A Secured Multi-Level Key Management Technique for Intensified Wireless Sensor Network

S.V. Manikanthan, T.Padmapriya

Abstract: ***Secure communication over the Wireless Sensor Network (WSN) is one of today's major concerns as the wireless communication medium suffers from a wide variety of security threats. Literature has suggested the multi-level key management protocol for establishing secure communication over WSN, where each transmission is established based on the availability of secured key. This work develops a key management protocol, namely Multi level Key Management (MLKM) protocol, for secure communication over the clustered WSN. The whole protocol is implemented in three stages, such as pre-deployment, key generation and key authentication and verification. In the first stage, the nodes are provided with the identity, and then the second stage uses the homomorphic encryption model to generate the necessary communication key. Finally, a mathematical model with multiple factors such as a hashing function, homomorphic encryption, dynamic passwords, profile sequence, random number, and EX-OR functions is developed in this work. The proposed MLKM protocol by authenticating the entities establishes the secured communication over the WSN. The entire work is compared and evaluated on the basis of several metrics with several states of art techniques. For communication overhead, detection accuracy, key memory storage and energy respectively, the proposed MLKM protocol achieved values of 0.122 kb, 0.929 kb, 2.332 kb and 14.586 joules.***

Keywords: ***Wireless Sensor Network (WSN), Secured Communication, Key Authentication, Key Management, Homomorphic Encryption.***

I. INTRODUCTION

Wireless Sensor Network (WSN) is one of the most widely used wireless communication platforms among vastly distributed sensors. In the wireless platform, WSN connects a large number of nodes as the nodes are distributed in thousands of kilometers from each other. The nodes ' storage capacity is generally low, and the computing power is also relatively limited. The sensor nodes in the network are displaced by one or more sink/gateway nodes to unattended locations[11]. In addition, the WSN has a platform without an ad hoc infrastructure. The sensor nodes in the WSN have their pressure, temperature, etc., and therefore each node will be adversely affected by the route established between them. Communication between the sources is established as the base station multi-hop communication.

Revised Manuscript Received on March 25, 2019
S.V. Manikanthan, Director, Melange Technologies, Puducherry
T.Padmapriya, Managing Director, Melange Technologies, Puducherry

Users prefer using the WSN for long - distance communication because it requires less power and easy maintenance [6]. The WSN uses multiple sink or gateway nodes to establish large - scale communication.

Due to its increased credibility, the WSN is preferred in domestic and surveillance systems, environmental monitoring, agriculture, healthcare, disaster management, military application, and sensor nodes, including analog to digital converters, microcontrollers, external memory, transceivers, and power sources[11]. Large WSN is computationally expensive to maintain and avoid this problem; the WSN prefers a clustering technique to group the sensor nodes together. WSN clustering allows the area to be separately divided and therefore makes maintenance much easier[18].

Communication based on clusters is now preferred. Cluster-based communication, however, adds extra overhead and burden in dense network scenarios to the Cluster Head (CH)[19], which eventually introduces delay and impedes network performance[20]. They are omnipresent and can be used for mission-critical applications like smart grid, smart purposes, health care, target monitoring, etc. Small, low-cost sensor nodes should be deployed on a large scale during these applications. The resource constraints of the WSN sensor nodes make it a challenging task to communicate between the sensor nodes, between the base station and sensor nodes and between all the sensor nodes. The messages within WSN must be encrypted[8][12] to ensure the confidentiality of the messages.

Several literature works have chosen to derive a secure transmission platform for WSN by enhancing node attack robustness. Secure communication platform makes the transmission resistant to node attacks. Node attacks are common in WSN, as an open platform is the architecture. Node attacks steal the key used to communicate primarily and steal / alter the message. One of WSN's main objectives is to establish secure communication in different adverse scenarios [16]. In recent years, communication between nodes in WSN is accomplished through Key Management Scheme (KMS), where a key for secure communication is established. By defining a set of mechanisms[4, 7], KMS establishes a secured communication service between the WSN nodes. While communicating with the KMS, a secret key for secure communication is established between the sensor nodes and the communication entities.

Also, for the secured communication, the secret keys used in the KMS must be constantly refreshed [29]. Furthermore, KMS[10] must ensure that the different keys used in the communication are securely generated, distributed and stored. This may fail in an adverse condition, so the expensive exponential

algorithms in key management only set the secured key instead of encrypting the messages[5].

Thus, in key management protocols, the encryption algorithm must be set separately. The encryption techniques used in key management protocols fall into three categories, being techniques of symmetry, asymmetry and hybridity[17]. As the WSN encryption scheme, dynamic group management protocol has been implemented in [21]. In [23 - 25], some of the key management services - related encryption protocols in WSN are discussed.

