neighbors. The transition from the waiting state to the voting state is in two stages: (1) when the node receives the initial vote message of any of the other participants (2) the node directly detects the malicious conduct of the target node.

On successful switchover from a waiting state to a voting state, Participant J begins its session timer $\triangle_S$ instantly. The predefined value of the timer $\triangle_S$ varies on the anticipated time to attain a revocation verdict. In the event the timer expires, the participant goes back to the waiting state for subsequent session $\mathcal{S} + 1$ and ends the revocation process for session $\mathcal{S}$.

Voting in revocation session: When the local participants through close monitoring discover misbehavior in the target node. Each local participants initiate and exercise their voting role by casting their votes against the malicious node in both sessions. To avoid the possibility of losing any vote cast close to the expiry time of the timer, due to the broadcasting delay. **Completion of revocation process:** After the voting phase. All the neighboring nodes of the compromised sensor proceed to the completion stage of the process. First, they compute the received votes during the voting session individually, and in any case, one of the nodes records fewer $\mathbb{t}$ votes against a certain compromised node. Then, it resolves the secret-sharing information of the node and calculates the corresponding hash value. Thereafter, the revocation initiator broadcast a revocation message to all the nodes in the network. The message-receiving nodes verify the authenticity of the message by comparing the stored hash value in their memory with the revocation message. In the event, the message is verified authentic.

iv. A CFL-based key management scheme (CFLRS)

The scheme was proposed in (Zhang *et al*., 2021) it is divided into cluster member revocation and cluster head revocation. In the cluster member revocation, it becomes crucial to revoke every that relate to the L-sensor node and bring up to date the routing structure about the node whenever the L-sensor node is destroyed in the cluster. The message for the revocation comprises the list of keys for revocation. The list is signed with the private key ID (IDSK) which is appended to the key list. Each L-sensor node also possess different identification public-private key pair. Therefore, any revocation message received is confirmed through the identification public key (IDPK) to confirm to confirm the integrity and authenticity of the message to deter adversary from transmitting false revocation messages.

5. Security and performance analysis of the centralized key revocation schemes

This section presents the security analysis of the reviewed centralized protocols. The evaluation discussion centered around vital security attacks associated with CIA security requirements. As stated in section 5, acronyms to the assigned names of each selected candidates will be used to describe each protocol in throughout the analysis.

1. Security analysis: The analysis is carried out based on the CIA goals and the attendant attack scenarios. In performing the analysis, the study assumed the selected schemes are equipped with a system for detecting compromised nodes. However, a major setback with the centralized methods is the single point of failure. This setback creates the opportunity for an attacker to impersonate the central authority and commence revocation attacks. The use of authenticated broadcast messages in KMMR, KRRP, and PGMP guarantee the scheme's resilience to revocation attacks. To ensure the freshness of every transmitted data, and to protect against the replay of old messages by an adversary. Both KMMR and KRRP use nonce to prevent replay attacks. However, PGMP protocol utilizes a message digest attached to the broadcast message to defend against both spooled and replay attacks. Similarly, during PGMP's group creation, the broadcast message is bound with different timestamps to prevent the replay of old messages. To defend against node capture attacks, KRRP and EKMF protocols ensure timely updates of the session key. Also, the KMMR scheme generates and exchanges keys on the fly, while the temporary keys are erased from the node's memories after the key distribution session. In the same way, PGMP and EKMF can protect against node compromise attacks, since the network nodes only store the partial key materials. The secure node addition mechanism of KRRP and EKMF protocols guarantees their resilience to sybil and sinkhole attacks. The PGMP scheme is resistant to perfect forward and backward secrecy. Due to unique and random group key $G_{\mathcal{K}}$ updates, anytime a node is added or deleted to or from the group. Furthermore, to defend against DoS attack PGMP protocol encrypt and attach MAC to all its broadcast and unicast messages to identify any modification.

