New Digital Multisignature Scheme with Distinguished Signing Responsibilities

Luu Hong Dung and Nguyen Hieu Minh
Faculty of Information Technology, Le Qui Don Technical University
100 Hoang Quoc Viet, Ha Noi, Viet Nam

Summary
A multisignature scheme is a digital signature scheme that allows multiple signers to generate a single signature in a collaborative and simultaneous manner. In this paper we first review of the digital multisignature schemes using elliptic curvers and elliptic curve version of the multisignature scheme with distinguished signing responsibilities. Then, we propose a new multisignature scheme with distinguished signing responsibilities. In this scheme, each group member has distinguished signing responsibility and partial contents of the message can be verified without revealing the whole message. Our proposed scheme is more efficient than the scheme reviewed and capable of application in practice.

Keywords
*Multisignature scheme, Elliptic curve, Distinguished signing responsibilities***.**

I. INTRODUCTION

Digital signatures can be classified into two main categories: single signature and multiple signature (or multisignature). Single signature refers to the cases where only one party signs a document, while multiple signature refers to the cases where more than one party sign a single document.

The digital signature schemes in use today can be classified according to the hard underlying mathematical problem which provides the basis for their security [1]:

1. Integer Factorization schemes, which base their security on the intractability of the integer factorization problem. Examples of these include the RSA and Rabin signature schemes.
2. Discrete Logarithm schemes, which base their security on the intractability of the (ordinary) discrete logarithm problem in a finite field. Examples of these include the ElGamal, Schnorr, DSA, and Nyberg-Rueppel signature schemes.
3. Elliptic Curve schemes, which base their security on the intractability of the elliptic curve discrete logarithm problem. For example, in American standard ECDSA and Russian standard GOST R 34-2001.

The indicated problems are hard, if the used primes and elliptic curves satisfy special requirements [2, 3].

In 1983, Itakura and Nakamura [4] proposed the first multisignature scheme. It let multiple signers collaboratively sign the same message and the resultant multisignature can be verified by a group of verifiers to check whether it is valid or not. Since then, several multisignature schemes have been proposed [5-7].

The application of digital multisignature can be found in some secret sharing applications. For example, a company's policy may require multiple managers to sign any business contract. Digital multisignature scheme enables this internal policy effectively. Each manager has to use his individual secret key to sign the same document and all individual signatures can be combined into a single multisignature. However, to any external verifier, this multisignature is just a normal signature that can be verified by using the company's public key, which is a product of all public keys of the signers. In the multisignature schemes proposed in [8], all group members hold the same responsibility of signing the document.

In fact, there are some applications that need to use multisignatures with distinguished signing responsibilities. For example, a company releases a document that may involve financial department, engineering department and program office. Each entity is responsible of preparing and signing a particular section of the document. The signing responsibility of engineering department may have no interest to read the content prepared by the financial department. However, the combination of all sections represents the company's document. The company's document should be easily verified by any outsider using company's public key. For the sake of confidentiality, same verifier may be restricted to access and verify only some sections of the document.

In this paper, we first review of the digital multisignature schemes using elliptic curvers [8] and elliptic curve version of the multisignature scheme with distinguished signing responsibilities [9]. Then, we propose a new multisignature scheme with distinguished signing responsibilities.

We will organize this paper as follows: In section II, we will introduce elliptic curve digital schemes. Brief reviews of the digital multisignature schemes using elliptic curvers in [8] will be introduced in Section III. In Section IV, we will describe the elliptic curve version of the multisignature scheme with distinguished signing responsibilities proposed

Manuscript received January 5, 2010
Manuscript revised January 20, 2010

in [9]. In Section V, we will propose a new multisignature scheme with distinguished signing responsibilities. Section VI, we will present example for our scheme. Finally, a conclusion will be given in Section VII.

II. Elliptic Curve Digital Signature Schemes

Many researchers have examined elliptic curve cryptosystems, which were firstly proposed by Miller [10] and Koblitz [11]. The elliptic curve cryptosystems which are based on the elliptic curve logarithm over a finite field have some advantages than other systems: the key size can be much smaller than the other schemes since only exponential-time attacks have been known so far if the curve is carefully chosen [2], and the elliptic curve discrete logarithms might be still intractable even if factoring and the multiplicative group discrete logarithm are broken.

