New Digital Multisignature Scheme with Distinguished Signing Responsibilities

Luu Hong Dung and Nguyen Hieu Minh
Faculty of Information Technology, Le Quy 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 curves 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.

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 curves [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 curves 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
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 over 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 a 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, $1 \leq i \leq t$ 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 $\text{GF}(p)$ of degree $n$, generating finite field $\text{GF}(p^n)$ and assume that $a$ is a root of $f(x)$ in $\text{GF}(p^n)$.
2. Two field elements $a, b \in \text{GF}(p^n)$, which define the equation of the elliptic curve $E$ over $\text{GF}(p^n)$ (i.e., $y^2 = x^3 + ax + b$ in the case $p > 3$, where $4a^3 + 27b^2 \neq 0$).
3. Two field elements $x_1$ and $y_1$ in $\text{GF}(p^n)$, which define a finite point $P = (x_1, y_1)$ of prime order $q$ in $E(\text{GF}(p^n))$ ($P \neq O$, where $O$ denotes the point at infinity).
4. The converting function $c(x)$: $\text{GF}(p^n) \rightarrow Z_p$ which is given by:

$$c(x) = \sum_{i=0}^{n-1} c_i p^i \in Z_p, x = \sum_{i=0}^{n-1} c_i a_i \in \text{GF}(p^n), 0 \leq c_i \leq p.$$ 

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 $Q = Q_1 + Q_2 + \ldots + Q_t = dP = (x_{dP}, y_{dP})$, where $d = d_1 + d_2 + \ldots + d_t (\text{mod } q)$.
3. Let $H$ be a one-way hash function such as SHA-1.

B. Generating the Multisignature

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

1. Randomly selects a number $k_i \in [1, q - 1]$ and computes:

$$R_i = k_i P = (x_{R_i}, y_{R_i}), 1 \leq i \leq t.$$ 

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 $r_i$, $1 \leq i \leq t$, are available through the broadcast channel, each signer computes the commitment $r$ as:

$$r = r_1 + r_2 + \ldots + r_t (\text{mod } q).$$ 

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:

$$(r_i H(m) \text{mod } q) Q_i - (r_i s_i \text{mod } q) P = (x_{s_i}, y_{s_i}), 1 \leq i \leq t$$

and check

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

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

C. Verifying the Multisignature

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

$$(r_i H(m) \text{mod } q) Q_i - (r_i s_i \text{mod } q) P = (x_{s_i}, y_{s_i}), 1 \leq i \leq t.$$ 

Adding the above equations from 1 through $t$, we obtain

$$(r H(m) \text{mod } q) Q - (r s \text{mod } q) P = (x_r, y_r).$$ 

where $s = s_1 + s_2 + \ldots + s_t (\text{mod } q)$, $Q = Q_1 + Q_2 + \ldots + Q_t = dP = (x_{dP}, y_{dP})$ and $r = c(x_r) (\text{mod } q)$. In other words, the verifier computes the point $(x_r, y_r)$ and check if $r = c(x_r) (\text{mod } q)$. If this is true, then $(r, s)$ is accepted as the
valid multisignature of the message $m$ signed by the users $U_i$, $1 \leq i \leq t$.

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 truster 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 $U_i$, $1 \leq i \leq t$. 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 $H(m)$ 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 $U_i$, $1 \leq i \leq t$, randomly selects a number $k_i \in [1, q - 1]$ and computes $R_i = k_i P = (x_{R_i}, y_{R_i})$, $1 \leq i \leq t$.
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.
3. The values $r_i$ is broadcast to the other signer.
4. Each signer $U_i$ uses his secret keys, $d_i$ and $k_i$, to sign the message $M = H(H(m_1), H(m_2), ..., H(m_t))$, where $H(H(m_1), H(m_2), ..., H(m_t))$ means the hash value of the concatenation of $H(H(m_1), H(m_2), ..., H(m_t))$. The signer $U_i$ computes
   $$s_i = d_i M - k_i r_i (\mod q),$$
   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
   $$(r_i M \mod q)Q_i - (r_i s_i \mod q)P = (x_{e_i}, y_{e_i}), 1 \leq i \leq t \mod q$$
and check
   $$(r_i M \mod q)Q_i - (r_i s_i \mod q)P = (x_{e_i}, y_{e_i}), 1 \leq i \leq t.$$

