Computers, Materials & Continua
DOI: 10.32604/cmc.2022.023168
Article

Tech Science Press

New Collective Signatures Based on the Elliptic Curve Discrete Logarithm Problem

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

[1]School of Computer Science, Duy Tan University, Da Nang, Vietnam
[2]Department of Information Technology, Ha Noi, Vietnam
[3]ITMO University, St. Petersburg, Russia
*Corresponding Author: Tuan Nguyen Kim. Email: nguyenkimtuan@duytan.edu.vn
Received: 30 August 2021; Accepted: 26 November 2021

Abstract: There have been many digital signature schemes were developed based on the discrete logarithm problem on a finite field. In this study, we use the elliptic curve discrete logarithm problem to build new collective signature schemes. The cryptosystem on elliptic curve allows to generate digital signatures with the same level of security as other cryptosystems but with smaller keys. To extend practical applicability and enhance the security level of the group signature protocols, we propose two new types of collective digital signature schemes based on the discrete logarithm problem on the elliptic curve: i) the collective digital signature scheme shared by several signing groups and ii) the collective digital signature scheme shared by several signing groups and several individual signers. These two new types of collective signatures have combined the advantages of group digital signatures and collective digital signatures. These signatures have a fixed size and do not depend on the number of members participating in the creation of the final collective signature. One of the advantages of the proposed collective signature protocols is that they can be deployed on top of the available public key infrastructures.

Keywords: Elliptic curve; signing group; individual signer; collective signature; group signature

1 Introduction

One of the important applications of asymmetric cryptosystems is digital signatures (DS). Authenticatic systems based on digital signatures are quite popular today. Digital signature protocols are developed based on difficult problems in mathematics such as the factorization problem of a large integer, the discrete logarithmic problem on finite field, the discrete logarithmic problem on the elliptic curve, or the problem of finding the roots modulo large primes (a new hard problem). To meet the authentic requirements of many different practical applications, many types of different digital signature schemes have been researched and published: Single (individual) digital signature schemes [1]; Blind digital signature schemes [2,3]; Group digital signature schemes [4–6]. Collective digital signature schemes [3].

In practice there are documents that need i) to be signed by several different signing groups; ii) to be signed by a number of different signing groups and a number of different individual signers. We rely on the advantages of the collective digital signature and the advantages of the group digital signature to build signatures for these cases. This is still a single digital signature but signed by i) all the given signing groups; ii) all the given signing groups and the individual signers. In this paper, we propose the novel digital signature schemes that help to generate signatures for the two cases just mentioned.

The discrete logarithm problem on the elliptic curve has been standardized in the standard ECDSA [7,8] and the standard GOST R34.10-2001 [9]. We use both standards to build the proposed collective signature schemes.

We use a combination of the collective signature generation technique [10] and the group signature generation technique [6] in building the new collective signature schemes. In these schemes, the public key of signers is masked. This is similar to the blind signature protocols described in [6].

2 Related Basis Collective Digital Signature Schemes

The digital signature protocol on the elliptic curve be described in detail at [11]. We based on this protocol to propose protocols of collective digital signature of a signing collective. There are two schemes built here, one based on ECDSA standard and one based on GOST R34.10-2001 standard. These schemes are used to build the collective signature schemes for signing groups and for individual signers.

2.1 Constructing the Collective Signature Scheme Based on the Standard ECDSA

Let the elliptic curve domain $D = (p, a, b, P, q, h)$. Suppose that there are n signers who want to sign the same message m. Each signer randomly selects an integer d_i, $d \in [1, q-1]$, $1 \leq i \leq n$, and calculates the corresponding public key as the point Q_i: $Q_i = d_iP$. Let F_H is a one-way hash function, SHA-1 or SHA-2. The collective digital signature scheme using ECDSA includes the following procedures:

- *The signature generation procedure on the message m*

1. Each signer selects random number $k_i \in [1, q-1]$ and computes: $R_i = k_iP$ and sends (broadcasts) R_i to all signers.
2. One of a signing collective calculates the common randomization value as the point R:

$$R = R_1 + R_2 + \cdots + R_n = (x_R, y_R) \tag{1}$$

3. He/She calculates the first part of the collective digital signature e: $e = x_RH \mod \delta$; with δ is a large prime. The length of δ is $|\delta| = 160$ bits; H is a hash value calculated from the message m: $H = F_H(m)$. The value e is broadcasted to the other signer.
4. Each signer computes his/her shared signature s_i, using the secret values d_i, k_i and the value e, as follows:

$$s_i = (k_i - ed_i) \mod q \tag{2}$$

5. One of a signing collective calculates the second element of the CDS:

$$s = s_1 + s_2 + \cdots + s_n \mod q \tag{3}$$

The pair ($e,\ s$) is the CDS on message M.

- *The signature verification procedure on the message m*

This protocol use the collective public key y.

1. The first, calculates the collective public key Q:

$$Q = Q_1 + Q_2 + \cdots + Q_n \tag{4}$$

2. Using the signature ($e,\ s$) calculates value R':

$$R' = eQ + sP = (x_{R'}, y_{R'}) \tag{5}$$

3. Calculate e':

$$e' = x_{R'}H \ mod\ \delta \tag{6}$$

4. Comparing e' with e.

If $e' = e$, then the collective signature on the message m is valid. Otherwise the signature is invalid.

