논문 2023-4-14 http://dx.doi.org/10.29056/jsav.2023.12.14

Improved User Authentication in Hierarchical Wireless Sensor Networks

Dong-Hoon Kim*, Ki Young Lee**†

Abstract

One of the most vulnerable issues for WSN security are outside attacks. To defense against these attacks efficiently, user authentication is an important subject. We present an improved user authentication scheme in Hierarchical Wireless sensor networks. This scheme performs one-way, cascading mutual authentication between three parties: the gateway, user, and cluster head in the network. In order to counter threats such as replay attacks, S/Key algorithm-based one-time passwords are used for each authentication. After secure session establishment between user and cluster head, a secret key agreement protocol between cluster head and the sensor node belonging to it is applied for secure communication. The proposed protocol's security is evaluated against various attacks like guessing, stolen-verifier, playback, bypass, impersonation, and data integrity. Additionally, the performance analysis reveals its efficiency, requiring only three transmissions, enabling reduced network congestion and energy consumption compared to other schemes. This protocol ensures mutual authentication among all components —gateway, user, and sensor node— reducing masquerade attack risks, albeit with a slightly increased computational overhead due to encryption processes.

keywords : User Authentication, Hierarchical Wireless Sensor Networks, Cluster Formation, D2D Communication, Gateway

1. Introduction

Wireless sensor networks (WSNs) serve various real-time applications, such as vehicular tracking and habitat monitoring. These networks consists of numerous low-cost, battery-operated sensor nodes, each with limited computational capabilities. Data gathered by these nodes is relayed in an ad-hoc manner to a gateway or base station, which connects to an application system. The primary aim is to access data from specific nodes securely, often requiring user authorization.

Given the resource limitations of sensor nodes, simple and energy-efficient user authentication is crucial. To mitigate computational overheads during key distribution, WSNs are often organized into hierarchical structures known as hierarchical

* Department of Software, Kwangwoon University
** Department of Info and Telecom Eng, Incheon National University
† Corresponding Author : Ki Young Lee (email: kylee@inu.ac.kr)
Submitted: 2023.12.08. Accepted: 2023.12.15. Confirmed: 2023.12.20.

wiress sensor networks (HSWN) which is dipicted in Fig. 1. In HSWNs, clusters —each overseen by a cluster-head (CH)— reduce communication complexity. These CHs communicate with nodes within their cluster and with the gateway, forming the backbone of data transmission.

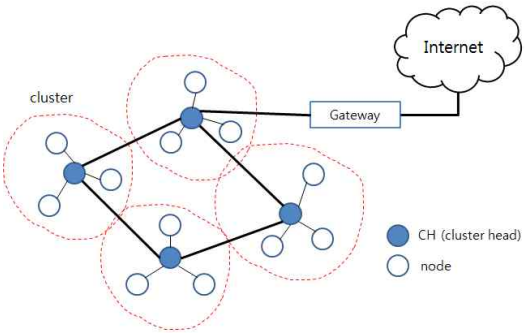

Fig. 1. A hierarchical wireless sensor network.

The challenge in HSWNs lies in identifying CHs efficiently. The concept of a connected dominating set (CDS) helps designate CHs by minimizing the number of such nodes. Algorithms like the generalized self-pruning rule aid in distributed CDS construction, vital in WSNs lacking centralized administration.

This paper explores a user authentication scheme within HSWNs, leveraging a 3-way, ring-structured, and one-way chaining authentication approach. By employing a secure session between users and cluster heads, we ensure data access with robust security measures against various outside attacks. The subsequent sections delve deeper into related works, the proposed scheme, security analysis, and concluding remarks.

2. Related Works

Over the preceding years, several user authentication schemes have emerged for WSNs, with some relying solely on passwords, while others incorporates two or more factors for authentication. Wong et al. introduced a dynamic solution based on strong-passwords for WSNs characterized by minimal computational load involving straightforward operations like hash functions and XOR operations [1]. Further enhanced dynamic user authentication schemes are outlined in [2-5]. Tseng et al. identified certain security issues within Wong et al.'s scheme, notably susceptibility to replay attacks and the potential exposure of passwords by any sensor node. By implementing enhancements to Wong at al.'s scheme, they provide resistance against replay attacks and mitigate the risk of password leakage [2]. Vaidya et al. highlighted that prior dynamic user authentication schemes lack mutual authentication between the user and gateway [5]. M. L. Das highlighted the vulnerability of existing protocols to threats such as multiple users sharing the same login ID and stolen verifier attacks. In response, Das introduced a two-factor user authentication scheme for WSNs, exclusively leveraging a one-way hash function. This proposed solution effectively addresses numerous risks, including scenarios with multiple users having identical login identities, stolen-verifier attacks, guessing, impersonation, and replay attacks [6]. Both K. S. Arikumar et al. and B. Vaidya et

al. have identified certain security vulnerabilities within Das's scheme. They propose enhancements such as password changes and the implementation of mutual authentication between the gateway and sensor nodes to enhance its security [7][8]. However, these improved schemes necessitate additional packet transmissions for mutual authentication, potentially leading to increased network traffic.

