Design issues of "an anonymous authentication and key agreement protocol in smart living"

Zhengjun Cao, Lihua Liu

Abstract. The Li et al.'s scheme [Computer Communications, 186 (2022), 110-120)] uses XOR operation to realize the private transmission of sensitive information, under the assumption that if only one parameter in the expression $a = b \oplus c$ is known, an adversary cannot retrieve the other two. The assumption neglects that the operands b and c must be of the same bit-length, which leads to the exposure of a substring in the longer operand. The scheme wrongly treats timestamps as random strings to encrypt a confidential parameter. These misuses result in the loss of sensor node's anonymity, the loss of user anonymity and untraceability, insecurity against off-line password guessing attack, and insecurity against impersonation attack. The analysis techniques developed in this note is helpful for the future works on designing such schemes.
Keywords: Authentication; Anonymity; Key agreement; Impersonation attack

1 Introduction

A wireless sensor network (WSN) usually consists of tens to thousands of wireless sensor nodes that communicate through wireless channels for information sharing and cooperative processing [12]. It can be deployed on a global scale for environmental monitoring and habitat study, in factories for condition based maintenance, in buildings for infrastructure health monitoring, in homes to realize smart homes, in bodies for patient monitoring, etc [3, 11, 13]. A wireless sensor node consists of sensing, computing, communication, actuation, and power components [5]. These components are integrated on a single or multiple boards. The security and privacy of WSNs have gained much attention [1, 9]. In 2023, Tyagi and Kumar [14] presented a multi-factor user authentication and key agreement scheme for WSNs using Chinese remainder theorem. Darbandeh and Safkhani [6] proposed a secure authentication protocol for WSNs based on RFID tags. Khah et al. [7] suggested a dynamic and multi-level key management method for WSNs. Chen et al. [4] designed a provably-secure authenticated key agreement protocol for remote patient monitoring systems.

In 2022, Li et al. [8] proposed an anonymous authentication and key agreement protocol in smart living. The scheme uses only very low-cost bit-wise operation and hashing in order to realize the private transmission of sensitive information. The scheme treats timestamps as random strings to encrypt a confidential parameter, and also views a user's identity as a random string. In this note, we show that the scheme has some design issues, including the loss of sensor node's anonymity, the loss of user anonymity and untraceability, insecurity against off-line password guessing attack, and

Z. Cao is with Department of Mathematics, Shanghai University, Shanghai, 200444, China.
L. Liu is with Department of Mathematics, Shanghai Maritime University, Shanghai, 201306, China.
Email: liulh@shmtu.edu.cn

insecurity against impersonation attack. To the best of our knowledge, it is first time to develop these analysis techniques for such authentication and key agreement protocols.

2 Review of the Li et al.'s scheme

In the considered scenario, there are three parties: users, gateway, and sensor nodes. Let $h(\cdot)$ be a hash function with the output length 256 bits. The scheme depends mainly on XOR encryption. The involved notations are listed below (Table 1). The scheme is designed to meet many security requirements, including users' anonymity and untraceability, sensor nodes' anonymity, forward security and backward security, resisting replay attack, stolen smart card attack, impersonation attack, off-line password guessing attack, and insider attack.

The administrator stores some fundamental operation functions in the memory of smart card SC, gateway node GWN, and sensor node N_j. Then, the administrator selects an identity ID_{SC} and a random number R_{SC} for the smart card SC, and stores $\{ID_{SC}, R_{SC}\}$ to the user's authentication table and SC's memory. The registration phase can be depicted as below (Table 2).

Table 1: Notations and descriptions

GWN, X	gateway node, and its master key
U_i, ID_i	the ith user, and its identity
N_j, ID_j	the jth sensor node, and its identity
SC, ID_{SC}	smart card, and its identity
PW_i	U_i's password
TS_i	time stamp
$\triangle T$	tolerable transmission delay
$\|$	concatenation operator
$\oplus$	XOR operator
$R_U, R_{GWN}, R_N, R_{SC}$	random number generated by user, gateway, sensor node, and smart card, respectively

