Cryptanalysis of the IoT notion-based Authentication and Key Agreement Scheme for Wireless Sensor Networks*

Sooyeon Shin and Taekyoung Kwon†
Graduate School of Information, Yonsei University, Seoul, 03722, South Korea
shinsy80@gmail.com, taekyoung@yonsei.ac.kr

Abstract

In WSNs (Wireless Sensor Networks) that can be deployed for IoT (Internet of Things) applications, secure and reliable user authentication and key agreement is an operational challenge and active research area. Most recently, Tai et al. showed that the Turcanović et al.'s scheme suffers from two fatal security flaws; user anonymity violation and session key leakage using the compromised sensor node. They then proposed an improvement of Turcanović et al.'s scheme based on the IoT notion for heterogeneous ad hoc WSNs by taking the following five factors into consideration: user anonymity, no complex computations, mutual authentication, user friendly, and ensuring the correctness of the session key earlier. However, we find that the Tai et al.'s scheme achieves user anonymity but does not provide sensor node anonymity and mutual authentication between a user and a sensor node and still has security problems. In this paper, we show the security problems of Tai et al.'s scheme in details. We also briefly present the solutions of those problems.

Keywords: Internet of Things, Wireless Sensor Networks, User Authentication, Key Agreement

1 Introduction

WSNs play a vital role in IoT environments since they cover a wide application field for IoT. In WSNs, small, wireless, heterogeneous, and ad hoc sensor nodes are deployed in an area of interest (e.g., home, building, factory, forest, hostile area, etc.) and interconnected to provide the sensed data to the remote end users. Due to the wireless nature of the communication channel and the resource-constrained nodes, they vulnerable to various security and privacy risks. To protect WSNs from the security threats user authentication and key agreement is one of the most essential security services.

Many two-factor-based authentication schemes have been proposed [8, 7, 9, 14, 15] since Das et al. introduced a user authentication scheme for WSN based on password and smart card as two factors in 2009 [5]. In 2014, Turkanović et al.proposed an efficient user authentication and key agreement scheme for heterogeneous ad hoc WSNs by employing only hash function and XOR (exclusive-OR) operation [13]. Turkanović et al.'s scheme used a different authentication model from the most previous research, in which a use contacts and authenticates directly with a sensor node. Turkanović et al. claimed that their scheme provides energy efficiency, user anonymity, mutual authentication between all parties, password protection, password changing, and dynamic node addition and also is resilient to cryptographic attacks. However, their scheme was later proved insecure and vulnerable [3, 6, 1, 12].

Most recently, in 2017, Tai et al. also showed that Turkanovic et al.'s scheme [13] suffers from two fatal security flaws: it does not provide user anonymity and session key shared between another sensor

Research Briefs on Information & Communication Technology Evolution (ReBICTE), Vol. 3, Article No. 12 (November 15, 2017)

*This work was supported in part by the National Research Foundation of Korea (NRF-2016-R1C1B2011095) and also by the MSIT(Ministry of Science and ICT), Korea, under the ITRC(Information Technology Research Center) support program (IITP-2017-2016-0-00304) supervised by the IITP (Institute for Information & communications Technology Promotion).

†Corresponding author: Graduate School of Information, Yonsei University, 50 Yonsei-ro Seodaemun-gu, Seoul, 03722, South Korea, Tel: +82-2-2123-4523

node and user who has ever connected to a compromised sensor node can be leaked [12]. Tai et al. proposed an improvement of Turkanović et al.'s scheme with preserving advantages of Turkanović et al.'s scheme and remedying its security flaws. They claimed that their improved scheme provides user anonymity, no complex computations, mutual authentication between all parties, user friendly, and ensuring the correctness of the session key earlier. However, we found that Tai et al.'s scheme is susceptible to several attacks and has security flaws.

In this paper, therefore, we aim to identify and present the vulnerabilities and security flaws of Tai et al.'s scheme [12]. We show that their scheme suffers from sensor spoofing attack with sensor node capturing, privileged-insider attack, and session-specific temporary information attack. We also show that their scheme fails to provide sensor node anonymity and mutual authentication between user and sensor node.

The rest of this paper is organized as follows. Section 2 provides a brief review and of Tai et al.'s scheme. Section 3 describes the security problems of the reviewed scheme. Section 4 presents the solutions of the security problems of the Tai et al.'s scheme. Finally, we conclude the paper in Section 5.

