A Secure User Authentication and Key Agreement Scheme for HWSN Tailored for the Internet of Things Environment

Hamidreza Yazdanpanah, Mohammadreza Hasani Ahangar, Mahdi Azizi and Arash Ghafouri
Department of Information and Communications Technology
Imam Hossein University
Tehran, Iran
Email: [hryazdanpanah@ihu.ac.ir], [mrhasani@ihu.ac.ir], [mazizi@ihu.ac.ir], [krghafouri@ihu.ac.ir]

***Abstract*— Internet of things (IOT) is the term used to describe a world in which the things interact with other things through internet connection or communication means, share the information together and or people and deliver a new class of capabilities, application and services; the world in which all things and heterogeneous devices are addressable and controllable. Wireless Sensor Networks (WSN) play an important role in such an environment, since they include a wide application field. Researchers are already working on how to integrate WSN better into the IoT environment. One aspect of it is the security aspect of the integration. In 2014, Turkanovi´c proposed a lightweight user authentication and key agreement protocol for heterogeneous WSN(HWSN) based on the internet of things concept. In this scheme, remote user can access a single desired sensor node from the WSN without the necessity of firstly connecting with a gateway node (GWN). Moreover, this scheme is lightweight because it based on a simple symmetric cryptography and it uses simple hash and XOR computations. Turkanovi´c et al.'s scheme had some security shortages and it was susceptible to some security attacks. Recently Sabzinejad Farash et al. proposed an efficient user authentication and key agreement scheme for HWSN tailored for the Internet of Things environment based on Turkanovi´c et al.'s scheme. Although their scheme is efficient, we found out that this scheme is vulnerable to some cryptographic attacks. In this paper, we demonstrate some security weaknesses of the Sabzinejad Farash et al.'s scheme and then we propose an improved and secure mutual authentication and key agreement scheme.**

Keywords— Internet of Things; Wireless Sensor Networks; Vulnerability; Mutual Authentication; Key agreement

I. INTRODUCTION

We are surrounded by pervasive, smart interconnected objects which engage us with new applicative perspectives on our everyday lives, like wearables, smartphones, smart cars, wireless sensors, RFID and other smart things. This can be seen as the Internet of Things. Wireless Sensor Networks (WSN) play an important role in such an environment, since they include a wide application domain [1,2]. Today's WSN can be heterogeneous, large-scale and have mobile nodes. Hence we can talk about the use of WSN for smart city, smart home, monitoring, healthcare, smart agriculture, smart industrial and smart supply chain [3,4]. In view of the IoT concept, the heterogeneity of a WSN is not the only thing rapidly adapting, hence the infrastructure has moved from mainly infrastructure based networks, where nodes can only communicate directly with the base station, to ad hoc networks whereby nodes can also communicate directly with each other and with rest of the world. When a remote user wants to access a particular node of the WSN, such a user needs to be authorized and, if done positively, allowed to gather data from or send commands to the node. The most significant and distinct characteristic of WSNs is their resource constrained architecture with limited computational power, transmission range and battery life, therefore a lightweight security solution is required. Moreover, Heterogeneous WSN consists of at least one sink node, also called gateway node (GWN). GWN are bigger, more secure and have more computational and communicational capabilities [1,5]. By facilitating with authenticated mechanism, the GWN help to process mutual authentication and key agreement protocols by playing the lightweight role of a trusted third party entity [6]. A key challenge is how to enable the establishment of a shared cryptographic key in a secure and lightweight manner, between the sensor node and the user outside the network. Mutual authentication is also needed for such a scenario, and is very important because all parties need to be sure of the legitimacies of all the entities involved [7]. An important amount of user authentication schemes for WSN were presented in the literature, whereby the most of them applies the principle where a user firstly connects to the gateway node in order to access the data from the WSN or a single sensor node.

In 2014, Turkanovi´c et al. [7] proposed a lightweight user authentication and key agreement scheme for heterogeneous WSN(HWSN) based on the internet of things concept. Turkanovi´c et al.'s scheme is lightweight because it based on a simple symmetric cryptography and it uses simple hash and XOR computations. This scheme initiates the authentication and key agreement protocol by firstly contacting the specific sensor node and uses four-step authentication model that is the most appropriate for the mentioned scenario, when a remote user wants to connect to a node inside a WSN. The authentication procedure is run over the gateway node but the user never interacts with it directly, since this is done by the chosen sensor node. Also this scheme uses smart cards to authenticate the users. Recently Farash et al.'s [8] proved that the Turkanovi´c et al.'s scheme had some security shortages and it was susceptible to some security attacks. They proposed an improved user authentication and key agreement scheme for heterogeneous ad hoc wireless sensor network. Unfortunately, although the authors claim higher security level and resilience to a vast list of cryptographic attacks, we have discovered that Farash et al.'s scheme has some security weaknesses and it is Vulnerable against smart card stolen attack and sensor node stolen attack. Also their scheme does not have respect for anonymity of sensor nodes. In this paper, we demonstrate some security weaknesses of the Farash et al.'s scheme and then we propose an improved and secure mutual authentication and key agreement scheme.

The remainder of the paper is organized as follows. In section 2 presents an overview of some related works. Section 3 has a brief glance at Farash et al.'s scheme. some vulnerabilities and security weaknesses of the scheme are demonstrated in Section 4. Section 5 presents our new solution. The security features comparison is given in Section 6. Finally, we conclude the paper in Section 7.

