Provable Secure Multi-Proxy Signature Scheme without Bilinear Maps

Namita Tiwari and Sahadeo Padhye
(Corresponding author: Sahadeo Padhye)

Department of Mathematics, Motilal Nehru National Institute Of Technology
Allahabad (U.P.), India.
(Email : sahadeomathrsu@gmail.com)

(Received July 06, 2013; revised and accepted Oct. 10 & Nov. 7, 2013)

Abstract

Multi-proxy signature (MPS) allows an original signer to authorize a group of proxy signers as his proxy agent to sign co-operatively a message. As per the literature, the relative computation cost of a pairing is approximately more than ten times of the scalar multiplication over elliptic curve group which indicates that pairing is a very expensive operation. Moreover the map-to-point function is also very expensive. Therefore, we propose a new MPS scheme without pairings having general cryptographic hash function after formalizing a security model. Our proposal is provable secure and much efficient than previously proposed schemes in practice.

Keywords: Bilinear pairings, digital signature, elliptic curve discrete log problem, multi-proxy signature, random oracle model

1 Introduction

The concept of proxy signature was firstly introduced by Mambo et al. [13], to sign the messages on behalf of original signer. In a proxy signature scheme, an authorized person, called the proxy signer, is delegated by the original signer to generate a proxy signature on behalf of the original signer. To delegate the signing rights, a warrant message is used which consist of the identity of original as well as proxy signer's group, delegation period, information about the message etc. Original signer generates the delegation by signing the message warrant. Proxy signatures can be verified using a modified verification equation such that the verifier can be convinced that the signature is generated by the authorized proxy entity of the original signer. On the other hand, proxy signature is needed in some other forms also that are described in the article [17] in detail. For example, two or more vice presidents can cooperatively make a significant decision or sign an important document on behalf of the president in his absence. MPS is the solution of such a problem which allows the original signer to delegate his/her signing power to a group of proxy signers such that all proxy signers must cooperatively generate a valid proxy signature.

On the other side, if a group of original signer want to authorize a proxy signer to generate a signature on behalf of the original signer group, to handle such a situation in 2000, Yi et al. [26] firstly proposed proxy multi-signature (PMS) scheme. After that, some other variants multi-proxy multi-signature (MPMS) schemes have also been proposed [7].

Since proxy signature appeared, many new proxy signature schemes [4, 5, 6, 8, 19, 21, 27] have been proposed. Motivated by the recent work [6], authors proposed the ID-based proxy multi-signature [16] and multi-proxy multi-signature [18] schemes without pairings. In this paper, we focus on MPS scheme. Till now, many MPS schemes [2, 10, 11, 20, 22, 23, 24, 25] *etc* from bilinear pairings and ElGamal type have been proposed. There are some literatures [1, 6] etc showing that the relative computation cost of a pairing operation is approximately more than ten times of the scalar multiplication over elliptic curve group. In addition, the map-to-point hash function is also very expensive cryptographic operation. Due to bilinear pairings and map-to-point hash function, the above schemes are less efficient and so not very applicable in practice. Therefore, schemes without bilinear pairings in general hash function setting with elliptic curve cryptography would be more appealing in terms of efficiency while maintaining the security.

Elliptic curve cryptography (ECC) was introduced by Koblitz [9] and Miller [14] independently in 1985 using the group of points on an elliptic curve defined over a finite field. Security of the cryptosystem based on ECC relies on elliptic curve discrete log problem (ECDLP). The main advantage of ECC is that it provides the same security level with smaller key size [12] than RSA and ElGamal cryptosystems. Smaller key means less management time

and smaller storage, which supplies convenience to realization by software and hardware. To achieve $1024-bits$ RSA level security, $512-bits$ supersingular elliptic curve and $160-bits$ non-supersingular elliptic curves are used in applications. In general, pairing is defined on the supersingular elliptic curve while the ECC without pairings uses non supersingular elliptic curves.

