Proxy Blind Multi-signature Scheme using ECC for handheld devices

Jayaprakash Kar

Department of Information Technology, Al Musanna College of Technology
Sultanate of Oman
jayaprakashkar@yahoo.com

Abstract. A proxy blind signature scheme is a special form of blind signature which allowed a designated person called proxy signer to sign on behalf of two or more original signers without knowing the content of the message or document. It combines the advantages of proxy signature, blind signature and multi-signature scheme. This paper describes an efficient proxy blind multi-signature scheme. The security of the proposed schemes is based on the difficulty of breaking the one-way hash function and the elliptic curve discrete logarithm problem (ECDLP). 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. This scheme utilizes a trusted third party called certificate authority to ensure that signatures can only be generated during valid delegation period. It satisfies the security properties of both proxy and blind signature scheme.

Keywords: ECDLP, blind signature, multi-signature, proxy signature.

1 Introduction

Blind signature scheme was first introduced by Chaum [4]. It is a protocol for obtaining a signature from a signer, but the signer can neither learn the messages he/she nor the signatures the recipients obtains afterwards. In 1996, mamo et al proposed the concept of proxy signature [3]. In proxy signature scheme, the original signer delegates his signing capacity to a proxy signer who can sign a message submitted on behalf of the original signer. A verifier can validate its correctness and can distinguish between a normal signature and a proxy signature. In multi-proxy signature scheme, an original signer is allowed to authorize a group of proxy members to generate the multi signature on behalf of the original signer. In 2000, Hwang et al. proposed the first multi-proxy signature scheme [5]. A proxy blind signature scheme is a digital signature scheme that ensures the properties of proxy signature and blind signature. In a proxy blind signature, an original signer delegates his signing capacity to proxy signer.

2 Preliminaries

2.1 Notations

Common notations used in this paper as follows.

- p : the order of underlying finite field.
- F_p : the underlying finite field of order p
- E : elliptic curve defined on finite field F_p with large order.
- G : the group of elliptic curve points on E.
- P : a point in $E(F_p)$ with order n , where n is a large prime number.
- $H(\cdot)$: a secure one-way hash function, where
- Let $U_O = U_1, U_2 \ldots U_{m_1}$ and $S_P = S_1, S_2 \ldots S_{m_2}$ be groups of m_1 original signers and m_2 proxy signers respectively.
- ID_{U_i} is the identity of the user U_i, $\forall i = 1, 2 \ldots m_1$;
- ID_{S_j} is the identity of the proxy signer S_j, $\forall j = 1, 2 \ldots m_2$
- (d_i, Q_i) : the private/public key pair of the original signers U_i, $1 \leq i \leq n$ where $Q_i = d_i \cdot P$
- M_w : A proxy warrant that contains information about identities of the original signers and proxy signer, delegation period etc.
- $\|$: Concatenation operation between two bit stings.

2.2 The finite field F_p

Let p be a prime number. The finite field F_p is comprised of the set of integers $0, 1, 2, \ldots p - 1$ with the following arithmetic operations [6] [7] [8]:

- Addition: If $a, b \in F_p$, then $a + b = r$, where r is the remainder when $a + b$ is divided by p and $0 \leq r \leq p - 1$. This is known as addition modulo p.
- Multiplication: If $a, b \in F_p$, then $a.b = s$, where s is the remainder when $a.b$ is divided by p and $0 \leq s \leq p - 1$. This is known as multiplication modulo p.
- Inversion: If a is a non-zero element in F_p, the inverse of a modulo p, denoted a^{-1}, is the unique integer $c \in F_p$ for which $a.c = 1$.

2.3 Elliptic Curve over F_p

Let $p \geq 3$ be a prime number. Let $a, b \in F_p$ be such that $4a^3 + 27b^2 \neq 0$ in F_p. An elliptic curve E over F_p defined by the parameters a and b is the set of all solutions $(x, y), x, y \in F_p$, to the equation $y^2 = x^3 + ax + b$, together with an extra point O, the point at infinity. The set of points $E(F_p)$ forms a Abelian group with the following addition rules [10]:

1. Identity : $P + \mathcal{O} = \mathcal{O} + \mathcal{P} = \mathcal{P}$, for all $P \in E(F_p)$
2. Negative : if $P(x, y) \in E(F_p)$ then $(x, y) + (x, -y) = \mathcal{O}$, The point $(x, -y)$ is dented as -P called negative of P.
3. Point addition: Let $P((x_1, y_1), Q(x_2, y_2) \in E(F_p)$,then $P + Q = R \in E(F_p)$ and co-ordinate $(x_3, y_3)$of R is given by $x_3 = \lambda^2 - x_1 - x_2$ and $y_3 = \lambda(x_1 - x_3) - y_1$ where $\lambda = \frac{y_2 - y_1}{x_2 - x_1}$
4. Point doubling : Let $P(x_1, y_1) \in E(K)$ where $P \neq -P$ then $2P = (x_3, y_3)$ where $x_3 = (\frac{3x_1^2 + a}{2y_1})^2 - 2x_1$ and $y_3 = (\frac{3x_1^2 + a}{2y_1})(x_1 - x_3)$- y_1

