Advances in Computer Science Research, volume 44
3rd International Conference on Wireless Communication and Sensor Network (WCSN 2016)

An Enhanced Secure Anonymity User Authentication Protocol for Hierarchical Wireless Sensor Networks

Tian-Yang Pan, Li-Quan Chen, Yuan-Fang Zhang
School of information science and technology, Southeast University, Nanjing, Jiangsu, China
E-mail: lqchen@seu.edu.cn

***Abstract*-The security of hierarchical wireless sensor network (HWSN) attracts more attentions with the rapid deployment of HWSN recently. In this paper, the user authentication technology of HWSN is investigated and a more efficient and more secure user authentication protocol is proposed. Overcoming the main faults of the previous protocols, this enhanced protocol makes the improvements in achieving the strong anonymity, preventing the attackers of guessing the password of the lost smart card, and updating the secret information independently and quickly. The security of the protocol is verified by using the Rubin Logic method. The computation load is evaluated and is acceptable for the real systems.**

Keywords-hierarchical wireless sensor network (HWSN); user authentication; user anonymity; smart card; secret information updating

I. INTRODUCTION

With the rapid development of the wireless sensor network (WSN) technology, the WSN has explored applications not only in the civil infrastructures, but also in the military detection and surveillance. The security of the WSN is attracting more and more attention recently. In 2009, Das proposed the first user authentication protocol for WSN concerning two factors of the smart card and password [1]. Since then, a series of protocols based on two factors or three factors (the other one is biometrics) were proposed. In 2012, Madhusudhan, etc. suggested the user anonymity as a new security factor [2]. In Ref. [3], the user identification cannot be achieved by attackers, but the information track might be followed, which is the weak user anonymity. Until now, most protocols support only weak user anonymity. Strong user anonymity refers that tracking the user information is impossible, which is one research point in this paper.

The hierarchical WSN (HWSN) structure has higher energy efficiency, longer lifetime and more operation possibility than the distributed WSN [4]. From the top, the HWSN is divided into three layers, the gateway, the cluster head and the sensor nodes. Some protocols have been suggested on the security of HWSN [3, 5-14], in which several problems exist in the user authentication. After investigating the main technology of authentication, an enhanced secure anonymity user authentication protocol will be proposed by combing three factors of smart card, user password, and user biometrics in this paper. The security and computation load will be evaluated, too.

II. BACKGROUND

So far, many user authentication protocols of WSN have been proposed. We analyze the typical schemes and conclude the information into Table I.

There are two ways to get strong user anonymity: to update pseudo-random identification while logging in, or to adopt totally random identification. The first method has the advantage of avoiding using asymmetric cryptographic algorithm, while the disadvantage is the dependency on the accessibility to the network [5]. The user identification will be invalid if the updated secret information of the user and the gateway is non-synchronized. The second method takes advantage of asymmetric cryptographic algorithm, which has the challenge of reducing computation load as much as possible. However, the latest protocols with this method neglect the possibility of improvement [6-8].

TABLE I. ANALYSIS OF THE SECURITY IN WSN USER AUTHENTICATION SCHEMES

Protocols (in chron. order)	*Anony-mity*	*No plain text password storage*	*Pass-word protec-tion*	*Risk prevention for smart card loss*	*User cancelling and re-registration*	*Temporary session key generation*	*Two-way authenti-cation*	*With biometrics authenti-cation*	*Anti-replay attack*	*Security recovery*
[13]	Weak	√	√	×	×	×	√	×	√	Weak
[14]	Weak	√	×	×	√	√	×	×	√	Weak
[9]	Weak	√	×	×	×	×	√	×	√	×
[10]	Weak	√	√	×	×	×	√	×	√	×
[6]	×	√	√	×	×	√	×	×	√	×
[11]	Weak	√	√	×	√	√	√	×	√	×
[3]	Weak	√	√	×	×	√	√	×	×	×
[12]	×	√	√	×	×	√	√	×	√	×
[7]	×	√	√	×	×	√	√	×	√	×
[8]	×	√	√	×	×	√	√	×	√	×
[5]	Weak	√	√	√	×	√	√	×	√	×