In this paper, we propose an efficient MPS scheme without bilinear pairings which has smaller key size than pairing based schemes. Proposed scheme is proven secure against adaptive chosen message attack [3] under ECDLP assumption. With the pairing-free realization, proposed scheme is much efficient than previous related schemes from pairings in practice. In addition, it is obviously much efficient in practice than ElGamal based MPS schemes since ECC provides the same security than ElGamal based cryptosystems at less bit parameters.

The rest of this paper is organized as follows. Some preliminary works are given in Section 2. The formal models of MPS scheme is described in Section 3. Our provable secure MPS scheme is presented in Section 4. We analyze the security of proposed scheme in Section 5. Section 6 presents the comparative analysis. Finally, conclusions are given in Section 7.

2 Preliminaries

2.1 Background of Elliptic Curve Group

An elliptic curve E over a prime finite field F_p (denoted by E/F_p) is the set of points (x, y) with $x, y \in F_p$ which satisfy the equation $y^2 = (x^3 + ax + b) \bmod p$, $a, b \in F_p$,point say $-R$. Then $P + Q$ is the reflected point $-R$. There is a together with an extra point $\{\infty\}$ (called the point at infinity). If the discriminant $\Delta = (4a^3 + 27b^2) \bmod p \neq 0$, equivalently, the polynomial $x^3 + ax + b$ has distinct factors then E/F_p is nonsingular i.e it does not have any cusp or node singularity. Therefore, we can define a binary operation (the point addition " + ") on the points of E/F_p as follows: Let $P, Q \in E/F_p$, l be the line joining P and Q (tangent line to E/F_p if $P = Q$), and R, the third point of intersection of l with E/F_p. Let l' be the vertical line through R which intersects the elliptic curve E/F_p at another problem that vertical line through P and $-P$ does not intersect elliptic curve E/F_p at a third point and we need a third point to define $P+(-P)$. Since there is no point in the plane that works, we create an extra point ∞ at infinity. Here ∞ is a point on every vertical line.

Thus elliptic curve with this binary operation " + " forms an additive abelian group $(E/F_p, \text{" + "}) = \{(x, y) : x, y \in F_p, E(x, y) = 0\} \cup \{\infty\}$. Let G be a cyclic additive subgroup of $(E/F_p, \text{"+"})$ with generator P of prime order n.

2.2 Mathematical Formulas for Addition on E/F_p

Suppose that we want to add the points $P_1 = (x_1, y_1)$ and $P_2 = (x_2, y_2)$ on the elliptic curve E as defined above.

Let the line connecting P_1 to P_2 be $L : y = mx + c$. Explicitly, the slope and y-intercept of L are given by

$$\begin{aligned} m &= \begin{cases} \frac{y_2-y_1}{x_2-x_1} \bmod p, & \text{if } P_1 \neq P_2 \\ \frac{3{x_1}^2+a}{2y_1} \bmod p, & \text{if } P_1 = P_2 \end{cases} \\ c &= (y_1 - mx_1) \bmod p. \end{aligned}$$

Now we find the intersection of E/F_P: $y^2 = (x^3+ax+b)$ $a, b \in F_p$, and L: $y = mx + c$ by solving $(mx + c)^2 = x^3 + ax + b$ under modulo p. We already know that x_1 and x_2 are solutions, so we can find the third solution x_3 by comparing the two sides of $x^3 + ax + b - (mx + c)^2 = (x-x_1)(x-x_2)(x-x_3) \bmod p$. Equating the coefficients of x^2, gives $m^2 = (x_1+x_2+x_3) \bmod p$ and hence $x_3 = (m^2 - x_1-x_2) \bmod p$. Then we compute y_3 using $y_3 = (mx_3+c) \bmod p$ and finally $P_1 + P_2 = (x_3, -y_3)$.

In Short: Addition algorithm for $P_1 = (x_1, y_1)$ and $P_2 = (x_2, y_2)$ on the elliptic curve E is:

1) If $P_1 \neq P_2$ and $x_1 = x_2$ then $P_1 + P_2 = \{O\}$.

2) If $P_1 = P_2$ and $y_1 = 0$ then $P_1 + P_2 = 2P_1 = \{O\}$.

