Computers, Materials & Continua
DOI: 10.32604/cmc.2023.029253
Article

Tech Science Press

Constructing Representative Collective Signature Protocols Using The GOST R34.10-1994 Standard

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

[1]School of Computer Science, Duy Tan University, Danang, Vietnam
[2]Department of Information Technology, Hanoi, Vietnam
[3]ITMO University, St. Petersburg, Russia
*Corresponding Author: Tuan Nguyen Kim. Email: nguyenkimtuan@duytan.edu.vn
Received: 28 February 2022; Accepted: 06 April 2022

Abstract: The representative collective digital signature, which was suggested by us, is built based on combining the advantages of group digital signature and collective digital signature. This collective digital signature schema helps to create a unique digital signature that deputizes a collective of people representing different groups of signers and may also include personal signers. The advantage of the proposed collective signature is that it can be built based on most of the well-known difficult problems such as the factor analysis, the discrete logarithm and finding modulo roots of large prime numbers and the current digital signature standards of the United States and Russian Federation. In this paper, we use the discrete logarithmic problem on prime finite fields, which has been implemented in the GOST R34.10-1994 digital signature standard, to build the proposed collective signature protocols. These protocols help to create collective signatures: Guaranteed internal integrity and fixed size, independent of the number of members involved in forming the signature. The signature built in this study, consisting of 3 components (U, R, S), stores the information of all relevant signers in the U components, thus tracking the signer and against the "disclaim of liability" of the signer later is possible. The idea of hiding the signer's public key is also applied in the proposed protocols. This makes it easy for the signing group representative to specify which members are authorized to participate in the signature creation process.

Keywords: Signing collective; signing group; discrete logarithm; group signature; collective signature; GOST standards

1 Introduction

As is known, digital signatures (DS) [1] are a key component of digital authentication systems [2]. Therefore, in order to have an authentication system that meets the specific requirements of a certain practical application, we must first build a corresponding digital signature. More precisely, it is necessary to build a DS scheme (DSS) that helps to create the desired digital signature.

Currently, many types of digital signatures have been researched and published such as single digital signatures (SDS), group digital signatures (GDS) [3–6], collective digital signatures (CDS) [7–9], single-blind digital signatures [10,11], blind collective digital signatures (BCS) [12], representative collective digital signatures (RCS) [9], ... The SDSs is used to authenticate a particular personal. Meanwhile, GDSs, CDSs and RCSs are used to authenticate a signing group of many members [13,14]. The difference between a RCS and a GDS and a CDS is that it is formed from group digital signatures of many different signing groups. These signing groups may have no members at all, in which case the group leader can be viewed as personal signers. The RCS fulfils the authentication requirement for the multi-functional collectives. It is a new form of authentication, which is now quite common in e-commerce applications operating in the environment of Internet.

The DSS is usually built based on one of three common difficult problems such as factor analysis problem, discrete logarithmic problem on finite prime fields [15,16], discrete logarithmic problem on Elliptic curve [17,18]. At present, there are also some newly proposed difficult problems such as the problem of finding the modulo roots of large primes [19], the discrete logarithmic problem on the ring of residual classes, the discrete logarithmic problem on the two-dimensional non-circular subgroup.

The above mentioned difficult problems are the basis for a number of standard single-digital signature schemes, which are RSA (Rivest–Shamir–Adleman) [20] signature scheme, Schnorr's signature scheme and ElGamal signature scheme. Some countries such as United States, Russian Federation or Belarus also rely on existing difficult problems to build their own digital signature standards or digital signature scheme standards. United States's digital signature scheme standards include DSS (Digital Signature Standard) and DSA (Digital Signature Algorithm). Similarly, Russian Federation has digital signature scheme standards such as GOST R34.10-1994 [21], GOST R34.10-2001 [21] and GOST R34.10-2012 [22]. These signature scheme standards not only inherit the level of security of the difficult problem it uses, but also have enhanced security features brought about by the scheme's own internals.

In this paper, we use the GOST R34.10-1994 standard (GOST-1994) [21], developed based on the logarithmic problem on prime finite field, to suggest two types of RCS scheme: (i) the CDS is shared by many signing groups and (ii) the CDS is shared by many signing groups and many personal signers. First, we use the single signature scheme built by this standard to build a CDS for a signing collective of m members. This scheme is resistant to two types of attacks, originating from the inside, which are quite common, that is, attack on the secret key and attack forgery scraping the signature of a member of the signing collective; Next, we build a GDS for a group of m signing members and a leader. The RSA asymmetric key pair generation algorithm, the signer's public key masking technique, and the mutual authentication mechanism are implemented in this scheme. These are considered the security advantages of our proposed group signature scheme; Finally, two new forms of RCS schemas are formed from the two base schemas CDS and GDS. All DSS built in this article are proven correct.

The security advantages and time costs of the proposed RCS schemes are analyzed, calculated and shown at the end of this paper.

2 The Related Basis Digital Signature Schemas

The following, the GOST-1994 of the Russian Federation will be used to develop two basic schemas: The CDS schema (in 2.1) and the GDS schema (in 2.2). In the GDS schema, the public key of the signing group member is hidden by masking techniques. These schemas will be used to build RCS schemas.