- *Proof of correctness of CDS*

Let us show that the proposed protocol generates the correct CDS ($e,\ s$).

Substituting the point $Q = Q_1 + Q_2 + \cdots + Q_n$ and the value $s = s_1 + s_2 + s_3 + \cdots + s_n$ in the right part of the verification equation $R' = eQ + sP$ we get:

$$R' = eQ + sP = e\sum_{i=1}^{n} d_iP + \sum_{i=1}^{n}(k_i - ed_i)P = \sum_{i=1}^{n} k_iP = R$$

Thus, the correctness of this protocol has been proved.

2.2 Constructing the Collective Signature Scheme Based on the Standard GOST R34.10-2001

Russian signature standard GOST R34.10-2001 specifies the DS algorithm based on the elliptic curve (EC) defined over the finite field $GF(p)$ with the equation $y^2 = x^3 + ax + b\ mod\ p$, where $a, b \in GF(p)$ and x and y are coordinates of the points on the elliptic curve.

Given an elliptic curve satisfying the requirements of the standard GOST R34.10-2001. Let the point G on this EC. G has the large prime order q. There are n members in the signing collective. Each of signers selects a private key k_j. His/her public key is Q_j: $Q_j = k_jG, j \in [1\ ,\ n]$. The collective signature scheme using the standard GOST R34.10-2001 includes the following procedures:

- *The signature generation procedure on the message m*

1. Each of signers selects a random value t_j. The he/she calculates a point on the EC:

$$R_j = t_jG; where j \in [1\ ,\ n] \tag{7}$$

2. He/She calculates the collective public key Q of the signing collective:

$$Q = Q_1 + Q_2 + \cdots + Q_n \tag{8}$$

(Q_j is the public key of individual signers; $j \in [1\ ,\ n]$)

3. He/She calculates a value R as follows:

$$R = R_1 + R_2 + \cdots + R_n \tag{9}$$

and then calculates a value r:

$$r = x_{\mathrm{Q}} x_{\mathrm{R}}\ mod\ q, \tag{10}$$

where $x_R(x_Q)$ is the abscissa of the points $R(Q)$ on the elliptic curve.

The value r is the first element of the collective digital signature.

4. Each of signers calculates his/her shared signature s_j in the collective digital signature as:

$$s_j = (rk_j + t_j e) \bmod q, \tag{11}$$

where $e\ =\ H\ mod\ q$ and H is the hash value of the signing document.

5. The second element of the signature is s:

$$s = s_1 + s_2 + \cdots + s_m\ mod\ q. \tag{12}$$

The pair $(r,\ s)$ is the collective digital signature on message M.

- *The signature verification procedure on the message m*

1. Calculates the collective public key Q:

$$Q = Q_1 + Q_2 + \cdots + Q_n \tag{13}$$

2. Calculates the point on the EC $R*$:

$$R^* = (se^{-1} \bmod q)G + ((q\ -\ r)e^{-1} \bmod q)Q \tag{14}$$

3. Calculates the value:

$$r^* = x_Q x_{R^*}\ mod\ q \tag{15}$$

and compares r^* and r. If $r^* = r$, then the collective signature on the message m is valid. Otherwise the signature is invalid.

- *Proof of correctness of CDS*

Let us show that the proposed protocol generates the correct CDS $(r,\ s)$.

Substituting the point $Q = Q_1 + Q_2 + \cdots + Q_n$ and the value $s\ = s_1 + s_2 + s_3 + \cdots + s_n$ in the verification equation R^*, we get:

$$\begin{aligned}
R^* &= (se^{-1}\ mod\ q)G + ((q - r)e^{-1}\ mod\ q)Q \\
&= \left(\sum_{j=1}^{n} s_j e^{-1}\ mod\ q\right) G + ((q - r)e^{-1} mod\ q\ ) \sum_{j=1}^{n} Q_j \\
&= \left(\sum_{j=1}^{n} (rk_j + t_j e)e^{-1} G\ mod\ q\right) + \left(\sum_{j=1}^{n} (k_j G)(q - r)e^{-1}\ mod\ q\right) \\
&= \left(\sum_{j=1}^{n} (Grk_j e^{-1} + t_j G)\ mod\ q\ \right) + \left(\sum_{j=1}^{n} (Gqk_j e^{-1} - Grk_j e^{-1})\ mod\ q\ \right) \\
&= \sum_{j=1}^{n} t_j G\ mod\ q = R
\end{aligned}$$

Thus, the correctness of this protocol has been proved.

3 The Proposed Group Digital Signature Schemes

In this section, we rely on the ECDSA standard [8] and the GOST standard R34.10-2001 to propose two new types of digital signature schemes for a group of signers. We are based on these basis schemes to construct the new collective signature schemes.