Definition 1. Elliptic Curve Discrete Logarithm Problem (ECDLP) *Given an elliptic curve E defined over a finite field F_p,a point $P \in E(F_p)$ of order n, and a point $Q \in < P >$,find the integer $l \in [0, n - 1]$such that $Q = l \cdot P$. The integer l is called discrete logarithm of Q to base P,denoted $l = log_p Q$.*

3 Proxy Blind signature

3.1 Model of Proxy Blind Signature

The schemes consist of delegation capability generation, delegation capability verification, proxy key generation, proxy blind signature generation and proxy signature verification [2]. The participant involve in this model are:

- An original signer, who delegates her signing capability to a proxy signer.
- A proxy signer generates a blind signature on behalf of the original signer. That means the original signer will get a signature from proxy signer without revealing the contents of the message.
- A verifier, who verifies the proxy signature and decides to accept or reject.
- A trusted party called certificate Authority, who certifies the public key.

An original signer selects a private key d_O and computes her public key Q_O as

$$Q_O \leftarrow \mathcal{KG}_{ECDLP}(params - ECDLP, d_O)$$

A proxy signer selects a private key d_P and computes his public key Q_P as

$$Q_P \leftarrow \mathcal{KG}_{ECDLP}(params - ECDLP, d_P).$$

- Delegation capability generation: It takes params ECDLP, original signer chosen parameters (k_O, r_O), original signer private key d_O, a warrant w as input and outputs signature σ_O on w. Procedurally,

$$\sigma_O \leftarrow \mathcal{S}_{ECDLP}(params - ECDLP, (k_O, r_O), d_O, w)$$

- Delegation capability verification: It takes params-ECDLP Q_O, w, σ_O as input and outputs **Result**, where **Result** $\in \{Valid, Invalid\}$.
- Proxy key generation(PKeyGen): It takes params-ECDLP, σ_O, d_P and random number as input; and outputs proxy key γ_P.
- Proxy signature generation: It takes params-ECDLP, proxy key γ_P and message m as input and outputs signature σ_P on m *i.e*

$$\sigma_P \leftarrow \mathcal{S}_{ECDLP}(params - ECDLP, \gamma_P, m)$$

- Proxy signature verification: It takes params-ECDLP Q_O, Q_P, m and σ_P as input and outputs **Result**, *i.e*

$$\textbf{Result} \leftarrow \mathcal{V}(params - ECDLP, (Q_O, Q_P), \sigma_P, m).$$

3.2 Phases of the proposed scheme

The scheme consists of the following five phases:

- Initialization
- Registration
- Proxy key generation
- Proxy blind multi-signature scheme
- Signature verification

The operation of each phase is described below.

Initialization Certificate Authority select random number $d \in [1, n-1]$ which will be the private key and compute the public key as $Q = \ d \cdot P$ and publishes Q, P and secure one way hash function $H(\cdot)$.

Registration Here all the original signers $U_i, 1 \leq i \leq m_1$ and proxy signer $S_j, 1 \leq j \leq m_2$ has to register with the CA as per the following steps.

- Step-I : Each original signer $U_i, 1 \leq i \leq m_1$ select random number u_i from $[1, n-1]$ and computes

$$R_i = u_i \cdot P, R = \sum_{i=1}^{m_1} R_i \tag{1}$$

$$\alpha_{U_i} = \ (R)_x u_i + \mathcal{H}(M_w, (R)_x, ID_{U_i}) \cdot (Q_i)_x d_i \mod n \tag{2}$$

where $(R)_x$ denotes x-co-ordinate of the point $R \in E(F_p)$ After computing each original signer U_i send respective α_{U_i} via public channel to the certificate authority CA.

