A New Security Enhancing Solution when Building Digital Signature Schemes

Kim Tuan Nguyen
Faculty of Computer Science, Phenikaa University, Ha Noi, Vietnam
tuan.nguyenkim@phenikaa-uni.edu.vn

Ha Nguyen Hoang
University of Sciences, Hue University, Hue, Vietnam
nguyenhoangha@hueuni.edu.vn (corresponding author)

Duy Ho Ngoc
Faculty of Information Technology, Military Technical Academy, Ha Noi, Vietnam
duyho84@gmail.com

Received: 29 January 2025 | Revised: 25 February 2025 | Accepted: 7 March 2025

 DOI: https://doi.org/10.48084/etasr.10370

ABSTRACT

This article introduces a new solution to enhance the security of the digital signature system. First, a new type of hard problem is proposed, which is then used to construct a digital signature scheme. The core difficulty lies in the Discrete Logarithm Problem (DLP) modulo, a composite number which extends the DLP over a finite field by replacing the prime modulus with a composite one. This change leads to a digital signature scheme built on the DLP modulo, a composite number having the same level of security as the schemes constructed simultaneously on two hard problems: the DLP and the Integer Factorization Problem (IFP). This can be seen as a new direction in using two hard problems concurrently to construct a digital signature scheme. The current study also demonstrates that the digital signature scheme built upon a newly proposed problem, achieves a higher security level of 128 bits while minimizing the signature size to 512 bits. That is, although the schemes are built on a single hard problem, a possible attacker must solve two hard problems simultaneously in order to break them. Furthermore, the proposed hard problem can be used to construct both single-signer and multi-signer digital signature schemes, demonstrating the security and applicability of the new hard problem, introduced in this paper.

Keywords-prime modulus; composite modulus; Schnorr's digital signature; collective digital signature

I. INTRODUCTION

Enhancing the security of digital signature schemes is an issue of great interest to many scientists and researchers in cryptography and information security, both domestically and internationally. Numerous solutions to this issue have been researched and published. The simplest solution that can be applied is to increase the size of the prime numbers in the IFP and increase the size of the modulus in the discrete logarithm problem. However, this size increase will result in slower processing speeds and larger digital signature sizes. Consequently, a common approach involves combining two difficult mathematical problems to construct a digital signature scheme [1-3]. The underlying idea is that to break such a scheme, a possible attacker would need to simultaneously solve both problems. Naturally, solving one difficult problem is hard enough, but solving two simultaneously is even harder. Two commonly used problems in this context are the IFP and the DLP over a finite field, modulo a prime number [4, 5]. However, practical experience has shown that this approach is not entirely foolproof.

Authors in [6] proposed a digital signature scheme based on the hardness of both IFP and the DLP. However, in [7], it was demonstrated that an attacker could break the previous study scheme by solving only the IFP. Meanwhile, authors in [8] argued that solving the DLP alone was sufficient to break the scheme. Authors in [9] further claimed that any attacker could forge signatures generated by the scheme proposed in [6] without solving either problem. Authors in [10] demonstrated that the digital signature scheme presented in [8] and which relied on the hardness of both the IFP and DLP was insecure and contained vulnerabilities. Also, in [10], a new signature scheme was proposed to address these shortcomings. However, in [8] it was shown that the scheme of [10] was also insecure, since an attacker could forge signatures generated by its scheme by solving only the DLP, regardless of the message [11].

Meanwhile, authors in [12] demonstrated that their digital signature scheme, a variant of DSA, is based on the hardness of both IFP and DLP. They showed that an attacker must solve both problems simultaneously to break their scheme. Also they proved that their scheme is resistant to common attacks, such as key-only attacks, chosen-message attacks, and known partial key attacks.

Thus, the problem at hand is how to simultaneously use two hard-to-solve problems (hereinafter called hard problems), among familiar ones, such as IFP, DLP, or the root extraction problem over a finite field, to construct a digital signature scheme while ensuring the fundamental requirement that an attacker must solve both hard problems used in the scheme's construction in order to break it. A new type of hard problem to address this issue is proposed: The DLP modulo, a composite number *n*.

This new hard problem is used as the basis for constructing both single-signer and collective digital signature schemes. The proposed schemes are built on a single hard problem but breaking these schemes requires an attacker to solve both hard problems simultaneously, namely IFP and the DLP. This shows that the security level of digital signature schemes based on the DLP modulo a composite number is equivalent to that of schemes built on both hard problems previously mentioned. Moreover, if the system parameters of the proposed schemes are chosen appropriately, the size of the signatures can be significantly reduced. The proposed schemes are based on the Schnorr digital signature standard, so their security and resistance to attacks have been validated.

In the context of quantum computing advancements, adopting post-quantum cryptographic algorithms is an optimal solution to enhance the security of digital signature schemes. These schemes rely on mathematically hard problems, such as lattices, code-based, or hash-based cryptography. A notable example includes the CRYSTALS-Dilithium and Falcon schemes [13], which integrate secure key generation mechanisms with zero-knowledge proof techniques to ensure correctness without revealing sensitive information. Additionally, incorporating "side-channel protection" and "code obfuscation" mechanisms mitigates practical attacks, such as timing analysis or information leakage through side channels. Combining post-quantum algorithms with supplementary security layers not only safeguards against quantum-based attacks, but also facilitates the transition to the new era of cryptography. The post-quantum approach is not addressed in this paper.

