A Strong Authentication Scheme with User Privacy for Wireless Sensor Networks

Pardeep Kumar, Andrei Gurtov, Mika Ylianttila, Sang-Gon Lee, and HoonJae Lee

Wireless sensor networks (WSNs) are used for many real-time applications. User authentication is an important security service for WSNs to ensure only legitimate users can access the sensor data within the network. In 2012, Yoo and others proposed a security-performance-balanced user authentication scheme for WSNs, which is an enhancement of existing schemes. In this paper, we show that Yoo and others' scheme has security flaws, and it is not efficient for real WSNs. In addition, this paper proposes a new strong authentication scheme with user privacy for WSNs. The proposed scheme not only achieves end-party mutual authentication (that is, between the user and the sensor node) but also establishes a dynamic session key. The proposed scheme preserves the security features of Yoo and others' scheme and other existing schemes and provides more practical security services. Additionally, the efficiency of the proposed scheme is more appropriate for real-world WSNs applications.

Keywords: Wireless sensor network, authentication, confidentiality, user privacy, session-key establishment.

Manuscript received Jan. 28, 2013; revised Apr. 29, 2013; accepted May 6, 2013.

The research conducted in this paper was funded by the Finnish funding agency for technology and innovation (Tekes) Massive Scale Machine-to-Machine Service (MAMMotH) project (Dnro 820/31/2011). This paper was in part supported by Academy of Finland project SEMOHealth. The fourth author was funded by 2012 Dongseo University Research Fund.

Pardeep Kumar (phone: +358 4 6548 8911, pradeepkhl@gmail.com) and Mika Ylianttila (mika.ylianttila@ee.oulu.fi) are with the Centre for Wireless Communication, University of Oulu, Oulu, Finland.

Andrei Gurtov (gurtov@hiit.fi) is with Helsinki Institute for Information Technology (HIIT), Helsinki, Finland.

Sang-Gon Lee (nok60@dongse.ac.kr) and HoonJae Lee (hjlee@dongseo.ac.kr) are with the Division of Computer & Information Engineering, Dongseo University, Busan, Rep. of Korea.

http://dx.doi.org/10.4218/etrij.13.0113.0103

I. Introduction

In recent years, wireless sensor networks (WSNs) have grown rapidly, accommodating plenty of application areas. The applications include smart buildings, hospital monitoring, volcano and forest monitoring, chemical plant and hydro plant monitoring, healthcare monitoring, and so on [1]. These WSNs consist of resource-hungry sensors (for example, MICAz [2], Telos [3], and so on) that have low computational power, low battery power, low bandwidth, and a small amount of memory. The primary goal of the real-world WSN is to glean the environment data, process and store it, and forward it to the user, either on demand or upon detection of an event. Thus, the novelty of wireless sensors is their ubiquitous nature that makes sensor data available to users anytime, accommodating users' needs and demands. The data collected in most of the real-world applications (for example, for hospitals, smart buildings, and so on) is vulnerable. Moreover, the broadcast nature of the sensor node makes it possible for a user to access sensor data within the networks, that is, on demand [1], [4]; later he/she can misuse the sensitive sensor data (here, sensitive data depends on the applications, for example, healthcare) for his/her personal reasons. Since, the sensor data is made available to the user on demand within the WSN, it is mandatory to authenticate the users before allowing access to the WSN sensitive data. Therefore, user authentication is the prime concern in real WSNs.

Moreover, due to the vulnerability of wireless communications, individuals have raised concern over the violation of privacy. Thus, user privacy is also a current important topic of research.

Indeed, until now, significant user authentication protocols have been proposed for resource constraint WSNs [1], [4]-[17],

and each protocol has its merits and demerits. In 2009, Das proposed a two-factor user authentication protocol, which is based on a password and smartcard [4]. Das demonstrated that his protocol is a safeguard to many real-time attacks. Unfortunately, in 2010, Chen and Shih showed that Das' protocol fails in mutual authentication and is susceptible to parallel-session attacks [5]. In addition, Chen and Shih proposed a robust mutual authentication protocol for WSNs in [5] and claimed that their protocol provides more security than the method proposed in [4]. In the same year (2010), He and others demonstrated that Das' protocol is susceptible to insider attacks and impersonation attacks, and that there is no provision for users to change their passwords [6]. To cope with these security pitfalls, He and others proposed an enhanced two-factor protocol [6]. Furthermore, Khan and Alghathbar also found that Das' protocol has severe security pitfalls, such as being susceptible to gateway-bypass attacks and insider attacks, and that there is no mutual authentication between the sensor and the gateway [7]. Thus, Khan and Alghathbar proposed an enhancement to Das' scheme to overcome such problems [7].

However, in 2011, Yoon and Yoo [18] demonstrated that Chen and Shih's scheme has five major vulnerabilities: user impersonation, gateway impersonation, sensor node impersonation, privileged insider, and a time synchronization problem for large-scale networks. In [19], the authors pointed out that [6] and [7] are vulnerable to information-leakage attack, fail to preserve the privacy of the user, have no mutual authentication, and are lacking in session key establishment. Very recently, Yoo and others [8] showed that the schemes presented in [5], [7], [9], and [10] are vulnerable to parallel-session attacks and gateway-bypass attacks and are lacking in mutual authentication. In addition, they proposed a security-performance-balanced user authentication scheme to overcome the security problems of the schemes presented in [4], [5], [7], [9], and [10].

Nevertheless, based on our analysis, Yoo and others' protocol is vulnerable to impersonation attacks, message-alteration attacks, and man-in-the-middle attacks, and it is not efficient (in terms of computation and communication cost) for real-time applications. Therefore, this paper proposes a strong user authentication scheme that protects the privacy of the user, wherein each user must prove their legitimacy using a password and a smartcard. The proposed scheme facilitates many security services, including mutual authentication for all entities (that is, sensor, gateway, and user), protection of user's privacy, maintaining confidentiality of wireless messages, and dynamic session key establishment. In addition, users are allowed to change their password at any time. Further, we show that the proposed scheme is strong against popular attacks, in contrast to those proposed in [5]-[10], and attains high efficiency at reasonable computation and communication costs.

The rest of the paper is structured as follows. Section II presents a review and the weaknesses of Yoo and others' protocol. We propose a strong authentication scheme for WSNs in section III. The security and performance analysis of our proposed scheme is discussed in section IV. Finally, in section V, conclusions are drawn.

II. Analysis of Yoo and Others' Protocol

1. Review of Yoo and Others' Protocol

This section presents a review of Yoo and others' [8] user authentication protocol based on the use of a smartcard and a password. The protocol is divided into three phases: registration, authentication, and password-change.

A. Registration Phase

To access the sensor network data, each user must register with the gateway (GW) node. User (U_k) passes their ID_k and $PPW_k=h(PW_k)\oplus b$ to the GW node using a secure channel. Here, b is a user secret number. Upon receiving the ID_k and PPW_k, the GW performs the following computations: $M_k=h(ID_k||PPW_k)$, $N_k=h(ID_k||PPW_k)\oplus h(K||J)$, and $L_k=h(J||ID_k)$. Here, K is a symmetric key only known by the GW node, and J is a secret number that is generated by the GW node.