2.1 The Collective Digital Signature Schema According to The GOST R34.10-1994 Digital Signature Standard (The CDS-2 Schema)

The common parameters are similar to those defined for GOST-1994 [21].

Assuming there are m signers who want to create a CDS on the message M (M). Private key and public key of *i-th* signer be x_i and $y_i = \alpha^{x_i} \ mod \ p$, where x_i is an integer chosen at random by the signer in the range $[1, q-1]$; Let F_H be a secure hash function and $H = F_H(M)$.

• **The algorithm to create CDS on M:**

Includes the following stages:

1. Each signer selects random number $k_i \in [1, q-1]$ and computes:

$$R_i = (\alpha^{k_i} mod \ p) mod \ q \tag{1}$$

and sends (broadcasts) R_i to all signers.

2. It is calculated the first element of the CDS:

$$R = R_1 R_2 \ldots R_m \ mod \ q \tag{2}$$

The values R is broadcasted to the other signers.

3. Each signer computes its signature share S_i using the hash function value H corresponding to the message M and the value R as follows:

$$S_i = k_i H + x_i R \ mod \ q \tag{3}$$

4. Calculate the second element of the CDS:

$$S = S_1 + S_2 + \ldots + S_m \ mod \ q \tag{4}$$

The (R, S) number pair is the CDS on M.

• **The algorithm to check CDS on M:**

The signature verifier does the following:

Input: $\alpha, p, q, M, (R, S), K_P = \{y_i | i = 1, 2, .., m\}$.

Output: *True*/*False*

[1]. **If** $(R = 0$ **or** $S = 0)$ **then**

Return *False*

[2]. $y \leftarrow 1$, **For** $i = 1$ **to** m **do**

[2.1]. $y \leftarrow y \times y_i mod \ p$

[3]. $H \leftarrow F_H(M)$

[4]. $R' \leftarrow (\alpha^{S/H} y^{-R/H} mod \ p) mod \ q$

[5]. **If** $(R' = R)$ **then**

Return *True*

Else

Return *False*

If $R' = R$ (True): The signature on M is valid; Otherwise, the signature on M is invalid, it is rejected.

• **Proof of the correctness of the CDS-2 scheme:**

To prove the correctness of this scheme, we only need to prove the existence of the check expression $R^* = R$ in the signature check procedure. Conspicuously, the test expression $R^* = R$ always exists. Indeed:

Substituting the value $S = \sum_{i=1}^{m} S_i \ mod \ q$ into the expression R^*, $R^* = (\alpha^{S/H} y^{-R/H} mod \ p) \ mod \ q$, we get:

$$
\begin{aligned}
R^* &= \left(\alpha^{\sum_{i=1}^{m} \frac{S_i}{H}} \left(\prod_{i=1}^{m} y_i\right)^{-\frac{R}{H}} mod \ p\right) mod \ q \\
&= \left(\prod_{i=1}^{m} \alpha^{\frac{S_i}{H}} \prod_{i=1}^{m} y_i^{-\frac{R}{H}} mod \ p\right) mod \ q \\
&= \left(\prod_{i=1}^{m} \left(\alpha^{\frac{S_i}{H}} \alpha^{\frac{-x_i R}{H}} mod \ p\right)\right) mod \ q \\
&= \left(\prod_{i=1}^{m} \left(\alpha^{\frac{(k_i H + x_i R)}{H}} \alpha^{\frac{-x_i R}{H}} mod \ p\right)\right) mod \ q \\
&= \left(\prod_{i=1}^{m} \alpha^{k_i} mod \ p\right) mod \ q \\
&= \left(\prod_{i=1}^{m} R_i\right) mod \ q = R
\end{aligned}
$$

So $R^* = R$ always exists.

2.2 The Group Digital Signature Schema According to The GOST R34.10-1994 Digital Signature Standard (The GDS-2 Schema)

Assuming there is a signing group that wants to create a GDS on the message M. Private key and public key of *i-th* signer be x_i and $y_i = \alpha^{x_i} \ mod \ p$, where $1 \leq i \leq m$; x_i is an integer chosen at random by the signer in the range $[1, q-1]$; Let F_H be a secure hash function and $H = F_H(M)$.

Private key and public key of the group manager be X and $Y = \alpha^X \ mod \ p$. Y is also the public key of the signing group so it is also used in the GDS-2 scheme.

In this schema, the group manager's internal public key is used, which is a number pair (n, e) and is generated as follows: (1) $n = wr$, where r and w are strong primes; and (2) $\varphi(n) = (w-1)(r-1)$; (3) chooses a integer number e that $e \in (1, \phi(n))$ and $gcd\,(e, \phi\,(n)) = 1$; (4) $d = e^{-1} \ mod \ \varphi(n)$, d is a private value of manager. The (n, e) is only for the signers of the group headed by the manager.