II. RELATED WORKS

Recent works on authentication protocols in the Internet of Things environment can be divided into two classes namely: authentication with certification, and certificate less authentication [9]. In the first class, authentication is achieved on the basis of digital certificates where each object must have its own digital certificate. Among these protocols, DTLS (Datagram Transport Layer Security) authentication handshake has been proposed for the IoT [10]. This protocol offers an authentication between the different objects. However, its high consumption of energy caused by asymmetric encryption based RSA [11] and the PKI certificates exchanges constitute its main drawbacks. For this reason, Elliptic Curve Cryptography (ECC) has raised as an interesting approach compared to RSA based algorithms. In fact, it is more energy saving and less key size for the same level of security [9]. In the second class, authentication protocols do not need certification. It uses cryptographic operations such as XOR, hash functions, and symmetric cryptography. This class is often known for its high energy saving.

In 2006, Wong et al. [12] presented a user authentication scheme for WSN based on symmetric cryptography. They used only hash based computation, thus providing a lightweight architecture. However, it was later shown that their scheme was also vulnerable to multiple attacks (e.g., many logged-in users with the same login-id attack, stolen verifier attack, etc.). Das [13] improved the security of Wong et al.'s scheme and proposed an efficient password-based user authentication with the help of the GWN, that used temporal credentials (timestamps) for verification. it did not ensure mutual authentication and user anonymity and was vulnerable to several attacks (denial-of-service attack, node capture attack). In 2010, Khan and Alghathbar [14] also presented an improvement of Das's scheme. They solved the mutual-authentication and unsecured password problems by introducing pre-shared keys and hashed passwords. Vaidya et al. [15], later showed that Khan and Alghathbar's scheme was also vulnerable to several security attacks and proposed an improved version of the scheme. Additional improvement of Das's scheme presented by Chen and Shih [16]. Their scheme provided mutual authentication between all parties involved in the key agreement process but was later shown as being vulnerable to several attacks (e.g. replay attack). Das et al. [17] and Xue et al. [18] later presented two user authentication and key agreement schemes for WSN using smart cards. They are both designed to support a remote user to effectively and securely connect to the nodes of a WSN. Both schemes provide security features like mutual authentication, password protection, key agreement, resilience against several attacks, and a dynamic node addition phase. Both schemes use hash and XOR computations and are therefore lightweight and highly appropriate for WSN. It was later shown by Turkanovi´c and Hölbl [19] that Das et al.'s scheme has some shortages and is infeasible for implementations. They have proposed an amended version of Das et al.'s scheme. In 2014 Turkanovi´c et al. presented a novel user authentication and key agreement scheme for WSN, that a different authentication model is used [7]. Turkanovi´c et al. believe that such an authentication model regards the role of a single electronic device inside the IoT environment. Recently Farash et al. [8] proved that the Turkanovi´c et al.'s scheme had some security shortages and it was vulnerable to some security attacks. They proposed an improved user authentication and key agreement scheme for heterogeneous ad hoc wireless sensor network.

III. BRIEF REVIEW OF FARASH ET AL.'S SCHEME

Farash et al. [8] proposed an efficient user authentication and key agreement scheme for heterogeneous wireless sensor network tailored for the Internet of Things environment. Farash et al.'s scheme consists of three fundamental phases: Pre-deployment phase, registration phase, login and authentication phase. The notations used in the scheme are summarized in Table 1.

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

Notation	Description
SC	Smart card
U_i	User i
S_j	Sensor node
r_i, r_j	Secret random nonce of user and sensor node
GWN	Gateway node
PW_i	User Password
ID_i	User ID
SID_j	Sensor node ID
X_{GWN-Sj}	Shared secure password between GWN and S_j
X_{GWN-Ui}	Shared secure password between GWN and User i
X_{GWN}	Secure password known only to the GWN
T_i	Current timestamps
ΔT	Time interval for the allowed transmission delay
SK	Session key of the protocol
h( )	Cryptographic one-way hash function
$\oplus$, $\|\|$	XOR, Concatenation operation
MI_i, MP_i	User's masked identity and password

A. Pre-deployment phase

According to Farash et al.'s scheme the pre-deployment phase is same as Turkanovi´c et al.'s scheme, which is the setup phase, runs offline and is done by a network admin using a setup server. During the pre-deployment phase, each regular sensor node $\{S_j|1<=j<=m\}$ is predefined with its identity SID_j and a randomly-generated secure password-key X_{GWN-Sj}, which is shared with the GWN and stored in the memory of the node. the GWN is predefined with a randomly-generated highly secure password key X_{GWN} which is known only to the GWN and is secretly stored in the memory of the GWN. Additionally, the GWN stores the corresponding secret password key shared with each sensor node $\{X_{GWN-Sj} \mid 1<=j<=m\}$. The shared key X_{GWN-Sj} is used for the purpose of the registration phase.

B. Registration phase

There are two registration phases after the sensor node deployment, one is between the U_i and GWN and the second is between the S_j and the GWN. The procedure for both registration phases is represented in Fig.1 and Fig.2. The registration is done by using a smart card, which is personalized by the GWN at the end of the phase. An important improvement and difference between Farash et al. proposed scheme and Turkanovi´c et al. proposed scheme is the fact that the shared keys (X_{GWN-Sj}) can be deleted from the memory of both the GWN and S_j after the successful registration phase, since they are only used for the purpose of this phase. This difference enables a GWN to add numerous additional nodes to the network without filling its memory. At first, we will discuss user registration phase:

The user U_i selects his/her ID_i, PW_i and then generates a random number r_i and computes $MP_i = h(r_i \| PW_i)$. Finally, the U_i sends the message $\{MP_i, ID_i\}$ to the GWN through secure channel. After receiving the message, GWN computes $e_i = h(MP_i \| ID_i)$ and $d_i = h(ID_i \| X_{GWN})$, using his/her secret password-key X_{GWN}. Then GWN computes $g_i = h(X_{GWN}) \oplus h(MP_i \| d_i)$ and $f_i = d_i \oplus h(MP_i \| e_i)$. Afterwards GWN stores $\{e_i, f_i, g_i\}$ into the memory of smart card and sends it to the user U_i. At the end, the user U_i stores r_i into the smart card and completes the user registration phase.