Thereafter, the GW node personalizes a smartcard to the user that contains the following parameters: $\{M_k, N_k, L_k, h(.)\}$. Then, U_k stores b in the smartcard so that the user does not need to memorize b. Thus, the smartcard contains $\{M_k, N_k, L_k, h(.), b\}$. Meanwhile, the GW node stores a unique secret key (that is, $Z_n=h(J||Sn)$) in each designated sensor node before the network deployment. Here, Sn is the identity of a sensor node.

B. Authentication Phase

The flow of the authentication phase is shown in Fig. 1. The authentication phase includes the login phase, the verification phase, and session key establishment.

Login phase. This phase is invoked when a user wants to access the sensor data on demand. User (U_k) inserts the smartcard into the terminal and enters keys (that is, ID_k and PW_k). After receiving the login request, the smartcard performs the following computations.

•YL-1: Computes $PPW_k=h(PW_k)\oplus b$ and $M_k*=h(ID_k||PPW_k)$ and compares $M_k*=M_k$; if yes, then local verification is done; otherwise, not.

•YL-2: Computes $DID_k= h(ID_k||PPW_k)\oplus h(L_k||T)$.

•YL-3: Transmits $\{DID_k, T, ID_k, RN1\}$ to the GW node; here,

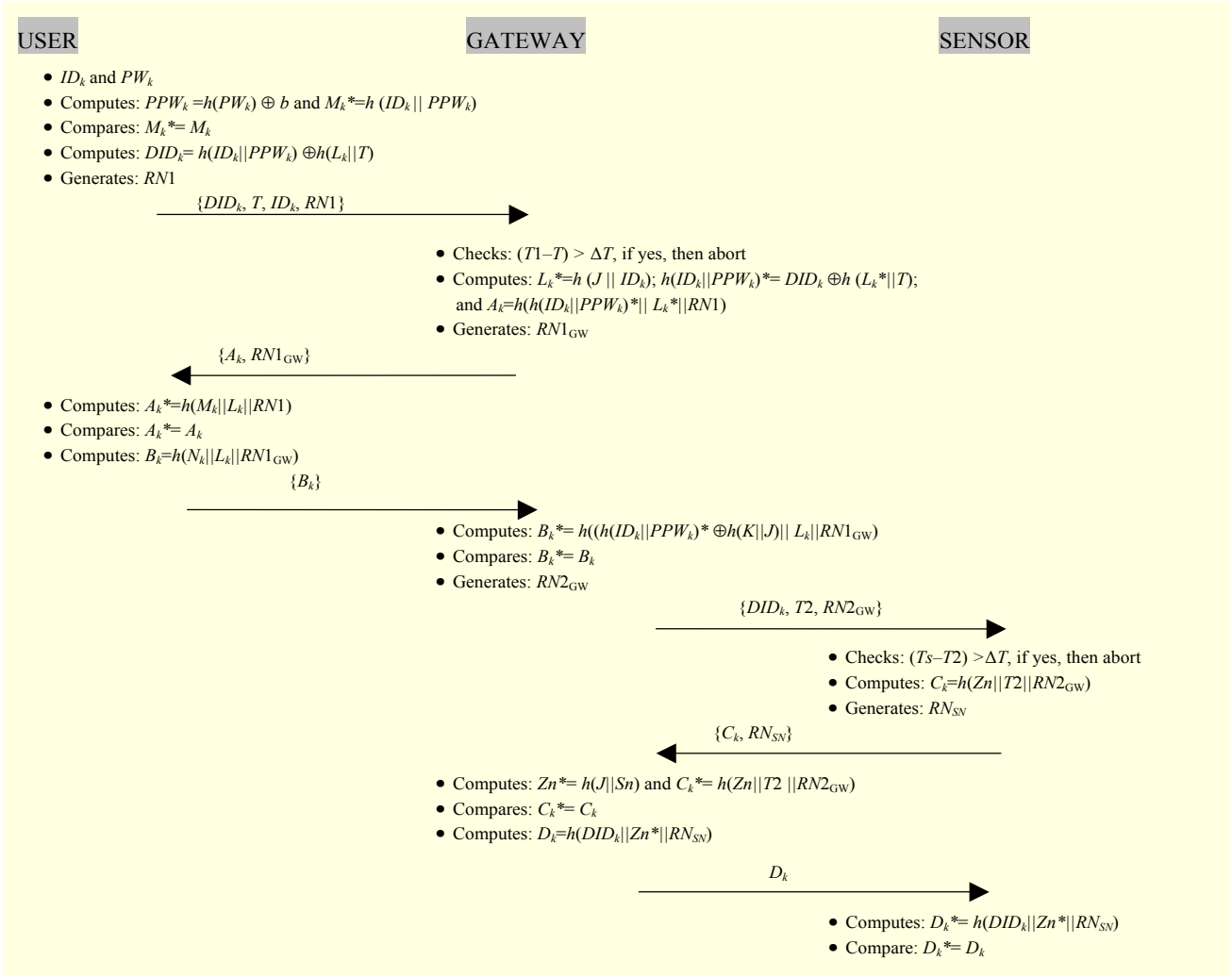


Fig. 1. Yoo and others' authentication phase.

T is the current timestamp of the U_k system, and $RN1$ is a random nonce generated by U_k.

Verification Phase. This phase consists of the following computations.

•YV-1: Upon receiving login request at time $T1$, the GW node validates T as follows: If $(T1 - T) > \Delta T$, terminates the request; otherwise, proceeds to the next step. Here, ΔT is the maximum communication delay allowed, and $T1$ is the current timestamp of the GW node.

•YV-2: Computes $L_k^* = h(J||ID_k)$; $h(ID_k||PPW_k)^* = DID_k \oplus h(L_k^*||T)$ and $A_k = h(h(ID_k||PPW_k)^*||L_k^*||RN1)$. Generates a random nonce, $RN1_{GW}$, and transmits the message $\{A_k, RN1_{GW}\}$ to U_k.

•YV-3: Upon receiving message from the GW node, U_k computes $A_k^* = h(M_k||L_k||RN1)$ and verifies whether it is equal to A_k. If not, terminates the system. Otherwise, U_k computes $B_k = h(N_k || L_k||RN1_{GW})$ and transmits B_k to the GW node.

•YV-4: GW node then computes $B_k^* = h((h(ID_k||PPW_k)^* \oplus h(K||J)||L_k||RN1_{GW})$ and checks whether it is equal to B_k. If yes, it means U_k is an authentic user and proceeds to the next step; otherwise, terminates the system.

•YV-5: GW node generates a random nonce, $RN2_{GW}$, and sends the message $\{DID_k, T2, RN2_{GW}\}$ to the nearest sensor nodes Sn, which U_k is looking for; here, $T2$ is the current timestamp of the GW node.

•YV-6: Upon receiving the GW node message, Sn first validates $T2$ as follows: $(Ts - T2) > \Delta T$; if yes, terminates the system; otherwise, proceeds to the next step. Here, Ts is the current timestamp of Sn. Computes $C_k = h(Zn||T2||RN2_{GW})$ and transmits the message $\{C_k, RN_{Sn}\}$ to GW node; here RN_{Sn} is a random nonce generated by Sn.

