A Pairing Free Identity Based Proxy Signature Scheme

Hassan Elkamchouchi[1], Eman Abou El-kheir[2], and Yasmine Abouelseoud[3]
Alexandria University[1,3], Kafr El-Sheikh University[2], Egypt

Abstract—Digital signature scheme is a method for signing an electronic message. As such, a signed message can be transmitted over a computer network in an authenticated manner. This paper introduces two identity based (id-based) digital signature schemes. The first one, is a signature scheme and the second its extension to a proxy signature scheme in which the original signer delegates his signing rights to a proxy signer. Both schemes don't use the bilinear pairings in the Signcryption and unsigncryption phases. Also, both schemes are based on the elliptic curve discrete logarithm problem (ECDLP). Moreover, the proposed schemes achieve the standard security requirements. the performance of the both schemes is examined. The proposed id-proxy signature scheme reduced computational complexity compared to other scheme in literature.

Index Terms— ID-Based , Digital Signature, Proxy Signature, ECDLP, Without Bilinear Pairings

1 Introduction

Digital signatures offer source authentication in cryptography. To handle the situations arising in the digital world related to authentication, different types of digital signatures have been developed [1]. The concept of a proxy signature was first introduced by Mambo et al. [2] in 1996. In a proxy signature scheme, generally, there are two entities: an original signer and a proxy signer. The original signer can delegate his signing power to a proxy signer. The proxy signer can generate a valid signature on behalf of the original signer. Since then, many proxy signature schemes have been proposed [3, 4, 5, 6].

The concept of Identity based cryptography (IBC) was first introduced by Adi Shamir in 1984 [11]. Its primary innovation was its use of user identity attributes, such as email address, phone number, IP address instead of digital certificates for encryption and signature verification. This feature significantly reduces the complexity of a cryptography system by eliminating the need for generating and managing user certificates[12].

Identity based signature is similar identity based encryption. It consists of the four algorithms Set-up, Extract, Signature Generation and Signature verification. In this scheme signer first obtain her private key associated with her identity and then he generates a signature for message and sends it to receiver. After receiving the signature and message the receiver checks the signature using the signer identity and PKG public key. If it is, he returns "Accept" otherwise he returns "Reject".

This paper proposes two id-based digital signature schemes; the first one is a id-based digital signature that satisfies unforgeability and verifiability properties, and the second is its extension to an id-based proxy signature scheme in which the original signer delegates his signing rights to a proxy. The receiver verifies the identities of both the original signer and the proxy signer as discussed in details in the rest of paper.

2 Identity Based Signature Scheme Structure

In this section, we describe the generic frame work for an identity based signature scheme. The frame work of an identity based deterministic signature scheme consists of the algorithms described below, namely Setup, Extract, Sign and Verify. An identity based signature scheme is deterministic if the signature on a message by the same user is always the same [13].

Setup: The private key generator (PKG) provides the security parameter k as the input to this algorithm, generates the system parameters params and the master private key msk. PKG publishes params and keeps msk secret.

Extract: The user provides his identity ID to the PKG. The PKG runs this algorithm with identity ID, params and msk as the input and obtains the private key . The private key is sent to user through a secure channel.

Sign: For generating a signature on a message m, the user provides his identity ID, his private key D, params and the message m as input. This algorithm generates a valid signature σ on message m by the user.

Verify: This algorithm on input a signature σ on message m by the user with identity ID, params, checks whether σ is a valid signature on message m by ID. If true it outputs "Valid", else it outputs "Invalid".

3 THE PROPOSED IDENTITY BASED SIGNATURE SCHEME

An id–based signature scheme consists of four phases; Setup, Key Generation, Signature Generation and Signature Verification phases [12].

3.1 Setup

Given security parameter k (usually 160), the PKG chooses q a large prime number with $q > 2^k$, (a, b) is a pair of integers which are smaller than q and satisfy $(4a^3 + 27b^2) \bmod q \neq 0$. E is the selected elliptic curve over the finite field $F_q : y^2 = (x^3 + ax + b) \bmod q$. P is the base point or generator of a group of points on E, denoted as G. Also, O is the point at infinity and n is the order of the point P, with n being a prime number, $n.P = O$ and $n > 2^k$. The PKG selects a cryptographic one way hash function $H : \{0,1\}^* \rightarrow Z_q$. The PKG selects a random number mk_{PKG} as the master key and computes the master public key $P_{pub} = mk_{PKG}.P$. The PKG keeps mk_{PKG} secret and publishes the system parameters $params = \{a, b, k, E, P, P_{pub}, H\}$