Das et al. introduced a dynamic password-based user authentication scheme tailored for hierarchical WSNs [9]. This plan utilizes direct collaboration of terminal nodes with the client, without the base station. It utilizes the acclaimed Dolev-Yao threat model [10], wherein two imparting parties (nodes) communicate through an insecure channel. In this threat model, the channel is deemed insecure, and the end-points (users, cluster heads, sensor nodes) are generally considered untrustworthy.

But later Turkanovic & Holbl [11] shown the faults in this scheme and proved it is infeasible for implementation. They [11] proposed a two-sided validation technique among sensor nodes, users and the base stations or gateway nodes using smart cards. This is a light weight authentication strategy, which is more alluring in IoT, due to its resource-constraint nature. They proposed an improved dynamic password-based user validation technique for hierarchical WSNs. They clearly state the flaws and reasons why Das et al.'s technique [9] cannot be implemented in reality and proposed flaw less scheme. It also reduced the number of hash functions. Thus, this provided security from the well- known attacks. But still vulnerable to some other attacks.

3. A User Authentication with Cluster Formation Scheme

Table 1. the list of notations

Notation	Description
U_i	the i-th user
ID_i, PW_i	the i-th user *identifier* and *password*
G	gateway node
ID_{Si}	the i-th sensor node *identifier*
CH_j	the cluster head in the j-th cluster
ID_{CH}	the identifier of the cluster head CHj
MK	the generated master key
E(·)	symmetric key encryption algorithm
D(·)	symmetric key decryption algorithm
H(·)	a secure one-way hash function
MAC	message authentication code
X_S	the private key maintained by a gateway
X_A	the public key shared between an user and a gateway
A || B	concatenation operation of A and B
A ⊕ B	exclusive OR operation of A and B

In this section, we describe the hierarchical wireless sensor network and our proposed user authentication scheme with cluster formation in more detail. Prior to delving into the discussion, Table 1 provides a comprehensive summary of the notations utilized across this paper.

3.1 Hierarchical Wireless Sensor Networks

The data from a node will be sent to the gateway either at regular interval or upon event detection. Typically, real-time data might not be accessed through the gateway or centralized server systems; rather, direct access from a node could be the norm. The primary goal of a WSN application is to access data from the designated node at a specific location and deliver it to a user. Thus, it is necessary that such an access is allowed only to a registered user. To access real-time information from the nodes, the user must first be authorized for both the nodes and the gateway, preventing unauthorized access to the nodes.

Given the limited resources and computational capacity of sensor nodes, an ideal user authentication scheme should prioritize simplicity and efficiency. To minimize computational burdens during key distribution across a WSN, the network is partitioned into clusters, as illustrated in Fig. 1. This type of a WSN organization is hierarchical wireless sensor networks (HSWN).

Every cluster is designated by a single cluster-head (CH). A CH serves as a node capable of establishing 1-hop wireless communication with all other nodes within the cluster. Sensor nodes therefore communicate only with the CH and other sensor nodes in the cluster. Finally, the hierarchy ends with the gateway, whereby it communicates only with the CHs of the network and to the outside world as an access point for the collected data.

How can we find CHs in a HWSN? This is a well-known problem of finding the minimum number of a connected dominating set (CDS). Within a CDS, every node either belongs to the CDS itself or maintains at least one (1-hop) neighboring node within the CDS. A dominating node plays a role as CH and then CH and its neighboring nodes consist of a single cluster. To minimize the number of relaying nodes involved in a broadcast task, only nodes designated as dominant are required to serve as routers for relaying the broadcasting packet. A highly efficient distributed algorithm, termed the Generalized Self-Pruning Rule, has been introduced for computing CDS solely based on local information. In WSNs, distributed CDS construction proves more effective, given the absence of centralized administration. Conversely, the sizable problem scale hampers centralized CDS computation.

The conventional representation of a WSN entails a graph G = (V, E), where V represents the set of vertices (nodes), and E denotes the set of edges symbolizing the available communication. For instance, if a node v is a physical neighbor of node u, then there exist (u, v) $\in$ E. Each node u requires a unique identifier, id(u) (commonly IP or MAC address). We define the neighborhood set N(u) of a node u as:

$$N(u) = \{v \in V \mid v \neq u \wedge (u,v) \in E\}$$

We adopt Wu and Li's pruning based CDS construction algorithm [12]. Initially all vertices are unmarked. They exchange their

neighborhood information with their one-hop neighbor. Hence, each node is aware of all its two-hop neighbors. The marking process employs the following simple rule: any vertex with two unconnected neighbors is marked as a dominator.

The set comprising marked vertices constitutes a connected dominating set, often containing redundant nodes. To refine the dominating set, two pruning principles are introduced, focusing on neighborhood subset coverage. The time complexity of this algorithm is O(Δ2), where Δ is the maximum degree. This algorithm has a linear performance ratio.

