Journal of Network and Computer Applications 36 (2013) 316–323

Contents lists available at SciVerse ScienceDirect

Journal of Network and Computer Applications

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

A temporal-credential-based mutual authentication and key agreement scheme for wireless sensor networks

Kaiping Xue *, Changsha Ma, Peilin Hong, Rong Ding

The Information Network Lab of EEIS Department, USTC, Hefei 230027, China

ARTICLE INFO

Article history:
Received 21 November 2011
Received in revised form
30 April 2012
Accepted 25 May 2012
Available online 7 June 2012

Keywords:
Temporal credential
Mutual authentication
Key agreement
Wireless sensor network
Gateway node

ABSTRACT

Wireless sensor network (WSN) can be deployed in any unattended environment. With the new developed IoT (Internet of Things) technology, remote authorized users are allowed to access reliable sensor nodes to obtain data and even are allowed to send commands to the nodes in the WSN. Because of the resource constrained nature of sensor nodes, it is important to design a secure, effective and lightweight authentication and key agreement scheme. The gateway node (GWN) plays a crucial role in the WSN as all data transmitted to the outside network must pass through it. We propose a temporal-credential-based mutual authentication scheme among the user, GWN and the sensor node. With the help of the password-based authentication, GWN can issue a temporal credential to each user and sensor node. For a user, his/her temporal credential can be securely protected and stored openly in a smart card. For a sensor node, its temporal credential is related to its identity and must privately stored in its storage medium. Furthermore, with the help of GWN, a lightweight key agreement scheme is proposed to embed into our protocol. The protocol only needs hash and XOR computations. The results of security and performance analysis demonstrate that the proposed scheme provides relatively more security features and high security level without increasing too much overhead of communication, computation and storage. It is realistic and well adapted for resource-constrained wireless sensor networks.

1. Introduction

Wireless sensor networks (WSNs) composing of a large number of sensor nodes can be deployed in any unattended environment, such as field observation, military battlefield and so on. In the past decade, WSNs have gained great achievements both in the academic circle and the industrial field. With the new developed IoT (Internet of Things) technology, remote authorized users are allowed to access reliable sensor nodes to obtain data and even are allowed to send commands to the nodes in WSN. Two aspects of this scene should be considered: On the one hand, only legitimate users can access specific sensor nodes to obtain data. On the other hand, the sensor node for access is required to be verified as a legitimate one. In order to ensure the above two points, mutual authentication between the user and the sensor node is required in the protocol design. Although existing schemes could provide perfect security transmission protocols in the link and network layers, how to design an efficient mutual authentication and key agreement scheme has not been well addressed. Because of the resource-constrained feature of sensor nodes in WSN, the mutual authentication and key agreement protocol requires to be lightweight on the premise of secure. Lightweight features can be reflected in the overhead of computation, communication and storage.

There are three main disadvantages of the traditional certificate-based authentication schemes in WSNs. Firstly, they all need a third party public key infrastructure, which is inconvenient to be deployed in WSN. Secondly, mutual authentication is based on asymmetric encryption, resulting in high computation cost of sensor nodes. Thirdly, the certificate validation operation is online. Although some improved schemes based on elliptic curve algorithm are proposed to reduce computational complexity, additional security infrastructures are still required. Online access to the CRL (Certificate Revocation List) for the certificate validity verification also brings a single point problem. Meanwhile, it is hard for the sensor node and the external infrastructure to set up an end-to-end communication path. Improved schemes based on offline CRL require each node to maintain a CRL list which is updated based on CRL broadcasting by a trusted third party. This is unrealistic for sensor nodes with limited storage and communication capacity. A large number of secret key sharing based schemes have difficulties in distributing and updating keys. To conclude, in order to achieve data accessing with authorization and security, designing mutual authentication and key agreement schemes is an important and difficult issue in WSNs.

Currently, some gateway node (GWN)-aid based authentication schemes are proposed, which make possible that the mutual authentication and key agreement protocol has both features of

* Corresponding author. Tel./fax: +86 551 360 1334.
E-mail address: kpxue@ustc.edu.cn (K. Xue).

1084-8045/$ - see front matter © 2012 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.jnca.2012.05.010