•YV-7: Now, the GW node computes $Zn^* = h(J||Sn)$ and $C_k^* = h(Zn||T2 ||RN2_{GW})$ and checks whether $C_k^* = C_k$. If yes, then generates a message, $D_k = h(DID_k||Zn^*||RN_{Sn})$, and sends to Sn.

•YV-8: Finally, Sn computes $D_k^* = h(DID_k||Zn^*||RN_{Sn})$ and

checks $D_k^*=D_k$; if yes, U_k is allowed to access *Sn* data; otherwise, not.

Session Key Establishment. A session key between U_k and GW node may be established as: $GWK_{Uk\text{-}GW}=h(RN1||RN1_{GW}||L_k)$, and between the GW node and *Sn* may be established as: $SnK_{Sn\text{-}GW} = h(RN_{Sn}||RN2_{GW}||Zn)$. Moreover, if a direct communication is required between the user and the *Sn*, then bilateral session key ($Skey_{Uk\text{-}Sn}$) can be computed through the GW node, as follows: GW node generates a random number $Skey_{Uk\text{-}Sn}$ and sends $RN1||Skey_{Uk\text{-}Sn}$ encrypted with L_k (shared between U_k and GW node) to U_k, and sends $RN_{Sn}||Skey_{Uk\text{-}Sn}$ encrypted with *Zn* (shared between the sensor node and the GW node) to *Sn*.

C. Password-Change Phase

•YPP-1: At the terminal, enters ID_k and PW_k and then enters new password, $NewPW_k$. The smartcard then performs $PPW_k=h(PW_k) \oplus b$ and $M_k^*=h(ID_k||PPW_k)$ and compares $M_k^*=M_k$; if not, then request will be rejected; otherwise, proceeds to the next steps.

•YPP-2: The smartcard performs: $NewM_k=h(ID_k||PPW_k)$, $h(K||J)=N_k\oplus h(ID_k||PPW_k)$, $NewN_k=h(ID_k||NewPPW_k)\oplus h(K||J)$, where $NewPPW_k=h(NewPW_k)\oplus b$. Finally, the smartcard replaces M_k with $NewM_k$ and N_k with $NewN_k$.

2. Weaknesses of Yoo and Others' Protocol

To analyze Yoo and others' protocol security, assume that an adversary (*Tom*) has full control over the wireless communication between the user, the GW, and the sensor node. He/she can eavesdrop on, alter, and intercept the wireless messages at any time. Under these assumptions, this subsection discusses the serious security flaws of Yoo and others' protocol.

1) Assume that *Tom* has intercepted one of the previous login messages (DID_k, T, ID_k, $RN1$) of a legal user (*Allen*) between the user and the GW node. Now, without knowing the password and identity of *Allen*, *Tom* can easily impersonate *Allen* to log into the GW node at time T^* ($>T$) [20]. The details of the impersonation attack are as follows.

Step 1: *Tom* → GW node: $\{DID_k, T^*, ID_k, RN1\}$. Here, T^* is the current timestamp of the system of *Tom*.

Step 2: Upon receiving the login request, the GW node checks the timestamp as $(T1-T^*) < \Delta T$. Since T^* is valid, the GW node will proceed to compute $L_k^*=h(J||ID_k)$, $h(ID_k||PPW_k)^*=DID_k\oplus h(L_k^*||T^*)$, and $A_k=h(h(ID_k||PPW_k)^*||L_k^*||RN1)$. Now, the GW node responds to *Tom* with $\{A_k, RN1_{GW}\}$.

By performing the above attack, the GW node accepts the login request of *Tom*, and *Tom* can simply imitate any user to log into the GW node at any time.

Moreover, in Yoo and others' protocol, since the user's identity and random number are floating as plain text, *Tom* can easily alter the previous login message (that is, ID_k with ID_{Tom} and $RN1$ with RN_{Tom},) and impersonate a legal user at T^* ($>T$), as follows.

Step 1: *Tom* → GW node: $\{DID_k, T^*, ID_{Tom}, RN_{Tom}\}$. Here, T^* is the current timestamp of the system of *Tom*.

Step 2: Since T^* is valid, the GW node will proceed to compute $L_{Tom}^*=h(J||ID_{Tom})$, $h(ID_k||PPW_k)^*=DID_k\oplus h(L_{Tom}^*||T^*)$, and $A_{Tom}=h(h(ID_k||PPW_k)^*||L_{Tom}^*||RN_{Tom})$. Now, the GW node responds to *Tom* with $\{A_{Tom}, RN1_{GW}\}$. Thus, Yoo and others' protocol is vulnerable to login message alteration attacks [12].

2) Yoo and others' protocol is vulnerable to man-in-the-middle (MITM) attacks. Indeed, avoiding an MITM attack in wireless communication is a difficult task, but an authentication protocol itself should detect the impact of an MITM attack early.

To generalize the MITM attack, it is assumed that *Tom* is active between the GW and the sensor node. In YV-5, *Tom* can capture the GW node message $\{DID_k, T2, RN2_{GW}\}$ at T^* ($>T2$) and easily alter the request to be $\{DID_k, T^*, RN_{Tom}\}$ by dropping the original $T2$ and $RN2_{GW}$. Here, T^* and RN_{Tom} are the current timestamp of the system of *Tom* and the random nonce, respectively. Thereafter, *Tom* forwards the altered request (that is, DID_k, T^*, RN_{Tom}) to the nearest sensor node. Upon receiving the request from *Tom*, the *Sn* performs the following.

Step 1: *Sn* validates the timestamp as $(Ts - T^*)<\Delta T$. Since T^* is valid, *Sn* will proceed to compute $C_k=h(Zn||T^*||RN_{Tom})$ and generates RN_{Sn}. It should be noted that, here, *Sn* does not know whether the request (that is, $\{DID_k, T^*, RN_{Tom}\}$) is coming from the legal GW node or the imposter (*Tom*). Thereafter, *Sn* sends the message to *Tom*, as described in Step 2.

Step 2: *Sn* → *Tom*: $\{C_{Tom}, RN_{Sn}\}$. Upon receiving the message from *Sn*, *Tom* can alter C_k with $C_{Tom}= h(DID_k, Zn_{Tom}, RN_{Tom})$. Here, Zn_{Tom} is a fake key generated by *Tom*. Then, *Tom* sends the altered message to the *Sn*, as described in Step 3.

Step 3: *Tom* → *Sn*: $\{C_{Tom}\}$. Obviously, the request from *Tom* (C_{Tom}) will be rejected by *Sn* due to $C_{Tom} \neq C_k$ (that is, $h(DID_k, Zn_{Tom}, RN_{Tom}) \neq h(DID_k||Zn||RN_{Sn})$, wherein *Zn* is a secret number, which is shared between the legal sensor and the GW node).

Consequently, with the very late detection of the MITM (*Tom*), Yoo and others' scheme is directly vulnerable to the denial of service attack on the sensor node. By generalizing the MITM attack, an adversary can easily make the sensor node run out of energy from attempting verification.

3) In mobile environments, the leakage of a user's identity may facilitate an unauthorized entity to track the user's current

location [21]. Thus, user privacy is one of the requirements of authentication protocols. However, in YL-3 of the authentication phase of Yoo and others' scheme, user identity (ID_k) is transmitted as plain text to the GW over the insecure public wireless channels for the login request (that is, $\{DID_k, T, ID_k, RN1\}$). Clearly, Yoo and others' scheme does not protect the privacy of the user.