Secondly, according to Farash et al.'s scheme the details of the S_j-GWN registration phase is as follows:
S_j chooses a random number r_j and computes MP_j, MN_j and sends $\{SID_j, MP_j, MN_j, T_1\}$ to the GWN through public channel (T_1 is the S_j's current timestamp). After receiving the registration message from the S_j the GWN initially checks the timestamp validity. If the validity does not hold, the GWN aborts any further operation and sends a rejection message to the S_j; otherwise, computes r'_j and own version of S_j's masked password MP'_j and check if it is equal to the received and original version of MP_j. If the verification is ok and the values are equal, the GWN confirms the legitimacy of the S_j. Then the GWN computes the values of x_j, e_j, d_j and f_j. Now, GWN sends $\{e_j, f_j, d_j, T_2\}$ to the S_j from public channel. S_j first checks the timestamp validity for a replay attack and then computes the value of x_j and its own version of the received value of f'_j and compares it to the received one. If the verification is ok, the S_j authenticates GWN. Also S_j computes $h(X_{GWN} \| 1) = d_j \oplus h(X_{GWN\text{-}Sj} \| T_2)$ and stores x_j and $h(X_{GWN\text{-}Sj} \| 1)$ into a memory. At the end, S_j deletes its shared key $x_{GWN\text{-}Sj}$ and sends a confirmation to the GWN. Also GWN deletes SID_j and $X_{GWN\text{-}Sj}$ from its memory.

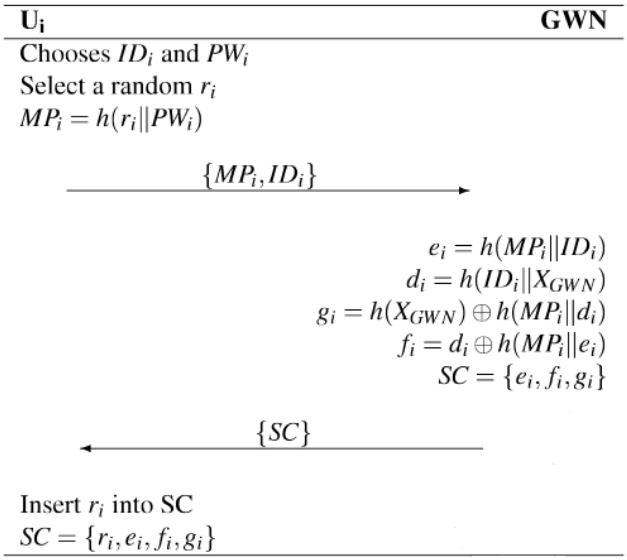

Figure 1. User Registration phase of Farash et al.'s scheme [8].

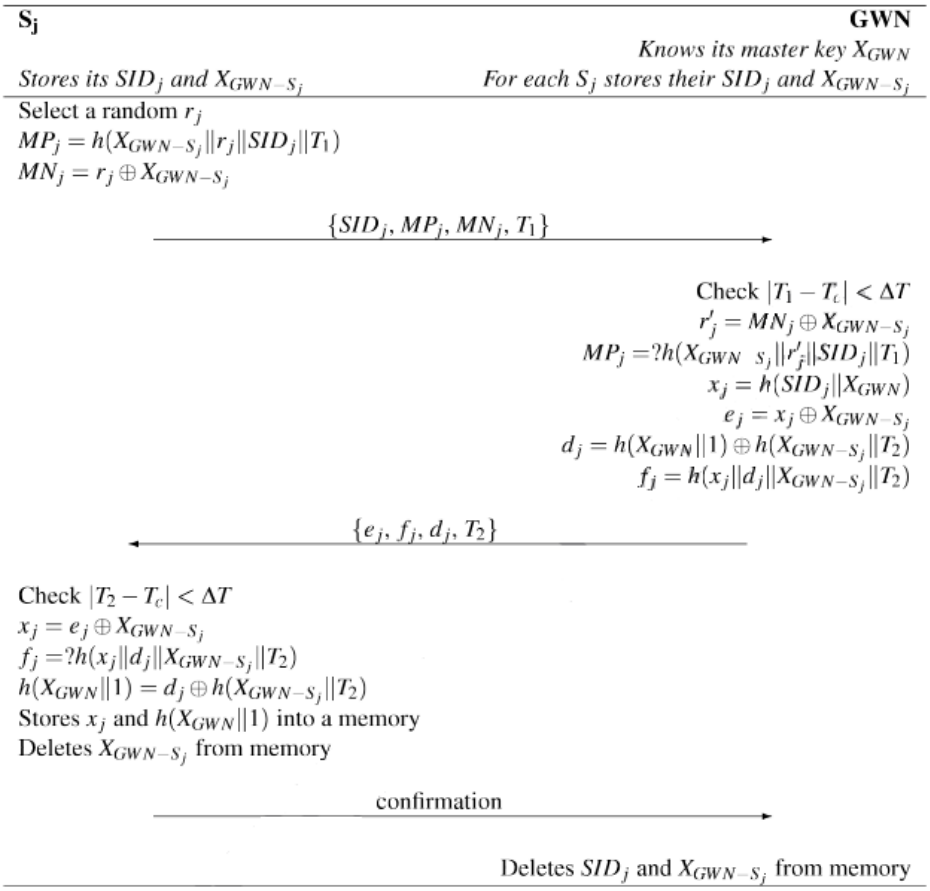

Figure 2. Sensor node registration phase of Farash et al.'s scheme [8].