3.1 The Group Digital Signature Scheme Based on the Standard ECDSA

Given an EC satisfying the requirements of the standard and the point G on the EC. G has the large prime order $\boldsymbol{q}$. There are m members in the signing group who want to sign the document M. Each of signers selects a private key $\boldsymbol{k_j}$. His/Her public key is $\boldsymbol{P_i}$: $\boldsymbol{P_i = k_iG, i = 1, \ldots, m}$. The group manager calculates his/her public key as the point $\boldsymbol{L = zG}$, where $\boldsymbol{z}$ is his/her private secret key. The GDS scheme using ECDSA includes the following procedures:

- *The signature generation procedure on the document M*

This procedure includes the following steps:

1. The group manager computes a hash value from the document: $H = F_H(M)$, where F_H is some hash function. Then he/she calculates values λ_i (masking coefficients):

$$\lambda_i = F_H(\mathrm{H}||x_{P_i}||F_H(\mathrm{H}||x_{P_i}||\mathrm{z})) \tag{16}$$

and the group manager sends the values λ_i to the i-th group member, where $i = 1, 2, \ldots, m$. Then he/she calculates the first element of the group signature U:

$$U = \lambda_1 P_1 + \lambda_2 P_2 + \cdots + \lambda_m P_m \tag{17}$$

2. Each of signers generates a random number $\rho_i < q$, *where* $i = 1, 2, \ldots, m$. And then he/she calculates the value:

$$R_i = \rho_i G \tag{18}$$

and he/she sends R_i to the group manager.

3. The next, the group manager selects the random number $\rho' < q$. Then he/she calculates the values:

$$R' = \rho' G \tag{19}$$

$$R = R' + R_1 + R_2 + \cdots + R_m \tag{20}$$

and

$$e = F_H(M||x_R||x_U) \bmod \delta \tag{21}$$

where δ is a large prime. The length of δ is $|\delta| = 160$ bits, x_R *and* x_U–abscissa of points R and U.The next element of the group signature is e. Then he/she sends this element (e) to all members of the group.

4. Each i-th signer calculates his/her shared signature (where $i = 1, 2, \ldots, m$):

$$s_i = \rho_i - e\lambda_i k_i \bmod q \tag{22}$$

and he/she sends s_i to the signing group manager.

5. The signing group manager uses the checking Eq. (20) to verify the correctness of each shared signature (s_i):

$$R_i = \lambda_i e P_i + s_i G \tag{23}$$

If all shared signatures s_i satisfy the Eq. (20), then he/she calculates his/sher share signature:

$$s' = \ \rho' + ze \ mod \ q \tag{24}$$

and he/she calculates s. s is the last element of the group signature:

$$s = s' + s_1 + s_2 + \cdots + s_m \ mod \ q \tag{25}$$

The triple (U, e, s) is the group digital signature of the signing group on the document M.

- *The signature verification procedure on the document M*

The verification procedure includes the following steps:

1. The verifier calculates the hash value from the document M: $H = F_H(M)$. Using the group public key L and the group signature (U, e, s) he/she calculates a value R^*:

$$R^* = sG - e(U + L) \tag{26}$$

2. He/She calculates a value e^*:

$$e^* = F_H(M||x_{R^*}||x_U) \ mod \ \delta \tag{27}$$

3. *Comparing* e with e^*. If $e^* = e$, then the signature on the document M is valid. Otherwise, the signature on the document M is invalid. He/She rejects the signature.

- *Proof of correctness of GDS*

Substituting the value $s = s' + s_1 + s_2 + \cdots + s_m \ mod \ q$, $U = \lambda_1 P_1 + \lambda_2 P_2 + \cdots + \lambda_m P_m$ and $L = zG$ in the verification Eqs. (3.11)–(26):

$$R^* = sG - e(U + L)$$

We get:

$$\begin{aligned} R &= (s' + s_1 + s_2 + \cdots + \ s_m)G \ - e(\lambda_1 P_1 + \lambda_2 P_2 \ + \cdots + \lambda_m P_m \ + zG) \\ &= (\rho' + ze + \rho_1 - e\lambda_1 k_1 + s_2 + \rho_2 \ - e\lambda_2 k_2 + \cdots + s_m + \rho_m - e\lambda_m k_m)G \\ &= R' + R_1 + R_2 + \cdots + R_m = R - e(\lambda_1 k_1 G + \lambda_2 k_2 G + \cdots + k_m z_m G + zG) \end{aligned}$$

and compute:

$$\begin{aligned} e^* &= F_H(M||x_{R^*}||x_U) \ mod \ \delta \\ &= F_H(M||x_R||x_U) \ mod \ \delta = e \end{aligned}$$

Thus, the correctness of the protocol has been proved.

3.2 The Group Digital Signature Scheme Based on the Standard GOST R34.10-2001

Given an EC satisfying the requirements of the standard and the point $\boldsymbol{G}$ on the EC. $\boldsymbol{G}$ has the large prime order $\boldsymbol{q}$. There are m members in the signing group who want to sign the document M. Each of signers selects private key $\boldsymbol{k_j}$. His/Her public key is $\boldsymbol{P_i}$: $\boldsymbol{P_i = k_iG,\ i = 1, \ldots, m}$. The group manager calculates his/her public key as the point $\boldsymbol{L = zG}$, where $\boldsymbol{z}$ is his/her private secret key. The GDS scheme using GOST R34.10-2001 includes the following procedures:

- *The signature generation procedure on the document M*

This procedure includes the following steps:

1. The group manager computes a hash value from the document: $H = F_H(M)$, where F_H is some hash function. The he/she calculates values λ_i (masking coefficients):

$$\lambda_i = F_H(H||x_{P_i}||F_H(H||x_{P_i}||z)) \tag{28}$$

and the group manager sends each value λ_i to the i-th group member, $wherei = 1, 2, \ldots, m$. Then he/she calculates the first element of the group signature U:

$$U = \lambda_1 P_1 + \lambda_2 P_2 + \cdots + \lambda_m P_m \tag{29}$$

2. Each of signers generates a random number $\rho_i < q$, $where\ i = 1, 2, \ldots, m$. And then he/she calculates the value:

$$R_i = \rho_i G(3.15) \tag{30}$$

and he/she sends R_i to the group manager.

