# A note on "ISG-SLAS: secure and lightweight authentication and key agreement scheme for industrial smart grid using fuzzy extractor"

Zhengjun Cao[1], Lihua Liu[2]

**Abstract**. We show that the key agreement scheme [J. Syst. Archit., 131:102698, 2022] fails to keep user anonymity and service provider anonymity, not as claimed. The scheme simply thinks that user anonymity is equivalent to protecting the target user's identity against exposure, while its long-term pseudo-identity can be exposed. We want to clarify that the true anonymity means that an adversary cannot attribute different sessions to different target users, even if the true identifier cannot be retrieved from the exposed pseudo-identifier.
**Keywords**: Authentication, Key agreement, Anonymity, Pseudo-identity, Fuzzy extractor

## 1 Introduction

The smart grid moves the energy industry into a new era of reliability, availability, and efficiency [1–3]. Its benefits include: more efficient transmission of electricity, quicker restoration of electricity after power disturbances, reduced operations and management costs for utilities, ultimately lower power costs for consumers [4], reduced peak demand, improved security [5, 6], etc.

Recently, Yu and Park [7] have presented a key agreement scheme for smart grid network, in which there are three entities: user, service provider (SP), and a trusted authority (TA). TA is responsible for the registration of all participants, and provides the necessary parameters and secret credentials to all participants. SP monitors and manages real-time data from smart meters, and provides useful smart grid services. An authorized user by TA collects the electricity usage information and transmits the information to SP over a public channel. Though the scheme is interesting, we find it flawed because it fails to keep user anonymity and service provider anonymity.

## 2 Review of the scheme

Given a biometric of user input $BIO_i$, $Gen(\cdot)$ chooses a biometric secret data $\sigma_i$ and a public reproduction parameter $\tau_i$, such that $Gen(BIO_i) = (\sigma_i, \tau_i)$. Given a noisy biometric of user input $BIO_i$, $Rep(\cdot)$ reproduces $\alpha_i$ using public a reproduction $\beta_i$, such that $Rep(BIO_i, \tau_i) = \sigma_i$. The biometric authentication is performed using the fuzzy extractor.

[1]Department of Mathematics, Shanghai University, Shanghai, 200444, China
[2]Department of Mathematics, Shanghai Maritime University, Shanghai, 201306, China.
Email: liulh@shmtu.edu.cn

Let $ID_i, ID_s, ID_{TA}$ be the identity of user $U_i$, $SP$ and $TA$, respectively. Let $EID_i, BIO_i, PW_i$ be $U_i$'s pseudo identity, biometric, and password, respectively. $\triangle T$ is the maximum transmission delay. $h(\cdot)$ is a hash function. $E_K(\cdot), D_K(\cdot)$ are the symmetric key encryption, decryption algorithms with key $K$. Let $K_{SP}, K_{TA}$ be the secret key of $SP$ and $TA$, respectively.

The scheme consists of registration phase, authentication and key agreement phase, biometric and password update phase. The registration phase, authentication and key agreement phase can be described as follows (see Table 1).