Elliptic curve cryptosystem is widely used in several digital signature schemes, such as threshold signature scheme, proxy signature scheme, blind signature, and so on. For the elliptic curve over finite field see more details in [12].

III. Digital Multisignature Schemes Using Elliptic Curves

In this section we describe the elliptic curve version of the multisignature scheme proposed in [8]. It contains three phases: key generation, multisignature generation, and multisignature verification. We review their scheme briefly as follows:

We assume that there are t signers, $1 \le i \le t$ to sign the same message $m \in \{0, 1\}^*$.

A. Key Generation

Firstly, we choose elliptic curve domain parameters ([2, 12]):

1. Choose p a prime and n an integer. Let $f(x)$ be an irreducible polynomial over GF(p) of degree n, generating finite field GF(p^n) and assume that α is a root of $f(x)$ in GF(p^n).
2. Two field elements a, $b \in$ GF(p^n), which define the equation of the elliptic curve E over GF(p^n) (i.e., $y^2 = x^3 + ax + b$ in the case $p > 3$, where $4a^3 + 27b^2 \neq 0$).
3. Two field elements x_p and y_p in GF(p^n), which define a finite point $P = (x_p, y_p)$ of prime order q in E(GF(p^n)) ($P \neq O$, where O denotes the point at infinity).
4. The converting function $c(x)$: GF(p^n) $\rightarrow Z_{p^n}$ which is given by:

$$c(x) = \sum c_k p^i \in Z_{p^n}, x = \sum_{i=0}^{n-1} c_k \alpha_i \in GF(p^n), 0 \le c_i \le p.$$

The operation of the key generation is as follows:

1. Each signer randomly selects an integer d_i from the interval [1, q - 1] and computes a corresponding public key as the point: $Q_i = d_i P$.
2. Compute the public key Q for all signers, which is equal to the sum of all individual public keys $Q = Q_1 + Q_2 + ... + Q_t = dP = (x_Q, y_Q)$, where $d = d_1 + d_2 + ... + d_t$ (mod q).
3. Let H be a one-way hash function such as SHA-1.

B. Generating the Multisignature

Each signer U_i, $1 \le i \le t$ executes next steps:

1. Randomly selects a number $k_i \in [1, q - 1]$ and computes:
$$R_i = k_i P = (x_{R_i}, y_{R_i}),\ 1 \le i \le t.$$
2. Converting the x-coordinate of point R_i into the integer $r_i = c(x_{R_i})$, where $c(x)$ is the converting function. The values r_i is broadcast to the other signer.
3. Once r_i, $1 \le i \le t$, are available through the broadcast channel, each signer computes the commitment r as
$$r = r_1 + r_2 + ... + r_t \ (\text{mod } q).$$
4. Uses his secret keys, d_i and k_i, to sign the message m. The signer U_i computes
$$s_i = d_i H(m) - k_i r \ (\text{mod } q).$$
5. Transmits the pair (m, s_i) to the clerk.

Once the clerk receives the individual signature (r_i, s_i) from U_i, he needs to verify the validity of this individual signature. The verification procedure is to compute the point

$$(r^{-1} H(m) \text{ mod } q)Q_i - (r^{-1} s_i \text{ mod } q)P = (x_{e_i}, y_{e_i}),\ 1 \le i \le t$$

and check

$$r_i = (x_{e_i}, y_{e_i}) \ (\text{mod } q),\ 1 \le i \le t.$$

Once all individual signatures are received and verified by the clerk, the multisignature of the message m can be generated as (r, s), where $s = s_1 + s_2 + ... + s_t$ (mod q).

C. Verifying the Multisignature

Since individual signatures (r_i, s_i), $1 \le i \le t$, satisfy

$$(r^{-1} H(m) \text{ mod } q)Q_i - (r^{-1} s_i \text{ mod } q)P = (x_{e_i}, y_{e_i}), 1 \le i \le t.$$

Adding the above equations from 1 through t, we obtain

$$(r^{-1} H(m) \text{ mod } q)Q - (r^{-1} s \text{ mod } q)P = (x_e, y_e).$$

where $s = s_1 + s_2 + ... + s_t$ (mod q), $Q = Q_1 + Q_2 + ... + Q_t = dP = (x_Q, y_Q)$ and $r = c(x_e)$(mod q). In other words, the verifier computes the point (x_e, y_e) and check if $r = c(x_e)$(mod q). If this is true, then (r, s) is accepted as the valid multisignature of the message m signed by the users U_i, $1 \le i \le t$.