The generalized scheme of the proposed group signature protocol includes the following steps [6,7]:

i) Considering the message M to be signed the group manager masks the public keys of the assigned signers. To mask the public key y_i of a signer: The group manager computes the exponent $\lambda_i = (H + y_i)^d \ mod \ n$, H is the hash value of M, and sends λ_i to the *i-th* member.

ii) Using λ_i each i-th member calculates their shared value in the collective signature and sends it to the group manager.

iii) The group manager verifies the shared value of all assigned members and calculates his shared value in the group signature. Then he/she calculates the group signature as triple (U, R, S), where S is sum of all shares; U is the product of the modified public keys of all signers.

• **The group signature generation algorithm on M**

It consists of stages:

1. The group manager does the following tasks:

- Computes hash value from message M:

$$H = F_H(M) \tag{5}$$

- Calculates masking coefficients λ_i:

$$\lambda_i = (H + y_i)^d \, mod \, n \tag{6}$$

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

- Computes the first element of the group signature U:

$$U = \prod_{i=1}^{m} y_i^{\lambda_i} mod \, p \tag{7}$$

2. The *i-th* signer does as follow:

- Randomly choose a number k_i, $k_i < q$, and then computes the value R_i:

$$R_i = \left(\alpha^{k_i} \, mod \, p\right) mod \, q \tag{8}$$

- Sends R_i to the signing group's manager (GM)

3. GM does as follow:

- Randomly choose a number K, $K < q$, and calculates R', R:

$$R' = \left(\alpha^K mod \, p\right) mod \, q \tag{9}$$

$$R = R'^{\prod_{i=1}^{m} R_i} mod \, q = \left(\alpha^{K+\sum_{i=1}^{m} k_i} mod \, p\right) mod \, q \tag{10}$$

4. The *i-th* signer does as follow:

- Calculates his/her shared value, S_i:

$$S_i = k_i H + x_i \lambda_i R \, mod \, q \tag{11}$$

- Sends S_i to GM

5. GM does as follow:

- Verifies the correctness of each S_i by checking R_i^*:

$$R_i^* = (\alpha^{S_i/H} y^{-R\lambda_i/H} mod \, p) mod \, q \tag{12}$$

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

$$S' = KH + XR \, mod \, q \tag{13}$$

- Calculates the third element of GDS, S:

$$S = S' + \sum_{i=1}^{m} S_i \ mod \ q \tag{14}$$

The tuple (U, R, S) is a GDS of the signing group on M.

• **The signature verification algorithm on M**

Done by signature verifiers:

Input: U, $Y, \alpha, p, q, M, (U, R, S)$

Output: *True*/*False*

[1]. **If** $(U = 0$ **or** $R = 0$ **or** $S = 0)$ **then**

Return *False*

[3]. $H \leftarrow F_H(M)$

[4]. $R' \leftarrow (\alpha^{S/H}(UY)^{-R/H} mod \ p) mod \ q$

[5]. **If** $(R' = R)$ **then**

Return *True*

Else

Return *False*

If $R^* = R$ (True): GDS on M is valid; Otherwise, GDS on M is invalid.

• **Proof of correctness of the GDS-2 schema**:

If $R^* = R$ exists, then the correctness of this schema is guaranteed.

Conspicuous:

$$\begin{aligned}
R^* &= \left(\alpha^{\frac{S}{H}} (UY)^{-\frac{R}{H}} \ mod \ p\right) mod \ q \\
&= \left(\alpha^{\frac{S'+\sum_{i=1}^{m} S_i}{H}} \left(Y \prod_{i=1}^{m} y_i^{\lambda_i}\right)^{-\frac{R}{H}} \ mod \ p\right) mod \ q \\
&= \left(\alpha^{\frac{S'}{H}} \prod_{i=1}^{m} \alpha^{\frac{S_i}{H}} \ Y^{-\frac{R}{H}} \prod_{i=1}^{m} y_i^{-\frac{R\lambda_i}{H}} \ mod \ p\right) mod \ q \\
&= \left(\alpha^{\frac{KH+XR}{H}} \alpha^{-\frac{XR}{H}} \prod_{i=1}^{m} \left(\alpha^{\frac{(k_i H + x_i \lambda_i R)}{H}} \alpha^{\frac{-x_i \lambda_i R}{H}}\right) mod \ p\right) mod \ q \\
&= \left(\alpha^{K} \prod_{i=1}^{m} \alpha^{k_i} mod \ p\right) mod \ q \\
&= \left(R' \prod_{i=1}^{m} R_i\right) mod \ q = R
\end{aligned}$$

Since $R^* = R$ always exist.

3 Constructing The New CDS Protocol Using The GOST R34.10-1994 Digital Signature Standard

The following, we use the CDS schema and the GDS schema built above to construct two types of RCS schemas [9]:

i) The CDS schema (type 1): This scheme helps to create a DS that deputize for a signing collective whose members are representatives, they are group leaders, for different signing groups.
ii) The CDS schema (type 2): This scheme helps to create a DS that deputize for a signing collective whose membership consists of two groups of members: That is, people who represent different signing groups. And, personal signers. These people do not belong to any sign group, they are also considered group representatives, but their group has no members.

3.1 The CDS Schema For Signing Groups (RCS.01)

Let g signing groups with public keys $Y_j = \alpha^{X_j} \, mod \, p$, where $j = 1, 2, \ldots, g$. X_j is the secret key of the *j-th* group manager, have intention to sign the message M.

