The New Collective Signature Schemes Based on Two Hard Problems Using Schnorr's Signature Standard

Tuan Nguyen Kim[1,*], Duy Ho Ngoc[2], Nin Ho Le Viet[1], and Nikolay A. Moldovyan[3]

[1] School of Computer Science, Duy Tan University, Danang, Vietnam; Email: hlvnin@dtu.edu.vn
[2] Department of Information Technology, Hanoi, Vietnam; Email: hongocduy027@gmail.com
[3] St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences, St. Petersburg, Russia; Email: nmol@mail.ru
*Correspondence: nguyenkimtuan@duytan.edu.vn

***Abstract*—Many types of digital signature schemes have been researched and published in recent years. In this paper, we propose two new types of collective signature schemes, namely i) the collective signature for several signing groups and ii) the collective signature for several individual signings and several signing groups. And then we used two difficult problems factoring and discrete logarithm to construct these schemes. To create a combination of these two difficult problems we use the prime module p with a special structure: p = *Nn* + 1 with *n* = rq, *N* is an even number, *r* and *q* are prime numbers of at least 512 bit. Schnorr's digital signature scheme and the RSA key generation algorithm are used to construct related basic schemes such as the single signature scheme, the collective signature scheme, and the group signature scheme. The proposed collective signature schemes are built from these basic schemes. The correctness, security level and performance of the proposed schemes have also been presented in this paper.**

***Keywords*—Schnorr's signature, collective signature, group signature, signing group, individual signings**

I. Introduction

Assume that there is a collective made up of several groups, each of which has a large number of members and is managed by a group leader. There are another few individual members in this collective that do not belong to any groups, but they are functionally equivalent to the group leaders. The problem is how to create a single digital signature [1–3] that represents this collective. The requirement of digital signature-based authentication [4, 5] for a multi-functional collective is quite common in today's cyberspace. Both group signature protocols [6–14] and collective signature [15, 16] ones can be used to produce a unique signature for a group of multiple signers, but they cannot be used to generate a common signature for a multi-level signing collective as described above. The reason for this is that the group signature scheme can only create a common signature for each group, and the collective signature scheme [16, 17] can only generate a signature for the group leaders and individual members, or for all collective members [16].

Therefore, we propose a new type of multi-signature scheme, the representative collective signature scheme, which is structured from the combination of the group signature scheme and the collective signature scheme.

Two stages are required to create the representative collective signature. Firstly, the group signature protocol is used to establish group signatures for each group of the collective. The collective signature protocol is then used to generate collective signatures from each group and every other individual. The final signature represents a signing collective made up of several signing groups and individual signers, and it comprises the information of everyone who participated in the formation of this signature.

Most of the digital signature schemes can be built based on a difficult problem or at the same time two difficult problems [10, 14, 17]. In this article, we utilize Schnorr's digital signature standard [5] to develop two types of representative collective signature schemes using two tough challenges simultaneously. For the discrete logarithm problem [18–22], we use a specially structured prime modulo, $p = Nn + 1$, where N is an even number, $n = rq$, r and q are prime numbers of at least $512\ bits$ or $1024\ bits$, used as the signer's secret keys. When attempting to find r and q from n, the factorization problem [23, 24] is applied. In these schemes, we use the value pair (e, d), generated by the RSA algorithm, as additional keys, to ensure that the difficulty of the factorization problem always exists in the proposed collective digital signature schemes. This necessity was analyzed by Moldovyan *et al.* in [25].

II. The Related Base Digital Signature Schemes

The Schnorr's digital signature protocol is built on the difficult problem of the discrete logarithm in the prime fields, with the input parameter set selected according to the DSA digital signature standard, but without constraints on size and structure of p and q. We propose a

Manuscript received April 11, 2022; revised May 28, 2022; accepted July 15, 2022; published February 14, 2023.

modification from the Schnorr's scheme by i) Choosing prime modulus with special structure, $p = Nn + 1$, where N is an even number, $n = rq$, r and q are large prime numbers having the $512\ bits$ size or $1024\ bits$ size (the primes r and q are such that the value 3 does not divide $r - 1$ nor $q - 1$); ii) Change the expression for calculating the value S in the the signature generation procedure and iii) Change the expression R^* in the signature checking procedure (S is replaced by the parameter S^e). A new prime modulus has been used for constructing the randomized signature security of which is based on the factorization of the value $n = (p - 1)/2$. It should be noted that, it is still possible to use S^2 in place of S in the signature checking expression, but it is not always possible to find S satisfying the expression $S^2 = k + xE \ mod\ n$. So we have combined using RSA algorithm to solve this problem.