The typical schemes of the published user authentication protocols of WSN are analyzed and the problems are classified into three types. The first one is the weakness or none of user anonymity [3, 5-14], which would cause being tracked easily. The second one is information leakage of the lost smart card [3, 6-14], which causes the attacker can do guesswork in a smaller space after extracting the information in the smart card. The third one is security recovery when system secret information is revealed [3, 5 -12], in which case, the system cannot update secret information conveniently because all users must register again. Our works would concentrate on solving these problems and propose an effective and safe user authentication scheme for HWSN.

III. PROPOSED PROTOCOL

To improve the security and overcome the faults of previous protocols, we propose an Enhanced Secure Anonymity User Authentication protocol (ESA-UA). The notations used in the protocol are listed in Table II. Our protocol consists of three phases: preparation phase, register phase, and authentication phase. In addition, we also provide the method of secure information update.

A. Preparation Phase

The sensor network is set up in this phase, with information loaded and contact key generated.

- Information pre-loaded on CH_j: ID_{CH_j}, r_{CH_j}, X_{CH_j}, and all ID_{S_j} it controls.
- Information pre-loaded on S_j : ID_{S_j} , ID_{CH_j} , authentication info $H_1(X_{CH_j} \| ID_{S_j})$.
- Setup contact key $SK_{CH_jS_j}$ between CH_j and S_j.

B. Register Phase

A user can access an HWSN system by following operations.

- Extract biometrics to get (σ_i, ϑ_i) . Select own ID_i&PW_i, and Compute $h(\sigma_i \| PW_i)$.
- Compute $MPW_i = PW_i \bmod n$, in which n is allowed to be altered with preference.
- Compute authentication info $A_i = h(h(ID_i) \| h(\sigma_i \| MPW_i))$, and save it in smart card.
- Send info $\{ID_i, h(\sigma_i \| PW_i)\}$ to GW.
- The GW generates a random number r_i and symmetrically encrypt r_i.
- Preserve outcome and the IDs of the user's authorized cluster nodes in user database.
- Compute $R_i = h(ID_i \| h(\sigma_i \| PW_i)) \oplus r_i$, and save R_i, $H_2(\cdot)$, and $H_3(\cdot)$ in smart card.

C. Authentication Phase

GW Operates following procedures after it receives the information from CH_j.

1) Authenticate CH node

- Confirm $T - T_2 < \Delta T$ to guarantee the time.
- Compute $Y' = kY, ID_{CH_j} = CID_{CH_j} \oplus h(Y')$, and search for the corresponding authentication info from the database.
- Compute the key $H_0(K_{BS_{CH}} \| ID_{CH_j})$, and decrypt r_{CH_j}.
- Calculate $Hash_2^*$ to compare with $Hash_2$, continue only if they are the same.

2) Authenticate the user

Compute $X' = kX = k\alpha P$, and get the user's $ID_i = CID_i \oplus h(X')$ with sensor node's $ID_{S_j} = h(X' \oplus h(X')) \oplus CID_{S_j}$.

- Retrieve user's ID and privilege from the database.
- Continue if ID_{CH_j} is included in user's privilege.
- Compute key $H_0(K_{BS_U} \| ID_i)$, and decrypt r_i.
- Compute$Hash_1^* = H_2(ID_i \| ID_{S_j} \| ID_{CH_j} \| X \| X' \| r_i \| T_1)$. Authentication succeeds if it equals $Hash_1$.