In conclusion, only KRRP protocol was observed to have implemented its security mechanism with a standard cryptographic primitive. Therefore, the scheme is expected to ensure end-to-end data security and privacy. Table three (3) presents a comparative summary of some attacks associated with WSNs. The attacks include node capture attack, replay attack, revocation attack, and forward and backward secrecy attack.

Yes, NO, Both, or Forward indicates whether the reviewed schemes can defend against or not any of the stated attacks. While NA indicates the information is not available.

Table 2. Comparative summary of security analysis of centralized key revocation schemes

Schemes	Node Capture Attacks	Replay Attacks	Revocation Attacks	Forward and Backward Secrecy
KMMR	Yes	Yes	Yes	Forward
KRRP	Yes	Yes	Yes	Both
PGMP	Yes	Yes	Yes	Both
EKMF	Yes	NA	Yes	NA

2. **Performance analysis:** Key management performance measurement is based on a set of evaluation metrics. Performance metrics are essential for effective analysis and documentation of performance gap and their causes (Kumar *et al.*, 2021). Typical metrics such as the message count exchanged to change cryptographic keys, the number of required encryption keys, operational cycle and the encryption key sizes are major metrics when measuring performance efficiency of any categories of key revocation schemes.

This section presents discussions and analysis to evaluate the performance of the selected representative of the centralized key revocation schemes based on the evaluation criteria stated above. The analysis is in terms of computation, communication, storage, and energy overhead. Accordingly, the work compares the selected schemes based on the stated performance requirements.

i. **Computation overhead**: The computation time is determined by the complexity and the number of cryptographic operations performed in the computational related task of the considered algorithm in a scheme. For instance, operations such as $\boldsymbol{T_{enc}/T_{dec}}$, $\boldsymbol{T_{xor}}$, $\boldsymbol{T_{\mathcal{H}}}$, $\boldsymbol{T_{rnd}}$, $\boldsymbol{T_{mul}}$, $\boldsymbol{T_{KeyGen}}$, $\boldsymbol{T_{\lambda+1}}$ denotes the operating time of symmetric encryption/decryption, exclusive-OR, hash function, random number generation, multiplication, key generation, and polynomial respectively. In KMMR, to achieve secure node revocation, less complex symmetric algorithm is utilized for the encryption and decryption processes. In the revocation process, each neighbor erases the pairwise key it shares with the compromised node and updates the local broadcast key. The overall encryption and decryption for the node revocation are $2(d-1)\cdot(d-1)=2(d-1)^2$. For KRRP, the size of the revocation list determines the execution time of the protocol. In this context, the central authority (sink) sends a list of the revoked nodes and a nonce encrypted with the recipient node's public key. The recipient node decrypts the message and sends back an acknowledgment message containing it nonce, encrypted with the public key of the sender. Thus, the overall computation overhead of KRRP is $2(d-1)\cdot(d-1)$. The overhead computation in PGMP arises from when each node carries out a polynomial evaluation of $(\lambda+1)$ terms, that involves multiplication and addition operations of $(\lambda+1)$. In general, $\lambda-(64-bit\times 64-bit)$ modular multiplications are required to compute the polynomial since ɋ has been chosen to be 64-bits. The study presents in table 4, different cryptographic parameters and the operations time utilized in the protocols under review. The computation cost of the reviewed centralized schemes is compared in the table 4. In the evaluation results shown in table 4, KRRP presents the lowest computation time of 4.69s as against a distant 11.02s, 13.37s, and 75.75s presented by KMMR, EKMF and PGMF respectively. Thus, KRRP protocol achieved better computation time than the related schemes in table 4.

Table 3. Execution time of the different cryptographic operations performed by various schemes.

