Computers, Materials & Continua
DOI:10.32604/cmc.2022.021641
Article

Tech Science Press

Lightweight Key Management Scheme Using Fuzzy Extractor for Wireless Mobile Sensor Network

Eid Rehman[1], Ibrahima Kalil Toure[2], Kashif Sultan[3], Muhammad Asif[4], Muhammad Habib[1], Najam Ul Hasan[5], Oh-Young Song[6,*] and Aaqif Afzaal Abbasi[1]

[1]Department of Software Engineering, Foundation University, Islamabad, 44000, Pakistan
[2]Centre Informatique, Universite Gamal Abdel Nasser de, Conakry, Guinea
[3]Department of Software Engineering, Bahria University Islamabad Campus, Islamabad, Pakistan
[4]GDC Ahmad Karak, Higher Education Department, Khyber Pakhtunkhwa, Pakistan
[5]Department of Electrical and Computer Engineering, Dhofar University, Salalah, Oman
[6]Department of Software, Sejong University, Seoul, 05006, Korea
*Corresponding Author: Oh-Young Song. Email: oysong@sejong.edu
Received: 09 July 2021; Accepted: 20 August 2021

Abstract: The mature design of wireless mobile sensor network makes it to be used in vast verities of applications including from home used to the security surveillance. All such types of applications based on wireless mobile sensor network are generally using real time data, most of them are interested in real time communication directly from cluster head of cluster instead of a base station in cluster network. This would be possible if an external user allows to directly access real time data from the cluster head in cluster wireless mobile sensor network instead of accessing data from base station. But this leads to a serious security breach and degrades the performance of any security protocol available in this domain. Most existing schemes for authentication and cluster key management for external users, exchange a number of messages between cluster head and base station to allow external to access real time data from the base station instead of cluster head. This increase communication cost and delay in such real time access information. To handle this critical issue in cluster wireless mobile sensor network, we propose a lightweight authentication and key management scheme using a fuzzy extractor. In this scheme, any external user can access data directly from the cluster head of any cluster without the involvement of the base station. The proposed scheme only uses the one-way hash functions and bitwise XOR operations, apart from the fuzzy extractor method for the user biometric verification at the login phase. The presented scheme supports scalability for an increasing number of nodes using polynomials. The proposed scheme increases the life-time of the network by decreasing the key pool size.

Keywords: Fuzzy extractor; user authentication; key management; cluster session key; wireless mobile sensor network

1 Introduction

The most recent innovative advances in Multi-Electro-Mechanical Systems (MEMS) have empowered the improvement of scaled-down sensor nodes [1]. These nodes are small in size having constrained communication and processing abilities with non-replaceable small power batteries. Besides, these nodes constrained storage capacity and commonly have transmission radio range.

Wireless sensor networks whose all or some sensors have the capability of movement around thef deployed area are called Wireless Mobile Sensor Network (WMSN) [2]. WMSN consists of a huge number of dispersed tiny mobile nodes possibly installed in a remote hostile environment. These nodes have limited power in terms of processing, memory storage, and most importantly the energy resources. After random deployment, sensor nodes are divided into different groups called clusters. Each group has selected one node as a representative node called cluster head (CH). The remaining nodes of a group are called member nodes and transmit their data toward the cluster head.

All the data are summarized on the cluster head and forward to BS [3]. Any external user can access data from a base station. A general structure of cluster base WMSN with the external user is shown in Fig. 1. Each cluster head has to manage a cluster of nodes and forward the send information of their member to the base station.

The given structure shown in Fig. 1, assumes an external user who went access the real-time information from the cluster head directly instead of a base station. Such access is provided both the external user and cluster head mutually authenticate each other with the help of the base station. The base station is mostly considered a trustworthy entity and will not be compromised by an attacker. Because of the wireless communication nature, different security issues occur in WMSN. Ordinary security frameworks utilizing open key cryptography make overhead with deference computational and transmission costs. Therefore, secure lightweight authentication and key management scheme need to be designed for WMSN, which should be efficient communicationally and computationally for an external user, sensor nodes, and base station [4].

Different nameless security schemes have been proposed for WMSN in the later past which shows differing natures and additional levels of security assurance at different prices. In this section, we talk about the current cryptography schemes that tended to the issue of key management for secure communication in WMSNs. Because of the resource's limitation in WMSNs, a comprehensive harmony between energy usage overhead and the security level is expected to moderate the safety threats. Some symmetric metrics, for example, Node-ID, message authentication code (MCA), nonce-number, and time-stamps. These are energy efficient parameters for cluster key management techniques. Additionally, this keeps away from the distinctive kind of attacks from a suspected node and stays away from compromised node attacks. Various security schemes presented for WMSNs utilize symmetric encryption, because of the simplicity of its execution [5]. Other than this, single node authentication has turned out to be not able to take care of the increasing transmission demand. When the services request is growing up day by day, a multiparty calculation is fundamental for nodes verification (authentication) concurrently and safely. Similarly, for inter and intracluster communication, member nodes jointly build a mutual session key called cluster key to allow secure exchange of messages [6].

The majority of the applications engaged with the WMSN are real-time based [7]. Along these lines, the external users are for the most part keen on getting to the real-time data from the base station instead of the cluster head of a specific cluster which may create a delay for such time-sensitive data. This occurs on the off chance that we permit the external user to straightforwardly access real-time information from the cluster head and not from the base station. As a rule, the data that is accumulated by the base station from the automatons cluster head occasionally, and accordingly, the gathered data

may not be in real time. Accordingly, to acquire the real time data from the cluster head, the user (for instance, driver of a rescue vehicle) needs to get to the information legitimately from a got to ramble given that the user is an authentic one to get to it from that cluster head is ordered to restrict un-authorized access information from the cluster head. On the premise of the received information from the specific cluster head, the external user can take a significant choice, for example, the driver of the rescue vehicle can pick the correct run time path which has fewer blockages to help that driver to spare the life of a patient. This requires designing an efficient external user authentication and key management scheme for providing real-time access in WMSN.

The rest of the paper is organized as follows. Section 2 provides the literature review of some well-known cluster key management algorithms for WMSN. Section 3 describes the network and threat model. Section 3.2 describes the System model. Section 4 presents some preliminaries and Section 5 presents a proposed scheme. Section 6 describes the security analysis of the proposed scheme with other schemes and Section 7 discusses the simulation performance of the proposed scheme.

2 Literature Review

Wireless Sensor Networks (WSNs) have a few applications including traffic observing, avalanche location, pipeline observing, fringe watch, restoration applications, accuracy horticulture, research facility mentoring, constant human services checking furthermore, military applications [8]. For all such types of applications, real-time information access is required by an approved user (external) from a specific node directly. Along these lines, user authentication is required for making secure communication.

By and large, security in WSN has been broadly researched in recent times. A large portion of the security arrangements has been composed either to protect WSNs from some known attacks (*e.g.*, particular sending, dark gap) or cautious procedures: for example, intrusion detection system [9] is proposed.