C. Login and Authentication phase

Before the authentication, the user U_i has to login via smart card. At the first of this phase, the U_i inserts his/her smart card into the terminal and inputs its username ID'_i and password PW'_i. The SC verifies the owner of the SC with the help of the secret data stored in it. The SC computes $MP'_i = h(r_i \| PW'_i)$ using the inserted password (PW'_i). Then the SC computes $e'_i = h(MP'_i \| ID'_i)$ and compares it with the stored version of e_i ($e'_i ?= e_i$). If the verification is ok, the SC acknowledges the legitimacy of the U_i (finishing the login phase). Now, the SC computes $d_i = f_i \oplus h(MP'_i \| e_i)$ and $h(X_{GWN}) = g_i \oplus h(MP'_i \| d_i)$. Then the SC uses three auxiliary variables (M_1, M_2, M_3) which it will send over the insecure channel to the GWN. After computing $M_1 = ID'_i \oplus h(h(X_{GWN} \| T_1))$, the SC chooses a random nonce K_i and then computes $M_2 = K_i \oplus h(d_i \| T_1)$ and $M_3 = h(M_1 \| M_2 \| K_i \| T_1)$. Afterwards the SC sends the authentication message $\{M_1, M_2, M_3\}$ to the GWN via an insecure(public) channel.

After receiving the authentication message, the sensor node S_j has to check timestamp for reply attack ($|T_1-T_c| < \Delta t$) and If the verification is ok, the S_j computes $ESID_j = SID_i \oplus h(h(X_{GWN} \| 1)T_2)$. Then the S_j chooses a nonce K_j that later is used for second part of the shared session key SK between user and sensor node. Next the S_j computes M_4 and M_5. After computing the necessary value, the sensor node S_j sends authentication message $\{M_1, M_2, M_3, T_1, T_2, ESID_j, M_4, M_5\}$ to GWN via an insecure(public) channel. After receiving the authentication message from the sensor node S_j, the GWN firstly checks timestamp for replay attack. Then the GWN computed $SID'_j = ESID_i \oplus h(h(X_{GWN} \|1) \|T_2)$ with the values received in the authentication message. Afterwards GWN can compute x'_j and then K'_j. Now, the GWN can verify the legitimacy of the sensor node S_j by computing its own version of the value $M'_5 = h(SID'_j \| M_4 \| T_1 \| T_2 \| K'_j)$ and compare the result with received M_5. If it is ok and S_j is verified, the GWN then goes on and verifies the legitimacy of the user U_i. The U_i first need to compute the ID_i and then d'_i and K'_i. Now, the GWN can verify the legitimacy of U_i by computing $M_3 = ?h(M_1 \| M_2 \| K'_i \| T_1)$. Also the GWN uses four auxiliary variables (M_6, M_7, M_8, M_9) and computes $M_6 = K'j \oplus h(d'_1 \| T_3)$, $M_7 = K'_i \oplus h(x'_j \| T_3)$, $M_8 = h(M_6 \| d'_i \| T_3)$ and $M_9 = h(M_7 \| x'_j \| T_3)$. At the end the GWN sends the confirmation message $\{M_6, M_7, M_8, M_9, T_3\}$ to the S_j via a public channel. When sensor node S_j receives the confirmation message from the GWN, it shows that the U_i is a legitimate user. Afterwards the S_j has to check timestamp for any

reply attack and then checks the legitimacy of the received message by computing its own version of M'_9. The S_j then compares M_9 =?$h(M_7 \| x_j \| T_3)$. If the verification is ok, the sensor node S_j approve the legitimacy of the received message and thus authenticates the GWN. After approving the legitimacy of GWN, S_j goes on to extract the U_i secret session key $K'_i = M_7 \oplus h(x_j \| T_3)$ and then computes $SK= h(K'_i \oplus K_j)$. At the end S_j computes $M_{10} = h(SK \| M_6 \| M_8 \| T_3 \| T_4)$ and sends a confirmation message{ M_6, M_8, M_{10}, T_3, T_4} to U_i. When the user U_i receives the confirmation message, first check timestamp for any reply attack ($|T_4-T_c| < \Delta T$). Then the user U_i computing its own version of $M'_8 = h(M_6 \| d_i \| T_3)$ and compares it with the received one(M'_8 =? M_8). If the verification is ok, the user U_i successfully verified the legitimacy of GWN. Also the U_i extracts the S_j's session key ($K'_j = M_6 \oplus h(d_i \| T_3)$) and then computes the final session key $SK= h(K_i \oplus K'_j)$. At the end of this phase U_i need to verify the legitimacy of the S_j . This is done by computing its own version of $M'_{10} =h(SK \| M_6 \| M_8 \| T_3 \| T_4)$ and compare it with the received version(M'_{10} =? M_{10}). If the verification is ok, the U_i authenticated the S_j and thus successfully end the authentication phase.

IV. WEAKNESSES OF FARASH ET AL.'S SCHEME

In this section, we discuss several security weaknesses of the scheme proposed by Farash et al. such as smart card stolen attack and sensor node stolen attack. Also their scheme does not have respect for anonymity of sensor nodes. The description of all the security shortcomings of Farash et al. is presented below:

A. Lack of respect for anonymity of sensor nodes

At the beginning of Registration phase, the attacker can monitor SID_j, when sensor node S_j send registration message {SID_j, MP_j, MN_j, T_1} to the GWN through insecure (public) channel and this is violation of end nodes anonymity.