Cryptographic parameters	$\boldsymbol{T_{enc/dec}}$	$\boldsymbol{T_{xor}}$	$\boldsymbol{T_{\mathcal{H}}}$	$\boldsymbol{T_{rnd}}$	$\boldsymbol{T_{mul}}$	$\boldsymbol{T_{KeyGen}}$	$\boldsymbol{T_{\lambda+1}}$
Operation time (ms)	2.69/2.82	2	0.054	2.78	1.63	2	74.07

Table 4. Comparative summary of computation analysis of the centralized key revocation schemes

Schemes	KMMR	KRRP	PGMP	EKMF
Computation	$\boldsymbol{2T_{enc}+2T_{dec}}$	$1T_{enc}+1T_{xor}$	$1T_{\mathcal{H}}+1T_{\lambda+1}+1T_{\boldsymbol{mul}}$	$3T_{rnd}+2T_{enc}$
Total Cost (ms)	11.02	4.69	75.75	13.37

ii. Communication overhead: The communication efficiency of the considered schemes was measured using the length and sizes of the messages transmitted by communicating entities. The incurred communication overhead in KMMR scheme comes from the number of messages evolving from the distribution of keys. During the creation of the network structure. Each message sent comprises of various fields, such as an address, key-value, nonce, timestamp, tier, and MAC fields, that vary in size between 8 and 32 bytes. In KRRP protocol, to carry out the revocation process. The sink sends a revocation request to node $\mathcal{R}$, in the same manner, the recipient node $\mathcal{R}$ sends an acknowledgment message back to the sink. Node $\mathcal{R}$ sends a new session key to node N in his communication range. And an acknowledgment message is sent to node $\mathcal{R}$ confirming the receipt of the new session key. In PGMP protocol, the pairwise key generation process does not require the key agreement phase by the concerned nodes. Hence, the process did not require message exchange. Based on the above description and the displayed results in table 6, the protocol EKMF presents the lowest communication overhead of 16 bits as against its competing KRRP 160bits. Accordingly, PGMP and KMMR recorded the highest communication overhead of 208 bits and 384 bits respectively.

iii. Storage overhead: The analysis of the storage cost of the considered solutions shows that. In KMMR protocol, each node is expected to store the following keying materials, LBK, PWK, TK, Nonce, and GBK. The number of keys stored in KMMR is always the same, notwithstanding the size of the network. Also, some of the keying credentials are stored in the flash memory, hence, the small memory requirement in KMMR. In KRRP, before deployment, each node needs to store in its memory a pair of secret and public keys, the public key of the sink, and a shared key with the sink. The PGMP protocol demands each node in the network to store preloaded secret materials, in the form of $(\lambda + 1)\ 64 - bits$ numbers. Given that, $\lambda = 100$, then each node in the network needs to store $(101\times\ 8bytes)$ in its memory. Memory requirement in EKMF demands each node to store a pair of its public and private keys, node ID, and the ID of the cluster head. In Table 8, the results show the protocol PGMP achieved a better storage overhead of 128 bits against 160 bits, 400 bits and 560 bits presented by KRRP, EKMF, and KMMR protocols respectively.

iv. Energy overhead: The implementation of KRRP and EKMF protocols was based on a symmetric and asymmetric algorithm (AES and ECC). These primitives are energy-demanding, thus KRRP requires more energy when compared with other selected solutions, like KMMR which was implemented based on AES. Meanwhile, PGMP protocol has negligible data communication cost, which is relative to the low power consumption of 0.09mj presented. In table 9, the work presents a comparative summary of the energy overheads as observed in the review. The results displayed in table 9 shows that only PGMP and KRRP protocols presents 0.09mJ and 211.11mJ energy overheads respectively.