A. The Single Signature Scheme (The SDS-2.1 Scheme)

In this scheme we select the parameter α having the order $n\ modulo\ p$. The primes r and q are elements of the secret keys; The signer's private keys are (x, d); The signer's public keys are (e, α, y). Where e is a randomly chosen integer $e \in Z_n$ such that $\gcd(e, n) = 1$. Calculate a secret d such that $ed \equiv 1\ mod\ \phi(n)$; The signer randomly chooses a secret key x $(1 < x < n - 1)$, $x \in Z_p^*$. y is calculated as follows $y = \alpha^x\ mod\ p$.

Let F_H be a one-way hash function such as SHA-1 or SHA-2, which produces the hash value H from the document M: $H = F_H(M)$. The signature scheme based on factoring and discrete logarithm problems is described as below:

- The signature generation procedure on the document M

It includes the following steps:

1. The signer generates the random value k, $k < n$, and then computes the value R:

$$R = \alpha^k\ mod\ p \tag{1}$$

2. The signer computes the value E:

$$E = RH\ mod\ \delta, \tag{2}$$

where δ is a large prime, $|\delta| = 160$ bits; and H is a hash value of the document M.

The value E is the first part of the signature.

3. The signer computes the value S:

$$S = (k + xE)^d\ mod\ n \tag{3}$$

Such that:

$$R = \alpha^{S^e} y^{-E}\ mod\ p \tag{4}$$

The pair of value (E, S) is the signer's signature on the document M.

- The signature verification procedure on the document M

It includes the following steps (by the verifier):

1. The verifier computes the value R^*:

$$R^* = \alpha^{S^e} y^{-E}\ mod\ p \tag{5}$$

2. The verifier computes the value E^*:

$$E^* = R^* H\ mod\ \delta \tag{6}$$

The verifier compares values E^* with E. If $E^* = E$: The signature is valid; Otherwise, the signature is invalid. It is rejected.

- Proof of correctness of the SDS-2.1 scheme

To prove the correctness of this signature scheme we only need to prove the existence of the equation $E^* = E$.

It is easy to see $R^* = R$. Indeed:

$$\begin{aligned} R^* &= \alpha^{S^e} y^{-E} \bmod p \\ &= \alpha^{k+xE} \alpha^{-xE} \bmod p \\ &= \alpha^k\ mod\ p = R \end{aligned}$$

Since $R^* = R$ so $E^* = E$ ($E^* = R^* H\ mod\ \delta = RH\ mod\ \delta = E$) is always exists.

The correctness of the SDS.2-1 scheme has been proved.

The collective signature scheme described below (the CDS-2.2 scheme) is built on the basis of this signature scheme (the SDS-2.1 scheme).

B. The Collective Signature Scheme (the CDS-2.2 Scheme)

We assume that there are m signers in the signing collective, $1 \le i \le m$, to sign the same document M. Each signer randomly selects an integer x_i from the interval $[1, n - 1]$ and computes a corresponding public key: $y_i = \alpha^{x_i}\ mod\ p$ (x_i is the secret key of the i-th user). Other parameters, other keys, and other secret values are chosen as in the case of the single signature scheme (at Section II.*B*).

The collective signature scheme based on factoring and discrete logarithm problems (CDS-2.2) is described as below:

- The collective signature generation procedure on the document M

It includes the following steps:

1. Each signer selects a random number k_i, $k_i \in [1, n - 1]$, and then computes the value R_i:

$$R_i = \alpha^{k_i}\ mod\ p \tag{7}$$

The signer sends R_i to all other signers in the signing collective.

2. One of the signers in the signing collective calculates the common randomization value R:

$$R = \prod_{i=1}^{m} R_i\ mod\ p \tag{8}$$

And calculates the first part of the collective signature:

$$E = RH\ mod\ \delta \tag{9}$$

where δ is a large prime, $|\delta| = 160$ bits; and H is a hash value of the document m.

The value E is sent to all signers in the signing collective.

3. Each signer computes it's a shared signature S_i:

$$S_i = (k_i + x_i E)^d \bmod n \quad (10)$$

4. One of the signers in the signing collective calculates the second element of the collective digital signature S:

$$S = (S_1^e + S_2^e + \cdots + S_m^e)^d \bmod n \quad (11)$$

The pair of value (E, S) is the collective digital signature of the signing collective, there are m signers, on the message M.

- The signature verification procedure on the document M

It includes the following steps (by the verifier):

1. The verifier computes the collective public key y:

$$y = \prod_{i=1}^{m} y_i \bmod p \quad (12)$$

2. The verifier computes the value R^*:

$$R^* = \alpha^{S^e} y^{-E} \bmod p. \quad (13)$$

3. The verifier computes the value E^*:

$$E^* = R^* H \bmod \delta. \quad (14)$$

4. The verifier compares values E^* and E. If $E^* = E$: The signature is valid; Otherwise, the signature is invalid. It is rejected.

