Provably Secure ID-Based Signature without Trusted PKG for Smart Grid

Wei Peng
GanSu Electric Power Design Institute
Lanzhou, China
pwlut@163.com

Xia Bai
GanSu Electric Power Design Institute
Lanzhou, China
907526249@qq.com

Abstract—By using Gap Diffie-Hellman groups, we construct an efficient ID-based signature scheme without trusted PKG for smart grid, which security relies on the hardness of the Computational Diffie-Hellman Problem (CDHP). In this scheme, PKG is prevented from forging a legal user’s signature because it only generates the partially private key. The scheme is proved to be secure against existential forgery on adaptively chosen message and ID attack, assuming CDHP is intractable. Our scheme not only satisfies security properties but also has a higher efficiency.

Keywords—smart grid; key escrow; id-based signature; random oracle model

I. INTRODUCTION

In traditional CA-based Cryptosystems, the binding between public key and identity of the signer is obtained via a digital certificate, issued by a Trusted Third Party called Certifying Authority (CA). To simplify the certificate management process, an ID-based Cryptosystem (IBC) based on integer factorization problem was proposed by Shamir [1], which allows a user to use his identity as the public key. But there are some drawbacks in ID-based systems. The most criticism against ID-based systems is that PKG knows the private key of all users, so it is able to impersonate any user to sign a document or decrypt an encrypted message. It implies that the PKG must be trusted unconditionally otherwise the systems will soon be collapsed. However, it would be difficult to assume the existence of a trusted party in smart grid, where the communication parties are changing frequently.

Several ID-based signature schemes based on the bilinear maps (pairings) have been proposed. These include Cha and Cheon’s [2] scheme and Hess’ [3] scheme. They gave a formal definition of unforgeability of ID-based signature against chosen message attack and proved that their schemes are secure in the random oracle model assuming the Computational Diffie-Hellman (CDH) problem is computationally intractable. Al-Riyami and Paterson [4] introduced and developed the notion of certificateless public key cryptography (CL-PKC). CL-PKC is a model for the use of public key cryptography, which is intermediate between the identity-based and traditional PKI approaches. In [5], the authors proposed an ID-based signature without trusted PKG from bilinear pairings. Gorantla and Saxena [6] proposed an efficient certificateless signature scheme. But their scheme was shown to be insecure [7]. Liu et al. [8] proposed an ID-based signature without trusted Private Key Generator, which solved the key escrow problem by binding two public keys with a same identity.

In this paper, we assume that there is only one PKG in our systems and the PKG is not a trusted party anymore. In our systems, if the dishonest PKG impersonate an honest user to sign a document, the user can provide a proof that the PKG is dishonest.

The rest of the paper is organized as follows. The formal model of a secure ID-based signature without trusted PKG is presented in Section II. In Section III, the proposed ID-based signature scheme is presented. The security and efficiency analysis of our schemes are given in Section IV. Finally, Section V concludes this paper.

II. BASIC MODELS

A. Framework of ID-based Signature without trusted PKG

An ID-based signature scheme without trusted PKG consists of four algorithms: Setup, Extract, Sign and Verify.

Setup: This algorithm is usually executed by the private key generator (PKG). On a unary string input $1^k$ where $k$ is a security parameter, it produces the public parameters params, which includes a description of a finite signature space, a description of a finite message space. The master secret $x_{PKG}$ is also the output, which is kept secret.

Extract: On an arbitrary string input $id$, it computes the private signing key $(x_1, x_2)$ with the help of master secret $x_{PKG}$, and the corresponding public verification key $(y_1, y_2)$.

Sign: Suppose the requester wants a message $m$ to be signed, after the execution of Sign algorithm, a signature $\sigma$ will be produced.

Verify: Input a signature $\sigma$, a message $m$ and the signer’s public verification key $(y_1, y_2)$, it outputs “true”
B. Attack Model for ID-based Signature without trusted PKG

We define unforgeability through the following game between a challenger C and an attacker A.