Table 9. Comparative summary of energy analysis of the centralized key revocation schemes

Schemes	KMMR	KRRP	PGMP	EKMF
Energy cost.	NA	211..11mJ	0.09mJ	NA

5.I Security and performance analysis of distributed key revocation schemes

1. Security analysis: The study analysis will consider the fundamental security goals of confidentiality, integrity, availability, and authenticity. Concerning their associated attacks, as a basis for our discussion. Accordingly, in HKMS, the pre-distribution of tokens and other keying materials ensures the schemes' resilience to key exposure and node capture attacks. Similarly, the DKRS scheme is also resilient to node capture attacks. However, further details on the security evaluation of the scheme were not provided. The MKMP scheme is robust to most regular attacks on routing such as, node replication attacks, Sybil attacks, and wormhole attacks. The protocol can defend against black hole and selective forwarding attacks. Since the routing decision is determined by the source based on location. In HKMS scheme, the pre-distribution of tokens and other keying materials makes the scheme resilient to key exposure attacks. Since there are no keys transmitted across the network. Furthermore, to defend against replay attacks, the scheme utilizes nonce for each data transmission. Similarly, the use of authenticated broadcast of packets by MKMP guarantees the protocols' resistance to sybil attack. More so, the protocol provides location-aware keys to authenticate each of the packets it sends. In addition, the use of a location-aware key ensures MKMPs' defense against node replication attacks. The CFLRS is design to be computationally hard, thus intercepting some component of the network cannot make the adversary breaks the entire key system. The analysis in table 10 shows that all the protocols are resilient to revocation attacks except DKRS. However, all the analyzed protocols are secured against node capture attacks.

Table 10**.** Comparative summary of security analysis of the distributed key revocation schemes

Schemes	Node Capture Attacks	Replay Attacks	Revocation Attacks	Forward and Backward Secrecy
HKMS	**Yes**	Yes	Yes	NA
MKMP	**Yes**	NA	Yes	Forward
DKRS	**Yes**	NA	NA	NA
CFLRS	**Yes**	**NA**	**Yes**	NA

2. Performance analysis of distributed key revocation schemes

i. Computation overhead: the analysis of the computation cost is based on the complexity and the number of related cryptographic operations on the utilized parameters. Examples of operations like pairing, scalar multiplications, additions, encryption, decryption, and verifications represented as $\boldsymbol{T}_{pair}$, $\boldsymbol{T}_{mul}$, $\boldsymbol{T}_{add}$, $\boldsymbol{T}_{enc/dec}$, $\boldsymbol{T}_{vrf}$ respectively. In HKMS, the computation requirements of the scheme are evaluated based on the operation cost of revocation check and signature cost, that is, pairing evaluation of 1P and 6G respectively. Hence, the scheme HKMS is only required to do a pairing evaluation. However, the computation requirements of DKRS include, vote verification, which demands a dot product operation that involves t multiplication and $t-1$ additions. Thus, the computation overhead is only $\mathcal{O}(1)$. Incidentally, the MKMP scheme did not provide details of the scheme's computation overhead. Nevertheless, the computation activities in the scheme include, the generation of $\mathbb{S}_{total}$ for each node, encryption, decryption, authentication, and verification of each vote cast against the targeted node. In CFLRS, the computation overhead is calculated based on the bilinear pairing during key generation. Based on the execution time of the cryptographic operations in table 11. The computation comparison results in table 12 shows DKRS protocol achieved 2.6ms better computation time as against 4.5, 5.51ms and 8.1ms presented by CFLRS, MKMP and HKMS protocols respectively.

Table 11. Execution time of the different cryptographic operations performed by various schemes.

Cryptographic parameters	$\boldsymbol{T}_{enc/dec}$	$\boldsymbol{T}_{add}$	$\boldsymbol{T}_{\mathcal{H}}$	$\boldsymbol{T}_{pair}$	$\boldsymbol{T}_{mul}$	$\boldsymbol{T}_{KeyGen}$	$\boldsymbol{T}_{vrf}$
Operation time (ms)	2.69/2.82	2	0.054	4.5	0.6	2	74.07

Table 12. Comparative summary of computation analysis of distributed key revocation schemes