Once all individual signatures are received and verified by the clerk, the multisignature of the message $m = (m_1, m_2, ..., m_t)$ can be generated as $(r, s)$, where $s = s_1 + s_2 + ... + s_t \mod q$. 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
   $$(r M \mod q)Q = (x, y),$$
where $s = s_1 + s_2 + ... + s_t \mod q$, $Q = Q_1 + Q_2 + ... + Q_t = dP = (x_e, y_e)$ and $r = c(x)(\mod q)$. In other words, the verifier computes the point $(x_e, y_e)$ and check if $r = c(x)(\mod q)$. 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 $H(m_1), H(m_2), ..., H(m_t)$, each signer needs to sign the message $M = H(H(m_1), H(m_2), ..., H(m_t))$. The computation of $H(H(m_1), H(m_2), ..., H(m_t))$ is faster than that of $H(m_1), H(m_2), ..., H(m_t)$ because each signer needs only to compute his own $H(m)$ and the other $H(m), j \neq i, 1 \leq i, j \leq t$, 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 \( M = H(H(m_1), H(m_2), ..., H(m_l)) \), 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: \( Q_i = d_i P \).
2. Compute the public key \( Q \) for all signers, which is equal to the sum of all individual public keys
   \[
   Q = \sum_{i=1}^{t} H(m_i) Q_i = (x_Q, y_Q), \text{ where } H(m_i) \text{ mean the hash value of } \text{ith signer.}
   \]
3. Let \( H \) be a one-way hash function such as SHA-1.

B. Generating the Multisignature

1. The signer \( U_i, 1 \leq i \leq t \), randomly selects a number \( k_i \in [1, q - 1] \) and computes \( R_i = k_i P = (x_{R_i}, y_{R_i}) \), \( 1 \leq i \leq t \).
2. Once \( R_i, 1 \leq i \leq t \), are available through the broadcast channel, each signer computes the commitment \( R \) as
   \[
   R = R_1 + R_2 + ... + R_t (\mod q).
   \]
3. The first part \( e \) of the signature \((e, s)\) is computed using formula:
   \[
   e = (x_R) \mod \delta,
   \]
   where choose \( \delta \) 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
   \[
   s_i = (k_i - eH(m_i)d_i) \mod q
   \]
   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
\[
( (x_R) \mod \delta ) Q_i + s_i P = (x_{R_i}, y_{R_i}), 1 \leq i \leq t
\]
and check
\[
R_i = (x_{R_i}, y_{R_i}) (\mod q), 1 \leq i \leq t.
\]
5. Compute the second part \( s \) of the signature:
   \[
s = \sum_{i=1}^{t} s_i \mod q
   \]

The multisignature of the message \( m = (m_1, m_2, ..., m_l) \) 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 \( e' = x_R \mod \delta \)
   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
\[
Q = \sum_{i=1}^{t} H(m_i) Q_i = \sum_{i=1}^{t} H(m_i) d_i P.
\]

Value \( s_i \) calculated by the formula:
\[
s_i = (k_i - eH(m_i)d_i) \mod q
\]
Thus
\[
\sum_{i=1}^{t} s_i = \sum_{i=1}^{t} k_i - e\sum_{i=1}^{t} H(m_i)d_i \mod q
\]

Value \( R' \) used to calculate the first part of the verify equation, calculated by the following formula:
\[
R' = eQ + sP = e\sum_{i=1}^{t} H(m_i) Q_i + \sum_{i=1}^{t} k_i - e\sum_{i=1}^{t} H(m_i)d_i P
\]
\[
= e\sum_{i=1}^{t} H(m_i)d_i P + \sum_{i=1}^{t} k_i - e\sum_{i=1}^{t} H(m_i)d_i P
\]
\[
= \sum_{i=1}^{t} k_i P = R.
\]

Next, \( e' = x_R \mod \delta \). Since each signer needs only to compute his own \( H(m_i) \).

In this scheme, instead of signing the message \( M = H(H(m_1), H(m_2), ..., H(m_l)) \) 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 \( H(H(m_1), H(m_2), ..., H(m_l)) \) because each signer needs only to compute his own \( H(m_i) \).
Since not calculated inverse element in process of verifying 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 multisignature.

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 multisignature corresponding to \( t \) signers.