2 Review of Tai et al.'s scheme [12]

In this section, we describe an IoT-notion-based authentication and key agreement scheme ensuring user anonymity for heterogeneous ad hoc wireless sensor networks by Tai et al. [12]. Notations of Tai et al.'s scheme are listed in Table. Tai et al.'s scheme has six phases: pre-deployment phase, registration phase, login phase, authentication phase, password-change phase, and dynamic node addition phase. We present essential phases of Tai et al. scheme. These phases are described as follows.

Table 1: List of notations used in Tai et al.'s scheme.

Notation	Description	Notation	Description
SC	Smart card	X_{GWN}, X_U	Secure password keys known only to the GWN
U_i	User	X_{GWN-i}	Shared secure password between GWN and U_i
S_j	Sensor node	X_{GWN-j}	Shared secure password between GWN and S_j
GWN	Gateway node	SK	Agreed session key of the user and sensor node
ID_i	Identity of the user U_i	T_x	Timestamp
PW_i	Password of the user U_i	ΔT	Time interval for the allowed transmission delay
SID_j	Identity of the sensor node S_j	$h(\cdot)$	Cryptographic one-way hash function
$\|\|$	Concatenation operation	$\oplus$	Bit wise XOR operation

2.1 Pre-deployment phase

Before registration, a network administrator predefines a pair of identifier SID_j and password X_{GWN-j} for each regular sensor node S_j, where $1 \leq j \leq m$ and m is the number of sensor nodes in the WSN. X_{GWN-j} is randomly generated and stored in S_j's memory. For GWN, the administrator predefines two secure password keys X_{GWM} and X_U that are only known to GWN and stored in GWN's memory. GWN stores SID_j and X_{GWN-j} for S_j.

2.2 Registration phase

There are two registration phases are needed after the sensor node deployment; user registration phase and sensor-node registration phase.

2.2.1 User registration phase

User registration is initiated by a user U_i on demand. After registration, U_i can access any sensor node.

(1) U_i chooses her/his identity ID_i and password PW_i and sends $\{ID_i, PW_i\}$ as the registration request to GWN via a secure channel.

(2) GWN randomly chooses a password key X_{GWN-i} for U_i and stores ID_i and X_{GWN-i} into its memory. It computes $f_i = h(ID_i||X_{GWN}), x_i = h(ID_i||PW_i||X_{GWN-i})$, and $e_i = h(PW_i) \oplus X_U$.

(3) GWN writes $\{f_i, x_i, e_i, X_{GWN-i}, h(\cdot)\}$ into a smart card's memory and issues this smart card to U_i via a secure channel.

2.2.2 Sensor-node registration phase

After the deployment of sensor nodes in the target field, this phase is conducted.

(1) S_j computes $MP_j = h(SID_j||T_1||X_{GWN-j})$, where T_1 is the S_j's current timestamp and sends the registration request $\{SID_j, MP_j, T_1\}$ to GWN.

(2) After receiving the registration request from the S_j, GWN checks $|T_1 - T_C| < \Delta T$, where T_C is the current timestamp of GWN. If not so, GWN transmits a rejection message to S_j.

(3) Otherwise, GWN finds the corresponding X_{GWN-j} using the received SID_j and computes $MP_j^* = h(SID_j||T_1||X_{GWN-j}$. GWN verifies $MP_j^* \stackrel{?}{=} MP_j$, if not so, GWN terminates this phase and sends a rejection message to S_j. Otherwise, GWN computes $f_j = h(SID_j||X_{GWN}), x_j = h(T_2||X_{GWN-j}), e_j = f_j \oplus x_j$, and $z_j = h(f_j||e_j||T_2||X_{GWN-j})$, where T_2 is GWN's current timestamp. GWN sends a response message $\{e_j, z_j, T_2\}$ to S_j.

(4) On obtaining GWN's response, S_j checks $|T_2 - T_C| < \Delta T$, where T_C is the current timestamp of S_j. If not so, S_j terminates this phase and sends a rejection message and a request to GWN for re-executing this phase. Otherwise, S_j computes $x_j^* = h(T_2||X_{GWN-j}), f_j^* = e_j \oplus x_j^*$, and $z_j^* = h(f_j^*||e_j||T_2||X_{GWN-j})$. S_j then verifies $z_j^* \stackrel{?}{=} z_j$, if not so, S_j asks GWN to re-send $\{e_j, z_j\}$. If S_j still cannot verify the resent $\{e_j, z_j\}$ successfully, this phase will be re-executed immediately. If $z_j^* = z_j$, S_j confirms that $f_j^* = f_j$ and stores f_j^* in its memory.