Schemes	HKMS	MKMP	DKRS	CFLRS
Computation	1P + 6G	$\mathbb{E}\mathcal{K}_{lu}\left(q_v(\mathcal{X}_{u,v}), \mathcal{X}_{u,v}\right)$ Enc/Dec	$1\boldsymbol{T}_{add} + 1\boldsymbol{T}_{mul}$	$1\boldsymbol{T}_{pair}$
Total Cost (ms)	8.1	5.51	2.6	4.5

ii. Communication overhead: The communication overhead of HKMS scheme comprises shared pairwise keys establishment. A node is preloaded with share credentials, hence, no need to transmit any data other than the node's ID that produces no overhead. Thus, the scheme has low communication overhead, except for updating the broadcast keys. That involves transmitting new broadcast keys to each sensor node with the aid of the new pairwise keys. In DKRS scheme, each voting message carries only t elements of one row in the vote matrix and t elements of one column in the public matrix. Thus, the communication overhead of DKRS is $\mathcal{O}(1)$. However, no performance analysis on the communication cost of MKMP scheme was presented. In CFLRS, the communication overhead is the total length of the sent and received messages in the protocol, thus, a total of $3G_1$ for sent messages and $4G_1$for received messages was observed. Given the above analysis and the results displayed in table 14, the protocol DKRS achieved 2.74 bits better communication efficiency as against the 16 bits and 64 bits presented by HKMS and MKMP respectively.

Table 13. Exchanged parameters for communication by various schemes.

Exchanged parameters	l_{ns}	$l_{\mathcal{PWK}}$	$l_{pk/prk}$	l_{ID}	l_{ts}	l_{mac}	l_{sk}	$l_{\lambda+1}$
Length (bits)	64	64	256	16	32	128	128	64

Table 14. Comparative summary of communication analysis of the distributed key revocation schemes.

Schemes	HKMS	MKMP	DKRS	CFLRS
Communication size	$\mathcal{ID}$s	$q_v(\mathcal{X}_{u,v})$	$\mathcal{O}(\log m)$	$7G_1$
Total cost (bits).	16	64	2.74	NA

iii. Storage overhead: For HKMS scheme, the sink stores $kt\mathfrak{h}$ coefficients in $\mathcal{GF}_{(\mathbb{P})}$, and each sensor node is required to store the $t + h$ coefficient in $\mathcal{GF}_{(\mathbb{P})}$. Thus, the sensors located at the lowest level of the network store the minimal number of coefficients. In the same way, the sink that is located at the highest level is required to store most coefficients. For DKRS scheme, each node stores $\mathcal{O}\ (\mathbb{S}_{total}\ .m\ .t)$. But the value of t does not depend on the network size and has a small constant value that is far less than m. Hence, the per-node memory need is approximately $\mathcal{O}\ (\mathbb{S}_{total}\ .m)$. In MKMP scheme, the storage cost of each node in the network is $m + 2t + 5$. Except for the base station that stores only one master key $\mathcal{K}_m$. Based on the fact that, the remaining keys can be obtained from the node's ID and location information captured in the data reports. The results displayed in table 16, shows that only MKMP and HKMS protocols presents results for storage overheads. Accordingly, the protocol MKMP achieved 74 bits storage efficiency as against the 128 bits storage cost in HKMS.

Table 15. Storage parameters for memory cost in various schemes

Storage parameters	$l_{\mathbb{S}_{total}}$	$l_{\mathcal{PWK}}$	$l_{pk/prk}$	l_{ID}	l_{msk}	l_{mac}	l_{sk}	$l_{\lambda+1}$
Length (bits)	NA	64	256	16	64	128	128	64

Table 16. Comparative summary of storage analysis of the distributed key revocation schemes

Schemes	HKMS	MKMP	DKRS	CFLRS
Storage	$2(t + h)$	$\mathcal{K}_m(m + 2t + 5)$	$\mathcal{O}\ (\mathbb{S}_{total}\ .m)$	$m * \mathbb{N} + 3\mathbb{N}$
Total cost (bits).	128	74	NA	NA

iv. Energy overhead: Although, the performance analysis of the energy cost of the considered distributed schemes was not presented in the studies. Therefore, the energy cost of every scheme's process execution is a function of its computation and communication overhead. However, most of the considered protocol in this study did not present the energy overhead of their studies, except CFLRS that describe the energy cost of their work as having significant advantage over other schemes in their study.