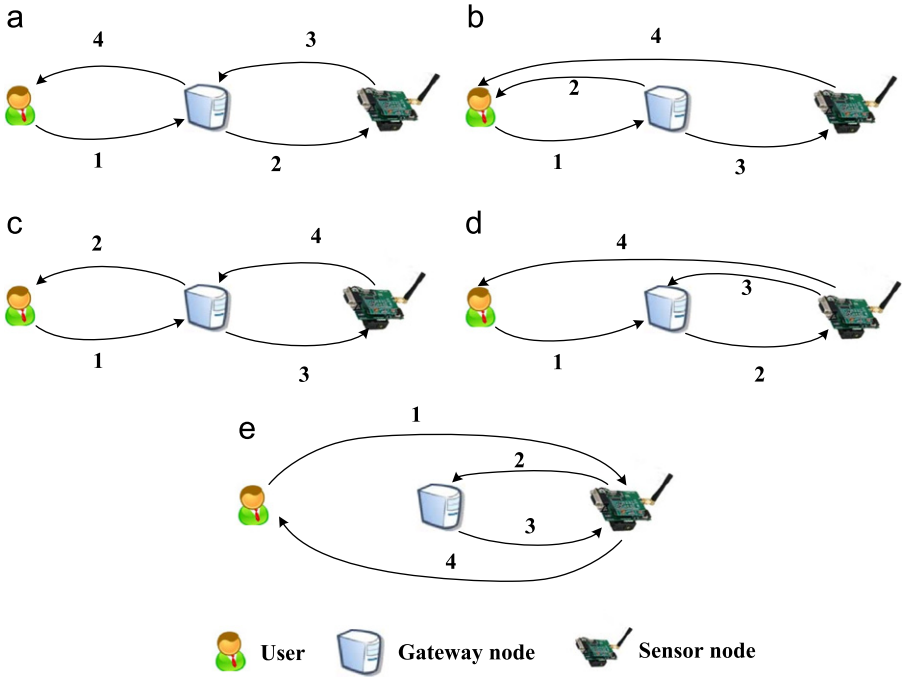

Fig. 1. Five basic authentication models for WSNs.

security and lightweight. GWN plays an important role in the network. In order to further reach the specific sensor node, remote users is required to reach GWN through Internet at first. Contrary, sensing data from the sensor nodes firstly gets to GWN, then further reaches the user end. If the data in the network is made available to the remote user on demand, mutual authentication between them must be ensured before allowing the remote user to access. With the aid of GWN, impenetrability of lightweight mutual authentication is going to be possible. Because of being usually deployed in harsh environments, authentication of the gateway node is also necessary for the user and the sensor. As in Fig. 1, there are five basic authentication models among the user, GWN, and the sensor node.

All of them need four messages to implement mutual authentication. Among them, Fig. 1(d) uses the recursive style to achieve mutual authentication among the user, GWN and the sensor node, in which Steps 3 and 4 can be processed in parallel. Most existing GWN-aid schemes are based on these five models. However, we find that most of these schemes have more or less security flaws. Because of unrealistic assumptions, some of them are not suitable for the resource-constrained wireless sensor networks. Wong et al. (2007) firstly proposed a hash based user authentication scheme, which is less complex, lightweight and dynamic. But some researchers found that it is vulnerable to stolen-verifier, replay, and forgery attacks. Das proposed a two factor method of user authentication (Das, 2009), which has become a frequently cited literature in this area of password based authentication (He et al., 2010; Khan and Alghathbar, 2010; Chen and Shih, 2010; Yeh et al., 2011; Xu et al., 2009; Song, 2010). Das's scheme, which implements password based authentication with the aid of GWN, is suitable for resource-constrained WSNs. In the beginning of Das's scheme, the user provides his/her user identity and password. Then after the verification, GWN issues a temporal credential for the user. When authenticating a user again, the GWN has no need to search the password with the key of its identity from the database. Meanwhile, GWN has no need to store any security information. Protocol processing only needs hash and XOR computations, with no additional symmetric encryption and asymmetric encryption. Das's scheme reduces the computational complexity, which applies to resource-constrained sensor nodes and user equipments. Unfortunately, this scheme has some security flaws and does not provide mutual authentication and key agreement. A series of schemes are subsequently put forward to improve it. He et al. (2010) proposed a similar protocol as in Das (2009). Although this scheme enhances password security, it did not essentially make up for the security flaws. In Khan and Alghathbar (2010), the authors presented several improvements. The first improvement is using hash value of the password instead of directly using the password. The improvement is reasonable, because in most password-based authentication systems, the hash value of the user password, rather than the plain password, is stored in the server. The second improvement is giving a password updating method. But in Khan and Alghathbar (2010), this improvement does not have practical significance, because the password has no specific signification in the initial verification process. The third one is providing mutual authentication among GWN and the sensor node based on the assumption of having a pre-shared key between GWN and each sensor node, which brings storage and lookup overhead to GWN. GWN needs to share a unique security key with each user and sensor node in Khan and Alghathbar (2010). Chen and Shih (2010) provides mutual authentications among the user, GWN, and the sensor node, but it is still vulnerable to replay, forgery and Bypassing attacks. Yeh et al. (2011) proposes an ECC-based user authentication and key agreement protocol. Xu et al. (2009) and Song (2010) both provide Deffie–Hellman key agreement (Diffie and Hellman, 1976) based mutual authentications between the user and the sensor node. Besides increasing computational complexity, Yeh et al. (2011), Xu et al. (2009), and Song (2010) also require additional storage overhead of public keys of other sensor nodes or users.

In this paper we propose an temporal-credential-based mutual authentication and key agreement scheme for WSNs. With the help of password-based authentication, GWN can issue a temporal credential to each user and sensor node. For a user, his/her temporal credential can be securely protected and stored openly in a smart card. For a sensor node, its temporal credential is related to its identity and must privately stored in its storage medium. Furthermore, based on using temporal credentials, the proposed protocol provides mutual authentication among the user, GWN, and the

sensor node. Key agreement is implemented between the user and the sensor node for secure remote access. The protocol only needs hash and XOR computations, which not only meets the security requirements, but also does not increase too much overhead. This is suitable for the scenario that a legitimate remote user obtains sensor data at any of the sensor nodes in WSNs.

The rest of this paper is organized as follows. In Section 2, we give our proposed temporal-credential-based mutual authentication and key agreement scheme for WSNs in detail. Then in Sections 3 and 4, we present security and performance analysis of the protocol. At last, in Section 5, we provide some concluding remarks.

2. Description of the proposed protocol

In this section, we will describe our proposed temporal-credential-based mutual authentication and key agreement scheme in detail. Before description, we firstly summarize the notations used throughout this paper in Table 1.