II. RELATED WORK

This section presents the survey of seven works of literature developed in WSN for key management. Qi Jiang et al.[1] used the Rabin cryptosystem module to present the lightweight authentication and key agreement protocols. As the actual security is established in this phase, the authentication phase in the key management requires more weighting. The Rabin cryptosystem module has been used since the model; the scheme has ensured safety against all possible attacks. The scheme provided the internet-integrated WSN with the lightweight authentication. In this work, the actual performance cannot be accurately measured and the cost of computing this model's gateway was found to be higher than the computational cost of the Das protocol and the protocol of Amin et al. Samir Athmani et al.[2] introduced the Dynamic Authentication and Key Management Mechanism for secured WSN communication.

The scheme considered the local information for the communication identification of the valid sensor nodes. This improved the WSN's energy efficiency and memory, but this method only solves the security issues introduced by the key distribution schemes and does not recognize other security issues, such as replay attack, privileged insider attack, etc. By developing flexible network arrangements for security-related features, R.Vijaya Saraswathi et al.[3] presented the dynamic and probabilistic key management protocol. For the secured transmission, it uses the pair-wise and group key management protocols. It also uses the access bloom filter. The Bloom filter's drawback is that without rebuilding the entire filter, one cannot remove existing items. It also suffered from overhead performance. It can't reserve growth space. By incorporating the SH3 algorithm, Pratusha Laxmi B and Chilambuchelvan A[5] proposed the WSN protocol for Geographic Secured Routing (GSR). While the protocol provides authentication of nodes and messages, the overhead computation was very low. The protocol provides improved performance and guaranteed packet delivery in the packed environment. This method suffered some disadvantages, such as high energy consumption, from various attacks on algorithms for sleep scheduling.

By exploiting security-related features present in the WSN, Priyanka and Mayank Dawe[7] presented the highly secure key management scheme. The scheme overcame problems during security enhancement, such as high node density, neighbor influence factor. The model did not develop an adaptive technique to mitigate the influence factor of the neighbor. Hash-based pre-distribution used in this method will result in some extra hash function storage overhead. Xinjiang Sun et al.[9] proposed the self-healing key management schemes by developing an enhanced collusion resistance module for broadcast authentication. The scheme has achieved increased security and therefore tolerates the effects of packet loss. The scheme has improved the rate of resource consumption. However, there is not much discussion here about security related issues related to WSN. In this work, the configurability of the self-healing capability, the security performance, and the adaptive size of the sliding window were not discovered under unreliable links. The group key management technique for the security improvement of the WSN module was presented by Purushothama B R and Arun Prakash Verma[8]. The scheme was specifically designed to mitigate the externally occurring attacks. Despite the performance of the scheme, it suffered from limited energy, memory, and power. Every key held by the sensors during each rekeying must be changed in this method, which adds computational burden on users and group controllers. There was also a need for secure channels, requiring additional communication costs.

Shraddha Deshmukh et al.[26] introduced the CL-EKM (Certificateless Effective Key Management) principle for secure WSN communication portrayed by hub versatility. In memory execution, correspondence, vitality, and time, this method was valuable. The disadvantage of this method is that the identity information is no longer the entire public key, which means that the public key of the user can not be discovered from the identity string of the user and the public key of the Key Generation Center (KGC). Jaewoo Choi et al.[27] developed a location-based key management scheme for WSNs, focusing on insider threats. He introduced a key revision process to address the problem of communication interference in location-dependent key management (LDK), which included grid-based location information. If a packet drop attack has been affected, each packet sent through the SNs sensor nodes has been dropped and the network loses its function.

The challenges involved in the secured WSN data communication are shown below:

- One of the most important challenges in WSNs is to secure network communication. Most research is based on key management that suffers from issues such as probabilistic key distribution between high-speed (HSN) and low-speed (LSN) networks, post-deployment non-scalability, node compromise mounting weakness, lack of resource-limited LSN memory storage and high-communication overhead[2].
- Source authentication broadcasting is a challenging WSN topic. This security service enables senders to transmit messages in a secure manner to multiple receivers[15]. Consequently, ensuring secure access to sensitive information in a WSN remains an ongoing research challenge,

partly due to the wide range of potential attacks and vectors of attack[13].

- Limitations such as processing, memory constraints, limited network capacity, limited reserves of energy, etc., make security

enhancement vulnerable in WSN[14].

- Key solutions for pre-distribution can never be applied to the WSN[3].
- Key transfers that can be challenging in the dynamic network prior to communication[3].

III. A CLUSTERED MODEL OF WSN NETWORK

This section outlines the proposed methodology for designing the WSN key management system and its architecture is shown in Figure 1.