2.3 Login phase

U_i need to login in order to access information from the WSN.

(1) U_i inserts her/his SC into the card reader and inputs ID_i and PW_i.

(2) SC computes $x_i^* = h(ID_i||PW_i||X_{GWN-i})$ using the inputted ID_i and PW_i and X_{GWN-i} stored in its memory. SC then verifies $x_i^* \stackrel{?}{=} x_i$, if not so, this phase will be terminated. If U_i inputs the wrong password more than three times, SC will be locked immediately. If $x_i^* = x_i$, SC chooses a random number K_i and computes $MI_i = h(T_1||h(PW_i) \oplus e_i) \oplus ID_i, Z_i = K_i \oplus h(T_1||X_{GWN-i})$, and $N_i = h(MI_i||ID_i||K_i||f_i||T_1||X_{GWN-i})$, where T_1 is U_i's current timestamp.

(3) U_i chooses a sensor node S_j and sends an authentication request $\{MI_i, Z_i, N_i, T_1\}$ to S_j via a public channel.

2.4 Authentication phase

With the help of GWN, U_i and S_j can authenticate each other and negotiate a session key shared between U_i and S_j.

(1) After receiving the authentication request from U_i, S_j checks $|T_1 - T_C| < \Delta T$, where T_C is the current timestamp of S_j. If not so, S_j terminates this phase and sends a rejection message to U_i. Otherwise, S_j chooses a random number K_j and computes $A_j = h(N_i||T_2||X_{GWN-j}) \oplus K_j$ and $B_j = h(A_j||K_j||T_2||f_j)$, where T_2 is the current timestamp of S_j. S_j then sends $\{MI_i, Z_i, N_i, T_1, SID_j, A_j, B_j, T_2\}$ to GWN.

(2) On obtaining $\{MI_i, Z_i, N_i, T_1, SID_j, A_j, B_j, T_2\}$ from S_j, GWN checks $|T_2 - T_C| < \Delta T$, where T_C is the current timestamp of GWN. If not so, GWN terminates this phase and sends a rejection message to S_j. Otherwise, GWN finds the corresponding X_{GWN-j} using the received SID_j and computes $K_j^* = h(N_i||T_2||X_{GWN-j}) \oplus A_j, f_j^* = h(SID_j||X_{GWN})$, and $B_j^* = h(A_j||K_j^*||T_2||f_j^*)$. GWN then checks $B_j^* \stackrel{?}{=} B_j$, if not so, GWN aborts all further actions and sends a rejection message to S_j. Otherwise, GWN authenticates successfully S_j.

(3) GWN computes $ID_i^* = MI_i \oplus h(T_1||X_U)$ and finds the corresponding X_{GWN-i} using ID_i^*. GWN computes $f_i^* = h(ID_i^*||X_{GWN}), K_i^* = Z_i \oplus h(T_1||X_{GWN-i})$, and $N_i^* = h(MI_i||ID_i^*||K_i^*||f_i^*||T_1||X_{GWN-i})$. GWN then checks $N_i^* \stackrel{?}{=} N_i$, if not so, GWN aborts all further actions and sends a rejection message indicating that U_i is illegal to S_j. Otherwise, GWN can confirm that U_i and S_j are legal.

(4) GWN computes $R_i = K_j^* \oplus h(T_3||N_i||f_i^*||X_{GWN-i}), R_j = K_i^* \oplus h(T_3||B_j||f_j^*||X_{GWN-j})$, and $F_{ij} = h(T_1||T_2||T_3||R_i||K_i^*||K_j^*)$, where T_3 is GWN's current timestamp. GWN then sends $\{R_i, R_j, F_{ij}, T_1, T_2, T_3\}$ to S_j.

(5) After receiving $\{R_i, R_j, F_{ij}, T_1, T_2, T_3\}$ from GWN, S_j checks $|T_3 - T_C| < \Delta T$. If not so, all further actions will be aborted and S_j sends a rejection message to GWN and U_i. Otherwise, S_j computes $K_i^* = R_j \oplus h(T_3||B_j||f_j^*||X_{GWN-j})$ and $F_{ij}^* = h(T_1||T_2||T_3||R_i||K_i^*||K_j)$. S_j then checks $F_{ij}^* \stackrel{?}{=} F_{ij}$, if not so, S_j asks GNW to resend the message. If S_j still cannot verify the resent message successfully, all further actions will be aborted and S_j sends a rejection message to GWN and U_i. Otherwise, if $F_{ij}^* = F_{ij}$, S_j computes the session key $SK = h(K_i^* \oplus K_j)$ shared with U_i and $R_{ij} = h(T_1||T_2||T_3||T_4||K_i^*||K_j||SK)$, where T_4 is S_j's current timestamp, and sends $\{R_i, R_{ij}, T_1, T_2, T_3, T_4\}$ to U_i.