- Step-2 : For each received $\alpha_{U_i}, 1 \leq i \leq m_1$, CA verifies whether the following equation holds:

$$\alpha_{U_i} \cdot P = \ (R)_x \cdot R_i + \mathcal{H}(M_w, (R)_x, ID_{U_i})(Q_i)_x \cdot Q_i, \forall i = \ 1, 2 \ldots m_1 \tag{3}$$

If it holds, then CA calculates α_{U_O} as their proxy shares as

$$\alpha_{U_O} = \sum_{i=1}^{m_1} \alpha_{U_i} \mod n \tag{4}$$

Then CA broadcast $(M_w, R, \alpha_{U_O}, ID_i)$ to proxy signer S_j $\forall j = \ 1, 2 \ldots m_2$.
After receiving each proxy signer S_j will verify whether the following equation holds.

$$\alpha_{U_O} \cdot P = \ (R)_x \cdot R + \sum_{i=1}^{m_1} \mathcal{H}(M_w, (R)_x, ID_{U_i}) \sum_{i=1}^{m_1} (Q_i)_x \cdot Q \tag{5}$$

If it does, each proxy signer S_j uses α_{U_O} as her proxy share and use to generate proxy key pairs with the original signers. To generate proxy signing private and public key she has to follow the following steps.

Proxy key pair generation In this phase, all the original signers will provide their signing capability to all the designated proxy signers S_j. All the original signers and proxy signers will jointly generate a proxy key pair for proxy signer such that only proxy signer knows the value of proxy signing private key $\tilde{d_j}$ and computes the public key $\tilde{Q_j}$. There is no need of any secure channel for communication between original signers and proxy signer. Following are the steps to generate proxy key pairs $(\tilde{d_j}, \tilde{Q_j}), \forall j = 1, 2 \ldots m_2$.

- Step-II : Each proxy signer S_j selects random number $t_j \in [1, n-1]$ and computes the proxy signing private key along with the original signers as

$$\tilde{d_j} = (t_j + \alpha_{U_j}) \mod n, \forall j = 1, 2 \ldots m_2 \tag{6}$$

  The corresponding public key will be

$$\tilde{Q_j} = \tilde{d_j} \cdot P \tag{7}$$

Proxy Blind Multi-signature generation The proxy blind multi-signature generation follows the following steps

- Step-I: Each proxy signer S_j randomly selects $r_j \in [1, n-1]$ and computes

$$T_j = r_j \cdot P, T = \sum_{j=1}^{m_2} T_j \tag{8}$$

$$\gamma_{S_j} = (T)_x \cdot r_i + \mathcal{H}(M_w, (T)_x, ID_{S_j})(Q_i)_x d_j \tag{9}$$

  Then it send to the requester R.
- Step-II : To get a blind signature of message M from each proxy signer S_j, the requester chooses two random number c_1 and c_2 from $[1, n-1]$ and computes the followings

$$Z_j = -c_1 c_2 \cdot \tilde{Q_j} + (c_1 \gamma_{S_j} + c_2) \cdot P \tag{10}$$

$$e = \mathcal{H}(Z_j \| M), e^* = e c_1^{-1} - c_2 \mod n \tag{11}$$

  Then the requester R delivers e^* to the proxy signer S_j.
- Step-III: After receiving e^*, each proxy signer S_j computes $\tilde{s_j}$ as follows

$$\tilde{s_j} = -e^* \cdot \tilde{d_j} + \gamma_{S_j} \mod n \tag{12}$$

  then it sends to R
- Step-IV : After receiving $\tilde{s_j}$, the requester R computes s_j for each proxy signer as

$$s_j = c_1 \tilde{s_j} + c_2 \mod n \tag{13}$$

  The individual proxy blind multi-signature is $\psi_j = (M_w, \alpha_{U_j}, M, e, s_j)$, $j = 1, 2, \ldots m_2$.

Signature verification Any person can verifies the validity of the signature by the following equation.

$$e = \mathcal{H}((s_j \cdot P + e \cdot \tilde{Q}) \| M) \tag{14}$$

If it is true, the verifier will accept it is a valid proxy blind multi-signature, otherwise reject it.

4 Security properties

The security properties for a secure blind multi-signature scheme are as follows

- **distinguishability** : The proxy blind multi-signature must be distinguishable from the ordinary signature.
- **Strong unforgeability**: Only the designated proxy signer can create the proxy blind signature for the original signer.

- **Non-repudiation**: The proxy signer can not claim that the proxy signer is dispute or illegally signed by the original signer.
- **Verifiability**: The proxy blind multi-signature can be verified by everyone. After verification, the verifier can be convinced of the original signer's agreement on the signed message.
- **Strong undeniably**: Due to fact that the delegation information is signed by the original signer and the proxy signature are generated by the proxy signer's secret key. Both the signer can not deny their behavior.
- **Unlinkability**: When the signer is revealed, the proxy signer can not identify the association between the message and the blind signature he generated.
- **Secret key dependencies**: Proxy key or delegation pair can be computed only by the original signer's secret key.
- **Prevention of misuse** : The proxy signer cannot use the proxy secret key for purposes other than generating valid proxy signatures. In case of misuse, the responsibility of the proxy signer should be determined explicitly.