3. The signing group manager selects the random number $\rho' < q$. The next he/she calculates the values:

$$R' = \rho' G, \tag{31}$$

$$R = R' + R_1 + R_2 + \ldots + R_m, \tag{32}$$

and

$$e = F_H(M||x_R||x_U)\ mod \delta, \tag{33}$$

where δ is a large prime. The length of δ is $|\delta| = 160$ bits, x_R and x_U are abscissa of points R and U.The next element of the group signature is e. Then he/she sends this element (e) to all members of the group.

4. Each i-th signer calculates his/her shared signature $(i = 1, 2, \ldots, m)$:

$$s_i = \lambda_i k_i + e\rho_i\ mod\ q \tag{34}$$

and then sends s_i to the signing group manager.

5. The signing group manager uses the checking Eq. (31) to verify the correctness of each shared signature (s_i):

$$R_i = s_i G e^{-1} - \lambda_i e^{-1} P_i \tag{35}$$

If all shared signatures s_i satisfy the Eq. (31), then he/she calculates his/her shared signature:

$$s' = e\rho' + z\ mod\ q \tag{36}$$

and he/she calculates s. s is the last element of the group signature:

$$s = s' + s_1 + s_2 + \cdots + s_m \; mod \; q \tag{37}$$

The triple $(U, \; e, \; s)$ is the group digital signature of the signing group on the document M.

- *The signature verification procedure on the document M*

This verification includes the following steps:

1. The verifier computes the hash value from the document M: $H \; = \; F_H(M)$. Using the group public key L and group signature $(U, \; e, \; s)$ he/she calculates a value R^*:

$$R^* = sGe^{-1} - (U + L)e^{-1} \tag{38}$$

2. He/She calculates a value e^*:

$$e^* = F_H(M||x_{R^*}||x_U) \; mod \; \delta \tag{39}$$

3. Comparing e with e^*. If $e^* = e$, then the signature on the document M is valid. Otherwise, the signature on the document M is invalid. He/She rejects the signature.

- *Proof of correctness of GDS*

Substituting the values $s \; = \; s' \; + s_1 + s_2 + \cdots + s_m \; mod \; q$, $U \; = \; \lambda_1 P_1 + \lambda_2 P_2 + \cdots + \lambda_m P_m$ and $L = zG$ in the verification Eq. (34):

$$R^* = sGe^{-1} - (U + L)e^{-1}$$

We get:

$$\begin{aligned} R^* &= (s' + s_1 + s_2 + \cdots + \; s_m)Ge^{-1} - (\lambda_1 P_1 + \lambda_2 P_2 \; + \cdots + \lambda_m P_m \; + zG)e^{-1} \\ &= (e\rho' + z + \lambda_i k_i + e\rho_i + \lambda_i k_i + e\rho_i + \cdots + \lambda_i k_j + e\rho_i)Ge^{-1} - (\lambda_1 k_1 G + \lambda_2 k_2 G + \cdots + \lambda_m k_m G + zG)e^{-1} \\ &= (e\rho' + z + \lambda_j k_j + e\rho_i + \lambda_i k_i \; + e\rho_i + \cdots + \lambda_i k_i + e\rho_i)Ge^{-1} - (\lambda_1 k_1 G + \lambda_2 k_2 G + \cdots + \lambda_m k_m G + zG)e^{-1} \\ &= (\rho' + \rho_1 + \rho_2 + \cdots + \rho_m)G \\ &= R' + R_1 + R_2 + \cdots + R_n = R \end{aligned}$$

and compute:

$$\begin{aligned} e^* &= F_H(M||x_{R^*}||x_U) \; mod \; \delta \\ &= F_H(M||x_R||x_U) \; mod \; \delta = e \end{aligned}$$

Thus, the correctness of the protocol has been proved.

3.3 Security Advantages of the Proposed Group Digital Signature Schemes

The group signature schemes proposed in Sections 3.1 and 3.2 have the following security advantages:

- According to formulas (16) and (17) or formulas (28) and (29): The masking coefficients λ_i and U are generated by the group manager. λ_i contains the group manager's private key. U contains the information of the public key of all members of the signing group who participated in the generation of the last group digital signature. Thus, only the group manager can know who participated in the signature creation because only he/she can open the signature. Since only he/she can calculate the masking coefficients λ_i and then calculate U.