In the second attack it is assumed that \( t - 1 \) signers that shares some multisignature \((e, s)\) with the \( t \)th signer are trying to compute the private key of the \( t \)th 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 \( U_t, 1 \leq i \leq t \), instead of signing the message \( H(m) \) their respective, \((t - 1)\) signer needs to sign the message \( H(m) \) their respective, \( 1 \leq i \leq t - 1 \), 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:

\[
\begin{align*}
a & = 55217678657376345553904163005997766622347333359784, \\
b & = 9717196, \\
p & = 55217678657376345553904163005997766622347333359787. 
\end{align*}
\]

This elliptic curve contains the number of points equal to the prime number \( V = 5521767865737634555390416228783886913339823841723 \), 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 \([1, q - 1] \):

\[
\begin{align*}
d_1 & = 8182108890892890101467333434019; \\
d_2 & = 3952504539403758278805851024791; \\
d_3 & = 9763160941600092631935520658071. 
\end{align*}
\]
3. Computes a corresponding public key as the point: \( Q_i = d_iP \)

a. Generate a point \( P \) of order \( q \):

\[
P = (405813899981766956976678358233335958495037969465, \\
76856892633603682571849521891630868249411641160) ;
\]

b. Generate points \( Q_1, Q_2, Q_3 \) according to the formula \( Q_i = d_iP \), where \( i = 1, 2, 3 \):

\[
\begin{align*}
Q_1 & = (24067676659281588994469061582174721883574602371,562377648521692290689031507020500800) \\
Q_2 & = (348708108378027085357389414044825237922683510732,1402026191996080196399482770468472598076052599809) ;
\end{align*}
\]
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 $H_3$:

\[
H_1 = 13570809221559791016831415475191722063371217866;
\]

\[
H_2 = 38124989900288191553165713506343766528143317705277;
\]

\[
H_3 = 8925999026871154131520337612117778680659192576033.
\]

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

\[
H_1Q_1 = (1386084349002545424511668926945575066838407867380, 4543633731958840101124845851877718410043745661021017);
\]

\[
H_2Q_2 = (299211431532419026428141393395396288778694972469, 53003270944211548760220649646876206269853312043729);
\]

\[
H_3Q_3 = (45235284879545229008786948778795556963796345154180, 4100826103480972798980996909165621993277331221811);
\]

\[
P = (22842653948590033809083988090464611548638254406, 1202278174553095231113538906020649902535727110543).
\]

C. Generating the Multisignature

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

\[
k_1 = 209088092262598268358460167862382379;
\]

\[
k_2 = 53603835268566637005838962025266418341;
\]

\[
k_3 = 7677118810723142352012317453400887449;
\]

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

\[
R_1 = (45333600752924466608850664400364711592205136618460, 117506133706223217958434868477324762101164050095);
\]

\[
R_2 = (1958279223872902047379336465285895435330140185477, 8836508908256232955144234242970494318564852573);
\]

\[
R_3 = (5038616028852959877509554081789667436853794753557, 20961315793304467724551688484534713038841468913);
\]

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

\[
R = (2597097970263610863546069436833994580002105418569, 33049150401044008138023742847398955500155219733833);
\]

5. The first part $e$ of the signature $(e, s)$ is computed using formula: $e = (x_R) \mod \delta$, $\delta = 718198218659321028998911$;

\[
e = 507900823307693208747389.
\]

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 $s_i = (k_i - eH(m_i)d_i) \mod q$:

\[
s_1 = 13344496387533338823743871727473122915443205289;
\]

\[
s_2 = 1887661653203847944710282450835612551620081427016;
\]

\[
s_3 = 485066916195599112555593180445550213335580302189827.
\]

7. Compute the second part $s$ of the signature:

\[
s = \sum_{i=1}^{t} s_i \mod q = 135003491329753790380292749387822009677600298049.
\]

8. The pair $(e, s)$ is a digital multisignature with distinguished signing responsibilities $(507900823307693208747389, 135003491329753790380292749387822009677600298049)$.

D. Verifying the Multisignature

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

\[
eQ = (455684817595887141400726723893121438602307875189, 288337729857475698317738795561807319731329543379);
\]
sP = (1360352815531577166684912233134001496389816081366,3543269787247235781900897404104279341644752005600);

R' = (259709797026361086354606943683399458002105418569,3304915040104400813802374282473985550015521973383);

2. Compute $e' = xR'$ mod $\delta$

$\delta = 7118198218659321028989011$:

$e' = 5079008233076932087473789$.

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 suits well for practical application.

REFERENCES


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).