3) If $P_1 \neq P_2$ (and $x_1 \neq x_2$), let $m = \frac{y_2-y_1}{x_2-x_1} \bmod p$ and $c = \frac{y_1x_2-y_2x_1}{x_2-x_1} \bmod p$.

4) If $P_1 = P_2$ and $y_1 \neq 0$, let $m = \frac{3{x_1}^2+a}{2y_1} \bmod p$ and $c = \frac{-x^3+ax+b}{2y} \bmod p$.

Then $P_1 + P_2 = ((m^2 - x_1 - x_2) \bmod p,\ (-m^3 + m(x_1 + x_2) - c) \bmod p)$. Scalar multiplication tP over E/F_p means $tP = P + P + ... + P(t \ \ times)$, that can be calculated using double-and-add method.

2.3 Complexity Assumption

Elliptic curve discrete logarithm problem (ECDLP): Given $x \in_R {Z_n}^*$ and P the generator of G and $Q \in G$, to compute x s.t. $Q = xP$ is called ECDLP and assumed to be intractable.

3 Formal Models of Multi Proxy Signature Scheme

The proposed model involves three parties: the original signer A, a set of l proxy signers $L = \{PS_1, PS_2, .., PS_l\}$, and a verifier. One of the proxy signers plays the role of clerk who combines all the partial proxy signatures and generates an MPS.

3.1 Definition of MPS Scheme

A MPS scheme is specified by the following polynomial-time algorithms.

- Setup: Given a security parameter k, this algorithm outputs the system parameters.
- Extract: It takes the security parameter k and outputs the secret-public key pair (sk_U, pk_U), for each user U participating in the scheme.
- DelGen: Given the systems' parameter, the original signer's private key and the warrant m_w to be signed, this algorithm outputs the delegation $W_{A\rightarrow PS_i}, (1 \leq i \leq l)$.
- DelVerif: The delegation verification algorithm, takes the original signer's public key, delegation $W_{A\rightarrow PS_i}, (1 \leq i \leq l)$ as inputs and verifies whether it is a valid delegation came from A.
- PKGen: The proxy key generation algorithm, takes $W_{A\rightarrow PS_i}, (1 \leq i \leq l)$ and some other secret information (for example, the secret keys of the executors) as inputs, and outputs a proxy signing key psk_{P_i} $\forall 1 \leq i \leq l$ for proxy signature.
- MPSign: The proxy signing algorithm, takes the proxy signing keys $psk_{P_i}, \forall 1 \leq i \leq l$ of all proxy signers and a message $m \in \{0,1\}^*$ as inputs, and outputs an MPS signature on behalf of A.
- MPVerif: The proxy verification algorithm, takes public keys of original signer, all proxy signers, and a proxy signature (m_w, σ, m, S) as inputs, and outputs 0 or 1. In the later case, (m, S) is a valid MPS for m by the proxy group L on behalf of the original signer A.

3.2 Security Model of MPS Scheme

We define the security of our MPS scheme under existential unforgeability against adaptive chosen message attack (EUF-ACMA) [3]. The security notion is based on the following game played between a challenger C and a probabilistic polynomial time adversary T under an experiment Exp_T^{MPS} of the adversary T.

- Setup: The challenger C runs this algorithm with input k and generates the public parameters. In addition C runs the Extract algorithm to obtain a public key pk and private key sk. The adversary T is given pk and system parameters while sk is kept secret.
- Queries. T can make the following queries adaptively to C.
  1) DelGen-query: T requests for the delegations with at-most q_s no of message warrant's for proxy signers with pk_{PS_i}, $(i = 1, 2, ...l)$ on behalf of original signer with pk_A adaptively. There exist a simulator S that simulates the DelGen oracle and outputs the corresponding valid delegations $W_{A\rightarrow PS_i}$ for each query.
  2) MPSign-query: T queries the signature oracle for at-most q_s no of messages under the obtained delegation $W_{A\rightarrow PS_i}$. There exist a simulator S that simulates the MPSign oracle and outputs the valid signature tuples.
- Output: Eventually, T outputs a tuple $(m_w,\ pk_A,\ pk_ps_i,\ \sigma,\ m,\ S)$, $(i = 1, 2, , l)$ and wins the game i.e Exp_T^{MPS} returns yes if
  1) Message warrant m_w and message m are not queried before for delegation and signature respectively.
  2) DelVerif $(pk_A, PK_ps_i; m_w, sigma) = valid$.
  3) MPVerif $(pk_A, PK_ps_i, m_w, sigma, m, S) = valid$.
  4) Otherwise returns No.