Table 1: The Yu-Park key agreement scheme

| User $U_i$: $\{ID_i, PW_i\}$ | TA: $\{K_{TA}\}$ | $SP$: $\{ID_s, K_{SP}\}$ |
|---|---|---|
| Input $ID_i, PW_i$. | | |
| Imprint $BIO_i$. Pick a nonce $R_i$ to compute $Gen(BIO_i) = (\sigma_i, \tau_i)$, $EPW_i = h(PW_i \| \sigma_i)$. | Compute $EID_i = h(ID_i \| R_i)$, $X_{US} = h(ID_i \| R_i \| K_{TA})$, $Q_1 = X_{US} \oplus h(EPW_i \| R_i)$, $Q_2 = h(ID_{TA} \| K_{TA}) \oplus X_{US}$, $W_1 = h(ID_{TA} \| K_{TA}) \oplus X_{US}$. | |
| $\xrightarrow[\text{[secure channel]}]{ID_i,\ EPW_i,\ R_i}$ | Store $\{W_1, EID_i\}$. | |
| Compute $K_i = h(ID_i \| \sigma_i \| PW_i)$, $Q_3 = R_i \oplus h(EPW_i \| ID_i \| \sigma_i)$, $EM_i = E_{K_i}(EID_i, Q_1, Q_2, Q_3, \tau_i)$. Store $EM_i$. | $\xleftarrow{Q_1,\ Q_2,\ EID_i}$ | Input $ID_s$. |
| | Retrieve $\{W_1, EID_i\}$, to compute $X_{US} = W_1 \oplus h(ID_{TA} \| K_{TA})$, $W_2 = ID_s \oplus h(K_{TA} \| ID_{TA})$. | $\xleftarrow{ID_s}$ |
| | Store $W_2$. $\xrightarrow{X_{US},\ EID_i}$ | Compute $ES_i = E_{K_{SP}}(X_{US})$. Store $\{ES_i,\ EID_i\}$. |
| Input $ID_i, PW_i$. Imprint $BIO_i$. Compute $\sigma_i = Rep(BIO_i, \tau_i)$, $K_i = h(ID_i \| \sigma_i \| PW_i)$, $(EID_i, Q_1, Q_2, Q_3, \tau_i) = D_{K_i}(EM_i)$, $EPW_i = h(PW_i \| \sigma_i)$, $R_i = Q_3 \oplus h(EPW_i \| ID_i \| \sigma_i)$, $X_{US} = Q_1 \oplus h(EPW_i \| R_i)$. Check $Q_2 = h(EID_i \| X_{US} \| EPW_i)$. If so, pick a nonce $RN_1$ and a timestamp $T_1$ to compute | | Check $\|T_1^* - T_1\| \le \triangle T$. Retrieve $ES_i$ with $EID_i$. Compute $X_{US} = D_{K_{SP}}(ES_i)$, $RN_1 = U_1 \oplus X_{US} \oplus T_1$, |
| $U_1 = RN_1 \oplus X_{US} \oplus T_1$, | $\xrightarrow[\text{[open channel]}]{EID_i,\ U_1,\ U_2,\ Auth_{U-S},\ T_1}$ | $ID_i = U_2 \oplus h(X_{US} \| RN_1 \| T_1)$. Check |
| $U_2 = ID_i \oplus h(X_{US} \| RN_1 \| T_1)$, $Auth_{U-S} = h(ID_i \| EID_i \| RN_1 \| X_{US} \| T_1)$. | | $Auth_{U-S} = h(ID_i \| EID_i \| RN_1 \| X_{US} \| T_1)$. If so, pick a nonce $RN_2$ and the timestamp $T_2$ to compute $S_1 = (ID_s, RN_2) \oplus h(X_{US} \| T_2 \| RN_1)$, |
| | $\xleftarrow{S_1,\ Auth_{S-U},\ T_2}$ | $SK = h(ID_i \| ID_s \| RN_1 \| RN_2)$, $Auth_{S-U} = h(RN_1 \| RN_2 \| X_{US} \| SK)$. |
| Check $\|T_2^* - T_2\| \le \triangle T$. If so, compute $(ID_s, RN_2) = S_1 \oplus h(X_{US} \| T_2 \| RN_1)$, $SK = h(ID_i \| ID_s \| RN_1 \| RN_2)$. Check $Auth_{S-U} = h(RN_1 \| RN_2 \| X_{US} \| SK)$. | | |

## 3 The loss of user anonymity

As for the anonymity, it argues that (see §5.1.9, Ref.[7]): *We assume that the malicious attacker (MA) can eavesdrop the exchanged messages. However, MA is impossible to calculate $U_i$'s real identity $ID_i$ and SP's real identity $ID_s$ without the secret credential $X_{US}$.*

We find the argument is not sound and misleading. In fact, an adversary can directly recover the pseudo-identity $EID_i$ by capturing messages transmitted via the open channel. Note that the pseudo-identity is issued by the trust authority TA in the registration phase, and is unchanged in different sessions. Therefore, the adversary can attribute different sessions launched by the user $U_i$ to the pseudo-identity $EID_i$. Though the adversary can not retrieve $ID_i$ from the equation $EID_i = h(ID_i \| R_i)$, the exposure of $EID_i$ does indeed thwart the intention of anonymity. We refer to the Fig.1 for the true signification of anonymity.

