AN EFFICIENT STAMPED PROXY PARTIALLY BLIND SIGNATURE SCHEME UTILIZING ELLIPTIC CURVE CRYPTOSYSTEM

Nedal Tahat\textsuperscript{1,8}, Khaled Alzubi\textsuperscript{2}, Maysam Abu-Dalu\textsuperscript{3}, Ala Qadomi\textsuperscript{4}

\textsuperscript{1,2,3,4}Department of Mathematics
Faculty of Sciences
The Hashemite University
Zarqa 13133, JORDAN

Abstract: Proxy partially blind signature allows a proxy signer to explicitly embed presaged common information $c$ into the blind signature without loss of blindness property. The focus of this paper is to present the efficient stamped proxy partially blind signature (SPPBS) scheme based on elliptic curve discrete logarithm problem (ECDLP) with collision-resistant cryptographic hash function which records the time stamp during signing phase of the proxy partially blind signature and to allow the original signer to revoke delegating power whenever necessary. The proposed scheme satisfies all the security properties of proxy blind signature, i.e. distinguishability, nonrepudiation, unforgeability, identifiability, unlinkability, and prevention of misuse. Furthermore, the proposed scheme requires minimal operation in proxy delegation, partially blinding, signing extraction and verification thus makes it very efficient. This can be implemented in low power and small processor handheld devices such as smart card, PDA etc which work in low power and small processor.

Key Words: partially blind signature, proxy signature, elliptic curve, hash function, stamped

Received: July 10, 2015

\textsuperscript{8}Correspondence author
1. Introduction

Blind signature as introduced by David Chaum [1] is a form of digital signature in which the content of a message is blinded before it is signed. It allows a user to acquire a signature from the signer without revealing the message content for personal privacy. The notion of partially blind signatures was presented in [2]. It is a generalized model of the blind signature. The signer imposes some common information such as the signature date or any other information that is known to the signer and the signature requester. Obviously, the information embedded in the signature is used for verification and validation. A proxy signature protocol introduced by Mambo et al [3], proxy blind signature is an important extension of basic proxy signature; it can be widely used in many practical applications.

The proxy signature and blind signature have respective advantages. In some real situations, we need to inherit the merits of both proxy and blind signatures. The first proxy blind signature was proposed by Lin et al [4] in 2000. Proxy blind signature scheme is a digital signature scheme that combines the properties of both proxy signature and blind signature. In a proxy blind signature scheme, the proxy signer is allowed to generate a blind signature on behalf of the original signer. In the proxy signature scheme, the proxy signer knows the original message \( m \), but in the proxy blind signature scheme, the proxy signer does not know the \( m \). Proxy blind signature scheme is a protocol played by two parties in which a user obtains a proxy signer’s signature for a desired message and the proxy signer learns nothing about the message [5,6,7,8].

Tan et al [9] proposed the concept of proxy blind signature, having the advantages of both the proxy signature and the blind signature. Later, Lal et al [10] pointed out that Tan et al’s proxy blind signature scheme suffer from a kind of forgery attack due to the signature receiver. In 2004, Xue et al [11] proposed a new proxy protected proxy blind signature scheme with warrant. In this scheme, the CA (Certificate Authority) is needed. In 2005, Li et al [12] proved that Xue et al scheme failed to satisfy the unforgeability and strong unlinkability property. In 2008, Yang et al [13] proposed a new proxy blind signature scheme which satisfied all the security requirements of both the blind signature scheme and the proxy signature scheme. In 2010, Binayak et al proved that an attacker can produce the proxy blind signature on the forged message \( m' \), see [14]. So, Yang et al scheme is not secure against forgeability attack. The proxy partially blind signature scheme should satisfy the following properties [18]:

- **Correctness**: the correctness of the signature of a message signed through
the signature scheme can be checked by anyone using the signer’s public key.

- **Distinguishability**: The proxy signature must be distinguishable from the standard signature
- **No-repudiation**: Neither the origin nor the proxy must be able to sign in place of the other party. In other words, they cannot deny their signatures against anyone
- **Enforceability**: Only a designated proxy signer can create a valid proxy signature for the original signer.
- **Identifiability**: The verifier can recognize the proxy and the original signers.
- **Partial blindness**: It allows a user to obtain a signature on a message without illuminating anything about the message to the signer.