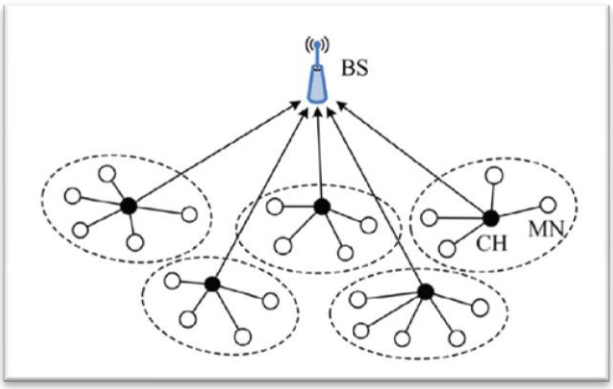

Fig. 1 A clustered model of WSN network

After regulating the network with the CHs and routing protocol, the clustered WSN initiates data transfer. The clustered WSN uses the cluster formation protocol Low Energy Adaptive Clustering Hierarchy (LEACH). One of the commonly used protocols for selecting the CH among the node group is the LEACH protocol. By considering the objective as 'the CH should consume minimal energy' the LEACH forms the cluster. This criterion increases energy efficiency in the clustering process. At a simulation time t, the LEACH protocol performs cluster formation. The CH's energy will be reduced after a certain length of time. Thus, at the given time interval t, the LEACH protocol will be simulated continuously to select the CH constantly.

LEACH protocol dynamic execution in WSN mainly allows interruption and error-free communication service. The communication between the nodes and Base Station (BS) is carried through the selected CH after selecting the CH. In order to establish a secure communication link between the WSN, a communication link must be established through a routing algorithm. The WSN nodes pass through 'hopping' the information from one node to another. Thus, with the multi-hop routing algorithm, the routing algorithm for communication in clustered WSN is done. The multi-hop routing algorithm identifies the secured routing path between source and destination and transfers the data packets either through the estimated routing path.

There are a total of P nodes in the clustered WSN, separated by distance. The nodes in the clustered WSN are represented as{ n1,n2,K,.ni, K, np} and the nodes are assigned to be CH after the clustering process, represented as{ c1,c2,K,.ck, K, cH} The nodes transfer the information from one node to another and finally, via the chosen CH, to the BS. By encrypting the message with the secret key, the nodes send the message from sender to receiver and thus preserve the message authentication. There are three subfields in the data packets sent from one node to another, such as header, data and code. Figure 2 shows the general structure of the message between the nodes being transferred.

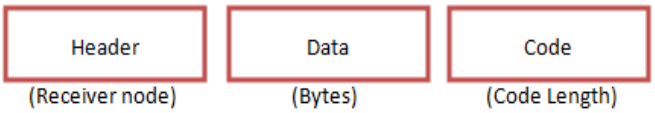

Fig. 2 General Format of the Message

IV. KEY MANAGEMENT WITH THE PROPOSED MLKM PROTOCOL FOR CLUSTERED WSN

This section describes the proposed MLKM protocol to establish a clustered WSN secure communication link. In three different phases, the proposed MLKM protocol performs key management. The MLKM protocol architecture is shown in Figure 3.

As shown in the figure above, the three phases of the MLKM are 1) Pre-Deployment, 2) Key Generation, and 3) Key Authentication and Verification phase. The proposed MLKM protocol can be seen as an improvement to the Goldwasser–Micali (GM) tool, so that the proposed MLKM protocol uses the public key to secure transmission.

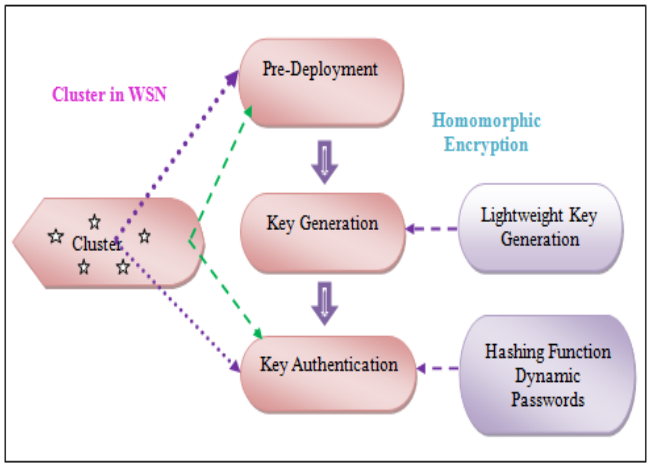

Fig. 3 MLKM protocol in WSN - based dynamic cluster

A. Pre-Deployment Phase

In the proposed MLKM protocol, the initial step is the pre-deployment phase where a specific identity is given to the nodes. In general, the pre-deployment phase assigns for each node, BS, and CH the possible identification. Since CH is one of the primary needs in WSN communication, pre-deployment takes place after WSN node clustering.

The pre-deployment phase loads to each node, CH and BS in the WSN the predefined network key A_{key}. For each transmission initiation, the WSN has its network parameter. For secure transmission, most of the WSN uses the 128-bit network key. To generate the network key, this paper adopts the homomorphic encryption[22].