II. THE PROPOSED HARD PROBLEM

The new type of hard problem presented in this work is the DLP modulo, a composite number *n*, while the values for the problem's parameters are also selected.

The problem is defined as follows: Find the number *x* given the numbers α and *y*, such that *x*, α and *y* are related by (1):

$$y = a^x \bmod n \tag{1}$$

where *n* is the product of two large prime numbers *p* and *q*:

$$n = p \cdot q$$

The element α serves a similar role to the generator *g* in the DLP modulo a prime number. Specifically, α is a generator of the multiplicative group of order γ. Therefore, in this composite modulo discrete logarithm problem, α must be chosen such that its order γ modulo *n* satisfies one of the following four conditions:

a. γ is a prime number such that $\gamma \nmid (p-1), \gamma \mid (q-1)$.

b. γ is a prime number such that $\gamma \nmid (p-1), \gamma \nmid (q-1)$.

c. γ is a prime number such that $\gamma \mid (p-1), \gamma \mid (q-1)$.

d. γ is a composite number and can be expressed as: $\gamma = \gamma' \cdot \gamma''$, where: $\gamma' \nmid (p-1), \gamma' \mid (q-1), \gamma'' \mid (p-1), \gamma'' \nmid (q-1)$.

The order γ of a positive integer α modulo *n*, denoted as $\gamma = ord_n(a)$, is the smallest positive integer γ that satisfies:

$$a^{\gamma} \equiv 1 \pmod n \tag{2}$$

For the proposed hard problem, the selection of values for α and γ is crucial, since they must not only satisfy the above constraints, but also ensure the necessary security level for the digital signature scheme built upon it. These values are selected based on the following three lemmas:

Lemma 1: Suppose that *p*, *q* are strong primes, $n = p \cdot q$, γ be/is the order of a generator of a group satisfying the two conditions: $\gamma \nmid (p-1), \gamma \mid (q-1)$, *a* be/is a generator of the multiplicative group of order γ (that is $a^{\gamma} = \bmod n$) , then (Greatest Common Divisor) $\mathrm{GCD}(a-1, n) = p$.

Proof: By applying Euler's theorem, we get:

$$\forall a \in Z_n^* : GCD(a,n) = 1,\ \exists \gamma : a^{\gamma} = 1\ (mod\ n),\ \gamma \mid \varphi(n),$$

where $\varphi(n)$ is the Euler's totient function of *n*: $\varphi(n) = (p-1)(q-1)$.

$$\forall \beta\ : GCD(\beta, n) = 1 \Rightarrow \beta^{\varphi(n)} = 1 (\bmod n) \Rightarrow$$

$$\Rightarrow \beta^{\varphi(n)} = a^{\gamma} = 1 (\bmod n) \Rightarrow \forall \beta\ \exists\ a' : \beta^{\varphi(\frac{n}{\gamma})} = a' \neq 1 (\bmod n).$$

$$\forall a\ \exists\ i \in Z_n : a = (a')^i = (\beta^i)^{\frac{\varphi(n)}{\gamma}} \neq 1 \Rightarrow a = \beta^{\frac{\varphi(n)}{\gamma}}\ (mod\ n) =$$

$$= \beta^{\frac{(p-1)(q-1)}{\gamma}}\ (mod\ n).$$

Considering that $\gamma \nmid (p-1), \gamma \mid (q-1)$, *a* becomes:

$a = \left(\beta^{(p-1)}\right)^{(q-1)/\gamma}\ (mod n)$. So if *n* is set to be *p* (*n*=*p*), then:

$a = \left(\beta^{(p-1)}\right)^{(q-1)/\gamma}\ (mod p)$ and $\beta^{(p-1)} = 1\ (mod\ p)$. So:

$a = 1^{(q-1)/\gamma} (mod\ p) \Rightarrow a = 1(\text{mod } p) \Rightarrow a - 1 = 0(\text{mod } p) \Rightarrow$
$\Rightarrow p \mid (a-1) \Rightarrow GCD(a-1, n) = p.$

There is a similar proof when $\gamma \mid (p-1), \gamma \mid (q-1)$. Thus, in both cases, the composite number n can be easily factored using the extended Euclidean algorithm by computing $\text{GCD}(a-1, n)$.

Lemma 2: Suppose that p, q are strong primes such that $q < p$, $n = p \cdot q$, γ is a prime order of a generator fulfilling the conditions: $\gamma \mid (p-1), \gamma \mid (q-1), \gamma^2 \nmid (p-1)$, and $\gamma^2 \nmid (q-1)$, a is a generator of the multiplicative group of order (that is: $a^{\gamma} = 1 \bmod n$), then $GCD(a-1, n) = 1$.

Proof: Applying Euler's theorem, we get:

$$\forall a \in Z_n^* : GCD(a, n) = 1,\ \exists \gamma : a^{\gamma} = 1\ (mod\ n),\ \gamma \mid \varphi(n)$$

where $\varphi(n)$ is the Euler's totient function of n, as before.

Since $\gamma \mid (p-1), \gamma \mid (q-1)$, and $\gamma \mid \varphi(n)$, it is concluded that:

$$\forall \beta : GCD(\beta, n) = 1 \Rightarrow \beta^{L(n)} = \beta^{\gamma} = 1(\text{mod } n)$$

and $\gamma^2 \mid \varphi(n)$.

where $L(n)$ is the generalized Euler's totient function of n, $L(n) = LCM(p-1)(q-1)$.