Prevention mechanism such as key management scheme is presented [10]. The key messages transmitted through an intermediate node ought to likewise be secure [11]. In any case, they are designed for static WSN [12] which requires a vast number of messages to set up and maintain an update key over the system. Also, the dynamic nature of WMSN (frequent mobility) requires keys to be refreshed when needed. This causes immense communication overhead on nodes with less energy and henceforth decreases their lifetime.

To build up secure keys among the member node of a cluster, the scheme in [13] proposed a Logical Key Hierarchy (LKH) where the whole cluster is represented like a tree. Leaf nodes *i.e.*, member nodes share symmetric keys. The cluster key is allocated to CH. Jen-Chiun Lin et al. presented One-way Key derivation (OKD) [14] that used the idea of one-way hash function like Dini et al. LKH scheme was additionally enhanced by Je et al. to consider the resources of each node during tree development. In these schemes, indirect path keys between leaf sensor nodes over the cluster are set up using the CH node of a neighboring cluster. Similarly, another tree-based cluster key management is presented [15] in which a leaf node can calculate keys toward CH.

One key exchange scheme is Localized Encryption and Authentication Protocol (LEAP) [16], which was proposed to secure the inter-cluster communication of WSNs. LEAP organized communication messages and presented four sorts of keys inside a network for security. Every one of the four keys was shared between individual nodes of the WSN. This scheme is very costly because large keys are used.

The paper [17] utilized two polynomial pools, common mobile and common static, on which they executed three-level architecture to pick up an improved level of security for WSNs. The pools have a sensor node with getting to focus and movable sinks. Keys are conveyed by the access point and the portable sinks. Pairwise key pre-distribution techniques are utilized for the authentication of a node with the assistance of polynomial keys.

One of the key management schemes [18] is presented for heterogeneous WMSNs using asymmetric key pre-distribution and hash function. It utilizes a seed key and hash capacity to understand the authentication of a mobile CH, however, it just allows CH mobility, and the entire members are static.

A key pre-distribution algorithm where BS provides seeds to sensor nodes to compute another key that gives satisfactory security was depicted in [19]. It permits secretly appropriating a secret to an arrangement of beneficiaries with just a single multicast correspondence [20]. A less expensive XOR-based re-keying scheme is presented in [21] which does not need message exchange in WSN for key distribution.

A dynamic polynomial-based key management scheme is presented in [22] where the master node is used for secure communication during cluster key establishment. In this scheme, some advanced nodes are used called H-sensor nodes responsible for key management. Every time H-Sensor nodes generate polynomial when change occurs in a cluster. It enhances the left-time of the sensor network by reducing the key pool size but this needs an advanced node which increases the cost of a network.

In [23] the author presents an energy efficient distributed deterministic key management algorithm (EDDK) for WMSN. EDDK concentrates on the establishment and updating of the pairwise keys, also the inter-cluster keys, and can settle a few imperfections in some current key management schemes. Construction of a neighboring table during key establishment not only gives the security to key support and information exchanges but it can likewise be utilized to adequately deal with the storage and refresh of the keys. By utilizing the elliptic curve digital signature algorithm in EDDK, both new and movable sensor nodes can join or leave or re-join a sensor network safely. The real reason for the low performance of EDDK is that it calculates the pairwise keys and changes in neighborhood impact on the estimation of pairwise keys, which may give the wrong example of pairwise keys and needs the recalculation of pairwise keys.

In [24] a new scheme called cluster based mobile key management system (CMKMS) considers two stages, first stage for key maintenance which sets up two private keys, home key for its cluster, and foreign key when a node move starting with one cluster then onto the next. The second stage keeps up the keys when CH moves starting with one group then onto the next. The proposed scheme enhances the efficiency of key management as far as security, energy saving, mobility, and network scalability. This scheme has efficiency because of using the RC4 algorithm for encryption and decryption.

Nabavi et al. [25] presents a novel key management scheme to enhance the energy efficiency, security, and scalability prerequisites by diminishing the computational complexity of the scheme. This scheme keeps running in two stages; in the first stage, it sets up the cluster and appoints the home and foreign keys to every node. The second stage keeps up the key update during the node and CH mobility. Besides, to improve energy efficiency, reduce computational overhead, and enhance encryption speed, the ECDSA encryption algorithm has been used.

Similarly, a lot of schemes [26–34] have been presented for dynamic key generation and authentication in cluster WMSN for a heterogeneous network, where advanced nodes are used for key generation and maintenance.

Turkanovic et al. [35] proposed a user-authentication technique for WSNs, which arranges a cluster key with an overall sensor node in a cluster. This scheme plan gives mutual authentication between the external user, sensor, and the base station. Their scheme is reasonable for the asset obliged sensor hubs as it utilizes just straightforward hash and bitwise XOR calculations.

Nonetheless, Farash et al. [36] pointed out a few security traps in Turkanovic et al. [35] plot scheme, for example, it doesn't give sensor obscurity and user discernibility, and it isn't likewise secure against man-in-the-middle, session key security, sensor impersonation, and stolen card attacks.

Rehman et al. [37] has proposed a scheme called P that generated polynomials whenever they are needed by nodes. Polynomials are generated dynamically when changes occur in the cluster to create a new cluster key (session key). The proposed key management scheme is secure against eavesdropping and node capturing by using efficient key management based on the dynamic generation of a polynomial. The proposed scheme has low communication, storage, and computation without compromising the security of key management. The number of keys stored in CH is considerably reduced and provides resistance against insider and outsider attacks. But this scheme does not consider the real-time data communication as required by an external user when went to join a cluster and access data directly from the cluster head instead of from the base station.

3 Threat and System Models

The threat model is based on Dolev-Yao model [38], initially, the nodes are deployed and can communicate over an open (insecure) channel. As the communication channel is open and the party like EU_i and CH_i are assumed to be untrusted.

Adversaries try to get important data from nodes or networks by deleting and modifying the transmitted data. The adversaries say Ai can be in the form of either active or passive attacks. Physical attacks can be launched by an adversary to compromising nodes and get secret data for future use. Inside and outside attacks can distribute the vital thing, caught data, interrupt the security collusively, and are called collusive physical attacks. For example, the CH_i node and the newly joined nodes can dispatch tricky attacks over the cluster key no longer having a place with them. Capturing node physical attacks is exceptionally destructive to the system if over-the-top cryptographic keys are still available inside a node. By and large, attacks influencing safe key distribution are eavesdropping and node capturing.

Our network consists of two types of nodes, External node user (EU_i) and base station BS as shown in Fig. 1. One type is a normal node that gather data from the environment and the second type of node is one which are selected as a cluster head (CH_i) using some well know algorithm [39]. These nodes have limited computation, communication, and storage capacity. The second type of node is BS having a lot of resources, powerful computation, and is the most trusted entity. To establish mutual authentication among the sensor nodes (sensors, cluster heads), each sensor needs to accomplish a shared authentication mechanism and create a dynamic shared cluster prior to the communication. At last, every node can confirm its authenticity using this shared with another. Any EUi can access real-time data directly from any CH_i after proper authentication and key agreement. BS provides authentication between EU_i and CH_i and after authentication EU_i and CH_i can establish a session key and start secure communication. The notations used in this study and its description are shown in Tab. 1.