Setup: The challenger C takes a security parameter \( k \) and runs the Setup algorithm. C sends the public system parameters to the attacker A and keeps the master key \( s_{PKG} \) itself.

Attack: The attacker A adaptively performs the following three queries:

1. Hash functions queries: C returns a hash value for the requested input.
2. Extract queries: When A submits an identity id, C runs Extract algorithm to return the private key \( x_2 \).
3. Sign queries: When A submits \((id, m)\) (id is a chosen identity by A) to ask a signature, C runs Sign algorithm with the attacker A. Then A obtains a valid signature \( \sigma = \text{Sign}(id, m, x_2) \).

Forgery: The attacker A outputs a forged signature \((\sigma, m)\), where \( m \) is a plaintext message. A wins the game if \( \sigma \) is a valid signature of \( m \) for \( id \) in the following three cases:

The adversary A maybe forge a signature colluding with a dishonest PKG. Thus there are three cases [9] to discuss.

Case 1: PKG is honest. A forges a valid signature with no help of a trusted PKG.

Case 2: PKG is semi-honest. In this case A has a piece of additional information \( x_2 \) from the PKG.

Case 3: PKG is malicious. It wants to impersonate an honest user whose identity information is \( id \), A forges a valid signature of the honest user.

Using this attack model, we can reduce the security of ID-based signature without trusted PKG to the hardness of CDHP.

Definition 1. An ID-based signature scheme is said to be existential unforgeable against adaptive chosen-message-and-identity attacks if no adversary has a non-negligible advantage in the above game.

III. ID-BASED SIGNATURE SCHEME WITHOUT TRUSTED PKG

Let \( G_1 \) be a cyclic additive group generated by \( P \) with the prime order \( q \). We introduce the following mathematical problems in \( G_1 \).

1. Discrete Logarithm Problem (DLP): Given two elements \( P, Q \in G_1 \), to find an integer \( n \in \mathbb{Z}_q^* \), such that \( Q = nP \) whenever such an integer exists.

2. Decision Diffie-Hellman Problem (DDHP): Given \( P, aP, bP, cP \in G_1 \) for \( a, b, c \in \mathbb{Z}_q^* \), to decide whether \( c = ab \mod q \).

3. Computational Diffie-Hellman Problem (CDHP): Given \( P, aP, bP \in G_1 \) for \( a, b \in \mathbb{Z}_q^* \), to compute \( abP \).

We assume through this paper that CDHP and DLP are intractable. When the DDHP is easy but the CDHP is hard on the group \( G_1 \), we call \( G_1 \) a Gap Diffie-Hellman (GDH) group.

In this section, we first define two cryptographic hash functions \( H_1:\{0,1\}^n \rightarrow G_1, H_2:\{0,1\}^n \times G_1 \rightarrow \mathbb{Z}_q^* \).

Setup: PKG randomly chooses \( s_{PKG} \in \mathbb{Z}_q^* \) and sets \( Q_{PKG} = s_{PKG}P \). The public parameters of the system are \( \text{params} = \{ G_1, G_2, e, P, Q_{PKG}, H_1, H_2 \} \). PKG keeps \( s_{PKG} \) secretly as the master key.

Extract: A user submits his identity information \( id \) and authenticates himself to PKG. The user then randomly chooses an integer \( s_i \in \mathbb{Z}_q^* \) as his partially secret key and computes \( Q_i = s_iP \) as his partially public key. Suppose the signer’s identity is given by the string \( id \), the other partially secret key of the identity is then given by \( s_2 = s_{PKG}Q_i \) where \( Q_i = H_1(id, Q_i) \), which is computed by the PKG and given to the signer. For a signer, \((Q_i, s_2)\) is his public key and \((s_1, s_2)\) is his private key.

Sign: To sign a message \( m \), the signer chooses a random integer \( k \in \mathbb{Z}_q^* \):

1. \( U = kP, r = H_2(m, U) \);
2. \( V = rs_1Q_2 + rs_2 = rs_1Q_2 + s_2; \)
   The signature is then the pair \((U, V)\).