Suppose also the *j-th* signing group includes m_j active personal signers (persons appointed to act on behalf of the *j-th* signing group).

The CDS scheme for signing groups (RCS.01) is as below.

•The CDS generation procedure on M

It include of steps:

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

- Based on the group signature generation procedure described above (Section 2.2) to generals masking parameters λ_{ji} for the signers of *j-th group*.

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

$$U_j = \prod_{i=1}^{m_j} y_{ji}^{\lambda_{ji}} mod \; p \tag{15}$$

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

- Computes R_j:

$$R_j = R_j' \prod_{i=1}^{m_j} R_{ji} \, mod \, q \tag{16}$$

- 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 and R:

$$U = \prod_{j=1}^{g} U_j \, mod \, p \tag{17}$$

$$R = \prod_{j=1}^{g} R_j \, mod \, q = \left(\alpha^{\sum_{j=1}^{g} k_j} mod \, p \right) mod \, q \tag{18}$$

U and R are the first and second elements of the CDS.

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

- Computes the shared signature of *j-th* group:

$$S_j = S'_j + \sum_{i=1}^{m_j} S_{ji} \; mod \; q \tag{19}$$

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:

$$R_j^* = \left(\alpha^{\frac{S_j}{H}} (U_j Y_j)^{-\frac{R}{H}} mod \, p\right) mod \, q \tag{20}$$

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

$$S = \sum_{j=1}^{g} S_j \, mod \, q \tag{21}$$

The tuple (U, R, S) is the CDS on M of the signing collective there are g signing groups.

• **The CDS verification algorithm on M**

Input: $\alpha, p, q, M, (U, R, S), \{Y_i | i = 1, 2, .., g\}$.

Output: *True*/*False*

[1]. **If** $(U = 0$ **or** $R = 0$ **or** $S = 0)$ **then**

Return *False*

[2]. $Y \leftarrow 1$, **For** $i = 1$ **to** g **do**

[2.1]. $Y_{col} \leftarrow Y \times Y_i mod \, p$

[3]. $H \leftarrow F_H(M)$

[4]. $R^* \leftarrow \left(\alpha^{\frac{S}{H}} (U Y_{col})^{-\frac{R}{H}} mod \, p\right) mod \, q$

[5]. **If** $(R^* = R)$ **then**

Return *True*

Else

Return *False*

If $R^* = R$ (True): The CDS is authenticated. Otherwise, the CDS is not authenticated.

• **Demonstrate the RCS.01 schema correctness:**

If we prove the existence of calculation formulas S_{ji} and R^* then the correctness of the RCS scheme described above is guaranteed. It's easy to see it always exists. Indeed:

$$R_j^* = \left(\alpha^{\frac{S_j}{H}} \left(U_j Y_j\right)^{-\frac{R}{H}} mod \, p\right) mod \, q$$

$$= \left(\alpha^{\frac{S'_j + \sum_{i=1}^{m_j} S_{ji}}{H}} \left(Y_j \prod_{i=1}^{m_j} y_{ji}^{\lambda_{ji}}\right)^{-\frac{R}{H}} mod \, p\right) mod \, q$$

$$= \left(\alpha^{\frac{S'_j}{H}} \prod_{i=1}^{m_j} \alpha^{\frac{S_{ji}}{H}} \; Y_j^{-\frac{R}{H}} \prod_{i=1}^{m_j} y_{ji}^{-\frac{R\lambda_{ji}}{H}} \; mod\; p\right) mod\; q$$

$$= \left(\alpha^{\frac{K_j H + X_j R}{H}} \alpha^{-\frac{X_j R}{H}} \prod_{i=1}^{m_j} \left(\alpha^{\frac{k_{ji}H + x_{ji}\lambda_{ji}R}{H}} \alpha^{\frac{-x_{ji}\lambda_{ji}R}{H}}\right) mod\; p\right) mod\; q$$

$$= \left(\alpha^{K_j} \prod_{i=1}^{m_j} \alpha^{k_{ji}} mod\; p\right) mod\; q$$

$$= \left(R'_j \prod_{i=1}^{m_j} R_{ji}\right) mod\; q = R_j$$

We see $R^* = R$ always exists. Indeed:

$$R^* = \left(\alpha^{\frac{S}{H}} \left(U Y_{col}\right)^{-\frac{R}{H}} mod\; p\right) mod\; q$$

$$= \left(\alpha^{\frac{\sum_{j=1}^{g} S_j}{H}} \left(\prod_{j=1}^{g} Y_j \prod_{j=1}^{g} U_j\right)^{-\frac{R}{H}} mod\; p\right) mod\; q$$

$$= \left(\prod_{j=1}^{g} \left(\alpha^{\frac{S_j}{H}} \left(Y_j U_j\right)^{-\frac{R}{H}}\right) mod\; p\right) mod\; q$$

$$= \prod_{j=1}^{g} R_j \, mod\; q = R$$

3.2 The CDS Schema For Signing Groups And Personal Signers (RCS.02)

The CDS schema in 3.2 is similar to RCS.01, but for signing groups and personal signers. In this case, the personal signer is treated as a signing group, but this group has no members, only the group manager. So $U_j = 1$.