TABLE II. ELEMENTS USED IN THE PROTOCOL

Notation	*Description*
U_i	User
CH_j	Cluster header node
S_j	Basic sensor node
GW	Gateway node
P	A point on ellipse curve E
k	Private key of GW
P_{Pub}	Public key of GW
PW_i	Password of user
ID_i	ID of user
σ_i	Extracted information of biometrics
ϑ_i	Auxiliary info of biometrics
r_i	Random number shared by U and GW
ID_{CH_j}	Identification of cluster head node
X_{CH_j}	Secret information shared by CH and S
r_{CH_j}	Random number shared by S and GW
ID_{S_j}	ID of sensor node
$h(\cdot)$. $H_n(\cdot)$	Single-way hash function
$E_K()$	Symmetric encrypt with key K
$D_K()$	Symmetric decrypt with key K
K_{BS_U}	Key of user database at server
$K_{BS_{CH}}$	Key of CH node database
$A \oplus B$	A xor B
$A \| B$	Connect A&B in series
T	Time stamp

3) Generate reply info

- Compute $TID_i = h(X' \| r_i) \oplus ID_i$, $TID_{CH_j} = h(Y' \| r_{CH_j}) \oplus ID_{CH_j}$, and $TID_{S_j} = h(r_{CH_j} \| Y') \oplus ID_{S_j}$.
- Compute $Hash_3 = H_3(ID_{CH_j} \| ID_i \| ID_{S_j} \| X \| X' \| Y \| r_i)$.
- Record time stamp T_3 , and compute $Hash_4 = H_3(ID_{CH_j} \| TID_i \| Hash_3 \| X \| Y \| r_{CH_j} \| T_3)$.
- Send info $\{TID_i, TID_{CH_j}, TID_{S_j}, Hash_3, Hash_4, T_3\}$ back to ID_{CH_j}. When CH_j receives the info, confirm that $T - T_3 < \Delta T$.
- Assure $ID_{CH_j} = h(Y' \| r_{CH_j}) \oplus TID_{CH_j}$, $ID_{S_j} = h(r_{CH_j} \| Y') \oplus TID_{S_j}$.
- Compute $Hash_4^*$. If it equals $Hash_4$, continue to authenticate. Record time stamp T_4 . Calculate $K_{CH_jU_i} = \beta X$ and authentication info $Hash_5 = H_3(ID_{CH_j} \| TID_i \| X \| Y \| K_{CH_jU_i} \| Hash_3 \| T_4)$. Send $\{TID_i, Y, Hash_3, Hash_5, T_4\}$ to U_i.

4) Communicate with S_j

- Record time stamp T_5.
- Calculate the session key between node and user $K_{S_jU_i} = h(K_{CH_jU_i} \| ID_{S_j})$.
- Compute the authentication info $X_S = H_1(X_{CH_j} \| ID_{S_j})$, $Hash_6 = H_1(ID_{CH_j} \| ID_{S_j} \| K_{S_jU_i} \| X_S \| T_5)$, and temporary session key $SK_{CH_jS_j}' = H_1(SK_{CH_jS_j} \| T_5)$ to encrypt.
- Send info $\{ID_{S_j}, E_{SK_{CH_jS_j}'}(ID_{CH_j} \| ID_{S_j} \| K_{S_jU_i} \| Hash_6), T_5\}$ to sensor node.

5) When U_i Receives the info

- Confirm that $T - T_4 < \Delta T$.
- Calculate $ID_i = h(X' \| r_i) \oplus TID_i$. Compute $Hash_3^*$ after confirmation, and only if $Hash_3^* = Hash_3$, continue.
- Compute $K_{CH_jU_i} = \alpha Y$. Calculate $Hash_5^*$.
- If $Hash_5^* = Hash_5$, authentication is successful, and the communication key is $K_{S_jU_i} = h(K_{CH_jU_i} \| ID_{S_j})$.

6) When S_j Receives the info

- Confirm that $T - T_5 < \Delta T$.
- Calculate $SK_{CH_jS_j}' = H_1(SK_{CH_jS_j} \| T_5)$ and decrypt to get $(ID_{CH_j} \| ID_{S_j} \| K_{S_jU_i} \| Hash_6)$.
- Compute $Hash_6^*$ and confirm $Hash_6^* = Hash_6$.
- Use $K_{S_jU_i}$ as the temporary session key, and start to communicate with U_i.

D. Secure Information Update

The leakage of secret information will be fatal to the user and system. It is necessary to update the secret information periodically and instantly as follows.