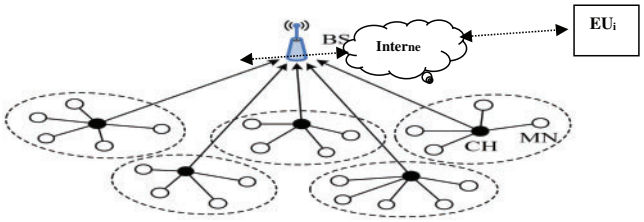

Figure 1: Cluster based network model of WMSN

Table 1: List of notations

Notations	Description
EU_i	External ith user
MB_i	External user ith mobile device
CH_i	ith cluster head
BS	Base station
EU_{Idi}, EU_{pwi}, EU_{Boi}	External user i's identity, password and biometric
$BS_{Id,}$ CH_{Id}	Base station and cluster head ID
BS_{Pid}, EU_{Pid}, CH_{Pid}	Base station, external user and cluster head pseudo identities
l, n	20 Bytes secret number of BS and EU_i
$r_{1,}$ r_2, r2	20 Bytes nonce of EU_i, BS and CH_i
EU_{RTi}, CH_{RTi}	EU_i and CH_i registration timestamp
EU_{MKi}, C_{MKi}	EU_i and CH_i 20 Bytes network cluster key
t_1, t_2, t_3	Current timestamp
T'	Message transmission delay
Gen/Rep(.)	Generation/Reproduction function (Fuzzy extractor)
φi	Biometric value of EU_i (biometric secret key)
P_{ti}	EU_i reproduction parameter
T_{er}	Fuzzy extractor error tolerance threshold
CSKi	Shared secret cluster key between member nodes, cluster head and external user

4 Preliminaries

This segment quickly presents fuzzy extractor work as follows. A fuzzy extractor alters the biometric input information into uniform random strings which at that point fills in as a biometric key [40]. Utilizing this calculation, any random length string Li could be transformed by consolidating a nonexclusive biometric input (pulse noise) Ji with the aide string Hi. This calculation needs two activities to work appropriately, for example, Gen and Rep. The Gen activity takes biometric input Ji and creates double output Li $\in \{0, 1\}$ *l* and an aide yield Hi $\in \{0, 1\} *$. The Li string is left well enough alone, while Hi is additionally put away. To recoup Li, the second activity Rep is utilized to utilize

the components Ji and Hi. To approve the rightness of the fuzzy extractor, the capacity ds (Ji, Ji∗) <= t and Gen (Ji) → (Li, Hi) is used. At that point, we get Rep (Ji∗, Hi) → Li, where ds show distance function and t_{err} as error threshold.

5 Proposed Scheme

The proposed scheme presented in this section consists of six steps: 1. Initial deployment, 2. External user registration, 3. External user login, 4. External user authentication and key management, 5. New cluster head addition and 6. is cluster head key management. One of the strong aspects of the proposed scheme is that it is based on a lightweight one-way hash function and bit-wise XOR operation as well as using fuzzy extractor technique which is only required for biometric verification of external users. A detailed explanation of each phase is given below.

In the initial deployment, it is the responsibility of BS to register each Ch_i. So, BS chose 20 Bytes unique secret number l and also the identity of CH_{Idi} of each CH. Pseudo ID is computed for each CH as $CH_{PId} = h(CH_{Idi}\|1)$and BS also selects 20 B network cluster key C_{Mki} to each CH_i. After that BS calculates temporary credential as $T_{Chi} = h(CH_{Idi}\|C_{Mki}\|CH_{RTi})$ and CH_{RTi} denotes registration timestamp.

As because of random mobility in WMSN, for the establishment of pair-wise key between two neighbour CH, BS uses asymmetric bivariate polynomial $p(x, y) = \sum_{i=1}^{k}\sum_{j=0}^{k} g_{i,j}x^i y^j \in GF(p)[x, y]$ of degree k over GF(p) and the coefficient of $g_{i,j}$ are taken from GF(P). p shows the selected larger prime and k must be larger than the number of selected CHs in the deployed network.

At all, BS station send the calculated values of {CHIdi, CH_{RTi}, T_{Chi}, $p(T_{Chi}, y)$} to CH in the secure channel, T_{Chi} is the CH_i temporary id. BS also stores the same information in its memory.

5.1 External User Registration

In this section, how the external user (firefighter) EU_i register in the network for accessing real-time data from a particular CH_i for providing relief to a particular region. For this EU_i first time register itself with BS using secure channel using the following step.

1. EU_i send registration request-massage including EU_{idi} to BS in secure channel. Upon receptions of this massage, BS compute $EU_{pid} = h(EU_{id} \| l)$ using 20 Bytes secret number l and at the same time BS calculates $BS_{Pid} = h(BS_{id} \| l)$ and $A = h(BS_{pid} \| EU_i)$ and temporary credential $T_{EUi} = h(EU_{idi} \| EU_{MKi} \| RT_{EUi})$. BS send reply massage to EU_i in secure channel including (RT_{EUi}, RT_{Chi}, RT_{BS}, T_{EUi}, A).

2. Upon the receipt of the reply massage from BS, EU_i selects his password PW_{Eui} and biometric EU_{Boi} at his mobile device MB_i. Widely used fuzzy extractor/verification [38] is applied from biometric verification. EU_i uses his MB_i to generate biometric key φ and its public parameter t_i as $Gen(EU_{Boi}) = (\varphi i, ti)$.

3. EU_i generates 20 B secret number n and computes $EU^{\sim}_{Pid} = EU_{Pid} \oplus h(PW_{Eui} \| \varphi)$, $CH^{\sim}_{Pid} = CH_{Pid} \oplus h(EU_{idi} \| PW_{Eui} \| \varphi)$, $T^{\sim}_{EUi} = T_{EUi}\ h(PW_{Eui} \| \varphi)$ and new temporary password $TPW_i = h(PW_{Eui} \| n)$, $BS^{\sim}_{Pid} = BS_{Pid} \oplus h(EU_{Pid} \| \varphi)$ and further more computes the below given as:

$$A^{\sim} = A \oplus h(EU_{Pid}\|\varphi i\|PW_{EUi}) \tag{1}$$

$$B = n \oplus h(PW_{EUi}\|EU_{idi}\|\varphi i) \tag{2}$$

$$C = h(A\|CH_{Pid}\|TPW_i\|\varphi i) \tag{3}$$

After all, U_i store the following information in its mobile device memory: {$EU^{\sim}_{Pid}$, $CH^{\sim}_{Pid}$, $BS^{\sim}_{Pid}$, T_{EUi}, $A^{\sim}$, B, C, t_i, Gen/Rep, h(.), t). BS also stores the following in its memory {EU_i, EU_{Pid}, T_{EUi}, BS_{Pid}}.