- Eqs. (23) and (25) or Eqs. (35) and (36) show that the shared signature $\boldsymbol{s}_i$ of all members of the signing group is checked by the group manager. Only if all is valid, the manager conducts to create the third part $\boldsymbol{s}$ of the group signature which contains his/her shared signature. Thus, if the member's shared signature is forged, it is also difficult to pass the test expression (23) or (35).
- Another security advantage of the proposed schemes is the use of random parameter $\boldsymbol{\rho}$ in the process of generating shared signatures of each group member. $\boldsymbol{\rho}$ is treated as the second private key. This private key can only be used once, so the secrecy is very high. This also means that, if someone wants to forge a shared signature of a group member, to pass a checking expression (26) or (35), one must solve 2 discrete logarithmic problems to find $\boldsymbol{k}$ and $\boldsymbol{\rho}$ (see formula (23) or (35)).

4 The Proposed Collective Digital Signature Schemes

The new collective signature schemes proposed in this section are built based on a combination of the collective digital signature schemes described in Sections 2.1 and 2.2 and the group digital signature schemes described in Sections 3.1 and 3.2. These collective digital signatures are formed in 2 steps: The first step, form the pre-signature; The second step, form the collective digital signature for signing groups. In Section 4.1, we use the signature standard ECDSA to construct the signature for a number of signing groups. Meanwhile, in Section 4.2, we rely on the signature standard GOST R34.10-2001 to construct the signature for a number of signing groups and a number of individual signers.

4.1 Collective Digital Signature Scheme for a Number of Signing Groups Using the Standard ECDSA

Let g signing groups. There are m active members in each signing group. Let the public key of j-th signing group is $L_j = z_j G$, where z_j is private key of the j-th group manager ($j = 1, 2, \ldots, g$). These signing groups want to sign on the document M. A set of members in the j-th signing group is denoted by m_j. The collective digital signature scheme for several signing groups includes following procedures:

- *The signature generation procedure on the document M*

1. Do the same as the GDS protocol described in Sections 2.1 and 3.1, the group manager of each signing group calculates masking parameters λ_{ji} for the signers in his/her group. Then he/she calculates the values U_j and R_j ($i = 1, 2, \ldots, m_j$):

$$U_j = \sum_{i=1}^{m_j} \lambda_{ji} P_{ji} \qquad (40)$$

$$R_j = R_j' + \sum_{i=1}^{m_j} R_{ij} \qquad (41)$$

and then he/she sends values U_j and R_j to all other signing group managers.

2. Each j-th signing group manager ($j = 1, 2, \ldots, g$) calculates the values U, R and e:

$$U = \sum_{j=1}^{g} U_j \qquad (42)$$

$$R = \sum_{j=1}^{g} R_j \tag{43}$$

$$e = F_H(M \parallel x_R \parallel x_U)\ mod\ \delta \tag{44}$$

where δ is a large prime. The length of δ is $|\delta| = 160$ bits.

The first element of this novel collective signature is U and e is the second element.

3. Each manager of j-th signing group calculates a shared signature of his/her signing group:

$$s_j = s'_j + \sum_{j=1}^{m_j} s_{ji}\ mod\ q \tag{45}$$

where s_{ji} is the i-th signer's shared signature who is a member of the j-th signing group. Then he/she sends a value s_j to all other signing group managers.

4. Each manager of j-th signing group uses the checking Eq. (42) to verify the correctness of each share signature s_j:

$$R_j = s_jG - e(U_j + L_j)\ mod\ q \tag{46}$$

If all shared signatures s_j satisfy the verification Eq. (46), then he/she calculates s. s is the last element of this novel collective signature:

$$s = \sum_{j=1}^{g} s_j\ mod\ q \tag{47}$$

The tuple $(U,\ e,\ s)$ created by this procedure is the collective signature of g signing groups on the document M.

- *The signature verification procedure on the document M*

1. The verifier calculates the collective public key of all the signing collective L:

$$L = \sum_{j=1}^{g} L_j \tag{48}$$

(L_j is the group public key of the j-th signing group)

2. He/She calculates the value R^*:

$$R^* = sG - (U + L)e \tag{49}$$

3. He/She calculates the value e^*:

$$e^* = F_H(M \parallel x_{R^*} \parallel x_U) \tag{50}$$

4. Comparing e^* with e. If $e^* = e$ then the combined collective signature is valid, i.e., it was really generated by the given m signing groups. Otherwise, the signature is invalid and it is rejected.

It is easy to see that, the U contains the information of all the signers, in all the signing groups that have participated in creating collective signatures for the signing groups on document M. The identification of these signers is done similar to the procedures are presented in [9].

- *Proof of correctness of the CDS for several signing groups*

Substituting the values $s = s' + s_1 + s_2 + \cdots + s_m \ mod\ q$, $U = \lambda_1 P_1 + \lambda_2 P_2 + \cdots + \lambda_m P_m$ and $L = zG$ in the verification equation R^* (49):

$$\begin{aligned} R^* &= sG - (U + L)e \\ &= \sum_{j=1}^{g} s_j G - \left(\sum_{j=1}^{g} U_j + \sum_{j=1}^{g} L_j \right) e = \sum_{j=1}^{g} s_j G - \sum_{j=1}^{g} (U_j + L_j) e \\ &= \sum_{j=1}^{g} (s_j G - (U_j + L_j) e) \\ &= \sum_{j=1}^{g} R_j = R \end{aligned}$$

and compute:

$$\begin{aligned} e^* &= F_H(M \parallel x_{R^*} \parallel x_U) \ mod\ \delta \\ &= F_H(M \parallel x_R \parallel x_U) \ mod\ \delta \\ &= e \end{aligned}$$

The correctness of the protocol has been proved.