Suppose x_j, $y_j = \alpha^{x_j}$, $j = g + 1, g + 2, \ldots, g + m$, are private key and public key of m personal signers participating in the protocol for generating the CDS for g signing groups and m personal signers.

Input parameters, public keys, and private keys are as in the CDS-02 diagram and the GDS-02 diagram. The CDS scheme for m personal signers and g signing groups (RCS.02) is as follows:

• **The procedure for generating the CDS for** g **signing groups and** m **personal signers on 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 2.2) to generals masking parameters λ_{ji} for the signers of *j*-th *group*.

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

$$U_j = \prod_{i=1}^{m_j} y_{ji}^{\lambda_{ji}} \, mod\, p \tag{22}$$

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

- Computes R_j:

$$R_j = R'_j \prod_{i=1}^{m_j} R_{ji} \, mod\, q \tag{23}$$

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

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

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

$$R_j = \left(\alpha^{K_j} \bmod p\right) \bmod q \tag{24}$$

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

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

$$U = \prod_{j=1}^{g+m} U_j \bmod p \tag{25}$$

$$R = \prod_{j=1}^{g+m} R_j \bmod q \tag{26}$$

where δ is a large prime having, $|\delta| = 160$ bits; $U = 1$ for $j = g+1, g+2, \ldots, g+m$.

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:

$$S_j = S_j^{'} + \sum_{i=1}^{m_j} S_{ji} \bmod q \tag{27}$$

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

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

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

$$S_j = K_j H + X_j R \bmod q \tag{28}$$

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

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

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

$$R_j^* = \left(\alpha^{\frac{S_j}{H}} (U_j Y_j)^{-\frac{R}{H}} \bmod p\right) \bmod q \tag{29}$$

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

$$R_j^* = \left(\alpha^{\frac{S_j}{H}} Y_j^{-\frac{R}{H}} \bmod p\right) \bmod q \tag{30}$$

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

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

$$S = \sum_{j=1}^{g+m} S_j \bmod q \tag{31}$$

The tuple (U, R, S) is the CDS on the message M of the signing collective there are g signing groups and m personal signers.

• **The algorithm for verification the CDS for *g* signing groups and *m* personal signers on M**

To check the validity of the received signature, the verifier performs the following steps:

Input: $\alpha, p, q, M, (U, R, S), \{Y_i | i = 1, 2, .., g + m\}$.

Output: *True*/*False*

[1]. **If** $(U = 0$ **or** $R = 0$ **or** $S = 0)$ **then**

Return *False*

[2]. $Y \leftarrow 1$, **For** $i = 1$ **to** $g + m$ **do**

[2.1]. $Y_{col} \leftarrow Y \times Y_i \, mod \, p$

[3]. $H \leftarrow F_H(M)$

[4]. $R^* \leftarrow \left(\alpha^{\frac{S}{H}} (U Y_{col})^{-\frac{R}{H}} mod \, p\right) mod \, q$

[5]. **If** $(R^* = R)$ **then**

Return *True*

Else

Return *False*

If $R^* = R$ (True): The CDS is authenticated; Otherwise, the CDS is not authenticated.

• **Demonstrate the RCS.02 schema correctness**:

If we prove the existence of calculation formulas S_j of each signing group R_j^* (35), S_j by each personal signer R_j^* (36) and expression R^* then the correctness of the RCS scheme described above is guaranteed.

i) Conspicuously, the formula S_{ji} always exists. Indeed:

$$
\begin{aligned}
R_j^* &= \left(\alpha^{\frac{S_j}{H}} \left(U_j Y_j\right)^{-\frac{R}{H}} mod \, p\right) mod \, q \\
&= \left(\alpha^{\frac{S_j' + \sum_{i=1}^{m_j} S_{ji}}{H}} \left(Y_j \prod_{i=1}^{m_j} y_{ji}^{\lambda_{ji}}\right)^{-\frac{R}{H}} mod \, p\right) mod \, q \\
&= \left(\alpha^{\frac{S_j'}{H}} \prod_{i=1}^{m_j} \alpha^{\frac{S_{ji}}{H}} \, Y_j^{-\frac{R}{H}} \prod_{i=1}^{m_j} y_{ji}^{-\frac{R\lambda_{ji}}{H}} mod \, p\right) mod \, q \\
&= \left(\alpha^{\frac{K_j H + X_j R}{H}} \alpha^{-\frac{X_j R}{H}} \prod_{i=1}^{m_j} \left(\alpha^{\frac{k_{ji} H + x_{ji} \lambda_{ji} R}{H}} \alpha^{\frac{-x_{ji} \lambda_{ji} R}{H}}\right) mod \, p\right) mod \, q \\
&= \left(\alpha^{K_j} \prod_{i=1}^{m_j} \alpha^{k_{ji}} mod \, p\right) mod \, q = \left(R_j' \prod_{i=1}^{m_j} R_{ji}\right) mod \, q = R_j
\end{aligned}
$$

ii) Conspicuously, the formula for checking the shared signature S_i shared by the R_j personal signer always exists.