4. For Login into the system, EU_i generates T_1, 20 B nonce r_1 and computes the following using a mobile device:

$$K_1 = EU_{Pid} \oplus h(BS_{Pid} \| T_1) \tag{4}$$

$$K_2 = CH_{Pid} \oplus h(T_{EUi} \| EU_{idi} \| T_1) \tag{5}$$

$$K_3 = h(BS_{Pid} \| T_{EUi} \| T_1) \oplus r_1 \tag{6}$$

$$K_4 = h(EU_{id} \| BS_{Pid} \| CH_{Pid} \| T_{EUi} \| r_1 \| T_1) \tag{7}$$

Finally, EU_i send login-req including $M_1 = \{K_1, K2, K_3, K_4, T_1\}$ to BS using any open channel and the registration process is shown in Fig. 2.

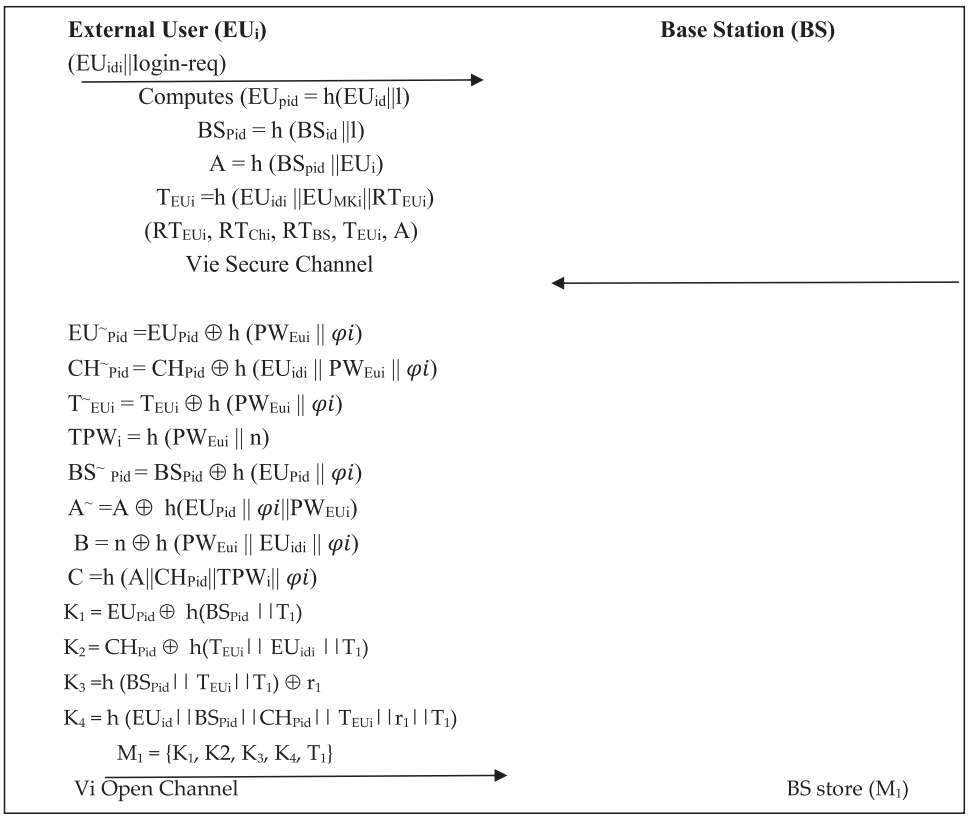

Figure 2: External user registration

5.2 Key Agreement

Upon the receipt of login-req massg from EU_i, the following communication step-wise are performed among BS, EU_i, and particular CH_i for the establishment of session key between EU_i and CH_i.

1. BS first checks the freshness of login-req masg by calculating $|T_1 - T^{\sim}_1| < T^{\sim}$ and when false then terminate it, otherwise BS computes $EU_{Pid} = K_1 \oplus h(BS_{Pid} \parallel T_1)$. For extraction of T_{EUi}, BS calculates the following:

$$CH_{Pid} = K_2 \oplus h(T_{EUi} \| EU_{idi} \| T_1) \tag{8}$$

$$r_1^{\sim} = K_3 \oplus h(BS_{id} \| T_{EUi} \| T_1) \tag{9}$$

$$K_4^{\sim} = h(EU_{idi} \| BS_{Pid} \| CH_{Pid} \| T_{EUi} \| r_1^{\sim} \| T_1) \tag{10}$$

When $K^{\sim}_4 = K_4$, then EU_i is truly authenticated by BS, otherwise, BS terminates session.

2. BS send auth-req masg to CH_i in open channel, including $\{K_5, K_6, K_7, T_2\}$, generates new nonce r_2 and computes K_5, K_6, K_7 as follows:

$$K_5 = h(T_{CHi} \| CH_{Pid}) \oplus H(BS_{Pid} \| r_1 \| r_2) \tag{11}$$

$$K_6 = h(T_{Chi} \| T_2) \oplus EU_{Pid} \tag{12}$$

$$K_7 = h(CH_{Pid} \| T_{Chi} \| h(BS_{Pid} \| r_1 \| r_2 \| T_1) \tag{13}$$

3. When CH_i received auth-reg masg, first check the freshness of this massage by computing $|T_2 - T^{\sim}_2| < T^{\sim}$. When this condition true, then CH_i computes the following:

$$EU_{Pid} = K_6 \oplus h(T_{Chi} \| T_2) \tag{14}$$

$$K_8 = K_5 \oplus h(T_{Chi} \sim \| \sim CH_{Pid}) \tag{15}$$

$$K_9 = h(CH_{Pid} \| T_{Chi} \| K_8 \| T_2) \tag{16}$$

CH_i check $K_9 = K_7$, if it is true then BS is authenticated by CH_i, otherwise CH_i terminates session.

After BS authentication, CH_i generates nonce r_3 and new timestamp T_3, and compute $K_{10} = h(CH_{Pid} \parallel EU_{Pid} \parallel T_3)\, r_3$, cluster session key $CSK_i = h(K_8 \parallel r_3 \parallel EU_{Pid} \parallel CH_{Pid})$ which shared with EU_{id}. CH_i also generates $K_{11} = h(EU_{Pid} \parallel CH_{Pid} \parallel r_3) \oplus K_8$ and $K_{12} = h(CSK_i \parallel T_3)$. CH_i send auth-reply massg $M_2 = \{K_{10}, K_{11}, K_{12}, T_3\}$ to EU_i in open channel.

4. Upon reception of auth-reply masg, EU_i first check freshness of this masseg by find $|T_3 - T_3| < T^{\sim}$, if true then computes $r^{\sim}_3 = K_{10} \oplus h(CH_{Pid} \parallel EU_{Pid} \parallel T_3)$. $K^{\sim}_8 = K_{11} \oplus h(EU_{Pid} \parallel CH_{Pid} \parallel r^{\sim}_3)$, $CSK^{\sim}_I = h(K^{\sim}_8 \parallel r^{\sim}_3 \parallel EU_{Pid} \parallel CH_{Pid})$ and $K_{13} = h(CSK^{\sim}_2 \parallel T_3)$ and if $K_{13} = K_{12}$ then CH_i is authenticated by EU_i. After that EU_i computes $CSK^{\sim}_I = CSK_i$ then maintain this cluster session key for communication as shown in Fig. 3.