4) In practice, mutual authentication must be performed between the two end parties (that is, user and sensor) to establish trust. In [8], the authors did not consider the need for mutual authentication. Hence, Yoo and others' scheme lacks a proper authentication procedure.

Indeed, Yoo and others proposed a robust user authentication scheme to overcome the weaknesses of [4], [5], [7], [9], [10]; however, our analysis reveals that their scheme has security flaws that may have major impacts on real-time WSN applications. Therefore, the problem of user authentication in WSNs remains unresolved. To remedy the problems in [5]-[10], a strong authentication scheme that provides the necessary security services to WSN applications at reasonable computation and communication cost is proposed in the next section.

III. Proposed User Authentication Scheme

Consider a wireless heterogeneous sensor network that consists of two types of devices, for example, low-resource devices (TelosB, MICAz) and high-resource devices (Stargate [22]), also known as the GW. The high-resource devices are tamper-resistant, but the low-resource devices are vulnerable to tampering. These devices are appropriately distributed in a confined area. We refer to [23]-[27] for a more comprehensive description of real-world WSNs. A user can access (on demand) the sensor data within the network using their personal digital assistant, mobile phone, or laptop. To query the sensor data, a user must register with the GW node and get a smartcard. Upon registration, he/she can query the sensor data within the network in a secure manner.

To execute the proposed scheme, we make the following assumptions.

i) A GW is a trusted party and never compromised.

ii) The GW and sensor share a long-term secret key $LT_{key}=h(GW_{id}||Sn_{id}||h(Y))$ using a key management scheme [28], [29]. Here, $h(.)$ is a one-way hash function, and Y is the high entropy secret number of a GW.

iii) The long-term key has a lifetime (for example, one year) that depends on the applications.

iv) All entities (user, GW, and sensor) strong symmetric cryptosystem that are sufficiently similar to one another (for example, RC5/Skipjack encryption and decryption algorithm, as suggested in [30], especially for WSNs).

Table 1. Notations and descriptions.

Notation	Description
ID_k, PW_k	Identity and password of user k (that is, U_k)
GW_{id}, Sn_{id}	Identity of GW and sensor node
J and X	GW secret numbers (that is, 256 bits)
b	User random number
$E_x[]$, $D_x[]$	Symmetric encryption and decryption using key x
$h(.)$	One-way hash function
$\|\|$, $\oplus$	Concatenation, XOR operation

The proposed scheme is divided into three phases: registration phase, authentication phase, and password-change phase. The notations and descriptions used in this paper are shown in Table 1.

1. Registration Phase

In this phase, a user passes his/her ID_k and $h(b\oplus PW_k)$ to the GW. Upon receiving the user registration request, the GW node performs the following steps: $A_k= E_j[ID_k||GW_{id}||h(X)]$ and $B_k =h(ID_k||h(b\oplus PW_k)\oplus A_k)$. Note, GW node secret numbers (that is, J and X) have specific lifetimes (for example, one year), and all users must reregister with the GW node after the secret numbers expire.

At this point, the GW node issues a smartcard to the user with following parameters: $\{A_k, B_k, h(.), h(X)\}$. Upon receiving the smartcard, the user enters a random number b into the smartcard; by doing so, the user does not need to memorize the random number b. Now, the smartcard has the following: A_k, B_k, $h(.)$, $h(X)$, and b.

2. Authentication Phase

This phase is divided into two subphases: login phase and verification phase. The flow of authentication phase is shown in Fig. 2.

Login phase. The login phase is invoked whenever a user wants to access the sensor data. U_k inserts the smartcard into the terminal and enters ID_k and PW_k. Upon receiving the login request, the smartcard performs the following steps.

•L-1: Computes $B_k^*=h(ID_k||h(b\oplus PW_k)\oplus A_k)$ and verifies $B_k^*=B_k$; if not, terminates the login request; otherwise, proceeds to the next step.

•L-2: Generates a temporary key ($M=h(h(X)||ID_k||T')$) and nonce C_k. Encrypts $h(ID_k)||h(X)||C_k||T'$ using M (that is, $P_k=$

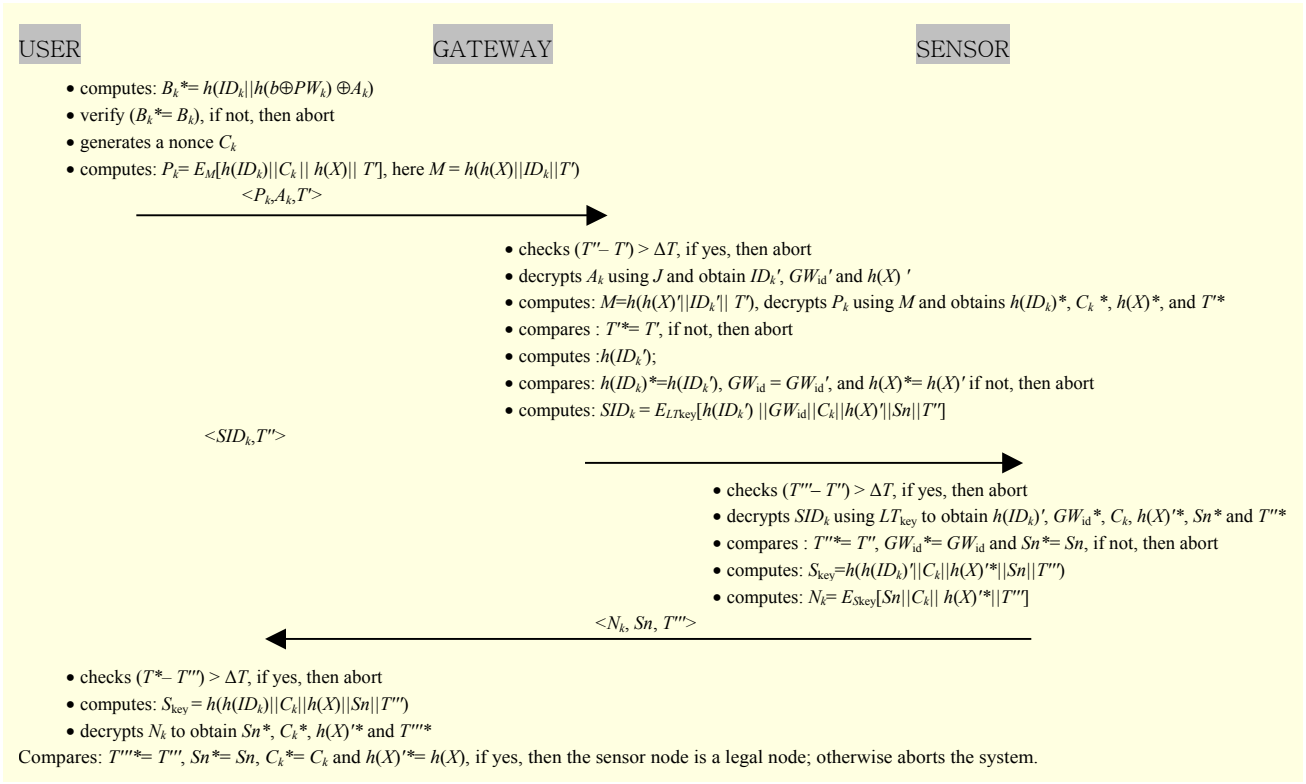


Fig. 2. Proposed scheme: authentication phase.