Recently, Pradhan and Mohapatra [15], also proposed a new proxy blind signature scheme based on ECDLP. They claim that their scheme is secure and satisfy all the required properties. Unfortunately, their scheme cannot hold the unlinkability property. To overcome this weakness, in 2014, Chande [16] and Tahat [17] proposed an improved proxy blind signature scheme was presented with the same intractable problem ECDLP. Through the verification of a proxy blind signature scheme the verifier cannot know whether signing (done by proxy signer) is within the delegation period or not. Proxy signer can make fool to the verifier by signing the message or document after the delegation period is over as there is no such precaution to record the time stamp during the proxy signing phase. Original signer cannot drag the delegation whenever necessary, so that a proxy signer may misuse the delegating power for signing. Hence, it is necessary to provide a time stamp during the signing phase of the proxy partially blind signature and to allow the original signer to dragging delegating power whenever necessary. Thus, in this study we propose a secure SPPBS scheme based on ECDLP plus our scheme requires minimal operation in proxy delegation, partially blinding, signing extraction and verification thus makes it very efficient.

The organization of the paper is sketched as follows. The Section 2 gives a brief review of the background of elliptic curve group and the definition of all the security requirement of a proxy blind signature. We present our proposed scheme and security analyses of the proposed scheme in Sections 3. We analyze
the security and performance of the new scheme in section 4. Finally, we give some conclusions in Section 5

2. Elliptic Curve Cryptograph

The theory of elliptic curve has been intensively studied in the pure mathematics field for 160 years and it was applied for factoring large integers in early 198’s. Miller [19] and Koblitz [20] independently found that the group of points on an elliptic curve is a proper group on which the discrete logarithm is intractable. These groups are suitable for implementing El-Gamal type cryptosystems by replacing the residue group of integers in El-Gamal by these groups.

2.1. An Elliptic Curve Group Over the Field $F_p$

Let the symbol $E(F_p)$ denote an elliptic curve $E$ over a prime finite field $F_p$, defined by an equation

$$y^2 = x^3 + ax + b, \quad a, b \in F_p,$$

and with the discriminant

$$\Delta = 4a^3 + 27b^2 \neq 0.$$

The points on $E(F_p)$ together with an extra point $O$ called the point at infinity form a group

$$G = \{(x, y) : xy \in F_p, E(x, y) = 0\} \cup \{O\}.$$

Let the order of $G$ be $n$. $G$ is a cyclic additive group under the point addition “+” defined as follows: Let $P, Q \in G$, $l$ be the line containing $P$ and $Q$ (tangent line to $E(F_p)$ at $P = Q$ if), and $R$, the third point of intersection of $l$ with $E(F_p)$. Let $l'$ be the line connecting $R$ and $O$. Then $P$ “+” $Q$ is the point such that $l'$ intersects $E(F_p)$ at $R$ an $O$ and $P$ “+” $Q$. Scalar multiplication over $E(F_p)$ can be computed as follows

$$tP = P + P + \cdots + P(t \text{ time}).$$

The following problems defined over $G$ are assumed to be intractable within polynomial time.

Elliptic curve discrete logarithm problem (ECDLP): For $x \in R$ $Z_n^*$ and $P$ the generator of $G$, such that $Q = xP$, compute $x$. 
3. Proposed Scheme

In this section, we propose a SPPBS scheme based on ECDLP. The proposed scheme is divided in five phases: System parameter initialization, Proxy delegation phase, partially blind signing phase, signature extraction, and signature verification phase.

3.1. System Parameter Initialization

The parameter used in the proposed scheme is:

- A finite point $G$ whose order is a large prime number in $E(F_q)$, where $G \neq O$ such that the order of $G$ is $n$;
- $U_o$: Original signer;
- $U_P$: Proxy signer;
- $U_R$: Signature requester;
- $AS$: Authenticated server as trusted third party;
- $x_o$: Original signer’s secret key;
- $y_o$: Original signer’s public key, $y_o = x_oG = (h_ol_o)$;
- $x_P$: Proxy signer’s secret key;
- $y_P$: Original signer’s public key, $y_P = x_PG = (h_Pl_P)$;
- $x_R$: Signature requester secret key;
- $y_R$: Signature requester’s public key $y_R = x_PG = (h_Rl_R)$;
- $H(\cdot)$: a cryptography secure one way hash function;
- $\parallel$: which denote the concatenation of two strings;
- $m_w$: message warrant;
- $m$: message.
3.2. Proxy Delegation Phase

- Original signer $U_o$ chooses a random number $\bar{\omega} \in \mathbb{Z}_n^*$, and then computes

\[ \rho = \bar{\omega}G = (\alpha\beta), \]  

where $u \equiv \alpha \pmod{n}$

\[ \delta = (x_o + \bar{\omega}H(m_w \parallel u)) \pmod{n}. \]  

Original signer $U_o$ sends $(\rho, \delta)$ along with the message warrant $m_w$ to the proxy signer $U_P$, and AS, via a secure channel.