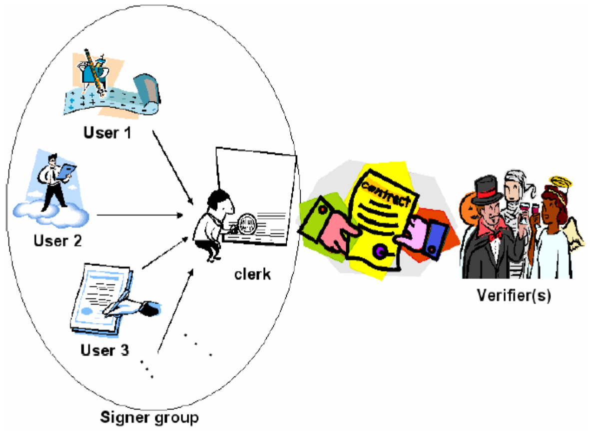

Figure 1. The graph of multisignature scheme.

By Figure 1, we can see that all signers send their partial multisignature to the clerk. The clerk checks all partial multisignature, construct them into a multisignature and sends it to the verifier(s). Of course the clerk is either a participant in signer group or the third trusty authority.

IV. MULTISIGNATURE SCHEME WITH DISTINGUISHED SIGNING RESPONSIBILITIES

In this section we describe the elliptic curve version of the multisignature scheme with distinguished signing responsibilities proposed in [9]. It contains three phases: key generation, multisignature generation, and multisignature verification. We review their scheme briefly as follows:

The elliptic curve domain and the key generation are the same as in Section III.

A. Generating the Multisignature

We assume that there are t signers U_i, $1 \le i \le t$. Instead of signing the same message m directly, each signer should prepare a section of message $m \in \{0, 1\}^*$ that he is responsible of and broadcast $H(m_i)$ to all other signers, where H is the one way hash function.

The operation of generating the multisignature with distinguished signing responsibilities is as follow:

1. The signer U_i, $1 \le i \le t$, randomly selects a number $k_i \in [1, q-1]$ and computes $R_i = k_iP = (x_{R_i}, y_{R_i})$, $1 \le i \le t$.
2. Converting the x-coordinate of point R_i into the integer $r_i = c(x_{R_i})$, where $c(x)$ is the converting function.
   The values r_i is broadcast to the other signer.
3. Once r_i, $1 \le i \le t$, are available through the broadcast channel, each signer computes the commitment r as
   $$r = r_1 + r_2 + \ldots + r_t \pmod q.$$
4. Each signer U_i, uses his secret keys, d_i and k_i, to sign the message $M = H(H(m_1), H(m_2), \ldots, H(m_t))$, where $H(H(m_1), H(m_2), \ldots, H(m_t))$ means the hash value of the concatenation of $H(H(m_1), H(m_2), \ldots, H(m_t))$. The signer U_i computes
   $$s_i = d_iM - k_ir \pmod q.$$

and transmits the pair (M, s_i) to the clerk.

Once the clerk receives the individual signature (r_i, s_i) from U_i, he needs to verify the validity of this individual signature. The verification procedure is to compute the point

$$(r^{-1} M \bmod q)Q_i - (r^{-1} s_i \bmod q)P = (x_{e_i}, y_{e_i}),\ 1 \le i \le t$$

and check

$$r_i = (x_{e_i}, y_{e_i}) \pmod q,\ 1 \le i \le t.$$

Once all individual signatures are received and verified by the clerk, the multisignature of the message $m = (m_1, m_2, \ldots, m_t)$ can be generated as (r, s), where $s = s_1 + s_2 + \ldots + s_t \pmod q$. Since each signer is responsible of preparing a section of message m, the pair (r, s) is a digital multisignature with distinguished signing responsibilities.