5 Correctness

Theorem 1 *The proxy blind signature $\psi_j = (M_w, \alpha_{U_j}, M, e, s_j)$, $j = 1, 2, \ldots m_2$ is universally verifiable by using the system parameters.*

Proof: The correctness of the signature is verified by the equation 14
To prove $e = \mathcal{H}((s_j \cdot P + e \cdot \tilde{Q})\|M)$, $\forall j = 1, 2, \ldots m_2$, we have to show $s_j \cdot P + e \cdot \tilde{Q}_j = -c_1c_2 \cdot \tilde{Q}_j + c_1\gamma_{s_j} \cdot P + c_2$. Since $e = \mathcal{H}(Z_j\|M)$, where $Z_j = -c_1c_2 \cdot \tilde{Q}_j + (c_1\gamma_{s_j} + c_2) \cdot P$.

$$\begin{aligned}
s_j \cdot P + e \cdot \tilde{Q}_j &= (c_1\tilde{s}_j + c_2) \cdot P + e \cdot \tilde{d}_j \cdot P \\
&= c_1\tilde{s}_j \cdot P + c_2 \cdot P + e\tilde{d}_j \cdot P \\
&= \{c_1(-e^*\tilde{d}_j + \gamma_{S_j}) + c_2\} \cdot P + e\tilde{d}_j \cdot P \\
&= -c_1e^*\tilde{d}_j \cdot P + c_1\gamma_{S_j} \cdot P + c_2 \cdot P + e\tilde{d}_j \cdot P \\
&= -c_1(ec_1^{-1} + c_2)\tilde{d}_j \cdot P + c_1\gamma_{S_j} \cdot P + c_2 \cdot P + e\tilde{d}_j \cdot P \\
&= -e\tilde{d}_j \cdot P - c_1c_2\tilde{d}_j \cdot P + c_1\gamma_{S_j} \cdot P + c_2 \cdot P + e\tilde{d}_j \cdot P \\
&= -c_1c_2\tilde{d}_j \cdot P + c_1\gamma_{S_j} \cdot P + c_2 \cdot P \\
&= -c_1c_2 \cdot \tilde{Q}_j + c_1\gamma_{S_j} \cdot P + c_2 \cdot P \\
&= -c_1c_2 \cdot \tilde{Q}_j + (c_1\gamma_{S_j} + c_2) \cdot P \\
&= Z_j
\end{aligned}$$

6 Security Analysis

Let us discuss the security of the proposed scheme. Basically, the security of the proposed schemes is based on the difficulty of breaking the one-way hash function [9] and the elliptic curve discrete logarithm problem (ECDLP) [10].

Theorem 1. Distinguishablity*: Anyone can easily distinguish the proxy blind multi-signature from the normal signature.*

Proof: Proxy key is different from original signer's private key and proxy key created by different proxy signers are different from each other, any proxy signature is distinguishable from original signer's signature and different proxy signer's signatures are distinguishable. The proxy blind multi-signature $(M_w, \alpha_{U_j}, M, e, s_j)$ contains the warrant M_w and proxy public key $\tilde{Q}_j$ includes all original signer public key Q_i.

Theorem 2. (Strong Unforgeability)*:Any party can not forge a valid proxy blind multi-signature.*

Proof: The proxy blind signature is generated by the proxy signers using their respective proxy private key $\tilde{d}_j$, which is obtained by combining the random number t_j and α_{U_j} of all the original signers and proxy signers. From α_{U_j} is obtained from Eq.(2) which contain the original signer's private key d_j. If an adversary attempts to forge the signature, he has to obtain both t_j and d_j. For that he has to solve ECDLP.

Theorem 3. (Prevention of misuse)*: The proposed scheme can prevent proxy key pair misuse.*

Proof: because the warrant M_w includes identity information for all original signers U_i and proxy signers S_j. Therefore it prevent the proxy key pair $(\tilde{d_j}, \tilde{Q_j})$ misuse.

Theorem 4. (Non-repudiation)*: The proposed scheme provides non-repudiation property.*

Proof: Neither the original signer nor any proxy signer obtains the private key of any other party. During the verification of a valid proxy blind multi-signature, the verifier can confirm the original signer's agreement in signature and involvement of the proxy signer into it because the proxy signature public key $\tilde{Q_j} = \tilde{d_j} \cdot P$ contains the public keys of all the original signers and proxy shares α_{U_j} uses by the proxy signers.