So: $\beta^{\varphi(n)} = \beta^{L(n)} = \beta^{\gamma} = \alpha^{\gamma} = 1(\text{mod } n).$

$$\exists \alpha', \beta' : \alpha' = \beta'^{\frac{(p-1)(q-1)}{\gamma^2}} (\text{mod } n) \neq 1,\ \alpha'^{\gamma} = 1(\text{mod } n)$$

Moreover: $\forall a\ \exists\ i \in Z_n : a = (a')^i (\text{mod } n)$

So: $\alpha = \beta'^{\frac{i(p-1)(q-1)}{\gamma^2}} (\text{mod } n) = \beta^{\frac{(p-1)(q-1)}{\gamma^2}} (\text{mod } n) \Rightarrow$

$\Rightarrow \gamma \mid \frac{(p-1)(q-1)}{\gamma^2}$. And since $\gamma^2 \nmid (p-1),\ \gamma^2 \nmid (q-1)$, it is concluded that: $\alpha = \beta^{\frac{(p-1)(q-1)}{\gamma^2}} = \beta^{\frac{L(n)}{\gamma^2}} \neq 1(\text{mod } n) \Rightarrow$

$\Rightarrow a \neq 1(\text{mod } p)$ and $a \neq 1(\text{mod } q) \Rightarrow GCD(a-1, n) = 1.$

Lemma 3: Suppose that p, q are two strong primes with $q < p$, $n = p \cdot q$, $\gamma = \gamma'\gamma''$ is the order of generator α, where γ' and γ'' are distinct primes satisfying: $\gamma' \mid (p-1)$, $\gamma'' \mid (q-1), \gamma' \nmid (q-1),\ \gamma'' \nmid (p-1)$, then $GCD(a-1, n) = 1$.

Proof: Applying Euler's theorem, we get:

$$\forall a \in Z_n^* : GCD(a, n) = 1,\ \exists \gamma : a^{\gamma} = 1\ (mod\ n),\ \gamma \mid \varphi(n).$$

$\gamma' \mid (p-1), \gamma'' \mid (q-1), \gamma = \gamma'\gamma'' \mid \varphi(n) \Rightarrow$
$\Rightarrow \forall \beta : GCD(\beta, n) = 1 \Rightarrow \beta^{L(n)} = \beta^{\gamma} = 1(\text{mod } n).$

So: $\beta^{\varphi(n)} = \beta^{L(n)} = \beta^{\gamma} = \alpha^{\gamma} = 1(\text{mod } n)$

$$\exists \alpha', \beta' : \alpha' = \beta'^{\frac{(p-1)(q-1)}{\gamma}} (\text{mod } n) \neq 1,\ \alpha'^{\gamma} = 1(\text{mod } n)$$

$\forall a\ \exists\ i \in Z_n : a = (a')^i (\text{mod } n) \Rightarrow$
$\alpha = \beta'^{\frac{i(p-1)(q-1)}{\gamma}} (\text{mod } n) = \beta^{\frac{(p-1)(q-1)}{\gamma}} (\text{mod } n) \Rightarrow$
$\Rightarrow \gamma \nmid \frac{(p-1)}{\gamma'} \frac{(q-1)}{\gamma''}.$

And since $\gamma' \nmid (q-1),\ \gamma'' \nmid (p-1)$, it is concluded that:

$\alpha = \beta^{\frac{(p-1)(q-1)}{\gamma}} = \beta^{\frac{L(n)}{\gamma}} \neq 1(\text{mod } n) \Rightarrow$
$\Rightarrow a \neq 1(\text{mod } p)$ and $a \neq 1(\text{mod } q) \Rightarrow GCD(a-1, n) = 1.$

Based on these lemmas, two possible choices for α are determined:

- α is selected to be an element with prime order $\gamma(\text{mod } n)$, where $n = pq,\ \gamma \mid (p-1),\ \gamma \mid (q-1)$, given that p and q are primes.

  In this case, the value of γ cannot be kept secret because it can be easily computed using:

  $$n - 1 = (u\gamma + 1)(v\gamma + 1) - 1 = uv\gamma^2 + u\gamma + v\gamma =$$
  $$= \gamma(uv\gamma + u + v) \quad (3)$$

  where u and v are factors.

- α is selected to be an element of composite order $\gamma = \gamma'\gamma''(\text{mod } n)$ where $n = pq,\ \gamma' \mid (p-1),\ \gamma'' \mid (q-1),$ $\gamma' \nmid (q-1)$, and $\gamma'' \nmid (p-1)$.

  In this case, the value of γ can be kept secret because it is computed by (3):

  $$n - 1 = (u\gamma' + 1)(v\gamma'' + 1) - 1 = u\gamma' v\gamma'' + u\gamma' + v\gamma'' \quad (4)$$

  However, all three values for γ, γ', γ'' must be kept secret because the expressions:

  $GCD(\alpha^{\gamma'} - 1, n) = p,\ GCD(\alpha^{\gamma''} - 1, n) = q$, exist.