Indeed:

$$\begin{aligned} R_j^* &= \left(\alpha^{\frac{S_j}{H}} Y_j^{-\frac{R}{H}} mod\ p\right) mod\ q \\ &= \left(\alpha^{\frac{K_j H + X_j R}{H}} \alpha^{-\frac{X_j R}{H}} mod\ p\right) mod\ q \\ &= \left(\alpha^{K_j} mod\ p\right) mod\ q = R_j \end{aligned}$$

iii) Same as above, $R^* = R$ always exists.

Indeed:

$$\begin{aligned} R^* &= \left(\alpha^{\frac{S}{H}} (UY_{col})^{-\frac{R}{H}} mod\ p\right) mod\ q \\ &= \left(\alpha^{\frac{\sum_{j=1}^{g+m} S_j}{H}} \prod_{j=1}^{g+m} \left(U_j Y_j\right)^{-\frac{R}{H}} mod\ p\right) mod\ q \\ &= \left(\prod_{j=1}^{g+m} \alpha^{\frac{S_j}{H}} \left(U_j Y_j\right)^{-\frac{R}{H}} mod\ p\right) mod\ q \\ &= \left(\left(\prod_{j=1}^{g} \alpha^{\frac{S_j}{H}} \left(U_j Y_j\right)^{-\frac{R}{H}} \prod_{j=g+1}^{g+m} \alpha^{\frac{S_j}{H}} Y_j^{-\frac{R}{H}}\right) mod\ p\right) mod\ q \\ &= \prod_{j=1}^{g} R_j \prod_{j=g+1}^{g+m} R_j\ mod\ q = R \end{aligned}$$

Since $R^* = R$, the correctness of the signature scheme has been proved.

4 Security Analysis And Performance Evaluation

4.1 Resistance to Attack of The New CDS Schemas

CDS schemes can be attacked by people who are not members of the signing collective (known as external attacks) or by people who are members of the signing collective (known as internal attacks), it happens more often. In this section we will analyze the two most common types of internal attacks. Specifically, the CDS-02 scheme is resistant to the following two types of attacks:

• **The first attack (the member signature forgery)**: Assuming that $n-1$ signers attempt to figure out a CDS respectively to n signers owning the public key $y = y' y_n\ mod\ p$, where $y' = \prod_{i=1}^{n-1} y_i\ mod\ p$, i.e., $n-1$ users unite their efforts to figure out an (R, S) number pair such that $R = (\alpha^{S/H} y' y_n^{-R/H} mod\ p)\ mod\ q$.

Assuming that they are able to do this. Thus, under our assumption the collective forger (i.e., the considered $n-1$ users) is able to figure out the CDS (R^*, S^*) respectively to public key $y = y' y_n'\ mod\ p$, where y_n' is a supposedly public key having the value $y_n' = y_n / y'\ mod\ p$.

The CDS satisfies the following relation:

$$\begin{aligned} R^* &= \left(\alpha^{\frac{S^*}{H}} y^{\frac{-R^*}{H}} mod\ p\right) mod\ q \\ &= \left(\alpha^{\frac{S^*}{H}} \left(y' y_n'\right)^{\frac{-R^*}{H}} mod\ p\right) mod\ q \end{aligned}$$

$$\Rightarrow R^* = \left(\alpha^{\frac{S^*}{H}} y_n^{\frac{-R^*}{H}} mod\ p\right) mod\ q$$

The last means that (R^*, S^*) is a personal DS of the n^{th} user. Thus, we have formally proved that possibility to forge the CDS leads to possibility to forge the personal DS corresponding to the DS algorithm used in the protocol. In other words the CDS protocol is not less secure against the forgery attack than the used DS algorithm. Since GOST-1994 algorithm is secure the considered collective signature protocol is also secure (We consider GOST 34.10-1994 as a secure DS algorithm since it has been widely investigated and used in practice).

• **The second attack (the member signature secret key forgery):** Assuming that $n - 1$ signers that shares a CDS (R, S) with the n^{th} signer are able to figure out the secret key of the n^{th} signer. Suppose also that (R^*, S^*) is a personal DS of the n^{th} user formed using the underlying DS algorithm and the signature (R^*, S^*) corresponds to the hash value H calculated from the message. Then we have:

$$R^* = \left(\alpha^{\frac{S^*}{H}} y^{\frac{-R^*}{H}} mod\ p\right) mod\ q \tag{32}$$

Attackers can generate values k_i and calculate $R_i = (\alpha^{k_i} mod\ p)\ mod\ q$ for $i = 1, 2, \ldots, n-1$. Then they computes parameters $R = (R^* \prod_{i=1}^{n-1} R_i\, mod\ p) mod\ q$ and S_i satisfying the equation:

$$R_i = \left(\alpha^{\frac{S_i}{H}} y_i^{-\frac{R}{H}} mod\ p\right) mod\ q \tag{33}$$

Using designation: $y^* = y^{\frac{R^*}{R}} mod\ p$

From (32) and (33) we get: $R = (\alpha^{S/H}\, Y^{-R/H}\ mod\ p)\ mod\ q$,

where $S = (S^* + \sum_{i=1}^{n-1} S_i) mod\ q$ and $Y = (y^* \prod_{i=1}^{n-1} y_i)\ mod\ p$.