An MPS scheme is said to be existential delegation and signature unforgeable against adaptive chosen message attack (DS-EUF-ACMA), if for any polynomial-time adversary T, $Pr[Exp_T^{MPS}(k) = yes]$ is negligible.

4 Proposed Scheme

In this section, we present an MPS scheme without pairings. The proposed scheme involves three parties: the original signer A, a set of l proxy signers $L = \{PS_1, PS_2, .., PS_l\}$, and a verifier. One of the proxy signers plays the role of clerk who combines all the partial proxy signatures and generates an MPS on message m which confirms the warrant m_w. Our scheme mainly consists of the following seven algorithms.

- Setup: Takes a security parameter k, and returns the system parameters $\Omega = \{F_p, E/F_p, G, P, H_1, H_2, \}$ as defined in 2.1. $H_1 : \{0,1\}^* \times G \rightarrow Z_n^*$ and $H_2 : \{0,1\}^* \times G \rightarrow Z_p^*$ are two cryptographic secure hash functions.
- Extract: Each participant U of the scheme picks at random $sk_U \in Z_n^*$ and computes $pk_U = sk_U P$. Thus (sk_U, pk_U) is the (secret, public) key pair of user U.
- DelGen: This algorithm takes A's secret key sk_A and a warrant m_w as inputs, and outputs the delegation $W_{A\rightarrow PS_i}$, $1 \leq i \leq l$ as follows:
  1) Generates a random $a \in Z_n^*$, computes $K = aP$.
  2) Computes $h_{i_A} = H_2(m_w, K, pk_{PS_i})$, $h_A = \sum_1^l h_{i_A}$ and $\sigma = (h_A sk_A + a) \bmod n$.

  A sends delegation $W_{A\rightarrow PS_i} = \{pk_A, pk_{PS_i}, m_w, K, \sigma\}$ to each proxy signer PS_i, $1 \leq i \leq l$.

- DelVerif: To verify the delegation $W_{A\to PS_i}$ on warrant m_w, each proxy signer PS_i first computes $h_{i_A} = H_2(m_w, K, pk_{PS_i})$ and $h_A = \sum_1^l h_{i_A}$, then checks whether $\sigma P = h_A pk_A + K$ holds. Accepts if it is equal, otherwise rejects.

- PKGen: If PS_i accepts the delegation $W_{A\to PS_i}$, he computes the proxy signing key psk_{PS_i}, $1 \le i \le l$ as follows: $psk_{PS_i} = (\sigma h_p + sk_{PS_i}) \bmod n$, where $h_p = H_1(m_w, pk_A, K)$. Using psk_{PS_i}, these proxy signers can cooperate to sign any message m which confirms to m_w on behalf of the original signer A.

- MPSign: Each proxy signer $PS_i, (1 \le i \le l)$ chooses $a_i \in Z_n^*$, computes $N_i = a_i P$ and broadcasts his N_i to the other $l-1$ proxy signers. Then each PS_i computes $S_{PS_i} = (psk_{PS_i} + a_i h) \bmod n$ where $h = H_2(m, N)$, $N = \sum_1^l N_i$ and sends $(pk_A, pk_{PS_i}, K, m_w, m, S_{PS_i})$, $(1 \le i \le l, i \ne j,$ if PS_j is designated as clerk) to the clerk as his partial proxy signature. The clerk verifies the partial proxy signatures by checking the equation $S_{PS_i}P = h_p(h_A pk_A + K) + pk_{PS_i} + hN_i$, where $N = \sum_1^l N_i$, $h = H_2(m, N)$, $h_A = \sum_1^l H_2(m_w, K, pk_{PS_i})$ and $h_p = H_1(m_w, pk_A, K)$. If it holds, then he combines $S = \sum_1^l S_{PS_i}$ and sends the tuple $(pk_A, pk_{PS_i}, K, N, m_w, m, S), \forall 1 \le i \le l$ to verifier.