For an illustrative example, there are 8 distinct nodes as shown in Fig. 2. Since N (d) = {a, b, c, e, f} and N (b) = {c, d}, d is marked. In a similar way, node f is marked as dominant. Thus, there are two clusters; {a, b, c, d, e} and {f, g, h}. In addition, the dominating nodes (i.e., CHs) are {d, f}. For a practical application, the gateway transmits key information to nodes via d or f.

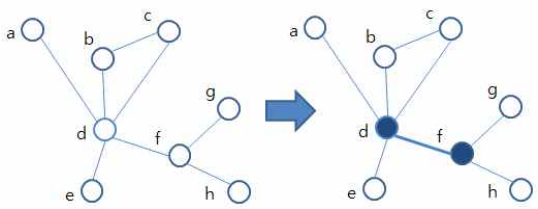

Fig. 2. A dominating set composed of nodes {d, f}.

One of the important issues are outside attacks, whereby user authentication is the first line defense against these attacks [13]. Another interesting thing is to restrict key distribution over a specific cluster to reduce communication overheads. That means keys will be sent to the nodes only which belong to the cluster. In WSN message broadcasting is a typical method for route discovery. This indicates that one host must transmit a specific message to all other nodes within the network. Through cluster formation, only cluster heads are designated as routers to relay the broadcast packet, ensuring broadcasting occurs by retransmitting through the fewest possible nodes. In addition, only the nodes belong to a specific cluster will receive key information during user authentication.

For this purpose, we employ a 3-way chaining user authentication scheme for HWSNs, illustrated in Fig. 3: (0) It is assumed that a gateway has m cluster heads authenticated during the deployment phase. (1) A user sends the message containing user's identifier, user's password and the cluster head to the gateway. Notice that the cluster head represents the cluster to which the node that the user wants to access belongs. (2) The gateway validates that user details and sends the same details to the cluster head. (3) If the details are legitimate, then that cluster head will reply to the user.

After the user can be recognized from the cluster head as a legitimate user, a secure session is established between the user and the cluster head. Through this secure session, the user can access data of the node belonging to the cluster head.

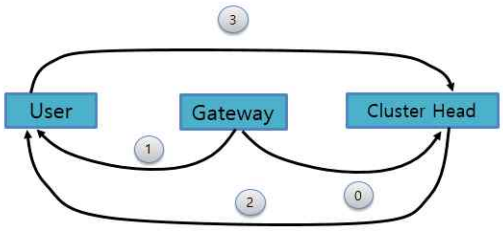

Fig. 3. A proposed user authentication scheme.

In this scheme, three distinct components — gateway, user, and cluster head— can authenticate each other mutually. Furthermore, our scheme employs the S/Key algorithm as a one-time password (OTP) for authentication [14]. At its core, our proposed algorithm establishes an OTP scheme between the user and the gateway. The password generated by the OTP remains valid for a single log-in session. With each new log-in session, the password table for a specific user needs to be updated based on their log-on data. Hence, our proposed scheme demonstrates resilience against replay attacks and stolen verifier scenarios. These processes solely rely on cryptographic one-way hash functions and XOR operations. The computation required for encryption and decryption using an OTP is regarded as relatively simple and can be accomplished using bit-level operations that are inherently supported by modern embedded processors.

3.2. Deployment phase

Before deploying cluster heads and sensor nodes into a target field, the gateway G assigns identifiers and master keys to the corresponding cluster heads and sensor nodes in offline. After complete that, the G knows all the CH_s and the nodes deployed in the field. The G assigns a unique identifier (IDCHj, IDSi) and a randomly generated unique master key MK_{CHj} for each cluster head CH_j. The MK_{CHj} of the CHj is known only to the G and to the CHj. Notice that each cluster head has the values of $\{ID_{CHj}, ID_{Si}\}$ into its own memory as shown in Fig. 4. Notice that X_S is the secret key known only to the G, whereas X_A is the one shared between the G and the user U_i.

During the deployment phase, the Si and the CHj in the field establish secure connection between them. We assume that each element locates his physical neighbor within the transmission range. For secure key establishments between them, W. Shen et al.'s key exchange protocol [15] is used. Their protocol adapts the well-known Diffie-Hellman based cryptographic protocols [16]. They have revised it to affect mutual authentication to protect the man-in-the-middle attack. Once the key establishment is completed, sensor nodes can engage in secure communication with other neighboring sensor nodes and their respective cluster head within the cluster. Cluster heads can also establish secure communication with neighboring cluster heads and ultimately with the gateway G.

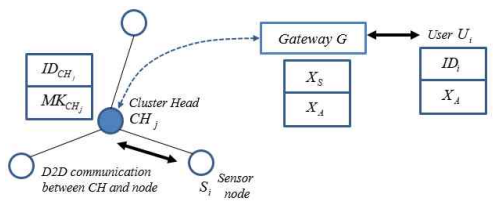

Fig. 4. The set-up results after the deployment phase.

3.3. User Authentication Scheme

Our scheme performs as follows: a user U_i who wants to access sensed data from a specific sensor node Si have to register to the WSN in a secure manner during the registration phase.