B. Key Generation phase

The next major phase is to generate key for an individual node, CH and BS after providing the identity for the nodes. The key generation phase helps to generate between the nodes private and public keys for transmission.

This paper adopts the GM Encryption Scheme [22] derived to generate the nodes ' private and public keys. The steps taken by the key generation GM scheme will be explained as follows:

Consider the WSN j^{th} node that initiates the transmission a nd requires both the private key and the public key for the p urpose of the transmission. The source node initially generat es two distinct large numbers of prime numbers u and v.

The product Q = u.v will then be calculated between the random numbers u and v. The nonresidue factor is calculated as

$$r_u^{(u-1)/2} = -1(\mathrm{mod}\, u) r_v^{(v-1)/2} = -1(\mathrm{mod}\, v)$$

Public Key Generation: The residual factor r and the product Q are used to generate the public key. The public key therefore consists of (r, Q).

Private Key generation: The private key is built based on **(u, v).**

The above steps are used to generate the nodes, CH and BS private and public keys. The node, CH and BS generated key is given as follows:

Node, $[N^R_{key}, N^U_{key}] = GM(u_1, v_1)$ (1)

CH, $[H^R_{key}, H^U_{key}] = GM(u_2, v_2)$ (2)

BS, $[B^R_{key}, B^U_{key}] = GM(u_3, v_3)$ (3)

Where, (u1,v1), (u2,v2) and (u3,v3) are set of declared large, distinct prime numbers for the node, respectively CH and BS.

C. Key Authentication and Verification phase

MLKM protocol's final phase is key authentication and verification. Figure 4 shows the communication flow in the key authentication and verification phase between the source and the destination node. It is necessary to establish a secure communication link between the nodes before sending a secured message across the communication platform. In the key authentication phase, the secured communication link is established between the sender and the receiver before the message is transmitted. The MLKM protocol generates the session key for each data transfer to get the cipher text.

Figure 4 shows the authentication and verification phase of the operation flow. With the generation of the session key, the data transfer is initiated and the transfer will take place once the secured communication link is established. During the key authentication and verification phase, the entire data flow is briefed as follows:

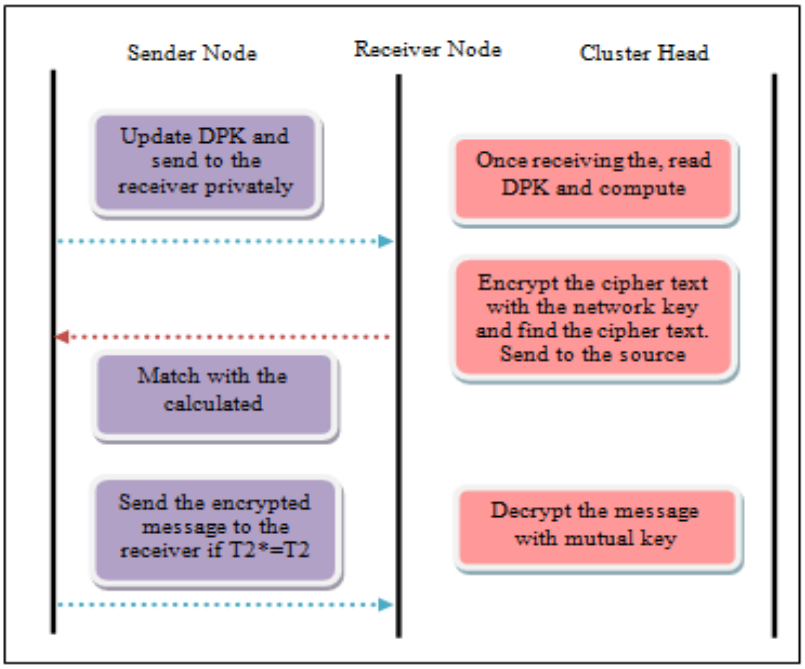

Fig. 4 Key Authentication and Verification phase

Initiate the data transfer

The source node initiates the message transmission and the message has its own identity, network key, and code. There are several changes to the message corresponding to the nodes to cope with the security credentials. Consider the neither source node n_i initiates the transfer of data to the n_j destination node. Each node in the network holds the key to the dynamic profile that holds the routing information. The key used in the node for the dynamic profile is represented as DPK_{ij}. Here, i and j refer respectively to the source and destination node index. The dynamic profile key contains the encrypted message identity information, receiver identity and the mutual key between n_i and n_j nodes. The dynamic profile key's general format is given as follows,

$$DPK_{ij} = \begin{bmatrix} E(Zid) & E(Rid) & E\left(W^{ij}_{key}\right) \\ M & M & M \end{bmatrix} \begin{matrix} Data_1 \\ Data_2 \\ M \\ Data_x \end{matrix} \quad (4)$$

Where the message identity is indicated by Z_{id}, R_{id} indicates the recipient node identity and W^{ij}_{Key} refers to the mutual key generated between the source and the recipient. The E() function indicates the encryption performed using the existing GM algorithm using the S^{+}_{Key} session key.