- MPVerif: To verify the signature (pk_A, pk_{PS_i}, K, N, m_w, m, S), $\forall 1 \le i \le l$ for message m, the verifier does as follows.

  Checks whether the message m confirms to the warrant m_w. If not, stop. Otherwise, continue. Checks whether the l proxy signers are authorized by the original signer in the warrant m_w. If not, stop. Otherwise, continue. Computes $h_{i_A} = H_2(m_w, K, pk_{PS_i})$, $h_A = \sum_1^l h_{i_A}$, $h_p = H_1(m_w, pk_A, K)$ and $h = H_2(m, N)$, then checks whether the equation: $SP = lh_p(h_A pk_A + K) + \sum_1^l pk_{PS_i} + hN$ holds. If holds then accepts otherwise rejects it.

Correctness. Since, $S_{PS_i}P = h_p(h_A pk_A + K) + pk_{PS_i} + hN_i$ and $N = \sum_1^l N_i$, we have,

$$\begin{aligned} SP &= \sum_1^l S_{PS_i}P \\ &= \sum_1^l [h_p(h_A pk_A + K) + pk_{PS_i} + hN_i] \\ &= lh_p(h_A pk_A + K) + \sum_1^l pk_{PS_i} + hN. \end{aligned}$$

5 Security Analysis

In this section, we will examine the security of our proposed scheme. Assume there is an adversary T who can break our proxy signature scheme (say $\sum$). We will construct a polynomial-time algorithm F that, by simulating the challenger C and interacting with T, solves the ECDLP.

Theorem 1. *Consider an adaptively chosen message attack in the random oracle model(ROM) against $\sum$. If there is an attacker T that can break $\sum$ with at most q_{H_2} H_2-queries and q_s signature queries within time bound t and non negligible probability ε. Then there exist an algorithm that solves ECDLP with non-negligible probability.*

Proof. Suppose an attacker T can break $\sum$ through adaptively chosen message attack then $Pr[Exp_T^{MPS}(k) = yes]$ is non negligible. We will show that using the ability of T and forking lemma [15], an algorithm F can be constructed for solving the ECDLP. Forking reduction technique works because the challenger sets the random oracle answers so that one set of questions from adversary are answered with a number of completely independent sets of answers.

For this purpose F sets $\{F_p, E/F_p, G, P, P_{pub}, H_1, H_2\}$ as system parameters and answers T's queries 3.2 as follows.

Case 1. (*Existential Delegation Unforgeable under Adaptive Chosen Message Attack*). The challenger C interacts with forger T and responds as follows.

- Setup: C starts to obtain public key pk and private key sk. The adversary T is given pk.

- DelGen-query: T is allowed to query the delegation oracle for $m_w, pk_A, pk_{PS_i}, \forall 1 \le i \le l$. There exist a simulator S that simulates the oracle and outputs (σ, K) that satisfies the equation $\sigma P = h_A pk_A + K$. Thus σ is a valid signature on m_w for pk_A.

- Output: If T can forge a valid delegation on warrant m_w without knowing the secret key with the probability $Pr[Exp_T^{MPS}(k) = yes] = \varepsilon \ge 10(q_{H_2}+1)(q_{H_2}+q_s)/2^k$ where m_w has not been queried to the delegation oracle (as Lemma 4 of [15] aims), then a replay of F with the same random tape but different choice of H_2 will output two valid delegations $\{pk_A, pk_{PS_i}, m_w, K, \sigma, h_A\}$ and $\{pk_A, pk_{PS_i}, m_w, K, \sigma', h_A'\}$.