The user U_i must be authenticated from the gateway G at time T_1 and then from the cluster head CH_j at time T_2 ($>T_1$). After the U_i can be approved from the CH_j as an authorized user, a secure session is created between the U_i and the CH_j. Through this secure session, the U_i can request data of the node S_i from the CH_j. This scheme can provide mutual authentication among them, i.e., U_i, G and CH_j except S_i with small amount of transmission.

3.3.1. Registration Phase

The registration phase occurs when a user aims to securely register with the WSN. OTP, functioning as a dynamic password, remains valid for a single login session. Hence, this scheme is not vulnerable to replay attacks. Even if an attacker manages to acquire an OTP, they can only use it for a single time period; however, it becomes invalid once that period expires. In order to derive the subsequent password in the series from the preceding one, one would need to determine a method for computing the inverse of the hash function H. Indeed, given that H was deliberately selected to be a one-way function, calculating its inverse presents an exceedingly challenging task. We used SFHA-256 as the one-way hash function which has a better performance than the SHA-256 [17].

The S/Key algorithm stands out as one of the simplest methods suitable for WSNs, primarily because it relies solely on a one-way hash function and executes XOR operations [18]. The S/Key algorithm comprises two distinct phases: the password generation phase and the authentication phase.

In the password generation phase, a user generates his/her OTPs by performing one-way hash function with his/her password (PW) or previous output of the hash function iteratively. Upon generating the n-sequenced OTPs, both the user's ID, IDi, and the n-th OTP Hn(PWi), the last generated password, are to be submitted to the gateway. Notice that the user discards the initial password PWi. If a user wants to change his/her password, then the user performs this process again. It's important to note that as the size of n-sequenced OTPs expands, it requires more resources and longer execution times. Therefore, determining the size of n-sequenced OTPs becomes a tradeoff between resource constraints and the robustness of user authentication.

When the user authentication scheme starts, the user U_i and the gateway G need to perform the following steps:

Step 1: The user U_i provides the identifier ID_i and the generated n-sequenced OTPs, $H^1(PW_i)$, $H^2(PW_i),\cdots,H^N(PW_i)$, to the gateway G via a secure channel and the G stores them to its memory.

Step 2: The G retrieves the masked password $RPW_i = H^{N-k}(PW_i)$ from its memory. When U_i want to login at the first

time, k is set to 0 and RPW_i becomes $H^N(PW_i)$. At the next login, k is increased by 1. That means $RPW_i=H^{N-1}(PW_i)$. Therefore, the OTP value is referred to in the reverse order of the generated sequence. Then it computes the tamper proof secret information $h(ID_i||X_s)\oplus h(PW_i||X_A)$. Note that the secret key X_S is only known to the G. The another secret key X_A is shared between U_i and the G.

Step 3: The G then selects all m deployed cluster heads in the network, $CH_1,CH_2,\cdots,CH_m$ which have been deployed during the deployment phase, and computes the m key-plus-id combinations $\{(K_j,ID_{CHj})|1\leq j\leq m\}$, where $K_j=E_{MKCHj}$ $(ID_i \,||\, ID_{CHj} \,||\, X_S)$ and E represents a symmetric key encryption algorithm.

The overview of the registration phase is depicted in Fig. 5.

3.3.2. Login Phase

If the registered user U_i wants to access real-time data from the WSN, the U_i needs to perform the following steps:

Step 1: The U_i sends both the identifier ID_i and the k-th one-time password $RPW_i=H^{N-k}(PW_i)$ to the gateway G.

Step 2: Notice that RPWiG is the masked password stored in the gateway's memory and RPWi is the one just sent from the U_i. The G verifies whether RPWiG = RPWi. If this verification fails, it indicates that U_i entered their password incorrectly, leading to the termination of the scheme. Otherwise, the following steps are executed.

Step 3: The G computes $\gamma_i = h(ID_i \,||\, X_S)$, $\delta_i = h(RPW_i^G \,||X_A)$, and $\alpha_i = \gamma_i \oplus \delta_i$. In addition, using the system's current timestamp T_1, the G computes $p_i = h(\delta i \,||T_1)$.

Step 4: The U_i selects a cluster head, CH_j, of the cluster to which the sensor node S_i belongs. Corresponding to the CH_j, the G selects the encrypted master key MK_{CHj} of CH_j, from its memory and computes an authenticated message $E_{Kj}(ID_i \,||\, ID_{CHj} \,||\, p_i \,||\, \alpha_i \,||\, T_1)$. Notice that E represents a symmetric