One of the important elements of the proposed MLKM protocol is the session key.

The session key is generated temporarily, which lives during the transfer of data between the node i and j for the communication session. The following terms indicate the GM scheme encryption.

$$E[Z_{id}] = GM(Z_{id}, S^{+}_{key}) \quad (5)$$

$$E[R_{id}] = GM(R_{id}, S^{+}_{key}) \quad (6)$$

$$E[W^{ij}_{key}] = GM(W^{ij}_{key}, S^{+}_{key}) \quad (7)$$

Update DPK and send session key to the receiver privately

The source updates the information in the next step and sends the session key privately to the recipient. Until the end of the communication the session key is retained.

Compute the ciphertext T1 at the receiver node

The receiver understands after obtaining the session key from the sender that the source attempts to communicate through the secured communication link. The MMKM protocol only performs data transfer when the secured data link is established. The recipient checks the validity of the source by generating ciphertext for this type of communication. The receiver initially generates the T1 ciphertext. Based on intermediate message I, the ciphertext T1 is obtained. The intermediate message I depends on the session key decrypted information and depends on the equation below.

$$I = \begin{bmatrix} D\left(E(Z_{id})|S^{+}_{key}(received)\right)||D \\ \left(E(R_{id})|S^{+}_{key}(received)||D\left(E(Z_{id})|W^{ij}_{key}(received)\right)\right) \end{bmatrix} \quad (8)$$

Where, S^{+}_{Key} (received) refers to the receiver's session key, and D) (refers to the decryption performed on message entities based on the session key received. After identifying the intermediate message I, using the hash function, the ciphertext T_i is identified. Ciphertext generation T_1 can be defined as follows,

$$T_1 = h(I) \quad (9)$$

Where the hash function is indicated by h(). The ciphertext T_1 is formed after the hashing operation is performed over the intermediate message. Ciphertext generation T_1 can be referred to as the safety layer 1.

Generate T_2 and send to the source node

After establishing the first layer of security, by generating the ciphertext 2, referred to as T_2, the receiver creates the next layer of security. Based on the ciphertext T_1, the ciphertext T_2 is created. The next security layer is awarded by the network key, thus generating the ciphertext T_2 as follows.

$$T_2 = \left[D\left(E(T_1)|A_{key}\right)||H^{u}_{key}\right] \quad (10)$$

Where, the network key is represented by Akey. The receiver will reply to the sender request by sending the ciphertext T_2 after generating T_2. It is obvious that the MLKM protocol recognizes the message through multilevel security level, thereby reducing the likelihood of theft of security.

Compute T_2* at the sender side

Once the receiver receives the ciphertext T_2, the sender attempts to decode the information. Initially, the sender creates the I* intermediate message with message id, receiver id, and mutual key, and is presented as follows.

$$I^{*} = \left[Z_{id}||R_{id}||W^{ij}_{key}\right] \quad (11)$$

Where, I* refers to the sender-side intermediate information. The sender reconstructs from this information the ciphertext 1, indicated as T_1*, and it is indicated as,

$$T^{*}_{1} = h(I^{*}) \quad (12)$$

Where, T^{*}_{1} shows the ciphertext 1 on the side of the transmitter. Using the hash function h), (reconstruction of ciphertext 1 is made possible. The sender finds the text2 cipher, i.e. T_2*, by using the header node's public key. The wording for the T_2* is referred to as follows:

$$T^{*}_{2} = E\left(T^{*}_{1}|A_{key}||H^{u}_{key}\right) \quad (13)$$

Once the T_2* ciphertext is calculated on the sender side, this information can be used to validate the receiver node's authenticity.

Match T_2 and T^{*}_{2} at the sender side

The actual message transmission occurs in this step. The sender validates the receiver's authenticity by matching the calculated $\mathbf{T^{*}_{2}}$ ciphertext to the receiver T_2 ciphertext. Once both the ciphertext matches, i.e. $\mathbf{T^{*}_{2}}$ = T2, the sender declares the receiver to be valid and initiates the original data transfer. The communication will be terminated if the cipher does not match, and the session key generated for the communication will expire.

Send the encrypted information to the receiver

The source sends the original message to the receiver after matching the ciphertext by encrypting the message with the mutual key. The message sent to the receiver by encryption is shown below:

$$E_{message} = E\left(Z|W^{ij}_{key}\right) \quad (14)$$

Decrypt the message at the receiver side

The message received by the receiver is represented as $E_{message}$(received) after receiving the encrypted message. Upon receiving the message, only through decryption can the information be used by a receiver. The decryption in the receiver node uses the mutual key and can therefore be represented as

$$Z(received) = D\left(E_{message}(received)\right)|W^{ij}_{key} \quad (15)$$

It is ensured in the above mentioned MLKM protocol that the MLKM protocol performs multilevel security scheme to ensure privacy between nodes. Either private or public key is used in existing works to ensure security, but the proposed MLKM protocol uses private as well as public keys to ensure security.