5.3 New/Existing Cluster Head Addition

As in WMSN, all nodes are mobile and can move in any direction. So, it is the possibility that CH_i can move from an existing cluster and join a new cluster. Because of mobility in WMSN, the node may be moved from one cluster to another or a new cluster head can be added to the network. A scalable key management scheme needs the capacity of adding a new node to the network. These new nodes require building a new cluster shared cluster key (CSKi) with the existing cluster head for authentication. When a new CH_j tries to join a cluster, it sends a join request message containing ID to the correspondence CH_i. BS plays an important role in the authentication process.

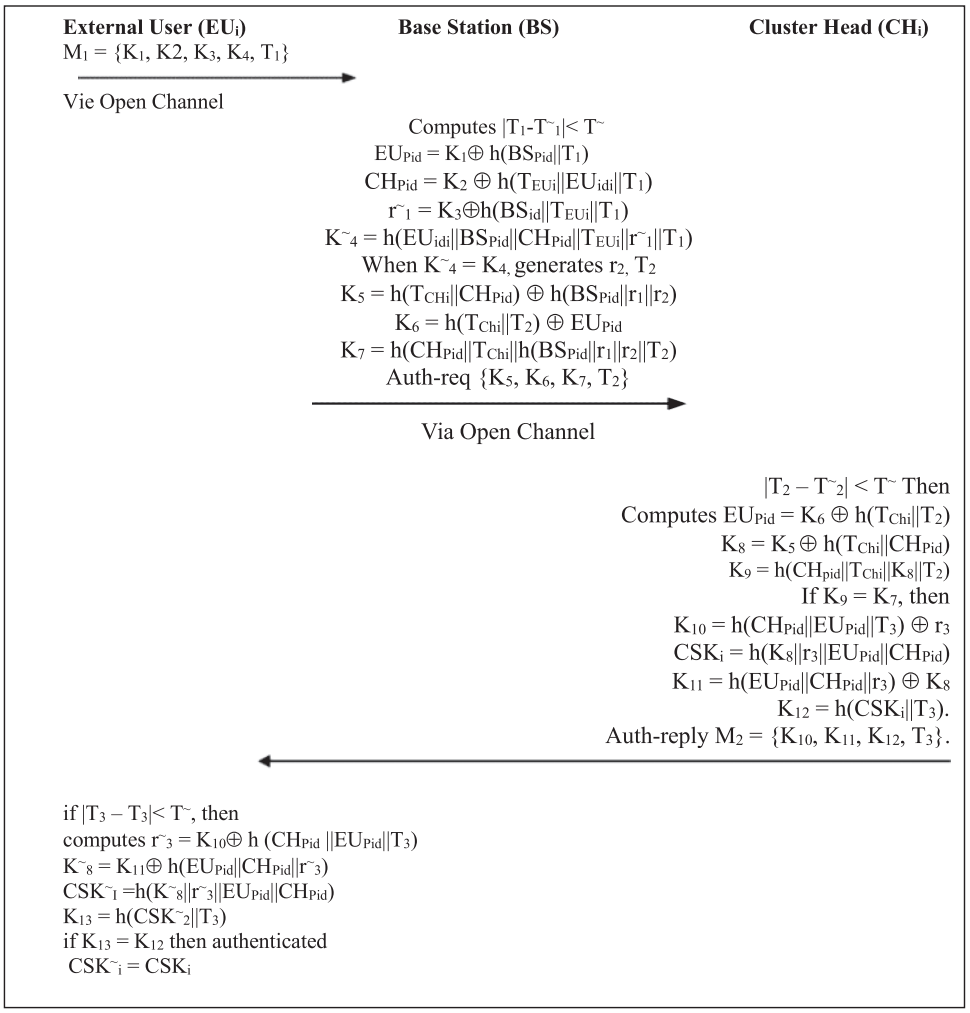

Figure 3: Authentication and key agreement

When CH_i leaves their cluster and tries to join their neighbour cluster then BS generates used their CH_{idi} for the calculation of new-pseudo identity $CH_{Pid}^{N} = h(CH_{idi} \parallel l)$ using the l secrete key of BS. BS also select 20B new network key C_{MKi}^{N} and computes new temporary credential as $T_{Chi}^{N} = h(CH_{idi} \parallel C_{MKi}^{N} \parallel CH_{RTi}^{N})$. Furthermore, BS also generates a new temporary ID CH_{Tidi} and computes the polynomial $p(CH_{Tidi}, y)$.

BS station send some data {CHTidi, CH_{Pid}^{N}, T_{Chi}^{N}, $p(CH_{Tidi}, y)$} to the migrated CH_i. BS also broadcast all the nodes of the network about the new change.

5.4 Dynamic Key Management

The management of mobility in WMSN is an important and critical issue. As discus earlier, because of the mobility of nodes, CH_i can share their data to neighbouring CH_j. For sharing of information between neighbouring cluster heads (CH_i, CH_j), there must be first secure communication

between these nodes. This requires pair wise key management between these nodes. For the establishment of pair wise key between these nodes (CH_i, CH_j), we use a polynomial based key management scheme [41].

First CH_i sends their temporary identity CH_{Tidi} to CH_j and CH_j also sends their CH_{Tidj} to CH_i for pair wise key establishment. CH_i computes shared secret cluster key $CSK_{i,j}$ using share polynomial as:

$$CSK_{i,j} = p(CH_{idi}, CH_{Tidi}) \tag{17}$$

In the same way CH_j also calculates share secret key using their polynomial as:

$$CSK_{i,j} = p(CH_{Tidi}, CH_{Tidi}) \tag{18}$$

$$= p(CH_{idi}, CH_{Tidj}) \tag{19}$$

$$= SK_{i,j} \tag{20}$$

Hence CH_i can communicates with neighbouring CH_j using share key $CSK_{i,j}$ through polynomial p(x, y).

6 Security Analysis

This section analyses the security features of the proposed scheme. The threats and attack models try to affect the key management in cluster communication of two types, one is inside attacks and the second is outside attacks. The proposed scheme is secure against physical capture and offline password guessing attack. Assume if attackers say A find the ID's of EU_{idi} or extract information $\{EU^{\sim}_{Pid}, CH^{\sim}_{Pid}, EU_{Tid}, A, B, C, t\}$ from EU_{idi} biometric device by physical capture or stolen this device after the completion of registration of the external user. After performing power analysis attack [41], A can computes secret credential $n = B \oplus h(PW_{EUi} \parallel EU_{idi} \parallel \varphi)$. Without EU_{idi} biometric key ϕ, A cannot verify their guessed password and at the same time with φ, it is very difficult to compute $EU_{pid,}$ CH_{pdi} and T_{Eui}.

The proposed scheme is secured again external user impersonation attack. Assume an attackers say A send login-req message including $M^{\sim} = \{M^{\sim}_1, M^{\sim}_2, M^{\sim}_3, M^{\sim}_4\}$ to BS on behalf of valid EU_i. But it's very hard for A to computes valid login-req on behalf without some security credential $\{EU_{Pid}, CH_{Pid}, C_{NKi}, \varphi)$.

Due to massages exchange having nonce and new timestamps in login, authentication, and key agreement, the proposed scheme is secure against anonymity and un-traceability attacks. Because of different messages, the attacker is unable to trace out EU_i, CH_i, and BS.