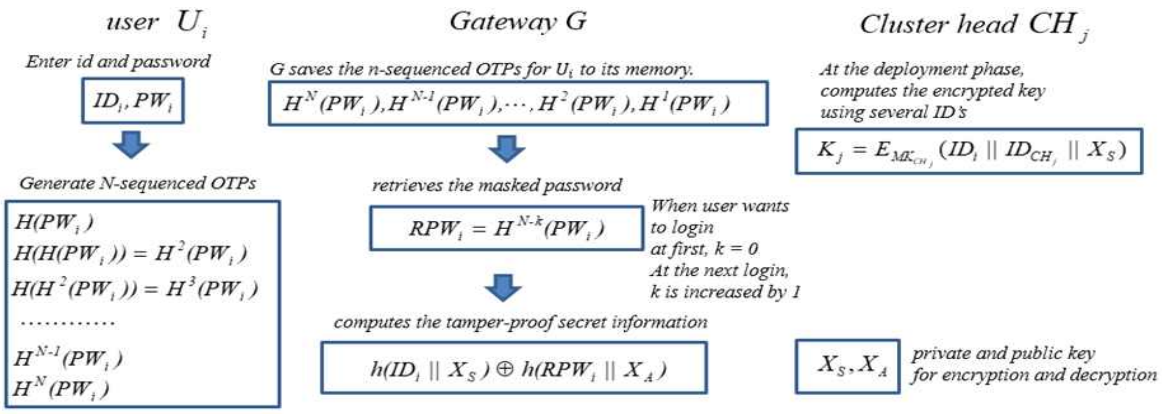

Fig. 5. the set-up results after registration phase

key encryption algorithm. Finally, the user U_i sends the message $<ID_i \| ID_{CHj} \| E_{Kj}(\cdots)>$ to the G, via a public channel.

Step 5: The current reference of the OTP should be changed to the new one, $RPW_i^G = H^{(N-(k+1))}(PW_i)$ for the purpose of next password verification.

3.3.3. User Authentication Against the Gateway

On receiving the message $<ID_i \| ID_{CHj} \| E_{Kj}(\cdots)>$ from the U_i, G performs following operations to authenticate the user U_i. The initial step involves checking whether the time taken to completely transmit the message is within an acceptable timeframe. If the message is received within a reasonable time, the message (IDi || IDCHj || pi || αi || T1) can be extracted from the encrypted message by decoding the request message using the computed key Kj = EMKCHj(IDi || IDCHj || XS). To verify the message, it computes a message authentication code (MAC) by using T1, IDi, and XS and then compares it with the MAC received.

Step 1: The G computes a key K_j using the stored master key E_{MKCHj} of the CH_j as $K_j = E_{MKCHj}(ID_i \| ID_{CHj} \| X_S)$. Using this K_j, the G decrypts $D_K[E_{Kj}(\cdots)]$ and thus, it retrieves the message, $(ID_i' \| ID_{CHj}' \| p_i \| \alpha_i \| T_1)$.

Step 2: The G checks if $ID_i' = ID_i$ and also if $ID'_{CHj} = ID_{CHj}$. If these holds, the G further checks if $|T_1 - T_1^*| < \Delta T_1$, where T_1^* represents the current system timestamp of G, and ΔT_1 denotes the expected time interval for the transmission delay. Now, if it holds, the G further computes $h((\alpha_i \oplus h(ID_i' \| X_S)) \| T_1)$. This equation becomes $h(\delta_i \oplus T_1)$ and it should be p_i. If the result equals to p_i, then the G accepts login U_i's request and U_i is considered as a valid user by the G. Otherwise, the scheme terminates.

Step 3: Utilizing the current system timestamp T_2, the G computes $q_i = h((\alpha_i \oplus h(ID_i \| X_S)) \| T_2)$ and generates a ciphertext message encrypted using the master key E_{MKCHj} of the CH_j as E_{MKCHj} $(ID_i \| ID_{CHj} \| q_i \| \alpha_i \| T_1 \| T_2)$. The G sends the message $<ID_i \| ID_{CHj} \| E_{MKCHj}(\cdots)>$ to the CH_j.

Step 4: After receiving the message in the Step 3 from the G, the cluster head CH_j decrypts $E_{MKCHj}(\cdots)$ using its own master key E_{MKCHj} as D_{MKCHj} $[E_{MKCHj}(\cdots)]$ and it produces the decrypted message $(ID_i' \| ID_{CHi}' \| q_i \| \alpha_i \| T_1 \| T_2)$. CH_j then checks if $ID_i' = ID_i$ and also if $ID_{CHj}' = ID_{CHj}$. If these holds, it further checks if $|T_2 - T_2^*| < \Delta T_2$, where T_2^* is the current time stamp of the and is the expected time interval for the transmission delay.

If it holds, the CH_j computes $h((\alpha_i \oplus h(ID_i \| X_S)) \| T_2)$. The CH_j then checks if it equals to q_i. If it does not hold, the scheme terminates. Otherwise, if it holds, the U_i is considered as a valid user and authenticated by the CH_j.

The overview of the user authentication phase is depicted in Fig. 6.