6. Conclusion

This paper presents and discusses the backgrounds of key management systems in WSNs. A literature survey on key revocation schemes was conducted with emphasis on four identified categories viz centralized, distributed, decentralized, and hybrid. The work presents an overview of the identified categories, including some major features and taxonomy. Furthermore, a thorough review of some selected candidates of centralized and distributed schemes was carried out, and a comparative analysis of security and performance requirements was discussed. Findings from the review show that there are problems with performance parameters in almost all the schemes reviewed. Therefore, the need for efficient key revocation techniques and the use of resource-considerate algorithms are open issues for the research community. As a future work, the review can be extended to other categories of key revocation mentioned.

Declarations

Ethics approval and consent to participate

Not Applicable

Consent for publication

All authors have read and consented to the submission of the manuscript.

Availability of data and material

Not Applicable.

Competing interests

All authors declare no competing interests.

Funding

There was no funding for the current report.

References

Ahlawat, P. and Dave, M. (2021). An attack-resistant key pre-distribution scheme for wireless sensor networks. *Journal of King Saud University-Computer and Information Sciences*, 33(3), 268-280. https://doi.org/10.1016/j.jksuci.2018.03.002

Albakri, A., Harn, L., Song, S. (2019). Hierarchical key management scheme with probabilistic security in a wireless sensor network (WSN). *Security and communication networks*, *2019*. Chaib N, Lagraa N, Yagoubi MB, Lakas A. (2016). SDRP: a secure distributed revocation protocol for vehicular environments. Security and Communication Networks, 9(4), 279-289. https://doi.org/10.1002/sec.561

Chan, H., Gligor, V. D., Perrig, A., Muralidharan, G. (2005). On the distribution and revocation of cryptographic keys in sensor networks. *IEEE Transactions on dependable and secure computing*, *2*(3), 233-247.

Chao, C. H., Yang, C. F., Lin, P. T., Li, J. S. (2013). Novel distributed key revocation scheme for wireless sensor networks. *Security and Communication Networks*, *6*(10), 1271-1280.

Chinniah P, Krishnamoorthi S. (2019). An Efficient Elliptic Curve based Key Management Scheme for Distributed Sensor Networks. European Journal of Engineering Research and Science, 4(6), 111-116.

Dabhade, V. D. and Alvi, A. S. (2021). Review of wireless sensor network security schemes. In *Intelligent Computing and Networking*, pp. 41-51. Springer, Singapore. https://doi.org/10.1007/978-981-15-7421-4_4

Dini, G. and Savino, I.M. (2006) An efficient key revocation protocol for wireless sensor networks. In 2006 International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM'06), pp. 3-pp. IEEE. https://doi:10.1109/wowmom.2006.23

Doerr, L., Heigl, M., Fiala, D., Schramm, M. (2019). Comparison of energy-efficient key management protocols for wireless sensor networks. In *Proceedings of the 1st International Electronics Communication Conference* (pp. 21-26).

Eschenauer L, Gligor V. D. (2002). A key-management scheme for distributed sensor networks. In Proceedings of the 9th ACM conference on Computer and communications security, pp. 41-47. https://doi:10.1145/586110.586117

Ferng, H. W., Nurhakim, J., Horng, S. J. (2014). Key management protocol with end-to-end data security and key revocation for a multi-BS wireless sensor network. *Wireless networks*, *20*(4), 625-637.