The key authentication and verification phase operation flow is shown in algorithm 1.

Algorithm. 1 Pseudo code of the proposed MLKM protocol

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

MLKM protocol
Parameters: nodes, CH, BS
//Pre-deployment phase
For each node, CH, BS
Find the network key Akey using GM algorithm
Assign the network key Akey to nodes, cluster head, BS
End for
//Key generation phase
For every node
Generate $[N_{Key}^{R}, N_{Key}^{U}]$ using GM algorithm
End for
For every CH
Generate $[H_{Key}^{R}, H_{Key}^{U}]$ using GM algorithm
End for
For BS
Generate $[B_{Key}^{R}, B_{Key}^{U}]$ using GM algorithm
End for
For every data sample
For j^{th} node in i^{th} cluster
Define the session key S_{Key}^{+}
Perform encryption over the message identity, receiver identity, and mutual key
Assign $E[M_{id}]$=GM(M_{id}, S_{Key}^{+})
Assign $E[R_{id}]$=GM(R_{id}, S_{Key}^{+})
Assign $E[M_{Key}^{ij}]$=GM(M_{Key}^{ij}, S_{Key}^{+})
//Key Authentication and Verification phase
Initiate the transmission
Source updates DPK and sends S_{Key}^{+} to the receiver
The receiver computes the code message I using equation (8)
Compute the ciphertext T_1 using equation (9)
Compute the ciphertext T_2 using equation (10)
Source finds the code message I* using (11)
Source finds T_1* and T_2* using (12) and (13) respectively
If ($\mathbf{T_2^*} * = T2$)
Initiate data transfer
Else
Declare the j^{th} node as the attacker
End for

V. RESULTS AND DISCUSSION

The simulation results of the proposed MLKM protocol are presented in this section. The simulation results of the proposed MLKM protocol are compared with several state of the art techniques defined respectively in [1], [2] and [3]. Standard evaluation metrics are used to evaluate the performance of state-of - the-art techniques.

A. Experimental Setup

The entire MLKM protocol work is implemented in the NS2 simulator, and the PC with Ubuntu OS, 4 GB RAM, and Intel I3 processor is used in addition.

B. Simulation Setup

MLKM protocol simulation platform can be done by initializing the following parameters as shown in table 1.

Table. 1 Simulation Parameters

Number of nodes	150
Area Size	1000X1000
MAC	IEEE 802.11
Transmission Range	250 meters
Simulation Time	50 seconds
Traffic Source	CBR
Packet Size	512
Sources	2,4,6,8 and 10
Rate	100,200,300,400 and 500 Kb
Initial Energy	9.1 Joules
Transmission Power	0.660 Watts
Receiving Power	0.395 Watts

Here, two different scenarios assess the MLKM protocol, 1) With attack, and 2) Without attack. Simulation is done for the first type by introducing the selective packet drop attack and attacking Black Hole. Each node is free of all kinds of network attacks in the second type of simulation platform.

C. Performance Metrics

Specifically designed MLKM platform used for secure transmission in WSN. The effectiveness of the proposed MLKM protocol is measured by three metrics, such as key memory storage, overhead communication and detection accuracy, and these metrics are given as follows,

1) Key memory storage
2) Communication overhead
3) Detection accuracy

D. Comparative Techniques

Comparison of the performance of the proposed MLKM protocol with other state of the art techniques such as Efficient Dynamic Authentication and Key Management (EDAK)[2], three-factor[1], and probabilistic key[3]. The comparative techniques description is given as follows:

- EDAK
- Three-factor authentication
- Probabilistic key

E. Comparative Discussion

Table 2 presents the comparative discussion of the techniques performance against the MLKM protocol proposed.

Table 2. Comparative Discussion for without Attack

Evaluation Metrics	Comparative Techniques			
	Probabilistic Key	Three factor authentication	EDAK	Proposed MLKM
Communication overhead	0.174	0.287	0.305	0.122
Detection accuracy	0.915	0.880	0.875	0.929
Key memory storage	3.384	4.301	3.722	2.332
Energy	13.925	13.469	13.389	14.586

The comparative model performance against the proposed MLKM protocol is discussed in Table 2. The discussion suggests that the current probabilistic key authentication scheme has

values of 0.174 kb, 0.915, 3.384 kb and 13.925 joules respectively as overhead communication, detection accuracy, key memory storage and energy. The proposed MLKM protocol achieves the overall improved performance with the values of 0.122, 0.929, 2.332 and 14.586 as the overhead communication, accuracy detection and key memory storage.