3.3.4. Session Establishment between Cluster Head and User

Because the user U_i has successfully passed

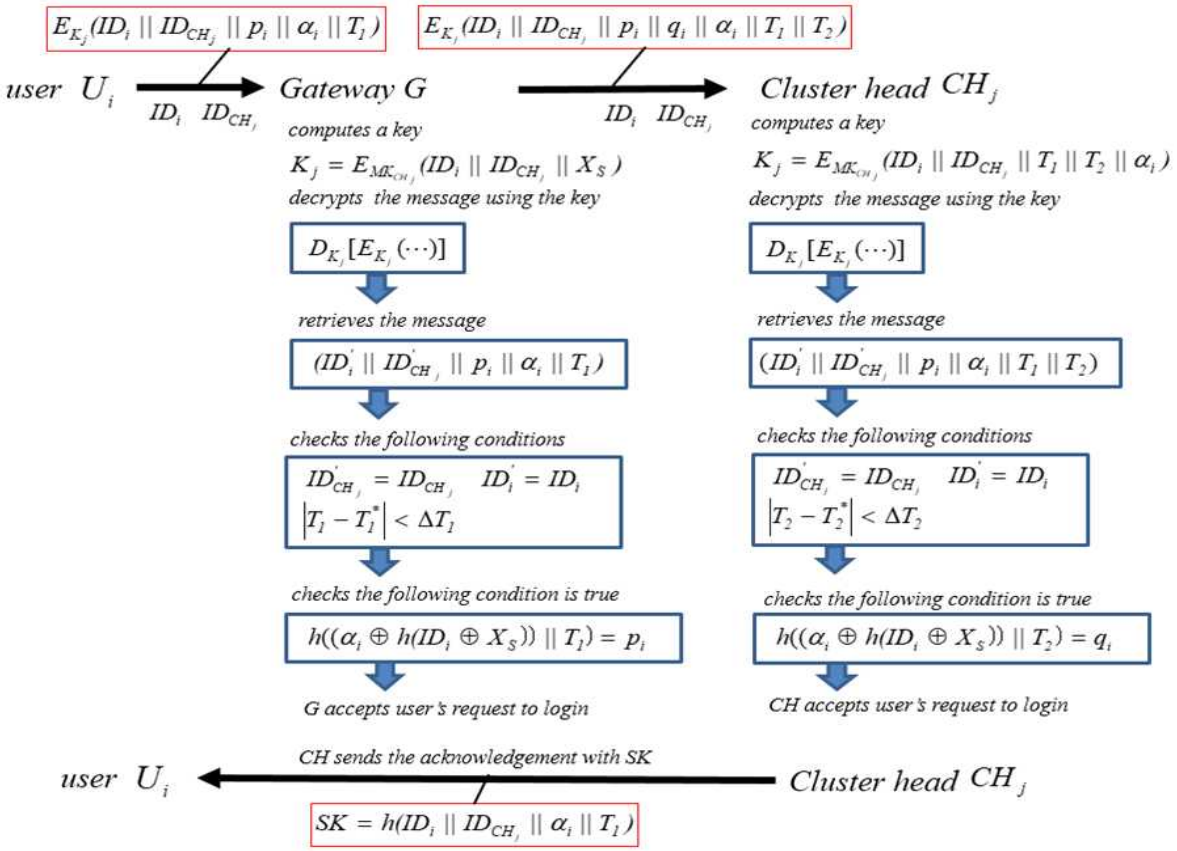

Fig. 6. the process of the user authentication scheme

the authentication phase, the CH_j can establish a session with the authenticated user U_i using a session key S_K. Then the U_i can communicate securely with the CH_j.

Step 1: The CH_j computes a secret session key, S_K, which is shared with the user U_i where $S_K = h(ID_i||ID_{CHj}||e_i||T_1)$. Then, CH_j transmit an acknowledgement to the user U_i through other cluster heads and G responding to U_i's query

Step 2: Upon receiving the acknowledgement from CH_j, U_i computes the same secret session key shared with CH_j using its previous system timestamp T_1, ID_i, ID_{CHj}, and α_i as $S_K = h(ID_i||ID_{CHj}||\alpha_i||T_1)$. Thus, both the U_i and the CH_j communicates securely in future using the derived secret session key S_K.

3.3.5. D2D Communication between Cluster Head and Node

Finally, the U_i can access real-time data from the specific sensor node S_i belonging to the cluster managed by the cluster head CH_j. The U_i transmits information about the node S_i to the CH_j through the secure session established before. The communication between

CH_j and S_i is a sort of D2D(Device-to-Device) communication. For secure key establishments between them, W. Shen et al.'s key exchange protocol [15] is used.

Let S_i be a sensor node belonging to the cluster head CH_j. At the same time, the S_i is the node that the user U_i wants to access. At the initial stage, the CH_j and the S_i choose their Diffie-Hellman parameter a and b to compute g^a and g^b, respectively. Notice that g is a generating element. The CH_j and the S_i respectively generate k-bit random strings N_{CHj} and $N_{Si.}$ $M_{CHj}=ID_{CHj}||g^a||N_{CHj}$ and $m_{Si}=ID_{Si}||g^b||N_{Si}$ are formed by concatenation, in which ID_{CHj} and ID_{Si}. The CH_j also needs to compute the commitment/opening (c, d) for $m_{CHj} = ID_{CHj} ||g^a || N_{CHj}$.

After the initial stage, the C_{Hj} and the S_i perform the following message exchange over their D2D communications channel. User CH_j sends the c, the commitment value m_{CHj} of to the S_i; after receiving c, the S_i sends m_{Si} to the CH_j. In return, the CH_j sends the decommit value d to the S_i. The S_i opens the commitment and gets $m_{CHj} = ID_{CHj} ||g^a || N_{CHj}$.