- Proof of correctness of the CDS-2.2 scheme

To prove the correctness of this signatue scheme we only need to prove the existence of the equation $E^* = E$. It is easy to see $R^* = R$. Indeed:

Substituting the value $S = (\sum_{i=1}^{m} {S_i}^e)^d \bmod n$ in the right part of the verification equation $R^* = \alpha^{S^e} y^{-E} \bmod p$, we get:

$$\begin{aligned} R^* &= \alpha^{S_1^e + S_2^e + \cdots + S_m^e} \prod_{i=1}^{m} y_i^{-E} \bmod p \\ &= \prod_{i=1}^{m} \alpha^{S_i^e} \prod_{i=1}^{m} \alpha^{x_i(-E)} \bmod p \\ &= \prod_{i=1}^{m} \alpha^{k_i + x_i E} \prod_{i=1}^{m} \alpha^{x_i(-E)} \bmod p \\ &= \prod_{i=1}^{m} \alpha^{k_i} \bmod p = \prod_{i=1}^{m} R_i \bmod p = R \end{aligned}$$

Since $R^* = R$ so $E^* = E$ ($E^* = R^* H \bmod \delta = RH \bmod \delta = E$) is always exists.

The correctness of the signature scheme has been proved.

It is easy to see that, in this scheme, none of the signers generates his/her individual signature. The signer generates only its shared signature in the collective signature that corresponds exactly to the given document M and to the assigned set of m users. Besides, it is computationally difficult to manipulate with shares $S_1, S_2, \ldots, S_m$, and compose another collective digital signature, relating to some different set of users.

III. The Proposed Signature Schemes

In this part, we first construct a group signature scheme for a signing group of m members using the group signature protocol provided in [9]. Then, we utilize this scheme and the collective signature scheme mentioned in Section II.*B*, as the basic schemes, to build two types of the representative collective signature scheme: i) the collective signature for several signing groups and ii) the collective signature for several individual signings and several signing groups

A. Constructing the Group Signature Scheme (GDS-3.1)

Suppose there is a signing group of m signers who want to sign the document M. Each of the signers selects a private key x. His/Her corresponding public key is $y_i = \alpha^{x_i} \bmod p$, $i = 1, 2, \ldots, m$. The public key Y of the group manager is a public key of the group and is calculated as follows $Y = \alpha^X \bmod p$, where X is the manager's private key. The value Y is used in the signature verification procedure of the GDS-3.1 scheme. Other parameters, other keys, and other secret values are chosen as in the case of the single signature scheme (at Section II.*B*).

Let F_H is some specified hash function.

The group signature scheme based on factoring and discrete logarithm problems (GDS-3.1) is described as follows:

- The group signature generation procedure on the document M

It consists of stages:

1. The group manager does the following tasks:

- Computes hash value from document M:

$$H = F_H(M) \quad (15)$$

- Calculates masking coefficients λ_i:

$$\lambda_i = F_H(H \,||\, y_i \,||\, F_H(H \,||\, y_i \,||\, X)) \quad (16)$$

- Sends each value λ_i to the corresponding *i-th* group member

- Computes the first element of the group signature U:

$$U = \prod_{i=1}^{m} y_i^{\lambda_i} \bmod p \quad (17)$$

2. Each *i-th* signer in the signing group does the following tasks:

- Generates a random number k_i, $k_i < n$, anh then computes the value R_i:

$$R_i = \alpha^{k_i} \bmod p \quad (18)$$

- Sends R_i to the group manager

3. The group manager does the following tasks:

- Generates the random number K, $K < q$, and then computes the values R', R, E:

$$R' = \alpha^K \bmod p \quad (19)$$

$$R = R' \prod_{i=1}^{m} R_i \bmod p = \alpha^{K+\sum_{i=1}^{m} k} \quad (20)$$

$$E = F_H(M||R||U) \bmod \delta \quad (21)$$

where δ is a large prime, |δ| = 160 bit.

- Sends value E to all signers in signing group

E is the second element of the group signature.

4. Each *i-th* signer in the signing group does the following tasks:

- Computes his/her shared signature S_i:

$$S_i = (k_i + x_i \lambda_i E)^d \bmod n \quad (22)$$

- Sends S_i to the group manager

5. The group manager does the following tasks:

- Verifies the correctness of each shared signature S_i by checking equality:

$$R_i = \alpha^{S_i{}^e} y^{-\lambda_i E} \bmod p \quad (23)$$

- If all signature shared signatures S_i satisfy the last verification equation, then he/she computes his shared signature:

$$S' = (K + XE)^d \bmod n \quad (24)$$

- Computes the third element of the group signature S:

$$S = (S'^e + \sum_{i=1}^{m} S_i{}^e)^d \bmod n \quad (25)$$

The tuple (U, E, S) is a group signature of the signing group on the document M.