B. Discovering the user password

The authentication scheme proposed by Farash et al.'s is a two factor authentication mechanism that uses smart card and password. When this smart card is obtained for any reason by the attacker, he/she can access to the sensitive information in the smart card such as SC= {r_i, e_i, f_i, g_i} [20]. In this situation the attacker has r_i and he/she can obtain MP'_i by examining the different cases of PW_i in $MP'_i = h(r_i \| PW'_i)$ at the login phase. Examining the different cases of PW_i can be done by dictionary attack (brute force attack). Then the attacker uses the result (MP'_i) and computes $d_i = f_i \oplus h(MP'_i \| e_i)$. Afterwards, since the user U_i sends {M_1, M_2, M_3, T_1} to the S_j via insecure (public) channel, the attacker can monitor this information. Hence he/she can compute $K_i = M_2 \oplus h(d_i \| T_1)$. Now, the attacker uses the result of K_i and computes $M_3 = h(M_1 \| M_2 \| K_i \| T_1)$ and make finds that password guessing was successful. At this time the attacker knows the user password and the user session key(K_i).

C. Discovering the user ID

After guessing the correct password, the attacker can obtain user ID by examining the different cases of ID in e_i =? $h(MP'_i \| ID_i)$. It should be noted, the attacker extracted e_i from the smart card. At this time, the attacker can compute $h(h(X_{GWN})\|T_1) = M_1 \oplus ID'_i$ and then compute $SID_j = ESID_j \oplus h(h(X_{GWN}\|1)\|T_2)$ that this means lack of respect for anonymity of sensor nodes. Also it should be noted, the attacker monitors $ESID_j$ at the authentication phase; when S_j sends authentication message to the GWN through unsecure(public) channel.

D. Discovering the session key

As mentioned above (4.b), finally the attacker can compute the value of K_i. afterwards, he/she can monitor M_6 at the authentication phase; when GWN sends confirmation message to the sensor node S_j through unsecure(public) channel. Now the attacker computes $K'_j = M_6 \oplus h(d'_i \| T_3)$ and then obtains the value of SK by computing $SK = h(K_i \oplus K'_j)$.

$\mathbf{U_i}$	$\mathbf{S_j}$	GWN
Knows its ID_i *and* PW_i *Has a* $SC = \{r_i, e_i, f_i, g_i\}$	*Stores* SID_j, x_j *and* $h(X_{GWN}\|\|1)$	*Stores its master key* X_{GWN}

User: Inserts SC into a terminal
User: Inputs PW_i' and ID_i'
SC: $MP_i' = h(r_i\|\|PW_i')$
SC: $e_i = ?h(MP_i'\|\|ID_i')$
SC: $d_i = f_i \oplus h(MP_i'\|\|e_i)$
SC: $h(X_{GWN}) = g_i \oplus h(MP_i'\|\|d_i)$
SC: $M_1 = ID_i' \oplus h(h(X_{GWN})\|\|T_1)$
SC: Chooses a random nonce K_i
SC: $M_2 = K_i \oplus h(d_i\|\|T_1)$
SC: $M_3 = h(M_1\|\|M_2\|\|K_i\|\|T_1)$
User: Chooses S_j

$$\xrightarrow{\{M_1, M_2, M_3, T_1\}}$$

Checks $|T_1 - T_c| < \Delta T$
$ESID_j = SID_i \oplus h(h(X_{GWN}\|\|1)\|\|T_2)$
Chooses a random nonce K_j
$M_4 = h(x_j\|\|T_1\|\|T_2) \oplus K_j$
$M_5 = h(SID_j\|\|M_4\|\|T_1\|\|T_2\|\|K_j)$

$$\xrightarrow{\{M_1, M_2, M_3, T_1, T_2, ESID_j, M_4, M_5\}}$$

Checks $|T_2 - T_c| < \Delta T$
$SID_j' = ESID_i \oplus h(h(X_{GWN}\|\|1)\|\|T_2)$
$x_j' = h(SID_j'\|\|X_{GWN})$
$K_j' = M_4 \oplus h(x_j'\|\|T_1\|\|T_2)$
$M_5 = ?h(SID_j'\|\|M_4\|\|T_1\|\|T_2\|\|K_j')$
$ID_i' = M_1 \oplus h(h(X_{GWN})\|\|T_1)$
$d_i' = h(ID_i'\|\|X_{GWN})$
$K_i' = M_2 \oplus h(d_i'\|\|T_1)$
$M_3 = ?h(M_1\|\|M_2\|\|K_i'\|\|T_1)$
$M_6 = K_j' \oplus h(d_i'\|\|T_3)$
$M_7 = K_i' \oplus h(x_j'\|\|T_3)$
$M_8 = h(M_6\|\|d_i'\|\|T_3)$
$M_9 = h(M_7\|\|x_j'\|\|T_3)$

$$\xleftarrow{\{M_6, M_7, M_8, M_9, T_3\}}$$

Check $|T_3 - T_c| < \Delta T$
$M_9 = ?h(M_7\|\|x_j\|\|T_3)$
$K_i' = M_7 \oplus h(x_j\|\|T_3)$
$SK = h(K_i' \oplus K_j)$
$M_{10} = h(SK\|\|M_6\|\|M_8\|\|T_3\|\|T_4)$

$$\xleftarrow{\{M_6, M_8, M_{10}, T_3, T_4\}}$$

Check $|T_4 - T_c| < \Delta T$
$M_8 = ?h(M_6\|\|d_i\|\|T_3)$
$K_j' = M_6 \oplus h(d_i\|\|T_3)$
$SK = h(K_i \oplus K_j')$
$M_{10} = ?h(SK\|\|M_6\|\|M_8\|\|T_3\|\|T_4)$

Figure 3. Login and Authentication phase of Farash et al.'s scheme [8].