(6) On obtaining $\{R_i, R_{ij}, T_1, T_2, T_3, T_4\}$ from S_j, U_i checks $|T_4 - T_C| < \Delta T$, if not so, U_i aborts all further actions and sends a rejection message to S_j. Otherwise, SC computes $K_j^* = R_i \oplus h(T_3||N_i||f_i||X_{GWN-i})$, the session key $SK^* = h(K_i \oplus K_j^*)$ shared with S_j, and $R_{ij}^* = h(T_1||T_2||T_3||T_4||K_i||K_j^*||SK^*)$. It then checks $R_{ij}^* \stackrel{?}{=} R_{ij}$, if not so, U_i asks S_j to re-send the message $\{R_i, R_{ij} T_1, T_2, T_3, T_4\}$. If the resent message is still not verified successfully, U_i terminates this phase and sends a rejection message to S_j. Otherwise, if $R_{ij}^* = R_{ij}$, U_i can confirm that GWN and S_j are legal and the computed SK^* is equal to S_j's SK.

3 Cryptoanalysis of Tai et al.'s scheme [12]

This section presents the security problems of Tai et al.'s scheme. We discuss the security weaknesses of the scheme and show that an attacker can mount different types of attacks on Tai et al.'s scheme.

3.1 No sensor node anonymity

In the authentication phase, the sensor node S_j sends the request message $\{MI_i, Z_i, N_i, T_1, SID_j, A_j, B_j, T_2\}$ to the gateway node GWN through an insecure channel. As can be clearly seen, if an attacker $\mathcal{A}$ intercepts this request message from the public channel, $\mathcal{A}$ can obtains S_j's identity SID_j. Thus, the anonymity of sensor nodes is not preserved in Tai et al.'s scheme.

3.2 Lack of mutual authentication

Mutual authentication of all involved parties is highly essential in a user authentication and key agreement scheme. Tai et al. stated that their scheme provides mutual authentication between any two of a gateway node, a sensor node, and a user. However, in Tai et al's scheme, a user cannot authenticate a sensor node.

In Tai et al.'s scheme, U_i should authenticate the chosen sensor node S_j by the help of GWN. However, in the last step of the authentication phase, S_j delivers only one value R_i received from GWN to U_i and R_i does not include any information to authenticate S_j. U_i utilizes this value to extract K_j^* for computing SK that will be shared with S_j in this session. Moreover, U_i verifies only session key through $R_{ij}^* \stackrel{?}{=} R_{ij}$ and does not verify the source authentication of the message $\{R_i, R_{ij} T_1, T_2, T_3, T_4\}$. In other words, U_i does not check whether the message is truly from the selected S_j with SID_j by herself/himself during the login phase. Due to the lack of mutual authentication, an attacker is able to perform the sensor node spoofing attack in the following section.

3.3 Sensor node spoofing attack with sensor node capturing

Since WSNs are installed in unattended or hostile environments, an attacker can easily capture or compromise a sensor node and extract important information stored inside its memory. In Tai et al.'s scheme, if an attacker $\mathcal{A}$ compromise one sensor node, $\mathcal{A}$ can masquerade any non-compromised and legitimate sensor node to which a user is tying to log in.

Suppose an attacker $\mathcal{A}$ compromise a sensor node S_j and obtain SID_j, X_{GWN-j}, and f_j from the compromised S_j. When a user U_i wants to log into the sensor node S_k, to launch a sensor node spoofing attack, $\mathcal{A}$ performs the following steps:

(1) When U_i sends $\{MI_i, Z_i, N_i, T_1\}$ to S_k, $\mathcal{A}$ intercepts that message and randomly chooses K_j'. Then, $\mathcal{A}$ computes $A_j' = h(N_i || T_2' || X_{GWN-j}) \oplus K_j'$ and $B_j' = h(A_j' || K_j' || T_2' || f_j)$ using S_j's compromised parameters X_{GWN-j} and f_j and the current timestamp T_2'. $\mathcal{A}$ sends $\{MI_i, Z_i, N_i, T_1, SID_j, A_j', B_j', T_2'\}$ to GWN.