- The signature verification procedure on the document M

It includes the following steps:

1. The verifier computes the hash function value from the document *M:* $H = F_{\mathrm{H}}(M)$

2. The verifier computes value R^*:

$$R^* = \alpha^{S^e} (UY)^{-E} \bmod p \quad (26)$$

3. The verifier computes value E^*:

$$E^* = F_H(M||R^*||U) \bmod \delta \quad (27)$$

4. The verifier compares the values E^* with E. If $E^* = E$: The group signature is valid; Otherwise, the group signature is invalid. It is rejected.

- Proof of correctness of the GDS 3.1 scheme

To prove the correctness of this signatue scheme we only need to prove the existence of the equation $E^* = E$. It is easy to see $R^* = R$. Indeed:

$$\begin{aligned} R^* &= \alpha^{S^e} (UY)^{-E} \bmod p \\ &= \alpha^{S'^e + \sum_{i=1}^{m} S_i{}^e} \left(\alpha^X \prod_{i=1}^{m} y_i{}^{\lambda_i} \right)^{-E} \bmod p \\ &= \alpha^{(K+XE)+\sum_{i=1}^{m} (k_i + x_i \lambda_i E)} \alpha^{-XE} \prod_{i=1}^{m} \alpha^{-x_i \lambda_i E} \bmod p \\ &= \alpha^{K+\sum_{i=1}^{m} k_i} \bmod p = R \end{aligned}$$

Since $R^* = R$ so $E^* = E$ ($E^* = R^* H \bmod \delta = RH \bmod \delta = E$) is always exists.

The correctness of the signature scheme has been proved

B. Constructing the Collective Digital Signature for Several Signing Groups

Let g signing groups with public keys $Y_j = \alpha^{X_j} \bmod p$, where $j = 1,2, \dots, g$. X_j is the secret key of the j-th group manager, have intention to sign the document M. Suppose also the *j-th* signing group inclues m_j active individual signers (persons appointed to act on behalf of the *j-th* signing group).

Other parameters, other keys, and other secret values are chosen as in the case of the single signature scheme (at Section II.*B*).

The collective signature scheme for several signing group (RCS.01-3) is described as below.

- The collective signature generation procedure on the document M

It consists of stages:

1. Each *j*-th *group* manager in the signing collective does the following tasks:

- Based on the group signature generation procedure described above (Section III.*A*) to generals masking parameters λ_{ji} for the signers of *j*-th *group.*

- Computes the value U_j (where $i = 1, 2, \dots, m_j$):

$$U_j = \prod_{i=1}^{m_j} y_{ji}^{\lambda_{ji}} \bmod p \quad (28)$$

U as the shared element of the *j-th* group in the first element of the collective signature.

- Comutes the randomizing parameter R_j:

$$R_j = R'_j \prod_{i=1}^{m_j} R_{ji} \bmod p \quad (29)$$

- Sends values U_j and R_j to all other group managers in the signing collective.

2. Each *j-th* group manager in the signing collective computes values U, R and E:

$$U = \prod_{j=1}^{g} U_j \bmod p \quad (30)$$

$$R = \prod_{j=1}^{g} R_j \bmod p = \alpha^{\sum_{j=1}^{g} k_j} \bmod p \quad (31)$$

and

$$E = F_H(M||R||U) \bmod \delta \quad (32)$$

U and E are the first and second elements of the collective signature.

3. Each *j-th* group manager does the following tasks:

- Computes the shared signature of j-th group:

$$S_j = (S_j'^e + \sum_{i=1}^{m_j} S_{ji}{}^e)^d \bmod n \quad (33)$$

where S_{ji} in the shared signature of the *i-th* signer in the *j-th* group.

- Sends S_j to other group managers in the signing collective.

4. Each *j-th* group manager does the following tasks:

- Can verify the correctness of each shared signature S_j by cheaking equality:

$$R_j^* = \alpha^{S_j{}^e}(U_j Y_j)^E \ mod\ p \quad (34)$$

- If all shared signatures S_j satisfy the last verification equation, then the third element S of the collective signature is computed:

$$S = \left(\sum_{j=1}^{g} S_j{}^e\right)^d \bmod n \quad (35)$$

The tuple (U, E, S) is the collective signature on the document M of the signing collective there are g signing groups.

- The signature verification procedure on the document M

It includes the following steps:

1. The verifier computes the collective public key shared by all signing groups:

$$Y_{col} = \prod_{j=1}^{g} Y_j \ mod\ p \quad (36)$$

2. The verifier computes the value R^*:

$$R^* = \alpha^{S^e}(UY_{col})^{-E} \ mod\ p \quad (37)$$

3. The verifier computes the value E^*:

$$E^* = F_H(M||R^*||U) \ mod\ \delta \quad (38)$$

4. The verifier compares the values E^* with E. If $E^* = E$: The collective signature is valid. Otherwise, the collective signature is invalid. It is rejected.