Thus, to construct digital signature schemes with a 128-bit security level, the values γ, γ', and γ'' should be chosen as follows:

$|\gamma| \geq 256$ bits, $|\gamma'| \geq 128$ bits, $|\gamma''| \geq 128$ bits.

Additionally, the composite modulus n must be the product of two strong primes p and q, with corresponding sizes:

$|p| \geq 2464$ bits and $|q| \geq 1532$ bits .

With this parameter set, the difficulty of IFP for *n* and DLP modulo *p* are equivalent and equal to $O(2^{128})$ multiplications.

The ability to solve DLP modulo *n* can lead to solving a combination of two hard problems: IFP (factoring *n* into its prime factors) and DLP modulo each of the prime factors of *n*, namely *p* and *q*.

Indeed, since $y = a^x \bmod n$ and $n = p \cdot q$, we have:

$$\begin{cases} y = a^x \bmod p \\ y = a^x \bmod q \end{cases} \quad (5)$$

These are clearly two instances of DLP modulo, the prime numbers *p* and *q*.

When solving these two problems, it is possible that the values of *x* obtained from the problem modulo *p* and the problem modulo *q* are not equal. That is:

$$\begin{cases} y = a^{x_1} \bmod p \\ y = a^{x_2} \bmod q \end{cases} \quad (6)$$

In this case, the specific cases of the order of the positive integer modulo *n* are considered:

- If element α has prime order γ satisfying the conditions: $\gamma|(p-1)$, $\gamma|(q-1)$, then:

$$a^{\gamma} = 1 \bmod n \Rightarrow \begin{cases} a^{\gamma} = 1 \bmod p \\ a^{\gamma} = 1 \bmod q \end{cases} \Rightarrow$$

$$\Rightarrow y = a^x \bmod n \Rightarrow \begin{cases} y = a^{x_1} \bmod p \\ y = a^{x_2} \bmod q \end{cases} \quad (7)$$

So, it is getting clear that the values of x_1 and x_2 obtained from solving the DLP modulo *p* and *q*, respectively, are identical $x_1 = x_2$.

This result shows that solving DLP modulo *n* requires solving two sub-problems simultaneously: (i) IFP of the composite number *n* and (ii) DLP modulo, a prime factor of *n*, preferably the smaller one. It is not necessary to solve both DLP modulo *p* and *q* simultaneously.

Thus, the problem of finding the discrete logarithm modulo *n* is actually the same as factoring the number *n* and solving DLP modulo *q* (where $q < p$ and $n = p \cdot q$). Because of this, the choice of *p* and *q* must be in a way that makes it very difficult to solve DLP modulo *q*.

- If element α has prime order $\gamma = \gamma'\gamma''$, which is a composite number satisfying the conditions $\gamma'|(p-1)$, $\gamma''|(q-1)$, $\gamma'|(q-1)$, and $\gamma''|(p-1)$, then:

$$a^{\gamma} = 1 \bmod n \Rightarrow \begin{cases} a^{\gamma'} = 1 \bmod p \\ a^{\gamma''} = 1 \bmod q \end{cases} \Rightarrow$$

$$\Rightarrow y = a^x \bmod n \Rightarrow \begin{cases} y = a^{x_1} \bmod p \\ y = a^{x_2} \bmod q \end{cases}$$

So, the solutions x_1 and x_2 obtained from solving the DLP for *p* and *q* will not be the same. To find the secret value *x*, the following system of equations has to be solved:

$$\begin{cases} x \equiv x_1 \bmod \gamma' \\ x \equiv x_2 \bmod \gamma'' \end{cases} \quad (8)$$

Using the Chinese Remainder Theorem, the value of *x* is gained:

$$x = (x_1 c_1 \gamma\gamma' - 1 + x_2 c_2 \gamma\gamma'' - 1) \bmod \gamma \quad (9)$$

where:

$$\begin{cases} c_1 = (\gamma'\gamma - 1) \bmod \gamma' \\ c_2 = (\gamma''\gamma - 1) \bmod \gamma'' \end{cases} \quad (10)$$

This result demonstrates that solving DLP modulo a composite number *n* where $n = p \cdot q$, requires simultaneously solving three sub-problems: (i) IFP of *n*, (ii) DLP modulo *p*, and (iii) DLP modulo *q*.

III. THE PROPOSED HARD PROBLEM-BASED DIGITAL SIGNATURE SCHEMES

The proposed DLP modulo a composite number *n* is utilized as a foundation for constructing two digital signature schemes: (i) a single-signer digital signature scheme (DSS-0824) and (ii) a multi-signer digital signature scheme (CDS-0824). This demonstrates the versatility and applicability of the proposed problem.

A. Proposed Digital Signature Scheme (DSS-0824)

1) System Parameters and Keys

To generate the system parameters (*p, q, n,* α*,* γ) and keys for the scheme, the following steps should be performed:

- Select two strong prime numbers p and q, and compute their product $n = p \cdot q$.
- Generate a generator α of the prime order subgroup of order γ modulo *n* satisfying the conditions: $\gamma|(p-1)$, $\gamma|(q-1)$, and $|\gamma| = 256$ bits.
- Generate the secret and public key pair as follows: Choose a secret key *x* with length of 256 bits. The public key *y* is computed using the formula $y = a^x \bmod n$.

Thus, the public parameters are (*n,* α*,* γ), the private parameters are (*p, q*), the public key is *y*, and the secret key is *x*.