The $SK_{i,j}$ is established through the distribution of polynomial which provides the secure establishment of key between member nodes. Sharing of shared secret cluster key $SK_{i,j}$ through polynomial among cluster heads restrict attacker to know about cluster key. Without knowing the $SK_{i,j}$, an attacker cannot intercept member communication individually and not be able to modify it. The proposed scheme ensures integrity and confidentiality through resistance against insider attacks.

The non-member nodes of the cluster can eavesdrop on the cluster information through outsider's attack. When the CH sends the polynomial without encryption then the outsider attacker attempts to eavesdrop that communication. Without knowing the $SK_{i,j}$, and even with no encryption, it is very hard to recover the cluster key. On the off chance that the adversary tries the experimentation approach, the exponential log implements extra complexity. This approach may occupy the attacker to discover the log value for the separate ID value. Deriving the $SK_{i,j}$ using polynomial factorization is additionally

exceptionally hard. To guess the cluster key, an extended polynomial needs really O(n log n) solutions for a polynomial extension problem. Here, the presence of cluster key in the polynomial forces is difficult in the polynomial factorization and making the polynomial factorization with a specific end goal to break the proposed scheme is non-deterministic polynomial-time (NP) hard.

The proposed scheme also secures against Denial-of-Service (DoS) attack and men-in-the-middle attack. The EU_i in the proposed scheme verifies mutually established session key CSK_i between the user and CH_i with the help of BS. The verification of parameters by comparing $K\tilde{}_4 = K_4$ ensures the authenticity of the recovered session key by the user, while K4 is computed by embedding various factors including the current session key CSK_i.s

7 Result and Discussions

The proposed solution has been validated through simulation and compares its performance with the rest of schemes including EKMS, EDDK, SKM, and PKM. The result comparison among the proposed scheme and with the rest of the schemes has been carried out using the following simulation parameters shown in Tab. 2. The proposed system efficiency is analyzed based on the cost effectiveness by using these parameters: communication overhead, computational overheads, storage overhead, energy consumption, and average latency.

Table 2: Simulation parameters

Parameters	Values
Mobility model	Random way point
Number of sensor nodes	100
Length of data packet	256 Bit
Length of control packet	50 Bit
Initial energy	1Joule
Interface queue type	Proposed scheme
Communication model	Bi-directional
Simulation area	1000 × 1000 m2
Node speed	1–15 m/s
Maximum queue	50 packets
Simulation time	300 s

Communication overhead is the measurement of the number of bits transmission for the establishment of cluster key in case of new external node addition or existing external node migration between CH and member nodes of the cluster. Fig. 4 shows the communication overhead for establishing sessions among all participating nodes in case of news external user addition to cluster. The main reason of the lower performance of EDDK is the usage of pairwise key and local cluster key for each node in the cluster. In case of new node addition first pairwise keys are established, then a session key will be established which increases communication cost. Similarly, EKMS and SKM used local cluster and foreign keys for the establishment of new cluster key which also increases communication overhead. Our proposed scheme has a low communication overhead because session key establishment and authentication of external users are performed local without base station collaboration. There is no need to exchange any messages between BS and cluster head for session key establishment and

external user authentication. Hence the proposed scheme has superior performance in communication cost compared with the other three schemes for secure new node addition in the cluster.

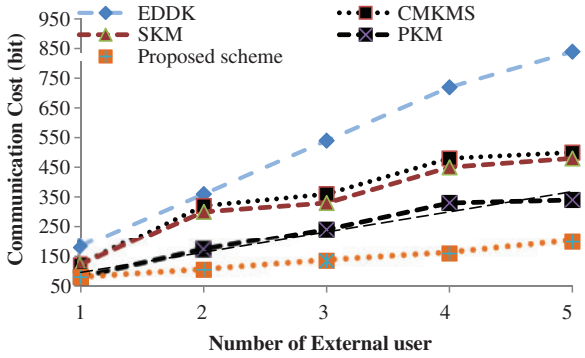

Figure 4: Communication cost in case of external node

Similarly, communication overhead in case of increasing external user is shown in Fig. 5. Every external user sends leave and joint request to their alternative CH for leaving and joining new one. CH should update their session for their member as authentication process is completed locally. In EDDK, node has exchange two keys for leaving and two for joining which increases communication overhead. Similarly, EKMS and rest schemes has higher communication cost when existing node move from one cluster to another cluster. Our proposed scheme performed better then because authentication and session key are manged at every cluster head locally. So, the proposed scheme performed better as compared with the rest of the schemes.

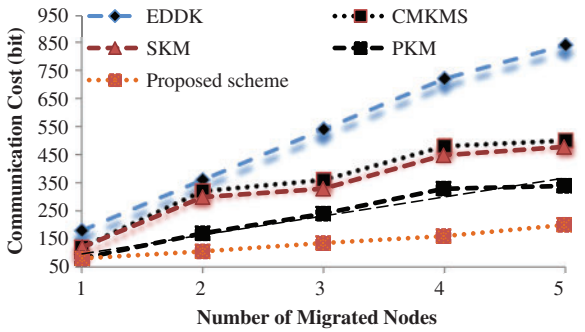

Figure 5: Communication overhead in case of node migration

Tab. 3 show the Storage overhead of the proposed scheme, EDDK, EDMS and SKM with other schemes. The amount of storage capacity required to store security parameters for session key generation and authentication of external user is considered is storage overhead. EDDK has the worst storage due to the fact that each sensor node requirements for storing a neighbour table because each node have to generate new common key for the new node addition.

Additionally, CH need more storage for storing all member nodes keys and tables. EKMS and scheme has also high storage overhead because of usage: of two keys *i.e.*, one is home key and another is foreign key for the generation of cluster key. The proposed scheme has less storage overhead because CH store only one ID and (m + 1) log2(p) + l(20) bytes data for session key establishment an authentication of external user.

Table 3: Storage overhead comparison

Name of scheme	Data stored in CH	Data stored in external user	Data stored in BS
EDDK	n + n ∗ neighbor table	2 + 3 ∗ neighbour table	N + 3 ∗ neighbour table
CMKMS	2 ∗ n + polynomial	2 keys + polynomial	2 ∗ N + polynomial
SKM	2 ∗ n + polynomial	2 keys + polynomial	2 ∗ N + polynomial
PKM	N + 1+ polynomial	1 key + polynomial	N + 1 + polynomail
Proposed scheme	(m + 1) $\log_2(p)$ + l(20 B) + CH_{Pid}	ID + 20 B + EU_{Pid} + 20 B C_{MKi} + φ	(m+1) $\log_2(p)$ + 100 B + l (20 B) + C_{MK} (20B)

Tab. 4 and Fig. 6 show the computation overhead comparison of the proposed scheme with the rest of the three schemes for establishing a session key and authentication of the user. Encryption and decryption plus one pseudo random generation function execution operations are performed to established a secure cluster key in EDDK.

Similarly, EKMS and SKM schemes also need encryption and decryption operations for establishing cluster keys for the entire cluster. In the proposed scheme single encryption and decryption operation are required for establishing session key and for external user authentication. Thus, the proposed scheme performed better performance in term of less computation cost.