4.2 Collective Digital Signature Scheme for a Number of Signing Groups and a Number of Individual Signers Using the Standard GOST R34.10-2001

Within the framework of the CDS scheme for signing groups described in Section 4.1 several individual signers can be also involved, if it will be assumed that the individual signature (e, s) is considered as triple (O, e, s), where O if the elliptic curve point in infinity, i.e., each individual signer contributes the share O in the first part of the last collective signature.

Suppose L_j $(L_j = k_j G)$ and k_j, where $j = g + 1, g + 2, \ldots, g + m$, are public key and private 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 g signing groups and m individual signers includes the following procedures.

- *The signature generation procedure on the document M*

1. The manager of each j-th signing group $(j = 1, 2, \ldots, g)$ generates masking parameters λ_{ji} for signers in his/her signing group. Then he/she calculates the values U_j and R_j $(i = 1, 2, \ldots, m_j)$:

$$U_j = \sum_{i=1}^{m_j} \lambda_{ji} P_{ji} \tag{51}$$

$$R_j = R_j' + \sum_{i=1}^{m_j} R_{ji} \tag{52}$$

Then he/she sends the values U_j and R_j to all other group managers and individual signers. Each j-th individual signer ($j = g + 1, \ldots, g + m$) selects a random value $\rho_j < q$. The next he/she calculates the value R_j :

$$R_j = \rho_j G \tag{53}$$

where $j = g + 1, g + 2, \ldots, g + m$. After sents to all signers (group managers and individual signers).

2. Each j-th group manager ($j = 1, 2, \ldots, g$) and each individual signer ($j = g + 1, g + 2, \ldots, g + m$) computes values:

$$U = \sum_{j=1}^{g+m} U_j, \tag{54}$$

$$R = \sum_{j=1}^{g+m} R_j \tag{55}$$

and

$$e = F_H(M||x_R||x_U) \ mod \ \delta, \tag{56}$$

where δ is a large prime. The length of δ is $|\delta| = 160$ bits, $U_j = O$ for ($j = g + 1, g + 2, \ldots, g + m$);

The first element of this novel collective signature is U and e is the second element.

3. Each manager of j-th signing group calculates a shared signature of his/her signing group

$$s_j = s'_j + \sum_{i=1}^{m_j} s_{ji} \ mod \ q \tag{57}$$

where s_{ji} is the i-th signer's shared signature who is a member of the j-th signing group. The he/she sends s_j to other signing group managers. Each j-th individual signer ($j = g + 1, g + 2, \ldots, g + m$) calculates his/her shared signature:

$$s_j = \lambda_j k_j + e\rho_j \ mod \ q \tag{58}$$

4. Each manager of j-th signing group uses the checking Eqs. (59) and (60) to verify the correctness of each shared signature s_j:

$$R_j = s_j G e^{-1} - (U_j + L_j) e^{-1} \tag{59}$$

for $j = 1, 2, \ldots, g$ and

$$R_i = s_i G e^{-1} - \lambda_i e^{-1} P_i \tag{60}$$

for $j = g + 1, g + 2, \ldots, g + m$.

If all shared signatures s_j satisfy the verification Eqs. (59) or (60), then he/she calculates s. s is the last element of this novel collective signature:

$$s = \sum_{j=1}^{g+m} s_j \ mod \ q \tag{61}$$

The tuple $(U,\ e,\ s)$ created by this procedure is the collective signature of g signing groups and m individual signers on the document M.

- *The signature verification procedure on the document M*

1. The verifier calculates the collective public key of signing collective:

$$L = \sum_{j=1}^{g+m} L_j \tag{62}$$

(L_j is the group public key of the j-th signing group or of the j-th individual signer)

2. He/She calculates the value R^*:

$$R^* = sGe^{-1} - (U + L)e^{-1} \tag{63}$$

3. He/She calculates the value e^*:

$$e^* = F_H(M||x_{R^*}||x_U) \tag{64}$$

4. Comparing e with e^*. If $e^* = e$, then the combined collective signature is valid, i.e., it was really generated by the given m signing groups and g individual signers. Otherwise, the signature is invalid and it is rejected.

Same as in Section 4.1, the U contains the information of all the signers, in all the signing groups that have participated in creating collective signatures for the signing groups on document M.