Then we have

$$\sigma P = h_A pk_A + K \quad (1)$$

$$\sigma' P = h_A' pk_A + K. \quad (2)$$

From Equations (1) and (2), we have

$$(\sigma - \sigma')P = (h_A - h_A')sk_A P.$$

Let $u = \sigma - \sigma'$ and $v = (h_A - h_A')^{-1}$, then

$$sk_A = uv \bmod n.$$

Table 1: Cryptographic operation time (in milliseconds)

Operation	Modular exp.	O_P	M_P	M_E	H_M	General hash
Time	5.31	20.04	6.38	2.21	3.04	< 0.001

Table 2: Computational cost comparison

Scheme	Extract	DelGen	DelVerif	PKGen	MPSign	MPVerif	Total
Scheme [10]	$1M_P$	$1M_P + 1H_M$	$1H_M + 2O_P$	$1M_p + 1H_M$	$2M_P + 1H_M + 3O_p$	$1H_M + 2O_P$	$5M_P + 5H_M + 7O_P$
Scheme [23]	$1M_P$	$1M_P + 1H_M$	$1H_M + 2O_P$	$1M_p + 1H_M$	$2M_P + 1H_M + 3O_p$	$2M_P + 1H_M + 2O_P$	$6M_P + 5H_M + 7O_P$
Our scheme	$1M_E$	$1M_E$	$2M_E$	$0M_E$	$5M_E$	$4M_E$	$13M_E$

Table 3: Running time comparison (in ms)

Scheme	Extract	DelGen	DelVerif	PKGen	MPSign	MPVerif	Total
Scheme [10]	9.42	9.42	49.50	6.38	61.36	49.50	185.58
Scheme [23]	9.42	19.14	46.46	6.38	50.74	58.92	191.06
Our scheme	2.21	2.21	6.63	≈ 0	15.47	13.26	39.78

According to Lemma 4 [15] the ECDLP can be solved with probability $\varepsilon^{'} \geq 1/9$ and time $t^{'} \leq 23\ q_{H_2} t/\varepsilon$.

Case 2. (*Existential Signature Unforgeable under Adaptive Chosen Message Attack*). From Case 1, it is clear that the adversary T can not generate a valid delegation. In this Case the challenger C interacts with forger T as follows.

- Setup: C starts to obtain public key pk and private key sk. The adversary T is given pk.
- MPSign-query: T is allowed to query the signature oracle for m under the delegation $W_{A \to PS_i} = \{pk_A, pk_{PS_i}, m_w, K, \sigma\}$. There exist a simulator S that simulates the oracle and generates a tuple (N, S) that satisfies the equation $SP = lh_p(h_A pk_A + K) + \sum_1^l pk_{PS_i} + hN$.
- Output: If T can forge a valid signature on message m with the probability $Pr[Exp_T^{MPS}(k) = yes] = \varepsilon \geq 10(q_{H_2} + 1)(q_{H_2} + q_s)/2^k$ where m has not been queried to the signature oracle, then a replay of F four times with the same random response but different choices of H_2, will output four valid signatures $(pk_A, pk_{PS_i}, K, N, m_w, m, S^j, h_A^j, h^j)$, $\forall 1 \leq i \leq l$ and $j = 1, 2, 3, 4$.

Then we have

$$S^j P = lh_p(h_A^j pk_A + K) + \sum_1^l pk_{PS_i} + h^j N. \qquad (3)$$

If sk_A, a, b, y denote elliptic curve discrete logarithms of $pk_A, K, \sum_1^l pk_{PS_i}$ and N respectively. Then from equation (3), we have

$$S^j = lh_p(h_A^j sk_A + a) + b + h^j y, \quad j = 1, 2, 3, 4.$$

Since, in the above four equations, the unknowns sk_A, a, b, y neither have any power nor multiplied together. So these equations are linear. We consider that with high probability the determinant of the system obtained by the above four linear equations is non zero and so these equations are linearly independent.

Therefore, there exist an algorithm F that solves the above four linearly independent equations, and outputs sk_A as the solution of the ECDLP with probability $\varepsilon^{'} \geq 1/9$ and time $t^{'} \leq 23 q_{H_2} t/\varepsilon$ (Lemma 4 [15]). □

6 Comparative Analysis

In this section, we will compare the efficiency of our scheme with the schemes [10, 23]. We use the running time of different cryptographic operations calculated by [6] in some cryptographic environment for such efficiency comparison as given in Table 1.