Our proposed protocol has three phase: (1) Registration phase; (2) Login phase; (3) Authentication and key agreement phase. We will introduce them as follows:

(1) *Registration phase*: This phase has two parts, respectively for users and for sensor nodes.

Assuming that each user has a secure password shared with GWN. The identity of the user and hash value of his/her password are stored on GWN's side. Also, each sensor node is also pre-configured a password, hash of which is stored in GWN's side.

Firstly, we give the details of this registration phase for users. In this phase, the user U_i submits his/her ID_i, timestamp value TS_1, and $H(TS_1\|H(PW_i))$ to GWN in an open and public environment. After the verification, the GWN issues a temporal credential to U_i. We illustrate the U_i's registration phase in Fig. 2, and describe the steps as follows:

Step U-1: $U_i \rightarrow GWN$: $\{ID_i, TS_1, VI_i\}$. The user gets his/her current timestamp value TS_1, and computes

$$VI_i = H(TS_1\|H(PW_i)) \quad (1)$$

Table 1
Notations in this paper (sorted alphabetically).

Symbol	Definition
U_i	A user
C_i, C_{GWN}, C_j	Conform information computed by U_i, GWN, S_j
DID_i, DID_{GWN}	Dynamic identity of U_i, GWN
$H(\cdot)$	The hash operation
ID_i	U_i's identity
K_{GWN-U}, K_{GWN-S}	Private security parameters only known to GWN
K_i, K_j	Key sharing randomly chosen by U_i, S_j
KEY_{ij}	Shard session key between U_i and S_j
GWN	The gateway node of the wireless sensor network
PW_i	The password of the user U_i
PKS_i, PKS_{GWN}, PKS_j	Security information to protect key sharing computed by U_i, GWN, S_j
P_i	Protected pseudonym of U_i
PTC_i	Protected temporal credential of U_i stored in the smart card
REG_j	Registration information of S_j
S_j	A sensor node
SID_j	S_j's identity
TC_i, TC_j	A temporal credential issued by GWN to U_i or S_j
TE_i	The expiration time of a user's temporal credential
TS	The timestamp value
VI_i	Verification information of U_i
"$\|$"	The bitwise concatenation operation
"$\oplus$"	The bitwise XOR operation

Then, U_i submits TS_1, VI_i and his/her ID_i to GWN in an open and public environment.

Step U-2: **GWN** $\rightarrow U_i$: (*Smart Card*) $\{H(\cdot), ID_i, H(H(PW_i)), TE_i, TC_i\}$. After receiving the message, GWN checks whether the transmission delay is within the allowed time interval ΔT. Here, ΔT is an empirical value in our scheme. If the time synchronization in the WSN can accurate to several minutes (<5 min), we can set $delta_t = 10$ min. We assume the current time is T^*_{GWN}. If $T^*_{GWN} - TS_1 > \Delta T$, GWN stops here and sends REJ message back to U_i. Else, GWN continues to take out its own copy of $H(PW_i)$ by using the key "ID_i", then GWN computes $VI^*_i = H(TS_1\|H(PW_i))$ and verifies whether $VI^*_i = VI_i$. If not, GWN stops here; otherwise, GWN further computes P_i, TC_i and PTC_i as follows:

$$P_i = H(ID_i\|TE_i)$$

$$TC_i = H(K_{GWN-U}\|P_i\|TE_i)$$

$$PTC_i = TC_i \oplus H(PW_i) \quad (2)$$

where TE_i is the expiration time of the temporal credential set by GWN or the trust third party (TTP). K_{GWN-U} is the GWN's private key only known to GWN. TC_i is the temporal credential for U_i issued by GWN. At last, GWN personalizes the smart card for U_i with the parameters: $\{H(\cdot), ID_i, H(H(PW_i)), TE_i, PTC_i\}$.[1]

Secondly, we give the details of the registration phase for sensor nodes. Before the deployment, each sensor node S_j is configured with its identity SID_j and its random password PW_j. After the deployment, the sensor node S_j submits its SID_j, timestamp value TS_2, and $H(TS_2\|H(PW_j))$ to the GWN in an open and public environment. After the verification of S_j's legitimacy, GWN issues a temporal credential TC_j to S_j. We illustrate the S_j's registration phase in Fig. 3, and describe the steps as follows:

Step S-1: $S_j \rightarrow$ **GWN**: $\{SID_j, TS_2, VI_j\}$. The sensor node S_j gets its current timestamp value TS_2, and computes

$$VI_j = H(TS_2\|H(PW_j)) \quad (3)$$

Then, S_j submits its SID_j, TS_2, and VI_j to GWN in an open and public environment.

Step S-2: **GWN** $\rightarrow S_j$: $\{TS_3, REG_j\}$. After receiving the message, GWN checks whether the transmission delay is within the allowed time interval ΔT. We assume the current time is T^*_{GWN}. If $T^*_{GWN} - TS_2 > \Delta T$, GWN stops here and sends REJ message back to S_j. Else, GWN continues to take out its own copy of $H(PW_j)$ by using the key "SID_j". Then GWN computes $VI^*_j = H(TS_2\|H(PW_j))$ and verifies whether $VI^*_j = VI_j$. If not, GWN stops here; otherwise, GWN further computes as follows:

$$TC_j = H(K_{GWN-S}\|SID_j)$$

$$REG_j = H(H(PW_j)\|TS_3) \oplus TC_j \quad (4)$$

where TS_3 is the timestamp value, K_{GWN-S} is the GWN's private key known only to GWN. TC_j is the temporal credential for S_j issued by GWN. Then GWN sends TS_3 and REG_j to the sensor node S_j.