2) Signature generation algorithm for message M

The Schnorr digital signature scheme [14] is utilized as a foundation to develop the proposed scheme on.

The signer follows these steps to generate a digital signature for message *M*:

- Generate a random number *k*, which acts as a pseudo-random secret key, of 256 bits size and then compute *R* using:

$$R = a^k \bmod n \quad (11)$$

- Use a collision-resistant hash function F_H with a 256 bit output size to compute the first part of the signature, component *E*, using:

$$E = F_H(R \| M) \quad (12)$$

where $\|$ is the concatenation operator and F_H is a secure hash function (for example SHA-1).

- Compute the second part *S* of the signature using:

$$S = k + xE \bmod \gamma \quad (13)$$

Thus, the digital signature generated for the message *M* is the pair (E, S). The signature size is 512 bits ($|E| + |S| = 256 + 256 = 512$ bits).

3) Signature Verification Algorithm for Message M

The signature verifier follows these steps to verify the signature (E, S) on message *M*:

- Compute the value $\tilde{R}$ using:

$$\tilde{R} = y^{-E} a^S \bmod n \quad (14)$$

- Compute the value $\tilde{E}$ using:

$$\tilde{E} = F_H(\tilde{R} \| M) \quad (15)$$

- Compare *E* and $\tilde{E}$. If $E = \tilde{E}$, then the signature is valid and the integrity of *M* is assured. Otherwise, the signature is invalid.

4) Proof of Correctness of the DSS-0824 Scheme

The correctness of this scheme can be proven considering the following:

$$\tilde{R} = y^{-E} a^S \bmod n$$

$$\tilde{R} \equiv a^{-xE} a^{k+xE} \bmod n \equiv a^k \bmod n \equiv R \bmod n = R$$

Therefore: $R = \tilde{R}$.

Since $\tilde{E} = F_H(\tilde{R} \| M)$ and $R = \tilde{R}$, it is concluded that:

$\tilde{E} = F_H(R \| M) = E$. So, $\tilde{E} = E$, which confirms the correctness of the verified scheme.

B. Proposed Collective Digital Signature Scheme (CDS-0824)

A collective/group digital signature scheme is a specific type of multi-signature scheme. It allows a group of *m* signers to collaborate and generate a single signature on a document *M*. The signature generated by this scheme is called a collective digital signature. The verification of the signature and the subsequent authentication of the signers are performed only once on this signature.

1) System Parameters and Keys

The system parameters of this digital signature scheme are chosen as in the previous DSS-0824 scheme.

In this scheme, the signer group is fixed and consists of *m* signers, who are responsible for generating a collective signature on a message *M*. Each signer in the group has a 256 bit number private key x_i with $i = 1, 2, 3, ..., m$. The corresponding public key y_i of each signer is computed as $y_i = \alpha^{x_i} \bmod n$.

The public key *Y* of the signer group, which is used for signature verification and subsequent authentication of the signer group, is computed by:

$$Y = y_1 y_2 ... y_m \ mod n \quad (16)$$

To prevent forgery in the formation of the collective public key, the public keys y_i of each signer, must be made public within the signer group, and all members of this group must participate in the calculation of the collective public key *Y*. Only when *Y* is confirmed by all members of the signer group, can it be published as the public key of the signer group.

2) The Algorithm for Generating a Collective Digital Signature on Message M

The process of generating a group signature on message *M*, involving a group of *m* signers, consists of the following steps:

- Each signer in the group randomly selects a 256-bit number k_i with $i = 1, 2, 3, ..., m$, which acts as a temporary secret key, and then calculates R_i according to:

$$R_i = \alpha^{k_i} \bmod n \quad (17)$$

Then, the i signer transmits R_i to all members of the signing group.

- Any member of the signing group, representing the group, computes a common random value *R* for the entire group, calculated using:

$$R = R_1 R_2 .... R_m \ mod n \quad (18)$$

Next, the first component *E* of the group signature is calculated according to:

$$E = F_H(M \| R \| Y) \quad (19)$$

Then, E is sent to all members of the signing group.

- Each signer in the signing group computes their individual share S_i using:

$$S_i = (k_i + x_i E) \bmod \gamma \quad (20)$$

Then, S_i is sent to all members of the signing group.

The S_i provided by each member of the signing group, serve as the second shares, which will be combined to form the second component of the group signature.

The pair (R_i, S_i) is considered the individual signature of the i-th signer on the message *M*.

- After receiving the individual signatures (R_i, S_i) from all members of the signing group, a designated verifier, representing the group, will validate these signatures by calculating all R_i' using:

$$R_i' = y_i^E \alpha^{S_i}, \text{ for } i = 1, 2, ..., m \quad (21)$$

Then, R' is computed using:

$$R' = \prod_{i=1}^{m} R_i' \bmod n \quad (22)$$

Next, R' is compared with R. If $R' = R$, the validity of all individual signatures is confirmed, and the second component of the group signature is computed as described in the next step.

- Any member of the signing group, representing the group, computes the value of the second component S of the signature using:

$$S = (S_1 + S_2 + ... + S_m) \bmod \gamma \quad (23)$$

The group signature of the signing group consisting of *m* members on the message *M* is the pair (E, S).