$E_M[h(ID_k)||h(X)||C_k||T'])$ and transmits the message $<P_k, A_k, T'>$ to the GW node over an insecure public network; here, T' is the current timestamp of the system of the user.

Verification phase. This phase is invoked whenever the GW receives a login request $<P_k, A_k, T'>$ from the user at time T''. The GW node authenticates user U_k according to the following steps.

•V-1: Checks timestamp $(T''-T') > \Delta T$; if yes, terminates the system; otherwise, proceeds to the next step. Here, T'' is the current timestamp of the GW node and ΔT is an expected time interval for the message transmission delay.

•V-2: Decrypts A_k using the GW node secret J and obtains ID_k', GW_{id}', and $h(X)'$.

•V-3: Computes $M=h(h(X)||ID_k'||T')$ and decrypts P_k using M to obtain $h(ID_k)^*$, $h(X)^*$, C_k, and T'^*.

•V-4: Compares $T'^*=T'$; if not, terminates the system; otherwise, proceeds to the next step.

•V-5: Computes $h(ID_k')$ and compares $h(ID_k)^*=h(ID_k')$, $GW_{id}=GW_{id}'$, and $h(X)^*=h(X)'$. If the sensor node holds, then the GW node ensures that the user U_k is a legitimate user and proceeds to the next step; otherwise, terminates the system.

•V-6: Computes $SID_k=E_{LT_{key}}[h(ID_k')||GW_{id}||C_k||h(X)'||Sn||T'']$. Here, T'' is the current timestamp of the GW node. The GW node sends a message $<SID_k, T''>$ to nearest Sn.

•V-7: Upon receiving the GW message $<SID_k, T''>$, the sensor node validates the timestamp as follows: $(T'''-T'') > \Delta T$; if yes, terminates the system; otherwise, proceeds to the next step. Here, ΔT is considered to be an expected time interval for the message transmission delay, and T''' is the current timestamp of the sensor node.

•V-8: Now, Sn decrypts submessage SID_k using the long-term shared key (that is, LT_{key}) to obtain $h(ID_k')^*$, GW_{id}^*, C_k^*, $h(X)'^*$, Sn^*, and T''^*.

•V-9: Compares $T''^*=T''$, $GW_{id}^*= GW_{id}$, and $Sn^*=Sn$. If the GW node holds, then Sn ensures that the GW node is a legal node, and this request is for a legitimate user; otherwise, Sn terminates the system.

•V-10: Thereafter, the sensor node Sn computes a session key, $S_{key}=h(h(ID_k')^*||C_k^*||h(X)'^*||Sn||T''')$; here, T''' is the current timestamp of the sensor node.

•V-11: Computes $N_k=E_{Skey}[Sn||C_k|| h(X)'^*||T''']$ and sends the message $<N_k, Sn, T'''>$ to the user U_k.

•V-12: Upon receiving the sensor node message $<N_k, Sn, T'''>$, user validates the timestamp, as follows, $(T^*-T''') > \Delta T$; if yes, terminates the system; otherwise, proceeds to the next step. Here, ΔT is considered to be an expected time interval for the message transmission delay, and T^* is the current timestamp of the user system.

•V-13: Now, the user U_k computes the session key $S_{key}=h(h(ID_k)||C_k||h(X)||Sn||T''')$ and decrypts N_k to obtain Sn^*,

C_k*, $h(X)'$**, and T'''* and compares T'''*=T''', Sn*=Sn, C_k*=C_k, and $h(X)'$**=$h(X)$. If all the values are identical, then the sensor node is a legal node; otherwise, it is a fake node and terminates the system.

3. Password-Change Phase

The password-change/update phase is invoked when a user wants to change or update his/her password, as follows.

1) User U_k, inserts the smartcard into the terminal and enters ID_k and PW_k.
2) Now, the smartcard verifies user's credentials with pre-stored values, as follows:
   a) Computes B_k*= $h(ID_k||h(b\oplus PW_k)\oplus A_k)$.
   b) Verifies B_k*=B_k; if yes, proceeds to the next step; otherwise, password change request fails.
3) The smartcard asks for the new password $h(b_{new}\oplus PW_{knew})$ and computes B_{knew}= $h(ID_k||h(b_{new}\oplus PW_{knew})\oplus A_k)$.
4) Finally, the smartcard replaces B_k with B_{knew} and b with b_{new}.

IV. Analysis of Proposed Scheme

In this section, we discuss the security and performance analysis of the proposed scheme. We compare the proposed scheme with the work in [5]-[10].

1. Security Analysis

To analyze the security of the proposed scheme, we assume that an adversary (*Tom*) has full control over the wireless channels between all entities (that is, the GW, user, and sensor). He/she can easily intercept, eavesdrop on, insert information into, and delete the messages from the wireless communication. Moreover, as referred to in [31], we also assume that *Tom* may either (i) hack the password of user U_k or (ii) extract the smartcard secret numbers using [32], [33], but not both (i) and (ii) at the same time. Based on the above assumptions, the security analysis of the proposed user authentication scheme is as follows.

User-impersonation attack. Consider that *Tom* eavesdrops on a previous login message $<P_k, A_k, T'>$ and tries to impersonate a legal user (*Allen*) for login at the GW node at time T^* ($>T'$). *Tom* forwards the login message $<P_k, A_k, T^*>$ to the GW node at T^*. Upon receiving the message from *Tom* at time T'', the GW node checks T^* according to $(T''-T^*) > \Delta T$. It is very obvious that T^* is a valid time; however, the GW node will capture the modified time (T^*) of *Tom* because the submessage $P_k=E_M[h(ID_k)||C_k||h(X)||T']$ contains the current timestamp of the system of *Allen* (that is, T'). As a result, T^* is not verified (that is, $T^*\neq T'$) at the GW, and the GW node aborts the system.

Moreover, in the proposed scheme, *Tom* cannot alter the login messages of *Allen* since all the messages are kept confidential (for example, $P_k=E_M[h(ID_k)||C_k||h(X)'||T']$ and $A_k=E_j[ID_k||GW_{id}||h(X)]$) using secret keys ($M=h(h(X)||ID_k||T'$) and J), which are only known to the legal parties (*Allen* and GW node). Hence, the proposed scheme is strong against user impersonation attacks and message alteration attacks, whereas Yoo and others' protocol is susceptible to user impersonation attacks and message-alteration attacks.

MITM attack. It is obvious that *Tom* may attempt an MITM attack by modifying $<SID_k,T''>$ between the GW node and Sn at time T^*. However, this attempt will not succeed at T^* because submessage ($SID_k=E_{LTkey}[h(ID_k')||GW_{id}||C_k||h(X)'||Sn||T'']$) contains GW node actual time (T'') and it will not be verified at Sn in step V-9 (refer authentication phase) due to the $T^* \neq T''$. Moreover, the submessage SID_k is kept confidential using LT_{key}, and *Tom* does not have knowledge about LT_{key}, (= h ($GW_{id}||Sn_{id}||h(Y)$)), which is shared between the legal GW node and Sn. Thus, the proposed scheme is secure against MITM attacks.