$U_P$ verifies the equation

\[ \delta G = (y_o + \rho H(m_w \parallel u)) \pmod{n}. \]  

If the result is correct, then $(r, \delta)$ will be accepted.

- $U_P$ computes $S_{Pr} \equiv (\delta + x_P h_o) \pmod{n}$, as his/her proxy signature with a private key.

3.3. Partially Blinding Phase

- $U_R$ Signature requester prepares common information $c$, according to a pre-defined format. Then the value “$c$” is a common input of both the requester $U_r$ and proxy signer $U_P$. $U_r$ sends $c$ to $U_P$

- Proxy signer $U_P$ randomly selects an integer $\omega \in \mathbb{Z}_n^*$, and computes

\[ d = (\omega + cx_P)G. \]  

$U_P$, then sends $(\rho dm_w)$ to the signature requester $U_R$.

$U_R$ checks $U_o$ and $U_P$ identities and the delegation lifetime of the warrant $m_w$.

- If the above checking is successful, $U_R$ selects two random numbers $\sigma\gamma \in \mathbb{Z}_n^*$ and computes
\[ \bar{\rho} = (d + (\sigma + x_R)G + \gamma y_{Pr}) \cdot \] (9)

Here \( x_R \) is the private key of \( U_R \) and \( y_{Pr} = S_{Pr}G \)

\[ f = H (\bar{\rho} \parallel m) (modn), \] (10)

\[ f^* = (\gamma - f) (modn). \] (11)

If \( \bar{\rho} = 0 \), then \( U_R \) needs to select a new tuple \((\sigma\gamma)\) otherwise \( U_R \), sends \( f^* \)
to \( U_P \) and AS.

For signing blinded message, \( U_P \) must request a time stamp for the message. \( U_P \) transmits his identity and \((\delta, m_w, d)\) to AS. AS checks whether the received \( \delta \) from \( U_P \) and the received \( \delta \) from Alice is identical. If these two are same then AS check

\[ \delta G = (y_o + \rho H(m_w \parallel u)) (modn). \] (12)

If it satisfies, AS goes through the following steps

1. It is still in the valid proxy delegation specified in \( m_w \).

2. \( \rho \) is not in the revocation list. If \( \rho \) is in the revocation list, then it mean that the delegation is revoked.

- After that, AS choose a random number \( \omega_s \in Z_n^* \) and compute

\[ \rho_s = \omega_sG = (w\mu), \] where \( \lambda = w(modn), \] (13)

\[ D = H (\lambda \parallel timestamp \parallel f^*) (modn). \] (14)

AS sends \( D \) to the proxy signers \( U_P \) and \( U_R \). After receiving \( D \) \( g \), the proxy signer \( U_P \) computes

\[ \bar{\delta} = (\omega + f^*S_{Pr} + D) (modn), \] (15)

as the signed message and sends it to the signature requester \( U_R \).

### 3.4. Signing Extraction Phase

After receiving \( \bar{\delta} \) from \( U_P \), the signature requester \( U_R \) computes,

\[ \delta^* = (\sigma + \bar{\delta} - D) G(modn). \] (16)

Thus, the proxy blind signature on message \( m \) is the tuple \((mm_w\delta^*f)\).
3.5. Signature Verification Phase

Verifier can verify the proxy blind signature by checking whether
\[ f = H (\delta^* + cy + y_R + fy_{Pr} \parallel m) \mod n. \] (17)

4. Security Analysis and Performance Evaluation

In this section we carefully discuss some of the security properties and the performance evaluation of our proposed a stamped partially blind signature scheme.

4.1. Security Analysis

The proposed scheme satisfies the following security properties:

- **Correctness:** The correctness of the proposed scheme is shown in the following theorems

  **Theorem 1.** In the signature generation phase, verifier can verify the tuple proxy blind signature \((mmw, \delta^*f)\) using Eq. (17).