Theorem 5. (Unlinkability)*: The proposed scheme provides proxy unlinkability property.*

Proof: The proxy blind multi-signature ψ_j is generated by the parameters $(M_w, \alpha_{U_j}, M, e, s_j)$, $j = 1, 2, \ldots m_2$. For delegation, the tuple $(M_w, \alpha U_j)$ are provided by all the original signers. The proxy unlinkability holds if and only if there is no conjunction between $(\gamma_{S_j}, e^*, \tilde{s_j})$ and $(M_w, \alpha_{U_j}, M, e, s_j)$ as shown from Eq. (8) to (13). Moreover, the value γ_{S_j} is only included in Eq.(10) and connected to e through Eq.(11). For this, one must be able to compute Z_j which however is masked with two random numbers c_1 and c_2. Similarly, e^* and $\tilde{s_j}$ may be associated with the signature through Eq.(11) to Eq.(13). They fail again due to the use of random numbers. So the proposed scheme provides unlinkability.

7 Conclusion

In this article, an efficient proxy blind-multi signature have be proposed. It satisfies the security properties of both proxy and blind signature scheme. The security of the proposed schemes is based on the difficulty of breaking the one-way hash function and the elliptic curve discrete logarithm problem (ECDLP). The attractiveness of ECC will increase relative to other public-key cryptosystems as computing power improvements force a general increase in the key size. The benefits of this higher-strength per-bit include higher speeds, lower power consumption, bandwidth savings, storage efficiencies, and smaller certificates. Therefore this can be implemented in handheld devices such as smart card, PDA etc. The primary reason for the attractiveness of ECC over systems such as RSA and DSA is that the best algorithm known for solving the underlying mathematical problem namely, the ECDLP takes fully exponential time. In contrast, sub-exponential time algorithms are known for underlying mathematical problems on which RSA and DSA are based, namely the integer factorization (IFP) and the discrete logarithm (DLP) problems. This means that the algorithms for solving the ECDLP become infeasible much more rapidly as the problem size increases more than those algorithms for the IFP and DLP. For this reason, ECC offers security equivalent to RSA and DSA while using far smaller key sizes.

References

1. Min-Shiang Hwang, Shiang Tzeng and C.Tsai Generalization of Proxy signature based on elliptic curves *"Computer Standards & Interfaces" , 26 (2004) pp. 73-84*
2. M. Das, A. Sexsena and D.Pathak Algorithms and Approaches of Proxy Signature : A Survey *"International Journal of Network Security" Vol-3, pp. 204-283*
3. M.Mambo, K.Usda and E.Okamoto Proxy signature: Delegation of power to sign messages *"IEICE Transaction on Fundamentals", E79-A(1996), pp.1338-1353*, 1996
4. D.Chaum Blind Signature for Untraceable Payments, *In Crypto 82, New York, Plenum Press, pp.199-203*, 1983
5. S.J.Hwang and C.H.Shi A Simple multi-signature scheme, *"Proceeding of 10th National conference on Information Security, Taiwan"*, 2000.
6. N. Koblitz. *A course in Number Theory and Cryptography ,2nd edition* Springer-Verlag-1994
7. K. H Rosen *"Elementary Number Theory in Science and Communication", 2nd ed., Springer-Verlag, Berlin, 1986.*
8. A. Menezes, P. C Van Oorschot and S. A Vanstone *Handbook of applied cryptography. CRC Press, 1997.*

9. D. Hankerson, A .Menezes and S.Vanstone. *Guide to Elliptic Curve Cryptography, Springer Verlag, 2004.*
10. *"Certicom ECC Challenge and The Elliptic Curve Cryptosystem" available :http://www.certicom.com/index.php.*
11. Dwork C., Naor M and Sahai A *Concurrent zero-knowledge, in Proceedings of 30th ACM STOC'98, 409-418,1998*
12. Abdalla M., Bellare M. and Rogaway P *The oracle Diffie-Hellman assumptions and an analysis of DHIES, in Topics in Cryptology - CT-RSA 2001, LNCS, 2020, 143-158,2001*
13. Aumann, Y. and Rabin, M. *Authentication, enhanced security and error correcting codes, in Advances in Cryptology - Crypto'98, LNCS, 1462, 299-303.*
14. Diffie W and Hellman M.E 1976 *directions in cryptography, IEEE Transactions on Information Theory, 22, 644-654, 1976*
15. Shi Y and Li J *2005, Identity-based deniable authentication protocol, Electronics Letters, 41,241-242, 2005*
16. Shoup V Sequences of games: a tool for taming complexity in security proofs, in Cryptology ePrint Archive: Report 2004/332, available at: http://eprint.iacr.org/2004/332