3.2 Key Generation

The PKG generates the secret and public key pairs for the signer. It then sends the secret keys through a secure channel and publishes the public key and the identities. The PKG calculates the secret keys of the three communicating parties as follows: $d_a = (H(ID_a).mk_{PKG}) \bmod q$. The PKG calculates the signer public key as follows: $Q_a = d_a.P_{pub}$

3.3 Signature Generation

A signer chooses a random number $w \in [q-1]$ and computes:

- $r = [(w + d_a) \bmod q].P_{pub} = (u, v)$
- $s = (u + H(m).d_a) \bmod q$
- The signer sends $(u, s, H(m))$ to the verifier.

3.4 Signature Verification

The receiver computes:

- $v_1 = H(m).Q_a$
- $v_2 = [(s - u) \bmod q)].P_{pub}$
- If $v_1 = v_2$ accept the signature

4 SECURITY ANALYSIS OF THE PROPOSED SIGNATURE

4.1 Correctness

The correctness of the verification equation as follow:

$v_2 = [(s - u) \bmod q)].P_{pub} = (u + H(m).d_a - u).P_{pub}$

$= H(m).Q_a = v_1$

4.2 Security Properties

The security of the proposed two schemes is based on the elliptic curve discrete logarithm problem (ECDLP) [15]. Up till now, the ECDLP is considered to be hard under the following definition.

Definition 1: The Elliptic Curve Discrete Logarithm Problem (ECDLP) is defined as follows. Let G and Q be two points on an elliptic curve and G is of order n and n is a prime. The point $Q = k.G$, where $k < n$. Given these two points G and Q, find the discrete logarithm of Q to the base G; that is, k.

4.2.1 Unforgeability

Only the original signer with his/her secrete key d_a can produce both (r, s) because of the way they are computed: $r = [(w + d_a) \bmod q].P_{pub} = (u, v)$, $s = (u + H(m).d_a) \bmod q$. They depend on the sender secret key d_a. Therefore, only the original signer can generate a valid signature.

4.2.2 Verifiability

A verifier can be convinced of the agreement of the signer to the message contents by computing $v_1 = H(m).Q_a$, $v_2 = [(s - u) \bmod q)].P_{pub}$ Then, testing if $v_1 = v_2$, a verifier then accepts the signature .

4.3 Performance Analysis and Comparative Study

This section discusses the computational cost associated with the proposed ID-based signature scheme. Table 1 shows the symbol definitions that are used in the comparative study. A comparative study of the performance of the proposed scheme and other schemes in literature [17,18] is provided in Table 2. Clearly, the proposed scheme is the most efficient.

TABLE 1
TIME ABBREVIATIONS

Symbol	Operation
$T_{EC\text{-}mult}$	time complexity required for executing multiplication operation on elliptic curve E
$T_{EC\text{-}add}$	time complexity required for executing addition operation on elliptic curve E
T_{mult}	time complexity required for executing modulus multiplication over a finite field
$T_{inverse}$	time complexity required for executing inverse modulus over a finite field
$T_{pairings}$	time of executing a bilinear pairing operation

- *Hassan Elkamchouchi : Elec. Eng. Dept, Fac. of Eng., Alexandria University.E-mail: helkamchouchi@ieee.org*
- *Eman Abou El-kheir: Elec. Eng. Dept, Fac. of Eng., Kafr El-Sheikh University. E-mail: eman.abouelkhair@eng.kfs.edu.eg*
- *Yasmine Abouelseoud: Eng. Math. Dept, Fac. of Eng., Alexandria University. E-mail: yasmine.abouelseoud@gmail.com*

5 IDENTITY BASED PROXY SIGNATURE SCHEME STRUCTURE

An ID-based proxy signature scheme is specified by the following polynomial-time algorithms [16].

Setup: The private key generator (PKG) provides the security parameter k as the input to this algorithm, generates the system parameters params and the master private key msk. PKG publishes params and keeps msk secret.

Extract: The user provides his identity ID to the PKG. The PKG runs this algorithm with identity ID, params and msk as the input and obtains the private key . The private key is sent to user through a secure channel.

Delegate: The proxy-designation algorithm , takes as input the sender secrete key and the warrants mw and outputs the delegation params from the original signer to the proxy.

Delegate Verify: The designation-verification algorithm, takes as input the original signer identity and the delegation params and verifies whether is a valid delegation come from the original signer

Proxy Key Generation: The proxy key generation algorithm, takes as input the delegation params and some other secret information and outputs a signing key for proxy signature.