- *Proof of correctness of the CDS for several signing groups and several individual signers*

Substituting the values:

$$s = \sum_{j=1}^{g+m} s_j\ mod\ q,\quad U = \sum_{j=1}^{g+m} U_j\ and\ L = \sum_{j=1}^{g+m} L_j$$

in the verification Eq. (63):

$$R^* = sGe^{-1} - (U + L)e^{-1}$$

We get:

$$\begin{aligned} R^* &= \left(\sum_{j=1}^{g}(s'_j + \sum_{i=1}^{m_j} s_{ji})\ \sum_{j=g+1}^{g+m} s_j\right) Ge^{-1} - \begin{pmatrix} \sum_{j=1}^{g}\sum_{i=1}^{m_j}\lambda_{ji}P_{ji} + \sum_{j=g+1}^{g+m}\lambda_j P_j \\ + \sum_{j=1}^{g}\sum_{i=1}^{m_j}\lambda_{ji}L_{ji} + \sum_{j=g+1}^{g+m}\lambda_j L_j \end{pmatrix} e^{-1} \\ &= \begin{pmatrix} \sum_{j=1}^{g}\left(z_j + \rho_j' e + \sum_{i=1}^{m_i}(\lambda_{ji}k_{ji} + e\rho_{ji})\right) \\ + \sum_{i=\varepsilon+1}^{g+m}(\lambda_j k_j + e\rho_j) \end{pmatrix} Ge^{-1} - \left(\sum_{j=1}^{g}\sum_{i=1}^{m_i}\lambda_{ji}P_{ji} + \sum_{j=g+1}^{g+m}\lambda_j P_j + \sum_{j=1}^{g}\sum_{i=1}^{m_i} L_{ji} + \sum_{j=g+1}^{g+m} L_j\right) e^{-1} \\ &= \left(\sum_{j=1}^{g}\left(\rho_j' + \sum_{i=1}^{m_j}\rho_{ji}\right) + \sum_{j=g+1}^{g+m}\rho_j\right) G = \sum_{j=1}^{g}\left(R_j' + \sum_{i=1}^{m_j} R_{ji}\right) + \sum_{j=g+1}^{g+m} R_j = R \end{aligned}$$

and compute:

$$\begin{aligned} e^{*} &= F_H(M||x_{R^*}||x_U) \ mod \ \delta \\ &= F_H(M||x_R||x_U) \ mod \ \delta = e \end{aligned}$$

The correctness of the protocol has been proved.

4.3 Performance Evaluation of the Proposed Collective Digital Signature Schemes

Remember, the individual signer identification procedure needs the participation of the group manager of the signing groups who participated in the formation of the final collective signature. Thus, the computational complexity of this procedure is quite high. This will increase significantly as the number of signing groups, the number of individual signers increases.

We consider the performance of proposed schemes by comparing the time for the signature generation proceduce and the time for signature verification procedure of our schemes and the scheme in [12]. Both are collective signature schemes for signing groups, both are built on the difficult problem of the discrete logarithm problem, but the scheme in [12] is in the finite field (in Section 2), the scheme in this paper is in the EC combined with ECDSA standard (in Section 4.1).

Notations: T_h : Time cost of a hash operation in Z_p; T_s : Time cost of a scalar multiplication 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 [13]: $T_h \approx T_m$, $T_s \approx 29T_m$, $T_{inv} \approx 240T_m$, $T_e \approx 240T_m$.

Suppose there are g groups of signings and each j_{th} group has m_j signers. The time cost for generating signature components of the proposed digital signature scheme and the collective digital signature scheme in [12] is given in Tab. 1.

Table 1: Time cost of the signature scheme in [12] and the proposed signature scheme

	The CDS based on GF(P) [12]	The CDS based on ECDSA
Time for signature generation	$U = \sum_{j=1}^{g}(243m_j)T_m$ $E = \sum_{j=1}^{g}(241m_j + 241)T_m$ $S = \sum_{j=1}^{g}(484m_j + 1)T_m$ $Sum = \sum_{j=1}^{g}(968m_j + 242)T_m$	$U = \sum_{j=1}^{g}(31m_j)T_m$ $e = \sum_{j=1}^{g}(29m_j + 29)T_m$ $s = \sum_{j=1}^{g}(61m_j + 1)T_m$ $Sum = \sum_{j=1}^{g}(121m_j + 30)T_m$
Time for signature verification	$(483 + g)T_m$	$(58 + g)T_m$

Tab. 1 shows that the time cost for the generation of signature components and for the signature verification of the proposed collective signature scheme is much lower than that of the similar signature scheme in [12].This disparity will be large as the number of members that participate in signature formation increases when comparing the time cost for constructing the CDSs for several signing groups and several individual signers of the schemes in [12] (at Section 3) and the scheme in this paper (at Section 4.2).

5 Discussion

The protocols described in Section 4 represent a new extension of the collective signature scheme [9] proposed for generating a common signature that is shared by an arbitrary group of individual signers and has a fixed size. The uniqueness consists in many signing groups or several signing groups and several individual signers sharing a fixed-size single signature. That combination between the collective signature schemes [9] and the group signature schemes [6,14] is possible because the last schemes use the formation of the collective signature as a preliminary stage of the group signature formation. Analogous combining of the collective signature scheme with the group signature protocols of other types, for example, described in [15–17] appears to be impossible.

In the proposed protocols like the protocols in [10], the private keys used by each signer participating in the process of computing the collective signature is known only for the owner of the respective public key and the security justification of the security can be performed as reducing to the security of the individual signature protocol or to the group signature protocol, directly in line with security analysis provided by [9].

An important advantage of the proposed protocols relates to the possibility of using the existing public key infrastructure for their practical use. Besides, some official signature standards, for example [18–20], can be used as their base cryptoscheme.