Verify: On receiving a message \( m \) and signature \((U, V)\) the verifier computes:

1. \( T = H_1(id, Q_i), r = H_2(m, U), T = Q_i + Q_{PKG}; \)
2. Accept the signature if and only if \( e(V, P) = e(Q_2, T) \).
   It is straightforward to check that the verification equation holds for a valid signature.

IV. SECURITY AND EFFICIENCY ANALYSIS OF OUR SCHEME

A. Security

Theorem 1 (Correctness) Our scheme in section III is correct.

Proof:

\[
\begin{align*}
e(V, P) &= e(rs_1Q_2 + rs_2, P) \\
   &= e(r(s_1Q_2 + s_{PKG}Q_2), P) \\
   &= e(Q_2, (s_1 + s_{PKG})P) \\
   &= e(Q_2, s_{PKG}) \\
   &= e(Q_2, T)
\end{align*}
\]

This theorem is proved.
Theorem 2: Our ID-based signature scheme is secure against on existential adaptively chosen message and ID attacks under the assumption of CDHP is hard in \( G_1 \) and random oracle model.

**Proof:** We referred to the proof of unforgeability of the signature scheme by Pointcheval and Stern [10,11]. There are three cases to discuss:

**Case 1:** PKG is honest. A forges a valid signature with no help of a trusted PKG.

We suppose that \( H_1, H_2 \) are random oracles, and there exists a probabilistic polynomial time Turing machine \( A \) whose input only consists of public data. We assume that \( A \) can make \( q_1 \) queries to the random oracle \( H_1, q_2 \) queries to the random oracle \( H_2 \) and \( q_s \) queries to the signing oracle.

\( C \) gives \( A \) the system parameters \( Q_{PKG}=aP \). Note that \( a \) is unknown to \( C \). This value simulates the master key value for the PKG in the game.

**H_1-Queries:** A can query the random oracle \( H_1 \) at any time. \( C \) simulates the random oracle by keeping list of couples \((\Sigma,Q_{2,0})\) which is called the \( L_1\)-List, where \( \Sigma \) is a couple of \((id_i,Q_{1,0})\). When the oracle is queried with an input \( \Sigma \), \( C \) responds as follows:

1. If the query \( \Sigma \) is already on the \( L_1\)-List in the couple \((\Sigma,Q_{2,0})\), then \( C \) outputs \( Q_{2,0} \).
2. Otherwise \( C \) selects a random \( Q_{2,s} \), outputs \( Q_{2,s} \) and adds \((\Sigma,Q_{2,s})\) to the \( L_1\)-List.

**Extract-Queries:** A can query the partially private key for any identity \( id \) and the public key.

If \( Q_{2,s}\neq H_1(id_i,Q_{1,0}) \), \( C \) returns invalid. Otherwise, it outputs the partially private key \( S_{2,0} \) corresponding to \( id \), which is obtained by running Extract algorithm.

**H_2-Queries:** A can query the random oracle \( H_2 \) at any time. \( C \) simulates the random oracle by keeping list of couples \((\Sigma,Q_{2,0})\) which is called the \( L_2\)-List, where \( \Sigma \) is a triple of \((m_i,U_i')\). When the oracle is queried with an input \( \Sigma \), \( C \) responds as follows:

1. If the query \( \Sigma \) is already on the \( L_2\)-List, then \( C \) outputs the same answer.
2. Otherwise \( C \) selects a random \( r_i \), outputs \( r_i \) and adds \((\Sigma,r_i)\) to the \( L_2\)-List.