  **Proof.** The correctness of the Eq. (17) can be proved by the following equalities

  \[
  H(\delta^* + cy + y_R + fy_{Pr} \parallel m) = H ((\sigma + \tilde{\delta} - D) G + cx_P G + x_R G + fy_{Pr} \parallel m) \\
  = H ((\sigma + k + f^*S_{Pr} + D - D) G + cx_P G + x_R G + fy_{Pr} \parallel m) \\
  = H ((\sigma + \omega + (\gamma - f)S_{Pr}) G + cx_P G + x_R G + fy_{Pr} \parallel m) \\
  = H ((\omega + cx_P) G + \sigma G + \gamma S_{Pr} G + x_R G - f S_{Pr} G + fy_{Pr} \parallel m) \\
  = H ((\omega + cx_P) G + \sigma G + \gamma S_{Pr} G + x_R G - f S_{Pr} G + f S_{Pr} G \parallel m) \\
  = H (d + (\sigma + x_R)G + \gamma S_{Pr} G \parallel m) \\
  = H (d + (\sigma + x_R)G + \gamma y_{Pr} \parallel m) \\
  = H (\tilde{\rho} \parallel m) \\
  = f.
  \]

- **Identifiability:** In the agreement, when \(U_o\) sends \((\rho, \delta)\) to \(U_P\), \(U_o\) keep \(\rho\) and \(U_P\) together. When \(U_o\) meets a valid strong blind signature, it is easily to identify \(U_P\) through \(\delta\). In addition, there is the information of \((\rho, y_P)\) in the equation, through which \(U_o\) also can identify \(U_P\).
• **Nonrepudiation:** In this scheme the original signer does not know the proxy signer’s secret key $x_P$. Moreover, the proxy signer does not know the original signer’s secret key $x_o$. Thus, neither the original signer nor the proxy signer can sign in place of each other.

• **Unlinkability:** The proxy linkability holds if there is a conjunction between $(d, f^*\delta)$ and $(m, m_w, \delta^*, f)$ $d$ is only in Eq.(??) and relate to $f$ through equation Eq.(??). Proxy signer cannot find out the value of $d$ as it is masked by two random numbers $\sigma$ and $\gamma$. Hence, the proposed scheme satisfies the unlinkability.

• **Reliability:** The private key of proxy signature is a function value of $x_o$ and $x_P$, i.e., it relies on $x_o$ and $x_P$.

• **Partial blindness:** The partial blindness property of all signatures issued by the signer. The signatures contain clear common information according to a predefined format negotiated and agreed by all the requester and the proxy signer. Clearly, the requester is unable to change or remove the embedded information $c$ while keeping the verification of signature successful. In the proposed scheme, the requester $U_R$ has to submit the common information $c$ to the proxy signer. The proxy signer after that computes and sends $d = (\omega + cx_P)G(\text{mod} n)$ to $R$. If $U_R$ can successfully change or remove the common information $c$ from the signature, and then he/she can compute $d = (\omega + cx_P)G(\text{mod} n)$. However, deriving the secret key $x_P$ is an extremely difficult and time consuming.

• **Distinguishability:** As the proxy blind signature $(m, m_w, \delta^*, f)$ on the message $em$, contains $m_w$ (message warrant) anyone can easily differentiate between the proxy blind signature and normal signature. Hence, it satisfies the distinguishability property.

• **Unforgeability:** An adversary (including the original signer and the requester) wants to impersonate the proxy signer to sign the message $m$ He can intercept the delegation information $(\rho, \delta)$ but he cannot obtain the proxy secret key $S_{P_r}$. So for this scheme forgery is extremely hard.

If the original signer have an intention to forge a proxy blind signature with forgery attack for the message $\bar{m}$, he/she has to create a secret key $\bar{S}_{P_r}$ and calculate $\bar{y}_o \equiv \bar{S}_{P_r}G(\text{mod} n)$ consequently, the original signer must compute

$$\delta^* + y_P + y_R + y_{P_r} = d + (\sigma + x_R)G + \gamma y_{P_r}(\text{mod} n).$$
By using the equation \( d = (\omega + cx)G \) to equation \( \rho_s = \omega_sG \), the original signer has

\[
(\sigma + \delta - D + x_P + x_R)G + fy_{Pr} \equiv d + (\sigma + x_R)G + \gamma y_{Pr}(modn).
\]

Then

\[
(\gamma - f) \bar{S}_{Pr}G \equiv (\gamma - f)y_{Pr}.
\]

To find the value \( \bar{S}_{Pr} \) of original signer must find a solution to the above equation \( \bar{y}_o \equiv \bar{S}_{Pr}G \) which is ECDLP. Thus, the original signer fails to forge a signature.