(2) On receiving the above message from S_j, GWN performs the verification process as per step (2) in the authentication phase. Because M_i, Z_i, and N_i do not bound to S_k, GWN is unable to identify that they were actually sent to S_k, not to S_j. In addition, $\mathcal{A}$ used valid parameters of S_j to compute A_j' and B_j' and thus GWN trusts that the received message is valid and originated from the sensor node S_j chosen by U_i. GWN then computes R_i, R_j, and F_{ij} and sends $\{R_i, R_j, F_{ij}, T_1, T_2', T_3\}$ to $\mathcal{A}$ who is now impersonating the sensor node S_j.

(3) When receiving $\{R_i, R_j, F_{ij}, T_1, T_2', T_3\}$ from GWN, $\mathcal{A}$ obtains K_i^* using the compromised parameters f_j and X_{GWN-j} and computes $SK' = h(K_i^* \oplus K_j')$ and $R_{ij} = h(T_1 || T_2' || T_3 || K_i^* || K_j' || SK')$. $\mathcal{A}$ finally sends $\{R_i, R_{ij} T_1, T_2', T_3, T_4'\}$, where T_4' is the current timestamp of $\mathcal{A}$ to U_i.

(4) Upon receiving $\{R_i, R_{ij} T_1, T_2', T_3, T_4\}$ from S_j, U_i verifies the timestamp T_4' and obtains $K_j^* = R_i \oplus h(T_3 || N_i || f_i || X_{GWN-i})$. U_i then will successfully computes $SK^* = h(K_i || K_j^*)$ and verifies $R_{ij}^* \stackrel{?}{=} R_{ij}$.

In the end, $\mathcal{A}$ has succeeded in masquerading as the sensor node S_k.

3.4 Privileged-insider attack

It is common practice that users reuse passwords on multiple accounts [11, 4]. In such situations, if a privileged-insider, e.g., the system administrator, can misuse or disclose the user's passwords resulting in user impersonation at other application systems. This attack can happen when a user sends her/his password to the system administrator in plaintext [10].

In Tai et al.'s scheme, a user U_i sends the plaintext password to GWN in the registration phase. For convenience, if U_i submits the same password used in other systems to GWN, GWN can use the password to impersonate the victim user to access other systems. Thus, Tai et al.'s scheme is susceptible to privileged-insider attack.

3.5 Session-specific temporary information attack

Canetti and Krawczyk introduced a session-specific temporary information attack [2]. This attack implies that if the specific information temporarily generated for a session is leaked, the session key established in the specific session remains no more secure.

In Tai et al's scheme, U_i and S_j computes the session key agreed between them by solely depending on the temporary random numbers K_i and K_j generated by U_i and S_j, respectively. If these two temporary numbers K_i and K_j are leaked then an attacker $\mathcal{A}$ can easily compute the session key $SK = h(K_i \oplus K_j)$ established between U_i and S_j. Thus, the security of the session key is under threat in case of the leakage of session-specific temporary information.

4 Solutions

In this section, we briefly present solutions of security flaws of the Tai et al.'s scheme.

4.1 Sensor node anonymity

In the Tai et al.'s scheme, the anonymity of sensor node does not be guaranteed because it sends the plaintext ID (SID_j) in the authentication phase. To solve this problem, a similar way to the method for providing user anonymity can be utilized. In the registration phase, a sensor node submits a secret value (i.e., PW_j) and in the authentication phase, it masks SID_j with the secret value such that $MI_j = h(T_2||h(PW_j)) \oplus SID_j$. There is an alternative in which the GWN issues a new secret value (i.e., X_{S_j}) for a sensor node S_j in the registration phase and the sensor node uses the corresponding value when masking the SID_j in the authentication phase. The details of alternative is as follows.

- Sensor node registration phase

(3) GWN randomly chooses another secret value for S_j, X_{S_j} and computes $c_j = h(X_{S_j}||X_{GWN}), d_j = c_j \oplus x_j^*$, and $z_j = h(f_j||e_j||c_j||d_j||T_2||X_{GWN-j})$. GWN then appends d_j to the response message, such that $\{e_j, d_j, z_j, T_2\}$ and stores c_j with SID_j and X_{GWN-j} in the memory.