Replay attack. In a replay attack, *Tom* may collect messages over a public network and try to replay them to the GW node, Sn, and user. Assume that *Tom* intercepts login message $<P_k, A_k, T'>$ of *Allen* and tries to log into the GW node by replaying it. The verification of a replayed message will fail due to the time interval $(T''-T') > \Delta T$; here, T'' is the current timestamp of the GW node. For instance, *Tom* modifies the timestamp T' as (T^*) and then replays the login request. However, the request cannot pass the verification from the GW node because the temporary key (that is, $M=h(h(X)||ID_k'||T')$) is computed over $h(X)$, ID_k', and the current timestamp (T') of a legal user (*Allen*). A similar obstacle exists if *Tom* intercepts a valid GW node message $<SID_k, T''>$and tries to replay it to Sn. The verification of a replayed message will fail due to the time interval $(T'''-T'')>\Delta T$; here, T''' is the current timestamp of Sn. Likewise, verification of a replayed message will fail if *Tom* intercepts a valid message $<N_k, Sn, T'''>$ from the Sn and tries to replay it to the user. The verification of the replayed message will not succeed due to the freshness of the time interval $(T^*-T''')>\Delta T$; here, T^* is the current timestamp of the user. The freshness of every message is validated by the current timestamp (that is, $(T''-T')>\Delta T$, $(T'''-T'')>\Delta T$, and $(T^*-T''')>\Delta T$); thus, the proposed scheme is secure against replay attacks.

Privilege-insider attack. Assume that the GW insider can steal/forge a user's password and later try to impersonate that user. This attack could be very harmful in real-time applications [6]-[8]. However, in the registration phase, a user sends his/her password as a hashed value ($h(b\oplus PW_k)$) rather

than a plain password. An insider cannot see the user's password; more importantly, here, b is a high entropy number that is not revealed to the GW (that is, insider) [7], [8]. Thus, the proposed scheme is not susceptible to the privilege-insider attack.

Parallel session attack. In this attack, an adversary who does not know the user's identity and password wants to masquerade as a legitimate user by creating a valid login message from the information acquired by eavesdropping on the communication between the user and the GW node [20]. Assume that *Tom* intercepts the login message $<P_k, A_k, T'>$ of legal user *Allen* at time T_{Tom} and starts a new session with the GW node by sending an altered login message $<P_k, A_k, T_{Tom}>$. Upon receiving the login request from *Tom*, the GW node checks the validity of T_{Tom} and computes the temporary key (that is, $M_{Tom}=h(h(X)||ID_k'||T_{Tom})$). Because M_{Tom} is an incorrect temporary key, the submessage P_k ($=E_M[h(ID_k)||C_k||h(X)'||T']$) cannot be decrypted. Moreover, if the GW node computes the correct key, M (temporary key), the GW node will still reject the login request from *Tom* at step V-4 because $T_{Tom}^* \neq T'$. Consequently, the proposed scheme is secure against the parallel session attack.

GW-bypass attack. In this attack, an adversary can bypass the GW using a fabricated message and set up an independent communication with the sensor node and query for the data accordingly. Assume that *Tom* forwards a fabricated request to the sensor node $<SID_{Tom}, T''>$. However, the request from *Tom* will not be accepted and computed at *Sn* because *Tom* does not have knowledge about the LT_{key} ($=h(GW_{id}||Sn_{id}||h(Y))$), which is shared between the legal GW node and *Sn*. Hence, a GW-bypass attack will not work on the proposed scheme.

Gateway-masquerade attack. An intruder cannot acquire the secret key Y since it is a high entropy value and hashed with GW_{id} and Sn_{id} ($LT_{key}=h(GW_{id}||Sn_{id}||h(Y))$). Thus, an adversary cannot masquerade as a server.

Stolen-verifier attack. The proposed scheme is strong against the stolen-verifier since the GW does not maintain a password/verifier table.

Key-guessing attack. In the proposed scheme, the long-term key ($LT_{key} = h(GW_{id}||Sn_{id}||h(Y))$), the temporary key ($M = h(h(X)||ID_k||T')$), and the secret numbers (X and Y) are hashed values, which are difficult to invert. Moreover, the secret numbers are not transmitted as plain text; therefore, the attacker cannot apply a guessing attack. Hence, the proposed scheme is strong against key guessing attacks.

Many logged-in users with same login ID. In this attack, the login ID and password are verified using a verifier table, which should be stored on the system [4], [8]. However, in the proposed scheme, the GW node does not maintain a verifier table; hence, this attack is not applicable to the proposed scheme. Moreover, the proposed scheme exploits the smartcard computation to log into the network, and the login session is terminated if the smartcard is removed from the user system [4]. Hence, the proposed scheme is strong against many logged-in users with same login ID attacks.

Mutual authentication between the user and the sensor. In the proposed scheme, mutual authentication of U_k and Sn is achieved, maintaining trust for both entities. As shown in Fig. 2, when the GW node receives the login message $<P_k, A_k, T'>$ from the legitimate user, the GW node authenticates the user legitimacy by verifying the $h(ID_k)^*=h(ID_k')$, as shown in the authentication phase, step V-5, and sends another message to the sensor node (that is, $<SID_k, T''>$). When the sensor node receives message $<SID_k, T''>$ from the GW node, it ensures that the request is generated by the real GW node (as shown in the authentication phase, step V-9) and is generated for the legal user. Likewise, when the user receives a message from the sensor node (that is, $<N_k, Sn, T'''>$), then he/she ensures that the response is generated from the real sensor node (as shown in the authentication phase, step V-13). Hence, the proposed scheme maintains trust for both parties (that is, user and sensor), which was not the case in [5]-[10].

Message confidentiality. Generally, protocol messages are transmitted over public communication channels, so protocol messages should be confidential. In this regard, our proposed scheme provides confidentiality (for example, $P_k=E_M[h(ID_k)||C_k||h(X)||T']$, $A_k=E_f[ID_k||GW_{id}||h(X)]$, $SID_k=E_{LT_{key}}[h(ID_k')||GW_{id}||C_k||h(X)'||Sn||T'']$, and $N_k=E_{Skey}[Sn||C_k||h(X)'^*||T''']$). Additionally, the keys are confidential. Hence, the proposed protocol achieves message confidentiality.

User privacy. Assume that *Tom* intercepts the login message $<P_k, A_k, T'>$ but cannot get the user identity from the submessage P_k, which is confidential via temporary key M, and M is not easily computable. Furthermore, in submessage (P_k), the user's identity is hashed (that is, $h(ID_k)$), and it is difficult to invert a hash function. Hence, in the proposed scheme, an adversary cannot discover the user's real identity; only the GW node can discover the user's identity, as it is assumed that the GW node is a trusted party. This security feature was not considered in [8].

Strong session key establishment. A secure session key (that is, $S_{key}=h(h(ID_k')||C_k||h(X),||Sn||T''')$) is set up at the end of successful authentication to ensure that subsequent messages are transmitted securely. Hence, the proposed user authentication scheme facilitates secure session key establishment service, which is an important security service for the user authentication protocol.

2. Performance Analysis

This subsection discusses the performance analysis of the

Table 2. Computational cost comparison: proposed scheme and [5]-[10].