Step S-3: After receiving the message, S_j checks whether the transmission delay is within the allowed time interval ΔT. We assume the current time is T^*_j. If $T^*_j - TS_3 > \Delta T$, S_j

[1] Here, a smart card is not essential. As an alternative, *GWN* can send $\{H(\cdot), ID_i, H(H(PW_i)), TE_i, PTC_i\}$ to U_i over an open and public environment. Then U_i stores these message as a file, which is functionally similar to a smart card.

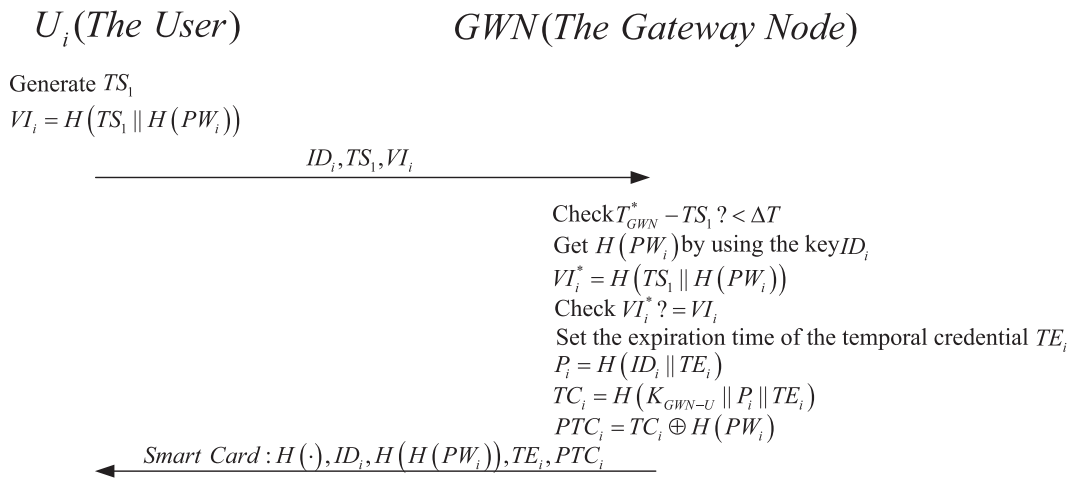

Fig. 2. Illustration of the registration phase of the user U_i.

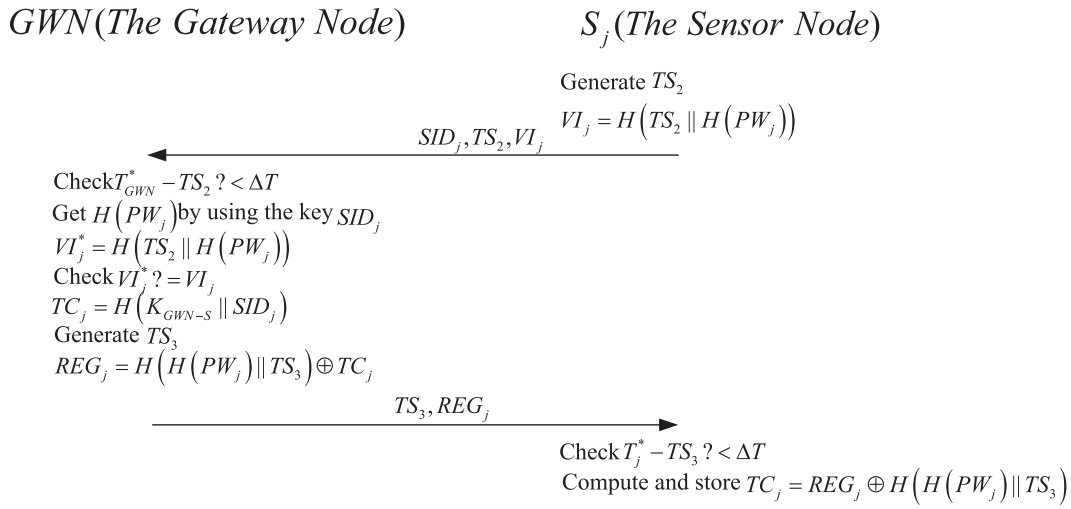

Fig. 3. Illustration of the registration phase of the sensor node S_j.

stops here. Otherwise, S_j continues to compute its temporal credential $TC_j = REG_j \oplus H(H(PW_j)\|TS_3)$ and stores it.

(2) *Login phase*: U_i inserts his/her smart card to a terminal, and enters his/her ID_i and PW_i. The terminal validates ID_i and PW_i with the stored ID_i and $H(H(PW_i))$ in the smart card. If the entered ID_i and PW_i are not matching, the smart card terminates the login request. Otherwise, U_i passes the verification and can read the information stored in the smart card. After the verification passing, U_i compute to get TC_i as follow:

$$TC_i = PTC_i \oplus H(PW_i) \tag{5}$$

(3) *Authentication and key agreement phase*: This phase achieves the goal of mutual authentication among the user, GWN and the sensor node, which consists of three parts. Meanwhile, a session key is negotiated between the user and the sensor node. Following the authentication model Model (d) described in Fig. 1, there are three steps. The first step is the user's legitimacy verification by GWN. The second step is GWN's legitimacy verification by the sensor node. The third step is legitimacy verification of the sensor node by the user and GWN. Besides the mutual authentication, the data communication between the user and the sensor node needs protection by encryption and MAC (Message Authentication Code). This requires both parties are able to securely negotiate the session key, which is not well addressed in the previous related works. Here we give a simple key agreement method which only uses hash function and XOR operation. The security of key agreement depends on the credibility of GWN, so legitimacy verification of GWN is also important for the user and the sensor node. We illustrate the authentication and key agreement phase in Fig. 4, and describe the steps as follows:

Step 1: $U_i \rightarrow$ **GWN**: {DID_i, C_i, PKS_i, TS_4, TE_i, P_i}. U_i generate a time-stamp value TS_4 and randomly chooses a key sharing K_i. Then U_i computes as follows:

$$DID_i = ID_i \oplus H(TC_i\|TS_4)$$

$$C_i = H(H(ID_i\|TS_4) \oplus TC_i)$$

$$PKS_i = K_i \oplus H(TC_i\|TS_4\|\text{“000”}) \tag{6}$$

where, the use of the binary number “000” is in order to securely distinguish $H(TC_i\|TS_4\|\text{“000”})$ and $H(TC_i\|TS_4)$, which is ensured by the feature of hash function.

Finally, U_i sends DID_i, C_i, PKS_i, TS_4, TE_i, and P_i to GWN.

Step 2: **GWN** $\rightarrow S_j$: {TS_5, DID_i, DID_{GWN}, C_{GWN}, PKS_{GWN}}. After receiving the message, GWN checks whether the transmission

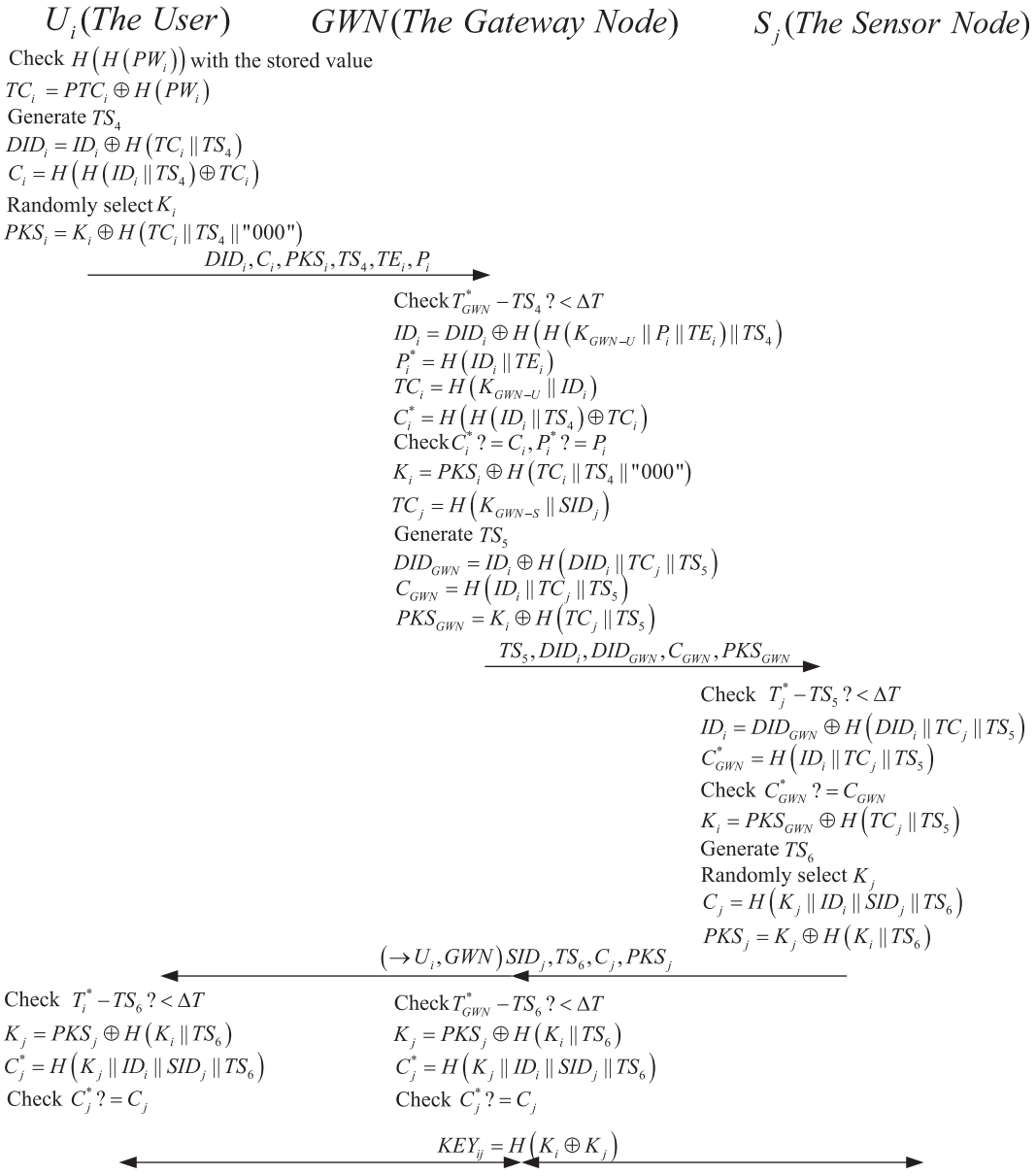

Fig. 4. Illustration of the login phase, the authentication and key agreement phase.