In the final stage, the CH_j and the S_i produce $N_{CHj}\oplus N_{Si}$ as the short authentication string for mutual authentication. That means, the CH_j and the S_i generate the k-bits string by $S_{CHj} = N_{CHj}\oplus N_{Si}'$ and $S_{Si} = N_{CHj}'\oplus N_{Si}$, in which N_{Si}' and N_{CHj}' are derived from messages received by the CH_j and the S_i.

Then the CH_j and the S_i verify if $S_{CHj}=S_{Si}$ via trusted channel. If the authentication strings match, the CH_j and the S_i accept each other's Diffie-Hellman parameters and compute the shared secret key $K=g^{ab}$ mod p. The rationale behind comparing the authentication strings before generating the Diffie-Hellman secret key is to allow CHj and Si to bypass the computation for secret key generation if the strings do not match.

4. Analysis of the Proposed Scheme

This section delves into the analysis of the security and efficiency of the proposed protocol.

4.1. Security Analysis

4.1.1. Guessing Attack

This kind of attack is very crucial for any ID/Password-based authentication systems. The proposed scheme uses OTP with the one-way hash function, SHA-512. The security is based on the strength of this hash function. Hence, guessing the initial password becomes exceedingly difficult.

4.1.2. Stolen-Verifier Attack

Our proposed protocol makes a change the password/verifier at the gateway whenever users' success their login process. The passwords used in this system are made of successive hash function which has irreversible property. Even if intruders can get the password database from the gateway, they are not able to masquerade the gateway. Also, they cannot obtain the use's initial password or next OTP value based on the stolen password database.

4.1.3. Playback Attack

To prevent the playback attack, the protocol should be very robust. In this protocol although an intruder can figure out the legitimate login information (ID/password), this information cannot be used to login to the system by the intruder. The gateway will modify login information when a valid user successfully login. The used OTP cannot authenticate the user any more. If the intruder gets a legitimate request information from the path between the gate way and sensor node, the playback attack can be protected by the time interval ΔT. Even if the attacker forgets the timestamp T_1, it remains ineffective due to the MAC. Altering the timestamp results in the sensor node calculating an updated MAC value, distinct from the one sent from the gateway. Similarly, the message from the sensor node to the user can withstand a playback attack.

4.1.4. Bypass Attack

If an intruder wants to obtain data from a sensor node, then he should send a data request message to that node. It, however, is not possible without knowing secret key, PRG. Therefore, bypass attack can be protected by this protocol.

4.1.5. Impersonation attack

Suppose an intruder obtains login information (ID, OTP) from the sensor network. The intruder, however, cannot impersonate valid user, because it is not feasible to calculate next OTP, H^{-1}(OTP) to login again. Moreover, the intruder cannot impersonate either the gateway or the sensor node without knowledge of the secret keys, PRG, PUG, and PRN.

4.1.6. Data integrity

The sensor nodes response for user's request of sensing data with MAC, which provides data integrity. If an intruder obtains this sensor's response message from the network and forge the sensing data, then MAC which is generated by the user will not be identical to the MAC which is sent from the sensor node. Therefore, this proposed protocol can provide integrity of sensing data.

4.2. Performance Analysis

In the Wireless Sensor Network, it is very essential to decrease the number of transmissions between sensors and gateway because number of transmissions may increase a chance of network congestion and energy consumption of a sensor node. The proposed scheme only needs three transmissions because we applied three-way chaining scheme.

This scheme can retrieve data from the network with reduced communication compared to other schemes. Most of other schemes need to communicate between user and gateway, and gateway and node to login process and authentication.

Our protocol provides mutual authentication at the all components, the gateway, user, and sensor node. Other schemes usually provide the mutual authentication between gateway and sensor only, or does not provide mutual authentication. Mutual authentication is very

impotent to reduce the risk of masquerade attack against each component.

The proposed protocol's use of three data transmissions, facilitated by chaining authentication, contrasts with existing methods requiring four transmissions for mutual authentication between gateway and user, and gateway and cluster head. This reduction by one transmission signifies a 25% decrease in transmission overhead, emphasizing the efficiency gains achieved by the proposed protocol. Although our protocol mandates encryption and decryption for mutual authentication, the computing resources required for encryption typically surpass those needed for hash or XOR operations. Thus, selecting an appropriate encryption algorithm during implementation becomes critical. Future research endeavors will focus on validating these aspects to further substantiate the protocol's efficacy and security measures.

5. Conclusion

In this paper, we introduced a hierarchical wireless sensor network authentication scheme employing a 3-way chaining method utilizing S/Key OTP. This scheme works as a ring structure and uses one-way communication. With this scheme three distinct components, gateway, user, and cluster head, can mutually authenticate each other. Additionally, we utilize the S/Key algorithm as a one-time password for authentication, built upon a secure and rapid hash algorithm. After secure session establishment between user and cluster head, a secret key agreement protocol between cluster head and the sensor node belonging to it is applied for secure communication.