B. Verifying the Multisignature

The verifier computes the point

$$(r^{-1} M \bmod q)Q - (r^{-1} s \bmod q)P = (x_e, y_e).$$

where $s = s_1 + s_2 + \ldots + s_t \pmod q$, $Q = Q_1 + Q_2 + \ldots + Q_t = dP = (x_Q, y_Q)$ and $r = c(x_e) \pmod q$. In other words, the verifier computes the point (x_e, y_e) and check if $r = c(x_e) \pmod q$. If this equality holds, the pair (r, s) is a digital multisignature with distinguished signing responsibilities of the message m.

Instead of signing the message $H(m_1, m_2, \ldots, m_t)$, each signer needs to sign the message $M = H(H(m_1), H(m_2), \ldots, H(m_t))$. The computation of $H(H(m_1), H(m_2), \ldots, H(m_t))$ is faster than that of $H(m_1, m_2, \ldots, m_t)$ because each signer needs only to compute his own $H(m_i)$ and the other $H(m_j)$, $j \ne i$, $1 \le i, j \le t$, has been computed by the other signer.

In the case some verifies only allowed to access partial contents of the message, the partial contents can still be verified using the group public key without revealing whole message. This feature can be achieved by just providing the one way hash values of the inaccessible contents to the verifier. But in fact this is very difficult to implement because of the complexity of verifying procedures at each verifier. So this scheme is not high realistic.

In the next section, we will propose a new digital multisignature scheme allows overcoming the drawbacks pointed out by this scheme.

V. OUR PROPOSED SCHEME

In this section we describe the proposed multisignature scheme with distinguished signing responsibilities.

The elliptic curve domain is the same as in Section III.

In this scheme, instead of signing the message $M = H(H(m_1), H(m_2), ..., H(m_t))$, each signer just needs to sign the message $H(m_i)$ their respective.

A. Key Generation

1. Each signer randomly selects an integer d_i from the interval $[1, q - 1]$ and computes a corresponding public key as the point: $Q_i = d_iP$.
2. Compute the public key Q for all signers, which is equal to the sum of all individual public keys $Q = \sum_{i=1}^{t} H(m_i)Q_i = (x_Q, y_Q)$, where $H(m_i)$ mean the hash value of ith signer.
3. Let H be a one-way hash function such as SHA-1.

B. Generating the Multisignature

1. The signer U_i, $1 \le i \le t$, randomly selects a number $k_i \in [1, q - 1]$ and computes $R_i = k_iP = (x_{R_i}, y_{R_i})$, $1 \le i \le t$.
2. Once R_i, $1 \le i \le t$, are available through the broadcast channel, each signer computes the commitment R as
$$R = R_1 + R_2 + ... + R_t \pmod q).$$
3. The first part e of the signature (e, s) is computed using formula:
$$e = (x_R) \bmod \delta ,$$
where choose δ is a prime greater than or equal to 160 bits [13].
4. Each signer U_i, uses his secret keys, d_i and k_i, to sign the message $H(m_i)$ their respective. The signer U_i computes
$$s_i = \left(k_i - eH(m_i)d_i\right) \bmod q$$
and transmits s_i to the clerk.

Once the clerk receives the individual signature (r_i, s_i) from U_i, he needs to verify the validity of this individual signature. The verification procedure is to compute the point

$$((x_{R_i}) \bmod \delta) Q_i + s_iP = (x_{e_i}, y_{e_i}), \ 1 \le i \le t$$

and check

$$R_i = (x_{e_i}, y_{e_i}) \pmod q, \ 1 \le i \le t.$$

5. Compute the second part s of the signature:
$$s = \sum_{i=1}^{t} s_i \bmod q$$

The multisignature of the message $m = (m_1, m_2, \ldots, m_t)$ can be generated as (e, s). Since each signer is responsible of preparing a section of message m, the pair (e, s) is a digital multisignature with distinguished signing responsibilities.

C. Verifying the Multisignature

1. Using the pair (e, s) compute value R':
$$R' = eQ + sP$$
2. Compute $e' = x_{R'} \bmod \delta$.
3. Compare values e' and e.