delay is within the allowed time interval ΔT. Assume the current time is T^*_{GWN}. If $T^*_{GWN} - TS_4 > \Delta T$, GWN stops here and sends REJ message back to U_i. Else, GWN compute as follows:

$$ID_i = DID_i \oplus H(H(K_{GWN-U} \| P_i \| TE_i) \| TS_4)$$

$$P_i^* = H(ID_i \| TE_i)$$

$$TC_i = H(K_{GWN-U} \| P_i^* \| TE_i)$$

$$C_i^* = H(H(ID_i^* \| TS_4) \oplus TC_i^*) \quad (7)$$

If $C_i^* \neq C_i$ or $P_i^* \neq P_i$, GWN rejects it and sends REJ message back to U_i. Else, GWN accepts U_i's login request and computes:

$$K_i = PKS_i \oplus H(TC_i \| TS_4 \| \text{“000”}) \quad (8)$$

Then GWN chooses a nearby suitable sensor node as the accessed sensor node (S_j, with the identity SID_j). GWN can compute S_j's temporal credential $TC_j (= H(K_{GWN-S} \| SID_j))$. After that, GWN computes DID_{GWN}, C_{GWN} and PKS_{ij} as follows:

$$DID_{GWN} = ID_i \oplus H(DID_i \| TC_j \| TS_5)$$

$$C_{GWN} = H(ID_i \| TC_j \| TS_5)$$

$$PKS_{GWN} = K_i \oplus H(TC_j \| TS_5) \quad (9)$$

where TS_5 is a timestamp value. Finally, GWN sends TS_5, DID_i, DID_{GWN}, C_{GWN} and PKS_{GWN} to S_j.

Step 3: $S_j \rightarrow U_i$, **GWN**: $\{SID_j, TS_6, C_j, PKS_j\}$. After receiving the message, the sensor node checks whether the transmission delay is within the allowed time interval ΔT. We assume the current time is T_j^*. If $T_j^* - TS_5 > \Delta T$, the sensor node rejects and stops here. Else, S_j computes as follows:

$$ID_i = DID_{GWN} \oplus H(DID_i \| TC_j \| TS_5)$$

$$C^*_{GWN} = H(ID_i \| TC_j \| TS_5) \quad (10)$$

If $C^*_{GWN} \neq C_{GWN}$, S_j rejects and stops. Else, S_j confirms the received message is from a legitimate GWN, and computes:

$$K_i = PKS_{GWN} \oplus H(TC_j \| TS_5) \quad (11)$$

Then S_j generates a timestamp value TS_6 and a randomly chosen key sharing K_j. Then S_j computes C_j and PKS_j as follows:

$$C_j = H(K_j \| ID_i \| SID_j \| TS_6)$$

$$PKS_j = K_j \oplus H(K_i \| TS_6) \quad (12)$$

Finally, S_j sends SID_j, TS_6, C_j and PKS_j to U_i and GWN.

Step 4: After receiving the message and passing the timeliness verification of TS_6, U_i and GWN can separately compute K_j and C^*_j as follows:

$$K_j = PKS_j \oplus H(K_i \| TS_6)$$

$$C^*_j = H(K_j \| ID_i \| SID_j \| TS_6) \quad (13)$$

For GWN, if $C^*_j = C_j$, it can confirm that S_j is a legitimate sensor node. For the user U_i, if $C^*_j = C_j$, he/she can confirm that S_j and GWN are both legitimate.

U_i and S_j can separately compute the shared session key KEY_{ij} as follow:

$$KEY_{ij} = H(K_i \oplus K_j) \quad (14)$$

The security of KEY_{ij} depends on the credibility of GWN, so GWN's legitimacy must be verified by U_i and S_j. Further, we can deserve the encryption key and the integrity protection key from KEY_{ij}, which is mentioned in IETF RFC (2008).

3. Security analysis of our protocol

In this section, we summarize security analysis of our proposed protocol and compare its security properties with some related schemes. Our proposed scheme has the following main security features:

A. *Mutual authentication*: Based on temporal credential signing and issuing, our proposed scheme provides mutual authentication among the user, GWN and the sensor node. The mutual authentication between the user and GWN is accomplished by the openly and securely verification of the capability of calculating temporal credential for the user. From Step 2 in the authentication and key agreement phase described in Section 2, GWN can verify the legitimacy of the user. From Step 4 in the authentication and key agreement phase, by checking whether $C_j = C^*_j$, the user can verify the legitimacy of the sensor node and GWN. This is because only legitimate GWN can compute the identity and the key sharing of the user and relay them to the sensor node. The mutual authentication between GWN and the sensor node is accomplished by the openly and securely verification of the capability of calculating temporal credential for the sensor node. From Step 3 in the authentication and key agreement phase, the sensor node can verify the legitimacy of GWN based on checking whether $C_{GWN} = C^*_{GWN}$. Also from Step 4 in the authentication and key agreement phase, based on checking whether $C_j = C^*_j$, GWN can verify the legitimacy of the sensor node.

B. *Resiliency of insider attacks and masquerade attacks*: Because of using one way hash function and P_i, the inside malicious user U_f cannot get any knowledge of another user U_i's password and other security information. Also because of using P_i, U_f cannot use his/her own security information to masquerade as U_i.

C. *Password protection*: Many users usually set semantic-similar but different passwords in different systems. Using the hash value of the user password can break the semantic relevance and prevent leakage of security information. Also in our proposed protocol, attackers cannot get K_{GWN-U} and K_{GWN-S}, which are only used to compute temporal credentials for the user and the sensor node.