Proxy Signature: For generating a signature on a message m, the proxy provides his identity ID, his private key , params and the message m as input. This algorithm generates a valid proxy signature σ on message m by the proxy.

Proxy Verify: This algorithm on input a signature σ on message m by the user with identity ID, params, checks whether σ is a valid signature on message m by ID. If true it outputs "Valid", else it outputs "Invalid".

5.1 Security Requirements For Any Identity Based Proxy Signature Scheme

Informally, the basic security properties for proxy signature schemes can be described as follows [14]:

Verifiability

From a proxy signature, a verifier can be convinced of the original signers agreement on the signed message.

Unforgeability

Only the designated proxy signer can generate a valid proxy signature on behalf of the original signer.

Identifiability

Anyone can determine the identity of the corresponding proxy signer from a proxy signature.

Undeniability

The designated proxy signer cannot deny a valid proxy signature generated by him.

Prevention of misuse

A proxy signing key cannot be used for purpose other than generating valid proxy signatures.

6 THE PROPOSED IDENTITY BASED PROXY SIGNATURE SCHEME

An id–based proxy signature scheme consists of six phases; Setup, Key Generation, Proxy Delegation, Proxy Key Generation, Proxy Signature Generation and Proxy Signature Verification phases. The proposed id-based proxy signature scheme is an extension to the proposed id-based signature scheme that discussed in section three.

6.1 Setup

The set up phase is similar as the id-based signature scheme.

6.2 Proxy Delegation

The original signer chooses a random number d and computes

- $T = d.P_{pub} = (\alpha, \beta)$
- $\sigma = (d - d_a.H(\alpha, m_w))mod\, q$
- The original signer sends (α, σ, m_w) to the proxy signer, where mw is a warrant specifying the identities of both the original signer and the proxy signer as well as the signing rights of the proxy agent and possibly a time frame for the validity of the warrant.

TABLE 2
A COMPARATIVE STUDY OF THE PERFORMANCE OF THE PROPOSED ID-BASED SIGNATURE SCHEME WITH THE SCHEMES IN [18, 17]

Phase	Cha-Cheon signature scheme[18]	Hess signature scheme[17]	The proposed scheme
Signature Generation	$1T_{EC\text{-}mult}+ 1T_{mult}+1T_h$	$2T_{EC\text{-}mult}+ 1T_{EC\text{-}add} +1T_{pairing}+1T_h$	$1T_{EC\text{-}mult}+1T_{mult}+1T_h$
Signature Verification	$1T_{EC\text{-}mult}+ 1T_{EC\text{-}add} +2T_{pairing}$	$2T_{pairing}+1T_h$	$2T_{EC\text{-}mult}+1T_h$
Total	$2T_{EC\text{-}mult} +1T_{mult}+1T_h$ $+1T_{EC\text{-}add} +2T_{pairing}$	$2T_{EC\text{-}mult}+ 1T_{EC\text{-}add} +3T_{pairing}+2T_h$	$3T_{EC\text{-}mult}+1T_{mult}+2T_h$

6.3 Delegate Verify and Proxy Key Generation

The proxy checks if $T \stackrel{?}{=} \sigma.P_{pub} + h(\alpha, m_w).Q_a$. If the equation holds, the proxy signer computes the secrete proxy key $skp = (d_p + \sigma) \bmod q$. Then, the proxy signer generates the signature.

6.4 Proxy Signature Generation

The proxy signer chooses a random number $w \in [q-1]$ and computes :

- $r = [(w + skp) \bmod q].P_{pub} = (u, v)$
- $s = (u + H(m).skp) \bmod q$
- signer sends $(\alpha, \sigma, m_w, u, s, H(m))$ to the verifier

6.5 Proxy Signature Verification

The receiver computes :

- $v_1 = H(m).[T - h(\alpha, m_w).Q_a + Q_p]$
- $v_2 = [(s - u) \bmod q)].P_{pub}$

If $v_1 = v_2$, accept the signature .The receiver verifies the identities of both the original signer as well as the proxy signer using the warrant.

6.6 Proxy key generation

The KGC calculates the secret keys for the sender and the proxy respectively as follows: $d_a = (H(ID_a).mk_{PKG}) \bmod q$,and $d_p = (H(ID_p).mk_{PKG}) \bmod q$ .The KGC calculates the public keys as follows; $Q_a = d_a.P_{pub}$; the sender's public key, and $Q_p = d_p.P_{pub}$; the proxy's public key.