Table. 3 Comparative discussion techniques with attacks

Evaluation Metrics	Comparative Techniques			
	Probabilistic Key	Three factor authentication	EDAK	Proposed MLKM
Communication overhead	0.292	0.406	0.396	0.242
Detection accuracy	0.724	0.656	0.673	0.749
Key memory storage	3.698	4.740	4.580	3.059
Energy	12.950	12.769	12.648	13.137

The performance of the comparative model against the proposed MLKM protocol is discussed in Table 3 while the system is under attack. The discussion suggests that the current three-factor authentication scheme has values of 0.406, 0.656, 4.740 and 12.769 as overhead communication, accuracy detection, key memory storage, and energy. The overall improved performance is achieved through the proposed MLKM protocol, with values of 0.242 kb, 0.749, 3.059 kb and 13.137 joules respectively as overhead communication, detection accuracy, key memory storage and energy.

Table 4. Key computation time of the comparative techniques

Methods	Key computation time (Sec)
Probabilistic Key	8
Three factor authentication	8.5
EDAK	7
Proposed MLKM	5

The key calculation time of the proposed method and the existing methods are shown in Table 4. The existing methods, probabilistic key, three-factor, and EDAK have the key computing time of 8 sec, 8.5 sec, and 7 sec respectively, while the proposed method has the minimum key computing time of 5 sec. From the analysis, it can be concluded that the proposed method provides the best performance in terms of overhead communication, detection accuracy, key memory storage and energy. The reason for this is that the proposed method develops the multilevel data communication security link to ensure security. Also, the proposed method uses homomorphic encryption, which has several advantages, such as solving confidentiality issues, ensuring privacy, etc.

VI. CONCLUSION

This work develops a key management protocol for the secure transmission of data over the WSN, namely MLKM. The proposed MLKM protocol establishes a secure communication connection between the nodes and sends the encrypted information through the secured link. By developing the multilevel security link, the proposed MLKM protocol ensures security over data communication. Specifically designed for the clustered WSN, the MLKM protocol has three stages, namely pre-deployment, key generation, and key authentication and verification. The pre-deployment phase carries out the identification task by providing WSN nodes with the identity and key. Homomorphic encryption is done in the next stage to find the key to encryption. The mathematical model with the secured factors such as hashing function, homomorphic encryption, dynamic passwords, profile sequence, random number and EX-OR functions for secure data transmission is developed in the final stage. The entire work is compared with several art techniques and evaluated based on measurements such as memory, key storage, size, communication overhead and use of bandwidth. Results of simulation reveal that the proposed MLKM protocol has achieved improved performance with values of 0.122 kb, 0.929, 2.332 kb and 14.586 joules respectively for communication overhead, detection accuracy, key memory storage and energy.

REFERENCES

1. Q. Jiang, S, Zeadally, J. Ma and D. He, "Lightweight three-factor authentication and key agreement protocol for internet-integrated wireless sensor networks," in IEEE Access, vol. 5, pp. 3376-3392, 2017.
2. Athmani, Samir, Azeddine Bilami, and Djallel Eddine Boubiche, "EDAK: An Efficient Dynamic Authentication and Key Management Mechanism for heterogeneous WSNs," in Future Generation Computer Systems, November 2017
3. Saraswathi, R. Vijaya, L. Padma Sree, and K. Anuradha, "Dynamic and probabilistic key management for distributed wireless sensor networks," proceedings of IEEE International Conference on Computational Intelligence and Computing Research, pp. 1-6, 2016.
4. Azarderakhsh, Reza, Arash Reyhani-Masoleh, and Zine-Eddine Abid, "A key management scheme for cluster based wireless sensor networks," Proceedings of IEEE/IFIP International Conference on Embedded and Ubiquitous Computing, vol. 2, pp. 222-227, 2008.
5. Laxmi, B. Prathusha, and A. Chilambuchelvan. "GSR: Geographic Secured Routing using SHA-3 algorithm for node and message authentication in wireless sensor networks." in Future Generation Computer Systems, vol.76, pp 98-105, 2017
6. Ramachandran, Shyamala, and Valli Shanmugam, "A two way authentication using bilinear mapping function for wireless sensor networks," in Computers & Electrical Engineering, vol.59, pp.242-249, 2017.
7. Ahlawat, Priyanka, and Mayank Dave, "An attack model based highly secure key management scheme for wireless sensor networks," Procedia Computer Science, vol.125, pp. 201-207, 2018.
8. B. R. Purushothama and A. P. Verma, "Security analysis of group key management schemes of wireless sensor network under active outsider adversary model," Proceedings of International Conference on Advances in Computing, Communications and Informatics, pp. 988-994, 2017.
9. Sun, Xinjiang, Xiaobei Wu, Cheng Huang, Zhiliang Xu, and Jianlin Zhong, "Modified access polynomial based self-healing key management schemes with broadcast authentication and enhanced collusion resistance in wireless sensor networks," Ad Hoc Networks, vol.37 , pp.324-336, 2016.