- Proof of correctness of the RCS.01-3 scheme

To prove the correctness of this signatue scheme we only need to prove the existence of the equation $E^* = E$. It is easy to see $R^* = R$. Indeed:

$$\begin{aligned} R^* &= \alpha^{S^e}(UY_{col})^{-E} \ mod\ p \\ &= \alpha^{\sum_{j=1}^{g} S_j{}^e}(\prod_{j=1}^{g} U_j Y_j)^{-E} \ mod\ p \\ &= \prod_{j=1}^{g} \alpha^{S_j^e}(U_j Y_j)^{-E} \ mod\ p \\ &= \prod_{j=1}^{g} R_j \ mod\ p \\ &= R \end{aligned}$$

Since $R^* = R$ so $E^* = E$ ($E^* = F_H(M||R^*||U) = F_H(M||\mathrm{R}||U) = E$) is always exists.

The correctness of the signature scheme has been proved.

C. *Constructing the Collective Digital Signature Scheme for Several Individual Signers and Several Signing Groups*

The collective signature generation procedure of this scheme is similar to that of the RCS.01-3 scheme, but for individual signers, *Uj* is equal to 1.

Suppose x_j and $y_j = \alpha^{x_j}$, where $j = g+1, g+2, \dots, g+m$, are a private key and a public key, correspondingly, of m individual signers participating in the protocol for generating the collective digital signature for g signing groups and *m* individual signers.

The collective signature scheme for m individual signers g signing groups (RCS.02-3) is described as below.

- The signature generation procedure on the document M

It consists of stages:

1. Each *j*-th *group* manager in the signing collective does the following tasks:

- Based on the group signature generation procedure described above (Section III.*A*) to generals masking parameters λ_{ji} for the signers of *j*-th *group.*

- Computes the value U_j (where $i = 1,2, \dots, m_j$):

$$U_j = \prod_{i=1}^{m_j} y_{ji}^{\lambda_{ji}} \ mod\ p \quad (39)$$

U as the shared element of the *j-th* group in the first element of the collective signature.

- Computes the randomizing parameter R_j:

$$R_j = R_j' \prod_{i=1}^{m_j} R_{ji} \ mod\ p \quad (40)$$

- Send values U_j and R_j to all other managers and all individual signers in the signing collective.

2. Each *j-th* individual signer ($j = g+1, g+2, \dots, g+m$) does the following tasks:

- Generates a random value K_j, $K_j < n$, and then computes the value R_j:

$$R_j = \alpha^{K_j} \ mod\ p \quad (41)$$

- Sent R_j to all group managers and other individual signers in the signing collective.

- Each *j-th* group manager and each *j-th* individual signer in the signing collective computes values U, R and E:

$$U = \sum_{j=1}^{g+m} U_j \ mod\ p \quad (42)$$

$$R = \sum_{j=1}^{g+m} R_j \ mod\ p \quad (43)$$

$$E = F_H(M||R||U) \bmod \delta \quad (44)$$

where δ is a large prime having, $|\delta| = 160$ bits; $U = 0$ for $j = g+1, g+2, \dots, g+m$. U and E are *the first and second elements* of the signature.

3. a) Each *j-th* group manager computes the shared signature of *j-th* group S_j:

$$S_j = (S_j'^e + \sum_{i=1}^{m_j} S_{ji}{}^e)^d \bmod n \quad (45)$$

where S_{ji} is the shared signature of the *i-th* signer in the *j-th* signing group.

And sends S_j to all individual signers and other group managers.

b) Each *j-th* individual signer computes his/her shared signature S_j:

$$S_j = (K_j + X_j E)^d \bmod n \quad (46)$$

And sends S_j to all group managers and other individual signers.

4. Each *j-th* group manager and each individual signers do the following tasks:

- Can verify the correctness of each share signatures S_j by checking equality:

$$R_j^* = \alpha^{S_j^e}(U_j Y_j)^{-E} \bmod p \quad (47)$$

For $j = 1, 2, \dots, g$ and

$$R_j^* = \alpha^{S_j^e} Y_j^{-E} \bmod p \quad (48)$$

For $j = g+1, g+2, \dots, g+m$.

- If all shares S_j satisfy the last verification equation, then the third element S of the collective signature is computed:

$$S = (\sum_{j=1}^{g+m} S_j{}^e)^d \bmod n \quad (49)$$

The tuple (U, E, S) is the collective signature on the document M of the signing collective there are g signing groups and m individual signers.

The first element U of the collective signature contains information about the all group members of each signing group who signed the document M.