- Revision of user password. Revise locally without connecting to the server.
- User/CH node authentication info update. User renew the old information r_i to $r_i' = h(r_i \oplus X')$. Then compute new R_i' and delete obsolete R_i.
- Revision of the private key in GW node. Update the private key k independently without updating user authentication info, CH node info and the info in database.
- Revision of the key of the database in which preserves authentication info. Decrypt the updated authentication info and use new keys K_{BS_U}' and $K_{BS_{CH}}'$ to generate new keys for user and CH node, and encrypt again.

By this procedure, the protocol can update the secret information independently, which enhances the system safety if the secret information is revealed.

IV. PERFORMANCE ANALYSIS

The security of the protocol will be investigated by the Rubin logic method and the computation load will be evaluated and compared with the published protocols in this section.

A. Security Analysis

With Rubin Logic [15], the security of the proposed protocol could be verified efficiently. After a series of logical analysis of classified combinations, including global sets, local sets and flow sets, and comparing to the published protocol, it is validated that this proposed ESA-UA has three enhanced security characteristics as follows.

1) *Strong user anonymity.*

The true user identity is camouflaged by $CID_i = h(X') \oplus ID_i^*$ and $TID_i = h(X' \| r_i) \oplus ID_i$. Only the involvers who know X' and r_i can get the user's identity because all the authentication information is processed by the hash function.

2) *Resistance of smart card extraction attack.*

Our protocol has the ability of changing the password locally and avoiding the attacker guessing the password by the password space $\mathcal{D}_{\mathcal{PW}}$. The identification of the user can be filtered by A_i, which utilizes modulus value of the password. The attacker could be easily detected by the server because the attacker must try many times within the password space $\mathcal{D}_{\mathcal{PW}}/n$.

3) *Secret information updating.*

User authentication info r_i ,password PW_i , server's private key k, and database's key K_{BS_U}, are all allowed to be updated to meet the security requirement of the system. It should be noticed that the published protocols require the user to register again when faced with updating these information.

B. Efficiency Analysis

In the authentication process of the ESA-UA, the computation efficiency is mainly divided into four parts: the user, the sensor node, the cluster head node, and the gateway node sides.

In a single authentication process, the user side requires 12 Hash operations and 3 elliptic curve dot product operations, the sensor node side need 2 Hash operations and 1symmetrical decryption operation. The computation load at the cluster head node is 10 Hash operations, 1 symmetric encryption operations and 3 dot product on ellipse curve. The computation load at gateway node includes 12 hash and 2 dot product operations. Computing resources used in Hash function and the symmetric calculation are much less than those in the asymmetric cryptic computation, so the latter is mainly considered.

From [16], the computation time for a single dot product on the elliptic curve is 0.21s on a chip of 20.57MHz Philips HiPerSmartTM smart card. At the user side, the total computation time is thus about 0.63s and will not exceed 0.7s even if the Hash operation is included.

The software of TinyECC is used to encrypt. If the system uses Imote2 node with 104MHz basic frequency as the cluster node, the computation time of a single dot product with Tiny ECC is 0.05s and three dot products cost about 0.15s. For a sensor node, the computation load is low because only one symmetric decryption and 2 hash operations are operated. The gateway node can afford the computation load due to its high computing power.

Table III compares the computation load between our protocol and others that also use ECC. The ECC, Hash and symmetric encrypt/decrypt operations are represented as M, H, and E, respectively. The computation load of an ECC operation is much higher than that of a Hash one, so "M" is listed solely if and only if there is an ECC operation.

TABLE III. COMPUTATION LOAD COMPARISON

Protocols	***User***	***Cluster head node***	***Sensor node***	***Gateway node***
Protocol[3]	3M	2M	-	2M
Protocol[6]	2M	-	3M	3M
Protocol[7]	3M	-	2M	1M
Protocol[8]	3M	-	2M	1M
ESA-UA (this work)	3M	3M	2H+E	2M

From Table III, the computation load of the sensor node of the ESA-UA is less than those of the published protocols although that of the cluster head node is higher. The user's computation loads of these protocols are nearly at the same level. Considering the strong anonymity of this proposed ESA-UA, the little increased computation load is acceptable.