7 Security Analysis and Comparative Study

7.1 Correctness

The proxy agent checks the equation : $T = \sigma.P_{pub} + H(\alpha, m_w).Q_a$

$= (d - d_a.h(\alpha, m_w)).P_{pub} + h(\alpha, m_w).Q_a$

$= d.P_{pub} - d_a.H(\alpha, m_w).P_{pub} + H(\alpha, m_w).Q_a = d.P_{pub} = T$

The receiver computes: $v_1 = H(m).[T - H(\alpha, m_w).Q_a + Q_p]$

$v_1 = H(m).[d.P_{pub} - H(\alpha, m_w).d_a.P_{pub} + Q_p]$

$v_1 = H(m).[d.P_{pub} - H(\alpha, m_w).d_a.P_{pub} + Q_p]$

$v_1 = H(m).(\sigma.P_{pub} + Q_p)$

Also, The receiver computes: $v_2 = [(s - u) \bmod q)].P_{pub}$

$v_2 = (u + H(m).skp - u).P_{pub}$

$v_2 = H(m).skp.P_{pub} = H(m).(d_p + \sigma).P_{pub}$

$v_2 = H(m).(d_p.P_{pub} + \sigma.P_{pub}) = H(m).(\sigma.P_{pub} + Q_p) = v_1$,

then the receiver accepts the signature if the equality holds.

7.2 Security properties

7.2.1 Distinguishability

The proposed proxy signature $(\alpha, \sigma, m_w, u, s, H(m))$ contains the warrant m_w while the normal signature does not, so both are different in the form. Also in the verification equation, public keys Q_a and Q_p , also and warrant m_w are used. So anyone can distinguish the proxy signature from a normal signature easly.

7.2.2 Verifiability

The verifier of a proxy signature can check easily that the verification equation $v_1 = H(m).[T - H(\alpha, m_w).Q_a + Q_p] = v_2$,where $v_2 = [(s - u) \bmod q)].P_{pub}$, if $v_1 = v_2$ accept the signature holds. In addition, this equation involves original signer's public key Q_a and warrant m_w , so any one can be convinced of the original signer's agreement on the proxy signer.

7.2.3 Unforgeability

In our scheme only the designated proxy signer can create a valid proxy signature, since the proxy private key $skp = (d_p + \sigma) \bmod q$ includes the private key d_p of the proxy signer and to compute d_p from Q_p is equivalent to solving the ECDLP.

7.2.4 Nonrepudiation

This is because of the presence of the warrant m_w and public keys Q_a and Q_p in the verification equation. Also, the generation of a proxy signature involves both the original and proxy signers' private keys d_a and d_p respectively. It is already proved that neither the original signer nor the proxy signer can sign in place of any other party. So the original signer cannot deny his delegation and the proxy signer cannot deny having signed the message m on behalf of original signer to another party.

7.2.5 Identifiability

In the proposed scheme, it can be checked who is original signer and who is proxy signer from the warrant m_w. Also, it clear from the verification equation $v_1 = H(m).[T - h(\alpha, m_w).Q_a + Q_p] = v_2$ where $v_2 = [(s - u) \bmod q)].P_{pub}$ that the public keys Q_a and Q_p are asymmetrical in position. So anyone can distin-

guish the identity of the proxy signer from the proxy signature.

7.2.6 Prevention of Misuse

The original signer generates the delegation (α, σ, m_w) where $T = d.G = (\alpha, \beta)$ and $\sigma = (d - d_a.h(\alpha, m_w)) \bmod q$ using its private key and sends it to the proxy. So the delegation cannot be modified or forged. Also the warrant m_w contains the limit of delegated signing capability. So it is not possible to sign the messages that have not been authorized by original signer

TABLE 3
THE PROPOSED PROXY SIGNATURE SCHEME COMPARED WITH THE SCHEMES IN[14]

7.3 Comparative study

The proposed proxy signature scheme is compared with the schemes in [14]. Table 3 shows the comparison in details.

From the comparison, it can be seen that the proposed proxy signature scheme requires less computational effort than the scheme with pairings [14] .