It is evident that the size of this signature is independent of the number of signers participating in generating the signature on message *M* and it is exactly equal to the sum of the lengths of γ and *E*. If a hash function F_H with a 256 bits output is chosen, then the size of the group signature is 512 bits ($|E| + |S| = 256 + 256 = 512$ bits).

To prevent forgery in the computation of *R*, the values of R_i must be publicly known within the signing group, and all members of the group must participate in the computation of *R*. Only when all members of the signing group have confirmed the value of *R* can it serve to compute the component *E* of the group signature.

3) Signature Verification Algorithm on Message M

The process of verifying a group signature on message *M*, generated by a signing group of *m* members, is performed by the verifier as follows:

- Compute the group public key *Y* using:

$$Y = y_1 y_2 ... y_m \ mod n$$

where y_i is the public key of each member in the signing group.

- Compute the value R' using:

$$R' = \alpha^S Y^{-E} \bmod n \quad (24)$$

- Compute the value E' using:

$$E' = F_H(M \| R' \| Y) \quad (25)$$

- Compare E' with *E*. If $E' = E$, then the group signature is valid and all members of the signing group have contributed to its creation. Otherwise, the group signature is invalid.

4) Proof of Correctness of the CDS-0824 Scheme

The correctness of the CDS-0824 group signature scheme is established through the correctness of the individual signature verification procedure (R_i, S_i) and the group signature verification procedure (E, S).

The correctness of the individual signature verification procedure for each member of the signing group is evident. The proof of the correctness of the group signature verification procedure goes as follows:

$$R' = \alpha^S Y^{-E} \bmod n \Rightarrow$$

$$R' = \alpha^{\sum_{i=1}^{m}(k_i + x_i E)} Y^{-E} \bmod n \Rightarrow$$

$$R' = \alpha^{\sum_{i=1}^{m} k_i} \alpha^{\sum_{i=1}^{m} x_i E} Y^{-E} \bmod n \Rightarrow$$

$$R' = \prod_{i=1}^{m} \alpha^{k_i} \left(\alpha^{\sum_{i=1}^{m} x_i} \right)^E Y^{-E} \bmod n \Rightarrow \quad (26)$$

$$R' = \prod_{i=1}^{m} R_i Y^E Y^{-E} \bmod n \Rightarrow$$

$$R' = R_1 R_2 .... R_m \bmod n \Rightarrow$$

$$R' = R$$

Since $E' = F_H(M \| R' \| Y)$ and $R' = R$, it follows that:

$$E' = F_H(M \| R' \| Y) = F_H(M \| R \| Y) = E \quad (27)$$

Therefore, $E' = E$. This proves the correctness of the group signature verification procedure and confirms the correctness of the proposed CDS-0824 group signature scheme.

IV. EVALUATION OF THE SECURITY LEVEL AND PERFORMANCE OF THE HARD PROBLEM AND THE PROPOSED SCHEME

A. On the Hardness of the Proposed Problem

It is crucial to note that if the computational complexity of factoring a composite number *n* is sufficiently high, the

security of the proposed digital signature schemes relies on the difficulty of both IFP and DLP modulo a prime number.

The difficulty of factoring a composite number $n = p \cdot q$, depends on the strength of the prime factors p and q and is primarily determined by the size of the smaller prime (assuming $|q| \leq |p|$, where $|q|$, $|p|$ denotes the bit-length of q and p, respectively). The difficulty of DLP modulo a prime p is approximately equivalent to the difficulty of factoring n. Therefore, assuming $n = p \cdot q$, where $|p| = 2|q|$, the difficulty of DLP modulo n, the difficulty of DLP modulo p, and the difficulty of factoring the composite number n are all approximately equal.

As a result, the security of the proposed cryptosystems based on DLP remains unaffected even if an algorithm exists that can solve either (i) the DLP or (ii) the IFP. Since the probability of both (i) and (ii) occurring simultaneously is extremely low, the chances of successfully breaking the cryptosystems based on DLP modulo composite numbers are also minimal. This demonstrates the computational difficulty and security of DLP modulo composite numbers, thereby ensuring the security of the digital signature schemes relying on this hard problem.

B. Resistance of the CDS-0824 Scheme to Attacks

The proposed single digital signature scheme (DSS-0824) not only achieves a high level of security due to its foundation on the proposed DLP modulo composite numbers, but also inherits the security advantages and resistance to attacks of the Schnorr signature scheme, upon which it is based. Therefore, only the resistance to attacks of the proposed group digital signature scheme (CDS-0824) will be discussed.

The security of a group signature scheme is evaluated based on its resistance to two common types of attacks: key exposure attacks and forgery attacks. In the context of group signatures, members of the group are more likely to launch attacks against the scheme they participate in than external adversaries. Therefore, the current study will focus on the two types of attacks on group signatures that originate from within the group.

1) First Type of attack: Forging the m^{th} Signer's Signature

Suppose that $m-1$ signers in a group of m members want to create a group signature on a document *M*. In other words, this group of $m-1$ individuals aims to forge the signature of the remaining signer, referred to as the m^{th} signer, within the signing group.

The group's public key can be rewritten as:

$$Y = Y^* Y_m \mod n \tag{28}$$

where $Y^* = \prod_{i=1}^{m} Y_i \bmod n$ and Y_m is the public key of the m^{th} signer.

