A note on “blockchain-assisted authentication and key agreement scheme for fog-based smart grid”

Zhengjun Cao¹, Lihua Liu²

Abstract. We show that the scheme [Clust. Comput. 25(1): 451-468, 2022] fails to keep anonymity, not as claimed. The scheme simply acknowledges 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 though the adversary cannot recover the true identifier from the long-term pseudo-identifier. We also clarify some misunderstandings in the scheme.

Keywords: Authentication, Key agreement, Fog-based smart grid, Anonymity, Public ledger

1 Introduction

The smart grid moves the energy industry into a new era of reliability, availability, and efficiency [1–3]. The benefits associated with the smart grid 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, Tomar and Tripathi [7] have presented a key agreement scheme for blockchain and fog computing based smart grid environment. Its security goals consist of mutual authentication, session key agreement, no online trust authority, identity anonymity, traceability and revocation, perfect forward secrecy, distributed data storage and access, and resistance to various attacks. Though the scheme is interesting, we find it is flawed.

2 Review of the scheme

The scheme [7] has five entities: trusted authority (TA), cloud server (CS), fog node (FN), smart meter (SM), and blockchain (BC). TA is a government electricity board or a private service provider, who is responsible for registering the smart grid and fog nodes and provides authentication parameters to registered entities. CS is a trusted entity that acts as a peer of the blockchain. It is responsible for verifying smart meters and fog nodes through blockchain. FN acts as a peer in the blockchain formed by multiple fog nodes and a cloud server. SM is a device inside a smart home responsible for sending

¹Department of Mathematics, Shanghai University, Shanghai, 200444, China
²Department of Mathematics, Shanghai Maritime University, Shanghai, 201306, China.
Email: liulh@shmtu.edu.cn
energy utilization data to the nearest fog node. BC operations are demonstrated by employing the consortium blockchain platform Hyperledger Fabric.

<table>
<thead>
<tr>
<th>Table 1: The Tomar-Tripathi key agreement scheme</th>
</tr>
</thead>
<tbody>
<tr>
<td>$SM_i: {ID_m}$</td>
</tr>
<tr>
<td>Pick $u \in \mathbb{Z}<em>q^*$ to compute the public key $Pb</em>{m} = u \cdot P$.</td>
</tr>
<tr>
<td>$SM_i: {ID_m, Pb_{m}, S_i}$</td>
</tr>
<tr>
<td>Pick $r \in \mathbb{Z}<em>q^*$ to compute $a = h_1(r)[S], A = a \cdot P,$ $S_i = h_1(S,P)[ID_m]$, $IP_i = S_i \oplus h_1(a \cdot Pb</em>{CS})[PID_m]$, $IP_i$ is the timestamp, $\kappa = h_3(h_i(Pb_{m}), IP_i, A, T_m, PID_{m_i})$, $M_i = (h_i(Pb_{m}), IP_i, A, T_m, \kappa)$</td>
</tr>
<tr>
<td>$SM_i: {ID_m, Pb_{m}, S_i}$</td>
</tr>
<tr>
<td>$FN_j: {ID_f, Pb_{fj}}$</td>
</tr>
<tr>
<td>Check the validity of timestamp. Compute $CP_{FN} = (b + \tau C, E_j = PID_{fj} \oplus PID_{m_i})$ $SK_{FN} = h_2(CP_{FN}, PID_{m_i}, PID_{fj}, l_a, T_CS)$.</td>
</tr>
<tr>
<td>Compute $PID_{fj} = E_j \oplus PID_{m_i}$, $CP_{SM} = (a + h_1(PID_{m_i})A)l_a, SK_{SM} = h_2(CP_{SM}, PID_{m_i}, PID_{fj}, l_a, T_CS)$. Check $h_2(PID_{m_i}, l_a, S_i, PID_{fj}, T_CS, SK_{SM}) = \mu_2$.</td>
</tr>
</tbody>
</table>