- The signature verification procedure on the document M

It includes the following steps:

1. The verifier computes the collective public key shared by all signing groups and individual signers:

$$Y_{col} = \prod_{j=1}^{g+m} Y_j \bmod p \quad (50)$$

2. The verifier computes the value R^*:

$$R^* = \alpha^{S^e}(UY_{col})^{-E} \bmod p \quad (51)$$

3. The verifier computes the value E^*:

$$E^* = F_H(M|| R^*|| U) \quad (52)$$

4. The verifier Compares the value E^* with E. If $E^* = E$: The collective signature is valid; Otherwise, the collective signature is invalid. It is rejected.

- Proof of correctness of the RCS.02-3 scheme

To prove the correctness of this signatue scheme we only need to prove the existence of the equation $E^* = E$. It is easy to see $R^* = R$. Indeed:

$$\begin{aligned} R^* &= \alpha^{S^e}(UY_{col})^{-E} \bmod p \\ &= \alpha^{\Sigma_{j=1}^{g+m} S_j{}^e}(\prod_{j=1}^{g} U_j \prod_{j=1}^{g+m} Y_j)^{-E} \bmod p \\ &= \alpha^{\Sigma_{j=1}^{g} S_j{}^e + \Sigma_{j=g+1}^{g+m} S_j{}^e}(\prod_{j=1}^{g} U_j \prod_{j=1}^{g} Y_j \prod_{j=g+1}^{g+m} Y_j)^{-E} \bmod p \\ &= \prod_{j=1}^{g} \alpha^{S_j{}^e}(U_j Y_j)^{-E} \prod_{j=g+1}^{g+m} \alpha^{S_j{}^e} Y_j{}^{-E} \bmod p \\ &= \prod_{j=1}^{g} R_j \prod_{j=g+1}^{g+m} R_j \bmod p \\ &= R \end{aligned}$$

Since $R^* = R$ so $E^* = E$ ($E^* = F_H(M|| R^*|| U) = F_H(M|| \mathrm{R}|| U) = E$) is always exists.

The correctness of the signature scheme has been proved.

IV. Security Analysis and Performance Evaluation

A. Security Advantages of the Proposed Collective Digital Signature Schemes

We first analyze the security advantages of the proposed group signature scheme (GDS-3.1). The operation this scheme shows that it has the following security advantages:

- The digital signature schemes in this paper are built based on the combination of two digital signature standards RSA and Schnorr's, so they inherit all the security advantages of two difficult problems factoring and discrete logarithmat the same time.
- There is no need to exchange or share security values, private keys, or secret keys between members of a signing group or between members of a signing group with the manager of that signing group. Therefore, the Internet environment is sufficient to implement this scheme.
- Signing group members and group managers can both use a pair of their private key and public key for both purposes: Forming private signatures and participating in group signature formation. As a result, this scheme can be fully deployed on existing PKI systems.
- Using the group manager's public key Y as the public key of the signing group makes it possible both to check the validity of the signature (of the verifier) and to change the set of participants that form the signature (of the group manager) have become much more convenient.

- The process of forming components, especially the S component, of the signature is done through 2 steps: i) First, all the signing group members, designated, participate in the creation of the signature. group digital pre-signature (Group digital pre-signature) under the control of the group leader and ii) Then, the group manager proceeds to create a group digital signature (Group digital signature) of the signing group, after confirming the correctness of signatures of all members. Of course, the final group signature includes the information and signature of the group leader. This proves it is difficult to fake a member signing with this scheme. At the same time, the responsibility and representativeness of the team leader here are very high.
- The U component of the signature contains the information of all the members of the signing group who participated in the formation of the group signature. Therefore, to identify this membership set, the group manager only needs to "open" the U component for review. This "opening" can only be done by the group manager because in U contains keys that contain his private key X. This means, the information of the members who have participated in the formation of the group signature is kept secret by the group manager.

The proposed representative collective signature scheme inherits all these security advantages of the group signature scheme

B. Performance Evaluation of the Proposed Collective Digital Signature Schemes

The performance of a digital signature scheme can be evaluated by calculating the time cost of signature generation and the time cost of signature verification. We do it this way. The time costs of representative collective signature schemes proposed in this paper are shown in Table I.