This means that the attackers have get the collective signature value (R, S) that corresponds to n signers, the nth signer owning the public key $y^* = \alpha^{x^*} mod\ p$. In accordance to our assumption the attackers are able to compute the secrete key x^* from (R, S). From (33) it is easy to get: $x = Rx^*/R^* mod\ q$.

Thus, the attackers have calculated the secret key of the nth user using his personal DS, i.e., it is formally proved that if the CDS protocol is insecure, then the underlying DS algorithm is also insecure (in other words, the CDS protocol is not less secure than the underlying DS algorithm).

The RCS schemes proposed in this article use the CDS scheme as the basic scheme, so it is resistant to these two types of internal attacks.

4.2 Security Advantages of The New CDS Schemas

- We use GOST-1994 to build the RCS schemes, so basically these schemes have all the security advantages that the discrete logarithm problem over prime finite fields and this signature standard provides.
- The collective signature scheme, the CDS-02 scheme, has the following two outstanding security advantages: (i) Any illegal actions on generating the values R_i and S_i and the signers that have produced such actions can be easily detected using the values R_i and S_i and the following equation: $R_i = (\alpha^{S_i/H} y_i^{-R/H} mod\ p) mod\ q$, which holds only if the values R_i and S_i are produced correctly by the *i-th* signer and (ii) None of the members generates a valid DS. This fact imparts to the CDS the property of the internal integrity. It is evident that the size of the CDS does not depend on m.

It is easy to show that the constructed representative collective signature schemes also have these security advantages.

- The group signature scheme GDS-02 has the following characteristics: The technique of "masking" the signing group member's public key is used to ensure the privacy of the signer; Information of all signers is contained in the U component. When a signature dispute occurs, the group manager can completely resolve it thanks to the information stored in this component; The request for mutual authentication between the members of the signing group and the group manager can be done through the formula to generate the λ mask coefficient and the internal key pair (n, e) generated from the RSA algorithm; There is absolutely no exchange or sharing of private keys and secret values between the signing team member and the team leader, so this scheme can be implemented on Internet or on existing PKI [23].

4.3 Performance of The New CDS Schemas

This cost of RCS schemas in this article are shown in Tab. 1. For details on the information in this table see [24]; For the notations and conventions in this table see [25].

Table 1: Time cost of the RCS schemes in this article

The schema	Time to create the signature	Time to verify the signature
RCS.01	$\mathrm{U} = \sum_{j=1}^{g} \left(481m_j + 1\right) T_m$	$(484 + g)$ T_m
	$\mathrm{R} = \sum_{j=1}^{g} \left(241m_j + 241\right) T_m$	
	$S = \sum_{j=1}^{g} \left(485m_j + 486\right) T_m$	
	$Sum = \sum_{j=1}^{g} \left(1207m_j + 728\right) T_m$	
RCS.02	$\mathrm{U} = \sum_{j=1}^{g} \left(481m_j + 1\right) T_m$	$(484 + g + m)\ T_m$
	$\mathrm{R} = \left[\sum_{j=1}^{g} \left(241m_j + 241\right) + 241m\right] T_m$	
	$S = \left[\sum_{j=1}^{g} \left(485m_j + 486\right) + 486m\right] T_m$	
	$Sum = \left[\sum_{j=1}^{g} \left(1207m_j + 728\right) + 727m\right] T_m$	

Tab. 1 shows that the time cost to create and to verify the signature of the RCS schema in this article is not much more extensive than compared to CDSs in [8].

5 Discussion

- Obviously, until now, only the representative collective digital signatures and the representative digital signature schemes can meet the requirement of authentication based on a single digital signature, that is, only one-time authentication, for all members in multi-level functional signing collectives. Members of this signing collective include: (i) People representing different signing groups, they are called group leaders, and (ii) Single signers, but acting as group leaders. This is why the representative collective signature is formed in two steps: First, the group leader of each group signs together with their members to create the group signature of that signing group. Then, all group's leaders and personal signers in the signing collective will together create a RCS of the signing collective. Subsequent authentication is based solely on this final signature.

- Since the group leader uses the secret key d, generated from the RSA algorithm, to generate the λ coefficient so receiving this coefficient, the signing group members can easily check to see if it came from their group leader or not by using the public key e to decrypt the λ coefficient.
- The formulas for generating λ coefficients and the U-component in the group signature generation procedure show that only the group leader can "open" the generated signature to identify all the people who participated in the generation of GDS of the signing group that he/she manages [6].
- The limitation of these new RCSs is the increase in the size of the signature. This limitation can be overcome if there is some signature scheme that helps to create a representative collective signature consisting of only two components but still contains the information of all the people who participated in the creation of this signature.