Table 4: Computational cost comparison

Name of schemes	Encryp/Decryp in CH	Encyp/Decryp in member node
EDDK	Encryp + Decryp + 1 pseudo random function	Encryp + Decryp
CMKMS	Encyp + Decryp	Encypt + Decryp
PKM	Encyp + Decryp	Encyp + Decryp
SKM	Polynomial function	Function
Proposed scheme	Encryption	Decryp

The amount of energy consumed during cluster key establishment when member nodes leave or join a cluster in EDDK, EKMS, and SKM schemes is compared with the proposed scheme. EDDK has lower performance in terms of energy consumption because the change in neighbouring nodes affects the calculation of keys, which may give the wrong instance of keys and re-calculation is needed. The main reason EKMS has lower energy efficiency is the usage of the RC5 algorithm for encryption and decryption involved during node leaving or joining of a cluster. Similarly, the scheme SKM also used the ECDSA algorithm for encryption and decryption in case of leaving or joining a cluster. The total energy required to compute a polynomial of the cluster is linear, while the rest of the three schemes have n multiple and n shows the number of nodes in a cluster. The energy consumption of computing a secure hash of sensor node id is 5.6 nJ per byte. For a cluster of n nodes, the hash computation of member node n consumed n ∗ 5.6 nJ. Fig. 6 shows the energy consumption of cluster key establishment of the proposed scheme compared with the rest of the three schemes.

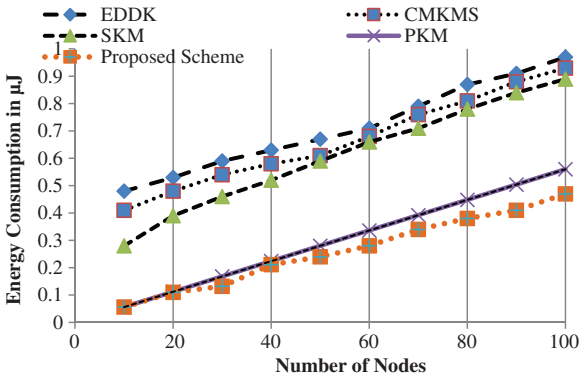

Figure 6: Energy consumption of nodes

8 Conclusion

This works proposed an efficient external user authentication and key management in real-time access information for cluster WMSN. One of the strong aspects of the proposed scheme is that it is based on a lightweight one-way hash function and bit-wise XOR operation as well as using fuzzy extractor technique which is only required for biometric verification of external user. The external users EU_i registered in the network for accessing real-time data from particular CH_i for providing relief to this region without the involvement of BS during the authentication process. BS is only involved during the registration process for the first time and the cluster session key is established without BS. The management of mobility in WMSN is achieved using dynamic key management for cluster WMSN. The proposed scheme ensures integrity and confidentiality through resistance against insider attacks. The proposed key management scheme is secure against eavesdropping and node capturing by using an efficient authentication for an external user and key management based on dynamic generation of polynomial and fuzzy extractor method. The proposed scheme has low communication, storage, and computation without compromising the security of external user authentication and key management in WMSN.

For future direction, this scheme will analyze the different security aspects of the network and further improvements will be made for efficient communication and quality of services.

Acknowledgement: I would like to thank all the people who contributed in some way to the work described in this article.

Funding Statement: This research was financially supported in part by the Ministry of Trade, Industry and Energy (MOTIE) and Korea Institute for Advancement of Technology (KIAT) through the International Cooperative R&D program. (Project No. P0016038) and in part by the MSIT (Ministry of Science and ICT), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2021-2016-0-00312) supervised by the IITP (Institute for Information & communications Technology Planning & Evaluation).

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] J. Yick, B. Mukherjee and D. Ghosal, "Wireless sensor network survey," *Computer Networks*, vol. 52, no. 12, pp. 2292–2330, 2008.

[2] C. Karlof, N. Sastry and D. Wagner, "TinySec: A link layer security architecture for wireless sensor networks," in *Proc. of the 2nd Int. Conf. on Embedded Networked Sensor Systems*, Baltimore MD, USA, pp. 162–175, 2004.

[3] A. Abbasi and M. Younis, "A survey on clustering algorithms for wireless sensor networks," *Computer Communications*, vol. 30, no. 14–15, pp. 2826–2841, 2007.

[4] D. Djenouri, L. Khelladi and A. Badache, "A survey of security issues in mobile ad hoc and sensor networks," *IEEE Communications Surveys & Tutorials*, vol. 7, no. 4, pp. 2–28, 2005.

[5] K. Venkatraman, J. Daniel and G. Murugaboopathi, "Various attacks in wireless sensor network Survey," *International Journal of Soft Computing and Engineering (IJSCE)*, vol. 3, no. 1, pp. 208–212, 2013.

[6] A. Diop, Y. Qi and Q. Wang, "An efficient and secure session key management scheme for cluster based wireless sensors networks," in *Proc. of Joint Int. Conf. on Pervasive Computing and the Networked World*, Istanbul, Turkey, pp. 33–44, 2013.

[7] I. Tomić and J. McCann, "A survey of potential security issues in existing wireless sensor network protocols," *IEEE Internet of Things Journal*, vol. 4, no. 6, pp. 1910–1923, 2017.

[8] A. Ameen, M. Liu and K. Kwak, "Security and privacy issues in wireless sensor networks for healthcare applications," *Journal of Medical Systems*, vol. 36, no. 1, pp. 93–101, 2012.

[9] A. Derhab, A. Bouras, I. M. R. Senouci and Imran, "Fortifying intrusion detection systems in dynamic ad hoc and wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 10, no. 12, pp. 608162, 2014.

[10] J. Zhang and V. Varadharajan, "Wireless sensor network key management survey and taxonomy," *Journal of Network and Computer Applications*, vol. 33, no. 2, pp. 63–75, 2010.

[11] A. Ghafoor, M. Sher, M. Imran and A. Derhab, "Secure key distribution using fragmentation and assimilation in wireless sensor and actor networks," *International Journal of Distributed Sensor Networks*, vol. 11, no. 9, pp. 542856, 2015.

[12] K. Saleem, M. S. Khalil, N. Fisal, A. Ahmed and M. Orgun, "Efficient random key based encryption system for data packet confidentiality in WSNs," in *Proc. of 12th IEEE Int. Conf. on Trust, Security and Privacy in Computing and Communications*, Melbourne, Australia, pp. 1662–1668, 2013.

[13] K. C. Wong, M. Gouda and S. S. Lam, "Secure group communications using key graphs," *IEEE/ACM Transactions on Networking*, vol. 8, no. 1, pp. 16–30, 2000.

[14] C. J. Lin, F. Lai and C. H. Lee, "Efficient group key management protocol with one-way key derivation," in *Proc. of IEEE Conf. on Local Computer Networks 30th Anniversary (LCN'05)*, Sydney, NSW, Australia, pp. 336–343, 2005.

[15] D. Estrin, R. Govindan, J. Heidemann and S. Kumar, "Next century challenges: Scalable coordination in sensor networks," in *Proc. of the 5th Annual ACM/IEEE Int. Conf. on Mobile Computing and Networking*, Seattle Washington USA, pp. 263–270, 1999.