D. *Password updating/changing*: After password based verification, the password of each user does not appear in input parameters of temporal credential computation. TE_i is the expiration time of TC_i, which determines TC_i's updating time. Password updating/changing can happen in anytime, which will only affect for re-issuing temporal credential in the next time. Before issuing a new temporal credential, GWN needs to verify the legitimacy based on the updated password. However in order to insure having no need of designing temporal credential revoking mechanism, we set TE_i to be close to the issuing time. For the particularity of the WSN application, its no need for the sensor node to update/change its password after the registration.

E. *Identity protection*: Using of DID_i and DID_{GWN} ensure that only legitimate GWN and the sensor node who have TC_j can compute the right ID_i. This can prevent the leakage of private user identity to malicious attackers.

F. *Key agreement*: In certain wireless sensor network environments, encryption and MAC are required to protect the data communication between the user and the sensor node. Both parties need to negotiate the session key in advance. Some related works provide key agreement schemes base on asymmetric encryption system such as ECC and Deffie–Hellman key exchange. In this paper, we only use hash function and XOR operation to design a simple key agreement scheme.

G. *Resiliency of stolen smart card attacks*: We assume that a smart card is stolen or lost, physical protection method cannot prevent the malicious attacker to get the stored security information $\{H(\cdot), ID_i, H(H(PW_i)), TE_i, PTC_i\}$. But without inputting correct password PW_i, the attacker cannot get right TC_i. After that he/she cannot provide right DID_i and C_i to GWN. Therefore our proposed protocol can resist to stolen smart card attacks.

H. *Resiliency of GWN bypassing attacks*: It is difficult to compute DID_{GWN} and C_{GWN} without TC_j in the user side, so he/she cannot bypass GWN to forge a verification message straightly to the sensor node S_j. Without right message from GWN, the sensor node cannot respond any other faked messages.

I. *Resiliency of replay attacks*: We introduce timestamp value in our protocol to address the replay attacks.

Security properties of the proposed protocol, compared with related works are summarized in Table 2.

4. Performance analysis of our protocol

The protocol implementation delay is mainly caused by the authentication and key agreement phase. Main differences in the protocol design compared with related works are also shown in this phase. Therefore, we mainly discuss the cost of authentication and key agreement process of our proposed scheme and related works in this section, including computation overhead, communication overhead and storage overhead.

In brief, compared with the related works, while providing relatively more security features and high security level, our proposed protocol does not increase too much overhead.

Table 2
Security comparison of our proposed scheme and related schemes.

Items	Ours	Wong et al. (2007)	Das (2009)	He et al. (2010)	Khan and Alghathbar (2010)	Chen and Shih (2010)	Yeh et al. (2011)	Xu et al. (2009)	Song (2010)
Mutual authentication between each two of the user, GWN and the sensor node	Y	N	N	N	Y	Y	Y	Y	Y
Resiliency of insider attacks	Y	Y	N	N	Y	N	Y	N	N
Resiliency of masquerade attack	Y	Y	N	N	Y	N	Y	N	N
Password protection	Y	N	N	Y	Y	N	Y	N	Y
password updating/changing	Y[a]	N	N	Y	Y	N	N	Y	Y
Identity protection	Y	N	Y	Y	Y	Y	N	N	N
Key agreement	Y	N	N	N	N	N	Y	Y	Y
Resiliency of stolen smart card attacks	Y	N	N	N	N	N	N	N	Y
Resiliency of GWN bypassing attacks	Y	N	N	N	N	N	Y	/	/
Resiliency of replay attacks	Y	N	Y	Y	Y	Y	N	Y	Y

Y: Yes; N: No.

[a] Users' and sensor nodes' passwords are not involved in computation of temporal credentials, so the function of temporal credential updating/changing is independent of password updating/changing in our scheme.

4.1. Computation overhead analysis

First we defined two computational parameters as follows.

- T_H denotes the time for the hash operation.
- T_{ECC} denotes the time for the ECC-160 encryption/decryption operation.

For example in the environment (CPU: 3.2 GHz, RAM: 3.0 G), we have run 100 times to get the average result. T_{ECC} is about 19 times faster than T_H (T_{ECC} is nearly 0.45 ms on average when using ECC-160. T_H is nearly 0.02 ms on average when using SHA-1).

Table 3 gives computation overhead comparison of five protocols, in which our protocol and Khan and Alghathbar (2010), Chen and Shih (2010), and Yeh et al. (2011) provide the function of mutual authentication among the user, GWN and the sensor node. Meanwhile, our protocol and Yeh et al. (2011) both provide the key agreement process. Our proposed protocol requires $7T_H$ for the user (in which $3T_H$ is related to the key agreement process), $10T_H$ for GWN (in which $2T_H$ is related to the key agreement process) and $5T_H$ for the sensor node (in which $2T_H$ is related to the key agreement process). Compared with Khan and Alghathbar (2010) and Chen and Shih (2010) which only provide simple mutual authentication process, our protocol does not increase too much computational complexity while providing more security features. Yeh et al. (2011) provides the key agreement process based on ECC encryption algorithm, which has large computational complexity than hash functions.

Assume T_H=0.45 according to our test. Compared with Khan and Alghathbar (2010) and Chen and Shih (2010), every sensor node in the authentication and key agreement phase of our proposed scheme require about additional 1.35 ms, which can be almost ignored. Similar conclusion also applies to each users and GWN who implement the authentication and key agreement phase.