If this equality holds, the pair (e, s) is a digital multisignature with distinguished signing responsibilities of the message m.

Proof formula in the process of verifying the multisignature:

The public key Q for all signers, which is equal to the sum of all individual public keys

$$Q = \sum_{i=1}^{t} H(m_i)Q_i = \sum_{i=1}^{t} H(m_i)d_iP .$$

Value s_i calculated by the formula:

$$s_i = \left(k_i - eH(m_i)d_i\right) \bmod q$$

Thus

$$\sum_{i=1}^{t} s_i \equiv \sum_{i=1}^{t} k_i - e\sum_{i=1}^{t} H(m_i)Q_i \bmod q$$

Value R' used to calculate the first part of the verify equation, calculated by the following formula:

$$R' = eQ + sP = e\sum_{i=1}^{t} H(m_i)Q_i + (\sum_{i=1}^{t} k_i - e\sum_{i=1}^{t} H(m_i)d_i)P$$

$$= e(\sum_{i=1}^{t} H(m_i)d_i)P + (\sum_{i=1}^{t} k_i - e\sum_{i=1}^{t} H(m_i)d_i)P$$

$$= \sum_{i=1}^{t} k_iP = R .$$

Next, $e' = x_{R'} \bmod \delta = x_R \bmod \delta = e$, i. e, the correctness of the procedures for generating and verifying digital signature is proved.

In this scheme, instead of signing the message $M = H(H(m_1), H(m_2), ..., H(m_t))$ of each signer, each signer just needs to sign the message $H(m_i)$ their respective. The computation of $H(m_i)$ is faster than that of $H(H(m_1), H(m_2), ..., H(m_t))$ because each signer needs only to compute his own $H(m_i)$.

Since not calculated inverse element in process of veriying as well as the operation of the scheme done faster.

However, in the case some verifies only allowed to access partial contents of the message, this scheme also has disadvantages such scheme in Section IV (must provide the one way hash values of the inaccessible contents to the verifier).

The proposed scheme possess the following advantages:

1. the digital signature length is sufficiently small and does not depend on number of signers (the multisignature length is equal to the length of individual signature provided by the underlying signature algorithm);
2. the standard public key infrastructure (PKI) is used;
3. the scheme can be efficiently used in practice for simultaneous signing a contract with distinguished signing responsibilities;
4. the secure is as secure as elliptic curve schemes is secure.

The last fact can be proved using the technique applied in [14] to prove security of the collective DS regarding to the following two types of general attacks.

The attack of the first type corresponds to forgery of the multisignatue.

The second type attack corresponds to scenario of the calculating the secret key of one of the signers, which shares a multisignature.

In the first attack it is assumed that $t-1$ legitimate signers attempt to create a multisignatue corresponding to t signers.

In the second attack it is assumed that $t-1$signers that shares some multisignatue (e, s) with the tth signer are trying to compute the private key of the tth signer.

It has been proved [14] that any successful method to perform any of the attacks allows breaking the underlying DS algorithm.

A modification of this scheme allows integrity checking more efficient and capable of application in practice is proposed as follows:

All steps are implemented remain, except the following changes: With t signers U_i, $1 \le i \le t$, instead of signing the message $H(m_i)$ their respective, $(t-1)$ signer needs to sign the message $H(m_i)$ their respective, $1 \le i \le t-1$, where message $m = (m_1, m_2, \ldots, m_{t-1})$. The last signer (who authorized the highest) needs to sign the message $H(m)$.

On the receipt, the verifier entitled to receive the full message will check the signature on behalf of the whole group.

VI. EXAMPLE

This example illustrates the signature generation and verification procedures in the digital multisignature scheme proposed.

The parameters used in the example to ensure sufficient magnitude to be applied in practice.

In the example shown simultaneously with the three signer signed the three documents in a contracts ($t = 3$, $m = (m_1, m_2, m_3)$).

A. Key Generation

Firstly, we choose elliptic curve domain parameters as follows:

a = 55217678657376345553904163005997766223473333 59784, b = 9717196, p = 55217678657376345553904163005997766223473333 59787.

This elliptic curve contains the number of points equal to the prime number V = 55217678657376345553904162287838869133398238 41723, i. e any of its point of order q, equal to the value V, i. e $q = V$.