Schemes	Registration phase		Authentication phase (login and verification)		
	User	GW	User	GW	Sensor node
[5]	-	3H	1H	5H	1H
[6]	1H	5H	5H	5H	1H
[7]	1H	2H	3H	5H	2H
[8]	1H	3H	5H	8H	2H
[9]	-	3H	5H+1S	9H+1S	2H+2S
[10]	-	4H	4H	6H	1H
Ours	1H	2H+1S	5H+2S	2H+3S	1H+2S

Table 3. Computational cost (password change phase) comparison: proposed scheme and [5]-[10].

Scheme	[5]	[6]	[7]	[8]	[9]	[10]	Ours
User	-	6H	4H	5H	-	4H	5H

Table 4. Security features comparison: proposed scheme and [5]-[10].

Security features	[5]	[6]	[7]	[8]	[9]	[10]	Ours
User privacy	No	No	No	No	No	No	Yes
Mutual authentication between sensor and user	No	No	No	No	No	No	Yes
Message confidentiality	No	No	No	No	No	No	Yes
Secure session key establishment	No	No	No	Yes	Yes	No	Yes
Password-change phase	No	Yes	Yes	Yes	No	Yes	Yes
Secure against message-alteration attack	No	No	No	No	No	No	Yes
Secure against privileged-insider attack	No	Yes	Yes	Yes	No	No	Yes
Secure against impersonation attack	No	No	No	No	No	No	Yes
Secure against parallel-session attack	No	No	No	Yes	No	No	Yes
Secure against gateway-bypass attack	No	No	partial	Yes	No	No	Yes
Secure against MITM	No	No	No	No	No	No	Yes

proposed scheme. We compare the performance to that of [5]-[10] in terms of computational and communication cost and security features. To evaluate the computational cost, we focus on the registration phase, authentication phase, and password-change phase. The following notations are used to analyze the computational cost:

- H = The time for performing a one-way hash function;
- S = The time for performing a symmetric encryption/decryption operation.

Computational cost. The computational cost for the registration phase at the GW is only a one-time job, and the GW can compute complex operations since it has a high-resource sensor. As shown in Table 2, the computational cost (that is, registration phase and authentication phase) of [5], [6], [7], [8], [9], and [10] is 10H, 17H, 13H, 19H, 19H+4S, and 15H, respectively. However, the computational cost of [5] and [7] is reasonable, whereas the computational cost of [6], [8]-[10] is very high and provides less security services. Furthermore, it can be clearly seen from Table 2 that our proposed scheme needs 11H and 8S and offers many security services.

In addition, the computational cost of the password-change phase is an optional service that is only performed on the (resource-rich) user side; however, the schemes in [6], [7], [8], and [10] require 6H, 4H, 5H, and 4H, respectively (as shown in Table 3); in [5] and [9], the authors did not consider this service. In the proposed scheme, the password-change phase requires 5H (that is, hash operations); refer to the information on the password-change phase in subsection III.3.

Communication cost. We compare the communication cost of the proposed scheme with that of Yoo and others' scheme. As shown in Fig. 2, the proposed scheme requires only three message exchanges to execute the whole scheme, whereas Yoo and others' scheme requires six message exchanges to execute the whole protocol, as shown in Fig. 1. Thus, the scheme in [8] is not efficient in terms of the communication cost.

Moreover, we summarize the security features of the proposed scheme and make comparisons with [5]-[10], as shown in Table 4. It is obvious that our scheme can achieve the paramount requirements for a secure and efficient user authentication protocol for WSN environment.

V. Conclusion

An efficient user authentication is always a big concern in WSNs. In 2012, Yoo and others presented a two-factor user authentication protocol as an enhancement to the work in [4], [5], [7], [9], [10]. However, in this paper, we showed that Yoo and others' scheme is vulnerable to impersonation attacks, message-alteration attacks, and MITM attacks and lacks the ability to protect user privacy and achieve proper mutual authentication. Moreover, the communication and computational costs of their scheme are too high. Consequently, the [4]-[10] schemes are not directly applicable to real-time sensor networks.

This paper presented a strong user authentication scheme

that maintains all the security features of [4]-[10] and provides additional security features for WSNs. The proposed scheme poses a viable defense against an adversary attempting to breach security. Further, our analyses showed that the proposed scheme suits real-time sensor networks in terms of the cost of computation and communication.

References

[1] K.H.M. Wong et al., "A Dynamic User Authentication Scheme for Wireless Sensor Networks," *Proc. IEEE Int. Conf. Sensor Netw.*, *Ubiquitous*, *Trustworthy Comput.*, Taichung, Taiwan, 2006.

[2] MICAz Datasheet, accessed 21 Dec. 2012. Available: http://www.openautomation.net/uploadsproductos/micaz_datasheet.pdf

[3] TelosB Datasheet. http://www.willow.co.uk/TelosB_Datasheet.pdf

[4] M.L. Das, "Two-Factor User Authentication in Wireless Sensor Networks," *IEEE Trans. Wireless Commun.*, vol. 8, no. 3, 2009, pp. 1086-1090.

[5] T.-H. Chen and W.-K. Shih, "A Robust Mutual Authentication Protocol for Wireless Sensor Networks," *ETRI J.*, vol. 32, no. 5, Oct. 2010, pp. 704-712.

[6] D. He et al., "An Enhanced Two-Factor User Authentication Scheme in Wireless Sensor Networks," *Ad Hoc Sensor Wireless Netw.*, vol. 0, 2010, pp. 1-11.

[7] M.K. Khan and K. Alghathbar, "Cryptanalysis and Security Improvement of Two-Factor User Authentication in Wireless Sensor Networks," *Sensors*, 2010, pp. 2450-2459.

[8] S.-G. Yoo, K.-Y. Park, and J. Kim, "A Security-Performance-Balanced User Authentication Scheme for Wireless Sensor Networks," *Int. J. Distr. Sensor Netw.*, 2012, Article ID 382810.

[9] D. Nyang and M. Lee, "Improvement of Das's Two-Factor Authentication Protocol in Wireless Sensor Networks," Cryptology ePrint Archive 2009/631, accessed 21 Apr. 2012. http://eprint.iacr.org/2009 /631.pdf

[10] H.F. Huang, Y.F. Chang, and C.H. Liu, "Enhancement of Two-Factor User Authentication in Wireless Sensor Networks," *Proc. 6th Int. Conf. Intell. Inf. Hiding Multimedia Signal Process.*, Darmstadt, Germany, 2010, pp. 27-30.

[11] P. Kumar et al., "RUASN: A Robust User Authentication Framework for Wireless Sensor Networks," *Sensors*, vol. 11, 2011, pp. 5020-5046.

[12] H. Lee et al., "Security Weaknesses of Dynamic ID-Based Remote User Authentication Protocol," *World Academy Sci., Eng., Technol.*, vol. 59, no. 35, 2009, pp. 190-193.

[13] H.R. Tseng, R.H. Jan, and W. Yang, "An Improved Dynamic User Authentication Scheme for Wireless Sensor Networks," *IEEE GLOBECOM*, Washington, DC, 2007, pp. 986-990.

[14] T.H. Lee, "Simple Dynamic User Authentication Protocols for Wireless Sensor Networks," *Proc. 2nd Int. Conf. Sensor Technol. Appl.*, Cap Esterel, France, 2008, pp. 657- 660.