Table 2: The user registration phase

U_i: $SC = \{ID_{SC}, R_{SC}, h()\}$		GWN: $X, h(), \{ID_{SC}, R_{SC}\}$
Choose ID_i, PW_i.		
Pick a nonce R_U.	$\xrightarrow[\text{[open channel]}]{ID_{SC},\ RPW_i,\ REG_i}$	Check $\{ID_{SC}, R_{SC}\}$. Generate
Compute $RPW_i = h(PW_i \| R_U)$,		the timestamp TS_i. Compute
$REG_i = ID_i \oplus h(R_{SC} \| RPW_i)$.		$ID_i = REG_i \oplus h(R_{SC} \| RPW_i)$,
		$US = h(ID_i \| X)$,
		$SE = TS_i \oplus h(US \| ID_i)$,
		$UR = US \oplus h(ID_i \| RPW_i)$,
Compute	$\xleftarrow{RSP_i}$	$UV = h(ID_i \| US \| RPW_i) \oplus TS_i$,
$(UR \| SE \| UV) = RSP_i \oplus h(ID_i \| R_{SC})$,		$RSP_i = h(ID_i \| R_{SC}) \oplus (UR \| SE \| UV)$,
$RE = R_U \oplus h(ID_i \| PW_i)$.		$XT_i = h(X \| TS_i) \oplus ID_i$.
Update $SC = \{UR, SE, UV, RE, h()\}$.		Store $\{XT_i, TS_i\}$.

Table 3: The authentication and key agreement phase

U_i knows ID_i^*, PW_i^* and has $SC = \{SE, RE, UR, UV, h()\}$.	GWN knows the master key X and $\{ID_j, SV_j\}$, $\{XT_i, TS_i\}$.	N_j knows $\{ID_j, SV_j\}$.
Insert SC into a terminal and input ID_i^*, PW_i^*, ID_j. The card SC computes $R_U^* = RE \oplus h(ID_i^* \Vert PW_i^*)$, $RPW_i^* = h(PW_i^* \Vert R_U^*)$, $US^* = UR \oplus h(ID_i^* \Vert RPW_i^*)$, $TS_i^* = SE \oplus h(US^* \Vert ID_i^*)$, $UV^* = h(ID_i^* \Vert US^* \Vert RPW_i^*) \oplus TS_i^*$. Check $UV = UV^*$. If true, then $ID_i^* = ID_i, PW_i^* = PW_i$. Choose a new timestamp T_1, compute $M_1 = (ID_j \Vert R_U) \oplus h(US \Vert ID_i \Vert TS_i)$, $M_2 = h(ID_i \Vert TS_i \Vert US \Vert T_1 \Vert ID_j)$. $\xrightarrow{M_1,\ M_2,\ T_1,\ TS_i}$	Check the timestamp. Retrieve XT_i by using TS_i. Compute $ID_i' = h(X \Vert TS_i) \oplus XT_i$, $US' = h(ID_i' \Vert X)$, $ID_j' \Vert R_U' = M_1 \oplus h(US' \Vert ID_i' \Vert TS_i)$, $M_2' = h(ID_i' \Vert TS_i \Vert US' \Vert T_1 \Vert ID_j')$. Check $M_2 = M_2'$. If so, generate TS_i^{new}, R_{GWN}, T_2. Retrieve SV_j' by using ID_j'. Compute $c = TS_i \oplus TS_i^{new}$, $M_3 = (R_U' \Vert R_{GWN}) \oplus h(SV_j')$, $M_4 = h(ID_j' \Vert R_{GWN} \Vert T_2 \Vert SV_j')$. $\xrightarrow{M_3,\ M_4,\ T_2}$	
		Check the timestamp. Compute $R_U'' \Vert R_{GWN}' = M_3 \oplus h(SV_j)$, $M_4' = h(ID_j \Vert R_{GWN}' \Vert T_2 \Vert SV_j)$. Check $M_4' = M_4$. If so, generate R_N, T_3, and compute $M_5 = R_N \oplus h(SV_j)$, $SK = h(R_U'' \Vert R_{GWN}' \Vert R_N)$, $M_6 = h(SK \Vert SV_j \Vert R_N \Vert T_3 \Vert ID_j)$. $\xleftarrow{M_5,\ M_6,\ T_3}$
	Check the timestamp. Compute $R_N' = M_5 \oplus h(SV_j)$, $SK' = h(R_U' \Vert R_{GWN} \Vert R_N')$, $M_6' = h(SK' \Vert SV_j' \Vert R_N' \Vert T_3 \Vert ID_j')$. Check $M_6' = M_6$. If so, choose T_4 to compute $M_7 = c \oplus US'$, $M_8 = (R_{GWN} \Vert R_N') \oplus h(R_U' \Vert US')$, $M_9 = h(SK' \Vert ID_i' \Vert TS_i^{new} \Vert US' \Vert T_4)$. $\xleftarrow{M_7,\ M_8,\ M_9,\ T_4}$	
Check the timestamp. Compute $c' = M_7 \oplus US, TS_i^{new1} = c' \oplus TS_i$, $R_{GWN}' \Vert R_N'' = M_8 \oplus h(R_U \Vert US)$, $SK'' = h(R_U \Vert R_{GWN}' \Vert R_N'')$, $M_9' = h(SK'' \Vert ID_i \Vert TS_i^{new1} \Vert US \Vert T_4)$. Check $M_9' = M_9$. If so, generate a nonce R_U^{new}, compute $SE^{new} = SE \oplus c'$, $RE^{new} = RE \oplus R_U \oplus R_U^{new}$, $RPW_i^{new} = h(PW_i \Vert R_U^{new})$, $UR^{new} = US \oplus h(ID_i \Vert RPW_i^{new})$, $UV^{new} = h(ID_i \Vert US \Vert RPW_i^{new}) \oplus TS_i^{new}$. $SC = \{SE^{new}, RE^{new}, UR^{new}, UV^{new}, h()\}$.		