**Sign-Queries:** For a sign request on \( m, C \) simulates the value of \( H_2 \) in the way as mentioned above. \((U',V')\) will be used as the answer, where

**Verify** \((id,m,Q_{1},Q_{2},U',V')=\text{True}\).

It follows from the forking lemma [11] that if \( A \) is a sufficiently efficient forger in the above interaction, then we can construct a Las Vegas machine \( A' \) that outputs two signed messages \((id,m,U_1,V_1)\) and \((id,m,U_2,V_2)\) with \( r_i\neq r_2 \). So we have the following equations:

\[
V_1=r_1s_1s_2+r_2s_2 \\
V_2=r_1s_1s_2+r_2s_2 \\
\]

From above equations we can get the following equation.

\[
(r_1-r_2)\frac{(V_1-V_2)}{2}=s_2Q_2+bP \\
\]

Let \( Q_2=bP \) then \( (s_1+s_{PKG})P=\frac{(s_1+s_{PKG})bP}{2} \). If \( A \) succeeds in time \( \leq T \) with probability \( \frac{1}{2} \), then \( C \) can solve the CDH problem in expected time \( t\leq 12068q_2+2T(10q_2^2) \).

**Case 2:** PKG is semi-honest. In this case \( A \) has a piece of additional information \( S_2 \) from the dishonest PKG. \( C \) gives \( A \) the parameters \( Q_{PKG}=aP \). Note that \( a \) is unknown to \( C \). This value simulates the master key value for the PKG in the game.

The same as case 1, \( C \) can play the simulation twice so that \( A \) should produce two valid signature \((id_m,m,U_1,V_1)\) and \((id_m,m,U_2,V_2)\) with \( r_1\neq r_2 \). Now we could get the following equations.

\[
V_1=r_1s_1Q_2+r_1s_2 \\
V_2=r_2s_1Q_2+r_2s_2 \\
\]

From above equations we can get the following equation.

\[
(r_1-r_2)\frac{(V_1-V_2)}{2}=(s_1+s_{PKG})Q_2 \\
\]

The new scheme has much pre-computation, so it has higher efficiency than the existing schemes. From Table 1, it is easy to see that our ID-based signature scheme is more efficient.
TABLE I COMPARISON OF COMPUTATION COST WITH EXITING SCHEME

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Pre-Sign</th>
<th>Sign</th>
</tr>
</thead>
<tbody>
<tr>
<td>AP[4]</td>
<td>$1Pa$</td>
<td>$2G_1M+1G_1A+1G_2E$</td>
</tr>
<tr>
<td>CZK[5]</td>
<td>/</td>
<td>$3G_1M+1G_1A$</td>
</tr>
<tr>
<td>GS[6]</td>
<td>/</td>
<td>$2G_1M+1G_1A$</td>
</tr>
<tr>
<td>LSKW[8]</td>
<td>$1Pa+1G_1M$</td>
<td>$2G_1M+1G_1A+1G_2E$</td>
</tr>
<tr>
<td>Our Scheme</td>
<td>$1G_1M+1G_1A$</td>
<td>$2G_1M$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Pre-Verify</th>
<th>Verify</th>
</tr>
</thead>
<tbody>
<tr>
<td>AP[4]</td>
<td>/</td>
<td>$4Pa+1G_2M+1G_2E$</td>
</tr>
<tr>
<td>CZK[5]</td>
<td>/</td>
<td>$4Pa+1G_2M$</td>
</tr>
<tr>
<td>GS[6]</td>
<td>/</td>
<td>$3Pa+1G_1M+1G_2E$</td>
</tr>
<tr>
<td>LSKW[8]</td>
<td>$1Pa$</td>
<td>$1Pa+1G_1A+1G_2E$</td>
</tr>
<tr>
<td>Our Scheme</td>
<td>$1Pa+1G_1A$</td>
<td>$1Pa+1G_2E$</td>
</tr>
</tbody>
</table>

V. CONCLUSION

In this paper, we have proposed a new ID-based signature scheme without trusted PKG for smart grid. Our scheme is proved to be secure against existential forgery on adaptively chosen message and ID attack, assuming CDHP is intractable. In our systems, if the dishonest PKG impersonation an honest user to sign a document, the user can provide a proof that the PKG is dishonest. The scheme not only satisfies security properties but also has a higher efficiency.

REFERENCES