(4) S_j computes $c_j^* = d_j \oplus x_j^*$ and $z_j^* = h(f_j^*||e_j||c_j^*||d_j||T_2||X_{GWN-j})$. It then verifies $z_j^* \stackrel{?}{=} z_j$, if so, S_j stores c_j^* with f_j^* in its memory.

- Authentication phase

(1) S_j sends $MI_j = SID_j \oplus h(T_2||c_j)$ instead of SID_j to the GWN.

(2) After checking the timestamp T_2, GWN computes $SID_j^* = MI_j \oplus h(T_2||c_j)$.

The above mentioned methods do not send the sensor node ID as the plaintext, thus they can provide the sensor node anonymity.

4.2 Prevention of sensor node spoofing attack and mutual authentication

There are two main reasons that the sensor node spoofing attack with sensor node capturing can be launched against the Tai et al.'s scheme. The message $\{MI_i, Z_i, N_i, T_1\}$ sent by the user to the sensor node at the step (3) of the login phase and the message $\{MI_i, Z_i, N_i, T_1, SID_j, A_j, B_j, T_2\}$ sent by the sensor node to the GWN at the step (1) of the authentication phase are not bound to each other. Thus, it is impossible for the GWN to check whether the sensor node to which the user wants to access and selects at the login phase is a sensor node that has sent the message to the GWN in the authentication phase. It is also impossible for the user to confirm that the message $\{R_i, R_{ij}T_1, T_2, T_3, T_4\}$ received at the step (6) of the authentication phase is from the sensor node that he or she selected in the login phase. These two reasons are also related to the lack of mutual authentication mentioned in Section 3.2. Therefore, to prevent the sensor node spoofing attack with sensor node capturing and to provide proper mutual authentication between a user and a sensor node, a user should include information about the sensor node (i.e., SID_j) to be accessed in the login message, such that $N_i = h(MI_i||ID_i||SID_j||K_i||f_i||T_1||X_{GWN-i}))$. Then, the GWN can check whether the sensor node that the user wants to access matches the sensor node that sent the message at the authentication phase. The GWN should also include SID_j in the $R_i = K_j^* \oplus h(T_3||N_i||SID_j^*||f_i^*||X_{GWN-i})$ used to extract K_j and further provide a message (i.e., D_{ij}) similar to F_{ij} to the user to confirm that the extracted K_j is correct. At this time, the GWN should use a secret value X_{GWN-i} shared with the user to prevent the message from being modified by the sensor node, such that $D_{ij} = h(T_1||T_2||T_3||R_i||K_i^*||K_j^*||X_{GWN-i})$.

Now, the user's login message and the authentication message of the sensor node are bound to each other and the user can confirm from the GWN that he/she has exchanged messages and shared the session key with the sensor node he/she selected in the login phase. Therefore, an attacker is unable to launch a sensor node spoofing attack with sensor node capturing and a user is able to authenticate a sensor node properly.

4.3 Prevention of privileged-insider attack

There is a method to prevent the privileged-insider attack regardless of whether users reuse passwords on multiple accounts. Instead of sending the plaintext password to the GWN in the registration phase, it is to allow the user to mask the password with a random value (i.e., a_i) known only to the user, such that $MPW_i = h(a_i||PW_i)$, and allow to use a different random number for each gateway node. In addition, the user does not store the random values used for password masking on the smart card, but stores the value (i.e., $b_i = a_i \oplus h(ID_i||PW_i)$) that can be extracted when the correct ID and password are entered at the login phase.

In this method, GWN does not know both the user's password PW_i and the random value a_i used for masking, thus the GWN can not used the user's password to access other systems by impersonating the user.

4.4 Prevention of session-specific temporary information attack

As a solution to prevent the session-specific temporary information attack, we change the method of computing the session key from $SK = h(K_i \oplus K_j)$ to $SK = h(h(ID_i||K_i) \oplus h(SID_j||K_j))$. The steps (4),

(5), and (6) in the authentication phase is partially modified to enable both user and sensor node to compute the changed session key. The modified version also includes the contents of Section 4.2.

(4) GWN computes $NK_i = h(ID_i||K_i^*), NK_j = h(SID_j||K_j^*), R_i = NK_j \oplus h(T_3||N_i||SID_j^*||f_i^*||X_{GWN-i}), R_j = NK_i \oplus h(T_3||B_j||f_j^*||X_{GWN-j})$, and $D_{ij} = h(T_1||T_2||T_3||R_i||NK_i||NK_j||X_{GWN-i}), F_{ij} = h(T_1||T_2||T_3||R_i||NK_i||NK_j)$, where T_3 is GWN's current timestamp. GWN then sends $\{R_i, R_j, D_{ij}, F_{ij}, T_1, T_2, T_3\}$ to S_j.