V. CONCLUSION

This paper proposed a three-factor enhanced secure anonymity user authentication protocol (ESA-UA) for HWSN. Compared with previous protocols, our protocol has three advantages of the strong anonymity, user information defense and secure information updating. With the strong anonymity, the user's identification would not be revealed, and the login track cannot be followed. With the user information defense, the attacker could not decrypt user's secret information by guessing attack or analyzing previous information even if user's smart card is lost and the information inside is extracted. With the ability to update secure information, all the secure information could be updated separately and independently without interfering the user's normal use.

The protocol's security has been validated by the Rubin Logic method. The computation load of a sensor node is very low while that of a cluster head node is a little bit higher, so the whole computation load is relatively light. This proposed protocol is suitable for the HWSN systems that require higher security.

ACKNOWLEDGMENT

This work is supported by the National Science Foundation of China (61372103).

REFERENCES

[1] M.L. Das. Two-factor user authentication in wireless sensor networks, IEEE Trans. Wireless Commun. 8 (3): 1086-1090 (2009).

[2] R. Madhusudhan, R. Mittal, Dynamic id-based remote user password authentication schemes using smart cards: a review, J. Network Comput. Appl. 35(4): 1235-1248 (2012).

[3] R. Maharana, An improved user authentication protocol for hierarchical wireless sensor networks using Elliptic curve cryptography, Master's Thesis, National Institute of Technology Rourkela, India, (2013).

[4] W. Zhang, B. Chen, M. Chen, Hierarchical Fusion Estimation for Clustered Asynchronous Sensor Networks, IEEE Trans. Automatic Control, 61 (10): 3064-3069, (2016).

[5] D. He, N. Kumar, J. Chen, C. Lee, N. Chilamkurti, S. Yeo, Robust anonymous authentication protocol for health-care applications using wireless medical sensor networks, Multimedia Systems 21: 49-60 (2015).

[6] H. Yeh, and T. Chen, et al, A secured authentication protocol for wireless sensor networks using elliptic curves cryptography. Sensors 11 (5): 4767-4779 (2011).

[7] Y. Choi, D. Lee, J. Kim, J. Jung, J. Nam, D. Won. Security enhanced user authentication protocol for wireless sensor networks using elliptic curve cryptography, Sens. 14: 10081-10106 (2014).

[8] A.K. Maurya, V.N Sastry, S.K. Udgata. Cryptanalysis and improvement of ECC- based security enhanced user authentication protocol for wireless sensor networks. SSCC 2015, CCIS 536: 134-145 (2015).

[9] T.H. Chen, W.K. Shih. A robust mutual authentication protocol for wireless sensor networks. ETRI J. 32 (5): 704-712 (2010).

[10] D. He and Y. Gao, et al, An enhanced two-factor user authentication scheme in wireless sensor networks. Ad Hoc Sensor Wireless Networks 10(4): 361-371 (2010).

[11] K. Xue, C. Ma, P. Hong, R. Ding, A temporal-credential-based mutual authentication and key agreement scheme for wireless sensor networks. J. Network Comput. Appl. 36 (1): 316-323 (2013).

[12] A.K. Das and P. Sharma, et al, A dynamic password based user authentication scheme for hierarchical wireless sensor networks. J. Network Comput. Appl. 35 (52): 1646-1656 (2012).

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

[14] C.Hu, H.Li; Y.Huo, T. Xiang, X. Liao, Secure and Efficient Data Communication Protocol for Wireless Body Area Networks, IEEE Trans. Multi-Scale Computing Systems, 2(2): 97-107, (2016).

[15] A. D. Rubin, P. Honeyman. Nonmonotonic cryptographic protocols, Proceedings of the Computer Security Foundations Workshop VII, (CSFW 7): 100-116(1994).

[16] M. Scott, N. Costigan, W. Abdulwahab. Implementing Cryptographic Pairings on Smartcards, International Association for Cryptologic Research 2006, CHES 2006, LNCS 4249: 134-147(2006).