[16] S. Zhu, S. Setia and S. Jajodia, "LEAP + Efficient security mechanisms for large-scale distributed sensor networks," *ACM Transactions on Sensor Networks (TOSN)*, vol. 2, no. 4, pp. 500–528, 2006.

[17] G. C. Agrawal and B. J. Kulkarni, "Enhancing the security in WSN using three tier security architecture," *International Journal of Innovative Research in Information Security (IJIRIS)*, vol. 1, no. 2, pp. 40–47, 2014.

[18] X. Yan, B. Li and X. Ye, "A key management scheme for mobile heterogeneous sensor networks," *Journal of Naval University of Engineering*, vol. 2014, no. 2, pp. 99–103, 2014.

[19] S. Banihashemian and G. A. Bafghi, "A new key management scheme in heterogeneous wireless sensor networks," in *Proc. of 12th Int. Conf. on Advanced Communication Technology (ICACT)*, Gangwon-Do South Korea, vol. 1, pp. 141–146, 2010.

[20] M. A. J. Naranjo, N. Antequera, G. L. Casado and A. J. López-Ramos, "A suite of algorithms for key distribution and authentication in centralized secure multicast environments," *Journal of Computational and Applied Mathematics*, vol. 236, no. 12, pp. 3042–3051, 2012.

[21] A. Ghafoor, M. Sher, M. Imran and K. Saleem, "A lightweight key freshness scheme for wireless sensor network," in *Proc. of 12th Int. Conf. on Information Technology-New Generations*, Las Vegas, NV, USA, pp. 169–173, 2015.
[22] M. Li, J. Long, J. Yin, Y. Wu and J. Cheng, "An efficient key management based on dynamic generation of polynomials for heterogeneous sensor networks," in *Proc. of 2nd Int. Conf. on Computer Engineering and Technology*, Chengdu, China, vol. 5, pp. V5–460, 2010.
[23] X. Zhang, J. Hep and Q. Wei, "EDDK: Energy-efficient distributed deterministic key management for wireless sensor networks," *EURASIP Journal on Wireless Communications and Networking*, vol. 2011, no. 1, pp. 1–11, 2011.
[24] D. S. Babar, R. N. Prasad and R. Prasad, "CMKMS: Cluster-based mobile key management scheme for wireless sensor network," *International Journal of Pervasive Computing and Communications*, vol. 10, no. 2, pp. 196–211, 2014.
[25] R. S. Nabavi and M. S. Mousavi, "A novel cluster-based key management scheme to improve scalability in wireless sensor networks," *International Journal of Computer Science and Network Security*, vol. 16, no. 7, pp. 150–156, 2016.
[26] A. Diop, Y. Qi, Q. Wang and S. Hussain, "An efficient and secure key management scheme for hierarchical wireless sensor networks," *International Journal of Computer and Communication Engineering*, vol. 1, no. 4, pp. 365–370, 2012.
[27] Y. Zeng, J. S. B. Zhao, X. Yan and Z. Shao, "A loop-based key management scheme for wireless sensor networks," in *Proc. of Int. Conf. on Embedded and Ubiquitous Computing*, Berlin, Heidelberg, Springer, pp. 103–114, 2007.
[28] F. Kausar, S. Hussain, T. L. Yang and A. Masood, "Scalable and efficient key management for heterogeneous sensor networks," *The Journal of Supercomputing*, vol. 45, no. 1, pp. 44–65, 2008.
[29] F. Kausar, S. Hussain, H. J. Park and A. Masood, "Secure group communication with self-healing and rekeying in wireless sensor networks," in *Proc. of Int. Conf. on Mobile Ad-Hoc and Sensor Networks*, Berlin, Heidelberg, pp. 737–748, 2008.
[30] Y. Sun, W. Trappe and R. K. Liu, "An efficient key management scheme for secure wireless multicast," in *Proc. of IEEE Int. Conf. on Communications. Conf. Proc., ICC 2002, Cat. No. 02CH37333*, New York, USA, vol. 2, pp. 1236–1240, 2002.
[31] B. Y. Saied, A. Olivereau and D. Zeghlache, "Energy efficiency in M2M networks: A cooperative key establishment system," in *Proc. of 3rd Int. Congress on Ultra-Modern Telecommunications and Control Systems and Workshops (ICUMT)*, Budapest, Hungary, pp. 1–8, 2011.
[32] S. Gao, H. Van, M. Kaltofen and V. Shoup, "The computational complexity of polynomial factorization," *American Institute of Mathematics*, San Jose, California, vol. 364, pp. 15–19, 2006.
[33] E. Rehman, M. Sher, S. H. A. Naqvi, K. B. Khan and K. Ullah, "Energy efficient secure trust based clustering algorithm for mobile wireless sensor network," *Journal of Computer Networks and Communications*, vol. 2017, pp. 1–8, 2017.
[34] E. Rehman, Ismail and S. H. A. Naqvi, "Ensuring quality of service using multi-criteria quadrant based clustering (MCQC) protocol for wireless sensor networks," *Journal of Information Communication Technologies and Robotic Applications*, vol. 8, no. 2, pp. 18–29, 2018.
[35] M. Turkanović, B. Brumen and M. Hölbl, "A novel user authentication and key agreement scheme for heterogeneous ad hoc wireless sensor networks, based on the internet of things notion," *Ad Hoc Networks*, vol. 20, pp. 96–112, 2014.
[36] S. M. Farash, M. Turkanović, S. Kumari and M. Hölbl, "An efficient user authentication and key agreement scheme for heterogeneous wireless sensor network tailored for the internet of things environment," *Ad Hoc Networks*, vol. 36, no. 6, pp. 152–176, 2016.
[37] E. Rehman, M. Sher, S. H. A. Naqvi and A. Ghani, "Polynomial based dynamic key management for secure cluster communication in wireless mobile sensor network," *Journal of Tehnički vjesnik*, vol. 27, no. 2, pp. 358–367, 2020.

[38] I. Cervesato, "The Dolev-Yao intruder is the most powerful attacker," in *Proc. of 16th Annual Symp. on Logic in Computer Science—LICS*, NW Washington DC, United States, vol. 1, 2001.
[39] E. Rehman, M. Sher, S. H. A. Naqvi, K. B. Khan and K. Ullah, "Secure cluster head selection algorithm using pattern for wireless mobile sensor networks," *Journal of Tehnički vjesnik*, vol. 26, no. 2, pp. 302–311, 2019.
[40] Y. Dodis, B. Kanukurthi, J. Katz, L. Reyzin, Smith *et al.,* "A robust fuzzy extractors and authenticated key agreement from close secrets," *IEEE Transactions on Information Theory*, vol. 58, no. 9, pp. 6207–6222, 2012.
[41] Blundo, Carlo, A. De Santis, A. Herzberg, S. Kutten *et al.,* "Perfectly-secure key distribution for dynamic conferences," in *Proc. of Annual Int. Cryptology Conf.*, Berlin, Heidelberg, Springer, pp. 471–486, 1992.