The receiver cannot forge the signature after receiving \((m, m_w, \delta^*, f)\) on message \(m\). When a receiver tries to forge a signature \((\bar{m}, \delta^*, \bar{f})\) for message \(\bar{m}\). He/she must verify that the equation given below is correct

\[
\delta^* + y_P + y_R + fy_{Pr} = d + (\sigma + x_R)G + \gamma y_{Pr}(modn).
\]

By using the equation \( d = (\omega + cx)G \) to equation \( \rho_s = \omega_sG \) he has

\[
\delta^* + y_P + y_R + fy_{Pr} = (\sigma + \delta - D)G + x_PG + x_RG + \bar{f}y_{Pr}
\]

\[
= (\sigma + \delta - D + x_P + x_R)G + \bar{f}S_{Pr}G
\]

\[
= d + (\sigma + (\gamma - f)S_{Pr} + x_R)G + \bar{f}S_{Pr}G
\]

\[
= d + (\sigma + x_R)G + \gamma y_{Pr}.
\]

From the above we can get,

\[
(\gamma - f) S_{Pr}G + (S_{Pr}\bar{f}) G \equiv (S_{Pr}\gamma)G \equiv (modn)
\]

This cannot hold true, as \( f \neq \bar{f} \) Therefore the receiver fails to forge a valid partially proxy blind signature on message \(\bar{m}\)

### 4.2. Performance Evaluation

This section shows that the computational complexity performance of the proposed SPPBS scheme. For facilitating the computational complexity, we denote the performance evaluation notations as follows:

- \( T_{mul} \) : the time complexity for executing the modular multiplication,
- \( T_{exp} \) : the time complexity for executing the modular exponentiation,
- \( T_{add} \) : the time complexity for executing the modular addition,
• $T_{ec-mul}$: the time complexity for executing the elliptic curve multiplication,

• $T_{ec-add}$: the time complexity for executing the elliptic curve addition,

• $T_h$: the time complexity for executing the hash-function.

The performance of our scheme is described in terms of computational complexity and communication costs. We ignore the negligible time performing for modular addition. The performance of the proposed signature is summarized as follows: The computational complexity for the proxy delegation phase, partially blinding phase and extraction and verification phase is shown in Table 1. The last column converts various operation units to $T_{mul}$, where

$$T_{exp} \approx 240T_{mul}, \quad T_{ec-mul} \approx 29T_{mul} \quad \text{and} \quad T_{ec-add} \approx 0.12T_{mul}, \quad T_h \approx 4T_{mul},$$
given by Koblitz et al [8]. Finally, the communication costs or size of parameters of the scheme (both signature generation and verification) is $13|n|$, where $|a|$ denotes the bitlength of $a$.

<table>
<thead>
<tr>
<th>Items time complexity</th>
<th>Complexity in $T_{mul}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Proxy delegation phase</td>
<td>$2T_{ec-mul} + 2T_{mul}$</td>
</tr>
<tr>
<td></td>
<td>$+T_{ec-add} + 2T_h$</td>
</tr>
<tr>
<td>Partially blinding phase</td>
<td>$5T_{ec-mul} + 3T_{mul}$</td>
</tr>
<tr>
<td></td>
<td>$+3T_h + 3T_{ec-add}$</td>
</tr>
<tr>
<td>Extraction and verification phase</td>
<td>$3T_{ec-mul} + 3T_{ec-add} + T_h$</td>
</tr>
<tr>
<td>Communication costs</td>
<td>$13</td>
</tr>
</tbody>
</table>

Table 1: Performance evaluation

5. Conclusions

In this paper, we proposed a new SPPBS scheme based on elliptic curve discrete logarithm. This proposed scheme holds all the security properties of both proxy and partially blind signature scheme. The security of the proposed schemes is hardness of solving ECDLP. When an abuse of a proxy is conducted in proposed
scheme, an original signer can identify the deviating proxy signer and terminate the abused proxies before the specified delegation time. In addition, from the table 1 we can see that our scheme has secure key exchange, less storage requirement, scalability and low complexity thus makes it very efficient so the result show that the proposed scheme is very effective for many applications.

References