(5) After receiving $\{R_i, R_j, D_{ij}, F_{ij}, T_1, T_2, T_3\}$ from GWN, S_j checks $|T_3 - T_C| < \Delta T$. If not so, all further actions will be aborted and S_j sends a rejection message to GWN and U_i. Otherwise, S_j computes $NK_j^* = h(SID_j||K_j), NK_i^* = R_j \oplus h(T_3||B_j||f_j^*||X_{GWN-j})$ and $F_{ij}^* = h(T_1||T_2||T_3||R_i||NK_i^*||NK_j^*)$. S_j then checks $F_{ij}^* \stackrel{?}{=} F_{ij}$, if not so, S_j asks GNW to resend the message. Otherwise, if $F_{ij}^* = F_{ij}$, S_j computes the session key $SK = h(NK_i^* \oplus NK_j^*)$ shared with U_i and $R_{ij} = h(T_1||T_2||T_3||T_4||NK_i^*||NK_j^*||SK)$, where T_4 is S_j's current timestamp, and sends $\{R_i, D_{ij}, R_{ij}, T_1, T_2, T_3, T_4\}$ to U_i.

(6) On obtaining $\{R_i, D_{ij}, R_{ij}, T_1, T_2, T_3, T_4\}$ from S_j, U_i checks $|T_4 - T_C| < \Delta T$, if not so, U_i aborts all further actions and sends a rejection message to S_j. Otherwise, SC computes $NK_i^* = h(ID_i||K_i), NK_j^* = R_i \oplus h(T_3||N_i||SID_j||f_i||X_{GWN-i})$, the session key $SK^* = h(NK_i^* \oplus NK_j^*)$ shared with S_j, and $D_{ij}^* = h(T_1||T_2||T_3||R_i||NK_i^*||NK_j^*||X_{GWN-i})$. It then checks $D_{ij}^* \stackrel{?}{=} D_{ij}$, if not so, U_i asks S_j to re-send the message. Otherwise, if $D_{ij}^* = D_{ij}$, U_i computes $R_{ij}^* = h(T_1||T_2||T_3||T_4||NK_i^*||NK_j^*||SK)$ and checks $R_{ij}^* \stackrel{?}{=} R_{ij}$. If not so, U_i asks S_j to re-send the message. If the resent message is still not verified successfully, U_i terminates this phase and sends a rejection message to S_j. Otherwise, if $R_{ij}^* = R_{ij}$, U_i can confirm that GWN and S_j are legal and the computed SK is equal to S_j's SK.

In the modified version, if the identities of the user and the sensor node is not known, it is impossible to compute the session key, even if the session-specific temporary information (K_i and K_j) is leaked. Also, it is difficult to derive the identities of the user and the sensor node from the messages exchanged due to user anonymity and sensor node anonymity. Therefore, the modified version is not vulnerable to the session-specific temporary information attack.

5 Conclusion

In this paper, we have reviewed the recently proposed Tai et al.'s authentication and key agreement scheme for heterogeneous ad hoc WSNs. We have then analyzed the security problems of Tai et al.'s scheme. We have pointed out that Tai et al.'s scheme failed to provide sensor node anonymity and mutual authentication. We have also identified that Tai et al.'s scheme failed to resist to sensor node spoofing attack with sensor node capturing, privileged-insider attack, and session-specific temporary information attack. We have briefly presented the solutions of the security flaws that we pointed out.

In the future work, based on the solutions mentioned in Section 4, we will propose an enhanced user authentication and key agreement scheme for heterogeneous ad hoc WSNs. We will also analysis security and performance of the enhanced scheme.

References

[1] R. Amin and G. Biswas. A secure light weight scheme for user authentication and key agreement in multi-gateway based wireless sensor networks. *Ad Hoc Networks*, 36:1:58–1:80, January 2016.

[2] R. Canetti and H. Krawczyk. Analysis of key-exchange protocols and their use for building secure channels. In *Proc. of the International Conference on the Theory and Application of Cryptographic Techniques Innsbruck (Eurocrypt'01), Austria, LNCS*, volume 2045, pages 453–474. Springer-Verlag, May 2001.