3 The flaws in Li et al.'s scheme

Though the Li et al.'s authentication and key agreement scheme [8] is interesting, we find it has some flaws, including the misuse of XOR operation, the loss of sensor node's anonymity, the loss of user untraceability, insecurity against stolen smart card attack, insecurity against off-line password guessing attack, and insecurity against impersonation attack.

3.1 The misuse of XOR operation

The Boolean logic operation XOR, denoted by $\oplus$, is widely used in cryptography which compares two input bits and generates one output bit [10]. If the bits are the same, the result is 0. If the bits are different, the result is 1. When the operator is performed on two strings, they must be of the same bit-length. Otherwise, the shorter string is usually stretched by padding some 0s to its left side. In this case, the partial string corresponding to the padding bits is eventually exposed to the adversary.

In the user registration phase, it specifies that

$$\begin{aligned}
\text{Encryption:} \quad & \underbrace{RSP_i}_{\text{ciphertext}} = \underbrace{h(ID_i\|R_{SC})}_{\text{secret key}} \oplus \underbrace{(UR\|SE\|UV)}_{\text{plaintext}} \\
\text{Decryption:} \quad & (UR\|SE\|UV) = RSP_i \oplus h(ID_i\|R_{SC}) \\
\text{where} \quad & UV = h(ID_i\|US\|RPW_i) \oplus TS_i
\end{aligned}$$

Actually, we have

$$\begin{aligned}
RSP_i &= \underbrace{h(ID_i\|R_{SC})}_{\text{256 bits}} \oplus (UR\|SE\| \underbrace{h(ID_i\|US\|RPW_i) \oplus TS_i}_{\text{256 bits}}) \\
&= UR\|SE\| \underbrace{(h(ID_i\|R_{SC}) \oplus h(ID_i\|US\|RPW_i) \oplus TS_i)}_{\text{256 bits}}
\end{aligned}$$

That means the substring $UR\|SE$ is entirely copied into the resulting string RSP_i. The adversary who has eavesdropped on the open channel and obtained the string, can successfully retrieve the parameters UR, SE. The confidential parameters are eventually exposed.