6 Conclusion

In this paper, first, we use GOST-1994 digital signature standard to build a CDS scheme and a GDS scheme. We then use these two schemes as the basis for building the proposed CDS schemes and protocols: (i) CDS for multiple signing groups; (ii) CDS for multiple signing groups and multiple personal signers. The RCSs are created on digital message M consisting of three components (U, R, S), with a fixed size, equal to $|p|+2|q|$, independent of the number of signing groups and personal signers. Information of every member participating in signature formation is stored in the U component. It is used for the signer and problem-solving the "disclaimer of liability" later. All signature schemes in this paper have been proven correct.

The article also shows that the collective digital signature protocol built according to GOST-1994 digital signature standard is capable of resisting two common types of inside attacks: (i) Attacks on the secret key of signing group members; (ii) Forging the signature of a member of the signing group. With the GDS scheme, the public key "masking" technique, through the λ coefficient, is applied to both the signing group member and the group leader. This not only ensures the privacy of every signing group member, but also creates mutual authentication between the signing group members and the group manager. This is one of the advantages of the group signature scheme that we propose in this study. Finally, we analyzed the security advantages and evaluated the performance, and time costs to create the signature and verify the signature, of each RCS scheme built.

The results obtained in this study show that the representative collective signature (RCS) has high feasibility and an acceptable level of security. However, because GOST-1994 is formed on the basis of the discrete logarithm problem on finite prime fields so there are some limitations that can be overcome if the scheme is formed according to the GOST R34.10-2001 digital signature standards or the GOST R34.10-2012 digital signature standards. This is our future work.

Funding Statement: This article is supported by Duy Tan University, Da Nang, Vietnam.

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. 183–186, 2009.

[2] J. Pieprzyk, T. Hardjono and J. Seberry, *Fundamentals of computer security*, Berlin: Springer-Verlag, 2003.

[3] D. Chaum and E. Heyst, "Group signatures," in *Advances in Cryptology-EUROCRYPT' 91*. Springer-Verlag, pp. 257–265, 1991.
[4] 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, pp. 332–351, 2016.
[5] Q. Alamélou, O. Blazy, S. Cauchie and Ph Gaborit, "A code-based group signature scheme," *Designs Codes and Cryptography*, vol. 82, no. 1–2, pp. 469–493, 2017.
[6] A. A. Moldovyan and N. A. Moldovyan, "Group signature protocol based on masking public keys," *Quasigroups And Related Systems*, vol. 22, no. 1, pp. 133–140, 2014.
[7] 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*, vol. 6, no. 6, pp. 1–5, 2016.
[8] 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, Springer, vol. 672, pp. 200–208, 2018.
[9] N. K. Tuan, H. N. Duy and N. A. Moldovyan, "Collective signature protocols for signing groups based on problem of finding roots modulo large prime number," *International Journal of Network Security & Its Applications*, vol. 13, no. 4, pp. 59–69, 2021.
[10] 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*, Berlin, Heidelberg, New York, Springer-Verlag, vol. 950, pp. 428–432, 1995.
[11] D. Chaum, "Blind signatures for untraceable payments," in *Proc. Advances in Cryptology–CRYPTO'82*, Plenum Press, pp. 199–203, 1983.
[12] 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.
[13] K. Itakura and K. Nakamura, "A public key cryptosystem suitable for digital multisignatures," *NEC Research and Development*, vol. 71, pp. 1–8, 1983.
[14] D. M. Tuan, "New elliptic curve digital multi-signature schemes for multi-section messages," in *Proc. Int. Conf. on Computing and Communications Technologies Research-Innovation and Vision for the future*, Vietnam, pp. 25–28, 2012.
[15] D. Poulakis and R. Rolland, "A digital signature scheme based on two hard problems," in *Computation, Cryptography, and Network Security*, Springer, pp. 441–450, 2015.
[16] N. A. Moldovyan, "Digital signature scheme based on a new hard problem," *Computer Science Journal of Moldova*, vol. 16, no. 2, pp. 163–182, 2008.
[17] 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.
[18] R. L. B. Daniel, "Generic groups, collision resistance, and ECDSA," *ACM journal: Designs, Codes and Cryptography*, vol. 35, no. 1, pp. 119–152, 2005.
[19] N. A. Moldovyan and V. A. Shcherbacov, "New signature scheme based on difficulty of finding roots," *Quasigroups and Related Systems*, vol. 20, no. 1, pp. 261–266, 2012.
[20] M. Punita and M. Sitender, "RSA and its correctness through modular arithmetic," in *Int. Conf. On Methods And Models In Science And Technology, ICM 2st-10, AIP Conf. Proc. 1324*, pp. 463–466, 2010.
[21] 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.
[22] 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.
[23] H. Yong, C. Fugui and Q. Peixin, "Research on digital signature based on digital certificate," in *Proc: Proc. of 14th Youth Conf. on Communication*, Scientific Research, pp. 467–470, 2009.

[24] T. N. Kim, D. H. Ngoc and N. A. Moldovyan, "Constructing collective signature schemes using problem of finding roots modulo," *Computers, Materials & Continua*, vol. 72, no. 1, pp. 1105–1122, 2022.
[25] C. Popescu, "Blind signature and BMS using elliptic curves, Studia univ babes–Bolyai, *Informatica*, pp. 43–49, 1999.