V. PROPOSED IMPROVED SCHEME

The proposed scheme is based on Farash et al.'s scheme, hence eliminates mentioned security shortages and vulnerabilities of their scheme. the functionality and efficiency of the proposed scheme stays at the same level as Farash et al.'s. A brief interpretation of this scheme will be presented below:

A. Pre-deployment phase

pre-deployment phase of the proposed scheme is same as Farash et al.'s scheme which was already described in section 3.1. It just should be noted that, the shared key X_{GWN-Sj} is used for the purpose of the registration phase.

B. Registration phase

The registration phase of our proposed scheme is divided into two parts, the user registration and sensor node registration part. The user registration process is done via a secure channel but the sensor node registration process is done via an insecure channel. At first, we describe the user registration phase of the proposed scheme.

The user U_i selects his/her ID_i, PW_i and then generates a random number r_i and computes $MP_i = h(r_i \| PW_i)$. Then, the U_i sends the message $\{MP_i, ID_i\}$ to the GWN through secure channel. After receiving the message, GWN computes $e_i = h(MP_i \| ID_i)$ and $d_i = h(ID_i \| X_{GWN})$, using his/her secret password-key X_{GWN}. Then GWN computes $g_i = h(X_{GWN}) \oplus h(MP_i \| d_i)$ and $f_i = d_i \oplus h(MP_i \| e_i)$. Afterwards GWN stores $\{e_i, f_i, g_i\}$ into the memory of smart card and sends it to the user U_i. At the end, the user U_i computes $q_i = r_i \oplus ID_i$ and stores q into the smart card ($SC = \{q_i, e_i, f_i, g_i\}$) and completes the user registration phase.

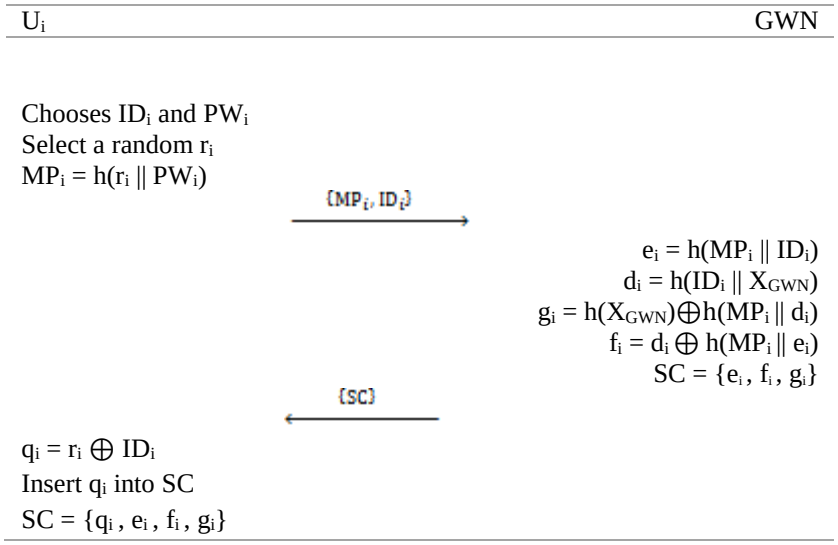

Figure 4. User registration phase of the proposed scheme.

Secondly, the sensor node registration phase of the proposed scheme is same as Farash et al.'s scheme which was already described in section 3.2. The depiction of it is presented in Fig. 2. A notable improvement and difference between our proposed scheme and Farash et al.'scheme in this phase is the fact that after computing $MP_j = h(X_{GWN-Sj} \| r_j \| SID_j \| T_1)$ and $MN_j = r_j \oplus X_{GWN-Sj}$ we need to compute $MZ_j = SID_j \oplus X_{GWN-Sj}$ and then sends $\{MZ_j, MP_j, MN_j, T_1\}$ to the GWN through public channel. In this case, the attacker cannot be able to monitor SID_j, when sensor node S_j send registration message to the GWN through insecure (public) channel. It should be noted; we have to store MZ_j in the memory of GWN (this means: for each S_j, GWN stores SID_j, X_{GWN-Sj} and MZ_j).