Gandino, F. and Servetti, A. (2019). Key recoverability in wireless sensor networks. *IEEE Access*, *7*, 164407-164417. https://doi:10.1109/ACCESS.2019.2952945

Gautam, A. K., and Kumar, R. (2018). A comparative study of recently proposed key management schemes in the wireless sensor network. In *2018 International Conference on Computing, Power and Communication Technologies (GUCON)*, pp. 512-517. IEEE

Guermazi, A., Belghith, A., Abid, M., Gannouni, S. (2017). KMMR: An efficient and scalable key management protocol to secure multi-hop communications in large scale wireless sensor networks. *KSII Transactions on Internet and Information Systems (TIIS)*, *11*(2), 901-923.

Hegde, M., and Andrew, J. (2023). A Lightweight Authentication Framework for Fault-tolerant Distributed WSN. *IEEE Access*.

Huanan Z, Suping X, Jiannan W. (2021). Security and application of wireless sensor network. *Procedia Computer Science*, *183*, 486-492. https://doi.org/10.1016/j.procs.2021.02.088

Hussain, S. Z. and Kumar, M. (2021). Secured Key Agreement Schemes in Wireless Body Area Network-A Review. *Indian Journal of Science and Technology*, *14*(24), 2005-2033. https://doi.org/10.17485/IJST/v14i24.1708

Khan, M. A., Nasralla, M. M., Umar, M. M., Iqbal, Z., Rehman, G. U., Sarfraz, M. S., Choudhury, N. (2021). A survey on the noncooperative environment in smart nodes-based Ad Hoc networks: Motivations and solutions. *Security and Communication Networks*, *2021*, 1-17.

Kumar, V., Malik, N., Dhiman, G., Lohani, T. K. (2021). Scalable and storage efficient dynamic key management scheme for wireless sensor network. *Wireless Communications and Mobile Computing*, *2021*, 1-11.

Mall D, Konaté K, Pathan ASK. (2013). On the key revocation schemes in wireless sensor networks. In IEEE International Conference on Green Computing and Communications and IEEE Internet of Things and IEEE Cyber, Physical and Social Computing, pp. 290-297. DOI 10.1109/GreenCom-iThings-CPSCom.2013.66.

Mansour I, Chalhoub G, Lafourcade P (2015) Key management in wireless sensor networks. Journal of sensor and actuator networks, 4(3), 251-273. doi:10.1109/inm.2005.1440863.

Mansour I, Chalhoub G, Lafourcade P, Delobel F. (2014) Secure key renewal and revocation for Wireless Sensor Networks. In 39th Annual IEEE Conference on Local Computer Networks pp. 382-385. IEEE. https://doi:10.1109/lcn.2014.6925797

Mehmood, G., Khan, M. S., Waheed, A., Zareei, M., Fayaz, M., Sadad, T., and Azmi, A. (2021). An efficient and secure session key management scheme in wireless sensor network. *Complexity*, *2021*, 1-10.

Moara-Nkwe, K., Shi, Q., Lee, G. M., Eiza, M. H. (2018). A novel physical layer secure key generation and refreshment scheme for wireless sensor networks. *IEEE Access*, *6*, 11374-11387.

Muruganandam, S., Joshi, R., Suresh, P., Balakrishna, N., Kishore, K. H., Manikanthan, S. V. (2023). A deep learning-based feed forward artificial neural network to predict the K-barriers for intrusion detection using a wireless sensor network. *Measurement: Sensors*, *25*, 100613.

Nabavi S.R.and Mousavi S.M. (2018). A review of distributed dynamic key management schemes in wireless sensor networks. J. Comput., 13(1), 77-89. https://doi:10.17706/jcp.13.1.77-89

Nafi, M., Bouzefrane, S., & Omar, M. (2020). Matrix-based key management scheme for IoT networks. *Ad Hoc Networks*, *97*, 102003.

Nithya, B. (2020). Cluster-based key management schemes in wireless sensor networks: a survey. *Procedia Computer Science*, *171*, 2684-2693.