Phase	Bin Wang scheme from pairings [14]	The proposed scheme without pairings
Proxy delegation	$2T_{EC\text{-}mult} + 1T_{EC\text{-}add} + 1T_h$	$1T_{EC\text{-}mult} + 1T_h + 1T_{mul}$
Proxy key generation	$3T_{pairings} + 1T_h$	$2T_{EC\text{-}mult} + 1T_{EC\text{-}add} + 1T_h$
Proxy Signature generation	$2T_{EC\text{-}mult} + 3T_{EC\text{-}add} + 1T_h$	$1T_{EC\text{-}mult} + 1T_h + 1T_{mult}$
Proxy Signature verification	$4T_{pairings} + 2T_{EC\text{-}add} + 2T_h$	$3T_{EC\text{-}mult} + 2T_{EC\text{-}add} + 2T_h$
Total	$7T_{pairings} + 4T_{EC\text{-}mult} + 6T_{EC\text{-}add} + 5T_h$	$7T_{EC\text{-}mult} + 3T_{EC\text{-}add} + 4T_h + 2T_{mult}$

8 CONCLUSION

This paper proposes two schemes; the first is a digital signature with its security analysis discussion, and the second is a proxy signature with its security analysis discussion. Both schemes are more efficient than other schemes when compared with them. Both schemes are without bilinear pairing.

REFERENCES

[1] Z. Cheng, "Simple Tutorial on Elliptic Curve Cryptography", Chapter 2. ECC In Practice , December 1, 2004

[2] M. Mambo, K. Usuda, and E. Okamoto, "Proxy signatures: delegation of the power to sign messages," Transactions on Fundamentals of Electronic Communications and Computer Science, vol. E79-A, pp. 1338-1354, 1996.

[3] S. Kim, S. Park, and D. Won, "Proxy signatures," Proceedings of international conference on information and communications security (ICICS)'97, LNCS 1334, pp. 223-232, Springer-Verlag, 1997.

[4] B. Lee, H. Kim, and K. Kim, "Strong proxy signature and its applications," SCIS2001, vol. 2, no. 2, pp. 603-608, 2001.

[5] J. Lee, J. Cheon, and S. Kim, "An analysis of proxy signatures: Is a secure channel necessary," Cryptology-CT-RSA'03, LNCS 2612, pp. 68-79, Springer-Verlag, 2003.

[6] S. F. Tzeng, M. S. Hwang, and C. Y. Yang, "An improvement of nonrepudiable threshold proxy sig- nature scheme with known signers," Computers & Security, vol. 23, pp. 174-178, 2004.

[7] M. Tian and L. Huang,"Breaking A Proxy Signature Scheme From Lattices. International Journal of Network Security, Vol.14, No.6, PP.320-323, Nov. 2012

[8] Y. Kim and J. H. Chang, " Self Proxy Signature Scheme ", IJCSNS International Journal of Computer Science and Network Security, VOL.7 No.2, February 2007

[9] F. Zhang, R. Safavi-Naini and W. Susilo," An Efficient Signature Scheme from Bilinear Pairings and Its Applications", Springer-Verlag ,pp 277-290 PKC 2004, LNCS 2947

[10] S. Padhye, N. Tiwari," Improved Proxy Signature Scheme without Bilinear Pairings", In the Proceeding of 9th International Conference, QShine 2013, Greader Noida, India, January 11-12, 2013.

[11] A. Shamir, "Advance in Cryptology", Proceedings of CRYPTO 84, (1984) August 19-22, California, USA.

[12] A. Kumar and H. Lee, " Performance Comparison of Identity Based Encryption and Identity Based Signature", International Journal of Security and Its Applications , Vol. 6, No. 3, July, 2012

[13] S. Sharmila Deva Selvi, S. Sree Vivek, C. Pandu Rangan "Identity Based Deterministic Signature Scheme Without Forking-Lemma", IWSEC 2011: 79-95

[14] Bin Wang , " A new identity based proxy signature scheme ", IACR Cryptology ePrint Archive (2008)

[15] D. Johnson, A. Menezes, and S. Vanstone, " The elliptic curve digital signature algorithm (ECDSA) ",International Journal of Information Security 1 (1) (2001) 36–63.

[16] C. Gu and Y. Zhu, "An Efficient ID-based Proxy Signature Scheme from Pairings " , Inscrypt 2007: 40-

[17] F.Hess, Efficient Identity –based signature schemes based on pairings , In Selected Areas in Cryptography- SAC 2002 , pp.310-324, K.Nyberg and H.Heys (eds), Springer Verlag, 2003

[18] J.C.Cha and J.H.Cheon, An Identity based signature from Gap Diffie Hellman Groups, In proceeding PKC'03, LNCS,pp 18-30, Springer Verlag, 2003