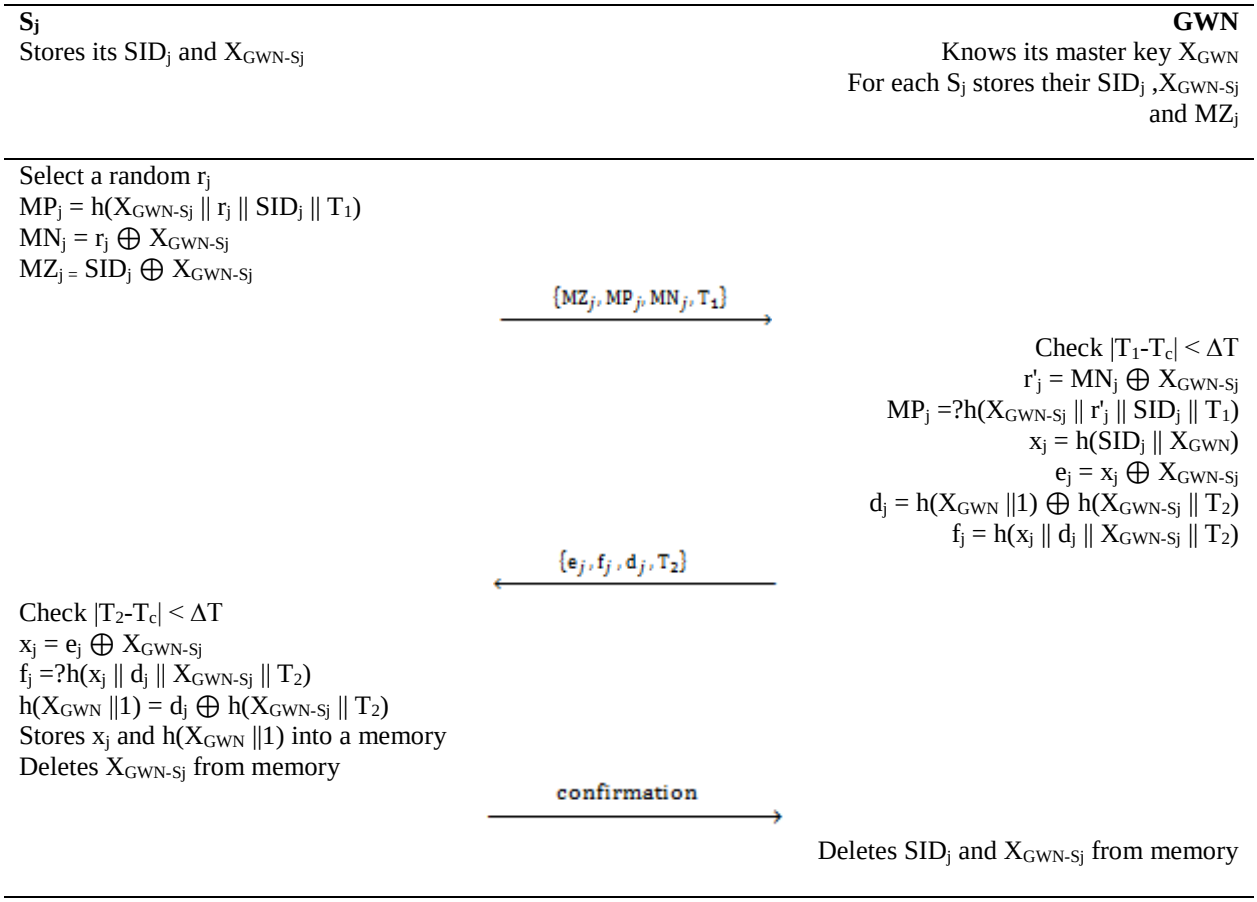

Figure 5. Sensor node registration phase of the proposed scheme.

C. Login and Authentication phase

Also the login and Authentication phase of our scheme is same as Farash et al.'s scheme which was already described in section 3.3. As mentioned above, the SC = {q_i, e_i, f_i, g_i}. A highlight improvement and difference between our proposed scheme and Farash et al.'scheme in this phase (Login phase) is the fact that we have to compute $r'_i = ID'_i \oplus q_i$ before computing $MP'_i = h(r'_i || PW'_i)$. In this case the attacker cannot be able to guess the correct password and ID because he/she does not have the value of r_i. In addition, when the attacker cannot be able to obtain r_i from the smart card, so he/she cannot compute the value of session key (As mentioned above in 4.b and 4.d).

U_i	S_j	GWN
Knows its ID_i and PW_i Has a SC ={q_i, e_i, f_i, g_i }	Stores SID_j , x_j and $h(X_{GWN} \| 1)$	Stores its master key X_{GWN}

User: Inserts SC into a terminal
User: Inputs PW'_i and ID'_i
SC: $r'_i = ID'_i \oplus q_i$
SC: $MP'_i = h(r'_i \| PW'_i)$
SC: $e_i = ?h(MP'_i \| ID'_i)$
SC: $d_i = f_i \oplus h(MP'_i \| e_i)$
SC: $h(X_{GWN}) = g_i \oplus h(MP'_i \| d_i)$
SC: $M_1 = ID'_i \oplus h(h(X_{GWN}) \| T_1)$
SC: Chooses a random nonce K_i
SC: $M_2 = K_i \oplus h(d_i \| T_1)$
SC: $M_3 = h(M_1 \| M_2 \| K_i \| T_1)$
User: Chooses S_j

$\{M_1, M_2, M_3, T_1\}$ →

Checks $|T_1 - T_c| < \Delta T$
$ESID_j = SID_j \oplus h(h(X_{GWN} \| 1) \| T_2)$
Chooses a random nonce K_j
$M_4 = h(x_j \| T_1 \| T_2) \oplus K_j$
$M_5 = h(SID_j \| M_4 \| T_1 \| T_2 \| K_j)$

$\{M_1, M_2, M_3, T_1, T_2, ESID_j, M_4, M_5\}$ →

Checks $|T_2 - T_c| < \Delta T$
$SID'_j = ESID_j \oplus h(h(X_{GWN} \| 1) \| T_2)$
$x'_j = h(SID'_j \| X_{GWN})$
$K'_j = M_4 \oplus h(x'_j \| T_1 \| T_2)$
$M_5 = ?h(SID'_j \| M_4 \| T_1 \| T_2 \| K'_j)$
$ID'_i = M_1 \oplus h(h(X_{GWN}) \| T_1)$
$d'_i = h(ID'_i \| X_{GWN})$
$K'_i = M_2 \oplus h(d'_i \| T_1)$
$M_3 = ?h(M_1 \| M_2 \| K'_i \| T_1)$
$M_6 = K'_j \oplus h(d'_i \| T_3)$
$M_7 = K'_i \oplus h(x'_j \| T_3)$
$M_8 = h(M_6 \| d'_i \| T_3)$
$M_9 = h(M_7 \| x'_j \| T_3)$

← $\{M_6, M_7, M_8, M_9, T_3\}$

Checks $|T_3 - T_c| < \Delta T$
$M_9 = ?h(M_7 \| x_j \| T_3)$
$K'_i = M_7 \oplus h(x_j \| T_3)$
$SK = h(K'_i \oplus K_j)$
$M_{10} = h(SK \| M_6 \| M_8 \| T_3 \| T_4)$

← $\{M_6, M_8, M_{10}, T_3, T_4\}$

Checks $|T_4 - T_c| < \Delta T$
$M_8 = ?h(M_6 \| d_i \| T_3)$
$K'_j = M_6 \oplus h(d_i \| T_3)$
$SK = h(K_i \oplus K'_j)$
$M_{10} = ?h(SK \| M_6 \| M_8 \| T_3 \| T_4)$

Figure 6. Login and Authentication phase of the proposed scheme.