Nour, B., Khelifi, H., Hussain, R., Mastorakis, S., Moungla, H. (2021). Access control mechanisms in named data networks: A comprehensive survey. *Acm computing Surveys (cSuR)*, *54*(3), 1-35.

Omar, M., Belalouache, I., Amrane, S., Abbache, B. (2018). Efficient and energy-aware key management framework for dynamic sensor networks. *Computers & Electrical Engineering*, *72*, 990-1005.

Rahman, M., and Sampalli, S. (2015). An efficient pairwise and group key management protocol for wireless sensor network. *Wireless Personal Communications*, *84*(3), 2035-2053. https://doi.org/10.1007/s11277-015-2546-4

Shamshad, S., Ayub, M. F., Mahmood, K., Kumari, S., Chaudhry, S. A., Chen, C. M. (2022). An enhanced scheme for mutual authentication for healthcare services. *Digital Communications and Networks*, *8*(2), 150-161.

Sheu, R. K., Pardeshi, M. S., & Chen, L. C. (2022). Autonomous Mutual Authentication Protocol in the Edge Networks. *Sensors*, *22*(19), 7632.

Wang Y, Ramamurthy B, Zou X (2007) KeyRev: An efficient key revocation scheme for wireless sensor networks. In IEEE International Conference on Communications (pp. 1260-1265). IEEE. DOI: 10.1109/ICC.2007.213.

Wang, J., Sun, C., Wang, H., Zhao, B., Gong, P. (2022). A CFL-Based Key Management Scheme for Routing-Driven Internet of Things. *Security and Communication Networks*, *2022*.

Wang, Y., Ramamurthy, B., Xue, Y. (2008). A key management protocol for wireless sensor networks with multiple base stations. In *2008 IEEE International Conference on Communications* (pp. 1625-1629). IEEE.

Wazid, M., Das, A. K., Kumar, N., & Vasilakos, A. V. (2019). Design of secure key management and user authentication scheme for fog computing services. *Future Generation Computer Systems*, *91*, 475-492.

Won, J., Seo, S. H., & Bertino, E. (2017). Certificateless cryptographic protocols for efficient drone-based smart city applications. *IEEE Access*, *5*, 3721-3749.

Xue, X., Shanmugam, R., Palanisamy, S., Khalaf, O. I., Selvaraj, D., Abdulsahib, G. M. (2023). A hybrid cross layer with harris-hawk-optimization-based efficient routing for wireless sensor networks. *Symmetry*, *15*(2), 438.

Yao W, Han S, Li X. (2015). LKH++ based group key management scheme for wireless sensor network. Wireless Personal Communications, 83(4), 3057-3073. https://doi:10.1007/s11277-015-2582-0

Zarezadeh, M., and Mala, H. (2019). Determining honesty of accuser nodes in key revocation procedure for MANETs. *Mobile Networks and Applications*, *24*, 903-912.

Zhang, J., Zhang, Q., Li, Z., Lu, X., Gan, Y. (2021). A lightweight and secure anonymous user authentication protocol for wireless body area networks. *Security and Communication Networks*, *2021*, 1-11.

Zhang, S., and Cao, D. (2023). A Blockchain based Provably Secure Anonymous Authentication for Edge Computing-enabled.

Zhang, W., Zhu, S., Cao, G. (2009). Predistribution and local collaboration-based group rekeying for wireless sensor networks. *Ad hoc networks*, *7*(6), 1229-1242.

Zhang, X., He, J., Wei, Q. (2011). EDDK: Energy-efficient distributed deterministic key management for wireless sensor networks. *EURASIP Journal on Wireless Communications and Networking*, *2011*, 1-11.

Zhang, Y., Shen, Y., & Lee, S. (2010). A cluster-based group key management scheme for wireless sensor networks. In *2010 12th International Asia-Pacific Web Conference* (pp. 386-388). IEEE.