6 Conclusion

Thus we have studied and proposed different types of signature schemes using the elliptic curves: i) the collective digital signature scheme for several signing groups (using ECDSA); ii) the collective digital signature scheme for a number of signing groups and a number of individual signers (using GOST R34.10-2001). For each one, we designed the signature generation procedures, the signature verification procedures, prove their correctness. The security advantages and the performance of the novel collective digital schemes were also presented.

With the novelty that the group of authors presented in this article hopes to have practical applications in practice. In the future, we will study and develop collective signature schemes as proposed in this paper, but it only consists of 2 components E and S, like the signatures in [1–4,6,15] by using the EC.

The computational difficulty of the described collective signature protocols is about m times higher than the difficulty of the group signature protocols [6,12]. However the shared signatures of the signing groups can be computed simultaneously, therefore in the case of the parallel computation, their performance is comparable with performance with their base group signature scheme. In the future, we will also study to construct new collective signature protocols under this approach.

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

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

References

[1] S. Radack, *Updated Digital Signature Standard Approved as Federal Information Processing Standard (FIPS) 186-3*, National Institute of Standards and Technology, FIPS Publication, pp. 186–193, 2009.

[2] D. Chaum, “Blind signatures for untraceable payments,” in *Proc. Advances in Cryptology–CRYPTO’82*, Plenum Press, pp. 199–203, 1983.

[3] J. L. Camenisch, J. M. Piveteau and M. A. Stadler, “Blind signatures based on the discrete logarithm problem,” in *Proc. Advances in Cryptology–EUROCRYPT’94, Lecture Notes in Computer Science*, vol. 950, Berlin, Heidelberg, New York: Springer-Verlag, pp. 428–432, 1995.

[4] Q. Alamélou, O. Blazy, S. Cauchie and Ph. Gaborit, “A code-based group signature scheme,” *Designs, Codes and Cryptography*, vol. 82, no. 1–2, 2017.

[5] R. Xie, C. Xu, C. He and X. Zhang, “A new group signature scheme for dynamic membership,” *International Journal of Electronic Security and Digital Forensics*, vol. 8, no. 4, 2016.

[6] 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.

[7] N. Koblitz, “Elliptic curve cryptosystems,” *Mathematics of Computation*, vol. 48, pp. 203–209, 1987.

[8] D. Johnson, A. J. Menezes and S. Vanstone, *The Elliptic Curve Digital Signature Algorithm (ECDSA)*, Certicom, 2001.

[9] V. Dolmatov, A. Chuprina and I. Ustinov, “Use of GOST signature algorithms in DNSKEY and RRSIG resource records for DNSSEC,” *RFC5933*, pp. 1–8, 2010. [Online]. Available: https://netzikon.net/rfc/RFC5933-Use-of-GOST-Signature-Algorithms-in-DNSKEY-and-RRSIG-Resource-Records-for-DNSSEC.html/.

[10] N. A. Moldovyan and A. A. Moldovyan, “Blind collective signature protocol based on discrete logarithm problem,” *International Journal of Network Security*, vol. 11, no. 2, pp. 106–113, 2010.

[11] S. Vanstone, D. Hankerson and A. J. Menezes, “Guide to elliptic curve cryptography,” in *Springer*, Chap. 4.4.1, 2004.

[12] N. K. Tuan, V. L. Van, D. N. Moldovyan, H. N. Duy and A. A. Moldovyan, “Collective signature protocols for signing groups,” in *Proc. Information Systems Design and Intelligent Applications. Advances in Intelligent Systems and Computing*, India, 2018.

[13] C. Popescu, “Blind signature and BMS using elliptic curves,” in *Studia Univ Babes–Bolyai*, Informatica, pp. 43–49, 1999.

[14] N. A. Moldovyan, N. H. Minh, D. T. Hung and T. X. Kien, “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.

[15] A. A. Bolotov, S. B. Gashkov and A. B. Frolov, “Elementary introduction to elliptic curve cryptography,” in *cryptography Protocols on the Elliptic Curves*, KomKniga, Moskow, 2006.

[16] A. Komarova, A. Menshchikov and T. Klyaus, “Analysis and comparison of electronic digital signature state standards GOST R34.10-1994, GOST R34.10-2001 and GOST R34.10-2012,” in *Proc. the 10th Int. Conf.*, Jaipur, India, 2017.

[17] J. Pieprzyk, T. Hardjono and J. Seberry, *Fundamentals of Computer Security*, Berlin: Springer-Verlag, 2003.

[18] J. Lee, H. Kim, Y. Lee, S. M. Hong and H. Yoon, “Parallelized scalar multiplication on elliptic curves defined over optimal extension field,” *International Journal of Network Security*, vol. 4, pp. 99–106, 1 2017.

[19] A. Beresneva, A. Epishkina, O. Isupova, K. Kogos and M. Shimkiv, “Special digital signature schemes based on GOST R 34.10-2012,” in *Proc: Electrical and Electronic Engineering Conf. (EIConRusNW)*, IEEE NW Russia Young Researchers, 2016.

[20] V. Dolmatov and A. Degtyarev, “GOST R34.10-2012: Digital signature algorithm,” *RFC 7091*, pp. 1–20, 2013. [Online]. Available: https://datatracker.ietf.org/doc/html/rfc7091/.