The proposed protocol effectively mitigates replay attacks, scenarios involving multiple users logged in with the same login ID, stolen-verifier attacks, and various other threats. Moreover, the proposed protocol facilitates mutual authentication among all components with a minimal number of transmissions. We've demonstrated the efficiency of the proposed protocol through comparisons with related methodologies.

Acknowledgement

This work was supported by an Incheon National University Research Grant in 2020.

References

[1] K. Wong, Y. Zheng, J. Cao and S. Wang, “A Dynamic User Authentication Scheme for Wireless Sensor Networks”, Proc. of IEEE Int. Conf. on Sensor Networks, Ubiquitous, and Trustworthy Computing (SUTC'06), IEEE Computer Society, 2006, pp.244-251.

[2] H. R. Tseng, R. H. Jan and W. Yang, “An improved dynamic user authentication scheme for wireless sensor networks”, IEEE Globecom'07, 2007, pp.986-990.

[3] L. Hu, Y. Yang, and X. Niu, “Improved Remote User Authentication Scheme Preserving User Anonymity”, 5th Conf. on Communication Networks and Service Research, 2007.

[4] Manik Lal Das, “Two-Factor User Authentication in Wireless Sensor Networks”, IEEE Trans. On Wireless Communications, 2009, Vol.8, No.3, pp. 1086-1090.
[5] B. Vaidya, J. Rodrigues, and J. H. Park, “User authentication schemes with pseudonymity for ubiquitous sensor network in NGN”, Int. Journal of Communication Systems, 2010, Vol.23, No.9-10, pp.1201-1220.
[6] M. L. Das, “Two-Factor User Authentication in Wireless Sensor Networks”, IEEE Trans. On Wireless Communications, 2009, Vol.8, No.3, pp. 1086-1090.
[7] B. Vaidya, D. Makrakis, and H. T. Mouftah, “Improved two-factor user authentication in wireless sensor networks”, IEEE 6th Int. Conf. of Wireless and Mobile Computing, Networking and Communications (WiMob), 2010, pp.600-606.
[8] K. S. Arikumar and K. Thirumoorthy, “Improved User Authentication in Wireless Sensor Networks”, Proc. of ICETECT, 2011, pp.1010-1015.
[9] A. K. Das, P. Sharma, S. Chatterjee, and J. K. Sing, “A Dynamic Password-based User Authentication Scheme for Hierarchical Wireless Sensor Networks”, Journal of Network and Computer Applications, 2012, Vol. 35, pp. 1646-1656.
[10] D. Dolev and A. Yao, “On the security of public key protocols”, IEEE Trans. On Information Theory, 1983, Vol.29, No.2, pp. 198-208. G. O. Young, "Synthetic structure of industrial plastics (Book style with paper title and editor)" in Plastics, 2nd ed. vol. 3, J. Peters, Ed. New York: McGraw-Hill, 1964, pp. 15-64.
[11] M. Turkanovic and M. Holbl, “An Improved Dynamic Password-based User Authentication Scheme for Hierarchical Wireless Sensor Networks”, IEEE Global Communications Conference, 2014.
[12] J. Wu, F. Dai, M. Gao, and I. Stojmenovic, “On Calculating Power-Aware Connected Dominating Set for Efficient Routing in Ad Hoc Wireless Networks”, Journal of Communication and Networks, 2002, Vol. 5, No. 2, pp.169-178.
[13] R. Shaik and G. Swain, “User Authentication in Internet of Things: A survey”, International Journal of Pharmacy and Technology, 2016, Vol. 8, No.4, pp. 22036-22050.
[14] D.H. Kim, Y.S. Hong, and K.Y. Lee, “User Authentication with Distributed Cluster Formation in Wireless Sensor Networks”, Frontier and Innovation in Future Computing and Communications, springers. LNEE, 2014, Vol 301, pp-75-85.
[15] W. Shen, W. Hong, X. Cao, B. Yin, D. M. Shila and Y. Cheng, “Secure Key Establishment for Device-to-Device Communications”, IEEE Globecom, 2014.
[16] W. Diffie and M. E. Hellman, “New directions in cryptography”, IEEE Trans. Inform. Theory, 1976, vol. IT-22, pp. 644-654.
[17] Siddharth Agarwal, Abhinav Rungta, R.Padmavathy, Mayank Shankar and Nipun Rajan, “An Improved Fast and Secure Hash Algorithm”, Journal of Information Processing Systems, 2012, Vol. 8, No. 1, pp. 119~132.
[18] Haller, N., “The S/KEY One-Time Password System”, Proceedings of the ISOC Symposium on Network and Distributed System Security, San Diego, CA, 1994.

Authors

Dong-Hoon Kim

2022.2. Ph.D in Dept. of Information and Telecommunication Engineering from Incheon National University
2022.3-present A Postdoctoral researcher in Kwangwoon University
<Research interests> Radar signal processing, AI-based signal processing

Ki Young Lee

1993.12 Ph.D. in ECE from Univ. of Alabama, USA
1994.3-present Professor of Incheon National University (Dept of Info & Telecom Eng.)
<Research interests> Information Security System, User Authentication