To successfully forge the signature, the group of $m-1$ signers must generate a pair of values (E^*, S^*), corresponding to a valid group signature and satisfying the verification equations:

$$R^* = Y^{-E^*} a^{S^*} \bmod n \text{ and } E^* = F_H(M \,\|\, R^*) \tag{29}$$

Suppose that a group of $m-1$ signers can 'compute' a valid collective signature (E^*, S^*), corresponding to the public key of the collective signer: $Y = \prod_{i=1}^{m} Y_i \bmod n$. This means that the pair (E^*, S^*) satisfies the verification expression R^*. So:

$$\begin{aligned} &R^* \equiv Y^{-E^*} \alpha^{S^*} mod\, n \equiv \left(Y^* Y_m\right)^{-E^*} \alpha^{S^*} \; mod \text{ n} \Rightarrow \\ &\Rightarrow R^* \equiv \left(Y^*\right)^{-E^*} Y_m{}^{-E^*} \alpha^{S^*} \equiv \\ &\alpha^{-E^* \sum_{i=1}^{m-1} x_i} Y_m{}^{-E^*} \alpha^{S^*} \Rightarrow \\ &\Rightarrow R^* \equiv Y_m{}^{-E^*} \alpha^{S^* - E^* \sum_{i=1}^{m-1} x_i} \; mod\, n \Rightarrow \\ &\Rightarrow R^* \equiv Y_m{}^{-E^*} \alpha^{S^{**}} \end{aligned} \tag{30}$$

where:

$$S^{**} = S^* - E^* \sum_{i=1}^{m-1} x_i \tag{31}$$

Thus, the group of $m-1$ signers, referred to as the forgery group, has successfully computed a valid signature (E^*, S^{**}) on document *M* for the m^{th} signer, since $E^* = F_H(M \,\|\, R^*)$ and the pair (E^*, S^{**}) passes the signature verification procedure of the DSS-0824 scheme, which is the underlying scheme of the group signature scheme CDS-0824.

Since the DSS-0824 scheme is based on the Schnorr signature scheme, compromising the security of this group signature scheme would inherently compromise the Schnorr signature scheme. However, given that the Schnorr signature scheme has been proven secure, this group signature scheme is also regarded as secure against forgery attacks.

2) Second Type of Attack: Recovering the Secret Key of the m^{th} Signer

Suppose that a group of $m-1$ signers, within a group of m members, shares a group signature (R, S) with the m^{th} signer and is attempting to compute the private key of this signer. Let's refer to this group of $m-1$ signers as the attacking group.

The attacking group is aware of the values R_m and S_m generated by the m^{th} signer. These values satisfy the verification equation $R_m = Y_m{}^{-E} \alpha^{S_m} mod\, n$. Notably, the values R_m and E are beyond the attackers' control (since R_m is computed as $R_m = \alpha^{k_m} mod\, n$, where k_m is a randomly

generated value by the m^{th} signer, and E is the output of a hash function).

Assuming that the hash function used in the protocol is sufficiently secure, the attacking group cannot select a value R that would allow them to generate a specific chosen value of E. This implies that, similar to the Schnorr signature scheme, to compute the private key of the m^{th} signer, the attackers would need to solve the discrete logarithm problem, either: i) by finding $k_m = log\, R_m$, and then computing $x_m = E^{-1}(S_m - k_m) mod\, \gamma$, or ii) by computing $x_m = log\, Y_m$. However, it is well-known that the DLP is computationally intractable.

In conclusion, the attacking group cannot compute the private key of the m^{th} signer due to the computational difficulty of the discrete logarithm problem. As a result, the group signature scheme remains secure against attacks aimed at obtaining the private key of an individual signer.

C. Performance of the Proposed Signature Scheme CDS-0824

The evaluation of the performance of the proposed signature schemes was performed by calculating the computational cost in terms of the time needed for generating a signature (the signature generation procedure) and verifying the validity of a signature (the signature verification procedure).

The underlying scheme DSS-0824 is designed in a way that closely mirrors the Schnorr signature scheme, which is known for its security and computational efficiency. Consequently, the computational cost of the underlying scheme is not addressed in this discussion.

Some conventions used in the formula for calculating the time cost of operations in the two processes of the collective digital signature system are:

- T_h: Time cost of the hashing operation on Z_p.
- T_{inv}: Time cost of the inversion operation on Z_p.
- T_e: Time cost of the exponentiation operation on Z_p.
- T_m: Time cost of the multiplication operation on Z_p.

The conversions are: $T_h \approx T_m$, $T_{inv} \approx 240T_m$ $T_e \approx 240T_m$.

The data displayed in Table I show that the time cost of the proposed scheme, CDS-0824, is only slightly reduced compared to the LD 1.02 scheme, as both are built based on the Schnorr signature standard. Thus, while the time cost of the CDS-0824 scheme does not increase, the security level and the signature size have been significantly improved. This, once again, highlights the advantages of the proposed DLP modulo composite numbers and the schemes built on its basis.

TABLE I. TIME COST OF THE CDS-0824 SCHEME COMPARED TO THE LD 1.02 SCHEME.