TABLE I. TIME COST OF THE PROPOSED COLLECTIVE SIGNATURE SCHEME: RCS.01-3 AND RCS.02-3

The scheme	Time cost for	
	Signature generation	Signature verification
RCS. 01-3	$U = \sum_{j=1}^{g}(243m_j + 1)\,T_m$ $E = [\sum_{j=1}^{g}(241m_j + 240) + 1]T_m$ $S = [\sum_{j=1}^{g}(1204m_j + 1731) + 240]T_m$ $Sum = [\sum_{j=1}^{g}(1688m_j + 1972) + 241]T_m$	$(723 + g)T_m$
RCS. 02-3	$U = \sum_{j=1}^{g}(243m_j + 1)\,T_m$ $E = [\sum_{j=1}^{g}(241m_j + 240) + 241m + 1]T_m$ $S = [\sum_{j=1}^{g}(1204m_j + 1731) + 1200m + 240]T_m$ $Sum = [\sum_{j=1}^{g}(1688m_j + 1972) + 1441m + 241]T_m$	$(723 + g + m)T_m$

Notations: T_h: Time cost of a hash operation in Z_p; T_{inv}: Time cost of a inverse operation in Z_p; T_e: Time cost of an exponent operation in Z_p; T_m: Time cost of a modular multiplication in Z_p. According to [26]: $T_h \approx T_m, T_{inv} \approx 240T_m, T_e \approx 240T_m$.

Table I shows that the time cost for the generation of signature components and for the signature verification of the proposed collective signature schemes are is much higher than that of the similar signature scheme in [24]. This is considered as a limitation that needs to be overcome for schemes built on two difficult problems factoring and discrete logarithm [23, 27].

V. CONCLUSION

In this paper, we have succeeded in using simultaneously two difficult problems factoring and discrete logarithm to build two types of representative collective signature schemes, which are: i) the collective signature scheme for many signing groups and ii) the collective signature scheme for many individual signers and many signing groups. These types of schemes are essential for the multi-level authentication requirements of many information exchange applications in today's network environment [12]. In addition, the proposed signature schemes is also easy to deploy on existing PKI systems [19].

The simultaneous combination of two difficult problems factoring and discrete logarithm is demonstrated by choosing a prime modulo p with a special structure, $p = \mathrm{N}n + 1$, with $n = rq$, r and q are large prime numbers having the 512 bit size or 1024 bit and add the key pair (e, d) from the RSA algorithm. The security level of the proposed collective signature schemes is inherited from the base scheme which has been analyzed in Section IV.*A*. That is, to break the proposed collective signature scheme, the attacker must also solve two difficult problems simultaneously. The paper also calculated and compared the performance of the two proposed schemes with the performance of some other schemes and also analyzed the security advantages of this scheme. Currently, we have not found a study related to the representative collective digital signatures, so we cannot yet compare the performance of the proposed signature scheme with the performance of similar signature schemes. We will do this in the future.

CONFLICT OF INTEREST

The authors declare that they have no conflicts of interest to report regarding the present study.

AUTHOR CONTRIBUTIONS

All authors contributed to the formation of this article. Specifically as follows: Tuan Nguyen Kim and Nin Ho Le Viet build the representative collective signature schemes and proves the correctness of these schemes; Nikolay A. Moldovyan builds the two-component group signature scheme; Duy Ho Ngoc analyzes security and evaluates the

performance of the proposed schemes; all authors had approved the final version.

FUNDING

We received funding for this research from Duy Tan University, Danang, Vietnam. https://duytan.edu.vn/.