VI. SECURITY FEATURES COMPARISON

In Table 2, we have compared the security features of our proposed scheme with Farash et al.'s. The proposed scheme provides sensor node anonymity, password and user ID protection, mutual authentication and key agreement.

Table 2. Comparison of security features between the proposed scheme and Farash et al.'s scheme.

SECURITY FEATURES	*Farash et al.'s scheme*	*Proposed scheme*
Mutual authentication	Yes	Yes
Key agreement	Yes	Yes
Password and ID protection	No	Yes
User anonymity	Yes	Yes
Sensor node anonymity	No	Yes
Resilience against SC attack	No	Yes
Session key protection	No	Yes

VII. CONCLUSION

In this paper we propose a secure mutual authentication and key agreement scheme for HWSN tailored for the Internet of Things environment. This scheme is based on Farash et al.'s proposed scheme and consists of three fundamental phases: 1. Pre-deployment phase, 2. registration phase, 3. login and authentication phase. In this scheme, the remote user can access a single desired sensor node from the WSN without the necessity of firstly connecting with a gateway node (GWN). Moreover, this scheme is lightweight because it based on a simple symmetric cryptography and it uses simple hash and XOR computations. Although this scheme is efficient, we found out that this scheme has some shortages and it is vulnerable to some cryptographic attacks. In this paper, we demonstrate some security weaknesses of the Farash et al.'s scheme and then we propose an improved and secure mutual authentication and key agreement scheme. The proposed scheme provides mutual authentication between all parties, password and user ID protection and sensor node anonymity.

REFERENCES

[1] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, Wireless sensor networks: a survey, Computer networks, vol. 38, no. 4, 2002.

[2] L. Atzori, A. Iera, G. Morabito, The Internet of things: a survey, Computer Networks, vol. 54, no. 15, 2010.

[3] H. Duce, Internet of Things in 2020: Roadmap for the future, 2008.

[4] V Sivaraman, A Dhamdhere, H Chen, A Kurusingal, An experimental study of wireless connectivity and routing in ad hoc sensor networks for real-time soccer player monitoring, Ad Hoc Networks, vol. 11, no. 3, 2013.

[5] M Di Francesco, SK Das, G Anastasi, Data collection in wireless sensor networks with mobile elements: A survey, Transactions on Sensor Networks, vol. 8, no. 1, 2011.

[6] K. Xue, C. Ma, P. Hong, R. Ding, A temporal-credential-based mutual authentication and key agreement scheme for wireless sensor networks, Network and Computer Applications, vol. 36, no. 1, 2013.

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

[8] M. Sabzinejad Farash, M. Turkanović, S. Kumaric, M. Hölbl, An efficient user authentication and key agreement scheme for heterogeneous wireless sensor network tailored for the Internet of Things environment, Ad Hoc Networks, vol. 36, Part. 1, January 2016.

[9] H.Khemissa, D.Tandjaoui, A novel lightweight authentication scheme for heterogeneous wireless sensor networks in the context of Internet of Things, Wireless Telecommunications Symposium (WTS), 2016.

[10] T. Kothmayra, C. Schmittc, W. Hub, M. Brünigb, G. Carlea; DTLS based security and two-way authentication for the Internet of Things, Ad Hoc Networks, vol. 11, no. 8, 2013.

[11] D. Giri, T. Maitra, R. Amin, P. D. Srivastava, An Efficient and Robust RSA-Based Remote User Authentication for Telecare Medical Information Systems, Journal of Medical Systems, 39: 145. 2015.

[12] K.H.M. Wong ; Y. Zheng ; J. Cao ; Shengwei Wang, A dynamic user authentication scheme for wireless sensor networks, Sensor Networks, Ubiquitous, and Trustworthy Computing, 2006.

[13] M.L. Das, Two-factor user authentication in wireless sensor networks, Wireless Communication, vol. 8, no: 3, 2009.

[14] M.K. Khan, K. Alghathbar, Cryptanalysis and security improvements of 'two-factor user authentication in wireless sensor networks, Sensors, 2010.

[15] B. Vaidya, D. Makrakis, H.T. Mouftah, Improved two-factor user authentication in wireless sensor networks, in: Wireless and Mobile Computing, Networking and Communications, IEEE, 2010.

[16] T.H. Chen, W.K. Shih, A robust mutual authentication protocol for wireless sensor networks, vol. 32, no. 5. 2010.

[17] A. Das, Kumar, P. Sharma, S. Chatterjee, S.J. Sing, Kanta, A dynamic password-based user authentication scheme for hierarchical wireless sensor networks, Network and Computer Applications, vol. 35, no. 5, 2012.

[18] K. Xue, C. Ma, P. Hong, R. Ding, A temporal-credential-based mutual authentication and key agreement scheme for wireless sensor networks, Network and Computer Applications. Vol. 36, no. 1, 2012.

[19] M. Turkanovi´c, M. Hölbl, An improved dynamic password-based user authentication scheme for hierarchical wireless sensor networks, Electron. Electric, vol. 19, no, 6, 2013.

[20] E. D. Messerges, T.S. and R. Sloan, Examining smart-card security under the threat of power analysis attacks, IEEE Transactions on Computers, vol. 51, no. 5, 2002.