10. R Vijaya Saraswathi, L Padma Sree and K. Anuradha, "Key Management Schemes in Wireless Sensor Networks: A Survey" CiiT International Journal of Wireless Communication, Vol 8, No 05, May 2016.
11. Srinivas, Jangirala, Sourav Mukhopadhyay, and Dheerendra Mishra, "Secure and efficient user authentication scheme for multi-gateway wireless sensor networks," Ad Hoc Networks, vol. 54, pp. 147-169, 2017.
12. Razaque, Abdul, and Syed S. Rizvi, "Secure data aggregation using access control and authentication for wireless sensor networks," Computers & Security, vol.70, pp. 532-545 2017.
13. Amin, Ruhul, SK Hafizul Islam, Neeraj Kumar, and Kim-Kwang Raymond Choo, "An untraceable and anonymous password authentication protocol for heterogeneous wireless sensor networks," in Journal of Network and Computer Applications, vol. 104, pp. 133-144, 2017.
14. Challa, Sravani, Ashok Kumar Das, Vanga Odelu, Neeraj Kumar, Saru Kumari, Muhammad Khurram Khan, and Athanasios V. Vasilakos, "An efficient ECC-based provably secure three-factor user authentication and key agreement protocol for wireless healthcare sensor networks," Computers & Electrical Engineering, 2017.
15. Chang, Shang-Ming, Shiuhpyng Shieh, Warren W. Lin, and Chih-Ming Hsieh, "An efficient broadcast authentication scheme in wireless sensor networks," in ACM Symposium on Information, computer and communications security, pp.311-320, 2006.
16. He, X., Neidermeier, M., Meer, H, "Dynamic key management in wireless sensor network: a survey" in Journal of Network and Computer Applications, vol 36, pp. 612-622 2013.
17. Zhang, J., Varadharajan, V ,"Wireless sensor network key management survey and taxonomy" Journal of Network and Computer Applications, vol.33, no.2, pp.63–75, 2010.
18. Zahedi, Abdulhamid, "An efficient clustering method using weighting coefficients in homogeneous wireless sensor networks," in Alexandria Engineering Journal 2017
19. Ouchitachen, Hicham, Abdellatif Hair, and Najlae Idrissi, "Improved multi-objective weighted clustering algorithm in Wireless Sensor Network." Egyptian Informatics Journal, vol.18, no. 1, pp 45-54, 2017.
20. Sharma, Sparsh, and Ajay Kaul, "Hybrid fuzzy multi-criteria decision making based multi cluster head dolphin swarm optimized IDS for VANET., Vehicular Communications, vol. 12, pp. 23-38, 2018.
21. R Vijaya Saraswathi , L Padma Sree, and K Anuradha, "DynGKM: Dynamic Group Key Management Scheme for Cluster Based Wireless Sensor Networks," International Journal of Network Security, 2019.
22. Xun Yi, Russell Paulet, Elisa Bertino, "Homomorphic Encryption and Applications," Springer Briefs in Computer Science, 2014
23. Wu, Q., Mu, Y., Susilo, W., Qin, B. and Domingo-Ferrer, J., "Asymmetric group key agreement," In Annual International Conference on the Theory and Applications of Cryptographic Techniques, Springer, Berlin, Heidelberg, pp. 153-170, 2009.
24. Zhang, L., Wu, Q., Qin, B. and Domingo-Ferrer, J., "Identity-based authenticated asymmetric group key agreement protocol," In International Computing and Combinatorics Conference, Springer, Berlin, Heidelberg, pp. 510-519, 2010.
25. Zheng, X., Huang, C.T. and Matthews, M., "Chinese remainder theorem based group key management," In Proceedings of the 45th annual southeast regional conference, pp. 266-271, 2007.
26. Shraddha Deshmukh, Prof. A. R. Bhagat Patil, and Harshad Nakade, "Implementation of Effective Key Management Strategy with Secure Data Aggregation in Dynamic Wireless Sensor Network," International Journal of Scientific Research in Computer Science, Engineering and Information Technology, vol.4, no. 2, pp. 358-364, 2018.
27. Jaewoo Choi, Jihyun Bang, LeeHyung Kim, Mirim Ahn, and Taekyoung Kwon,"Location-Based Key Management Strong Against Insider Threats in Wireless Sensor Networks," IEEE Systems Journal, vol. 11, no. 2, pp. 494 - 502, June 2017.
28. Houda Moudni , Mohamed Er-rouidi , Hicham Mouncifand Benachir El Hadadi, "Fuzzy Logic based Intrusion Detection System against Black Hole Attack in Mobile Ad Hoc Networks" International Journal of Communication Networks and Information Security (IJCNIS) Vol. 10, No. 2, August 2018.
29. Reegan, A. Selva, and E. Baburaj, "Key management schemes in wireless sensor networks: a survey," Proceedings of International Conference on Circuits, Power and Computing Technologies, pp. 813-820, 2013.