[3] C. C. Chang and H. D. Le. A provably secure, efficient, and flexible authentication scheme for ad hoc wireless sensor networks. *IEEE Transactions on Wireless Communications*, 15(1):357–366, January 2016.

[4] A. Das, J. Bonneau, M. Caesar, N. Borisov, and X. Wang. The tangled web of password reuse. In *Proc. of the Network and Distributed System Security Symposium (NDSS'14), San Diego, California, USA*, volume 14, pages 23–26. Internet Society, 2014.

[5] M. L. Das. Two-factor user authentication in wireless sensor networks. *IEEE Transactions on Wireless Communications*, 8(3):1086–1090, March 2009.

[6] M. S. Farash, M. Turkanović, S. Kumari, and M. Hölbl. An efficient user authentication and key agreement scheme for heterogeneous wireless sensor network tailored for the internet of things environment. *Ad Hoc Networks*, 36:1:152–1:176, January 2016.

[7] D. He, Y. Gao, S. Chan, C. Chen, and J. Bu. An enhanced two-factor user authentication scheme in wireless sensor networks. *Ad Hoc & Sensor Wireless Networks*, 10:361 – 371, January 2010.

[8] H. F. Huang, Y. F. Chang, and C. H. Liu. Enhancement of two-factor user authentication in wireless sensor networks. In *Proc. of the 6th International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP'10), Darmstadt, Germany*, pages 27–30. IEEE, October 2010.

[9] M. K. Khan and K. Alghathbar. Cryptanalysis and security improvements of 'two-factor user authentication in wireless sensor networks'. *Sensors*, 10(3):2450–2459, March 2010.

[10] S. Kumari, M. K. Khan, and M. Atiquzzaman. User authentication schemes for wireless sensor networks: A review. *Ad Hoc Networks*, 27:159–194, April 2015.

[11] E. Stobert and R. Biddle. The password life cycle: User behaviour in managing passwords. In *The Proc. of the Symposium On Usable Privacy and Security (SOUPS'14), Menlo Park, California, USA*, pages 243–255. USENIX Association, July 2014.

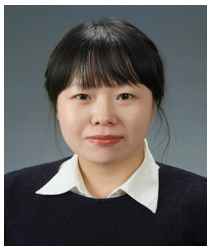
[12] W.-L. Tai, Y.-F. Chang, and W.-H. Li. An IoT notion–based authentication and key agreement scheme ensuring user anonymity for heterogeneous ad hoc wireless sensor networks. *Journal of Information Security and Applications*, 34:2:133–2:141, June 2017.

[13] M. Turkanović, B. Brumen, and M. Hölbl. A novel user authentication and key agreement scheme for heterogeneous ad hoc wireless sensor networks, based on the Internet of Things notion. *Ad Hoc Networks*, 20:96–112, September 2014.

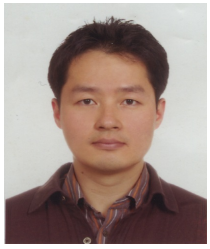
[14] M. Turkanović and M. Hölbl. An improved dynamic password-based user authentication scheme for hierarchical wireless sensor networks. *Elektronika ir Elektrotechnika*, 19(6), 2013.

[15] K. Xue, C. Ma, P. Hong, and R. Ding. A temporal-credential-based mutual authentication and key agreement scheme for wireless sensor networks. *Journal of Network and Computer Applications*, 36(1):316–323, January 2013.

Author Biography



Sooyeon Shin received her B.S., M.S., and Ph.D. degrees in computer science and engineering from Sejong University, Seoul, Korea, in 2004, 2006, and 2012, respectively. From 2012 to 2013, she was a post-doctoral researcher at Sejong University. In 2013, she joined Yonsei University, Seoul, Korea, to continue her post-doctoral research. Her current research interests include cryptographic protocol, privacy preservation, user authentication, computer network security, wireless sensor network security, and usable security.



Taekyoung Kwon received his B.S., M.S., and Ph.D. degrees in computer science from Yonsei University, Seoul, Korea, in 1992, 1995, and 1999, respectively. He is currently an Associate Professor of information at Yonsei University, Seoul, Korea. From 1999 to 2000, he was a Post-Doc Researcher at the University of California, Berkeley, CA, USA. From 2001 to 2013, he was a professor of computer engineering at Sejong University, Seoul, Korea. In 2013, he returned to Yonsei University, Seoul, Korea. His current research interests include applied cryptography, cryptographic protocol, network protocol, usable security, and human-computer interactions.