The scheme consists of four phases: System setup, Blockchain initialization, Registration, Mutual authentication and key agreement. In the setup phase, TA selects the elliptic curve $E$ with base point $P \in G$, where $G$ is a group generated by $P$ of prime order $q$. TA picks $s \in \mathbb{Z}_q^*$ as private key and sets $Pb_{TA} = s \cdot P$ as public key. TA publishes system public parameters as $par = [G; P; q; Pb_{TA}; h_1, \cdots, h_7]$. The public key for CS is set as $Pb_{CS} = v \cdot P$, where $v \in \mathbb{Z}_q^*$ is the
corresponding secret key. Let \( h_1, \ldots, h_7 \) be hash functions, defined as follows.

\[
\begin{align*}
  h_1 : \{0,1\}^* \rightarrow \mathbb{Z}_q^* , & \quad h_2 : \{0,1\}^* \times \mathbb{G} \rightarrow \mathbb{Z}_q^* , \quad h_3 : \{0,1\}^* \times \{0,1\}^* \times \mathbb{G} \times \mathbb{Z}_q^* \rightarrow \mathbb{Z}_q^* , \\
  h_4 : \{0,1\}^* \times \{0,1\}^* \times \mathbb{G} \times \mathbb{Z}_q^* \rightarrow \mathbb{Z}_q^* , & \quad h_5 : \mathbb{G} \times \{0,1\}^* \times \{0,1\}^* \rightarrow \mathbb{Z}_q^* , \quad h_6 : \{0,1\}^* \times \mathbb{G} \times \{0,1\}^* \times \{0,1\}^* \rightarrow \mathbb{Z}_q^* , \\
  h_7 : \mathbb{G} \times \{0,1\}^* \times \{0,1\}^* \times \mathbb{G} \rightarrow \mathbb{Z}_q^* . & \quad
\end{align*}
\]

We refer to Table 1 for other related phases.

### 3 The loss of anonymity

Though the scheme is interesting, we find it is flawed. As for the anonymity, it argues that (see §5.3, Ref.[7])

> In the proposed protocol, the real identity of an \( SM_i \) is included in \( \overline{S_i} = h_1(S_i P) ID_{m_i} \), which is further hidden in \( IP = \overline{S_i} \oplus h_1(a P_{b_{CS}}) PID_{m_i} T_{m_i} \). To calculate the real identity of smart meter, attacker needs to solve the ECDL (Elliptic Curve Discrete Logarithm) problem. Therefore, the proposed scheme ensures identity anonymity.

We find the argument is not sound and misleading. In fact, an adversary can directly recover \( h_1(P_{b_{m_i}}) \) by capturing the message \( M_1 \) transmitted via the public channel. He then uses it to query the public ledger to extract the pseudo-identity \( PID_{m_i} \). 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 to the \( PID_{m_i} \) using the hash value \( h_1(P_{b_{m_i}}) \) as an indexing token. Though the adversary can not directly retrieve the real identity \( ID_{m_i} \) from the equation \( PID_{m_i} = h_2(ID_{m_i}, s \cdot P_{b_{CS}}) \), the exposure of \( PID_{m_i} \) does indeed thwart the intention of anonymity. We refer to the Fig.1 for the true signification of anonymity.

![Fig.a: The false anonymity](image1.png)  
![Fig.b: The true anonymity](image2.png)

**Figure 1:** The false anonymity versus true anonymity

Notice that the identity of a person or thing is the characteristics that distinguish it from others. The real identifier \( ID_{m_i} \) could be a regular string, and the pseudo-identifier \( PID_{m_i} \) is a random string. In Fig.a, the identifier \( ID_{m_i} \) uniquely corresponds to the pseudo-identifier \( PID_{m_i} \), and
different sessions (launched by this entity) can be attributed to the unique pseudo-identifier. In this case, the unique pseudo-identifier can be eventually used to recognize this entity.