Scheme	**Time cost of procedure**	
	Signature generation	**Signature verification**
CDS-0824	Compute E: $(241m+1)T_m$ Compute R': $482mT_m$ Compute S:$2mT_m$	$241mT_m$+$482mT_m$
LD 1.02	Compute E: $(242m+1)T_m$ Compute R': $482mT_m$ Compute S:$2mT_m$	$241mT_m$+$482mT_m$

V. CONCLUSION

In this paper, it is demonstrated that the proposed solution for enhancing the security of the new digital signature scheme is feasible and meets expectations, with improvements being noted in both security level (128 bits) and signature length (512 bits). This solution is based on the Discrete Logarithm Problem (DLP) modulo a composite number. Thus, only one proposed hard problem is utilized to construct the digital signature scheme. Breaking these schemes, though, requires an attacker to solve two, or even three, hard problems simultaneously: the DLP modulo a prime number and the Integer Factorization Problem (IFP). The proposed single-signer digital signature scheme (DSS-0824) is constructed based on the Schnorr digital signature standard, thereby benefiting from the security advantages of this standard.

The proposed signature scheme (CDS-0824) is built on the DSS01 scheme, providing a strong guarantee of security and safety. The paper also highlights that the size of the system parameters is crucial to the security level of the scheme. For the scheme to achieve 128-bit security, the values of the parameters γ, γ', γ'', p, and q must be chosen as follows:

$|\gamma| \geq 256$ bits, $|\gamma'| \geq 128$ bits, $|\gamma''| \geq 128$ bits

$|p| \geq 2464$ bits and $|q| \geq 1532$ bits

This is considered a new aspect of the discrete logarithm modulo, a complex number problem that has been proposed.

The proposed hard problem was successfully used to construct digital signatures and collective digital signatures. This DLP modulo a composite number could help in building blind digital signature schemes, blind collective digital signature schemes, and representative collective digital signature schemes in the future. The quantum resistance issue of the proposed scheme will also be considered in the future [15]. If this succeeds, once again, the feasibility and practicality of the newly proposed hard problem will be validated.

CONFLICT OF INTEREST

The authors confirm that there are no conflicts of interest related to this study.

REFERENCES

[1] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Commun. ACM*, vol. 21, no. 2, pp. 120–126, Oct. 1978, https://doi.org/10.1145/359340.359342.

[2] J. Pieprzyk, T. Hardjono, and J. Seberry, *Fundamentals of Computer Security*. Berlin, New York: Springer, 2003.

[3] D. Wong, *Real-World Cryptography*. New York: Manning, 2021.

[4] E. S. Ismail, N. M. F. Tahat, and R. R. Ahmad, "A New Digital Signature Scheme Based on Factoring and Discrete Logarithms," *Journal of Mathematics and Statistics*, vol. 4, no. 4, pp. 222–225, Dec. 2008, https://doi.org/10.3844/jmssp.2008.222.225.

[5] N. M. F. Tahat, E. S. Ismail, and R. 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.

[6] J. He and T. Kiesler, "Enhancing the security of El Gamal's signature scheme," *IEE Proceedings - Computers and Digital Techniques*, vol. 141, no. 4, pp. 249–252, Jul. 1994.

[7] L. Harn, "Enhancing the security of El Gamal's signature scheme," *IEE Proceedings - Computers and Digital Techniques*, vol. 142, no. 5, pp. 376–376, Sep. 1995.

[8] N. Y. Lee, "Security of Shao's signature schemes based on factoring and discrete logarithms," *IEE Proceedings - Computers and Digital Techniques*, vol. 146, no. 2, pp. 119–121, Mar. 1999.

[9] S. Wei, "A New Digital Signature Scheme Based on Factoring and Discrete Logarithms," in *Progress on Cryptography: 25 Years of Cryptography in China*, K. Chen, Ed. Boston, MA: Springer US, 2004, pp. 107–111.

[10] S. F. Tzeng, Yang ,Cheng Ying, and M. S. and Hwang, "A new digital signature scheme based on factoring and discrete logarithms," *International Journal of Computer Mathematics*, vol. 81, no. 1, pp. 9–14, Jan. 2004, https://doi.org/10.1080/00207160310001614954.

[11] J. Pollard and C. Schnorr, "An efficient solution of the congruence x^2 + ky^2 = mpmodn," *IEEE Transactions on Information Theory*, vol. 33, no. 5, pp. 702–709, Sep. 1987, https://doi.org/10.1109/TIT.1987.1057350.

[12] S. Vishnoi and V. Shrivastava, "A new Digital Signature Algorithm based on Factorization and Discrete Logarithm problem," *International Journal of Computer Trends and Technology*, vol. 3, no. 4, pp. 653–657, 2012.

[13] N. Q. Luc, T. T. Nguyen, D. H. Quach, T. T. Dao, and N. T. Pham, "Building Applications and Developing Digital Signature Devices based on the Falcon Post-Quantum Digital Signature Scheme," *Engineering, Technology & Applied Science Research*, vol. 13, no. 2, pp. 10401–10406, Apr. 2023, https://doi.org/10.48084/etasr.5674.

[14] C. P. Schnorr, "Efficient signature generation by smart cards," *Journal of Cryptology*, vol. 4, no. 3, pp. 161–174, Jan. 1991, https://doi.org/10.1007/BF00196725.

[15] T. T. Nguyen, N. Q. Luc, and T. T. Dao, "Developing Secure Messaging Software using Post-Quantum Cryptography," *Engineering, Technology & Applied Science Research*, vol. 13, no. 6, pp. 12440–12445, Dec. 2023, https://doi.org/10.48084/etasr.6549.