REFERENCES

[1] J. Pieprzyk, T. Hardjono, and J. Seberry, *Fundamentals of Computer Security*, Heidelberg: Springer-Verlag, 2003.
[2] K. Ganeshkumar and D. Arivazhagan, "Generating a digital signature based on new cryptographic scheme for user authentication and security," *Indian Journal of Science and Technology*, 2014.
[3] National Institute of Standards & Technology, "Digital signature standard," Federal Information Processing Standards Publication 186-3, 2009.
[4] M. Girault, G. Poupard, and J. Stern, "On the fly authentication and signature schemes based on groups of unknown order," *Journal of Cryptology*, no. 19, pp. 463-487, 2006.
[5] C. P. Schnorr, "Efcient signature generation by smart-cards," *Journal of Cryptology*, vol. 4, no. 3, pp. 161-174, 1991.
[6] R. Seetha and R. Saravanan, "Digital signature schemes for group communication: A survey," *International Journal of Applied Engineering Research*, no. 11, pp. 4416-4422, 2016.
[7] A. C. Enache, "About group digital signatures," *Journal of Mobile, Embedded and Distributed Systems*, no. 4, pp. 193-202, 2012.
[8] Q. Alamélou, O. Blazy, S. Cauchie, *et al.*, "A code-based group signature scheme," *Designs. Codes and Cryptography*, vol. 82, no. 1-2, 2017.
[9] A. A. Moldovyan and N. A. Moldovyan, "Group signature protocol based on masking public keys," *Quasigroups and Related Systems*, no. 22, pp. 133-140, 2014.
[10] N. Tahat, E. Ismail, and R. Ahmad, "A new blind signature scheme based on factoring and discrete logarithms," *International Journal of Cryptology Research*, vol. 1, no. 1, pp. 1-9, 2009.
[11] R. Xie, C. Xu, C. He, *et al.*, "A new group signature scheme for dynamic membership," *International Journal of Electronic Security and Digital Forensics*, vol. 8, no. 4, 2016.
[12] D. Chaum, "Blind signatures for untraceable payments," in *Proc. of CRYPTO'82, Advances in Cryptology*, Plenum Press, 1983, pp. 199-203.
[13] R. S. Rajasree, "Generation of dynamic group digital signature," *International Journal of Computer Applications*, no. 98, pp. 1-5, 2014.
[14] N. Minh, D. Binh, N. Giang, *et al.*, "Blind signature protocol based on difficulty of simultaneous solving two difficult problems," *Journal of Applied Mathematical Sciences*, vol. 6, no. 139, pp. 6903-6910, 2012.
[15] N. A. Moldovyan, N. H. Minh, D. T. Hung, *et al.*, "Group signature protocol based on collective signature protocol and masking public keys mechanism," *International Journal of Emerging Technology and Advanced Engineering*, no. 6, pp. 1-5, 2016.
[16] N. A. Moldovyan, "Blind collective signature protocol," *Computer Science Journal of Moldova*, no. 19, pp. 80-91, 2011.
[17] A. Berezin, N. A. Moldovyan, and S. Victor, "Cryptoschemes based on difficulty of simultaneous solving two different difficult problems," *Computer Science Journal of Moldova*, vol. 21, no. 2, pp. 280-290, 2013.
[18] J. L. Camenisch, J. M. Piveteau, and M. A. Stadler, "Blind signatures based on the discrete logarithm problem," in *Proc: Advances in Crypology – EUROCRYPT'94 Proc, Lecture Notes in Computer Science*, Berlin Heidelberg New York: Springer-Verlag, 1995, vol. 950, pp. 428-432.
[19] S. Selvakumaraswamy and U. Govindaswamy, "Efficient transmission of pki certificates using ecc and its variants," *The International Arab Journal of Information Technology*, vol. 13, no. 1, pp. 38-43, 2016.
[20] A. B. Nimbalkar, "The digital signature schemes based on two hard problems: factorization and discrete logarithm," *Advances in Intelligent Systems and Computing, Cyber Security*, vol. 729, pp. 493-498, 2018.
[21] N. A. Moldovyan, "Blind signature protocols from digital signature standards," *Journal of Network Security*, no. 13, pp. 22-30, 2011.
[22] N. A. Moldovyan and A. A. Moldovyan, "Blind collective signature protocol based on discrete logarithm problem," *Journal of Network Security*, no. 11, pp. 106-113, 2010.
[23] N. A. Moldovyan, "Digital signature scheme based on a new hard problem," *Computer Science Journal of Moldova*, no. 16, pp. 163-18, 2008.
[24] N. K. Tuan, V. L. Van, D. N. Moldovyan, *et al.*, "Collective signature protocols for signing groups," in *Information Systems Design and Intelligent Applications*, Singapore: Springer, 2018.
[25] D. V. Binh, N. H. Minh, and N. A. Moldovyan, "Digital signature schemes from two hard problems," in *Multimedia and Ubiquitous Engineering*, Dordrecht: Springer, 2013, pp. 817-825.
[26] C. Popescu, "Blind signature and BMS using elliptic curves," *Studia univ babes–bolyai, Informatica*, pp. 43-49, 1999.
[27] J. Lee, H. Kim, Y. Lee, *et al.*, "Parallelized scalar multiplication on elliptic curves defined over optimal extension field," *International Journal of Network Security*, vol. 4, pp. 99-106, 2017.

Tuan Nguyen Kim was born in 1969, received B.E, and M.E from Hue University of Sciences in 1994, and Hanoi University of Technology in 1998. He has been a lecturer at Hue University since 1996. From 2011 to the present, he is a lecturer at School of Computer Science, Duy Tan University, Da Nang, Vietnam. His main research interests include Computer Network Technology and Information Security.

Duy Ho Ngoc was born in 1982. He received his Ph.D. in Cybersecurity in 2007 from LETI University, St. Petersburg, Russia Federation. He has authored more than 45 scientific articles in cybersecurity.

Nin Ho Le Viet was born in 1988, received B.E from Da Nang University in 2011. In 2015, he received his MSIT from Duy Tan University. He has been a faculty at Duy Tan University since February 2016. His main research interests include is Software Engineering, Information Security and Machine learning for Cybersecurity.

Nikolay A. Moldovyan is an honored inventor of Russian Federation (2002), a laboratory head at St. Petersburg Institute for Informatics and Automation of Russian Academy of Sciences, and a Professor with the St. Petersburg State Electrotechnical University. His research interests include computer security and cryptography. He has authored or co-authored more than 60 inventions and 220 scientific articles, books, and reports. He received his Ph.D. from the Academy of Sciences of Moldova (1981).