4 The misunderstanding of public key

Public key, in a narrow sense, is a cryptography key that can be obtained and used by anyone to encrypt messages intended for a particular recipient [8]. It can also be used to verify signatures generated by the particular entity. All in all, public key is easily obtained by anyone and can be used to recognize its owner. The scheme has neglected the signification of public key.

In order to authenticate the smart meter $SM_i$, the fog node $FN_j$ uses the hash value $h_1(Pb_{m_i})$ to query the public ledger for extracting the pseudo-identifier $PID_{m_i}$. The hash value is directly exposed to an outer adversary. Since $Pb_{m_i}$ is the public key of the smart meter, and the hash function $h_1$ is also publicly accessible, the adversary can test each smart meter’s public key $\chi$ such that $h_1(\chi) = h_1(Pb_{m_i})$. Once such a key $\chi$ is found, we have $\chi = Pb_{m_i}$, due to the collision-free property of the hash function $h_1$. Using the public key $Pb_{m_i}$, the adversary can recognize the target smart meter $SM_i$. Therefore, the scheme fails to keep anonymity.

5 The misunderstanding of ledger

As we see, in the block-chain scenario, the ledger is public and sustained by all participants. But we find the scheme has neglected this basic fact. In the scheme, the cloud server $CS$ needs to query the public ledger with $h_1(Pb_{m_i}), h_1(Pb_{f_j})$ to extract $ID_{f_j}, ID_{m_i}, D_i$. Since the hash values $h_1(Pb_{m_i})$ can be retrieved by an outer adversary from the message $M_1$ or $M_2$, the adversary can also query the public ledger to extract the target identity $ID_{m_i}$.

6 The repetitive specification of hash functions

The scheme needs to use 7 hash functions. See the following computations:

$$\begin{align*}
h_1(PID_{m_i}∥h_1(Pb_{m_i})), & \quad h_2(ID_{f_j}, s ∥ Pb_{CS}), \quad h_3(h_1(Pb_{m_i}), IP_i, A, T_m, PID_{m_i}), \\
h_4(h_1(Pb_{m_i}), h_1(Pb_{f_j}), IP_i, A, B, \tilde{B}, T_m, T_f, K_f, PID_{f_j}), & \\
h_5(l_a, PID_{f_j}, T_{CS}, SK_{CS}), & \quad h_6(PID_{m_i}, l_b, S_i, PID_{f_j}, T_{CS}, SK_{CS}), \quad h_7(CP_{FN}, PID_{m_i}, PID_{f_j}, l_a, T_{CS}).
\end{align*}$$

These notations are really tedious. Since they have a same codomain $\mathbb{Z}_q^*$, it only needs to specify a unique hash function $h : \{0, 1\}^* \rightarrow \mathbb{Z}_q^*$. In this case, all strings of different components are concatenated. Now, the related computations become

$$\begin{align*}
h(PID_{m_i}∥h(Pb_{m_i})), & \quad h(ID_{f_j}∥s ∥ Pb_{CS}), \quad h(h(Pb_{m_i})∥IP_i∥A∥T_m∥PID_{m_i}), \\
h(h(Pb_{m_i})∥h(Pb_{f_j})∥IP_i∥A∥\tilde{B}∥T_m∥T_f∥K_f∥PID_{f_j}), & \\
h(l_a∥PID_{f_j}∥T_{CS}∥SK_{CS}), & \quad h(PID_{m_i}∥l_b∥S_i∥PID_{f_j}∥T_{CS}∥SK_{CS}), \quad h(CP_{FN}∥PID_{m_i}∥PID_{f_j}∥l_a∥T_{CS}).
\end{align*}$$
7 Conclusion

In this note, we show that the Tomar-Tripathi key agreement scheme is flawed because it is not explicitly organized and expressed. The findings in this note could be helpful for the future work on designing such key agreement schemes.

References