3.2 The loss of sensor node's anonymity

As for the sensor node's anonymity, it argues that: *" In our protocol, the real identity ID_j of the sensor node does not explicitly exist in any communication messages, so the adversary cannot directly obtain the sensor's ID_j according to the communication messages on the public channel. Furthermore, the adversary cannot compute $ID_j\|R_U = M_1 \oplus h(US||ID_i\|TS_i)$ without knowing US and the user's real identity ID_i."* We find the argument is nor sound. Actually, in the authentication and key agreement phase, it specifies that

$$\begin{aligned}
\text{Encryption:} \quad & \underbrace{M_1}_{\text{ciphertext}} = \underbrace{(ID_j\|R_U)}_{\text{plaintext}} \oplus \underbrace{h(US\|ID_i\|TS_i)}_{\textit{nonce}}, \\
\text{Decryption:} \quad & ID_j\|R_U = M_1 \oplus h(US||ID_i\|TS_i), \\
\text{where} \quad & R_U = RE \oplus h(ID_i\|PW_i).
\end{aligned}$$

Hence, we have

$$\begin{aligned}
M_1 &= (ID_j\| \underbrace{RE \oplus h(ID_i\|PW_i)}_{\text{256 bits}}) \oplus \underbrace{h(US\|ID_i\|TS_i)}_{\text{256 bits}} \\
&= ID_j\|(RE \oplus h(ID_i\|PW_i) \oplus h(US\|ID_i\|TS_i))
\end{aligned}$$

which means the sensor node's identity ID_j is entirely copied into the string M_1. An adversary who has captured the data via the open channel, can easily recover the identity ID_j.

3.3 The loss of user untraceability

The user untraceability says that an adversary cannot trace a target user in different sessions. As for this property, it argues that: *"Since each user accesses the gateway irregularly and new users register to the gateway, there is no connection between the dynamic sequence TS_i used in this session and TS_i^{new} used in the next session. Besides, the communication messages $\{M_1, M_2, T_1, TS_i\}$ are different since the user uses different random number R_U in each session."* We find the argument is not sound. In fact,

$$c = TS_i \oplus TS_i^{new}, \quad US = h(ID_i \| X), \quad M_7 = c \oplus US$$

where ID_i is the user's identity, X is the GWN's master key. Note that *the timestamps TS_i and TS_i^{new} cannot be viewed as two random strings*, which are publicly available to any adversary. So, the adversary can obtain c. Using the captured M_7 via the open channel, the adversary can obtain $h(ID_i\|X) = US = c \oplus M_7$. The hash value $h(ID_i\|X)$ is invariable for different sessions for the same user, because the identity ID_i and the master key X are two long-term parameters in the proposed scheme. Although the adversary cannot recover the identity ID_i from the hash value $h(ID_i\|X)$ due to its one-way property, he can trace a target user by checking the consistency of this hash value. In fact, for a different identity $\widehat{ID}$, the probability of event that $h(ID_i\|X) = h(\widehat{ID}\|X)$ is negligible, due to the collision-free property of the hash function.

3.4 The loss of user anonymity

The user anonymity means that an adversary cannot obtain a user's real identity. But we find the scheme fails to keep this property. In fact, the adversary can capture the parameters M_1, M_2, T_1, TS_i via the open channel. By the above analysis (see §3.2, §3.3), we know, the adversary can also recover ID_j and US.

In order to launch a session, the user ID_i needs to inquire about the target sensor node's identity ID_j. If the identity ID_j is not publicly available, the user cannot complete the later procedure. We want to stress that a user's identity is the characteristics that *distinguish* it from others. So, it is publicly available for the purpose of distinguishing members, not a confidential parameter [2].

Let Υ be the set of all identities in the system. Since $M_2 = h(ID_i||TS_i\|US\|T_1\|ID_j)$, the adversary can test the following equation

$$M_2 = h(\psi||TS_i\|US\|T_1\|ID_j), \quad \forall\ \psi \in \Upsilon \tag{1}$$

to determine which ψ is the target identity. Practically, the size of Υ is moderate, and the success probability of this exhaust search attack is not negligible. All in all, the scheme [8] has falsely treated the identity ID_i as a random string.

3.5 Insecurity against off-line password guessing attack

When an adversary gets a legitimate user's smart card $SC = \{SE, RE, UR, UV\}$, he acquires the parameters stored in it. In this case, the adversary can launch the off-line password guessing attack.