[15] L.-C. Ko, "A Novel Dynamic User Authentication Scheme for Wireless Sensor Networks," *Proc. IEEE ISWCS*, Reykjavik, Iceland, 2008, pp. 608-612.

[16] Z. Benenson, N. Gedicke, and O. Raivio, "Realizing Robust User Authentication in Sensor Networks," *Workshop Real-World Wireless Sensor Netw.*, Stockholm, Sweden, 2005.

[17] B. Vaidya, J.J.P.C. Rodrigues, and J.H. Park, "User Authentication Schemes with Pseudonymity for Ubiquitous Sensor Network in NGN," *Int. J. Commun. Syst.*, vol. 23, issue 9-10, Sept.-Oct. 2010, pp. 1201-1222.

[18] E.-J. Yoon and K.-Y. Yoo, "Cryptanalysis of Robust Mutual Authentication Protocol for Wireless Sensor Networks," *Proc. 10th IEEE Int. Conf. Cog. Inf. Cog. Comput.*, Banff, Alberta, Canada, 2011, pp. 392-396.

[19] P. Kumar and H.-J. Lee, "Cryptanalysis on Two User Authentication Protocols Using Smartcard for Wireless Sensor Networks," *Proc. 7th IEEE Conf. Wireless Adv.*, King's College, London, U.K., June 2011, pp. 241-245.

[20] T.-H. Chen, H.-C. Hsiang, and W.-K. Shih, "Security Enhancement on Two Remote User Authentication Schemes Using Smart Cards," *Future Generation Comput. Syst.*, vol. 27 Apr. 2011, pp. 377-380.

[21] D. Wang and C. Ma, "On the (In)security of Some Smart-Card-Based Password Authentication Schemes for WSN." Available: https://eprint.iacr.org/2012/581.pdf

[22] Crossbow Stargate Datasheet, accessed 21 Dec. 2012. Available: http://platformx.sourceforge.net/home.html

[23] G. Manes et al., "A Wireless Sensor Network for Precise Volatile Organic Compound Monitoring," *Int. J. Distr. Sensor Netw.*, 2012, Article ID 820716.

[24] X. Lin et al., "SAGE: A Strong Privacy-Preserving Scheme against Global Eavesdropping for eHealth Systems," *IEEE J. Sel. Areas Commun.*, vol. 27, no. 4, May 2009, pp. 365-378.

[25] L. Krishnamurthy et al., "Design and Deployment of Industrial Sensor Networks: Experiences from a Semiconductor Plant and the North Sea," *Proc. SenSys*, San Diego, CA, USA, 2-4 Nov. 2005, pp. 64-75.

[26] A. Koubaa and M. Alves, "A Two-tiered Architecture for Real-Time Communications in Large-Scale Wireless Sensor Networks: Research Challenges," Technical Report (TR-050701), v. 1.0, July 2005, accessed 18 Jan. 2013. Available: http://www.open-zb.net/publications/tr-hurray-050701.pdf

[27] H.-R. Tseng, R.-H. Jan, and W. Yang, "A Robust Password-based Authentication Scheme for Heterogeneous Sensor Networks," *Commun. Institute Inf. Comput. Mach.*, vol. 11, no. 3, 2008, pp. 1-13.

[28] A. Gurtov, M. Komu, and R. Moskowitz, "Host Identity Protocol: Identifier/Locator Split for Host Mobility Identity and

Multihoming," *Internet Protocol J.*, vol. 12, no. 1, Mar. 2009, pp. 27-32.

[29] Y. Zhang et al., "A Secure Hierarchical Key Management Scheme in Wireless Sensor Networks," *Int. J. Distr. Sensor Netw.*, 2012, Article ID547471.

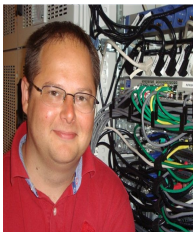
[30] C. Karlor, N. Sastry, and D. Wagner, "TinySec: A Link Layer Security Architecture for Wireless Sensor Networks," *Proc. ACM SenSys*, Baltimore, MD, USA, Nov. 3-5, 2004, pp. 162-175.

[31] C. Chen et al., "Lightweight and Provably Secure User Authentication with Anonymity for the Global Mobility Networks," *Int. J. Commun. Syst.*, vol. 24, no. 3, 2011, pp. 347-362.

[32] P. Kocher, J. Jaffe, and B. Jun, "Differential Power Analysis," *Proc. Adv. Cryptology*, Santa Barbara, CA, USA, Aug. 1999, pp. 388-397.

[33] T.S. Messerges, E.A. Dabbish, and R.H. Sloan, "Examining Smart-Card Security Under the Threat of Power Analysis Attack," *IEEE Trans. Comput.*, vol. 51, no. 5, May 2002, pp. 541-552.

Pardeep Kumar received his MTech in computer science and engineering from Chaudhary Devilal University, Sirsa (Hr.), India, in 2006, and his Ph.D. in computer science from Dongseo University, Rep. of Korea, in 2012. Since April 2012, he has been working with the Centre for Wireless Communications and Department of Communication Engineering, University of Oulu, Finland. His current research interests include secure wireless communications, security in sensor networks, body area networks, secure routing protocols, Internet of Things, and computer networks.



Andrei Gurtov received his MSc (2000) and PhD (2004) in computer science from the University of Helsinki, Finland. He is presently a visiting scholar at the International Computer Science Institute (ICSI), Berkeley. He was a professor at the University of Oulu in the area of wireless Internet in from 2010 to 2012. He is also a principal scientist leading the Networking Research Group at the Helsinki Institute for Information Technology (HIIT). Previously, he worked at TeliaSonera, Ericsson NomadicLab, and the University of Helsinki. Dr. Gurtov is a co-author of over 130 publications, including two books, research papers, patents, and IETF RFCs. He is a senior member of IEEE.

Mika Ylianttila received his PhD in communication engineering at the University of Oulu in 2005. He has worked as a researcher and professor in the Department of Electrical and Information Engineering. He is the director of the Center for Internet Excellence (CIE) research and innovation unit. He is a part-time professor in the Department of Communications Engineering in the field of broadband communications networks and systems, with special focus on wireless Internet technologies. His research interests include the future Internet, mobile networking and applications, Internet of Things, 3D Internet, and M2M and P2P networking. He is a senior member of IEEE.

Sang-Gon Lee received his BEng, MEng, and PhD degrees in electronics engineering from Kyungpook National University, Rep of Korea, in 1986, 1988, and 1993, respectively. He is a professor in the Division of Computer & Information Engineering, Dongseo University. He was a visiting scholar at QUT, Australia, from August 2003 to July 2004 and at the University of Alabama at Huntsville, USA, from July 2012 to Jun 2013. His research areas include information security, network security, wireless mesh/sensor networks, and the future Internet.

HoonJae Lee received his BS, MS, and PhD degrees in electronic engineering from Kyungpook National University, Daegu, Rep. of Korea, in 1985, 1987, and 1998, respectively. He is currently a professor in the Department of Information Communication Engineering at Dongseo University. From 1987 to 1998, he was a research associate at the Agency for Defense Development (ADD). His current research interests include developing secure communication systems, side-channel attacks, and ubiquitous sensor network/radio frequency identification security.