4.2. Communication overhead analysis

For mutual authentication, our protocol and (Khan and Alghathbar, 2010; Chen and Shih, 2010; Yeh et al., 2011) all require four times of message transmission according to different models described in Fig. 1. (Khan and Alghathbar (2010) uses Model (a). Chen and Shih (2010) uses Model (c). Yeh et al. (2011) uses Model (e). Our protocol uses Model (d).) In Model (d) Steps 3 and 4 can be done in parallel at the same time, which can cut down some transmission delay. Assuming the length of the identity of a sensor node is 128-bit, the length of each hash value is also 128-bit, the length of the timestamp value is 24-bit. The message send by each sensor node is 51-byte. compared with it, the messages send by each sensor node in Khan and Alghathbar (2010), Chen and Shih (2010), Xu et al. (2009) and Song (2010) separately 19-byte, 16-byte, 35-byte and 51-byte. Because of providing more security, the message transmission overhead is increased accordingly, but the message send to GWN by each sensor node (this message is also relayed to the accessing user) is only a little longer than that in relate works. Nevertheless, this is acceptable in the system design for sensor node in the WSN.

Table 3
Computation overhead comparison of our protocol and some related works.

Protocols	User	GWN	Sensor node
Ours	$4T_H+3T_H$	$8T_H+2T_H$	$3T_H+2T_H$
Das (2009)	$3T_H$	$4T_H$	T_H
Khan and Alghathbar (2010)	$3T_H$	$5T_H$	$2T_H$
Chen and Shih (2010)	$4T_H$	$5T_H$	$2T_H$
Yeh et al. (2011)	T_H+2T_{ECC}	$4T_H+4T_{ECC}$	$3T_H+2T_{ECC}$

Here, T_H denotes the time for the hash operation; T_{ECC} denotes the time for the encryption/decryption operation in ECC-160 algorithm.

4.3. Storage overhead analysis

Storage overhead cannot be ignored in protocol design, especially for resource limited WSNs. In order to enhance security, Khan and Alghathbar (2010) requires GWN to share a unique security key with each user and each sensor node. All of these security keys must be stored in GWN's storage medium. During the implementation of the protocol, the lookup operation to search specific key is required by GWN. Chen and Shih (2010) have no additional storage overhead but has some security flaws. In Yeh et al. (2011), GWN needs to store and maintain public keys of all users. In our scheme, the user needs to store the parameters ($H(\cdot)$, ID_i, $H(H(PW_i))$, TE_i, PTC_i) in the smart card. These parameters can be stored in an open and public environment. *GWN* needs to keep K_{GWN-U} and K_{GWN-S}. The identities and hash values of passwords of users and sensor nodes can be stored *GWN* or the other TTP (Trust Third Party). Each sensor node needs to store its identity and its temporal credential. After the registration phase, its password can be removed from the memory.

5. Conclusions

In this paper, we propose a lightweight temporal-credential-based mutual authentication and key agreement scheme for

wireless sensor networks. In comparison with existing schemes, our proposed protocol not only provides relatively more security features and high security level, but also has low costs of communication, computation and storage. This scheme is suitable for the scenario that the legitimate user is allowed to access sensor data in any specific sensor node in the environment of resource-constrained wireless sensor network. The temporal credential issued by GWN can be further modified to be issued by a trust third party. This can further cut down the load of GWN. In addition, there is only a centralized GWN in the network, which could be performance and security bottleneck, our scheme can be improved by design a decentralized GWNs.

Acknowledgments

The authors sincerely thank the anonymous reviewers for their valuable comments that have led to the present improved version of the original manuscript. This work is supported by the National Natural Science Foundation of China under Grant no. 60903216, the National S&T Major Project of China under Grant nos. 2010ZX03003-002 and 2011ZX03005-006.

References

Chen TH, Shih WK. A robust mutual authentication protocol for wireless sensor networks. ETRI Journal 2010;32(5):704–12.

Das ML. Two-factor user authentication in wireless sensor networks. IEEE Transactions on Wireless Communications 2009;8(3):1086–90.

Diffie W, Hellman ME. New direction in cryptography. IEEE Transaction on Information Theory 1976;22(6):644–54.

He D, Gao Y, Chan S, Chen C, Bu J. An enhanced two-factor user authentication scheme in wireless sensor networks. Ad Hoc & Sensor Wireless Network 2010;10(4):361–71.

IETF RFC, RFC5246, The Transport Layer Security (TLS) Protocol, August 2008.

Khan MK, Alghathbar K. Cryptanalysis and security improvements of 'two-factor user authentication in wireless sensor networks'. Sensors 2010(10):2450–9.

Song R. Advanced smart card based password authentication protocol. Computer Standards & Interfaces 2010(32):321–5.

Wong KHM, Zheng Y, Cao J, Wang S. A dynamic user authentication scheme for wireless sensor networks. In: Proceedings of the IEEE international conference on sensor networks, ubiquitous, and trustworthy computing. Taichung, Taiwan; 5-7 June, 2007. p. 32–58.

Xu J, Zhu W, Feng D. An improved smart card based password authentication scheme with provable security. Computer Standards & Interfaces 2009;31(4):723–8.

Yeh HL, Chen TH, Liu PC, Kim TH, Wei HW. A secured authentication protocol for wireless sensor networks using elliptic curves cryptography. Sensors 2011(11):4767–79.