Where M_E, M_P, H_M, O_P stand for one ECC based scalar multiplication, pairing based scalar multiplication, Map-to-point hash function and pairing operation respectively.

Computational cost and running time analysis of our scheme with schemes [10, 23] are given in Tables 2 and 3, respectively.

From the above Table 2, it is clear that the running time of MPSign algorithm of our scheme is 14.54% of scheme [10] as well as of scheme [23]. Total running time of our scheme is 20.50% of scheme [10] and 17.85% of the

scheme [23].

Note: Although our proposed scheme is based on ECC, it does not use pairings. Therefore one can easily conclude by efficiency comparison that our proposal is much more efficient than other existing MPS schemes from pairings.

7 Conclusion

In this paper, we proposed an efficient provable secure multi-proxy signature scheme based on ECC without using pairings that also avoids the map-to-point hash function. For this proposal, we first defined a model and then proved the security of proposed scheme against adaptive chosen message attack under ECDL-assumption. Compared with previous schemes, the new scheme reduces the running time of signing algorithms heavily. Therefore, our scheme is more efficient and applicable than the previous related schemes in practice.

References

[1] L. Chen, Z. Cheng, and N. P. Smart, "Identity-based key agreement protocols from pairings," *International Journal of Information Security*, vol. 6, pp. 213–241, 2007.

[2] C. Feng and C. Zhenfu, "A secure identity-based multi-proxy signature scheme," *Computer and Electrical Engineering*, vol. 35, pp. 86–95, 2009.

[3] S. Goldwasser, S. Micali, and R. L. Rivest, "A digital signature scheme secure against adaptive chosenmessage attacks," *SIAM Journal of Computing*, vol. 17, no. 2, pp. 281–308, 1988.

[4] C. Gu and Y. Zhu, "Provable security of ID-based proxy signature schemes," in *Networking and Mobile Computing*, LNCS 3619, pp. 1277–1286, Springer-Verlag, 2005.

[5] C. Gu and Y. Zhu, "An efficient ID-based proxy signature scheme from pairings," in *Information Security and Cryptology*, LNCS 4990, pp. 40–50, Springer-Verlag, 2008.

[6] D. He, J. Chen, and J. Hu, "An ID-Based proxy signature schemes without bilinear pairings," *Annalas of Telicommunications*, vol. 66, no. 11-12, pp. 657–662, 2011.

[7] S. J. Hwang and C. C. Chen, "New multi-proxy multi-signature schemes," *Appllied Mathematics and Computation*, vol. 147, pp. 257–67, 2004.

[8] H. Ji, W. Han, and L. Zhao et al, "An identity-based proxy signature from bilinear pairings," in *Proceedings of WASE International Conference on Information Engineering*, pp. 14–17, 2011.

[9] N. Koblitz, "Elliptic curve cryptosystems," *Mathenmatics of Computation*, vol. 48, no. 177, pp. 203–209, 1987.

[10] S. Li and F. Zhang, "A new Multi-Proxy signature from bilinear pairing," *Journal of Electronics (China)*, vol. 24, no. 1, pp. 90–94, 2007.

[11] Z. Liu, Y. Hub, X. Zhang, and H. Maa, "Provably secure multi-proxy signature scheme with revocation in the standard model," *Computer Communication*, vol. 34, pp. 494–501, 2011.

[12] M. Lochter, J. Merkle, *Elliptic Curve Cryptography (ECC) Brainpool Standard Curves and Curve Generation*, RFC 5639, Mar. 2010.

[13] M. Mambo, K. Usuda, and E. Okamoto, "Proxy signatures: Delegation of the power to sign messages," *IEICE Transaction Fundamentals*, vol. E79-A(9), pp. 1338–1353, 1996.

[14] V. Miller, "Uses of elliptic curves in Cryptography," in *Proceedings of Advances in Cryptology-Crypto 85*, pp. 417–426, Santa Barbara, USA, Aug. 1985.