B. Generating the Multisignature

1. Generate elliptic curve with the parameters listed above.
2. Each signer randomly selects an integer d_i from the interval $[1, q - 1]$:

d_1 = 8182108890892890101467333434019;

d_2 = 3952504539403758278808581024791;

d_3 = 9763160941600092631935520658071.

3. Computes a corresponding public key as the point: $Q_i = d_i P$

a. Generate a point P of order q:

P = (40581389988176995699766783582333359584950379 69465, 76856892633603682571849521891630868249411614416 0);

b. Generate points Q_1, Q_2, Q_3 according to the formula $Q_i = d_i P$, where $i = 1, 2, 3$:

Q_1 = (24067676659281588994469061658217472188835746 02371,56237764852169229068903150720500806020534563 6991);

Q_2 = (34870810837802708535738941404482523792268351 0732,14020261919960801963994827704684725980760525 99809)

Q_3 = (430716607783351930106332253302416200509102 5020313,528029631254915602814890591421557065 5514986217509).

4. Compute the public key Q for all signers, which is equal to the sum of all individual public keys $Q = \sum_{i=1}^{t} H(m_i)Q_i$

a. Accepts three messages, submitted H_1, H_2 and H_3:

H_1= 135708092215597910168314154751917220633712178686;

H_2= 381249899002881915531657135063437665281433177 0527;

H_3= 892599902687114513152033761211777868065919257 6033.

b. Forms the collective public key in the form of point Q by the formula $Q = \sum_{i=1}^{t} H(m_i)Q_i$

H_1Q_1 = (138608434900254542451166892694557506683840 7867380,454363373195884010112484581877184100437 4561021017)

H_2Q_2 = (299211431532419026428141393395396288778694 972469,530032709442115487602206494687620629685 3312043729)

H_3Q_3 = (452352848795452290087869487789555696379634 5154180,410082610348097279898099690919652619932 7733122181)

P=(228426539485900338090938878090464611548638254 406,120227817455309523113538906020964990253572 7110543).

C. Generating the Multisignature

1. The first, second and third signer generates random k_1, k_2 and k_3, respectively:

k_1 = 20908809226259826835844601678623823 79;

k_2 = 536038352685666370058389620526641834 1;

k_3 = 767711881072314235201231745340088744 9.

2. Then the first, second and third signer generates points R_1, R_2 and R_3, respectively, according to the formula $R_i = k_iP = (x_{R_i}, y_{R_i})$:

R_1 = (453336007529244660885066440036471159220513 6618460,117506133706223217958434868647732476210 1164050095);

R_2 = (195827922382790204737933646528589543533014 0185477,883650890825623295514423424297049431856 4852573);

3. R_3 = (503861602885295987750955408178966743685 3794753557,209613157933044677924551688484534713 038841468913).

4. Generate the R-point formula $R = R_1 + R_2 + R_3 \pmod q$):

R = (259709797026361086354606943683399458000210 5418569,330491504010440081380237428247398555001 5521973383).

5. The first part e of the signature (e, s) is computed using formula: $e = (x_R) \bmod \delta$,

δ= 711819821865932102898901 1;

e = 507900823307693208747378 9.

6. Each signer U_i, uses his secret keys, d_i and k_i, to sign the message $H(m_i)$ their respective. The signer U_i computes $s_i = (k_i - eH(m_i)d_i) \bmod q$

s_1 = 133444963875333388923743187271473122915443205289;

s_2 = 188766165320384794471028245083561255162008142 7016;

s_3 = 485069616195599112555931808455502133558030218 9827.

7. Compute the second part s of the signature: $s = \sum_{i=1}^{t} s_i \bmod q$

s = 135003491329753790380292749387822009677600298 0409.

8. The pair (e, s) is a digital multisignature with distinguished signing responsibilities (5079008233076932087473789,13500349132975379038029 27493878220096776002980409).