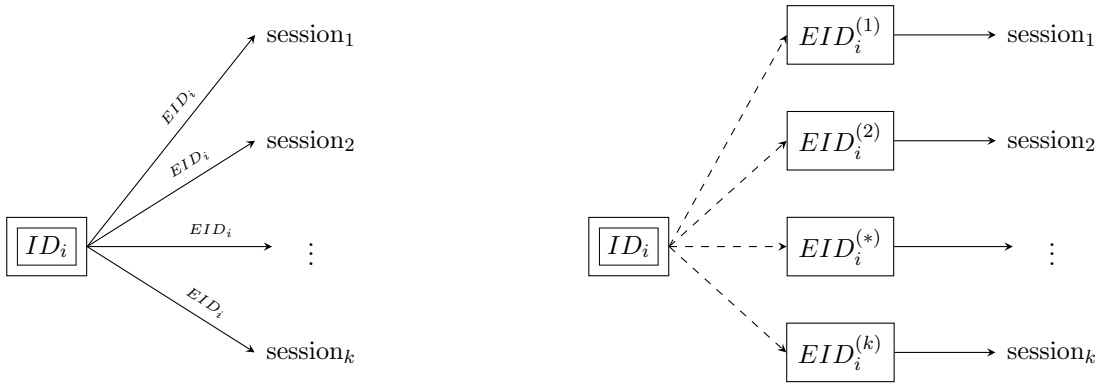


Fig.a: The false anonymity

Fig.b: The true anonymity

Figure 1: The false anonymity versus true anonymity

We want to stress that the identity of a person or thing is the characteristics that distinguish it from others. In Fig.a, the identifier $ID_i$ uniquely corresponds to the pseudo-identifier $EID_i$, and different sessions (launched by this entity) can be attributed to the unique pseudo-identifier. In this case, the pseudo-identifier can be eventually used to recognize this entity.

## 4 The loss of SP anonymity

The Boolean logic operation XOR is widely used in cryptography which compares two input bits and generates one output bit. When the operator is performed on two strings, they must be of a same bit-length. Otherwise, the shorter string should be stretched by padding some 0s to its left side. The scheme has neglected the basic property and presented a flawed equation.

Suppose the output length of hash function $h(\cdot)$ is 256, such as SHA-256. By the following equations

$$
\begin{aligned}
X_{US} &= Q_1 \oplus h(EPW_i \| R_i), \\
U_1 &= RN_1 \oplus X_{US} \oplus T_1, \\
U_2 &= ID_i \oplus h(X_{US} \| RN_1 \| T_1)
\end{aligned}
$$

we find, the effective bit-length of operands $Q_1, RN_1, ID_i$ is 256. That means the effective bit-length of operand $(ID_s, RN_2)$ in the equation

$$S_1 = (ID_s, RN_2) \oplus h(X_{US} \| T_2 \| RN_1)$$

is 256, too. Generally, the nonce $RN_2$ has the same bit-length as the nonce $RN_1$. In this case, the string of identity $ID_s$ is entirely copied into $S_1$. With the captured $S_1$, the adversary can easily recover the identity.

## 5 Conclusion

In this note, we clarify the signification of anonymity and show that the Yu-Park key agreement scheme fails to keep anonymity. The findings in this note could be helpful for the future work on designing such key agreement schemes.

## References


[1] T. Docquier, et al.: Performance evaluation methodologies for smart grid substation communication networks: a survey. Comput. Commun. 198: 228-246 (2023)

[2] M. Nafees, et al.: Smart grid cyber-physical situational awareness of complex operational technology attacks: a review. ACM Comput. Surv. 55(10): 215:1-215:36 (2023)

[3] S. Vahidi, et al.: Security of wide-area monitoring, protection, and control (WAMPAC) systems of the smart grid: a survey on challenges and opportunities. IEEE Commun. Surv. Tutorials 25(2): 1294-1335 (2023)

[4] R. Cardenas, et al.: Modeling and simulation of smart grid-aware edge computing federations. Clust. Comput. 26(1): 719-743 (2023)

[5] K. Adewole and V. Torra: DFTMicroagg: a dual-level anonymization algorithm for smart grid data. Int. J. Inf. Sec. 21(6): 1299-1321 (2022)

[6] K. Sarieddine, et al.: A real-time cosimulation testbed for electric vehicle charging and smart grid security. IEEE Secur. Priv. 21(4): 74-83 (2023)

[7] S. J. Yu and K. S. Park: ISG-SLAS: secure and lightweight authentication and key agreement scheme for industrial smart grid using fuzzy extractor. J. Syst. Archit. 131: 102698 (2022)