Concretely, he first recovers the user's identity ID_i and the parameter US by the above analysis (see §3.4). He then makes use of the following relations

$$R_U = RE \oplus h(ID_i \| PW_i),$$
$$RPW_i = h(PW_i \| R_U),$$
$$US = UR \oplus h(ID_i \| RPW_i),$$

to construct the challenging equation

$$US = UR \oplus h(ID_i \| h(\phi \| RE \oplus h(ID_i \| \phi))), \forall\, \phi \in \Phi \tag{2}$$

where US, UR, ID_i, RE have been recovered, and Φ is the given password dictionary. Once a password satisfies the Eq.(2), the original password or an equivalent password is searched out.

3.6 Insecurity against impersonation attack

By the sections §3.4 and §3.5, we know, an adversary who has captured the smart card SC can recover the target user's identity ID_i and password PW_i. Having the two confidential parameters, the adversary can impersonate the user to start other sessions. Therefore, the scheme is insecure against impersonation attack.

4 Conclusion

We show that the Li et al.'s authentication and key agreement scheme has some flaws. It seems difficult to fix the scheme because of its simple encryption mechanism. The findings in this note could be helpful for the future work on designing such schemes.

References

[1] P. Alimoradi, A. Barati, and H. Barati. A hierarchical key management and authentication method for wireless sensor networks. *Int. J. Commun. Syst.,*, 35(6), 2022.

[2] Z. Cao. A note on "efficient provably-secure dynamic ID-based authenticated key agreement scheme with enhanced security provision". *IEEE Trans. Dependable Secur. Comput.,*, doi: 10.1109/TDSC.2023.3302300, 2023.

[3] C. M. Chen, L. Chen, Y. Huang, S. Kumar, and J. M. T. Wu. Lightweight authentication protocol in edge-based smart grid environment. *EURASIP J. Wirel. Commun. Netw.,*, 2021(1):68, 2021.

[4] C. M. Chen, S. Liu, X. Li, SK H. Islam, and A. K. Das. A provably-secure authenticated key agreement protocol for remote patient monitoring iomt. *J. Syst. Archit.,*, 136:102831, 2023.

[5] M. L. Chiang, H. C. Hsieh, T. L. Lin, T. P. Chang, and H. W. Chen. Dynamic weight-based connectivity recovery in wireless sensor and actor networks. *J. Supercomput.,*, 80(1):734–760, 2024.

[6] F. G. Darbandeh and M. Safkhani. SAPWSN: a secure authentication protocol for wireless sensor networks. *Comput. Networks,*, 220:109469, 2023.

[7] S. A. Khah, A. Barati, and H. Barati. A dynamic and multi-level key management method in wireless sensor networks (WSNs). *Comput. Networks,*, 236:109997, 2023.

[8] F. Li, X. Yu, Y. Cui, S. Yu, Y. Sun, Y. Wang, and H. Zhou. An anonymous authentication and key agreement protocol in smart living. *Comput. Commun.,*, 186:110–120, 2022.

[9] M. Malik, K. Gandhi, and B. Narwal. AMAKA: anonymous mutually authenticated key agreement scheme for wireless sensor networks. *Int. J. Inf. Secur. Priv.,*, 16(1):1–31, 2022.

[10] A. Menezes, P. Oorschot, and S. Vanstone. *Handbook of Applied Cryptography.* CRC Press, USA, 1996.

[11] M. Raja, T. Koduru, and R. Datta. Protecting source location privacy in IoT-enabled wireless sensor networks: The case of multiple assets. *IEEE Internet Things J.,*, 9(13):10807–10820, 2022.

[12] P. Roy, A. K. Tripathy, S. Singh, and K. C. Li. Recent advancements in privacy-aware protocols of source location privacy in wireless sensor networks: A survey. *Comput. Sci. Inf. Syst.,*, 19(2):857–886, 2022.

[13] T. Sampradeepraj, V. A. Devi, and S. P. Raja. Secure multicasting in wireless sensor networks using identity based cryptography. *Concurr. Comput. Pract. Exp.,*, 35(1), 2023.

[14] G. Tyagi and R. Kumar. Multi-factor user authentication and key agreement scheme for wireless sensor networks using Chinese remainder theorem. *Peer Peer Netw. Appl.,*, 16(1):260–276, 2023.