D. Verifying the Multisignature

1. Using the pair (e, s) compute value R': $R' = eQ + sP$

eQ = (455684817959588714140072672389132143860230 7875189,288377928957475698317738795561807371973 1329543379);

sP = (1360352815531577166684912233134001496389816
081366,3543269787247235781900897404104279341644752005600);

R'= (2597097970263610863546069436833994580002105
418569,3304915040104400813802374282473985550015521973383);

2. Compute $e' = x_{R'} \bmod \delta$
δ= 711819821865932102898901 1:

e' = 507900823307693208747378 9.

3. Compare values e' and e.
e' = 5079008233076932087473789 = e.

The comparison shows that the parameters e' and e coincide. The coincidence of the values of e' and e means that the digital multisignature is authentic.

VII. Conclusion

A new multisignature scheme with distinguished signing responsibilities have been proposed.

In this scheme, each group member has distinguished signing responsibility and partial contents of the message can be verified without revealing the whole message. Thus the proposed scheme is efficient as solutions of the problems of simultaneous signing a contract and package of contract, which suites well for practical application..

References

[1] Don Johnson, Alfred Menezes and Scott Vanstone, "The Elliptic Curve Digital Signature Algorithm (ECDSA)," Certicom, 2001.

[2] N. Koblitz, *A Course in Number Theory and Cryptography*, Springer-Verlag. Berlin, 2003. - 236 p.

[3] Pieprzyk J., Hardjono Th., and Seberry J., *Fundamentals of Computer Security*, Springer-Verlag. Berlin, 2003. - 677 p.

[4] K. Itakara and K. Nakamura, "A Public Key Cryptosystem Suitable for Digital Multisignatures," NEC Res. Dev. 71 1983, pp. 1-8.

[5] L. Harn and T. Kiesler, "New Scheme for Digital Multisignature," IEEE Electron. Lett. Vol. 25 (15), 1989, pp. 1002-1003.

[6] K. Ohta and T. Okamoto, "A Digital Multisignature Scheme Based on The Fiat-Shamir Scheme," ASIACRYPT'91, 1991, pp. 139-148.

[7] T. Wu and S. Chou, "Two ID-based Multisignature Protocols for Sequential and Broadcasting Architecture," Comput. Commun. Vol. 19 (10), 1996, pp. 851-856.

[8] L. Harn, "Group-oriented (t,n) Threshold Signature and Multisignature," IEEE Proceedings Computer and Digital Techniques, Vol. 141, No. 5, pp. 307-313, 1994.

[9] C. Popescu, "A Digital Multisgnature Scheme with Distinguished Signing Responsibilities," Studies in Informatics and Control, Vol .12, No. 3, Sep 2003.

[10] V. Miller, "Uses of Elliptic Curves in Cryptography," In Advances in Cryptology, Proceedings of Crypto'85, Lecture Notes in Computer Sciences, Springer-Verlag, pp. 417-426, 1986.

[11] N. Koblitz, "Elliptic Curve Cryptosystems," Mathematics of Computation, Vol. 48, pp. 203-209, 1987.

[12] A. Menezes, *Elliptic Curve Public Key Cryptosystems*, Kluwer Academic Publishers, 1993.

[13] Minh N. H. and Moldovyan N. A., "Protocols for Simultaneous Signing Contracts," 2009 IEEE International Conference on Advanced Technologies for Communications, October 12-14, 2009, Hai Phong, Vietnam.

[14] Minh N. H., Moldovyan N. A., Minh N. L, "New Multisignature Protocols Based on Randomized Signature Algorithms," 2008 IEEE International Conference on Research, Innovation and Vision for the Future in computing & Communication Technologies, University of Science - Vietnam National University, Ho Chi Minh City, July 13-17, 2008. Proc. PP. 23.pdf, 2008.

Luu Hong Dung is a Lecturer with the Le Qui Don Technical University (Ha Noi, Viet Nam). His research interests include cryptography, communication and network security. He has authored or co-authored more than 15 scientific articles, books chapters, reports and patents, in the areas of his research. He received his Master from the Le Qui Don Technical University (2001).

Nguyen Hieu Minh is a Lecturer with the Le Qui Don Technical University (Ha Noi, Viet Nam). His research interests include cryptography, communication and network security. He has authored or co-authored more than 30 scientific articles, books chapters, reports and patents, in the areas of his research. He received his Ph.D. from the Saint Petersburg Electrical Engineering University (2006).