[15] D. Pointcheval and S. Jacque, "Security arguments for digital signatures and blind signatures," *Journla of Cryptology*, vol. 13, no. 3, pp. 361–396, 2000.

[16] N. Tiwari and S. Padhye, "An ID-Based proxy multi signature scheme without bilinear pairings," in *First International Conference on Security Aspects in Information Technology (InfoSecHiComNet '11)*, pp. 83–92, Haldia, India, Oct. 2011.

[17] N. Tiwari and S. Padhye, "Analysis on the generalization of proxy signature," *Security and Communication Network*, vol. 6, pp. 549–556, 2013.

[18] N. Tiwari, S. Padhye, and D. He, "Efficient ID-based multi-proxy multi-signature without bilinear maps in ROM," *Annalas of Telecommunication*, vol. 68, no. 3-4, pp. 231–137, 2013.

[19] A. Wang, J. Li, and Z. Wang, "A provably secure proxy signature scheme from bilinear pairings," *Journal of Electronnics (China)*, vol. 27, no. 3, pp. 298–304, 2010.

[20] T. S. Wu, C. L. Hsu, and H. Y. Lin, "Self-certified multi-proxy signature schemes with message recovery," *Jorunal of Zhejiang University Science*, vol. 10, no. 2, pp. 290–300, 2009.

[21] W. Wu, Y. Mu, and W. Susilo et al., "Identity-based proxy signature from pairings," in *The 4th International Conference, Authentic and Trusted Computing (ATC'07)*, pp. 22–31, Hong Kong, China, July 2007.

[22] L. Xiangxue and C. Kefei, "Id-based multi-proxy signature, proxy multi-signature and multi-proxy multi-signature schemes from bilinear pairings," *Appllied Mathematics and Computation*, vol. 169, pp. 437–450, 2005.

[23] L. Xiangxue, C. Kefei, and L. Shiqun, "Multi-proxy signature and proxy Multi-signature schemes from bilinear pairings," in *Parallel and Distributed Computing: Applications and Technologies (PDCAT'04)*, pp. 591–595, Singapore, Dec. 2004.

[24] C. Xiaofeng, Z. Fangguo, and K. Kwangjo, "ID-Based Multi-Proxy signature and blind multisignature from bilinear pairings," in *Proceeding of KIISC Conference*, p. 1119, Korea, Nov. 2003.

[25] Q. Xue and Z. Cao, "Improvement of multi-proxy signature scheme," in *Proceeding of IEEE Fourth International Conference on Computer and Information Technology*, pp. 450–455, Sep. 2004.

[26] L. J. Yi, G. Q. Bai, and G. Z. Xiao, "Proxy multisignature scheme: A new type of proxy signature scheme," *Electronics Letters*, vol. 36, no. 6, pp. 527–528, 2000.

[27] J. Zhang and W. Zou, "Another ID-based proxy signature scheme and its extension," *Wuhan University Journal of Natural Science*, vol. 12, pp. 133–136, 2007.

Namita Tiwari received her B.Sc. degree from C. S.J.M. University, Kanpur, India in 2006 and M.Sc. degree in Mathematics from Indian Institute of Technology, Kanpur, India in 2008. She did her Ph.D. from Motilal Nehru National Institute of Technology, Allahabad, India in 2013. She is a life member of Cryptology Research Society of India (CRSI). Her area of interest is Digital Signature.

Sahadeo Padhye received his B.Sc.and M.Sc. degree in Mathematics form Pt. Ravishankar Shukla University, Raipur. Chhattisgarh, India in 1999 and 2001. Council of Scientific and Industrial Research (CSIR), India has granted him Junior Research Fellowship (2002-2004). He did his Ph.D. form Pt. Ravishankar Shukla University, Raipur, India. He is a life member of Cryptology Research Society of India (CRSI)and Indian Mathematical Society and a member of International Association of Cryptologic Research (IACR). His area of interest is Public Key Cryptography based on elliptic curve and digital signature. Presently he is working as Assistant Professor in the Department of Mathematics, Motilal Nehru National Institute of Technology, Allahabad, India.